TOWSON UNIVERSITY
OFFICE OF GRADUATE STUDIES

# PREDICTING THE TERMINAL BALLISTICS OF KINETIC ENERGY PROJECTILES USING ARTIFICIAL NEURAL NETWORKS

by

John Robert Auten Sr.

A Dissertation

Presented to the faculty of

Towson University

in partial fulfillment

of the requirements for the degree

Doctor of Science

Department of Computer & Information Sciences

Towson University
8000 York Rd.
Towson, MD. 21252

Date of Submission: December 7, 2016

**TOWSON UNIVERSITY**
**OFFICE OF GRADUATE STUDIES**

**DISSERTATION APPROVAL PAGE**

This is to certify that the dissertation prepared by **John R. Auten Sr** entitled **Predicting the Terminal Ballistics of Kinetic Energy Projectiles using Artificial Neural Networks** has been approved by the dissertation committee as satisfactorily completing the dissertation requirements for the degree **Doctor of Science in IT**.

Chairperson, Dissertation Committee: Robert J. Hammell II — 17 Nov 2016
Date

Committee Member: Josh Dehlinger — 11-17-16
Date

Committee Member: Jinjuan Feng — 11-17-16
Date

Committee Member: Sungchul Hong — Nov. 17, 2016
Date

Chairperson, Doctoral Program: Chao Lu — Nov. 17, 2016
Date

Janet V DeLany
Dean of Graduate Studies — 11/29/16
Date

# Acknowledgments

I would like to thank my advisor Dr. Robert J. Hammell II he has been a mentor, colleague, and friend. Without his guidance and encouragement, completion of this work would not have been possible. I would also like to thank my wife, Karen Lee Auten and son, John Robert Auten Jr. for their support and understanding.

# Abstract

Predicting the Terminal Ballistics of Kinetic Energy Rods Using Artificial Neural Networks

John Robert Auten Sr.

The U.S. Army requires the evaluation of new weapon and vehicle systems through the use of experimental testing and Modeling & Simulation (M&S). Traditional M&S has worked well over the years but can be a lengthy process and often cannot provide quick results for studies involving new threats encountered in theater. So, there is increased focus on rapid M&S efforts that can provide accurate and fast results.

Accurately modeling the penetration and residual properties of a ballistic threat as it progresses through a target is an extremely important part of determining the effectiveness of the threat against that target.

This dissertation presents research on the application of Artificial Neural Networks (ANNs) to the prediction of the terminal ballistics of Kinetic Energy Projectiles (KEPs). By shifting the computational complexity of the problem to the fitting (regression) phase of the methodology, performance during analyses are improved when compared to other terminal ballistic models for KEPs. Another improvement in performance can be realized by removing the need for input preparation by a Subject Matter Expert (SME) prior to using the methodology for an analysis.

This research shows that ANNs can be used to model the terminal ballistics of KEPs and that they are capable of being used for single element and multiple element targets. It is also shown that the runtimes of an ANN are drastically faster than the current state-of-the-art model.

# Contents

# List of Tables

# List of Figures

# Acronyms

| | |
|---|---|
| **ACAT** | Acquisition Category |
| **AIC** | Akaike Information Criterion |
| **AICc** | Akaike Information Criterion with Correction |
| **ANN** | Artificial Neural Network |
| **CFRP** | Carbon Fiber Reinforced Composite |
| **AP** | Armor Piercing |
| **APCBC** | Armor Piercing Capped Ballistic Cap |
| **APCNR** | Armor Piercing Composite Non-Rigid |
| **APCR** | Armor Piercing Composite Rigid |
| **APDS** | Armor Piercing Discarding Sabot |
| **APE** | Absolute Percentage Error |
| **APFSDS** | Armor Piercing Fin Stabilized Discarding Sabot |
| **APFSDS-T** | Armor Piercing Fin Stabilized Discarding Sabot-Tracer |
| **ARL** | Army Research Laboratory |
| **BAD** | Behind Armor Debris |
| **BVLD** | Ballistic Vulnerability/Lethality Division |
| **CAD** | Computer Aided Design |
| **DNN** | Deep Neural Network |
| **DoD** | Department of Defense |
| **DU** | Depleted Uranium |
| **EFP** | Explosively Formed Penetrator |
| **FEA** | Finite Element Analysis |
| **FSP** | Fragment Simulating Projectile |

| | |
|---|---|
| **GA** | Genetic Algorithm |
| **GPU** | Graphics Processing Unit |
| **GUI** | Graphical User Interface |
| **IUA** | Individual Unit Action |
| **JAGA** | Java API for Genetic Algorithms |
| **JAXB** | Java Architecture for XML Binding |
| **JGAP** | Java Genetic Algorithms Package |
| **JOONE** | Java Object Oriented Neural Engine |
| **KE** | Kinetic Energy |
| **KEP** | Kinetic Energy Projectile |
| **LM** | Levenberg-Marquart |
| **LFT&E** | Live-Fire Test & Evaluation |
| **MAPE** | Mean Absolute Percentage Error |
| **MBT&E** | Mission-Based Test and Evaluation |
| **MCC** | Matthews Correlation Coefficient |
| **MLP** | Multi-Layered Perceptron |
| **MPE** | Mean Percentage Error |
| **M&S** | Modeling & Simulation |
| **MSE** | Mean Squared Error |
| **NBC** | Nuclear, Biological, Chemical |
| **OCR** | Optical Character Recognition |
| **PDML** | Penetration Database Markup Language |
| **PE** | Percentage Error |
| **RBF** | Radial Basis Function |
| **RHA** | Rolled Homogeneous Armor |
| **RNN** | Recurrent Neural Network |
| **S4** | System-of-Systems Survivability Simulation |
| **SAPE** | Symmetric Absolute Percentage Error |
| **SCG** | Scaled Conjugate Gradient |

| | |
|---|---|
| **SCJ** | Shaped Charged Jet |
| **SD** | Stochastic Dominance |
| **SLAD** | Survivability/Lethality Analysis Directorate |
| **SMAPE** | Symmetric Mean Absolute Percentage Error |
| **SME** | Subject Matter Expert |
| **SoS** | System-of-Systems |
| **SPE** | Symmetric Percentage Error |
| **SVM** | Support Vector Machine |
| **SwRI** | Southwest Research Institute |
| **V/L** | Vulnerability/Lethality |
| **VSL** | Visual Simulation Lab |
| **WC** | Tungsten Carbide |
| **WHA** | Tungsten Heavy Alloy |
| **WMRD** | Weapons Material Research Directorate |
| **XML** | Extensible Markup Language |

# Chapter 1

# Introduction

## 1.1 Background

When a U.S. Soldier takes a weapon system into the field for the first time, that Soldier needs to know that the weapon system will perform as expected. In order to ensure that the U.S. Department of Defense (DoD) acquires systems that are safe, effective, and perform as expected; they test the system and use modeling and simulation to augment the results from the tests. The U.S. DoD requires that Acquisition Category (ACAT) I systems undergo Live-Fire Test & Evaluation (LFT&E) [1] to determine the Vulnerability/Lethality (V/L) of that system. V/L simulation models are validated to those live-fire tests and then accredited so that they can be used for future studies involving that system.

V/L simulation models are used to analyze the vulnerability of military systems against the lethality of weapons systems. These models provide information that is critical to protecting the lives of U.S. soldiers.. V/L model results are used as inputs in force-on-force models [2, 3]. U.S. Army force-on-force simulation models are used to simulate battlefield scenarios that provide the U.S. Army with situational outcomes for planning purposes and information that is used for making decisions about procurement in the acquisition life cycle.

Both force-on-force and V/L models contain ballistics sub-models of varying types and levels of fidelity. Ballistics is the science of mechanics that deals with the flight, behavior, and effects of projectiles, bombs, rockets, or shells [4]. The projectile, or the projectiles generated from a bomb or rocket warhead, could be further defined as fragments, Shaped Charged Jets (SCJs), Explosively Formed Penetrators (EFPs), or Kinetic Energy Projectiles (KEPs).

The field of ballistics is divided into several sub-fields: interior, intermediate, exterior, and terminal. Interior ballistics is the study of the propulsion of a projectile from the moment that the charge is ignited until the moment that it leaves the muzzle. Intermediate, also known as transitional ballistics, is the study of a projectile's behavior after it leaves the muzzle until the moment that the pressure behind the projectile is equalized. Exterior ballistics is the study of a projectile's flight after the propulsive forces are no longer acting on it. Terminal ballistics is the study of the impact of a projectile and another object [5,6]. There are simulation models used for all four of the sub-fields of ballistics, and they all vary in their levels of fidelity.

Force-on-force and V/L models are primarily concerned with modeling exterior and terminal ballistics. Terminal ballistics models are used to determine if a threat has perforated a particular target and what the residual capability of that threat is after perforation. It is the first step in determining the damage due to a target and threat interaction, and has an impact on all of the results that are based on that damage outcome. Force-on-force and V/L models typically use lower fidelity models because of the large scope of what they are modeling. V/L models are used for full vehicles against thousands of impacts by threats, so if the terminal ballistics model used is high fidelity, then typically it will take too long to run. Many force-on-force models use a simple lookup value method using averaged probabilities of kill for a given vehicle and threat pairing. Examples of such methods are the use of lethal areas and Individual Unit Actions (IUAs) [7].

A lethal area is a measure of a projectile's ability to incapacitate a target component and can be used to generate estimates of the projectile's ability to kill. Lethal areas take into account the delivery accuracy of the projectile and the vulnerable areas of the target. A vulnerable area ($A_V$) is the summation of the probability of killing a target over a given area in a particular attack direction to the target. Vulnerable areas are calculated by dividing the plane of impact in a particular direction into a grid of equally sized cells. For each cell, a probability of kill given a hit ($P_{k|h}$) is determined. The $P_{k|h}$ is multiplied by the area of the cell and then all the cells are summed together to generate the $A_V$. $A_V$s can then be used with the target presented area ($A_P$) to determine the $P_{k|h}$ for the target at a specific attack aspect [8].

IUAs are calculated for a particular aspect angle by summing up the likelihood of a given aim dispersion randomly hitting each grid cell combined with the loss-of-function associated with that grid cell. It represents the most probable loss-of-function for the target at that aspect angle against that particular threat [9]. When given to force-on-force models, many different aspect angles are provided to cover the many possible angles of attack. As with lethal areas, $P_{k|h}$ is an important part in calculating IUAs, because they are used in determining the loss-of-function values used by IUAs. In order to calculate each $P_{k|h}$, terminal ballistics must be modeled for the determination of how far a ballistic threat can penetrate into the interior of the target. The modeling of the terminal ballistics in many cases is handled by empirical or semi-empirical models that have been fit using a combination of experimental test data and Finite Element Analysis (FEA). In past efforts it has taken too long to run a more detailed V/L model to determine the kill of a vehicle in force-on-force models. If faster algorithms can be developed at the terminal ballistics level of modeling then it may be possible to speed up the V/L models and allow for their use in force-on-force models directly.

To summarize, terminal ballistics model results are rolled up into empirical models. The terminal ballistics empirical model results are rolled up into vulnerable areas using V/L models. The vulnerable area results are rolled up into lethal areas using effectiveness models. Lastly, lethal areas are rolled up into a distribution of battle field scenario results using force-on-force models. The fidelity of the inputs used in higher-level models is typically sacrificed for the speed of calculation needed at that higher level. This may be changing with the development of new models and methods in the U.S. Army Research Laboratory (ARL), Survivability/Lethality Analysis Directorate (SLAD), such as System-of-Systems Survivability Simulation (S4), MUVES-S2, and Visual Simulation Lab (VSL).

An agent-based modeling approach is used by the S4 to model the emergent behavior of the System-of-Systems (SoS) on the battlefield. The intent is to model the survivability of U.S. Army systems in the mission context [10]. The S4 currently uses $P_{k|h}$ lookup tables to determine if a target vehicle has been eliminated from the battlefield. There have been attempts in the past to provide a V/L service that can run target and threat interactions on-the-fly when S4 needs an interactions result. The most recent plan was to use the next generation of V/L simulation that was being developed by the U.S. ARL, SLAD, called MUVES 3. MUVES 3 was to serve as the primary code used within the SLAD Ballistic Vulnerability/Lethality Division (BVLD) to conduct V/L analyses. MUVES 3 was designed to be an integrated, collaborative work environment that would provide a wide range of metrics to support Mission-Based Test and Evaluation (MBT&E) of networked SoS. Those metrics could be provided to force-on-force models through the use of a V/L service. MUVES 3 was to provide robust, on-demand V/L estimates to the S4 and other force-level models [11]. However, the MUVES 3 program was shutdown in 2013 and the proposed solutions are to be eventually migrated to the current V/L model, MUVES-S2.

There is still a strong desire within the U.S. Army to have high fidelity results within force-on-force models such as S4. When that direct linkage happens, it will be possible to provide more accurate results to the force-on-force model through the V/L model, but the speed of the V/L model and its sub-models will still be a concern. If a sub-model in a V/L model can improve on its speed of calculation while maintaining or improving its accuracy, then it will have a positive impact on the overall speed of the V/L model and the performance of the combined force-on-force and V/L models that could be using it. The aim of this research was to develop a terminal ballistics sub-model that would provide an improvement in runtime while maintaining or improving accuracy, when compared to the current state-of-the-art model.

In addition to enabling a linkage of force-on-force and V/L models, such a model would also benefit the VSL tool. VSL is being developed in response to the need for a tool that can provide a rapid analytical response to problems that arise in the battlefield and that can support LFT&E test shot selections. The development of VSL started off as a dissertation research project showcasing the capability of processing Computer Aided Design (CAD) model ray-tracing and V/L analysis on Graphics Processing Unit (GPU) cores. The dissertation project was never completed, but the research was transitioned over to the U.S. ARL, SLAD. VSL provides real-time manipulation and display of 3-dimensional target CAD models, with limited V/L analysis capability. The current V/L capability of VSL is limited to a terminal ballistic empirical model used for SCJs and a simple damage model that calculates if a critical component has been perforated or hit [12]. An accurate and fast terminal ballistic model for KEPs is needed by VSL and the aim of this research was to provide a sub-model that can meet that need.

## 1.2 Problem Definition

Simulation modeling, V/L modeling in particular, is consistently battling with the trade-off of speed and accuracy. Recent trends have required quick turn-around analyses to support protecting U.S. soldiers in the battlefield, but the accuracy of the models is still of importance. As a component of the V/L modeling process, the speed and accuracy of terminal ballistics models are important. Two of the primary drivers of time for the models are preparation of model inputs and model runtime. The problem is to find a model that requires very little subject matter expertise for preparation, runs very fast, and is still accurate. This research developed a model for predicting the terminal ballistics of KEPs that can be used in a V/L model where speed and accuracy are both of importance.

## 1.3 Objectives

The objective of this research was to develop an accurate and generalized Artificial Neural Network (ANN) based terminal ballistics model for KEPs that is usable in a V/L modeling environment and provides improvements in speed while maintaining or improving accuracy.

## 1.4 Research Questions

There are numerous terminal ballistics models available to the ballistician; however, they typically require too much time for preparation, have relatively long run times, or are not accurate. The primary research question to be answered by this study was:

*RQ1: Can an ANN, used to model the terminal ballistics of KEPs, be developed that is both fast, accurate, and generalized?*

The question was further broken down into sub-questions that addressed components of the primary research question:

**RQ1.1:** *What target or threat parameters have the most influence on terminal ballistics results?*

The determination of which parameters are important to the process of penetration is important because it will directly influence the overall design of any ANNs that will be developed.

**RQ1.2:** *Can an ANN be used to produce a generalized, accurate model of the terminal ballistics of a KEP against monolithic metallic targets?*

This question is important because it is the first step in developing an expanded capability for modeling real world target and threat interactions. The modeling of penetration through a single plate of metal armor should not pose as many problems as a complex target will.

**RQ1.3:** *Can the ANN from RQ1.2 be used to model the terminal ballistics of a KEP against multi-element metallic targets?*

This question gets to the determination of usability in V/L models. Real world targets have complex armor packages made of different metals and air gaps.

**RQ1.4:** *How does the <u>speed</u> (execution time and elapsed time) of an ANN based terminal ballistics model compare to the Segletes hybrid model of the Frank-Zook and Walker-Anderson models?*

This question gets to the speed of the model and looks at two ways of measuring it. The execution time will address the speed of the algorithm itself and the elapsed time will address the overall speed of the models.

**RQ1.5:** *How does the accuracy of an ANN based terminal ballistics model compare to the Segletes hybrid model of the Frank-Zook and Walker-Anderson models?*

This question addresses the issue of accuracy of the model and how it compares to other models currently in use.

## 1.5 Approach

The approach of this research was broken into four interdependent phases. The first phase served to build a strong base of training data for use in the development of the ANN. The following two phases were for iterative development of the ANN. The first of the two was the simplest and the second one pushed the envelope a little farther by building on the work from the previous phase. The last phase compared the performance of the ANN and the best performing model that is currently used for modeling the terminal ballistics of KEPs. Table 1.1 provides a breakdown of which research questions are answered by which phases of this research.

Table 1.1: Phases of approach

| Phase | Research Questions Addressed |
|---|---|
| 1 | RQ1.1 |
| 2 | RQ1.2 |
| 3 | RQ1.3 |
| 4 | RQ1.4, RQ1.5 |

### 1.5.1 Phases of Approach

#### 1.5.1.1 Phase 1: Collect and Document Experimental Test Data

This phase consisted of collecting all currently available test data and designing and implementing a method of storage. Extensible Markup Language (XML) was used to store the terminal ballistics test data for this research [13]. XML was used due to its simplicity and ability to be accessed using Java, since the ANN was implemented using Java.

#### 1.5.1.2 Phase 2: Implementation of an ANN to Model KEPs Against Monolithic Metallic Armor

This phase consisted of designing and implementing an ANN that was capable of modeling KEPs impacting monolithic armor. For this phase the ANN was limited to a single element and only metallic materials. There are two primary types of test data for this scenario; the first is an impact into a semi-infinite block of armor and the second is an impact into a finite element of armor. The first type will typically contain results pertaining to depth of penetration and possibly crater diameter and volume. The second type will typically contain results pertaining to residual velocity, residual mass, residual length, and possibly hole size.

#### 1.5.1.3 Phase 3: Implementation of an ANN to Model KEPs Against Multi-Layered Metallic Armor

This phase built on the work from the previous phase and added in the complexity of having more than one element in the target array. As a KEP penetrates through a target array it forms a zone of plastic deformation in front of it. As the KEP is penetrating, the material in front of it is compressing out to a certain distance from the interface of the target and penetrator. If the plastic zone extends across a plate boundary into another plate of a different material, the rod will sense the resistance of current remaining material in the plate it is in, but also the material from the other plate that is within the plastic zone (see Figure 1.1). This adds a significant complication to modeling penetration through a multi-element target. The method of iteratively applying the ANN designed in the previous phase to each element was used in this research and was found to be effective.

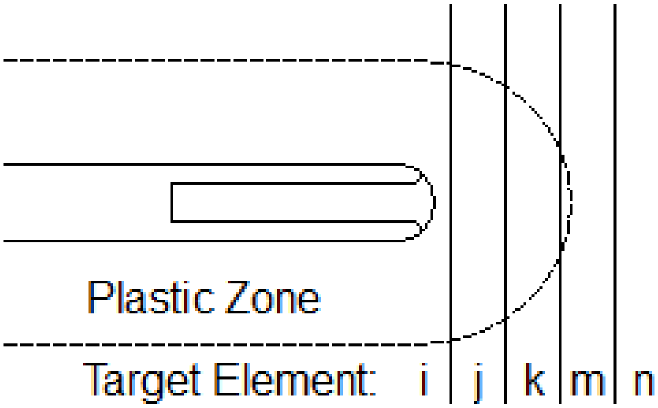


Figure 1.1: Rod plastic zone entrainment in multiple elements [14]

#### 1.5.1.4 Phase 4: Comparison of ANN Model to Segletes Hybrid Rod Model

This phase compared the speed and accuracy of the developed model to a model that is currently in use in a V/L model. The comparison of the two models examined not only the accuracy and run-times, but also the time required for preparation of inputs for the models. This comparison provided a clearer understanding of how successful the development of the ANN was.

### 1.5.2 Methods and Tools

Phases two and three involved an iterative approach to development; the following steps were performed in each of those phases [15]:

1. Data Preparation

   The data were prepared for use in the ANN; this included the process of omitting correlated inputs and normalizing of the inputs and outputs.

2. Design

   Using the equations for approximating the number of hidden layers and hidden nodes, initial designs were generated.

3. Build and Test Prototype Designs

   The prototype designs were constructed using the chosen software and were prepared for training. They were trained using k-fold cross-validation methodology. The prototype design with the best k-fold cross-validation error was selected for the final training process.

4. Train and Validate Selected Prototype

   The selected prototype was trained using the methods chosen during the design step. A cross-validation set of data was used to stop training when overfitting was identified.

5. Optimimize Prototype

   If the prototype was able to converge on a solution and the error was acceptable then testing proceeded, otherwise the design step was revisited.

6. Test Prototype

   During this step the trained prototype was tested against data that was not used for training to determine how well it interpolated and extrapolated. If the error was found to be acceptable then the proceeded, otherwise the design step was revisited.

7. Analyze Performance

   During this step the run-time and accuracy of the ANN was analyzed over the entire data set and compared to the current state-of-the-art model (the Segletes model).

8. Document Results

   After each phase the results of the design, development, and analysis of the ANN were documented.

The implementation of the ANN was done using Java, primarily because there were a lot of open-source libraries available for use that could be leveraged for completion of this research. After an initial search three possible libraries were found for implementing ANNs; the first was the Java Object Oriented Neural Engine (JOONE), the second was Neuroph, and the third was Encog. For the implementation of a Genetic Algorithm (GA), two possible libraries were found; the first was Java Genetic Algorithms Package (JGAP) and the second was Java API for Genetic Algorithms (JAGA).

## 1.6 Chapter Summary

In summary, this dissertation developed an ANN to model KEPs against single element targets and multiple element targets. The first phase of this research was to collect and prepare the data and the following two phases were to implement the ANN for single element targets and then multiple element targets. The last phase was to compare the performance of the ANN against the model that is the current standard for modeling KEPs, the Segletes model.

The work performed in the first phase, to collect and prepare the data for the ANN, will have further impact at the U.S. ARL. The database will be available to other researchers, within the U.S. ARL, to use for their research projects.

The ANN developed in this dissertation will provide other simulation models within the U.S. ARL with a fast running and accurate model for predicting the terminal ballistics of KEPs. The improvements provided by using the ANN, will help enable the use of V/L simulation models in larger force-on-force models, providing better vulnerability information to the decision makers of the U.S. Army which is critical to the survivability of our combat forces.

## 1.7 Roadmap

This paper is organized into seven chapters, this chapter being the first. The second chapter will provide background information into all of the key areas of this research, to include: V/L modeling, terminal ballistics, ANNs, data quality issues, analysis methods, and the like. The third chapter will provide an overview of the software that was developed and used for this research. The fourth chapter will outline the effort that went into working on the experimental test data for this research. The fifth chapter will provide details on the results from developing a Multi-Layered Perceptron (MLP) for monolithic metallic armor. The sixth chapter will provide details on the results from applying the developed MLP against multi-element metallic armor. The seventh and final chapter will provide a summary of the benefits and results from this research effort.

# Chapter 2

# Literature Review

## 2.1 Vulnerability/Lethality (V/L) Modeling

Vulnerability/Lethality (V/L) simulation models are used to analyze the vulnerability of military systems against the lethality of weapons systems. V/L models typically consist of a Computer Aided Design (CAD) model (an example of a CAD model is shown in Figure 2.1) of the target system, engineering definitions for the systems and sub-systems in the target, engineering inputs for the probability of component dysfunction given a hit ($P_{cd|h}$) for the target critical components, methodologies for determining system capabilities after a ballistic event, and algorithms for modeling the physical interaction of the target and the ballistic threat. For the purposes of this study, the ballistics of the physical interaction of the threat and the target are of interest.

The V/L taxonomy is used to rationalize the process that occurs to a system during a ballistic impact event. There are five levels in the taxonomy and each level represents the state of the system at discrete moments during the ballistic event. The levels are mapped together by operators that describe the transition from one state to the next. The five levels of the taxonomy are listed below [3]:

**Level 0** Threat-Launch Initial Conditions

This describes the initial conditions of the target prior to being fired upon.

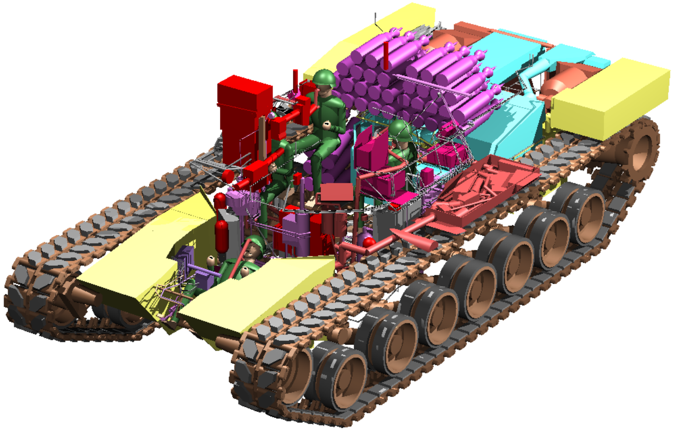


Figure 2.1: A CAD target model [16]

**Level 1** Threat-Target Interaction Initial Conditions

This describes the initial conditions of the target and the threat prior to the threat interacting with the target.

**Level 2** Target Damaged Components

This describes what components of the target have been damaged due to the interaction with the threat.

**Level 3** Target Measures-of-Capability

This describes the capability of the target given the damage inflicted upon the critical components.

**Level 4** Target Measures-of-Effectiveness (Utility)

This describes how the target capability effects the mission.

Figure 2.2 shows how the V/L taxonomy operators map from levels 1 through 4. Of particular interest is the operator that maps “Level 1” to “Level 2” in the taxonomy. That operator ($O_{1,2}$) defines how the threat physically interacts with the target to generate

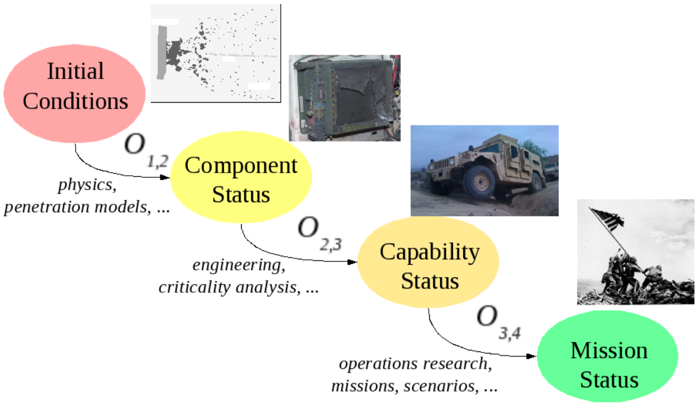

Initial Conditions
O 1,2
physics, penetration models, ...
Component Status
O 2,3
engineering, criticality analysis, ...
Capability Status
O 3,4
operations research, missions, scenarios, ...
Mission Status

Figure 2.2: V/L taxonomy (Level 1 to Level 4)

damage on the target components. In V/L simulations the interaction of the target and threat is modeled as a shotline going through the target. Operator $O_{1,2}$ can consist of one or many shotlines depending on the threat of interest. If the threat is a fragment, it could fracture upon impact and separate into several shotlines of smaller fragments. Another example would be a Shaped Charged Jet (SCJ) that impacts armor and generates Behind Armor Debris (BAD) which could be thousands of fragments with each requiring its own shotline. A single interaction could require many shotlines to fully analyze the ballistic event.

Imagine a main battle tank with rough dimensions of 2.4 m height, 3.7 m width, and 9.8 m length. Breaking the presented area of the side of that tank into cells that are 50 mm x 50 mm could result in 7611 cells total. Doing the same for the front of the tank would result in roughly 2854 total cells. Those two sets of cells are called views; averaging them would give a rough approximation of how many cells to expect for any particular view. For this simple example, a set of 26 views will be used. The azimuths of those views will start at 0 ° and increment by 45 ° until 315 °. Each of those azimuths will be combined with an elevation from the following, -45 °, 0 °, and 45 °. The remaining two views consist of an azimuth and elevation of (0 °, -90 °) and (0 °, 90 °). Since the simulation is stochastic, sampling is done within each cell of a view. So, in each cell the simulation could make 10 sample runs and in addition, each of those runs may need to call the penetration model 10 times. A total of 26 views with 5233 cells per view, 10 samples per cell, and 10 model calls per sample amounts to 13 605 800 total calls to the terminal ballistics model for this sample scenario run [7,8]. For each shotline in an analysis, system capability is determined based on which components are damaged. Before that determination can be made, the model must determine if the components were hit. Determination of a hit on a component is performed by calculating how far the threat can penetrate into the target on the shotline.

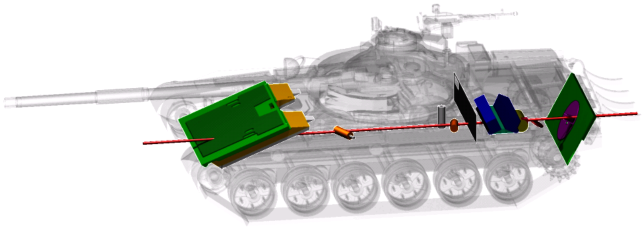


Figure 2.3: A shotline through a target [17]

An example of a shotline going through a vehicle can be seen in Figure 2.3. The components that intersect with the shotline are considered “threatened” and are highlighted in the figure. They are, in order from front to back; glacis armor, armor piercing round, armor piercing round, high explosive round, firewall, engine, starter, transmission sump, fan, and rear armor. How far along the shotline the threat can penetrate will determine which “threatened” components are actually hit. For example, the threat may be able to perforate everything along the shotline until it reaches the engine block, therefore everything after the engine block would not be hit.

Terminal ballistics models, also known as penetration models, are used to determine how far a projectile travels on a shotline. Once the distance traveled is known, the critical components that were hit by the projectile are also known. For all of the shotlines, the penetration model needs to accurately determine the penetration of the threat for each impact along a shotline. Due to the large number of shotlines and the need for accuracy, the calculation speed and accuracy of a penetration model are important.

On a particular shotline there can be many objects in the path of the projectile, so if the projectile perforates after impacting the first object on the shotline it may impact another object. The projectile may encounter armor plating, structural material, target components, ammunition, or personnel and for each of them a terminal ballistics model is applied to determine if the projectile will perforate the object or be defeated [2]. The first impact event will use the initial “Level 1” inputs for the terminal ballistics model, but each time a terminal ballistics model is run for subsequent impact events along the shotline, the results from the previous impact are used as inputs.

Some of the results from the terminal ballistics model may be used to determine the damage on a critical component in the target. Typically the damage or dysfunction to a critical component is determined using empirical models based on mass and velocity, hole size (typically a function of projectile diameter), or energy deposited (a function of mass and velocity). For each of those cases the residual parameters of the projectile after impact are needed for determination of damage [2].

Not only is it important to be accurate in predicting perforation of components in the target, but it is also important to be accurate in predicting the projectile's residual parameters since they are important to determining the damage inflicted to the target and residual penetration capability.

## 2.2 Terminal Ballistics Modeling

Terminal ballistics models fall into three general categories [2]:

1. Empirical

   Models that are derived from experimentation and observation rather than theory. These models are typically simple functions that allow statistically good fits to data.

2. Analytical/Phenomenological

   Analytical models are typically simple, closed-form phenomenological models. Phenomenological models use basic physical principles and basic material properties to simulate physical events. Some of these models could be considered semi-empirical if the closed-form model by itself is inadequate to describe the phenomena without parametric fitting.

3. Numerical/Phenomenological

   Numerical models are typically complex phenomenological models. These types of models are usually based on numerical approximations to the partial differential equations of fluid mechanics.

Each type of model has a place in a terminal ballistics modeling “toolbox”. Which type of model to employ usually involves a trade-off of speed and accuracy. Numerical models are normally very accurate, but require too much time and computational power for use in V/L analyses. They are however, sometimes used to augment experimental test data for fitting of empirical and semi-empirical models that are used in V/L analyses [2]. Analytical models are normally built using assumptions about the physical processes that occur during the penetration process. The physics-based equations that form the assumptions for the analytical model are solved for, if possible, and are modeled using numerical integration if needed. Analytical models are much faster than numerical models but are generally harder to design effectively and require support from Subject Matter Experts (SMEs) in order to run them accurately [2]. Empirical models are typically the simplest and therefore fastest of the three but are normally the least accurate when trying to interpolate or extrapolate from the data used for fitting. These models also require SME support for any applications outside of the initial fitting. Regardless of which model is used, each implementation of them is typically specific to a threat and target interaction and therefore not a generalized solution.

## 2.3 Kinetic Energy Projectiles

The term Kinetic Energy Projectile (KEP) is generally applied to an Armor Piercing (AP) projectile fired from a high-velocity rifle or cannon [6]. Figure 2.4 illustrates several categories of KEP rounds; Armor Piercing Capped Ballistic Cap (APCBC), Armor Piercing Composite Rigid (APCR), Armor Piercing Composite Non-Rigid (APCNR), Armor Piercing Discarding Sabot (APDS), and Armor Piercing Fin Stabilized Discarding Sabot (APFSDS).

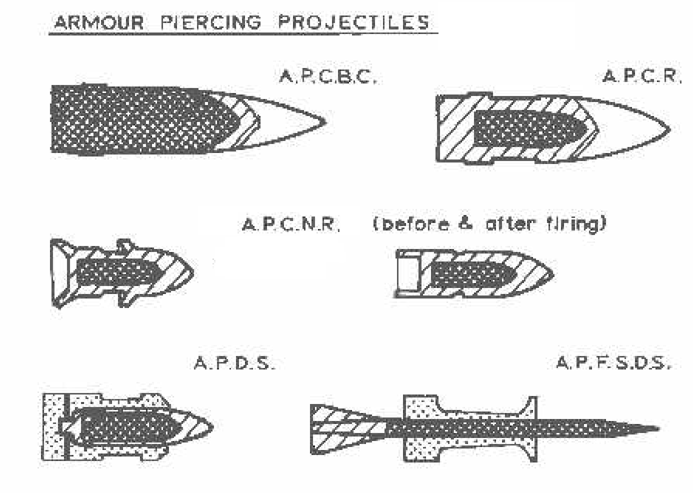


Figure 2.4: Types of AP rounds [18]

The data used for this research consisted of various types of KEPs, but the most prevalent type in the database was APFSDS. This is advantageous, because most modern large caliber Kinetic Energy (KE) rounds are APFSDS rounds. What follows is a brief overview of the APFSDS round.

The primary parts of an APFSDS round are the penetrator, ballistic cap, sabot, and fins (See Figure 2.5). If the round also has a tracer component then it is designated as an Armor Piercing Fin Stabilized Discarding Sabot-Tracer (APFSDS-T) round. The penetrator core or rod of an APFSDS round is typically made of Depleted Uranium (DU) alloy, high-strength steel, Tungsten Carbide (WC) composite, or Tungsten Heavy Alloy (WHA) [6]. The length of the penetrator is typically less than the length of the overall round; there are two primary reasons for the difference. First, the fins typically extend past the end of the penetrator slightly for APFSDS rounds and a distance of about one penetrator diameter for APFSDS-T rounds. Second, in many designs the penetrator is shorter then the tip of the ballistic cap, this means that the cap is normally hollow. Some experts propose an estimated working length for the penetrator equal to the round length minus two times the penetrator diameter [19]. The ballistic cap or windscreen is typically placed on the nose of the round to provide a more streamlined shape for better aerodynamic characteristics. Normally the ballistic cap will have very little influence on the penetration of the projectile; however, it could effect the penetration if a strong material is used.

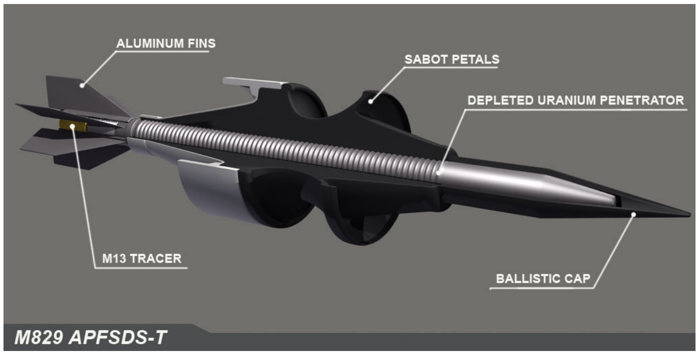


Figure 2.5: An APFSDS round [20]

The penetrator is normally much smaller than the diameter of the gun bore, so something is needed to seal the propulsive gases behind the penetrator to propel it down the barrel. The sabot holds the penetrator by interfacing with ribs along the length of the penetrator and filling in the “windage” (the gap between the projectile and the barrel). The sabot is typically composed of two or three “petals” (sections) that break away from the penetrator after exiting the gun barrel. APFSDS rounds are fired from smoothbore guns and therefore do not spin due to rifling like APDS rounds. APFSDS rounds use fins made from light weight materials like aluminum or titanium to provide stabilization during flight. For penetrators that have a high aspect ratio (length divided by diameter) there is typically a “jacket” of a lighter material, like steel or titanium, used to help absorb some of the elastic bending vibrations present during flight. The jacket material also helps keep the penetrator core from fracturing during the penetration of targets with air gaps [19].

### 2.3.1 Kinetic Energy Projectile Penetration Process

There are some penetrator types that rely on stored chemical energy to provide the energy required for penetration, such as SCJs and Explosively Formed Penetrators (EFPs). KEPs do not take any stored energy with them after launch; all of their energy comes from the KE they have from being launched from a weapon system. KE is the movement energy of an object and can be calculated using equation 2.1.

$$E_k = {}^1\!/_2 \cdot m \cdot v^2 \tag{2.1}$$

Where:

$E_k$ is the kinetic energy (J)

$m$ is the mass (kg)

$v$ is the velocity ($^{m}/_{s}$)

"The penetration of armor by a KE projectile converts the projectile's energy into plastic work (done to the target and penetrator material), usually in a process where the projectile is inverted and eroded away while it opens and burrows a cavity in the armor material" [6]. Figure 2.6 provides a visual example of how two different materials behave during the penetration process. The illustration on the left is an illustration of the erosion process for a WHA projectile and the illustration on the right is for a DU projectile. As the projectile starts the penetration process, both the target (not illustrated) and the KEP start a plastic deformation process. Both the target and projectile materials begin to backflow behind the interface of the penetration. For WHA projectiles there is a larger penetration tunnel due to the material shearing late in the backflow process. For DU projectiles the shearing of the material happens earlier in the the backflow process and leads to two important effects. The first is a smaller diameter penetration tunnel and the second is a "sharpening" of the projectile throughout the penetration process. Those two effects lead to better performance for DU than a similar mass WHA.

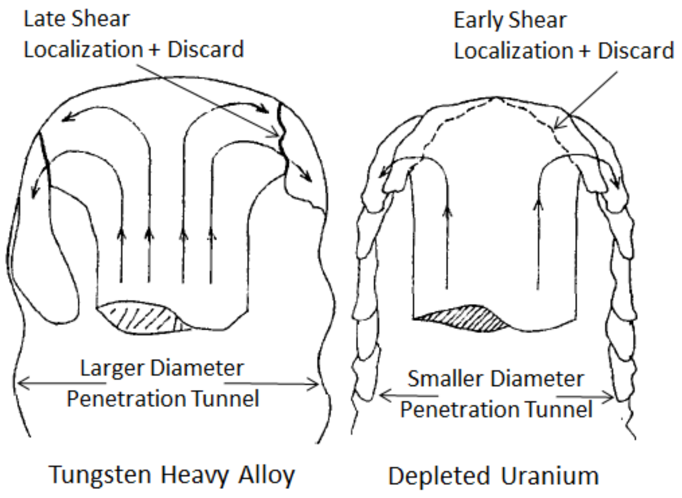


Figure 2.6: Example of erosion during penetration of Tungsten Heavy Alloy and Depleted Uranium [21]

The erosion of the rod and the target, as well as the backflow of those materials can be seen in Figure 2.7. The figure is of a 2D image from a Eulerian ALE3D simulation of a tungsten rod impacting an Rolled Homogeneous Armor (RHA) target (image and information provided by Steve Schraml of the U.S. Army Research Laboratory (ARL)).

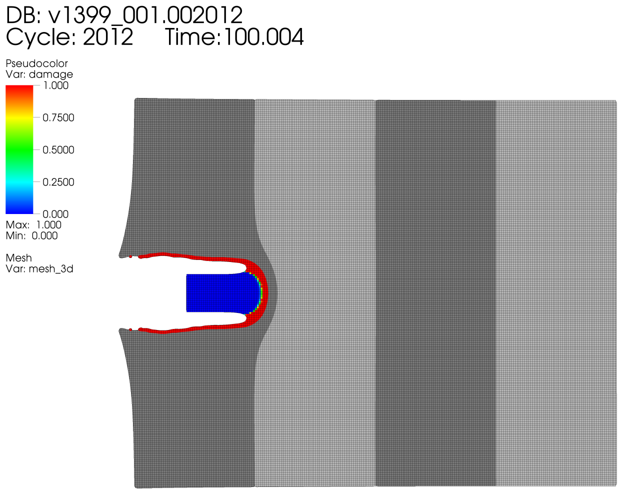


Figure 2.7: Finite element analysis of KEP

## 2.4 Kinetic Energy Projectile Terminal Ballistics Models

As mentioned in the previous section, there are three primary types of models that are typically used for modeling terminal ballistics. In the following sections an overview of several models that are currently used for modeling the terminal ballistics of KEPs are provided. Due to the need for use in V/L simulations, these models must have fast runtimes. Therefore, no numerical/phenomenological (i.e. hydrocode, Finite Element Analysis (FEA), and the like) models are given; all of the models provided are either empirical or analytical/phenomenological.

## 2.4.1 Empirical Models

### 2.4.1.1 Lanz-Odermatt Model

The Lanz-Odermatt model is an empirical model designed for modeling tungsten, depleted uranium, and steel long rod penetrators. The model has 3 coefficients of fit for tungsten, 3 for depleted uranium, and 4 for steel. There are an additional 3 coefficients of fit that are material independent. The working length ($L_w$) of the penetrator is calculated by converting the mass of the nose into an equal mass cylinder and adding that length to the length of the penetrator. For a cylindrical rod the working length of the rod is equal to the length of the rod. The equations of the Lanz-Odermatt model are provided in equations 2.2, 2.3, and 2.4 [22–24].

$$\frac{P}{L_w} = a \cdot \frac{1}{\tanh\left(b_0 + b_1 \cdot \frac{L_w}{D}\right)} \cdot \cos^m \theta \cdot \sqrt{\left(\frac{\rho_P}{\rho_T}\right)} \cdot e^{\frac{-s^2}{v_T^2}} \tag{2.2}$$

$$s = \begin{cases} \sqrt{\frac{(c_0 + c_1 \cdot BHN_T) \cdot BHN_T}{\rho_P}} & \text{if penetrator is tungsten or depleted uranium} \\ \sqrt{\frac{(c_0 \cdot BHN_T^k) \cdot BHN_P^n}{\rho_P}} & \text{if penetrator is steel} \end{cases} \tag{2.3}$$

$$\begin{aligned} \frac{v_R}{v_T} &= 1 + c_2 \cdot \ln\left(1 - {}^{d}\!/\!{}_{d_{lim}}\right) \\ \frac{L_R}{L_w} &= 1 - (1 - c_3) - \left({}^{d}\!/\!{}_{d_{lim}}\right) - c_3 - \left({}^{d}\!/\!{}_{d_{lim}}\right)^2 \end{aligned} \tag{2.4}$$

Where:

| | |
|---|---|
| $a$ | is a coefficient of fit |
| $b_0$ | is a coefficient of fit |
| $b_1$ | is a coefficient of fit |
| $BHN_P$ | is the hardness of the penetrator (BHN) |
| $BHN_T$ | is the hardness of the target (BHN) |
| $c_0$ | is a coefficient of fit |
| $c_1$ | is a coefficient of fit |

$c_2$ is a coefficient of fit

$c_3$ is a coefficient of fit

$d$ is the thickness of the target (mm)

$D$ is the penetrator diameter (mm)

$d_{lim}$ is the limit thickness of penetrator (mm)

$k$ is a coefficient of fit

$L_R$ is the residual length (mm)

$L_w$ is the working length (mm)

$m$ is a coefficient of fit

$n$ is a coefficient of fit

$P$ is the penetration channel line of sight length (mm)

$\rho_P$ is the penetrator density $({}^{\mathrm{kg}}/_{\mathrm{m}^3})$

$\rho_T$ is the target density $({}^{\mathrm{kg}}/_{\mathrm{m}^3})$

$s$ is a coefficient of fit

$\theta$ is the angle of obliquity (°)

$v_R$ is the residual velocity $({}^{\mathrm{km}}/_{\mathrm{s}})$

$v_T$ is the impact velocity $({}^{\mathrm{km}}/_{\mathrm{s}})$

### 2.4.1.2 Konrad Frank Version of Lanz-Odermatt Model

One of the simplest empirical models used for KEPs is the Konrad Frank modified version of the Lanz-Odermatt model [25]. The modified model equation (see 2.5) is fit to the independent variable of $v_s$ and the dependent variable ${}^{P_L}/_{L}$. This model is fit to penetration data from tests into semi-infinite targets and can be used to estimate the penetration of a KEP at a given velocity. Saucier [26] documented the process for applying this model iteratively for multi-element targets and derived the equations for residual velocity (see 2.6) and residual length (see 2.7).

$$\frac{P_L}{L} = a\, e^{-\left(\frac{b}{v_s}\right)^2} \tag{2.5}$$

$$v_r = v_s \left[ 1 - (1-c) \frac{v_s^2}{b^2} \ln \left( 1 - \frac{P}{P_L} \right) \right]^{-1/2} \tag{2.6}$$

$$L_r = L_s \left( 1 - \frac{P}{P_L} \right)^c \tag{2.7}$$

Where:

$a$ is a coefficient of fit

$b$ is a coefficient of fit

$c$ is a coefficient of fit

$L_r$ is the penetrator residual length

$L_s$ is the penetrator striking length

$P$ is the penetration

$P_L$ is the penetration limit

$v_r$ is the residual velocity

$v_s$ is the striking velocity

The Konrad Frank modified version of the Lanz-Odermatt model is very effective at predicting the penetration of KEPs against semi-infinite targets. However, it does not do as well against multi-element targets and has difficulty predicting the residual velocity and mass after a perforation of the target [13]. The model is also not generalized, it is fit to a specific target-threat pairing and the coefficients of fit are only good for that pairing. This means that when a new pairing needs to be modeled, new tests need to be performed to provide data for fitting.

### 2.4.2 Phenomenological Models

The phenomenological models used for KEPs are all based on a modified version of Bernoulli's incompressible fluid flow equation (2.8) [27].

$$\frac{1}{2}\,\rho_p\,(V-U)^2 = \frac{1}{2}\,\rho_t\,U^2 \tag{2.8}$$

Where, the density of the penetrator is represented by $\rho_p$ and the density of the target material is $\rho_t$. $V$ is the velocity of the rear of the penetrator and $U$ is the penetration velocity or the velocity of the front of the penetrator. Equation (2.8) defines the balance of forces at the penetrator-target interface and is used to determine penetration. It assumes that the impact velocities are very high, so that the strengths of the penetrator and target are negligible compared to the hydrodynamic or inertial forces [6].

The Bernoulli equation has been modified further and expanded over the years and as the equations in the phenomenological models have grown more complex they have also become more difficult to solve. This has led to the necessary use of numerical integration to solve for the equations and an increase in time to run the models. What follows in the next five sections is a brief history of the modification and expansion of the Bernoulli equation for use in modeling KEPs.

#### 2.4.2.1 Alekseevski-Tate Model

The modified Bernoulli equation was extended independently by Alekseevskii [28] and Tate [29] to account for material strengths. The Alekseevskii-Tate model incorporates material strength effects by adding two new terms to (2.8) and is shown in (2.9). The first term, $Y$ represents the strength of the penetrator material, and the second term, $R$ represents the strength of the target material [6].

$$\frac{1}{2}\rho_p (V-U)^2 + Y = \frac{1}{2}\rho_t U^2 + R \tag{2.9}$$

The Alekseevskii-Tate model is useful for roughly approximating depth of penetration, but it tends to under-predict penetration because it assumes a uniform resistance through the target [13].

#### 2.4.2.2 Frank-Zook Model

Another modification was proposed in 1991 by Frank and Zook [30] to make the resistance of the target a function of where in the target the interface was located. This modification allowed the target to have an initial resistance value that either increased or decreased as penetration proceeded into the target. As the interface approached the rear of the target, the resistance would either approach zero or that of the initial resistance of the next target element. The model also incorporates an equation for the penetrator strength as a function of the ratio of current length and initial length. The new model works very well if the elements in the target array are of significant thickness; in other words, if the plastic zone of deformation in front of the penetrator is confined to the current element and the next element in the array. This is because the Frank-Zook model only looks at the resistance of the current and next elements, therefore it is incapable of accounting for any resistance after the next element [13].

#### 2.4.2.3 FATEPEN Model

FATEPEN is a computer code developed by Applied Research Associates, Inc. (ARA) for multiple sequential penetration calculations of complex targets comprised of spaced elements. As such, FATEPEN penetration calculations are focused on predicting post-perforation penetrator mass, velocity, and angular momentum vector for use in determining the encounter conditions for the next impact. FATEPEN is comprised of a collection of closed-form, analytical/empirical engineering models, for ideal penetrator and target impact geometries (e.g., normal, unyawed impacts by compact or elongated penetrators, thin-plate/thick-plate penetration mechanisms, etc.) supplemented by rational transition/interpolation formulas and approximations to account for non-ideal encounter geometries (e.g., yawed penetrators and oblique impacts). Together, these models enable FATEPEN to select and/or transition (interpolate) between the relevant ideal penetration models in accord with changes in the penetrator due to mass loss and orientation changes between impacts. This approach enables FATEPEN to accomplish penetration calculations for a wide variety of penetrator shapes and materials, while the user need only input the initial description of the penetrator. FATEPEN handles KEPs based on length to diameter ratio of the projectile. If the ratio is large than the model uses a highly modified version of the Alekseevskii-Tate model. If the ratio drops during the penetration process it then transitions to handling the penetrator as a fragment and uses penetration equations appropriate for that application [31, 32].

#### 2.4.2.4 Walker-Anderson Model

In 1995, Walker and Anderson [33] proposed a modification of the Alekseevskii-Tate model to include three primary changes: (1) the addition of a constant crater radius and zone of plastic deformation to calculate the velocity profile of the projectile in the target, (2) the calculation of projectile deceleration by elastic waves, and (3) the determination of shear behavior by using the shear stress gradient in terms of gradients of the velocity field and the velocity flow field to evaluate the velocity gradient.

#### 2.4.2.5 Segletes Model

In 2000, Segletes [34] proposed a hybrid version of the Frank-Zook and the Walker-Anderson models that integrated the resistance of all elements of a target array that are entrained in the plastic zone of deformation in front of the penetrating KEP. The new model, known as the Segletes model, was specifically designed with the intent to model multiple element targets effectively.

The Bernoulli equation was modified to account for the plastic zone of deformation in the KEP and in the target (see 2.10). In the equation, variables with a dot above them (such as $\dot{s}$) represent a rate of change for that variable. A variable with a bar over them (such as $\overline{H}$) represent the "averaged" value of that variable across the plastic zone of deformation. Along with the Segletes version of the Bernoulli equation, the model is based on three simplified principles:

- The change in penetration ($\dot{P}$) is equal to the penetration velocity ($u$) (see 2.11).
- The change in length of the penetrator ($\dot{L}$) is equal to the difference of the penetrator velocity ($v$) and the penetration velocity ($u$) (see 2.12).
- The change in velocity ($\dot{v}$) is equal to the fraction of penetrator strength ($Y$) and penetrator density ($\rho_p$) times the difference of the length of the penetrator ($L$) and the plastic zone in the penetrator ($s$) (see 2.13).

$$\left(k_p - \frac{\dot{s}}{2\dot{L}}\right)\rho_p(v-u)^2 + Y - \frac{\rho_p s}{2}(\dot{v}+\dot{u}) = k_t\overline{\rho_t}u^2 + \overline{H} + X_u\frac{\dot{u}}{u} + X_\alpha\frac{\dot{\alpha}}{\alpha} + X_R\frac{\dot{R}}{R} \tag{2.10}$$

$$\dot{P} = u \tag{2.11}$$

$$\dot{L} = -(v-u) \tag{2.12}$$

$$\dot{v} = -\frac{Y}{\rho_p(L-s)} \tag{2.13}$$

Where:

$\alpha$ is the extent of target-plastic zone in multiples of crater radii

$H$ is the target resistance

$k_p$ is the penetrator Bernoulli shape factor

$k_t$ is the target element Bernoulli shape factor

$L$ is the length of the penetrator

$\rho_p$ is the density of the penetrator

$\rho_t$ is the density of the target

$R$ is the crater radius

$s$ is the penetrator plastic zone extent

$u$ is the rate of penetration

$v$ is the penetrator velocity

$Y$ is the penetrator strength

$X_u$ is a homogenized function for $u$

$X_\alpha$ is a homogenized function for $a$

$X_R$ is a homogenized function for $R$

The U.S. ARL performed a study in 2012 to determine which KEP ballistics model was the most accurate, and the Segletes model was chosen out of a total of six models that were compared [13]. In the report from the study, the Segletes model was noted for performing very well at predicting terminal ballistics against multi-element targets.

### 2.4.3 Generalization Methods for All Models

Most models are not sophisticated enough to be used for all possible pairings of threat and target types as well as impact conditions. This usually means that an initial fitting of the model is done to a limited set of experimental test data and that sub-models are used to extrapolate outside of that data. In most cases the sub-models are not designed to specifically work with a given model and therefore inefficiencies exist when they are used together.

The base case of experimental test data typically used to fit a model is for a given threat versus a RHA steel target of a given thickness and at various velocities (the number of which is usually limited by funding constraints). When a terminal ballistics model is used in V/L modeling the threat may change, as it traverses through the target, but it is a near certainty that more than one target material will be encountered at a large range of velocities and orientations. Fitting a model to data that covers the possible range of materials and encounter conditions is a more robust solution to generalization than using sub-models that are not included in the fitting process.

#### 2.4.3.1 Yaw Impact Scaling

One of the encounter conditions that has an effect on the capability of a projectile to penetrate a target is the yaw of the projectile at impact. Total yaw is the angle between the longitudinal axis of a penetrator and the velocity vector of the penetrator's center of mass. The total yaw is composed of two orthogonal angles called pitch, measured in the vertical plane, and yaw, measured in the horizontal plane. Total yaw is commonly called yaw, so care must be taken when using experimental test data for measured total yaw. Yaw is largely ignored in V/L modeling, even though it can have a drastic affect on penetration. Methods do exist for the modeling of the effect of total yaw on penetration. One example method to model yaw calculates a critical yaw for the given projectile and then scales penetration based on the ratio of current yaw and critical yaw [35]. The equations for that process are given in (2.14), (2.15), and (2.16).

$$\gamma_{cr} = \sin^{-1}(\frac{D_h - D_p}{2\,L_s}) \tag{2.14}$$

$$\frac{D_h}{D_p} = 1 + \frac{v_s}{3000} + \frac{v_s^2}{8000000} \tag{2.15}$$

$$\gamma_\alpha = \begin{cases} 1 & \text{if } \gamma < \gamma_{cr} \\ \cos(\frac{\gamma}{\gamma_{cr}}) & \text{if } \gamma \geq \gamma_{cr} \\ 0 & \text{if } \frac{\gamma}{\gamma_{cr}} \geq 90^\circ \end{cases} \tag{2.16}$$

Where:

$\gamma_{cr}$ is the critical yaw angle, at which yaw begins to effect penetration

$D_h$ is the penetration hole diameter

$D_p$ is the diameter of the penetrator

$L_s$ is the striking length of the penetrator

$v_s$ is the striking velocity of the penetrator

$\gamma$ is the total yaw of the penetrator

$\gamma_\alpha$ is the amount to adjust penetration by to account for total yaw

#### 2.4.3.2 Obliquity Adjustment

Another encounter condition that has an effect on the capability of a projectile to penetrate a target is the obliquity of the impact. Obliquity is the measure of the angle of the velocity vector of the penetrator and the surface of the target. Obliquity is modeled by increasing the effective thickness of the armor to account for the increased line-of-sight thickness of the target (the concept can be seen in Figure 2.8). If the shotline is thought of as the hypotenuse of a right triangle, the normal thickness of the target as one of the sides of the triangle, and the obliquity angle as the angle between the hypotenuse and the adjacent normal thickness, then (2.17) can be used to solve for the line-of-sight thickness of the target.

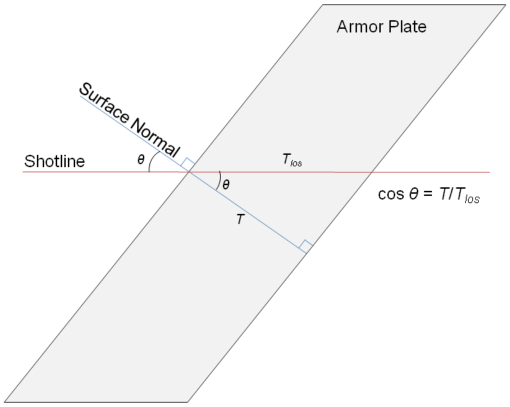


Figure 2.8: Illustration of the difference between normal thickness and line-of-sight thickness

$$T_{los} = \frac{T}{\cos\theta} \tag{2.17}$$

Where:

$T_{los}$ is the line of sight thickness of the target

$T$ is the normal thickness of the target

$\theta$ is the obliquity of the target

This method of modeling obliquity is limited because it assumes that the only effect obliquity has on penetration is to increase the thickness of the target. There are other physical phenomena involved in the process (especially at higher obliquities), such as ricochet and deflection.

#### 2.4.3.3 Density Scaling

One method that is used throughout V/L analyses to force a more generalized model, is density scaling [26]. Density scaling is the process of scaling a predicted penetration into steel by the ratio of the target density and density of steel and is given in (2.18). Steel is used as the reference material because most of the experimental test data available are for projectiles fired into RHA (a type of steel). The effectiveness of density scaling is questionable and although it forces a generalization for all target materials it does not generalize for different threats.

$$T_{RHA} = \frac{\rho}{\rho_{RHA}} \cdot T \tag{2.18}$$

Where:

$T_{RHA}$ is the equivalent thickness of a RHA target

$T$ is the thickness of the target

$\rho_{RHA}$ is the density of RHA

$\rho$ is the density of the target

#### 2.4.3.4 Mass Efficiency Scaling

Mass efficiency scaling uses the comparative penetration of a material of interest ($P$) and that of RHA ($P_{RHA}$) to scale the target thickness to an equivalent thickness of RHA. Mass efficiency is calculated using (2.19). The target thickness is multiplied by the mass efficiency and by the density ratio of the target material ($\rho$) and RHA ($\rho_{RHA}$), resulting in an target effective thickness [26]. The equation for target effective thickness is given in (2.20). Mass efficiency is a more effective way of scaling from one material to another, but requires experimental test data to fit each material type and is a function of the velocity and type of the penetrator [36].

$$e_m = \frac{P_{RHA} \cdot \rho_{RHA}}{P \cdot \rho} \tag{2.19}$$

$$T_{eff} = e_m \cdot \frac{\rho}{\rho_{RHA}} \cdot T \tag{2.20}$$

Where:

$e_m$ is the mass efficiency of the target

$P_{RHA}$ is the penetration depth into RHA

$\rho_{RHA}$ is the density of RHA

$P$ is the penetration depth into the target

$\rho$ is the density of of the target

$T_{eff}$ is the effective thickness of RHA

## 2.5 Artificial Neural Networks

The basis for Artificial Neural Networks (ANNs), the artificial neuron, was proposed in 1943 by McCulloch and Pitts [37]. Throughout the 1950s artificial neurons were arranged in layers to create perceptrons which were used for pattern recognition (see Figure 2.9). The ANN in Figure 2.9 is an example of a fully connected Multi-Layered Perceptron (MLP). In a fully connected MLP, each layer is fully connected with its adjacent layers and there are no recurrent connections or connections to non-adjacent layers. For a layer to be fully connected to another layer, each node in the first layer must have a connection to every node in the second layer [38]. Early research on perceptrons concentrated on the implementation of Boolean logic functions, which require the decision boundaries to discriminate perfectly between the different classes. Following the 1960s, it became generally accepted that the strength of neural networks was in their ability to analyze and recognize complex patterns in real-world data. Real-world data is intrinsically noisy, meaning that the decision lines are not clear because there will be some patterns that do not fall perfectly within the decision boundaries creating regions of overlap [15].

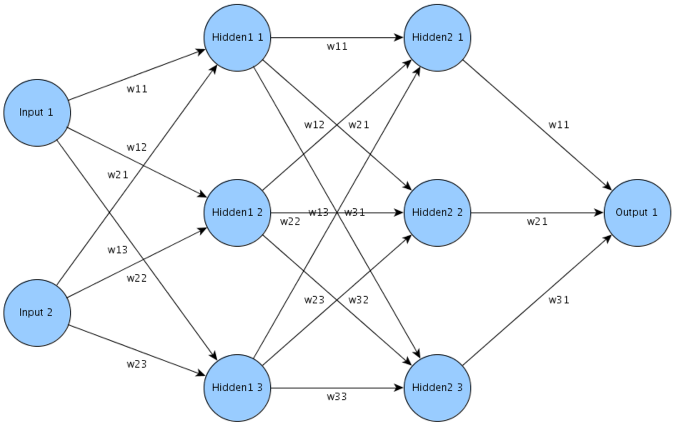


Figure 2.9: Example topology of a MLP

Tarassenko lists five key attributes of neural networks in the book “A Guide to Neural Computing Applications” [15]:

- Learning from Experience

  Neural networks are particularly suited to problems whose solution is complex and difficult to specify, but which provide an abundance of data from which a response can be learned.

- Generalizing from Examples

  A vital attribute of any practical self-learning system is the ability to interpolate from a previous learning ‘experience’. With careful design, a neural network can be trained to give the correct response to data that it has not previously encountered (This is often described as the ability to generalize on test data).

- Developing Solutions Faster with less Reliance on Subject Matter Expertise

  Neural networks learn by example, and as long as examples are available and an appropriate design is adopted, effective solutions can be constructed far more quickly than is possible using traditional approaches, which are entirely reliant on experience in a particular field. However, neural networks are not wholly independent of domain expertise which can be invaluable in choosing the optimal neural network design.

- Computational Efficiency

  Training a neural network is computationally intensive, but the computational requirements of a fully trained neural network when it is used on test data can be modest. For very large problems, speed can be gained through parallel processing, as neural networks are intrinsically parallel structures.

- Non-Linearity

  Many other processing techniques are based on the theory of linear systems. In contrast, neural networks can be trained to generate non-linear mappings and this often gives them an advantage for dealing with complex, real-world problems.

There exists a lot of publicly available data for penetration of KEPs into various materials that could be used to perform an intelligent generalized regression. However, the determination of an equation to be used for a generalized fitting could be difficult to ascertain. ANNs are a common tool for performing non-linear regression, especially when the parametric form of the function is unknown and when the number of parameters is large [39]. A specific type of ANN called a Multi-Layered Perceptron (MLP) has been shown to be a universal approximator, meaning it is capable of arbitrarily accurate approximation to an arbitrary mapping, if there are enough hidden neurons in the hidden layer [40]. A MLP should be able to accurately approximate the desired outputs, given that the appropriate parameters are used. The parameters in question are the inputs to the model, the topology of the MLP (to include the activation functions, number of layers, and number of neurons), the error function, the training method, and the test data.

### 2.5.1 Multi-Layered Perceptron

The application of a MLP for this research was chosen based on the work of I. Gonzalez-Carrasco, et al. [40], which found the application of MLPs to outperform Radial Basis Function (RBF) networks, Support Vector Machines (SVMs), and Recurrent Neural Networks (RNNs) for predicting perforation of steel, DU, or WHA KEPs against aluminum, steel or DU targets.

The structure of a MLP consists of a single input layer of neurons, a single output layer of neurons, and a variable amount of hidden layers of neurons. In the example MLP shown in Figure 2.9, there is one input layer containing two input neurons, two hidden layers with three hidden layer neurons each, and one output layer with one output neuron. The count of the number of layers of an MLP could have different values depending on which sources are asked, but the prevailing definition counts all of the computational layers when counting layers. This means that an MLP with an input layer, a hidden layer, and an output layer is a 2-layer MLP, because the input layer does not perform computations. This paper uses this definition when discussing the number of layers for an MLP.

The number of neurons in the input layer is determined by the number of inputs to the model, so for the example given there would be two inputs to the model. The number of neurons in the output layer is determined by the number of outputs to be predicted by the model, so for the example given there would be one output from the model.

The determination of the number of hidden layers and neurons contained in them is not as succinctly explained. In many cases the determination of the structure of the hidden layers is done by trial and error [41].

The number of hidden layers in a MLP affects complexity of the domain problem to be solved. A MLP with no hidden layers can classify linearly separable input data. Increasing to one hidden layer can create a hyperplane, two hidden layers combine hyperplanes to form convex decision areas, and three hidden layers combine convex decision areas to form convex decision areas that contain concave regions [42].

In general, increasing the number of hidden layers will improve the closeness-of-fit and decreasing the number of hidden layers will improve the smoothness (extrapolation capability) of the MLP [43].

As with the number of hidden layers, there exist heuristics for selecting the number of neurons in a hidden layer, but it is still done primarily by trail and error. In general, a higher number of neurons will improve accuracy but also increase the training time required for the MLP and could lead to over-fitting of the data [43, 44].

### 2.5.2 Artificial Neural Network Characteristics

#### 2.5.2.1 Data Requirements

Although the training data did not need to be prepared for use until the MLP was fully designed and implemented, the collection of experimental test data was the first task performed for this research. This is because the types, amount, and pedigree of experimental test data available had implications on the design of the MLP.

An approximation of the number of training data points required for a given network topology, or reciprocally the size limitation of a network topology due to the number of training data points can be found in (2.21) and (2.22) [15]. In (2.21), $n$ is the number of training data and $W$ is the total number of network parameters (the network parameters are the weights associated with the connections between the nodes in the MLP) that must be adjusted during training.

$$W \leq n \leq 10W \tag{2.21}$$

The parameter $W$ can be obtained by using (2.22), where $N$ is the number of layers in the MLP topology and $L_i$ is the number of neurons in the $i^{th}$ layer.

$$W = \sum_{i=1}^{N-1} (L_i + 1) L_{i+1} \tag{2.22}$$

The effect that MLP complexity has on the amount of training data required can be demonstrated by using (2.21) and (2.22). For example a simple 2-layer MLP with two input neurons, two hidden neurons, and one output neuron, the recommended number of training data fall between nine and ninety. For a more complex example, a 3-layer MLP with six input neurons, seven hidden neurons in the first hidden layer, six hidden neurons in the second hidden layer, and three output neurons, the recommended number of training data fall between one hundred eighteen (118) and one thousand one hundred eighty (1180). The more complex the MLP the more data are required for training. As Tarassenko states [15]:

> *Artificial Neural Network projects are data driven, therefore there is a need to collect and analyze data as part of the design process and to train the neural network. This task is often time-consuming and the effort, resources, and time required are frequently underestimated.*

In order to decrease the likelihood of poor predictions when extrapolating it is important to use training data that covers the range of all possible inputs. Figure 2.10 shows an example of what can happen if a region of the input space is omitted from the training data. The square marks are the data points that were used for the non-linear regression, the circular marks are the data points that were omitted, and the curved line shows the model predictions. The model predicts the training data very well and interpolates between the data points well, but because of the omitted data the wrong model was used for fitting, thus leading to poor extrapolation. The collection of experimental test data that is representative of the large space of possible input patterns and that can be used for training, testing, and validating the MLP, was one of the more difficult tasks involved in this research [45]. Therefore, a large part of the effort for this research was finding and documenting publicly available experimental test data for KEPs.

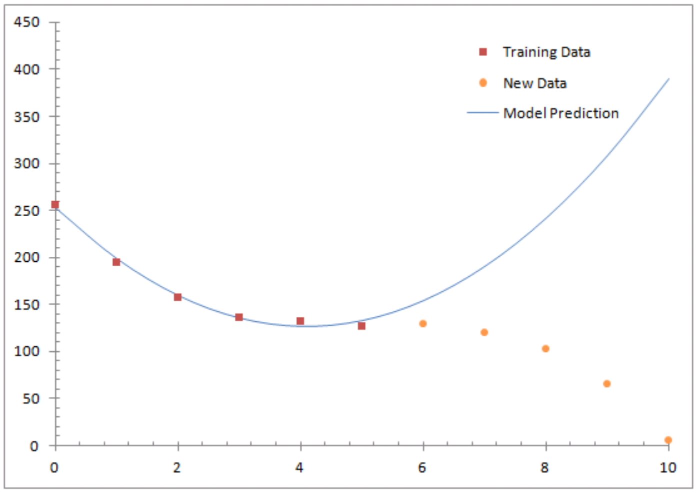


Figure 2.10: Incorrect extrapolation predictions

Experimental test data is inherently noisy, but hidden assumptions in the data collection methods or data processing methods could cause major differences in the data. As an example, suppose there are four reports containing experimental test data, and during the test events for all of the reports the KEP fractured into smaller pieces as it perforated the target. In report number 1, the residual mass is reported as the weight of the largest piece. In report number 2, the residual mass is reported as the weight of all of the pieces. In report 3, x-ray is used to approximate the length and diameter of the largest few pieces, and then the mass is calculated using the volume and density of the rod material. In report 4, a piece of the KEP that was embedded in the target is included in the residual mass calculation. Four similar test events, but with four different reported results.

The example given shows how important it was to find outliers in the training data and attempt to track down the cause of the discrepancies so that they could be fixed or omitted.

#### 2.5.2.2 Normalization

The selected input parameters for this model where not all the same units and in some cases differed by several orders of magnitude in value. So to make training easier for the MLP the data were normalized by means of a linear transformation. Bishop wrote that, "Input normalization ensures that all of the input and target variables are of order unity, in which case we expect that the network weights should also be of order unity" [46]. In addition, that basic assumption allows for a simpler approach for random weight initialization prior to training of the MLP. The data used for this research were normalized according to the values influenced by two factors. The first factor was to provide bounds where the model could be expected to be used for realistic KEP analysis and the second factor was due to limitations of the data currently available. The values used for normalizing the parameter are provided in Table 2.1

Table 2.1: Minimum and maximum bounds for normalization

| **Parameter** | **Minimum Value** | **Maximum Value** | **Minimum Normalized** | **Maximum Normalized** |
|---|---|---|---|---|
| Input Values | | | | |
| Striking Velocity ($V_s$) | 0.0 m/s | 2000.0 m/s | -1.0 | 1.0 |
| Total Yaw ($\gamma$) | 0.0 ° | 20.0 ° | -1.0 | 1.0 |
| Projectile Density($\rho_p$) | 0.0 g/$cm^3$ | 20.0 g/$m^3$ | -1.0 | 1.0 |
| Projectile Length ($l$) | 0.0 mm | 225.0 mm | -1.0 | 1.0 |
| Projectile Diameter($d$) | 0.0 mm | 15.0 mm | -1.0 | 1.0 |
| Projectile Hardness ($BHN_p$) | 0.0 BHN | 900.0 BHN | -1.0 | 1.0 |
| Target Density ($\rho_t$) | 0.0 g/cm$^3$ | 8.0 g/m$^3$ | -1.0 | 1.0 |
| Target Hardness ($BHN_t$) | 0.0 BHN | 700.0 BHN | -1.0 | 1.0 |
| Target Thickness($T$) | 0.0 mm | 160.0 mm | -1.0 | 1.0 |
| Target Young's Modulus($E$) | 0.0 GPa | 500.0 GPa | -1.0 | 1.0 |
| Target Obliquity ($\theta$) | 0.0 ° | 80.0 ° | -1.0 | 1.0 |
| Output Values | | | | |
| Perforation Outcome ($P$) | Non-Perf | Perf | -0.9 | 0.9 |
| Residual Velocity ($V_r$) | 0.0 m/s | 1700.0 m/s | -0.9 | 0.9 |
| Residual Mass ($M_r$) | 0.0 g | 120.0 g | -0.9 | 0.9 |

#### 2.5.2.3 Target Output Encoding

Target outputs were encoded to -0.9 and 0.9; this helps avoid saturating the sigmoid function. “If the targets were set to the asymptotes of the sigmoid it would tend to: a) drive the weights to infinity, b) cause outlier data to produce very large gradients due to the large weights, and c) produce binary outputs even when incorrect” [47]. Sigmoid activation functions cannot reach the extremes of their bounds. For example the hyperbolic tangent function will never be equal to -1.0 or 1.0, because it has asymptotes of -1.0 as $x \to -\infty$ and 1.0 as $x \to \infty$. This makes it very hard for an MLP to learn an output value of -1.0 or 1.0. To account for that difficulty the outputs for this model were encoded as a -0.9 for a non-perforation and 0.9 for a perforation to improve the ability of the MLP to learn the target outputs [48].

#### 2.5.2.4 Activation Functions

The most common type of activation function used are the sigmoidal functions because they are differentiable throughout the domain of the function. The two most common sigmoidal functions used are the logistic function and hyperbolic tangent function. The logistic function varies in range from 0.0 to 1.0 and the hyperbolic tangent function varies in range from -1.0 to 1.0 [40, 49, 50]. Since the hyperbolic tangent function ranges in value from negative and positive values, it provides faster training than functions that are all positive, because of better numerical conditioning [49]. Another activation function is the linear function; it is not typically used in the hidden layers of an MLP because a linear function of linear functions is still a linear function and therefore the model would only be able to learn linearly separable problems. However, the linear function is still commonly used by the output layer neurons. An example of all three activation functions is provided in Figure 2.11.

The activation function used in the output neurons is determined by the type of outputs desired and the type of regression being done. If the outputs for the MLP will be scalar and not bound then linear activation functions will be needed for the output neurons [49]. If the desired output type is boolean, then a sigmoid activation function may be desired.

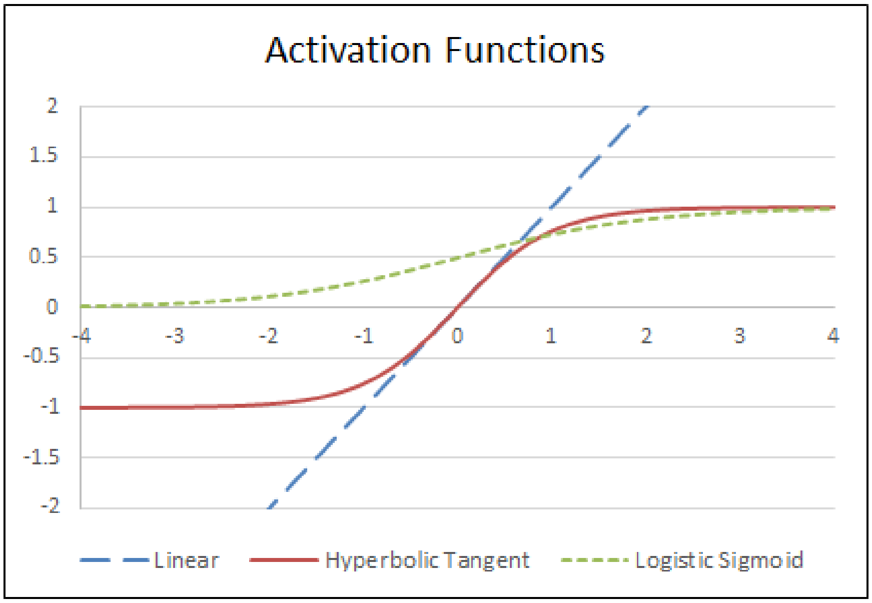


Figure 2.11: Example of three activation functions

If the output is not a linear regression problem, then the activation functions used in the hidden layers of the MLP need to be non-linear in order to be fit to a non-linear problem. The hyperbolic-sigmoid is the preferred activation function since its mean is centered on zero and leads to faster convergence of the network [51].

#### 2.5.2.5 Bias Neuron

The output of a neuron is computed by multiplying the input by the corresponding weight and passing the result through the sigmoid activation function [46]. The effect can be seen in Figure 2.12 for weights of 0.5, 1.0, 1.5, and 2.0.

If a bias neuron is used then the computation is modified to include the addition of the bias value before passing the value through the sigmoid function. In effect the use of a bias neuron allows for the shifting of the output of a neuron along the x-axis and can be seen in Figure 2.13.

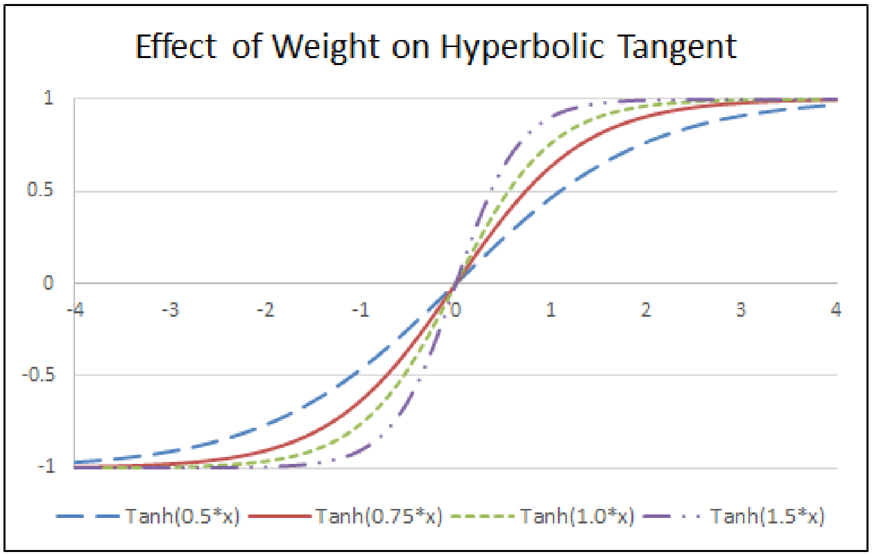


Figure 2.12: Example of how a neuron's output is effected by weight adjustment

#### 2.5.2.6 Generalization Techniques

As mentioned earlier, it is important to this research for any model developed to be a generalized solution. If non-representative data is used to train the MLP, then poor extrapolation could occur. Even if the data used for training the model represents the broad range of possible conditions that the model could be asked to predict from, there is still a concern that the model will not be effective at predicting results that are not part of the training data.

As an example, if the MLP is not properly designed then it could over-predict the training data and not provide a smooth fitting of the training data. Figure 2.14 shows an example of a model that has been overfit to the training data. The diagonal line represents a good fit to the training data points, but the curved line represents a solution that could come from a MLP if overfitting occurs.

There are techniques available to increase the likelihood of producing a generalized solution and reduce the risk of overfitting. One methodology that can be used is weight decay; it penalizes large weights in the network and causes the weights in the network to converge to smaller absolute values. Excessively large weights in the network can lead to

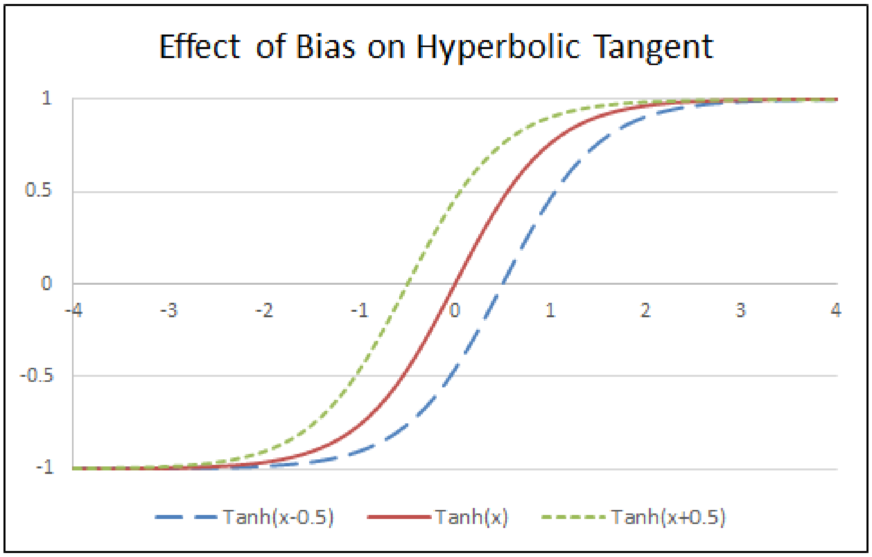


Figure 2.13: Example of how a neuron's output is effected by bias adjustment

excessive variance of the outputs from the network [49]. Another method for producing a more generalized model is to use early stopping during the training process, by testing the error of a validation set of data that is separate from the training group of data [40].

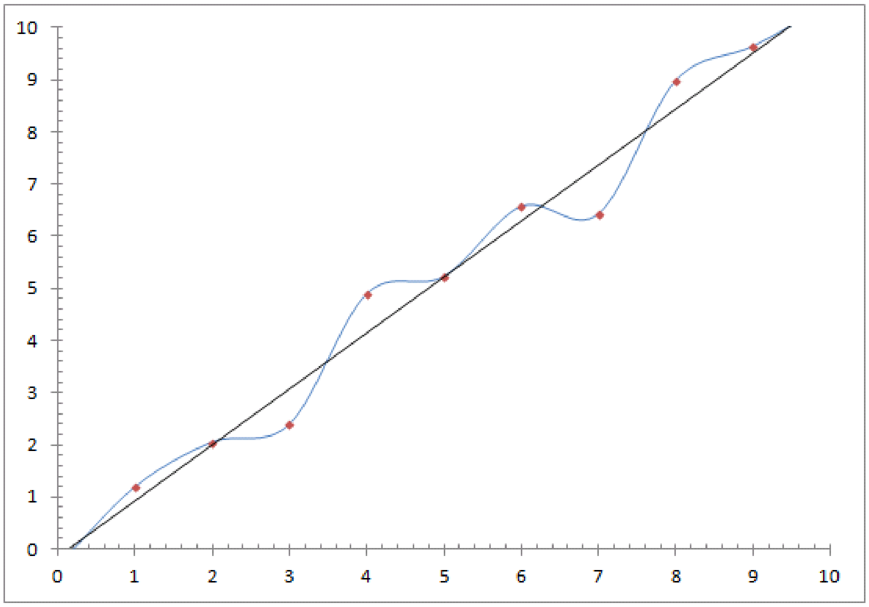


Figure 2.14: Example of the overfitting of data

#### 2.5.2.7 Early Stopping Strategies

There are three concerns that can arise while training an MLP that can be addressed using an early stopping strategy. One such concern is when the training error begins to oscillate around a minimum error value; this can lead to extremely long runtimes and/or a non-convergence. Another concern is when the training error is progressing but at a very low rate. When this occurs it can again lead to very long runtimes or a non-convergence. The last concern is model generalization. If a model is only trained to a minimal error on a training set, then it is less likely to perform well on new data, because it will be over-fit to the training data. One way to address that is by stopping training when a criteria, based on a validation set, has been met. It is possible for the error to go up in the validation set, but then go back down to a better error later on. This is important because if the training is stopped too soon then the global solution may not be reached. However, letting it run too long increases the runtime.

#### 2.5.2.8 Back-Propagation Methods

An important step in defining the MLP involves picking an appropriate learning method for the problem class being addressed [43]. The choice of learning method will determine how well the MLP will learn the patterns that it is being taught and includes the learning algorithm, error function, learning rate, and other optional methodologies. The optimization algorithms used for learning fall into two categories: direct (gradient-free) methods or gradient methods.

Direct methods use only the function values themselves to find the optima in question. Examples of direct methods include simulated annealing, perturbation methods, or genetic algorithms. The advantages of direct methods are that there is no need to derive or compute gradients and that the methods can find a global optimum. The disadvantages are that they can take too many iterations to converge to a solution and although they can come to a solution close to a global optimum, there is no guarantee that they will come to that exact solution.

Gradient methods use the gradient of the function to determine the optima in question and can be further defined as $1^{st}$ or $2^{nd}$ order. Examples of gradient methods include gradient descent, Newton method, Gauss-Newton method, and Levenberg-Marquart (LM) method. The primary difference between a $1^{st}$ order and $2^{nd}$ order method is the required number of iterations prior to convergence and speed of calculation. $1^{st}$ order methods only need to calculate the $1^{st}$ derivative of a function which requires less calculation time, but may take a less directed approach to finding the optimum. $2^{nd}$ order methods require longer to calculate $2^{nd}$ derivatives or the Hessian matrix, but take a more direct approach to finding the optimum [52]. Any of the example optimization methods can be used to find a minimum of an error function, however a global minimum for the error function is not guaranteed. In their paper, Danaher et al. states [53]:

> *Non-convergence to a global minimum is improved both by keeping the complexity of the network to a minimum, using improvements on the standard back-propagation algorithm such as the inclusion of momentum or by using a second-order solver such as the Levenberg-Marquart algorithm with multiple random starting values.*

#### 2.5.2.9 Global Optimization

A function can have multiple optima; Figure 2.15 shows an example function that contains four maximums and three minimums, but there is only one global (overall) maximum and only one global minimum. An optimization function that does not guarantee the convergence to a global optima could converge to a non-optimal solution if other methods are not used.

There are several techniques available to increase the likelihood of finding the global minimum for the error function. One technique that can be used is the method of momentum; momentum is used to resist changes to the direction of the weight changes. The main reason for using momentum is to reduce the chance of oscillating around a minimum; however there is a slight chance that since momentum can also speed up the weight adjustments it may skip over a small local minimum [54]. Momentum was not originally designed for finding global minimums and its probability of skipping a local minimum is small, so other techniques are better suited for this purpose.

Another technique that can be used is to sample several random potential weights for the network and start with the one that has the lowest error. The random sampling technique in no way guarantees a global minimum, but does help the learning process by allowing the network to start the learning process as close to a minimum solution as possible and could start the learning process close to a global minimum [55]. A disadvantage of this method is that since it is truly random it is not a directed approach and is therefore inefficient when compared to directed methods.

A technique that has gained popularity is to use a hybrid approach that attempts to utilize the benefits of direct and gradient optimization methods together. Initially a direct approach is used to get close to a global optimum because direct approaches are traditionally better equipped to do this than gradient methods. Direct methods, however, are typically inefficient in converging to the specific solution, so the next step is to apply a gradient method to assist in the convergence.

An example of this technique is the use of Genetic Algorithms (GAs); they can be used to determine starting weights for the network prior to the learning process beginning. Like with random sampling, using a GA does not guarantee a global minimum, but does increase the likelihood of finding it since it is a directed method and is more efficient than random sampling [54]. Once a criteria has been met by the GA the learning process begins using a gradient method for the determination of the required weights to reach the global minimum of the error function.

Genetic algorithms are global search methods that are based on principles like selection, crossover, and mutation [56]. Genetic algorithms are good at finding the approximate global optimum, but are inefficient at finding the exact solution. It is more efficient to get close to the global optimum solution using a genetic algorithm and then use a local search method such as back-propagation [57].

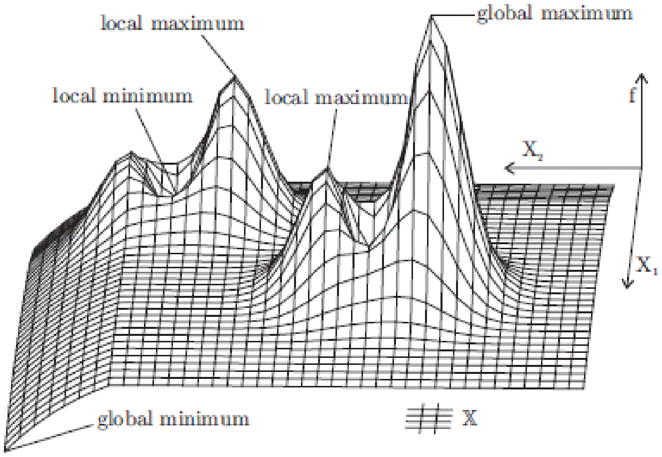


Figure 2.15: 3-Dimensional example of local and global optima [58]

#### 2.5.2.10 Training, Validation, and Testing Sub-Data Sets

When developing a predictive model it is imperative that the model be tested with data on which it has not been trained. For that reason, the database is divided into 2 subsets of data, a training/validation set and a test set. From that point forward, the test set of data is never used for training the model. After the training phase is completed the test set is run through the model and the results are reviewed to gauge performance.

#### 2.5.2.11 K-Fold Cross-Validation

K-fold cross-validation partitions the data into $k$ nearly equal sized folds. There are $k$ MLPs that are trained on $k-1$ folds of the data; for each MLP a different fold is left out. The fold that was left out during training is used to calculate the error of that MLP. The approximate generalized error of that MLP topology is the average of all $k$ MLPs and is defined in (2.23) [59]. Using k-fold cross validation to determine the generalization error gives a better prediction of how well a given topology will perform against new information [60].

$$E = \frac{1}{k}\sum_{i=1}^{k} E_i \tag{2.23}$$

Where:

$E$ is the generalization error of the network topology

$k$ is the number of folds of data

$E_i$ is the error of the $i^{\text{th}}$ network

#### 2.5.2.12 Cost Function

The cost function of a neural network, is a function that is used to measure how well the neural network performed with respect to a given training sample and the corresponding expected output. The cost function plays a significant role in the back-propagation process; it is used to calculate the error of the output layer. Two of the more common cost functions are the L1 norm (Manhattan metric) or the L2 norm (Euclidean norm). Use of the L2 norm leads to the Mean Squared Error (MSE) criterion and is commonly used in regression and classification problems. However, for cases where there are outlier data the L2 norm may over value the errors of the outlier data. For those situations, the L1 norm may be preferred because it weights the differences of the predicted and target values proportionally to their magnitude [40].

#### 2.5.2.13 ANN Topology Selection

An important step when designing an MLP is to determine the architecture of the network [43]. The architecture consists of the number of input layer neurons, hidden layer neurons, output layer neurons, number of hidden layers, and the activation functions that are used on each of the neurons. The number of input neurons is normally known during this phase because the data and problem space will dictate the inputs needed for the topology. The determination of the number of hidden layer neurons is normally a process of trial and error; however an approximation can be made using (2.24) [15].

$$J \sim \sqrt{IK} \tag{2.24}$$

Where:

$J$ is the number of neurons in the hidden layer

$I$ is the number of neurons in the input layer

$K$ is the number of neurons in the output layer

The output neurons will be governed by the desired regression outputs, however care needs to be taken in selecting outputs that are not highly correlated. Having too many output neurons adds to the complexity of the network, thus increasing the training data required and the likelihood of a non-convergent network. The number of hidden layers is determined by the complexity of the problem area being researched: “As the dimensionality of the problem space increases, the number of hidden layers should increase correspondingly” [43].

There is a trade-off between the smoothness and closeness-of-fit for a MLP when selecting the number of hidden layers. If too many hidden layers are added to the topology of the MLP there is an increased chance for over-fitting of the training data. Likewise, if too few hidden layers are added to the topology then there is an increased chance for under-fitting of the training data.

## 2.6 Applications of Artificial Neural Networks in the Field of Ballistics

In the initial literature search eight references were found that were applications of ANNs to terminal ballistics. Of those eight applications only two used actual experimental test data for training the ANN; the rest used Finite Element Analysis (FEA) to generate the training data. The two that used actual experimental test data were an application of Fragment Simulating Projectiles (FSPs) versus Kevlar [61] and the other application was for SCJs versus “sandwich packs” (packs of three materials layered together) [39].

KEPs are different than FSPs because the impact conditions and orientations are well defined for KEPs. FSPs could be impacting a target in any possible orientation, but KEPs will typically impact with their front forward and with minimal yaw. The amount of KE, imparted on a target from a KEP, is also typically going to be larger than that for a FSP; this is because there is typically more mass and in some cases more velocity at impact. Kevlar behaves very differently than metal targets, so a ANN developed for Kevlar is not likely to be usable for homogeneous metallic targets.

SCJs are different in that they are formed by the process of detonating a high explosive behind a liner material. Once the SCJ is formed from the liner material the velocity at impact is generally much faster than that of a KEP (10 000 $^{\mathrm{m}}/_{\mathrm{s}}$ for SCJs vs. 1400 $^{\mathrm{m}}/_{\mathrm{s}}$ for KEPs). At higher velocities the penetration process becomes more simplified in that the material hardnesses become less important and the primary material property driving the penetration process becomes density. The "sandwich pack" approach is limited in that the MLP requires three layers of materials, so a scenario of one, two, or more than three layers is not possible.

Two of the applications that utilized FEA to generate training data were an ANN for predicting spherical fragments against carbon fiber reinforced composite [62] and an application for KEPs against ceramic armor backed with aluminum or steel [45]. Although they do not represent the type of impact conditions that this research is trying to solve, they do show how effective ANNs are at predicting terminal ballistic events.

The remaining four applications are similar to the domain that this research is proposing; however there are some distinct differences that make this research effort unique. As mentioned none of the four applications utilize experimental test data for training; in addition, each may offer one of the following features but not one of them provides all features [40, 50, 53, 63]:

1. Prediction of penetration, residual length, residual velocity, and residual mass
2. Generalized usage for all valid homogeneous metallic target and rod materials
3. Impact conditions where obliquity and yaw are non-zero
4. A broad range of impact velocities ($200\,^{\mathrm{m}}/_{\mathrm{s}} \leq v_s \leq 5000\,^{\mathrm{m}}/_{\mathrm{s}}$)
5. A broad range of diameters ($1\,\mathrm{mm} \leq D \leq 30\,\mathrm{mm}$)
6. A broad range of lengths ($1\,\mathrm{mm} \leq L \leq 400\,\mathrm{mm}$)
7. A broad range of length/diameter ratios ($1 \leq {}^{L}/_{D} \leq 40$)

The numbers provided in the list above are notional, but do give a rough idea of the range of values desired for this research. As mentioned in previous sections, a training data set that represents the broad spectrum of possible target/threat encounters is important to the generalization of the MLP proposed in this research.

The four applications were trained using three types of materials for the projectile and three for the target. Although the materials that were used are commonly used in real world applications, they do not span the range of possible material types. The limited range of materials used for training reduces the ability of the MLP to extrapolate and interpolate, so generalization is reduced as well. The following list shows the range of the training data, used in the eight references, for the characteristics of projectile material, target material, striking velocity, projectile diameter, and projectile fineness ($^{L}/_{D}$):

- Projectile Materials: SAE 1006 Steel, Depleted Uranium, Tungsten Alloy
- Target Materials: SAE 1006 Steel, Aluminum, Depleted Uranium
- Velocities: $420\,^{\mathrm{m}}/_{\mathrm{s}} \leq v_s \leq 1200\,^{\mathrm{m}}/_{\mathrm{s}}$
- Diameters: $6\,\mathrm{mm} \leq D \leq 16\,\mathrm{mm}$
- Length/Diameter Ratios: $8 \leq {}^{L}/_{D} \leq 16$

All of the ANNs that were applied to terminal ballistics analyses were designed for use in predictions against targets consisting of a single element of armor, except for two cases. The first case was a MLP that was designed for the very specific case of a plate* of ceramic backed by a metallic plate. Generalization to a target consisting of one plate, more than two plates, or a different ordering of ceramic and metal plates was not addressed [45]. The second case was a MLP designed for modeling SCJs against "sandwich packs" of three materials [39]. As mentioned earlier, generalization to one, two, or more than three dependent plates is not addressed.

*The term plate is used to describe a single element of armor. The term comes from the fact that in the past a majority of armors were made from steel, or steel alloys, that are forged and not cast. The forging process typically results in a flat "plate" of armor.

Of particular interest from the second case is the application of the MLP iteratively for multiple packs. For cases of more than one “sandwich pack”, the MLP is fed inputs for the first pack and then the outputs for that pack are fed as inputs to another instance of the MLP. This process of feeding outputs into the next network as inputs can be performed for as many iterations as necessary. The iterative concept that they define as a “generic cassette-network” showed promise as a potential method of implementation for the MLP and was used in this research.

One of the applications uses an interesting approach for predicting residual values. Instead of using one MLP for determining perforation and residual values, the task was broken up into a MLP for classification (perforation and non-perforation) and if perforation was predicted, a second MLP for regression of the residual values [62]. The benefit of separating the two tasks is the reduction in complexity of the overall networks and therefore an increase in the likelihood of faster convergence.

## 2.7 Deep Learning

There has been recent success with ANNs containing more than 2 hidden layers of neurons. These types of networks are called Deep Neural Networks (DNNs) and they are capable of learning very complex problems. The reason they are just now becoming popular is due to recent advances in training techniques. Prior to the new training techniques, DNNs were thought of as too difficult to train and prone to overfitting issues [64]. Many of the early attempts at training DNNs using supervised learning and gradient descent resulted in networks that performed worse than ANNs with one or two hidden layers [64].

In general, the new approach taken for training DNNs is to first perform “pre-training” on each layer using unsupervised training techniques. This process is done to try and extract useful information at each layer and to convert very specific input values into more abstract concepts at subsequent layers [65]. After the initial “pre-training”, the network is further trained using traditional supervised methods, such as gradient descent back-propagation. Several recent studies have strongly suggested that a compact DNN could be used to represent a problem that a shallow ANN would require a very large number of neurons to represent [65].

Given the positive results from using a MLP for this research, the benefits of using a DNN was not investigated.

## 2.8 Summary

This chapter provided a brief overview of V/L modeling, KEP ballistics, terminal ballistic models for KEPs, ANNs, and some applications of ANNs to ballistics. Of particular note is the section on the Segletes model, which is the current standard for modeling KEPs. The MLP developed in this research will be compared to the Segletes model to determine performance. Also of importance is the section on applications of ANNs in ballistics, which this research builds on. None of the applications of ANNs in ballistics utilized experimental test data and provided enough range of target and threat materials to be generalized for use in V/L simulation models.

# Chapter 3

# Software Overview and Implementation Details

This dissertation leveraged available software when available, but also included a lot of time spent developing software to implement the database, Multi-Layered Perceptron (MLP), and the data analysis tools. The following sections will briefly go over some of the software developed and modified for this research, as well as the implemented design choices for the MLP.

## 3.1 Software Overview

The software used for this research came from one open source library and four software projects that were developed using Java. In total roughly 25 000 lines of code were developed to support this research project.

### 3.1.1 Encog

The initial search to find an open source Java library, for implementing an Artificial Neural Network (ANN), resulted in three libraries; Java Object Oriented Neural Engine (JOONE), Neuroph, and Encog. A further search for an open-source Java library for Genetic Algorithm (GA) implementation resulted in two libraries; Java Genetic Algorithms Package (JGAP) and Java API for Genetic Algorithms (JAGA). A deeper look into the performance of all three ANN libraries discovered an article that that benchmarked the performance of all three. The results of the benchmark were dramatically in favor of Encog. As summarized in the article, “Encog pretty much decimated the competition here. Even when Encog is forced to use a single thread, it beats the others by a huge mark. Encog is about 40 times as fast as Neuroph and 17 times as fast as JOONE” [66]. The article also laid out the features of each library, one of the features of Encog was that it could support both ANN and GA implementations. Encog was also capable of being run multi-threaded to improve on performance. Based on the positive reviews, excellent benchmark results, and feature set, Encog was selected as the library for use. Below is a brief description of Encog from the Encog website [67]:

> *Encog is an advanced machine learning framework that supports a variety of advanced algorithms, as well as support classes to normalize and process data. Machine learning algorithms such as Support Vector Machines, Artificial Neural Networks, Bayesian Networks, Hidden Markov Models, Genetic Programming and Genetic Algorithms are supported. Most Encog training algoritms are multi-threaded and scale well to multicore hardware. Encog can also make use of a GPU to further speed processing time. A GUI based workbench is also provided to help model and train machine learning algorithms. Encog has been in active development since 2008.*

Where necessary, the code provided by Encog was extended, modified, or replaced with code that would provide the needed capabilities for the selected MLP design considerations. An example of this was the creation of the ImprovedStopTrainingStrategy class to incorporate all of the planned early stopping strategies for the MLP training process. Other changes included the modification of three classes used for k-fold cross-validation, a class used for handling the thread pools, a class for performing a selective pruning algorithm, and a class used for the Scaled Conjugate Gradient (SCG) back-propagation.

### 3.1.2 PenDataModel

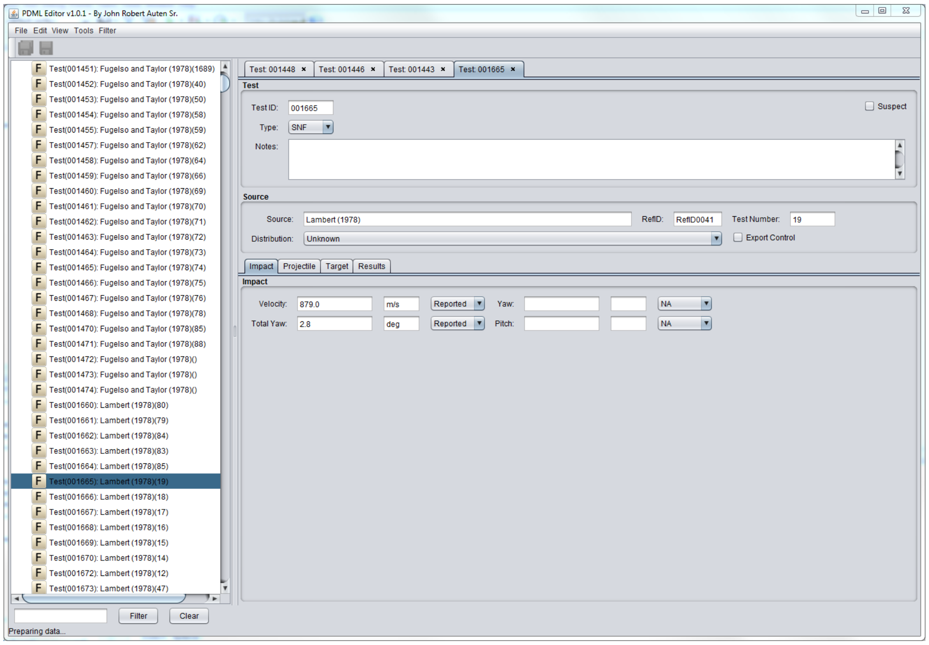
The PenDataModel is the data model used to interface with the Penetration Database Markup Language (PDML). Most of the code for this library is auto-generated from the PDML schema using Java Architecture for XML Binding (JAXB). Some of the other features that were implemented into the data model are; the ability to automatically mark records as suspect if the parameters required for the MLP are missing, filter the database based on passed parameters and then return an instance of the database that has been filtered, and search the database for duplicate records.

The process used for finding duplicate records was based on a paper by Elmagarmid, Ipeirotis, and Verykios [68]. It utilizes the Levenshtein distance (also known as edit distance) of the attributes for two records to determine the similarity of those attributes. Each attribute is given a score and all of the scores are averaged; a high score does not necessarily mean that they are duplicate records but it does flag them for further scrutiny.

The PenDataModel accounts for 2572 lines of code developed and an additional 2373 lines of code generated using JAXB and the PDML schema.

### 3.1.3 PDMLEditor

The PDMLEditor was developed to provide a Graphical User Interface (GUI) to work with the PDML. Once the PDML is loaded in the PDMLEditor, all of the records are populated into a tree on the left hand side of the window. If a record is double clicked it opens up an editor window for that record and from there changes can be made and saved for that record. The PDMLEditor also provides some analysis capabilities by producing a matrix of correlation plots for every attribute loaded in the program. It can also produce a correlation matrix without the plots. It can export the data out to other formats for analysis, generate a statistical analysis file, find duplicate records, list out all of the materials for the projectiles and targets in the database, and can filter the database based on several input parameters. An example of the PDMLEditor window can be seen in Figure 3.1. In total, there were 11600 lines of code developed for the PDMLEditor.



F Test(001451): Fugelso and Taylor (1978)(1689)
F Test(001452): Fugelso and Taylor (1978)(40)
F Test(001453): Fugelso and Taylor (1978)(50)
F Test(001454): Fugelso and Taylor (1978)(58)
F Test(001455): Fugelso and Taylor (1978)(59)
F Test(001457): Fugelso and Taylor (1978)(62)
F Test(001458): Fugelso and Taylor (1978)(64)
F Test(001459): Fugelso and Taylor (1978)(66)
F Test(001460): Fugelso and Taylor (1978)(69)
F Test(001461): Fugelso and Taylor (1978)(70)
F Test(001462): Fugelso and Taylor (1978)(71)
F Test(001463): Fugelso and Taylor (1978)(72)
F Test(001464): Fugelso and Taylor (1978)(73)
F Test(001465): Fugelso and Taylor (1978)(74)
F Test(001466): Fugelso and Taylor (1978)(75)
F Test(001467): Fugelso and Taylor (1978)(76)
F Test(001468): Fugelso and Taylor (1978)(78)
F Test(001470): Fugelso and Taylor (1978)(85)
F Test(001471): Fugelso and Taylor (1978)(88)
F Test(001472): Fugelso and Taylor (1978)()
F Test(001473): Fugelso and Taylor (1978)()
F Test(001474): Fugelso and Taylor (1978)()
F Test(001660): Lambert (1978)(80)
F Test(001661): Lambert (1978)(79)
F Test(001662): Lambert (1978)(84)
F Test(001663): Lambert (1978)(83)
F Test(001664): Lambert (1978)(85)
F Test(001665): Lambert (1978)(19)
F Test(001666): Lambert (1978)(18)
F Test(001667): Lambert (1978)(17)
F Test(001668): Lambert (1978)(16)
F Test(001669): Lambert (1978)(15)
F Test(001670): Lambert (1978)(14)
F Test(001672): Lambert (1978)(12)
F Test(001673): Lambert (1978)(47)

Preparing data...

**Test**

Test ID: 001665 □ Suspect
Type: SNF
Notes:

**Source**

Source: Lambert (1978) RefID: RefID0041 Test Number: 19
Distribution: Unknown □ Export Control

**Impact**

Velocity: 879.0 m/s Reported Yaw: NA
Total Yaw: 2.8 deg Reported Pitch: NA

Figure 3.1: A screen capture of the PDMLEditor window

### 3.1.4 BallisticsANN

The BallisticsANN was developed to enable the running of several different types of analysis; first, a run for ANN topology selection; second, a run for ANN training; and, finally, a run for comparison of the ANN to the Segletes model. When run for topology selection, there are three methods that can be used; focused, growth, and pruning. When running focused, the topologies of interest are passed to the program in an input file and each one is run to determine the best topology for the MLP. The growth method (also called constructive) starts with a minimal topology and then trains it until completion. When the topology has completed training, another neuron is added and training continues. This process is repeated until a set network size or if improvement in the network error does not meet a set threshold [69]. The pruning takes an opposite approach from that of growth. Instead of starting with a simple network it starts with a more complex network to begin training. After training has completed, the neuron that is the least significant is removed from the network and training then resumes. This is continued until as set number of neurons have been removed or if the network error increases past a set threshold [70].

The BallisticsANN was implemented to be multi-threaded to take advantage of the 24 cores that were present on the machine that was running it. The main class of BallisticsANN is ANNPerforation and in method main() a Runnable is created and placed in the java.awt.EventQueue using the invokeLater() method. The Runnable creates an instance of class AnnPerforation and then calls the runModel() method. The Runnable loads the PDML database and creates an instance of ModelSelectionRunnable and calls the execute() method. ModelSelectionRunnable is a extension of the SwingWorker class so when it executes it runs in a background thread. ModelSelectionRunnable calls the runFocused() method which sets up a CachedThreadPool ExecutorService. If the user set the thread pool size to 0, then the pool size is determined based on the number of cores available, otherwise it is set to the user defined value. ModelSelectionRunnable then creates a new Runnable for each network model that is to be trained and each one is added to the queue for the ExectorService. As a thread in the pool becomes available, one of the Runnables will be pulled off the queue and begin processing. Each of those Runnables will set up another thread pool for the sub runs for that model. That thread pool is handled the same way as the first. Figure 3.2 provides a visual of how the thread pools are setup in the BallisticsANN.

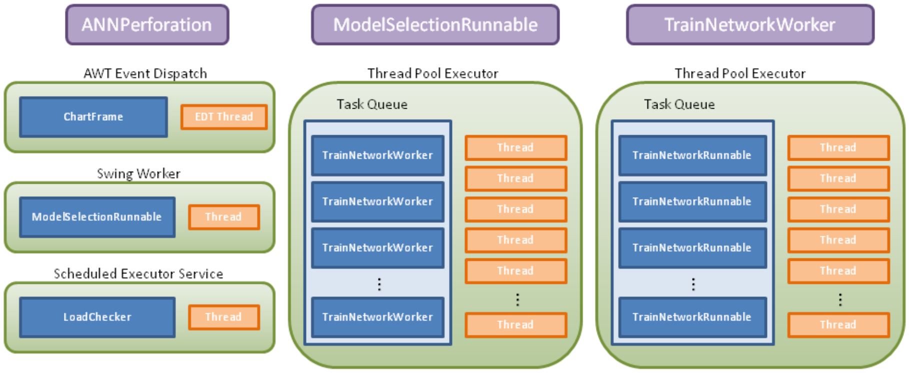


Figure 3.2: Threading of ANNPerforation Project

All of the user supplied inputs for BallisticsANN come from a properties file. Upon starting the program, the properties file is loaded and the variables are set based on the file's contents. Examples of things set within the file are as follows:

**Name** The name of the network; this is used for saving or loading of a network.

**GUI** A boolean value to turn on or off the GUI.

**PDML** A string representing the path to the PDML file.

**Program** An enumeration that sets the run as “Selection”, “Training”, or “Run”.

**Sub_Runs** Sets the number of random sample runs to do for a particular network topology.

**Training_Ratio** Sets the ratio for the split of training data to test data.

**K_Folds** Sets the number of folds to use for the cross-validation.

**Threads** Sets the number of threads to use for the training thread pool. A value of 0, is used to tell the system to determine the best number.

**Minimum_Improvement** This is used to stop training if the error has not improved by this much.

**Tolerated_Epochs** This is used to stop training if the error hasn't improved for this number of epochs.

**Alpha_Verification** This is used to stop training if the generalization error worsens by this percent.

**Maximum_Iterations** This stops training after this many epochs.

**Seed** Used to seed the random number generator.

**Run_Type** Sets the run type to use for the network. Valid values are “FOCUSED”, “GROWTH”, or “PRUNING”.

**GA_Pop** This sets how large the GA population is.

**GA_Iter** This sets how many iterations of the GA will be run.

**Network** This value is set as four integers separated by commas “#,#,#,#”. It sets the input layer, hidden layer 1, hidden layer 2, and output layer neuron counts for the network. If running “FOCUSED” this tag can be repeated for each topology to be run.

The BallisticsANN project accounts for 9203 total lines of code developed. Even though the number of lines of code is less than that for PDMLEditor, the BallisticsANN is much more complex code than any of the other projects developed for this research.

### 3.1.5 ANNPlotter

The final piece of software that was developed for this research was ANNPlotter. ANNPlotter was designed to visually show the neurons and connections of a trained network in such a way that one could easily see how much each neuron and connection were adding to the network performance. Each neuron is colored based on how the average of all of its outgoing connection weights (absolute value) compares to the other neurons in the same layer of the network. Each connection is colored based on the value of its weight; blue for negative and red for positive. The thickness of the connection corresponds to the magnitude of the weight value. Figure 3.3 shows an example of a MLP that was trained for this research. ANNPlotter was a fairly simple software project and only required 1091 lines of code to complete.

## 3.2 Implementation Details

This section will outline the design details that were implemented for the MLP in the software developed for this research. For more general information on these topics see Chapter 2.

### 3.2.1 K-fold Cross-Validation

K-fold cross-validation was used for cross-validating the MLP during the selection process based on the work of Gonzalez-Carrasco et al. [40]. Their paper further found that 10-fold cross-validation was a better choice than 5-fold cross-validation and leave-one-out cross-validation. Based on their finding 10-fold cross-validation was used.

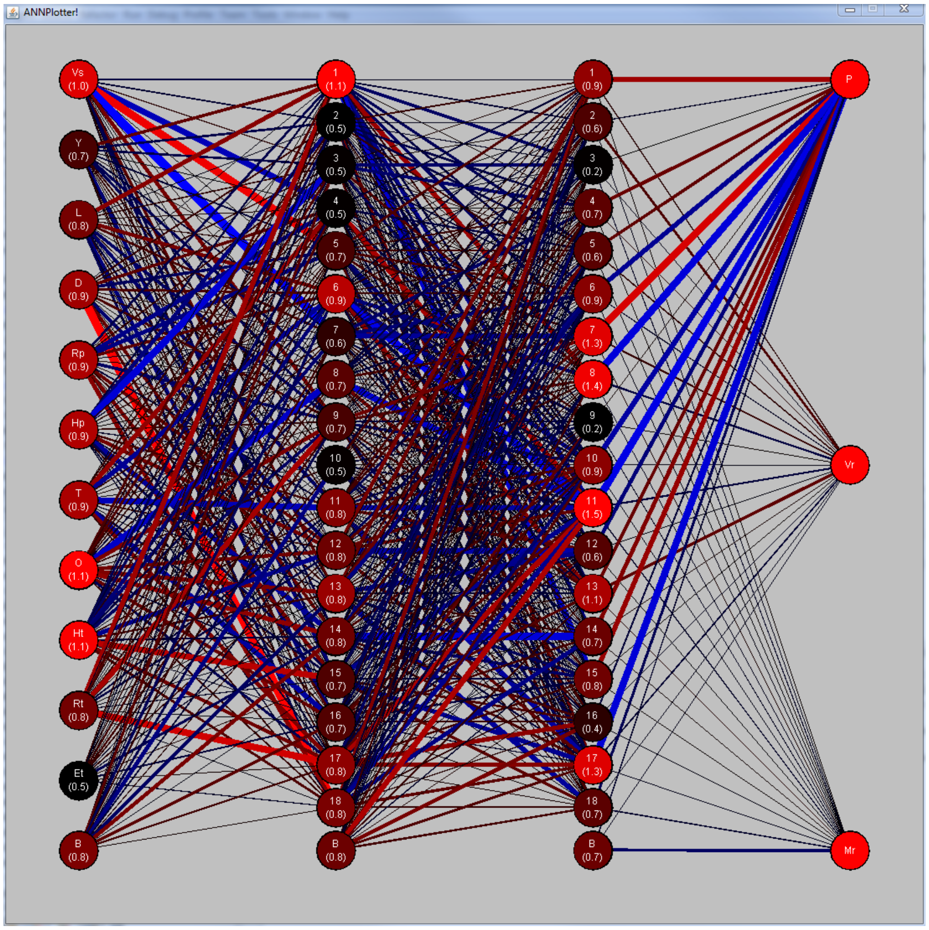


Figure 3.3: A screen capture of the ANNPlotter window

### 3.2.2 Splitting of the data into data sets

The data for this research was split into training, validation, and test sets so that the selected model could be trained, validated, and assessed. Figure 3.4 provides an illustrative view of how the data was split into the various subsets. The top row labeled "Database" represents the entire PDML database. The second row has two sets that represent the initial split of the database into a training/validation set and a test set. The rest of the sectioning of the data depends on the type of run being performed.

#### Topology Selection Run

If the run is for topology selection then the training/validation set is split into 10 equal sized folds of data (fourth row). A network is trained on 9 out of 10 of those folds, with the last fold being a validation fold. In Figure 3.4 network 1 is using the first fold as the validation fold and folds 2 through 10 are being used for training. For network 2 fold 2 is being used for validation and folds 1 and 3 through 10 are being used for training. The validation fold moves to the third, then fourth, fifth, sixth, seventh, eighth, ninth, and, finally, for the last network it will be the tenth fold.

### Training Run

For a training run the training/validation set is split into two subsets, a training set and a validation set. The validation set is used during the training process to stop training of the MLP before overfitting can occur. Although Figure 3.4 shows the validation set as being the same as the tenth fold of data, in reality the splitting of the training/validation set is completely random and is not guaranteed to contain the same data as any of the 10 folds shown.

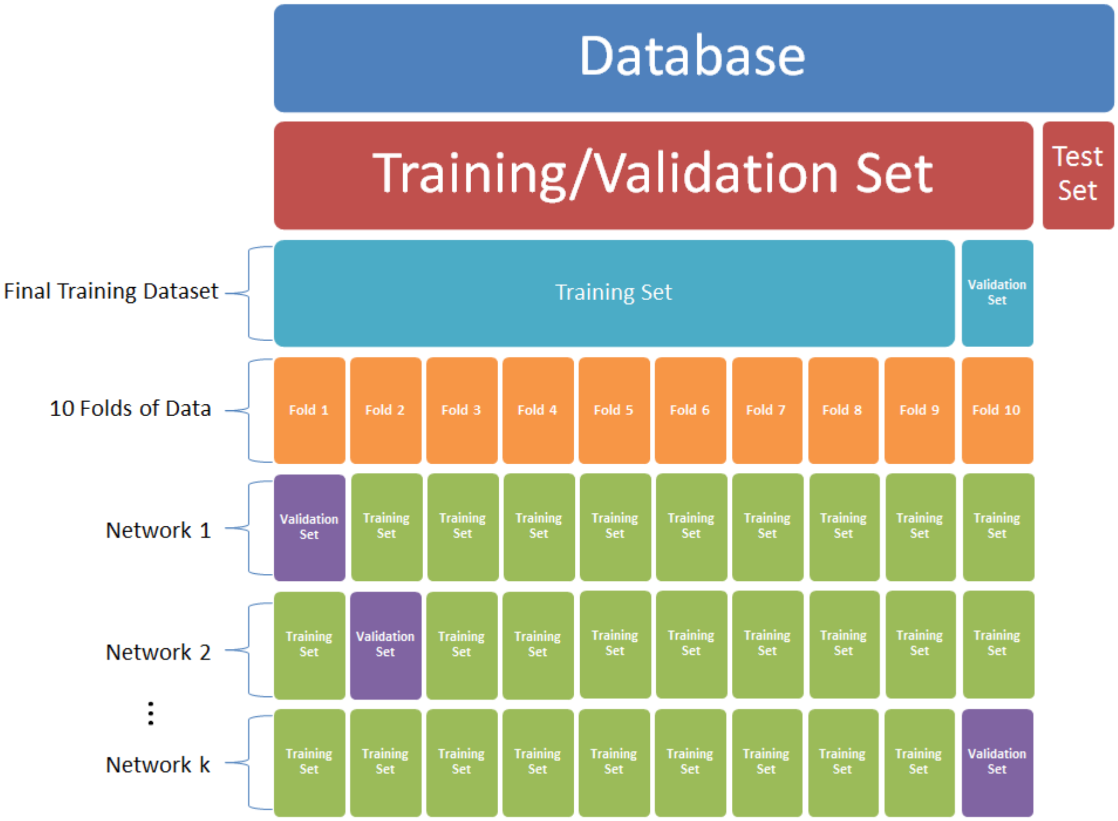


Figure 3.4: How the data was split apart during selection and training phases

### 3.2.3 Early Stopping Strategies

Early stopping was accomplished by using a combination of three methods. The first method was to stop the training process when an epoch limit was reached. The second method was to stop if the error was not improving by some threshold for a set number of epochs. The final method was to stop if the generalization loss was above a set value. The selection phase used the first and second method. The training phase used all three methods. The generalization loss method was only used during the training phase because of the use of k fold cross-validation during the selection phase. Since there are 10 separate networks being trained on different subsets of the data, the average error of those 10 networks could fluctuate frequently, making the use of the generalization loss method problematic.

#### Total Iterations Limit

The total iterations limit is simply a limit on the total number of epochs. If at any point during the training of the network the number of epochs is greater than the limit, training is stopped. If total training time was not a issue, then the total iteration limit could be omitted as a strategy. However, rarely is there a case where small improvement in error is worth the time invested in letting the training run longer.

The main concern that led to the use of a limit for total iterations was that the MLPs were batch run during training and it was thought that there could be a case where the training of one MLP might hold up the completion and saving of the other MLPs. To reduce the risk of this happening, a bound was placed on the total iterations for training the MLPs. In practice, the limit was set at a large enough value (100000) that it was never actually needed.

#### Stagnation

The stagnation approach is used to stop the training of the MLP when the improvement in error is smaller than some threshold for a set number of epochs. After each epoch of training the error from the last epoch is compared with the current error and if it is worse than the threshold, a counter is incremented. If there is an improvement in error and it was better than the threshold, then the counter is reset to 0. If at any point the counter is greater than the set number of tolerated epochs then training is stopped. The typical values used during the training of MLPs for this research, was 0.000001 for the minimum improvement criteria and 400 for the number of tolerated epochs.

### Generalization Loss

The generalization loss method was implemented based on the work of Prechelt [71]. It is calculated by checking the current error against the best error thus far in the training process. The mathematical equation for generalization loss is provided in (3.1). During the training phase an early stopping criteria of 7% was used for the generalization loss method.

$$GL(t) = 100 \cdot \left( \frac{E_{va}(t)}{E_{opt}(t)} - 1 \right) \tag{3.1}$$

Where:

$E_{va}(t)$ is the validation error of epoch $t$

$E_{opt}(t)$ is the lowest validation set error obtained in epochs up to $t$

$GL(t)$ is the generalization loss at epoch $t$

## 3.2.4 Back-Propagation Method

Instead of using Levenberg-Marquart (LM) for the back-propagation algorithm, the decision was made to use the SCG algorithm. The SCG algorithm is also a second-order solver and was already implemented in Encog, thus requiring no extra effort to implement. In addition, in a paper by Batra it is shown that the SCG performs comparable to the LM method in accuracy, but “faired better in terms of speed” [72].

## 3.2.5 Global Optimization

Due to difficulties implementing a genetic algorithm with k-fold cross-validation, a genetic algorithm was not used during the topology selection phase. However, during the training phase a genetic algorithm was used because k-fold cross-validation was no longer in use.

### Genetic Algorithms

The genetic algorithm cannot be run until completion, because it is only being trained on the training set of data. If it was optimized only to the training data there would be a risk of over-fitting. So the genetic algorithm was run for a reduced set of epochs to allow for starting weights that are close to local or global minimums. To increase the likelihood of finding the global minimum, multiple instances of the MLP were created and their weights were assigned based on the results from running a genetic algorithm. The genetic algorithm was used to provide a starting point for the weights that had a higher probability of being close to the global optimum solution [73]. From that starting point a back-propagation method was used to train the MLP and modify the weights to the optimal solution. Once each instance was finished training, the instance with the best validation error was selected as the final MLP model.

### Sub-Sampling of Network Topology

Another method used, in a attempt to train the MLP to the global optimal solution, was to train multiple network instances of the same topology. Each instance of the network was seeded with different starting weights and training progressed separately from any other instances of the network topology. By training more instances of the network topology with each one starting with different random weights, there is a better probability that at least one of those instances will be able to train to the global optimum solution. During the selection phase, 3 instances were created per topology and during the training phase 20 instances were created and trained. For the selection phase the network with the best error was then selected to move forward to the training phase. For the training phase the best network was selected to move forward to the comparison phase with the Segletes model.

## 3.2.6 Cost Function

The cost function used for this research is the L2 norm or Euclidean norm. The L2 norm was found to be the best to use for this type of problem [40].

$$S = \sum_{i=1}^{n} (y_i - f(x_i))^2 \tag{3.2}$$

Where:

| | |
|---|---|
| $S$ | is the sum of the square differences |
| $y_i$ | is the target value |
| $f(x_i)$ | is the estimated value |
| $n$ | is the total number of values |

### 3.2.7 Topology Selection

The topology selection process used for this research involved the selection of a subset of topologies that would would be capable of accurately representing the problem being modeled and not lead to overfitting. The topologies that were used during the selection process were chosen based on the approximation equations published by Tarassenko [15]; they are provided in (3.3) and (3.4). In (3.3), $n$ is the number of training data and $W$ is the total number of network parameters (the network parameters are the weights associated with the connections between the nodes in the MLP) that must be adjusted during training. The parameter $W$ can be obtained by using (3.4), where $N$ is the number of layers in the MLP topology and $L_i$ is the number of neurons in the $i^{th}$ layer. Prior to starting the selection process the number of data points available for training was already known (1877), so a matrix of scenarios of neuron counts for hidden layer 1 and hidden layer 2 was created and a small subset of the topologies that would satisfy (3.3) and (3.4) were chosen for the selection process.

$$W \leq n \leq 10W \tag{3.3}$$

$$W = \sum_{i=1}^{N-1} (L_i + 1)L_{i+1} \tag{3.4}$$

Where:

| | |
|---|---|
| $W$ | is the total number of network parameters |
| $n$ | is the number of training data |
| $N$ | is the number of layers in the ANN |

$L_i$ is the number of neurons in the $i^{th}$ layer

### 3.2.8 Data Quality and Cleaning

Typical problems with using large amounts of data include incorrect recording, incorrect data entry, duplication, and missing parameters.

Data preparation was a large part of the effort for this research, partly because of how important it is to have good data for any type of regression and partly because the data for this research was in such poor condition. In order to fully understand the process used to clean and prepare the data, one has to understand the problem itself. Hellerstein [74] lists four typical types of errors that occur in a database: data entry errors, measurement errors, distillation errors, and data integration errors. Data entry errors can be caused by keying, selection, formatting, spurious data, or omission mistakes [75]. Measurement errors consist of errors that occur during the process of measuring something that was then entered into the database. Distillation errors come from the preprocessing of raw data before entry into the database. For both of those errors the data was entered correctly, but the number being entered had errors. Data integration errors come from the integration of data from various sources or other databases when the assumptions associated with that data are not well understood.

Oliveira, Rodrigues, and Henriques [76] break down data quality problems into what level of information is needed to detect those problems. They break them into the following groupings; "An Attribute Value of a Single Tuple", "The Values of a Single Attribute", "The Attribute Values of a Single Tuple", "The Attribute Values of Several Tuples", "Relationships among Multiple Relations", and "Multiple Data Sources". Each one builds more complexity onto the previous one and they can be used to detect errors in phases.

The first phase of cleaning involved checking each field value for errors. By analyzing only the field values themselves ten data quality problems can be detected; they are provided below [77]:

- **Missing value** – The field is empty and the data is missing.

- **Syntax violation** – The format of the value in the field does not match that which is defined for it.

- **Outdated value** – The value in the field is out of date.

- **Interval violation** – The value in the field is out of the defined bounds.

- **Set violation** – If the field is an enumerated field then the value is not a part of the enumerated set.

- **Misspelled error** – There was a keying or wrong spelling entered during data entry.

- **Inadequate value to the attribute context** – The entry in the field should be in another field.

- **Value items beyond the attribute context** – More than one entry is in the field and part of it belongs in another field.

- **Meaningless value** – The entry in the field does not make sense for this field or any other field in the record.

- **Value with imprecise or doubtful meaning** – Lose of precision in a field due to abbreviations.

- **Domain constraint violation** – A violation of constraint related with the attribute, inherent to the domain.

After all of the individual fields were checked for errors, the search was expanded to look at the values for each particular attribute across all of the records. By analyzing the values of a particular attribute across all of the records, two additional data quality problems can be detected [77]:

- **Uniqueness value violation** – Two or more records representing different entities have the same value in an attribute that is supposed to be unique.

- **Synonyms existence** – Arbitrary use of syntactically different values with the same semantic meaning.

After the attributes were checked for errors, the search was expanded to look at the values for each attribute for each particular record. By analyzing the values of all of the attributes across a particular record, two additional data quality problems can be detected [77]:

- **Semi-empty tuple** – In this situation, a great number of tuple attributes are not fulfilled. If a given threshold (user defined) is surpassed the tuple is classified as semi-empty.
- **Inconsistency among attribute values** – There is a violation to an existing dependence among values of the tuple attributes.

By analyzing the values of all of the attributes across all of the records, two additional data quality problems can be detected [77]:

- **Redundancy about an entity** – The same entity is represented by an equal or equivalent representation in more than one tuple.
- **Inconsistency about an entity** – There are inconsistencies or contradictions among one or more attribute values of a same entity, represented in more than one tuple.

By analyzing the values of all of the attributes across all of the records and across multiple relationships, five additional data quality problems can be detected [77]:

- **Referential integrity violation** – In a tuple attribute which is foreign key there is a value that does not exist as primary key in the related relation.
- **Outdated reference** – In spite of referential integrity be respected, the foreign key value of a tuple is not updated and does not correspond to the real situation.
- **Syntax inconsistency** – Depending on the relation, there are different representation syntaxes among attributes whose type is the same.
- **Inconsistency among related attribute values** – There are inconsistencies among attribute values from relations where a relationship exists between them.

- **Circularity among tuples in a self-relationship** – It corresponds to cycle situations among two (direct circularity) or more (indirect circularity) related tuples in a self/reflexive-relationship.

By analyzing the values of all of the attributes across all of the records, across multiple relationships, and multiple sources, eight additional data quality problems can be detected [77]:

- **Syntax inconsistency** – Depending on the data source, there are different representation syntaxes among attributes whose type is the same.

- **Different measure units** – Depending on the data source, different measure units are used in attributes that are related.

- **Representation inconsistency** – Different sets of values, from the same type or not, are used in related attributes from distinct data sources to represent the same situations.

- **Different aggregation levels** – The detail level presented in different data sources by equivalent relations is not the same.

- **Synonyms existence** – Use of syntactically different values with the same semantic meaning in related attributes from distinct data sources.

- **Homonyms existence** – Use of syntactically equal values but with different semantic meaning in related attributes from distinct data sources.

- **Redundancy about an entity** – The same entity is represented by an equal or equivalent representation in more than one tuple from different data sources.

- **Inconsistency about an entity** – There are inconsistencies or contradictions among one or more attribute values of a same entity, represented in more than one tuple in different data sources.

### 3.2.9 Analysis Methods

The following subsections will detail the various methods used to analyze the results of the MLP. It is important to realize that each of these methods look at different parts of the performance of the model and that there is no single method that can determine the best model by itself. The overall performance of the model is evaluated using a combination of all of these methods.

#### Mean Squared Error

The first measure that will be used to compare the performance of the two models is the Mean Squared Error (MSE), which is defined in (3.5).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2 \tag{3.5}$$

Where:

$\hat{Y}_i$ is a vector of $n$ predictions

$Y_i$ is a vector of $n$ observed values

$n$ is the number of tested inputs with corresponding observations and predictions

#### Accuracy, Precision, False Positives, and False Negatives

One of the predictions that the MLP is expected to make is whether or not a perforation occurs as a result of the ballistic interaction. That question is a classification problem and what follows are measures that can be used to address performance of the MLP. Accuracy (ACC) is used to measure the performance of a classifier and is defined as the number of correctly classified items divided by the total number of items and is defined in (3.6). The other measures are the false positive rate (FPR) provided in (3.7), the false negative rate (FNR) provided in (3.8), the true positive rate (TPR) provided in (3.9), and the true negative rate (TNR) provided in (3.10) [78].

$$ACC = \frac{a + d}{a + b + c + d} \tag{3.6}$$

$$FPR = \frac{b}{a + b} \tag{3.7}$$

$$FNR = \frac{c}{c+d} \tag{3.8}$$

$$TPR = \frac{d}{d+c} \tag{3.9}$$

$$TNR = \frac{d}{d+c} \tag{3.10}$$

$$MCC = \frac{d \times a - b \times c}{\sqrt{(d+b)(d+c)(a+b)(a+c)}} \tag{3.11}$$

Where:

$a$ is the number of true negatives

$b$ is the number of false positives

$c$ is the number of false negatives

$d$ is the number of true positives

Table 3.1 provides a visual representation of the sets (a, b, c, d) of data. ACC is the percent of test outcomes that were correctly predicted by the model. FPR is the percent of non-perforation test outcomes that were incorrectly predicted by the model to be perforations. FNR is the percent of perforation test outcomes that were incorrectly predicted by the model to be non-perforations. TPR is the percent of perforation test outcomes that were correctly predicted by the model to be perforations. TNR is the percent of non-perforation test outcomes that were correctly predicted by the model to be non-perforations. The previous measurements can give misleading results if the data or the model predictions are skewed in one direction or another. The Matthews Correlation Coefficient (MCC) [79] is generally regarded as being one of the best measures to use for describing the results from a confusion matrix [80] and is provided in (3.11). The MCC varies from -1 (worst) to 1 (best).

Table 3.1: Confusion matrix

| Model Outcome: | | Data |
|---|---|---|
| Non-Perf | Perf | Result |
| (a) | (b) | Non-Perf |
| (c) | (d) | Perf |

### Data Point-by-Data Point Prediction Comparison

A comparison of both models for each test is performed by categorizing the outcome as one of three choices:

**MLP:** This was used when the MLP performed better than the Segletes model.

**Segletes:** This was used when the Segletes model performed better than the MLP.

**Same:** This was used when both models performed the same.

A model was considered better if one of two things occurred. First, if it had predicted the perforation correctly and the other model had not. Second, if it had a lower error, when both models predicted perforation correctly. The models were considered to have performed the same if one of two things occurred. First, if they both predicted perforation incorrectly. Second, if they both predicted the same error, when both models predicted perforation correctly.

Once each test was categorized, each category was tallied. The metric of interest was the percent of tests that the MLP or the Segletes model performed as well or better than the other model. So the tally for the particular model of interest is added to the tally for the category "Same" and then divided by the total tally of all of the categories, to get the percent of tests where that model did as well as the other model or better.

### Error Analysis

The first type of error analysis performed is the Percentage Error (PE) (also written as %Error) and it was calculated for each test using (3.12). The equation is undefined when, $O = 0$. In an attempt to address this concern a few categories of outcomes were created. If the observed outcome is a non-perforation then the values of $M_r$ and $V_r$ will be 0; this also corresponds with the confusion matrix outcomes (Table 3.1) of (a) and (b). For the case where the MLP predicted a non-perforation (case (a) from the matrix) the %Error is set as 0% since the model was correct and there is no error in the prediction. However, if the MLP predicted a perforation (case (b) from the matrix) then that data point is categorized as an incorrect prediction of perforation (false positive). If nothing else was done to the data this would skew the end results because nothing was done to address cases (c) and (d) from the matrix. So, for a situation where the observed value is not 0, corresponding to a perforation, and the MLP predicted a non-perforation (case (c) from the matrix) the data point is categorized as an incorrect prediction of non-perforation (false negative). For the last case where both observed and predicted are perforations (case (d) from the matrix), %Error is calculated using (3.12). The mean of all of the calculated PEs is called the Mean Percentage Error (MPE) and is provided in (3.13).

$$PE = 100 \cdot \left( \frac{O - P}{O} \right) \tag{3.12}$$

$$MPE = \frac{100}{n} \cdot \sum_{i=1}^{n} \left| \frac{|O_i - P_i|}{O_i} \right| \tag{3.13}$$

Where:

$n$ is the number of data points

$O$ is the observed value

$O_i$ is the $i^{th}$ observed value

$P$ is the predicted value

$P_i$ is the $i^{th}$ predicted value

One of the problems with using the MPE is that negative errors can offset positive errors in the result. Table 3.2 gives an example of two models, their predictions, and their associated PEs. In this example the PEs for Model A are much larger in magnitude then those for Model B. However if the MPE was the criteria for choosing the best model then Model A would be the model selected since its MPE is 1.5% compared to 1.625% for Model B.

To address the concern of the negative and positive values offsetting one another, the absolute value of the error can be used. The Absolute Percentage Error (APE) is the result of modifying (3.12) to use the absolute value and it is provided in (3.14). The mean of all of the calculated APEs is called the Mean Absolute Percentage Error (MAPE) and is provided

Table 3.2: MPE and MAPE Example

| Observed | Model A Prediction | Model B Prediction | Model A PE | Model B PE |
|---|---|---|---|---|
| 122 | 97.6 | 125.66 | 20% | -3% |
| 70 | 59.5 | 65.1 | 15% | 7% |
| 156 | 121.68 | 149.76 | 22% | 4% |
| 153 | 107.1 | 154.53 | 30% | -1% |
| 146 | 197.1 | 148.92 | -35% | -2% |
| 39 | 41.73 | 39.39 | -7% | -1% |
| 36 | 40.32 | 34.56 | -12% | 4% |
| 67 | 81.07 | 63.65 | -21% | 5% |
| | | MPE | 1.5% | 1.625% |
| | | MAPE | 20.25% | 3.375% |

in (3.15). For the example in Table 3.2 it is more clear looking at the MAPE that Model A is not as good as Model B. MAPE is very popular as a measure for forecast accuracy, this is due largely to its simplicity and ease of understanding [81].

$$APE = 100 \cdot \left( \frac{|O - P|}{O} \right) \tag{3.14}$$

$$MAPE = \frac{100}{n} \cdot \sum_{i=1}^{n} \left| \frac{|O_i - P_i|}{O_i} \right| \tag{3.15}$$

There is one very important issue that can arise from using MAPE in analysis, it is asymmetric. For example, for cases where the observed value is low or near zero, the percent error equations can return very large negative values if the predicted value is bigger than the observed value. However for positive errors, it can never be larger than 100% error. For example, imagine a case where the observed value is 0.2 g and the predicted value is 2 g, then the percent error is $-900\%$. That error is very large even though a prediction of residual mass of 2 g for a Kinetic Energy Projectile (KEP) is pretty good when the observed is 0.2 g. If the values are flipped with the observed equal to 2 g and predicted equal to 0.2 g, then the percent error is 90%. That error is not that large when compared to the earlier example of $-900\%$. According to Kolassa and Martin, “one important problem that has not received adequate attention which arises when MAPE is used as the basis for comparing different methods or systems: using the MAPE for comparisons rewards methods that systematically under-forecast This problem is poorly understood both among academic forecasters and practitioners in industry" [82].

Table 3.3: MAPE and SMAPE Example

| Observed | Model A Prediction | Model B Prediction | Model A PE | Model B PE | Model A SAPE | Model B SAPE |
|---|---|---|---|---|---|---|
| 1 | 0.91 | 1.4 | 9% | -40% | 4.7% | 16.7% |
| 5 | 4.5 | 5.5 | 10% | -10% | 5.3% | 4.8% |
| 2 | 1.84 | 2.5 | 8% | -25% | 4.2% | 11.1% |
| 20 | 17.6 | 21 | 12% | -5% | 6.4% | 2.4% |
| 25 | 22.5 | 26.25 | 10% | -5% | 5.3% | 2.4% |
| 39 | 34.71 | 39.39 | 11% | -1% | 5.8% | 0.5% |
| 36 | 32.4 | 36.36 | 10% | -1% | 5.3% | 0.5% |
| 67 | 58.29 | 67.67 | 13% | -1% | 7% | 0.5% |
| | | | MAPE | | SMAPE | |
| | | | 10.378% | 11% | 5.48% | 4.86% |

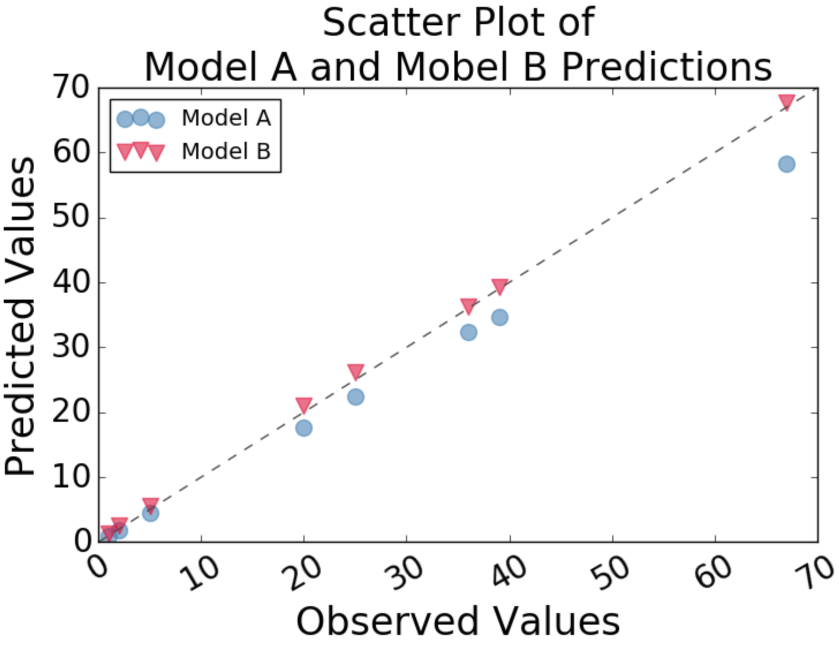


Figure 3.5: Scatter Plot of MAPE and SMAPE Example

The effect of asymmetry in MAPE can be seen in the example in Table 3.3. In the example, Model A has a tendency to under predict by about 10% for all values of concern. Model B has very large negative errors for cases where the observed value is small. For the case where the observed value is equal to 1, a PE of -40% appears to be very bad, but amounts to a difference of only 0.4 from the observed value. The MAPE places too much emphasis on low valued cases and prefers models that under predict [83, 84]. This can be seen in the MAPE values; Model A has a value of 10.375% and Model B had a value of 11%. If MAPE was used to select the best model then the choice would be Model A. Figure 3.5 is a scatter plot of the observed versus predicted values for Model A and Model B. In the plot the diagonal dashed line represents a perfect prediction, so the closer to the line the better the performance of a model. Model A has a tendency to under predict and the magnitude of the error gets larger as the values get larger. Model B is very close to the line for all of the values that are plotted. The MAPE scores Model A as better because of Model B's large percent error for the initial small values, but looking at the plot it should be evident that Model B is the better model.

The concern of asymmetry in MAPE has led to the development of Symmetric Mean Absolute Percentage Error (SMAPE). The version of SMAPE provided in (3.17) has a lower bound of 0% and an upper bound of 100%, and "offers a well designed range to judge the level of accuracy and should be influenced less by extremes" [81]. The equation is undefined when both $O = 0$ and $P = 0$, so for those situations the value is set to 0% since the prediction is correct. Unlike MAPE, SMAPE is defined for cases where only $O = 0$ or $P = 0$, and for those situations the equation sets the error percent to 100%.

$$SAPE = 100 \cdot \frac{|P - O|}{|O| + |P|} \tag{3.16}$$

$$SMAPE = \frac{100}{n} \cdot \sum_{i=1}^{n} \frac{|P_i - O_i|}{|O_i| + |P_i|} \tag{3.17}$$

As mentioned before, the absolute value was added in to alleviate the issue of negative values offsetting positive values in the calculation of the mean percentage error. However, when looking only at the individual data points there is no concern of how the value will aggregate. Removing the absolute value from the equation for the individual data points will allow for analysis of the type of error, over prediction or under prediction. For the purposes of this paper, the newly modified equation is called Symmetric Percentage Error (SPE) and is defined in (3.18).

$$SPE = 100 \cdot \frac{(O - P)}{|O| + |P|} \tag{3.18}$$

**Stochastic Dominance**

Stochastic Dominance (SD) is a form of stochastic ordering. SD is used in decision analysis to refer to situations where one prospect (a probability distribution over possible outcomes) can be ranked as superior to another prospect. SD is based on preferences regarding outcomes and the probability of those outcomes occurring [85, 86]. The outcomes used in this research are bins, ordered from lowest (left) to highest (right) absolute value percent error. The worst case bin is located on the far right and represents cases where there were false positives or false negatives. By ordering the bins in this fashion the preferred outcomes are farthest to the left and the least preferred outcomes are on the far right of the probability distribution.

$$F_X(z) = \int_{-\infty}^{z} f_X(u)\,\mathrm{d}u \tag{3.19}$$

The first-order SD is the Cumulative Distribution Function (CDF) and can be calculated using (3.19), defined in terms of the Probability Density Function (PDF) $f$. Higher-order dominance is defined using iterated integrals of the distribution given by the recursive sequence $D^s$ for $s = 1, 2, 3, \ldots,$ as shown in (3.20).

$$\begin{gathered} D^1(z) = F(x) \\ D^s(z) = \int_{-\infty}^{z} D^{s-1}(u)\,\mathrm{d}u = \frac{1}{(s-1)!}\int_{-\infty}^{z} (z-u)^{s-1} f(u)\,\mathrm{d}u \end{gathered} \tag{3.20}$$

The probability distributions being compared in this research are empirical and therefore discrete. An equation is needed to approximate $D^s$ for sample datasets. The SD for a sample datasets is calculated using (3.21), where $x_i$ is the $i^{th}$ bin of the empirical PDF.

$$D_n^s(X_k) = \frac{1}{(s-1)!}\sum_{i=1}^{k}(X_k - X_i)^{s-1} f_i \tag{3.21}$$

Higher n-order SD values are calculated by integrating the PDF $f$ over the bins and weighting the bin values to the left more than those to the right on the x-axis. Once the SD values are calculated, they need to be compared to determine which prospect is dominant. For two prospects, $A$ and $B$, we say that $A$ dominates $B$ at order $s$, if $D_A^s(z)$ is greater than or equal to $D_B^s(z)$ for all possible values of $z$ (see 3.22). In other words, the graph $D_A^s(z)$ lies above or at the graph of $D_B^s(z)$. For first order SD, that would mean that the CDF graph $F_A(z)$ lies above or at the CDF graph of $F_B(z)$.

$$\forall z, D_A^s(z) - D_A^s(z) \geq 0 \tag{3.22}$$

By definition, if a prospect dominates at a lower order, then it will also dominate at a higher order. This research compares the performance of the models by comparing each models first order SD. If neither model is dominate at the first order, then a comparison at the second order is done. This can be continued to $n^{th}$ order, but general practice typically does not go past third order and even the use of third order is limited (Personal communication, Dr. Joseph Collins of U.S. Army Research Laboratory Statistics Analysis Team, September 23, 2010). For comparing the models in this research, only first order SD will be used. This research is concerned with obvious dominance of one model over the other, so an ambiguous answer at the first order SD level will be recorded as such.

### Stochastic Dominance Examples

SD provides a method for comparing PDFs and is especially useful when no PDF is clearly better. Figure 3.6 provides an example of three possible PDFs for comparison. In the figure, Option 2 starts off with a higher probability in the first bin than the other two options, but then falls below Option 3 until the third bin. In this example, it is not clear which option is providing the highest probability earliest in the bins or the most reward (higher probability) for the lowest risk (earliest bins).

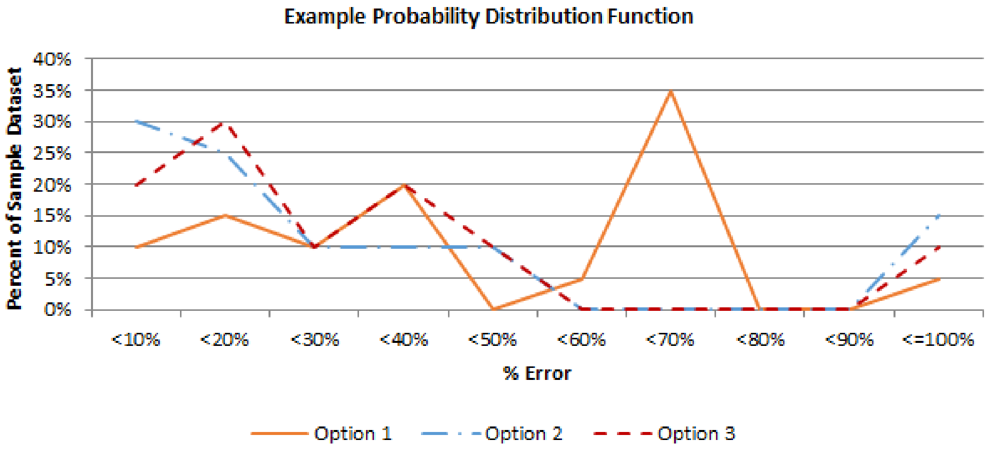


Figure 3.6: Example PDFs for explanation of SD

First order SD is equivalent to the CDF of a PDF. The first order SD for all three options is plotted in Figure 3.7 and can be used to compare the three options in the example we are examining. In the figure, it appears that Option 2 and Option 3 are better than Option 1, but it is still not clear since at some point (bin 7) Option 1 has a higher cumulative probability than the other two.

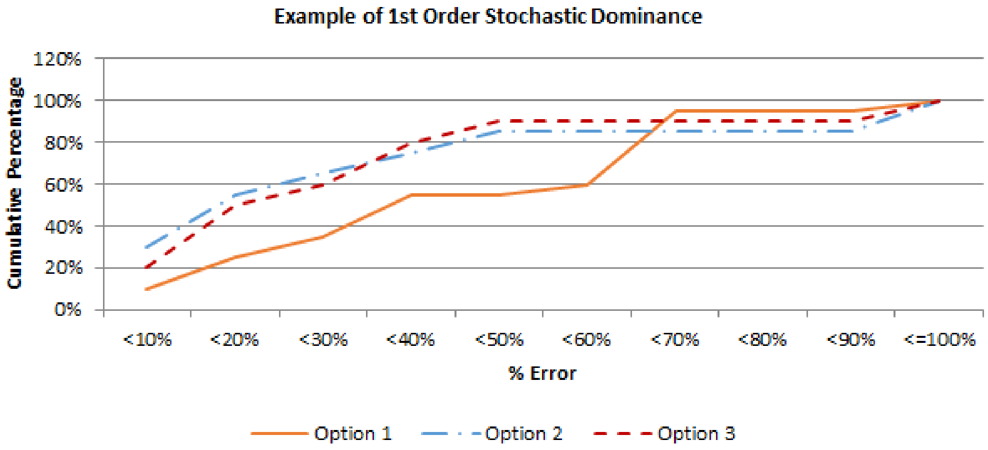


Figure 3.7: Example of first order SD (CDFs)

Since it is not clear which option is the best using first order SD, the next step is to use second order SD. Figure 3.8 is a chart of second order SD for the example we are examining. The second order SD figure confirms that Option 1 is not the best option because it never has a higher value than the other options. We now know that Option 2 and Option 3 are better than Option 1, but unfortunately we still do not know if Option 2 or Option 3 is the best option.

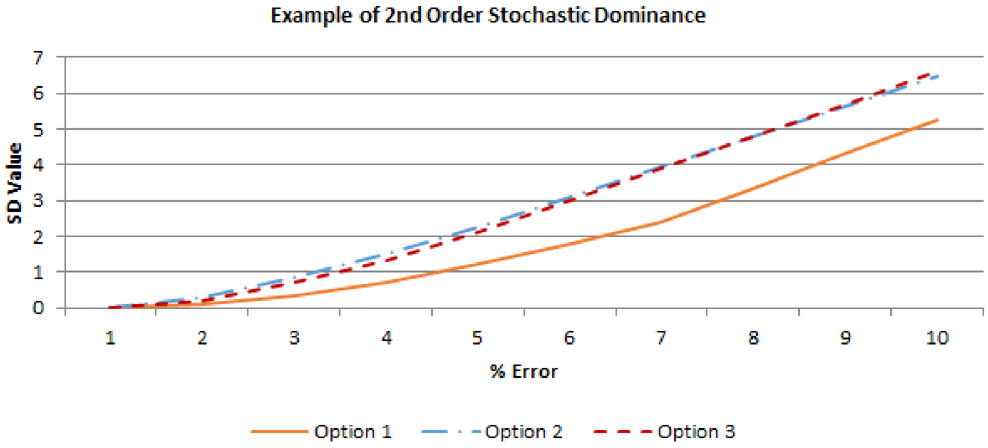


Figure 3.8: Example of second order SD

The final order of SD used for this example is the third order SD. Figure 3.9 is a chart of the third order SD for this example. Finally, at the third order SD, Option 2 is equal to or greater than the other options across all of the bins. Therefore, Option 2 has third-order stochastic dominance over the other options.

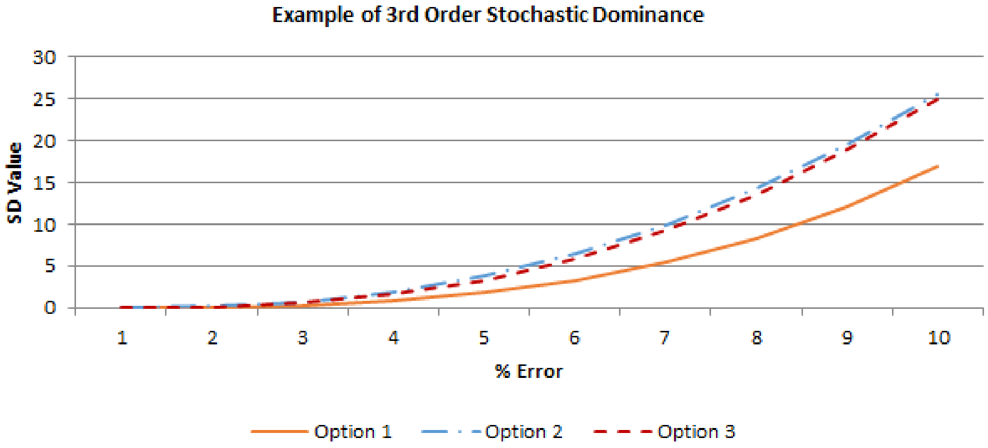


Figure 3.9: Example of third order SD

## 3.3 Summary

This chapter provided an overview of the software developed and modified for this research. Due to the uniqueness of this research, much of the software used had to be developed; in all there were roughly 25 000 lines of code developed for this research. That number doesn't even count the XML schema developed or the visual basic for applications (VBA) code used in Excel for processing the data from the QBasic database.

Also presented in this chapter are the implemented design choices for the MLP and the analysis methods used to compare the MLP and the Segletes model. Of particular note is the use of SPE in the comparison of the two models; the traditional usage of PE is flawed, in that it tends to favor models that under-predict [82].

# Chapter 4

# Experimental Test Data

The most difficult and time consuming effort of this research was in the acquisition and preparation of the experimental test data that was used to train, validate, and test the Multi-Layered Perceptron (MLP). There simply was not enough publicly available ballistic test data available to complete this research. Further work was needed to acquire limited distribution experimental test data to fill in many of the data gaps. There also exists a large amount of classified experimental test data available for future work that was not utilized due to the risk of accidental classified data release.

This chapter will focus on the database design, data acquisition, and data preparation performed for this research. The work detailed in this chapter was used to answer RQ1.1.

## 4.1 Database Design

When designing the database an attempt was made to allow for the storage of any information that might be useful to future analyses. The database was designed using an Extensible Markup Language (XML) schema (the schema is provided in Appendix B). The XML format that the schema defines is called the Penetration Database Markup Language (PDML). The PDML root element is a *PDML* element tag, all of the *Test* elements fall under that tag. Each *Test* element has five required and one optional sub-elements: *Source*, *Impact*, *Projectile*, *Target*, *Results*, and *Notes*. The *Notes* element is used to record any qualitative information of interest for that particular test. The sections that follow will briefly describe each of the sub-elements. A top level diagram of the PDML can be seen in Figure B.1 in Appendix B.

Before discussing each of the top level elements of PDML, it is worth defining what a *Metric* is in the PDML. The quantifiable data stored in the PDML is stored as a complex data type called a *Metric*. A *Metric* has two attributes and the content of the element is defined as a double value. The content is the value of the data it is storing (e.g. velocity, pressure, mass, and the like). The first attribute is *Units*, it is used to store the units associated with the content value. The second attribute is *Pedigree*, it is used to provide information on the origination/legacy of the value that is stored. The *Notes* element for the test can be used to provide further details about the pedigree. *Pedigree* is a string value and is restricted to the values listed below:

- Reported
- Converted
- Estimated
- Imputed
- Surrogated
- NA

PDML allows for a *Metric* to be of any type of unit. However, for simplicity of implementation during this research, all units were limited to those listed below. Conversion to values of like units was handled inside of the software developed to run the MLP. The units used within the PDML are:

- Length – Millimeter (mm)

- Hardness – Brinell Hardness (BHN)

- Mass – Grams (g)

- Time – Seconds (s)

- Velocity – Meters per second ($^{m}/_{s}$)

- Density – Grams per cubic centimeter ($^{\mathrm{g}}/_{\mathrm{cm}^3}$)
- Angle – Degrees (°)
- Pressure – Gigapascals (GPa)

### 4.1.1 Test

The *Test* element contains three attributes: *ID*, *Type*, and *Suspect*. The *ID* attribute contains a 6-digit unique key that identifies that particular test. The *Type* attribute defines the type of test using a 3-character identifier. The *Suspect* attribute is a boolean value that marks that test as a valid test or one that is suspect due to missing data, corrupt data, and the like.

The experimental test data was categorized into three main types; semi-infinite, finite, and limit test data. Semi-infinite test data comes from a penetration test into a material that is of such thickness that the area of plastic deformation in front of the projectile is not expected to reach the rear face of the target. By definition if it was not a semi-infinite target then it was considered a finite target. Finite test data comes from a test where the target material is of a finite thickness and under certain circumstances the projectile could perforate the target. Limit test data comes from many finite test series to determine at what velocity perforation would occur 50% of the time; this is known as the ballistic limit or $v_{50}$. The tests were further broken down by the type of target that was used in the test.

The following are the classifications used to define a test type in the database:

**SNI:** Consists of a single target element, containing no air gaps, with the final element having a semi-infinite target thickness

**SNF:** Consists of a single target element, containing no air gaps, with the final element having a finite target thickness

**MAI:** Consists of multiple target elements, containing air gaps, with the final element having a semi-infinite target thickness

**MAF:** Consists of multiple target elements, containing air gaps, with the final element having a finite target thickness

**MNI:** Consists of multiple target elements, containing no air gaps, with the final element having a semi-infinite target thickness

**MNF:** Consists of multiple target elements, containing no air gaps, with the final element having a finite target thickness

The first letter signifies if the target in that particular test had a (S)ingle armor element or (M)ultiple. The second letter signifies if the target for that particular test had an (A)irgap or (N)o air gap. The last letter signifies if the last target element from that test was semi-(I)nfinite or (F)inite.

### 4.1.2 Source

Within the *Test* element there must be one and only one *Source* element. The *Source* element is used to store information about the source of the data, it contains four attributes and one text content. The text content is typically the authors' names and the year of publication, for example "Allen and Rogers (1961)". The first attribute is *BibtexRefID*, it is a 9-character alphanumeric that identifies the source using a BibTeX reference ID key. The second attribute is *TestNumber*, it is a string value that can store any identifier that the original author used to identify that test. The third attribute is *Distribution*, it is used to store one of the following distribution categories:

- Unknown
- A. Approved for Public Release
- B. U.S. Government Agencies Only
- C. U.S. Government Agencies and Their Contractors
- D. DoD and DoD Contractors Only
- E. DoD Components Only
- F. Further Dissemination Only as Directed by the DoD Controlling Office or Higher DoD Authority

The fourth attribute is *ExportControl*, it is a boolean value that marks the source and data as being export controlled or not. The *Distribution* and *ExportControl* attributes help with the tracking of the releasability of the data used in this research.

### 4.1.3 Impact

Within the *Test* element there must be one and only one *Impact* element. The *Impact* element is used to store data pertaining to the impact conditions of the test, it contains four sub-elements. The sub-elements are *Velocity*, *TotalYaw*, *Yaw*, and *Pitch*. Each of those elements are defined as a *Metric* data type. A diagram of the *Impact* element from the PDML can be seen in Figure B.2 in Appendix B.

### 4.1.4 Projectile

Within the *Test* element there must be one and only one *Projectile* element. The *Projectile* element is used to store data pertaining to the projectile used in the test, it contains twenty four sub-elements. The sub-elements are *Material*, *Hardness*, *Density*, *Ductility*, *YoungsModulus*, *YieldStrength*, *UltimateTensileStrength*, *PoissonRatio*, *Toughness*, *TotalLength*, *EffectiveLength*, *CoreLength*, *Core Diameter*, *Mass*, *Nose*, *Fineness*, *EffectiveFineness*, *NoseLength1*, *NoseConeAngle1*, *NoseDiameter*, *NoseLength2*, *NoseConeAngle2*, *CRH*, and *NoseType*. The first nine elements are used to define the material mechanical properties of the projectile. The rest are used to define the dimensions of the projectile. A diagram of the *Projectile* element from the PDML can be seen in Figure B.3 in Appendix B.

### 4.1.5 Target

Within the *Test* element there must be one and only one *Target* element. The *Target* element is used to store data pertaining to the target used in the test, it contains one attribute and one sub-element. The attribute is *Elements* and it is used to define how many elements the target contains. The *Test* element contains one sub-element, which is the *Element* element. *Target* must have one *Element*, but it can also contain many of them as well.

The *Element* element is used to store the specifics of each element that makes up the target, it contains two attributes and eleven sub-elements. The first attribute is *Number* and it is used to store an integer representing the element's numerical order in the target. The second attribute is *Type* and it is used to record a string representing the element type (e.g. air, metallic, ceramic, or composite). The sub-elements are *Thickness*, *Obliquity*, *Material*, *Hardness*, *Density*, *Ductility*, *YoungsModulus*, *YieldStrength*, *UltimateTensileStrength*, *PoissonRatio*, and *Toughness*.

A diagram of the *Target* and *Element* elements from the PDML can be seen in Figure B.4 in Appendix B.

### 4.1.6 Results

Within the *Test* element there must be one and only one *Results* element. The *Results* element is used to store data pertaining to the results from the test, it contains seven sub-elements. Those elements are *Penetration*, *ResidualVelocity*, *ResidualLength*, *ResidualMass*, *CraterDiameter*, *CraterVolume*, and *LimitVelocity*. A diagram of the *Results* element from the PDML can be seen in Figure B.5 in Appendix B.

## 4.2 Data Acquisition

The initial population of data into the database came from a Southwest Research Institute (SwRI) report [87]. The report was digitally scanned and then processed using Optical Character Recognition (OCR). The data from the report were cleaned and formatted into something that was readable by a Java program. The Java program then pulled the data into the database and wrote it out in the PDML format. The record count in the database after the initial population of data was 2227. The age of the data is older, but since the basic design of Kinetic Energy Projectiles (KEPs) has changed relatively little over the years and even for the slightly different designs, the basic process of penetration is the same, the data is still relevant.

Through extensive searching for test data in reports and journal articles, six more sources [88–93] of data were found and entered by hand into the database. In total, those six documents added 371 more records to the database, bringing the total to 2598. Out of those 2598 records, there are 421 valid SNF test records and 19 valid MNF test records.

Through discussions with other Subject Matter Experts (SMEs) it was discovered that there was an old QBasic database of KEP data that had become unusable due to lack of support for QBasic and lack of maintenance of the database code. An offer was made to repair the old database in exchange for access to the data for this research and a verbal agreement was made. The old database was developed by the Weapons Material Research Directorate (WMRD) back in 1980s for storing data from KEP test events. The database was spread out over 306 QBasic binary files and the data was accessed using several different QBasic program files.

The program files were studied to determine the structure of the data in the database files. A file was written to read all 306 files and write their data out into a CSV file. After pulling all of the data out into CSV files, the cleaning process began. The data was corrupted in many places and suffered from data quality issues. Once the cleaning process was complete the data was written out into the PDML format and added to the PDML database. There are 7967 records in the QBasic database and at this time 2672 of those were valid. After adding those records to the PDML the total count of records in the database came to 5270.

## 4.3 Data Preparation

The WMRD database contained a lot of data, but most of the data suffered from data quality errors. The effort to detect and clean the errors in the database followed the techniques detailed in Section 3.2.8. The overall process can be broken into roughly four phases.

### First Phase

The first phase of detection and cleaning focused on checking the validity of the field values and making sure they were correct. The QBasic database was designed so that the attributes stored in each column of data were different depending on the target type of that record. So, if the record was defined as a “SIN” then it was a finite single element target and the data stored in column 22 was the “Exit Hole Length”. If the record was a “DBL” then it was a finite double element target and the data stored in column 22 was “Plug Velocity”. A mistake when keying the record type could cause all kinds of errors for the database. So before any of the field values could be checked, the record type first needed to be verified. To add even more to the complexity of the problem, many of the fields used special numbers to signify a special condition. As an example, the residual mass field would store a numerical value that represented the mass of the projectile after the ballistic event; if it was equal to “111” then it meant that the data was lost during the test, if it was equal to “222” then it meant that the data represented a fragment of the projectile, and if it was equal to “0” then it meant that it was a partial penetration. In some cases the person entering the data may have thought that the code for fragment was “2222” so that was entered instead of “222”. Very rarely did every field for a particular attribute follow the coding without deviation. That example is actually one of the more simple examples of this occurrence.

Before the fields were checked all of the records were separated out into the four types of targets; “SIN”, “DBL”, “TRI”, and “SI”. After the data was separated, each record was checked to see if the target type was correct. One of the main ways that this was accomplished was by checking the hardness column for that record. For reference, Table 4.1 shows the columns in the database that are associated with hardness for at least one of the target types. If a record was a “SIN” target type then column 29 should have a value in it that could be a reasonable BHN value for that target material. If after checking that value it seems to be too low, it is possible that it is a typographical error or it could be categorized incorrectly. If upon further inspection, columns 36, 37, and 38 all contain values more appropriate for BHN values then it is possible that the record is a “TRI” and not a “SIN”. Sometimes cross checking several different attributes was required before determination could be made if the target type was correct.

Table 4.1: Database Hardness Columns

| **Type** | **Column 5** | **Column 10** | **Column 15** | **Column 29** | **Column 36** | **Column 37** | **Column 38** |
|---|---|---|---|---|---|---|---|
| SI | Hardness | Original Diameter | Volume of Rise | Ent Hole Length | Alpha | Beta | Path Deviation |
| SIN | Alpha | Eta R | Pen Depth | Hardness | # Pcs Residual | Max Res Dia | Eta P |
| DBL | Alpha | First Plate Hardness | Second Plate Hardness | Rotation Rate | # of Pcs After 1st Pl | # of Residual Pcs | Cone Angle |
| TRI | Alpha | Eta 1 | Alpha 2 | Third MR 1 | Hardness of Plate 1 | Hardness of Plate 2 | Hardness of Plate 3 |

Once the records were correctly classified for target type analysis was started to check for possible outliers. Various analysis methods were used to expose outlier data and subject that data to scrutiny. Some of those methods were; statistical analysis, clustering analysis, pattern-based searches, and association rules [94].

### Second Phase

The process of correcting errors during the first phase led to major change in the format of the data spreadsheet. There was a lot of duplicate information in the database and many times a correction in one place meant that it needed to be corrected in many places. This led to the creation of a relationship based spreadsheet where the projectile material properties and the target description information were both pulled out into their own separate tables. Each record in those tables was given a unique key and the main data table used that key to reference the appropriate data. This simplified the cleaning process for those two separate tables of information dramatically. The other major modification to the data format was to combine all of the separate data types together on one sheet. This was accomplished by combining attributes that represented the same thing and keeping those that were unique separate. Using the hardness example from before, instead of having seven different columns to define up to three plates of armor hardness, they were combined into three columns. If a record was a "SIN" target type, then the fields for the second and third plate hardness were set to "N/A".

After everything was combined the process of scrubbing the projectile materials and target descriptions began. All of the projectile materials and target descriptions had a description field and sometimes there would be information about the projectile or target in that field. If the field contained information it was checked against the other appropriate fields to make sure they matched. If those other fields were empty, then they were populated with the appropriate values.

### Third Phase

Even after scrubbing the projectile and target sheets, they both had a significant amount of fields that were empty. Through further discussions with the WMRD SMEs, it was discovered that all of the raw data that the database was built from over the years were contained in a large library of binders (well over several hundred). In order to try and fill in the missing data in the database and correct some of the errors that were present, several boxes of the binders were cataloged and scanned (see Figures 4.1, 4.2, and 4.3).

Figure 4.1: Image of first group of binders

Figure 4.2: Image of second group of binders

In total, seventy-seven binders were cataloged, of those only twenty-five were completely checked for data; the rest were only partially completed. The process involved cross checking any of the test numbers or test series identifiers with those in the database. Once the binder could be linked to a series of records in the database, the field values could be checked if

Figure 4.3: Image of third group of binders

they were present in the binder. Likewise, each binder was checked for material property information for the projectiles and the targets. If that information was found, it then had to be linked to a test series if possible. There were some cases were material information was found but there was no indication which tests used that particular projectile, so it could not be corrected in the database.

### Fourth Phase

The final phase of data preparation was focused on dealing with missing data. There is no single solution to the problem of missing data, but through a combination of intelligent replacement and imputation methods, suitable values can be placed into the missing data locations with minimal detrimental effect to the ability of the MLP to learn the patterns in the data [95].

One method of intelligent replacement is accomplished by making the common assumption that the diameter of the KEP does not change during penetration and by using basic geometric equations. Equation (4.1) can be used to solve for mass ($m$), density ($\rho$), diameter ($d$), and length ($l$) as long as only one of the parameters are missing.

$$\frac{m}{\rho} = \pi({}^{d}\!/_{2})^2 l \tag{4.1}$$

Where:

$\rho$ is the density of the projectile

$d$ is the diameter of the projectile

$l$ is the length of the projectile

$m$ is the mass of the projectile

Likewise, in some records length or diameter will be missing and the fineness of a projectile will be recorded. Fineness is defined in (4.2). If a record has fineness and diameter or fineness and length, then the other value can be calculated.

$$f = \frac{l}{d} \tag{4.2}$$

Where:

$f$ is the fineness of the projectile

$l$ is the length of the projectile

$d$ is the diameter of the projectile

Another method of intelligent replacement that was used to fill missing data was to compare similar records. For example, if one record is for a steel target with a BHN hardness of 300 and another record is from the same test series with a steel target, but hardness is missing. If both targets were of similar thickness, then the hardness of 300 was used to fill in the missing value.

Surrogation was sometimes used to fill missing data when intelligent replacement could not be used. For example if the projectile for a record was steel and the density was missing, then the standard density of 7.85 $^{g}/_{cm^3}$ was used for that missing value.

Imputation of a value using an average of similar records was sometimes used as means to fill missing data when the other methods could not be used. Continuing the example of the steel target that was missing a BHN hardness value, imagine if there were ten other records that were from the same test series and they were all steel targets. If each one of those records had BHN hardness values that ranged from 292 to 310, what value should be used? In those types of situations an average of the similar records was used to impute a value for the missing value.

## 4.4 Parameter Selection

Once the experimental test data had been collected the next step in the process of defining the MLP was the determination of inputs to use for the model. The number of inputs in a MLP is limited by the number of available input parameters in the problem, but it is possible that not all of the available input parameters should be utilized [41]. There is often a desire to include too many inputs in the design of an MLP due to two common misconceptions; (1) since MLPs learn, they will be able to determine what input variables are important, and (2) like with expert systems, as much domain knowledge as possible should be included into the system [43]. Determination of the input parameters to the MLP is extremely important for two primary reasons. The first reason is that the required number of data points increases with the number of input parameters. The second reason is that including two inputs that are highly correlated introduces noise in the training data which can lead to a loss of generalization and could cause a non-convergence of the MLP [55].

The inputs for the MLP were chosen based on discussions with KEP SMEs, limitations in the data available, and reviewing current phenomenological models in use. The eleven inputs that were selected for use in the MLP are:

- Striking Velocity
- Total Yaw
- Projectile Length
- Projectile Diameter
- Projectile Density
- Projectile Hardness
- Target Thickness
- Target Obliquity
- Target Hardness
- Target Density
- Target Young's Modulus

Three outputs were selected for the MLP: determination of perforation of the target, residual velocity, and residual mass. These were selected because they are the three things needed to determine if the projectile will continue into the target and for determination of damage on components in Vulnerability/Lethality (V/L) models.

## 4.5 Data Analysis

Analysis of the data is important because the MLP can learn patterns in the data that may not have been intended. As an example, early in this research there was a large percent of data for tests that resulted in perforations (approximately 80%) and because of this the MLP was predicting perforations at a very high rate. Once more data was added to the training set that were from non-perforation tests the MLP was able to better predict perforation outcomes without a bias toward perforation.

Analysis of the data is also important because we want the MLP to be trained on data that has a range of values for each attribute and a good sampling inside of that range. If those conditions are met, the MLP should be able to generalize over the broad set of possible scenarios expected for an analysis.

First, a look at the entire database; after all of the data collection and data cleaning, the PDML database contained 4854 test records. The distribution of those records is as follows; 1463 SNI, 2758 SNF, 0 MAI, 571 MAF, 0 MNI, and 62 MNF.

Out of those 4854 records, 3034 were used with the MLP. One reason for the difference is that some of the records in the PDML are marked as “suspect”, meaning there is concern with the data for those records and they should not be used at this time. The other reason for the difference is that SNI data was not needed for this analysis and therefore was not used. The distribution of the records used with the MLP is; 0 SNI, 2455 SNF, 0 MAI, 556 MAF, 0 MNI, and 23 MNF.

### Parameter Distributions

The distribution of the input and output parameters from the SNF data are provided in Table 4.2.

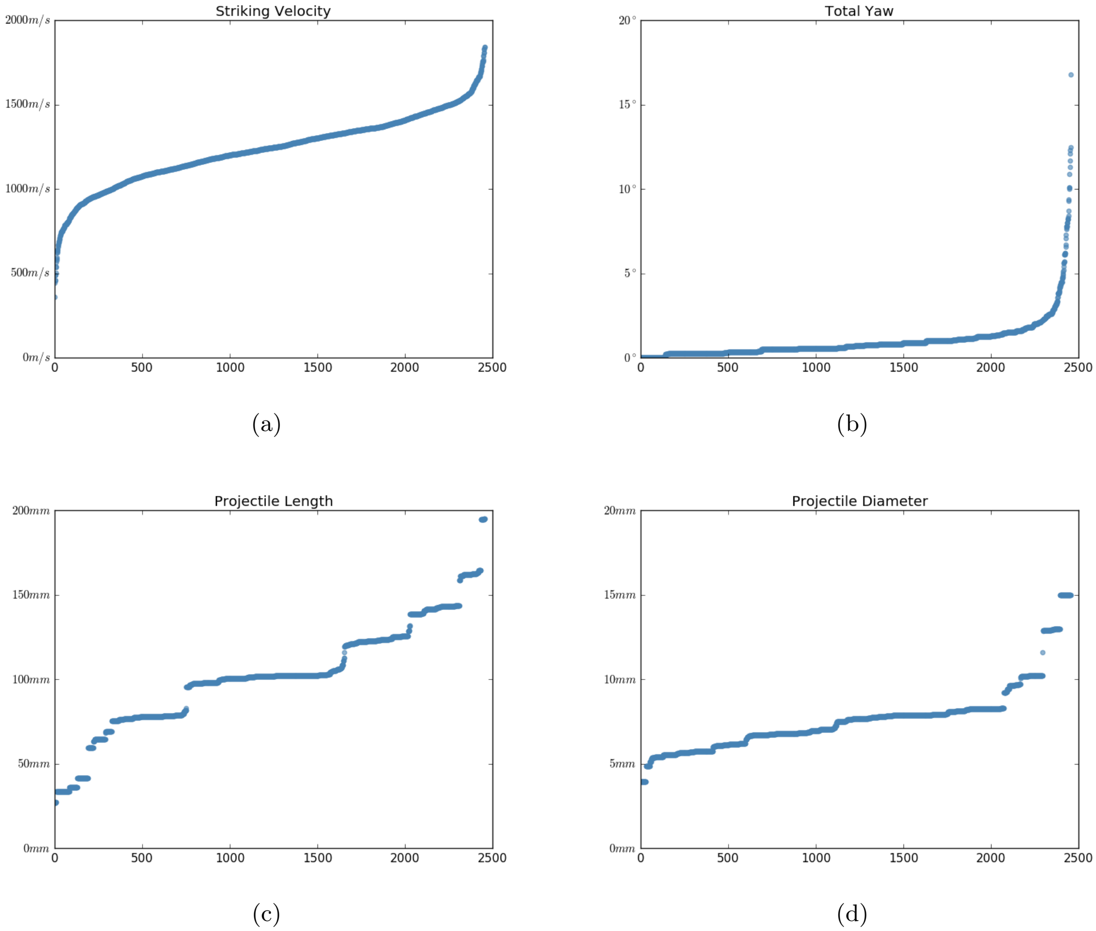
Table 4.2: Statistics for the input and output values of the ANN

| Parameter | Symbol | Mean | Std Dev | Minimum | Q1 | Median | Q3 | Maximum |
|---|---|---|---|---|---|---|---|---|
| Input Values | | | | | | | | |
| Striking Velocity (m/s) | $V_s$ | 1232.8 | 205.7 | 361 | 1106 | 1243 | 1365 | 1841 |
| Total Yaw (°) | $\gamma$ | 1 | 1.2 | 0 | 0.4 | 0.7 | 1.1 | 16.8 |
| Projectile Density (g/cm³) | $\rho_p$ | 16.6 | 3.4 | 7.7 | 17.3 | 17.7 | 18.6 | 19.3 |
| Projectile Length (mm) | $l$ | 102 | 32 | 27 | 77.9 | 101.9 | 123 | 195 |
| Projectile Diameter (mm) | $d$ | 7.7 | 2 | 3.9 | 6.6 | 7.7 | 8.1 | 15 |
| Projectile Hardness (BHN) | $BHN_p$ | 428.1 | 97.8 | 233.3 | 379.5 | 398 | 443.3 | 869.1 |
| Target Density (g/cm³) | $\rho_t$ | 7.6 | 1.2 | 2.7 | 7.9 | 7.9 | 7.9 | 7.9 |
| Target Hardness (BHN) | $BHN_t$ | 323.5 | 91.8 | 107 | 269 | 302 | 364 | 555 |
| Target Thickness (mm) | $T$ | 46.6 | 24.7 | 6.3 | 25.4 | 38.1 | 63.5 | 127 |
| Target Young's Modulus (GPa) | $E$ | 199.9 | 31.4 | 70 | 207 | 207 | 207 | 210 |
| Target Obliquity (°) | $\theta$ | 35.3 | 30.6 | 0 | 0 | 60 | 60 | 80 |
| Output Values | | | | | | | | |
| Perforation (Binary) | $P$ | 0.2 | 0.9 | -0.9 | -0.9 | 0.9 | 0.9 | 0.9 |
| Residual Velocity (m/s) | $V_r$ | 359.1 | 403 | 0 | 0 | 247 | 625 | 1609 |
| Residual Mass (g) | $M_r$ | 9.9 | 15.2 | 0 | 0 | 5.4 | 12.6 | 114.7 |

### Ordered Scatter Plots

Ordered scatter plots are a good way to show the distribution of a particular parameter. Figures 4.4, 4.5, 4.6, and 4.7 are ordered scatter plots of the various parameters used by the MLP. Each plot is created by taking every value in the database for a particular parameter and then sorting them in order. After they have been sorted, they are plotted in order by value. This type of plot can be used to visually show the distribution of a particular parameter. Striking velocity is shown in Figure 4.4a and the plot shows a good spread of values for that parameter. Total yaw is shown in Figure 4.4b, it can be seen in the plot that most of the values are below 5 °. Although more values above 5 ° would be preferable, most realistic ballistic impacts are going to occur below 5 °, so the distribution is good. The values for projectile length are shown in Figure 4.4c and although the distribution has some very obvious discontinuities, it does cover a good range of values. The distribution for projectile diameter shows that there may be a need to acquire more data with values above 10 mm (see Figure 4.4d).

Figure 4.5a shows projectile density and a very clear gap in data for materials other than steel (≈ 7.85 g/cm³), tungsten (≈ 17.0 g/cm³), and depleted uranium (≈ 19.0 g/cm³). The values for projectile hardness (see Figure 4.5b) cover a good range, but there is a gap between ≈ 600 BHN and ≈ 800 BHN. The distribution for target thickness (see Figure 4.5c) is not continuous, but that is to be expected. The target plates are typically purchased in



(a)

(b)

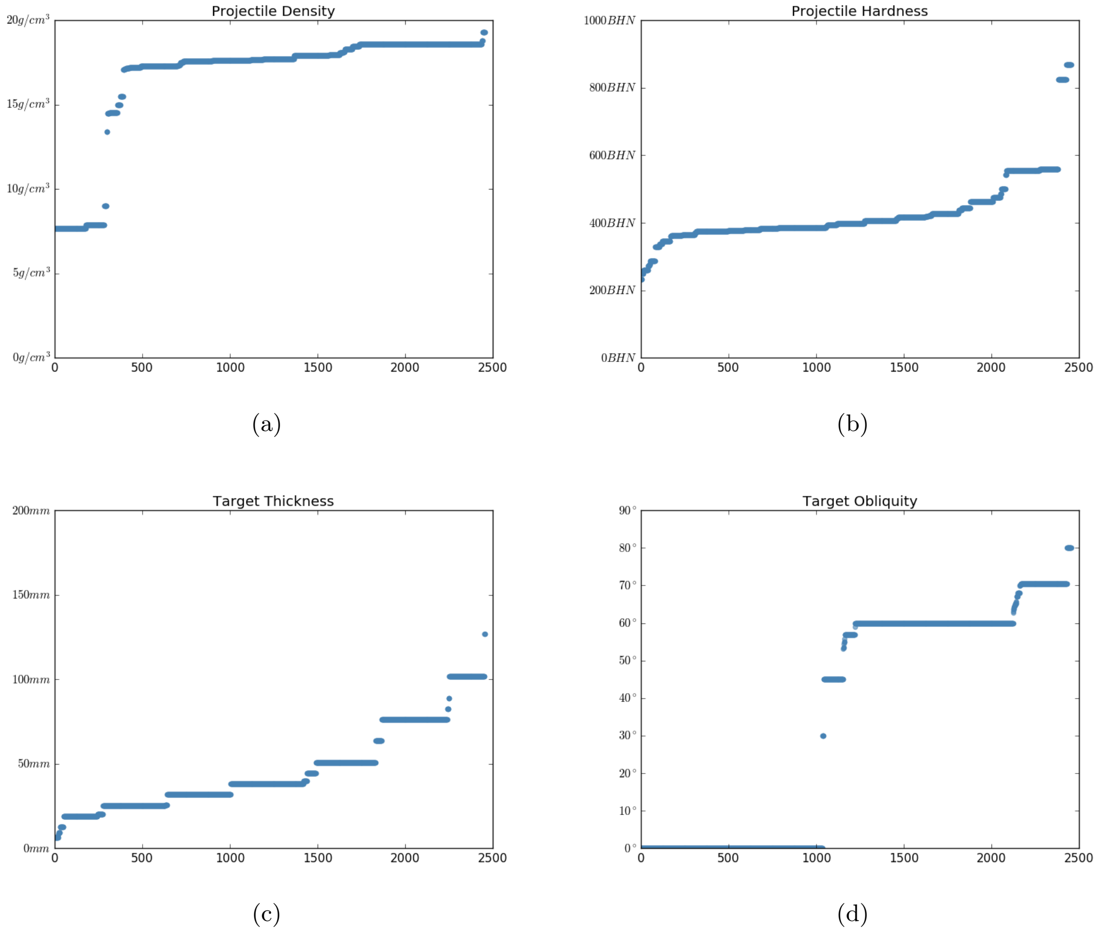
(c)

(d)

Figure 4.4: Ordered scatter plots for some input parameters of model

incremental thicknesses of 1/4 in (6.35 mm), so the values are expected to appear as a step function. The distribution for obliquity is shown in Figure 4.5d and almost all of the values are shown to be equal to 0 ° or greater than or equal to 45 °. Future work should try to get more data with obliquity values that fall between 0 ° and 45 °.

Target hardness (see Figure 4.6a) has a good spread of values from ≈ 100 BHN to ≈ 550 BHN. Even though there is a good sampling of hardness values, Figure 4.6a shows that most of the data is for steel (≈ 7.85 $\mathrm{g/cm^3}$) or aluminum (≈ 2.7 $\mathrm{g/cm^3}$) targets. This is where the data is lacking the most; future work must try to find data for a more diverse set of materials. If the data can not be found then attempts should be made to find funding to perform experimental tests and gather the data. Figure 4.6c further shows the lack of diversity for target materials; the target Young's modulus values are for only two types of



(a)

(b)

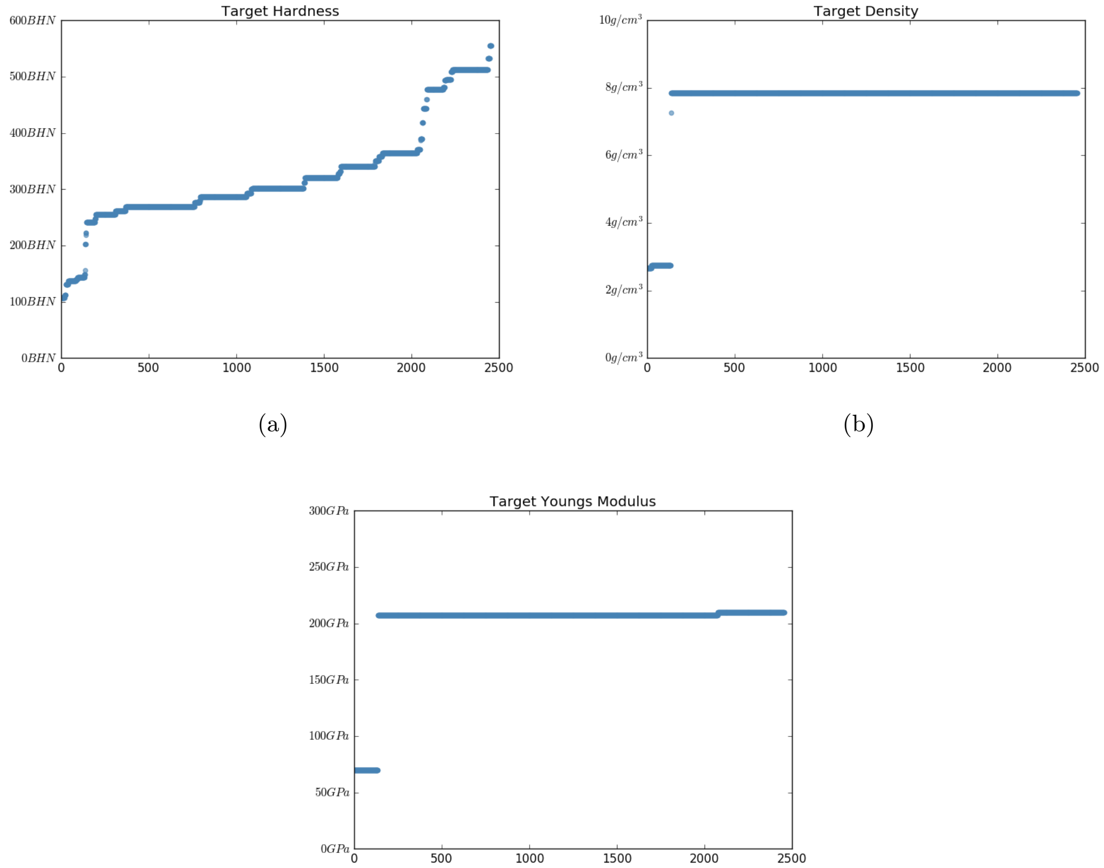
(c)

(d)

Figure 4.5: Ordered scatter plots for some input parameters of model

materials (steel and aluminum). Even though there is a strong desire to have data that covers more than steel and aluminum target materials, it should be noted that for a realistic encounter the two most likely materials to be used for armor are steel and aluminum.

The final set of ordered scatter plots (Figure 4.7) show the output parameters. In Figure 4.7a (residual velocity) and Figure 4.7b (residual mass) the tests that resulted in non-perforations can be seen as values of 0 m/s or 0 g, respectively. The values for residual velocity cover a good range and are distributed well. The residual mass values show a definite tendency to values below 20 g, the reason for this is not entirely clear but could be due to the fact that many of these tests are performed when near the ballistic limit of a threat. It is possible that most of these tests were near the ballistic limit due to reduction in usable rod length (and therefore mass) as opposed to reduction in usable velocity.



(a)

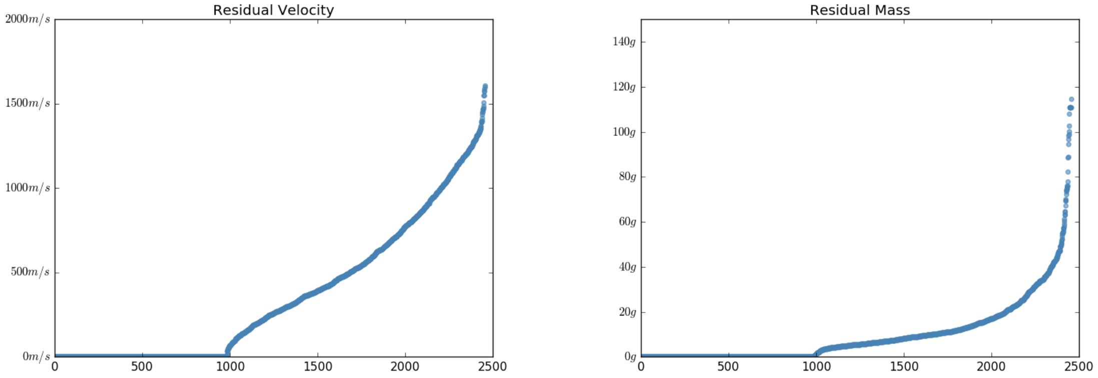
(b)

(c)

Figure 4.6: Ordered scatter plots for some input parameters of model

### Box and Whisker Plots

In order to show all of the parameters in one set of box and whisker plots, their values were normalized based on the values in Table 4.3. The box and whisker plots are shown in Figure 4.8. In the plots the tails mark the maximum and minimum values, the hollow boxes mark the $1^{st}$ quartile to $3^{rd}$ quartile, the line in the middle marks the median value, and the solid square box marks the average value. The plots clearly show the lack of spread of values for target density and target youngs modulus. Not as clear is the gap in values for target obliquity that was shown with the ordered scatter plots.



(a)

(b)

Figure 4.7: Ordered scatter plots for some output parameters of model

## 4.6 Summary

The data collection and preparation process was the most difficult and time consuming part of this research effort. The process of collecting data for this type of research is very likely to continue long after this effort is complete. This is due in part to the gaps in the data and the continuing push for use of new materials and new technology in KEPs.

There is a less than desirable spread of values in the data for the target and KEP materials. There is also a lack of data with obliquities that fall between 0 ° and 45 °. Future work should, and most likely will, attempt to fill in those gaps. However, the data are complete enough to accomplish the desired objectives of this research.

This work was able to determine the best input parameters to use in the model through trial and error, discussion with KEP SMEs, and by researching other models. The output parameters were selected based on the requirements for the problem space. The determination of the input and output parameters provided an answer to RQ1.1. The selected parameters are: Striking Velocity, Total Yaw, Projectile Length, Projectile Diameter, Projectile Density, Projectile Hardness, Target Thickness, Target Obliquity, Target Hardness, Target Density, Target Young's Modulus, Perforation, Residual Velocity, and Residual Mass.

Table 4.3: The normalization values used for the parameter box and whisker plots

| **Parameter** | **Minimum Value** | **Maximum Value** | **Minimum Normalized** | **Maximum Normalized** |
|---|---|---|---|---|
| | | Input Values | | |
| $V_s$ | 0.0 $^{m}/_{s}$ | 2000.0 $^{m}/_{s}$ | -1.0 | 1.0 |
| $\gamma$ | 0.0 ° | 25.0 ° | -1.0 | 1.0 |
| $\rho_p$ | 0.0 $^{g}/_{m^3}$ | 20.0 $^{g}/_{m^3}$ | -1.0 | 1.0 |
| $l$ | 0.0 mm | 250.0 mm | -1.0 | 1.0 |
| $d$ | 0.0 mm | 20.0 mm | -1.0 | 1.0 |
| $BHN_p$ | 0.0 BHN | 900.0 BHN | -1.0 | 1.0 |
| $\rho_t$ | 0.0 $^{g}/_{m^3}$ | 20.0 $^{g}/_{m^3}$ | -1.0 | 1.0 |
| $BHN_t$ | 0.0 BHN | 700.0 BHN | -1.0 | 1.0 |
| $T$ | 0.0 mm | 250.0 mm | -1.0 | 1.0 |
| $E$ | 0.0 GPa | 500.0 GPa | -1.0 | 1.0 |
| $\theta$ | 0.0 ° | 85.0 ° | -1.0 | 1.0 |
| | | Output Values | | |
| $V_r$ | 0.0 $^{m}/_{s}$ | 2000.0 $^{m}/_{s}$ | -1.0 | 1.0 |
| $M_r$ | 0.0 g | 150.0 g | -1.0 | 1.0 |

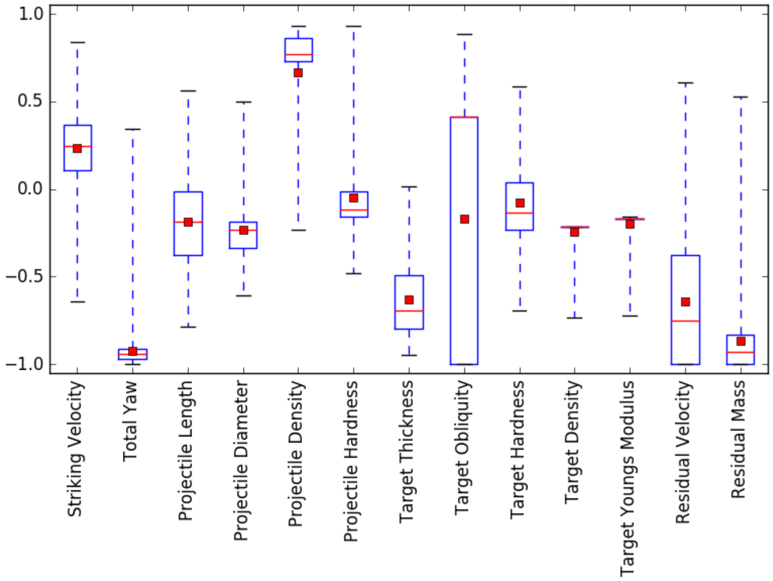


Figure 4.8: Box and Whisker Plots of Normalized Model Parameters

# Chapter 5

# MLP for Monolithic Metallic Armor

The first Multi-Layered Perceptron (MLP) developed was for the simplified problem of prediction of perforation against monolithic metallic armor. The next iteration of development then added in the regression values of residual velocity and residual mass. The following sections will provide the methodology of development, the training process results, the comparison against the Segletes model, and a summary. The work detailed in this chapter was used to answer RQ1.2, RQ1.4, and RQ1.5.

## 5.1 Methodology

The process for developing the MLP consisted of three phases; topology selection phase, training phase, and comparison phase. The following three sections will provide an overview of how each of those phases were accomplished.

### Topology Selection Phase

The topology selection phase was done as a focused search. That means that only the topologies of interest were run and constructive and pruning methods were not used. The topologies of interest were selected based on the process documented in Section 3.2.7. The selection process started by normalizing all of the parameters being used by the MLP. After normalization, networks were created for each topology based on the user selected number of subsamples. For example, if there were two topologies of interest and the user selected to have 15 subsamples, then there would be a total of 30 networks created. All of those created networks were then initialized with random weights. Additional networks were created, because 10-fold cross validation was being used. For each network created because of the subsampling number, 9 additional networks were created to make a total of 10. Each of those 10 networks were fed different folds of the training data and the errors of those 10 networks were averaged to determine a generalization error for that network subsample. Then the best generalization error of all of the subsamples was selected to represent the result for that given topology. During the selection process for the final MLP design, 3 subsamples were used. Since each topology was sub-sampled 3 times the best generalization error of those 3 networks was used for evaluation against the other topologies. The back-propagation method used for training the networks was Scaled Conjugate Gradient (SCG). For the early stopping strategies, a minimum improvement criteria of 0.000001 was used with a tolerated epochs criteria of 400. The maximum number of epochs allowed was 100 000.

During the selection phase, the following steps were taken:

1. The Penetration Database Markup Language (PDML) database was loaded and the records were prepared for use with the MLP.

2. 15% of the data was pulled aside for use as a test set of data when training was completed.

3. The remaining 85% was split into 10 folds of data for use with the k-fold cross validation methodology.

4. 3 instances of each topology were created so that each could be trained on the folded training data.

5. For each instance, 10 networks were created for the 10-fold cross-validation.

6. The error for a given MLP instance was calculated by averaging the error over the 10 folds, this was done to get a better estimate of the generalization error of a particular topology.
7. The best generalization error of the 3 instances for a topology was recorded.
8. Once all of the topologies had finished training the generalization errors were compared to determine the topology that would provide the best generalization performance with the lowest risk of over-fitting.

**Training Phase**

After a topology was chosen during the selection phase, the selected topology was used during the training phase. As with the selection process, sub-sampling was used to increase the chance of finding the global optimum solution. In addition to random sampling, a genetic algorithm was also used. The genetic algorithm was run prior to training to allow for starting weights that were close to the global minimum of the error function. To increase the likelihood of finding the global minimum, 20 instances of the MLP were created and their weights were assigned based on the results from running their own genetic algorithm. After the genetic algorithm had completed, a back propagation method was used to train the MLP and modify the weights to the optimal solution. Once each instance was finished training, the instance with the best validation error was selected as the final MLP to use in the comparison phase.

During the training phase, the following steps were taken:

1. The PDML database was loaded and the records were prepared for use with the MLP.
2. The same 15% of the data that was pulled aside during the selection phase, was again pulled aside for use as a test set of data when training was completed.
3. Out of the remaining data, 10% was pulled aside for the validation set and 90% was used for training the MLP.
4. 20 instances of the MLP were created so that each could be trained on the training data.

5. Prior to beginning the backpropagation training of each of the MLPs, they were all trained using a genetic algorithm to give good starting weights.

6. During backpropagation training, the validation error was checked after each epoch and if the error worsened by a certain percentage then training stopped and the weights associated with the best error were used.

7. Once all of the MLPs had finished training the one with the best validation error was selected as the best and was reported back with the correct weights for that MLP.

### Comparison Phase

The comparison phase for monolithic targets consisted of simply running the MLP and the Segletes model against the training, validation, and test sets of data. The results of both models were written out to a file for further statistical analysis.

During the comparison phase, the following steps were taken:

1. The PDML database was loaded and the records were prepared for use with the MLP and the Segletes model.

2. The same 15% of the data that was pulled aside during the selection and training phases, was again pulled aside for use as a test set of data.

3. Out of the remaining data, 10% was pulled aside for the validation set and 90% was used for the training set.

4. Each of the experimental test data in the three data sets were run with the MLP and the Segletes model.

5. The results of both models were output to a file for analysis.

### MLP Design Progression

The initial plan for implementing the MLP was to train two MLPs, one for prediction of perforation and one for prediction of the residual values after perforation. The initial decision to do that was based on a paper by Fernndez-Fdz, Puente, and Zaera [62]. They used an application of MLPs that broke the prediction of perforation and residual values into a two step process. Instead of using one MLP for determining perforation and residual values, the task was broken up into a MLP for classification (perforation and non-perforation) and a second MLP for regression of the residual values, if perforation was predicted by the previous MLP. The benefit of separating the two tasks is the reduction in complexity of the overall networks and therefore an increase in the likelihood of faster convergence.

The classification MLP was developed and the results were published in [96]. While developing the classification MLP, both the Akaike Information Criterion with Correction (AICc) and k-fold cross validation were being used during the topology selection phase. Those methods led to the selection of a MLP topology that was very simple in structure. The topology that was selected for that MLP contained only four neurons in the first hidden layer and one neuron in the second hidden layer (see Figure 5.1).

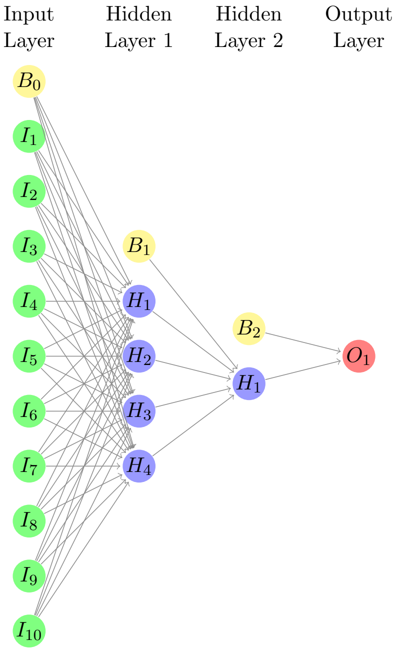


Figure 5.1: Selected Topology for First Version of MLP

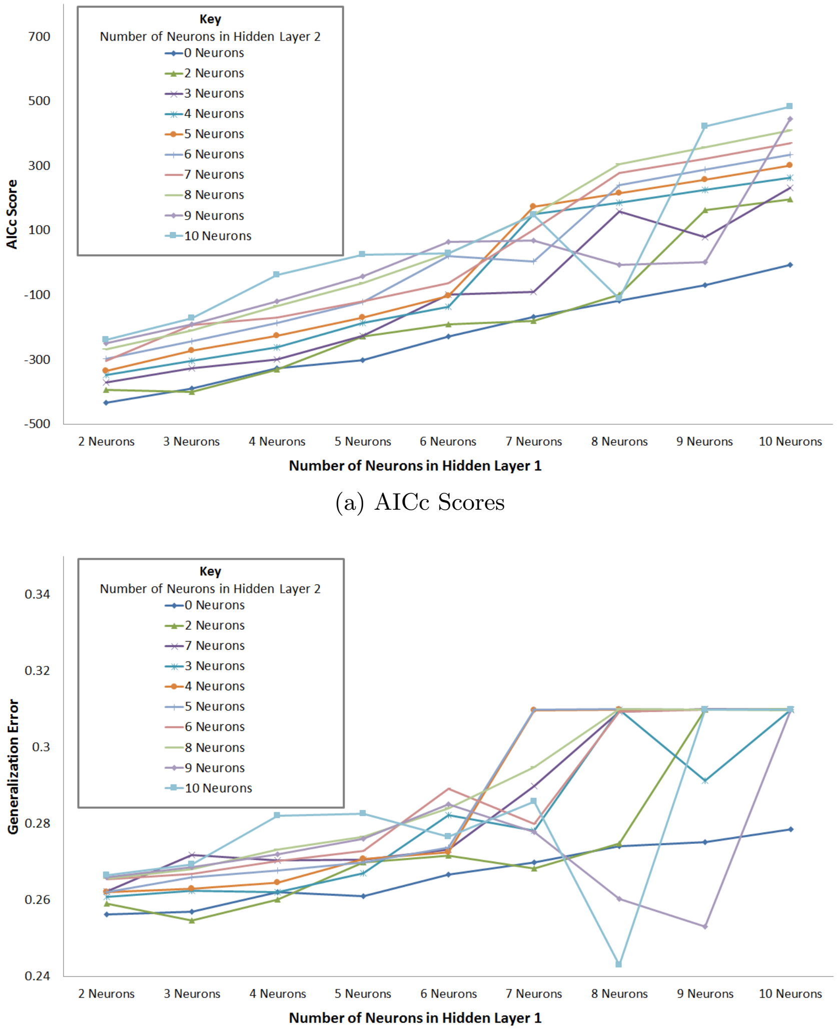
Early feedback from MLP Subject Matter Experts (SMEs), at the 2014 IEEE Symposium Series on Computational Intelligence, led to the removal of the AICc from the topology selection process. The AICc attempts to account for complexity in the scoring by "taking points off" for complexity. K-fold cross validation inherently accounts for overly complex networks in the generalization error. In effect, the networks that were more complex were being penalized twice. The removal of the AICc led to more complex networks, but the continued use of k-fold cross validation and early stopping strategies ensured that over-fitting was not a concern.

After training the regression MLP a problem of disconnect between the two networks was discovered. There were some cases where the classification MLP would predict a projectile would not perforate a target, but the regression MLP would predict residual values greater than zero. There were other cases where the classification MLP would predict perforation, but the regression MLP would predict residual values of zero. A decision was made to combine both of the problems into one MLP so that the predictions could be tied together more closely.

After the feedback from the conference and the dissconnect of the two MLP designs, the process of redesigning the MLP was started. During the topology selection phase a total of ninety topologies were run. The first eighty one topologies were setup with two layers having neuron counts ranging from two to ten in both layers. The last nine topologies were a single layer with neuron counts of two to ten. All of the topologies had three output neurons; perforation, residual velocity, and residual mass. The selection process for this MLP did not include the AICc; it was based only on the generalization error from k-fold cross validation. However, for comparison the AICc scores and generalization errors will both be provided. The AICc scores are shown in Figure 5.2a and the generalization errors are shown in Figure 5.2b. Had the topology been selected based on the AICc scores and not the generalization errors, then the topology that was selected would have consisted of a single hidden layer with only two neurons.

Based on the generalization errors, the best performing topology consisted of eight neurons in the first layer and ten neurons in the second layer (this topology can be seen in Figure 5.3).

After the QBasic database (from the Weapons Material Research Directorate (WMRD)) was added to the PDML database there were 2068 new data points available for use in the MLP. It was decided to perform another round of topology selection and training in order



(a) AICc Scores

(b) Generalization Errors

Figure 5.2: Selection Phase Results

to have the best possible solution for monolithic targets. The performance of the MLP against monolithic targets was an extremely important factor in how well it would perform against multi-element targets.

A total of 16 topologies were used during the selection process. Each of those networks were sampled 3 times and were 10-fold cross-validated. The topology with the best 10-fold cross-validated generalization error consisted of 11 input neurons, 16 hidden neurons in the $1^{st}$ hidden layer, 16 hidden neurons in the $2^{nd}$ hidden layer, and 3 output neurons in the output layer. Each layer, except the output layer, also had a bias neuron. The selected topology can be seen in Figure 5.4.

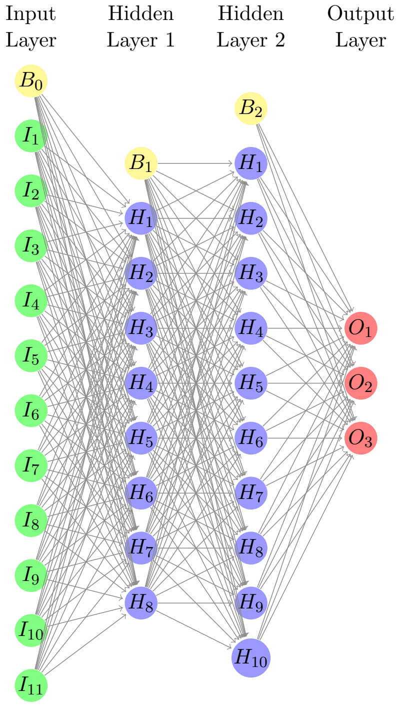


Figure 5.3: Selected Topology for Second Version of MLP

## 5.2 Training Process Results

The MLP was able to effectively learn the training data. Based on the Symmetric Absolute Percentage Error (SAPE), there were 4 tests in the training set that the MLP had a %Error worse than 80% that were not false positives or false negatives. There were 1877 total tests in the training set so they account for only 0.4% of the data in the training set. For residual mass there were 21 tests in the training set with %Error worse than 80% that were not false positives or false negatives. There were 1877 total tests in the training set so they account for only 1.1% of the data in the training set. For the training set the MLP had 596 true negatives, 153 false positives, 120 false negatives, and 1008 true positives.

No immediate patterns could be found in the tests that would lead to a conclusion that the MLP was having difficulty learning any particular type of projectiles or targets. The most likely cause for the false positives, false negatives, and the few tests that had higher than 80% SAPE is that those particular tests fall very close to the ballistic limit for their particular projectile and target pairing.

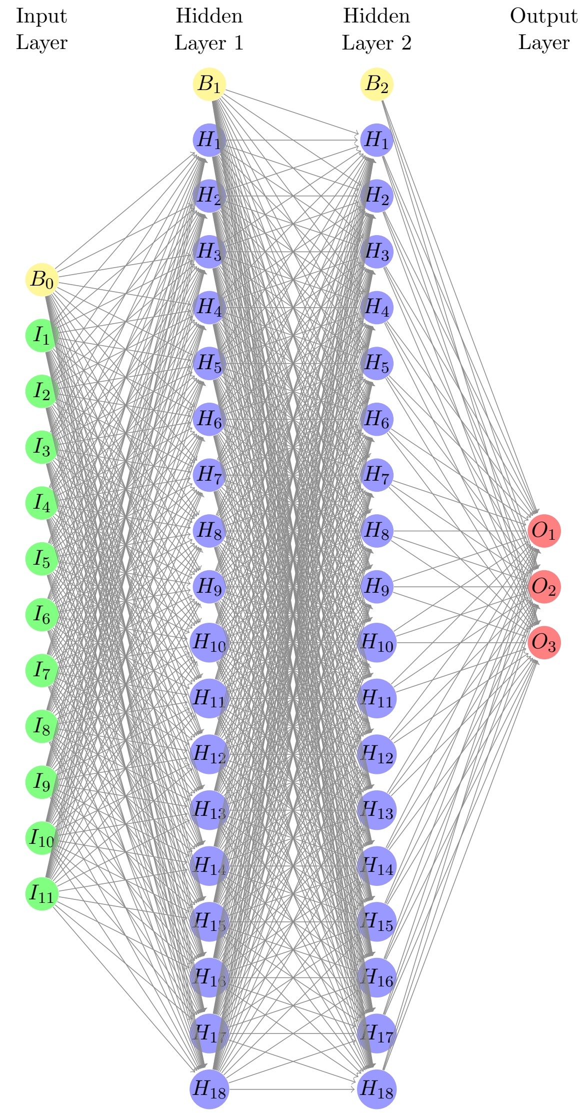


Figure 5.4: Final Selected Topology of the MLP

## 5.3 Comparison to Segletes Model

In this section the results will be presented for all three sets of data (training, validation, and test) for completeness; however, the set that is the most important to the results of the model is the test set.

### 5.3.1 Perforation Prediction Results

The MLP performed better than the Segletes model for SNF targets in all three sets of data. Table 5.1 provides some statistics on the perforation prediction capability of the MLP and the Segletes model; for clarity, the better values are shown in bold. The MLP performed better for for all three of the data sets for MSE, ACC, FPR, TNR, and MCC. The MLP performs better for FNR and TPR for the training set, but not for the validation and test sets. The MLP had four more false negatives for the validation set and three more false negatives for the test set, than the Segletes model. Those slightly higher counts of false negatives led to a higher (worse) score for FNR and lower (worse) score for TPR. For all three data sets, the MSE for the MLP was nearly half of the value that the Segletes model had.

Table 5.1: Perforation Prediction Statistics for Monolithic Targets

| Training Set | | |
|---|---|---|
| | MLP | Segletes |
| Mean Square Error (MSE) | **0.4712** | 1.0737 |
| Accuracy (ACC) | **85.5%** | 66.9% |
| False Positive Rate (FPR) | **20.4%** | 58.9% |
| False Negative Rate (FNR) | **10.6%** | 16.0% |
| True Positive Rate (TPR) | **89.4%** | 84.0% |
| True Negative Rate (TNR) | **79.6%** | 41.1% |
| Matthews Correlation Coefficient (MCC) | **0.69** | 0.28 |

| Validation Set | | |
|---|---|---|
| | MLP | Segletes |
| Mean Square Error (MSE) | **0.6356** | 1.0077 |
| Accuracy (ACC) | **80.4%** | 68.9% |
| False Positive Rate (FPR) | **17.9%** | 53.8% |
| False Negative Rate (FNR) | 20.6% | **17.6%** |
| True Positive Rate (TPR) | 79.4% | **82.4%** |
| True Negative Rate (TNR) | **82.1%** | 46.2% |
| Matthews Correlation Coefficient (MCC) | **0.6** | 0.31 |

| Test Set | | |
|---|---|---|
| | MLP | Segletes |
| Mean Square Error (MSE) | **0.8078** | 1.2205 |
| Accuracy (ACC) | **75.1%** | 62.3% |
| False Positive Rate (FPR) | **32.5%** | 64.9% |
| False Negative Rate (FNR) | 19.5% | **18.1%** |
| True Positive Rate (TPR) | 80.5% | **81.9%** |
| True Negative Rate (TNR) | **67.5%** | 35.1% |
| Matthews Correlation Coefficient (MCC) | **0.48** | 0.19 |

#### Perforation Prediction Pie Charts

The pie charts in Figures 5.5, 5.6, and 5.7 show the distribution of classification outcomes for the MLP and the Segletes model for the training, validation, and test sets for SNF targets respectively. For the training set the MLP had 596 true negatives, 153 false positives, 120 false negatives, and 1008 true positives (the percentages are shown in Figure 5.5a). For the same set of data, the Segletes model had 308 true negatives, 441 false positives, 181 false negatives, and 947 true positives (the percentages are shown in Figure 5.5b). For the validation set the MLP had 64 true negatives, 14 false positives, 27 false negatives, and 104 true positives (the percentages are shown in Figure 5.6a). For the same set of data, the Segletes model had 36 true negatives, 42 false positives, 23 false negatives, and 108 true positives (the percentages are shown in Figure 5.6b). For the test set the MLP had 104 true negatives, 50 false positives, 42 false negatives, and 173 true positives (the percentages are shown in Figure 5.7a). For the same set of data, the Segletes model had 54 true negatives, 100 false positives, 39 false negatives, and 176 true positives (the percentages are shown in Figure 5.7b).

### 5.3.2 Residual Value Estimation Results

The MLP significantly outperformed the Segletes model on prediction of residual values. Table 5.2 lists the MSE for the MLP and the Segletes model for all three data sets. For the test set, the MSE scores for residual velocity prediction by the MLP were roughly 70% better than the Segletes model scores and roughly 80% better for residual mass predictions. The Symmetric Mean Absolute Percentage Error (SMAPE) scores for the MLP were much lower (better) than the Segletes model. The MLP had a SMAPE score of 33.4% compared to the Segletes models score of 47.9% for residual velocity. For residual mass, the MLP had a SMAPE score of 35.6% compared to the Segletes models score of 55.1%.

#### Residual Value Scatter Plots

In the scatter plots (Figures 5.8 and 5.9), observed values are along the $x$-axis and predicted values are along the $y$-axis. If a model performed perfectly then the values would fall along a line with slope equal to 1 and a $y$-intercept equal to 0. Cases of incorrect perforation

Table 5.2: Mean square error and symmetric mean absolute percentage error statistics for residual value predictions of monolithic targets

| Training Data | | |
|---|---|---|
| | MLP | Segletes |
| MSE for Residual Velocity | **0.0491** | 0.2271 |
| MSE for Residual Mass | **0.0244** | 0.1174 |
| SMAPE for Residual Velocity | **24.6%** | 44.6% |
| SMAPE for Residual Mass | **26.6%** | 51.7% |

| Validation Data | | |
|---|---|---|
| | MLP | Segletes |
| MSE for Residual Velocity | **0.0545** | 0.1993 |
| MSE for Residual Mass | **0.02** | 0.0986 |
| SMAPE for Residual Velocity | **28.6%** | 42% |
| SMAPE for Residual Mass | **30.4%** | 49.2% |

| Test Data | | |
|---|---|---|
| | MLP | Segletes |
| MSE for Residual Velocity | **0.0726** | 0.2339 |
| MSE for Residual Mass | **0.0232** | 0.113 |
| SMAPE for Residual Velocity | **33.4%** | 47.9% |
| SMAPE for Residual Mass | **35.6%** | 55.1% |

prediction (false positives) can be seen plotted on the $y$-axis with a value of 0 for $x$. Cases of incorrect non-perforation prediction (false negatives) can be seen plotted on the $x$-axis with a value of 0 for $y$. Both of the scatter plots show the high number of false positives for the Segletes model. In particular, the high predicted residual velocity values for those false positives on the $y$-axis (Figures 5.8) Most of the false positives and false negatives for the MLP were very values, meaning that when the MLP was wrong about prediction, it was not very off on the residual value predictions.

### Residual Value Percent Error

The Symmetric Percentage Error (SPE) histograms can be seen in Figures 5.10 and 5.11. The false positives show up in the -100% bin on the far left of the plot and the false negatives show up in the 100% bin on the far right of the plot. Due to how the SPE is calculated, if the percent error is negative then that is indicative of an over estimation of the value observed and if the percent error is positive then that is indicative of an under estimation of the value observed. Overall, the MLP had lower percent errors for residual velocity (Figure 5.10) than the Segletes model and this can be seen in the combined higher percentage of tests that had percent errors from -10% to 0% (Bin 0%) and 0% to 10% (Bin 10%). This trend continued in the percent errors for residual mass (Figure 5.11) as well.

Based solely on Figures 5.12 and 5.13 and their associated values, it can be determined that the MLP is first order stochastic dominate over the Segletes model for both residual velocity and residual mass.

**Data Point-by-Data Point Prediction Comparison**

When comparing data from all three sets of data for monolithic targets (SNF) the MLP far surpassed the predictive capabilities of the Segletes model for both the classification problem of determination of perforation and the regression problem of predicting the residual velocity and mass. For prediction of residual velocity the MLP performed better than the Segletes model on 1146 data points and performed the same as the Segletes model on 604 data points. That means that the MLP performed as well or better than the Segletes model for 1750 out of 2455 data points (71%). For prediction of residual mass the MLP performed better than the Segletes model on 1379 data points and performed the same as the Segletes model on 604 data points. That means that the MLP performed as well or better than the Segletes model for 1990 out of 2455 data points (81%).

The most important set of data is the test set, so the following results are for that set of data only. The MLP still far surpassed the predictive capabilities of the Segletes model for both the classification problem of determination of perforation and the regression problem of predicting the residual velocity and mass when only the test set of data is compared. For prediction of residual velocity the MLP performed better than the Segletes model on 156 data points and performed the same as the Segletes model on 96 data points. That means that the MLP performed as well or better than the Segletes model for 252 out of 369 data points (68%). For prediction of residual mass the MLP performed better than the Segletes model on 184 data points and performed the same as the Segletes model on 96 data points. That means that the MLP performed as well or better than the Segletes model for 280 out of 369 data points (76%).

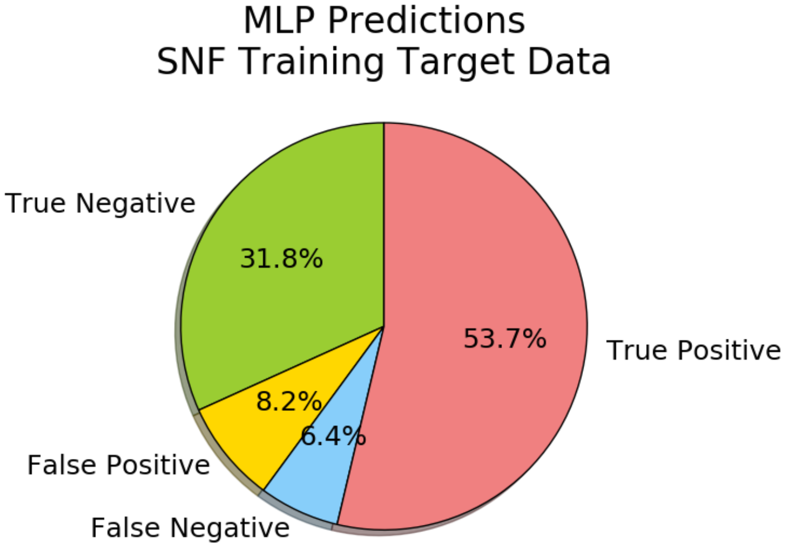
For those data points where the Segletes model performed better, there were 12 data points that resulted in false positives for the MLP and 26 that resulted in false negatives. There were 117 data points (32%) that the Segletes model did better than the MLP for predicting residual velocity. The 12 false positive predictions by the MLP accounted for 10% and the 26 false negative predictions by the MLP accounted for 22% of the tests. The remaining 79 data points were ones that the MLP correctly predicted perforation, of those data points the lowest error for the MLP was 2%, the highest error was -60%, and the average error was 21%. There were 89 data points (24%) that the Segletes model did better than the MLP for predicting residual mass. The 12 false positive predictions by the MLP accounted for 13% and the 26 false negative predictions by the MLP accounted for 29% of the data points. The remaining 51 data points were ones that the MLP correctly predicted perforation, of those data points the lowest error for the MLP was 1%, the highest error was 69%, and the average error was 27%.

### 5.3.3 Runtime Comparison

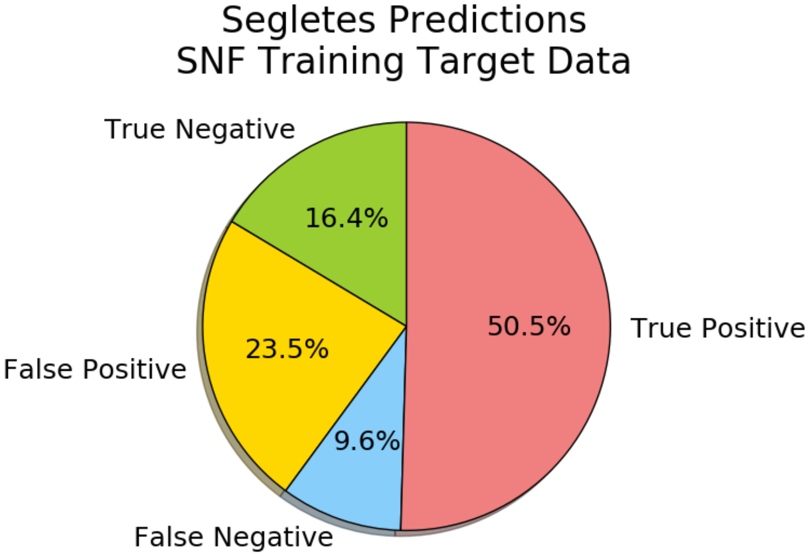
No attempt was made to optimize either of the models for runtime, so the runtimes are only comparable for the currently unoptimized state of both models. The Segletes model is a phenomenological model that is numerically integrated, so the runtimes are expected to be longer than that seen for the MLP. In Figure 5.14 the bins are runtime in milliseconds and the frequency is the number of runs that had runtimes that fell into that bin. As an example, in Figure 5.14 there were 2159 out of 2159 Artificial Neural Network (ANN) runs that took less than 0.25 ms for a value of 100% and there were 75 out of 2442 Segletes model runs that took between 0.25 ms and 0.5 ms for a value of 3%. The average runtime for the MLP model against SNF targets was 0.06 ms. The average runtime for the Segletes model against SNF targets was 1.61 ms. The overall difference in runtimes between the two models amount to only a few milliseconds, but in a Vulnerability/Lethality (V/L) simulation it can be expected that these models will be run millions of times. Therefore, a few milliseconds difference in runtime could become an overall difference in simulation runtime of hours.

## 5.4 Summary

This chapter provided results that can be used to answer RQ1.2, RQ1.4, and RQ1.5. The MLP was able to model the terminal ballistics of Kinetic Energy Projectiles (KEPs) and there were no systematic difficulties with predicting any of the experimental data presented to it. The results clearly demonstrate that a MLP (a type of ANN) can be used model the terminal ballistics of KEPs and do so in a manner that is generalized and accurate (RQ1.2). The MLP was more accurate than the Segletes model in predicting perforation for SNF targets. It also had better Mean Squared Error (MSE) and SMAPE scores than the Segletes model when predicting residual values. The MLP performed better in the data point-by-data point comparison. It performed better in almost every measure for prediction and it did it with a 96% decrease in average runtime. The results also demonstrate that the MLP is faster and more accurate than the Segletes model (RQ1.3 and RQ1.4). The MLP is far superior to the Segletes model for prediction of perforation, residual velocity, and residual mass against monolithic metallic targets.

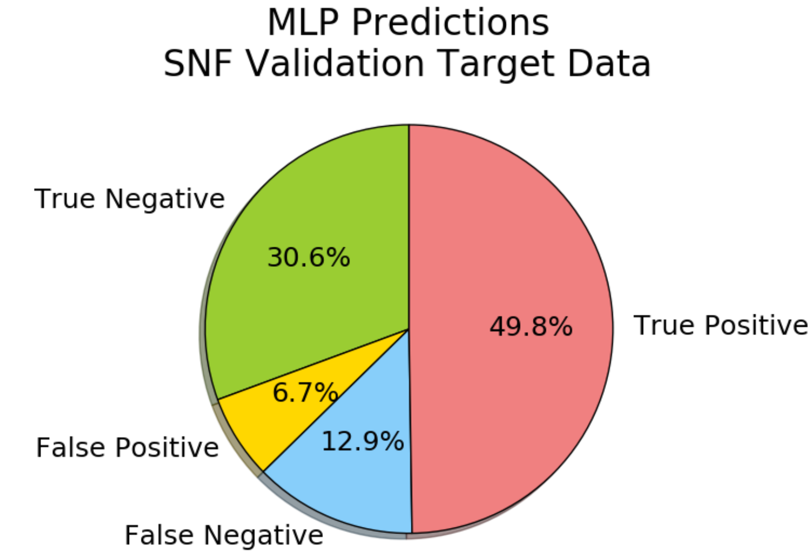


(a)

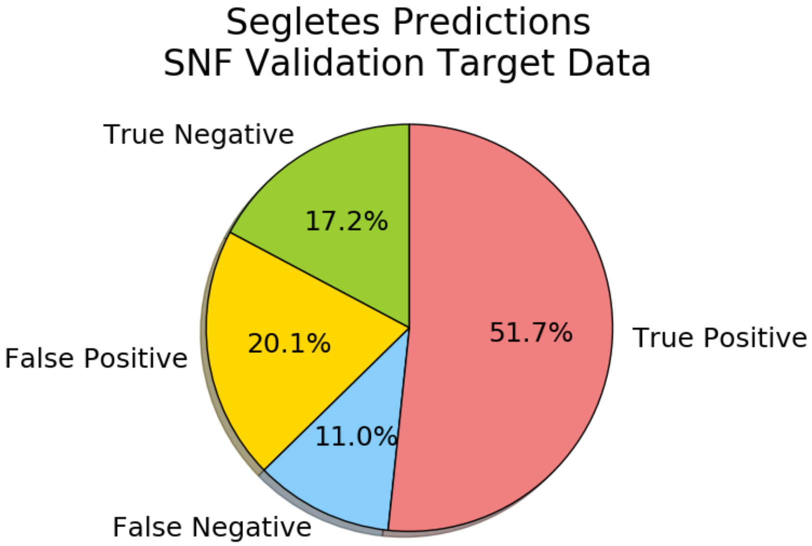


(b)

Figure 5.5: Distributions of classification outcomes of the MLP and the Segletes model for single element targets - Training Set

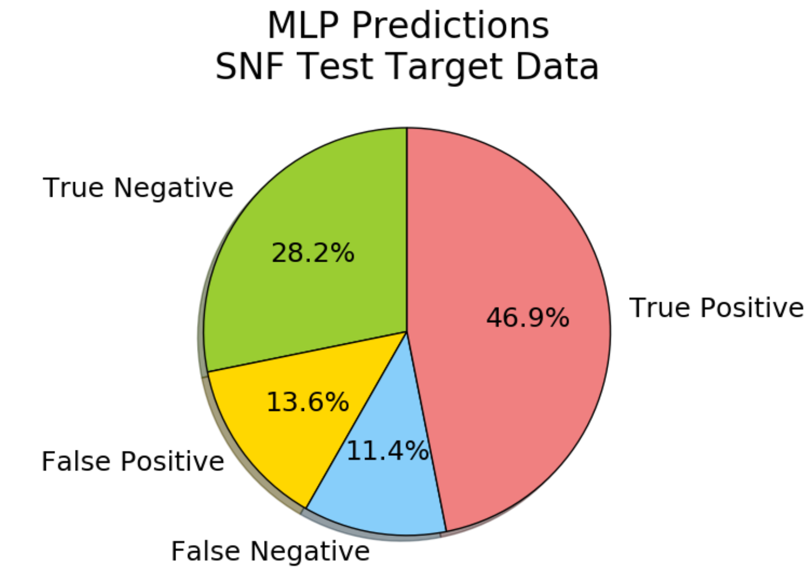


(a)

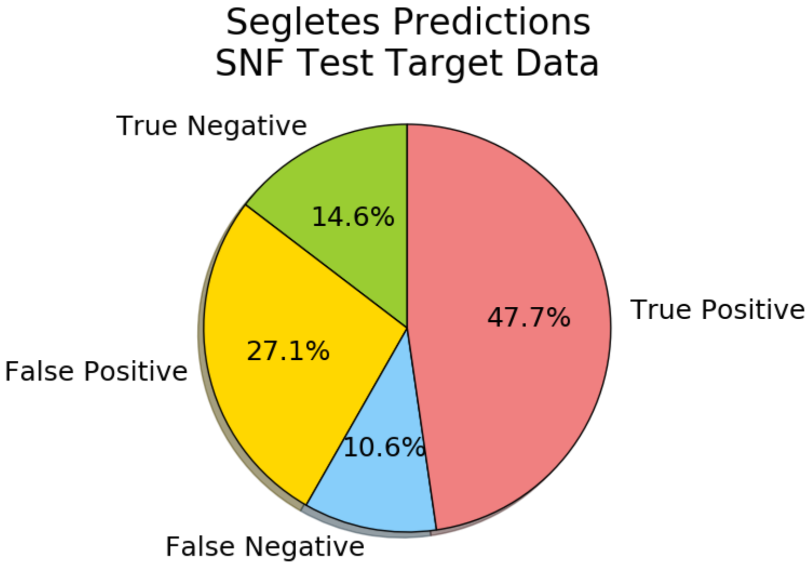


(b)

Figure 5.6: Distributions of classification outcomes of the MLP and the Segletes model for single element targets - Validation Set



(a)



(b)

Figure 5.7: Distributions of classification outcomes of the MLP and the Segletes model for single element targets - Test Set

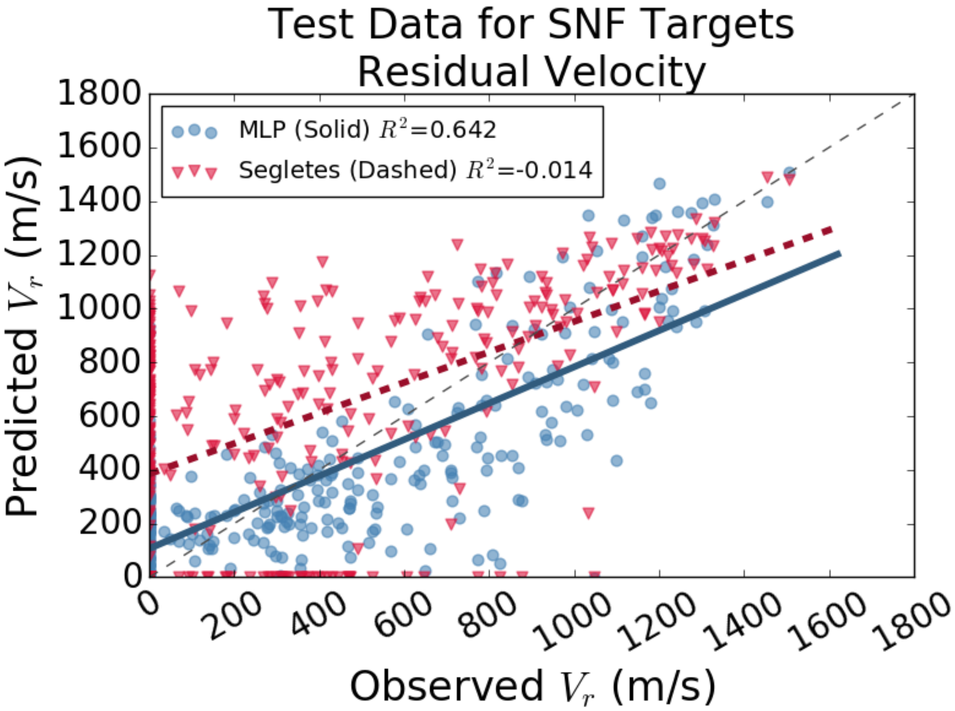


Figure 5.8: Scatter Plot for Residual Velocity

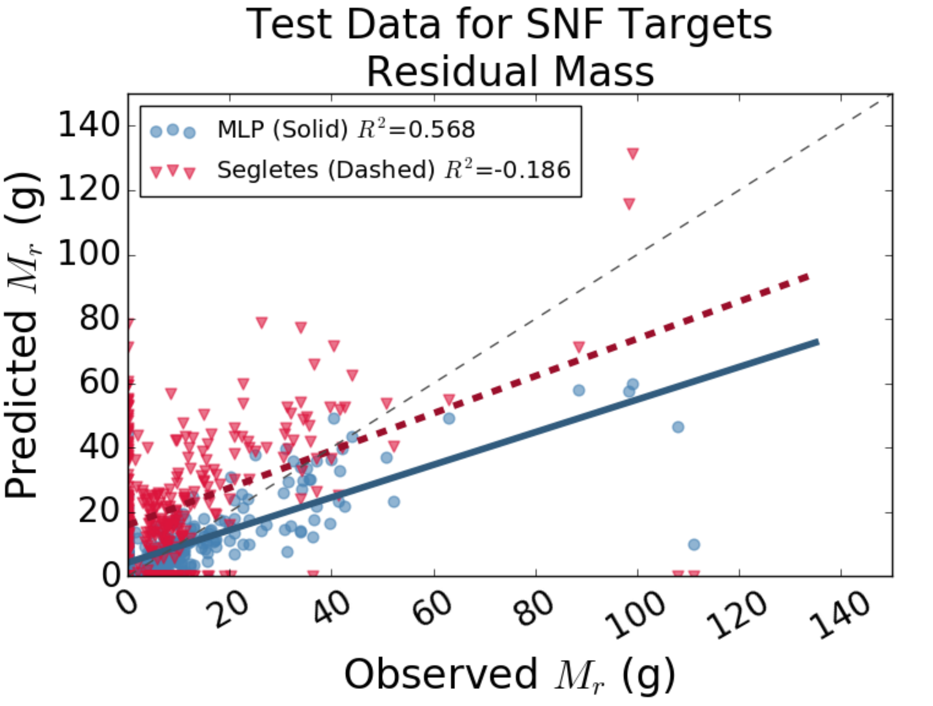


Figure 5.9: Scatter Plot for Residual Mass

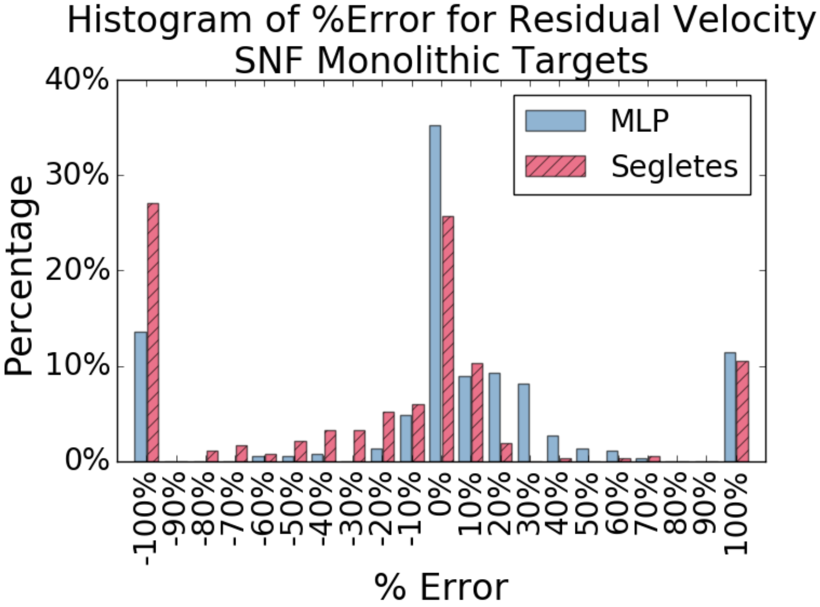


Figure 5.10: Symmetric Percent Error for residual velocity

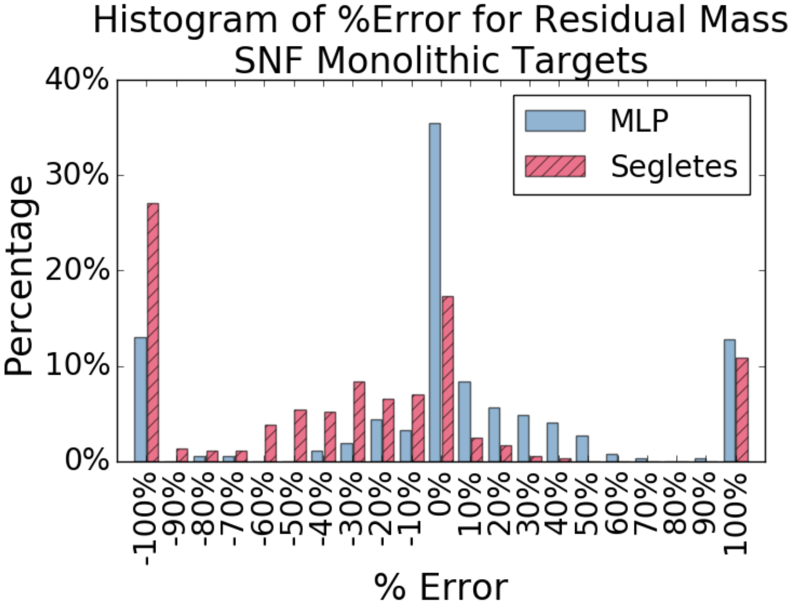


Figure 5.11: Symmetric Percent Error for residual mass

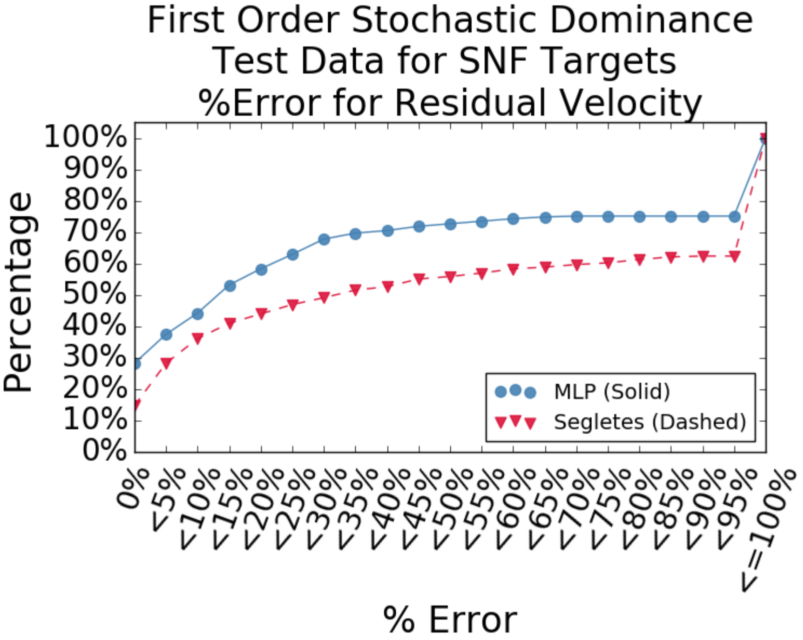


Figure 5.12: First Order Stochastic Dominance - Cumulative Percent Error for Vr

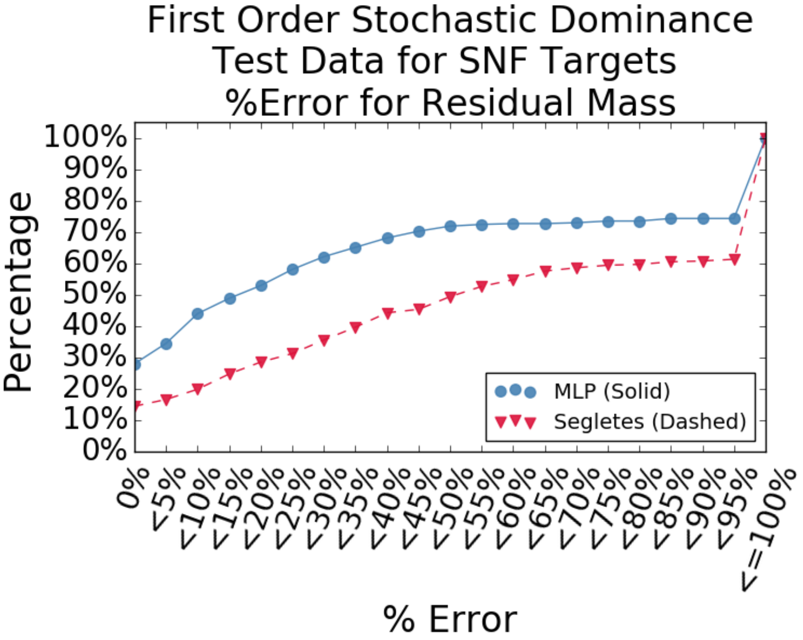


Figure 5.13: First Order Stochastic Dominance - Cumulative Percent Error for Mr

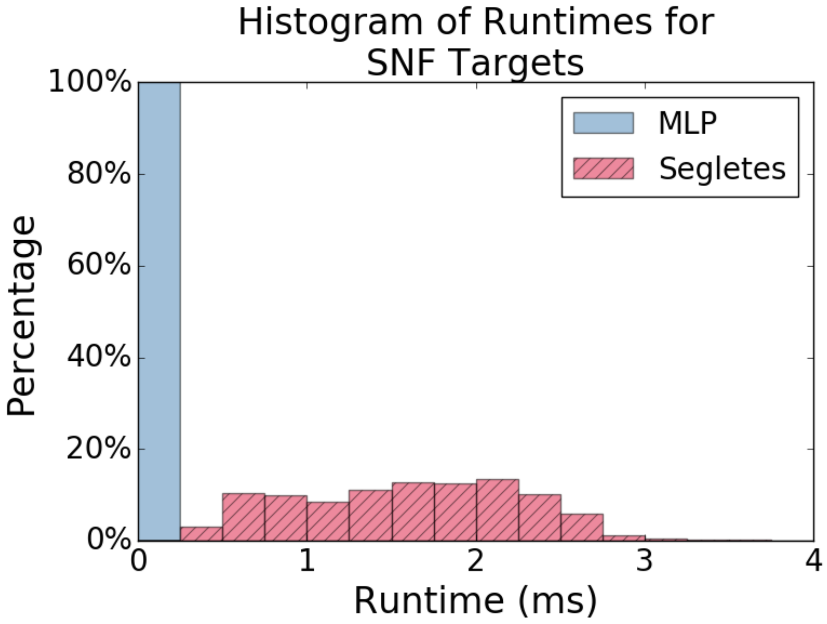


Figure 5.14: Runtimes

# Chapter 6

# MLP for Multi-Layer Metallic Armor

The Multi-Layered Perceptron (MLP) developed for monolithic metallic armor was used to predict the ballistic results of a Kinetic Energy Projectile (KEP) against multiple element targets by applying the MLP iteratively. This chapter will provide information on how it was applied iteratively, the results of the iterative process, a comparison to the Segletes model, and a summary. The work detailed in this chapter was used to answer RQ1.3, RQ1.4, and RQ1.5.

## 6.1 Methodology

The process for applying the MLP against finite multiple element targets (MAF and MNF), consisted of applying the MLP iteratively for each element. The iterative method was performed as follows:

1. Run the MLP against the current element of the target (first element, if this is the first iteration).

2. If the MLP predicted non-perforation then stop iterating.

3. If there are no more elements then stop iterating.

4. Use the outputs of the MLP from the current element to determine the inputs for the next element.

5. If the next element is air, skip that element and use the next element.

6. Start next iteration from Step 1 of this list.

Step 4 from the iterative method involves calculating the inputs for the next element from the outputs from the previous element. The MLP was not trained to predict residual length, however if the diameter of the projectile is assumed to stay constant an approximation can be calculated from (6.1).

$$l = \frac{m}{\pi(\frac{d}{2})^2\rho} \tag{6.1}$$

Where:

$l$ is the length of a cylinder

$m$ is the mass of a cylinder

$d$ is the diameter of a cylinder

$\rho$ is the density of a cylinder

## 6.2 Results

Based on the Symmetric Absolute Percentage Error (SAPE), there were 2 data points in the multiple element targets (MAF and MNF) set where the MLP had percent errors worse than 80%, that were not false positives or false negatives, when predicting residual velocity. Both of those data points were negative percent errors, meaning that the MLP overpredicted the residual velocity values for those data points. There were 579 total data points in the multiple element target set so they account for only 0.3% of the data in that set. For residual mass there were 5 data points in the multiple element targets set with %Error worse than 80% that were not false positives or false negatives. There were 579 total data points in the multiple element target set so they account for only 0.9% of the data in the set. For the multiple element targets set the MLP had 57 true negatives, 182 false positives, 58 false negatives, and 282 true positives. The high number of false positives shows that the MLP had a tendency to overpredict perforation.

When the MLP was trained it was trained on SNF targets that had air behind the target element. If that target was then replicated multiple times it would consist of alternating metallic elements and air elements. All of the MAF data that fit that criteria were defined as a "spaced" target. Likewise, all MNF targets and any MAF target that had two metallic elements without air between them was considered a non-spaced target. This division of the data was done to see if the MLP performed better on multi-element targets that were similar in makeup to the single element targets it was trained on. There were 541 "Spaced" target data points and 38 "NonSpaced" target data points in the multiple element target set.

The 2 data points with high %error for residual velocity and the 5 data points with high %error for residual mass were against "Spaced" targets. The MLP had 182 over predictions (false positives) and 58 under predictions (false negatives). All of the 182 data points with over predictions were against "Spaced" targets and those data points amount to 34% (182 out of 541) of all of the "Spaced" target data points. Of the 58 data points with under predictions, 34 were against "Spaced" targets and those data points amount to 6% (34 out of 541) of all of the "Spaced" target data points. The remaining 24 data points were "NonSpaced" targets and those data points amount to 63% (24 out of 38) of all of the "NonSpaced" target data points.

The results for the "NonSpaced" target data points show a definite tendency to under predict perforation for the MLP. This is most likely due to the MLP being applied in an iterative fashion. In essence, the iterative method is assuming independence for interaction at each element. However, they are not independent for any multiple element targets, but it is especially true for "NonSpaced" targets. As the projectile is penetrating an element there are effects already occurring in the elements proceeding it if the plastic zone of deformation has reached those elements. Similarly, when a projectile first strikes a target element it takes time for the plastic zone to form and for the materials to start flowing, but when it is moving from one element to the next and there is no air gap between them, the materials are already flowing and continue to flow. Since the MLP was trained on SNF targets, it is likely that the early process of plastic fluid flow formation was inherently learned by the model. By applying the MLP iteratively, we are over estimating the resistance of the target to the projectile which is leading to a high number of false negatives.

Since the MLP was trained against SNF data it is possible that it may have learned some of the back of the element effects (i.e. spalling, plugging, and the like). What it cannot account for is the effect of the air gap on degradation of the projectile. This should lead to an over-prediction of residual velocity for spaced targets. The results for the “Spaced” target data points show a definite tendency to over predict perforation for the MLP. This is most likely due to the MLP not being trained on MAF targets. When a projectile exits a target element into an air element there are forces that act upon it that degrade its performance slightly and in some circumstances can cause the projectile to fracture. The MLP would have no way of predicting those cases since it was never trained on data that contained those cases.

## 6.3 Comparison to Segletes Model

### 6.3.1 Perforation Prediction Results

The MLP performed comparable to the Segletes model for MAF and MNF targets (see Table 6.1). The Mean Squared Error (MSE) for both models were very close, their scores for ACC and MCC were also very close. The MLP had a higher FPR and the Segletes model had a higher FNR. That generally means that when the MLP was wrong it was typically because it predicted perforation when the data point had a result of non-perforation and that when the Segletes model was wrong it was typically because it predicted non-perforation when the data point had a result of perforation. Those results continued for the subset of “Spaced” target data points, with a slight improvement for the MLP. The “NonSpaced” target data points show where the MLP really struggled to predict perforation outcomes. The MLP had a FNR of 70.6%. It also appears that the Segletes model performs better for modeling “NonSpaced” targets, since the ACC was 71.1% and the MCC was 0.42 compared to 56.4% and 0.12 for “Spaced” targets.

Table 6.1: Perforation Prediction Statistics for Multiple Element Targets

| All Multiple Element Targets | | |
|---|---|---|
| | MLP | Segletes |
| Mean Square Error (MSE) | **1.343** | 1.3822 |
| Accuracy (ACC) | **58.5%** | 57.3% |
| False Positive Rate (FPR) | 76.2% | **45.2%** |
| False Negative Rate (FNR) | **17.1%** | 40.9% |
| True Positive Rate (TPR) | **82.9%** | 59.1% |
| True Negative Rate (TNR) | 23.8% | **54.8%** |
| Matthews Correlation Coefficient (MCC) | 0.08 | **0.14** |

| Spaced Targets Subset | | |
|---|---|---|
| | MLP | Segletes |
| Mean Square Error (MSE) | **1.2936** | 1.4134 |
| Accuracy (ACC) | **60.1%** | 56.4% |
| False Positive Rate (FPR) | 77.4% | **46.0%** |
| False Negative Rate (FNR) | **11.1%** | 41.8% |
| True Positive Rate (TPR) | **88.9%** | 58.2% |
| True Negative Rate (TNR) | 22.6% | **54.0%** |
| Matthews Correlation Coefficient (MCC) | **0.15** | 0.12 |

| NonSpaced Targets Subset | | |
|---|---|---|
| | MLP | Segletes |
| Mean Square Error (MSE) | 2.0463 | **0.9379** |
| Accuracy (ACC) | 36.8% | **71.1%** |
| False Positive Rate (FPR) | **0.0%** | **0.0%** |
| False Negative Rate (FNR) | 70.6% | **32.4%** |
| True Positive Rate (TPR) | 29.4% | **67.6%** |
| True Negative Rate (TNR) | 100.0% | **100.0%** |
| Matthews Correlation Coefficient (MCC) | 0.2 | **0.42** |

### Perforation Prediction Pie Charts

The pie charts in Figures 6.1, 6.2, and 6.3 show the distribution of classification outcomes for the MLP and the Segletes model for the set of all MAF and MNF data as well as the two subsets of that data (Spaced and NonSpaced). For all of the multiple element target data the MLP had 57 true negatives, 182 false positives, 58 false negatives, and 282 true positives (the percentages are shown in Figure 6.1a). For the same set of data, the Segletes model had 131 true negatives, 108 false positives, 139 false negatives, and 201 true positives (the percentages are shown in Figure 6.1b). For the Spaced set, the MLP had 53 true negatives, 182 false positives, 34 false negatives, and 272 true positives (the percentages are shown in Figure 6.2a). For the same set of data, the Segletes model had 127 true negatives, 108 false positives, 128 false negatives, and 178 true positives (the percentages are shown in Figure 6.2b). For the NonSpaced set, the MLP had 4 true negatives, 0 false positives, 24 false negatives, and 10 true positives (the percentages are shown in Figure 6.3a). For the same set of data, the Segletes model had 4 true negatives, 0 false positives, 11 false negatives, and 23 true positives (the percentages are shown in Figure 6.3b).

### 6.3.2 Residual Value Estimation Results

Both models struggled to predict residual values for multiple element targets. Table 6.2 lists the MSE for the MLP and the Segletes model for all three data sets. The MSE scores for residual velocity prediction by the Segletes model were roughly 47% better than the MLP scores. However, the MSE scores for residual mass prediction by the MLP were roughly 28% better than the Segletes model scores. The Symmetric Mean Absolute Percentage Error (SMAPE) scores were nearly identical to the Segletes model. The MLP had a SMAPE score of 51.8% compared to the Segletes models score of 49.5% for residual velocity. For residual mass, the MLP had a SMAPE score of 58.8% compared to the Segletes models score of 57.7%. The scores for the "Spaced" and "NonSpaced" targets are provided to further show the difference in how the MLP performed against the two different types of targets.

#### Residual Value Scatter Plots

In the scatter plots (Figures 6.4, 6.5, and 6.6), observed values are along the $x$-axis and predicted values are along the $y$-axis. If a model performed perfectly then the values would fall perfectly along a line with slope equal to 1 and a $y$-intercept equal to 0. Cases of incorrect perforation prediction (false positives) can be seen plotted on the $y$-axis with a value of 0 for $x$. Cases of incorrect non-perforation prediction (false negatives) can be seen plotted on the $x$-axis with a value of 0 for $y$. The tendency of the MLP to over predict can be seen on the $y$-axis in Figures 6.4a and 6.4b. The MLP has 182 points on the $y$-axis and 58 on the $x$-axis. In those same figures the incorrect predictions for the Segletes model are shown to be more balanced; there are 108 on the $y$-axis and 139 on the $x$-axis. For those data points where perforation was predicted correctly, the MLP tends

Table 6.2: Mean square error and symmetric mean absolute percentage error statistics for residual value predictions of multiple element targets

| All Multiple Element Targets | | |
|---|---|---|
| | MLP | Segletes |
| MSE for Residual Velocity | 0.2998 | **0.1587** |
| MSE for Residual Mass | **0.0886** | 0.1226 |
| SMAPE for Residual Velocity | 51.8% | **49.5%** |
| SMAPE for Residual Mass | 58.8% | **57.7%** |

| Spaced Targets | | |
|---|---|---|
| | MLP | Segletes |
| MSE for Residual Velocity | 0.2813 | **0.1616** |
| MSE for Residual Mass | **0.0864** | 0.1247 |
| SMAPE for Residual Velocity | **50.2%** | 50.5% |
| SMAPE for Residual Mass | **58.3%** | 59.3% |

| NonSpaced Targets | | |
|---|---|---|
| | MLP | Segletes |
| MSE for Residual Velocity | 0.5632 | **0.1173** |
| MSE for Residual Mass | 0.1191 | **0.0927** |
| SMAPE for Residual Velocity | 75.4% | **34.8%** |
| SMAPE for Residual Mass | 66% | **34.7%** |

to be better balanced between over and under estimations for residual velocity than the Segletes model. For residual mass, the MLP performed better than the Segletes model, but it appears to under estimate for residual masses that are large. This could be due to the lack of large valued residual masses in the training data. Out of 1877 total data points in the training set, 89 of them (less than 5%) contained residual masses greater than 40 g. For "NonSpaced" targets (Figures 6.6a and 6.6b) the MLP did poorly, primarily because of the high percentage of under predictions. For the MLP, under predictions accounted for 24 out of 38 of the data points (63%). For those data points where perforation was predicted correctly the MLP actually performed fairly well, but with the already mentioned slight tendency to under predict residual mass for large masses.

In order to get a better view of how well the two models performed predicting residual values the following scatter plots (Figures 6.7a and 6.7b) were created. They are the same as those just shown (Figures 6.4a and 6.4b), but with all cases of false positives and false negatives removed. Looking at Figure 6.7a it becomes a little clearer that for those data points where the MLP correctly predicted perforation, it has a better correlation then the Segletes model in predicting residual velocity. In other words the errors are slightly larger but are better centered around the true values. The Segletes model tends to slightly under-predict at larger residual velocity values. For residual mass (Figure 6.7b) the MLP has better prediction for residual masses less than 20 g, slightly under-predicts for values larger than 40 g, and severely under-predicts for values larger than 100 g. It appears that for the cases where the residual mass is greater than 100 g that there may be something wrong with the data because both models drastically under-predict for those data points.

### Residual Value Percent Error

The Symmetric Percentage Error (SPE) histograms can be seen in Figures 6.8, 6.9, and 6.10. The false positives show up in the -100% bin on the far left of the plot and the false negatives show up in the 100% bin on the far right of the plot. Due to how the SPE is calculated, if the percent error is negative then that is indicative of an over estimation of the value observed and if the percent error is positive then that is indicative of an under estimation of the value observed. Overall, the MLP performed comparable to the Segletes model. For residual velocity (Figure 6.8a), the Segletes model tended to under predict perforation and under estimate residual values. The MLP tended to over predict perforation, but was fairly balanced in the estimation of residual values. For residual mass (Figure 6.8b), again both models performed similarly. The Segletes model did have a higher percent of false negatives than the MLP, but the MLP had a higher percent of false positives than the Segletes model.

The “Spaced” targets and “NonSpaced” targets data are presented to again show how the MLP performed differently for those two types of targets. The “Spaced” targets can be seen in Figures 6.9a and 6.9b and since the “Spaced” targets accounted for most of the multiple element targets, the results are similar to the larger data set. For the “NonSpaced” targets there was a very high percent of false negatives, but both models performed well for those data points with correct perforation predictions. For residual velocity (Figure 6.10a) and residual mass (Figure 6.10b) most of the data points, where perforation was correctly predicted, resulted in percent errors in the range of -30% to 30%.

The cumulative distribution plots in Figure 6.11 show that the Segletes model has a slightly better performance than the MLP for multiple element targets (MAF and MNF). This is due in large part to the number of non-perforations that the Segletes model correctly predicted and how many the MLP incorrectly predicted. Those cases show up as 0% error in the plots and gives the Segletes model an immediate advantage. The performance of both models is strikingly similar, with only a sight edge given to the Segletes model. However, it is not definitive that the Segletes model is the better model because it is not first order stochastic dominant. Neither model is first order stochastically dominate over the other, so no definite claim of superiority can be made for either.

### Data Point-by-Data Point Prediction Comparison

For multiple element targets (MAF and MNF) the MLP performed on par with the predictive capabilities of the Segletes model for both the classification problem of determination of perforation and the regression problem of predicting the residual velocity and mass. For prediction of residual velocity the MLP performed better than the Segletes model on 217 data points and performed the same as the Segletes model on 195 data points. That means that the MLP performed as well or better than the Segletes model for 412 out of 579 data points (71%).

For prediction of residual mass the MLP performed better than the Segletes model on 252 data points and performed the same as the Segletes model on 195 data points. That means that the MLP performed as well or better than the Segletes model for 447 out of 579 data points (77%).

Out of all of the data points where the Segletes model performed better, there were 78 data points that resulted in false positives for the MLP and 20 that resulted in false negatives. There were 167 data points (29%) that the Segletes model did better than the MLP for predicting residual velocity. The 78 false positive predictions by the MLP accounted for 47% and the 20 false negative predictions by the MLP accounted for 12% of the data points. The remaining 69 data points were ones that the MLP correctly predicted perforation, of those data points the lowest error for the MLP was -3%, the highest error was -91%, and the average error (SMAPE) was 29%.

There were 132 data points (23%) that the Segletes model did better than the MLP for predicting residual mass. The 78 false positive predictions by the MLP accounted for 59% and the 20 false negative predictions by the MLP accounted for 15% of the data points. The remaining 34 data points were ones that the MLP correctly predicted perforation, of those data points the lowest error for the MLP was 1%, the highest error was -97%, and the average error (SMAPE) was 34%.

### 6.3.3 Runtime Comparison

Although the MLP did not perform as well on multiple element targets as it did on single element targets, it did manage to perform at least as well as the Segletes model for multiple element targets. Where the true benefit of the MLP shows is in the runtimes for the multiple element targets. As mentioned in the previous chapter, no attempt was made to optimize either of the models for runtime, so the runtimes are only comparable for the currently unoptimized state of both models.

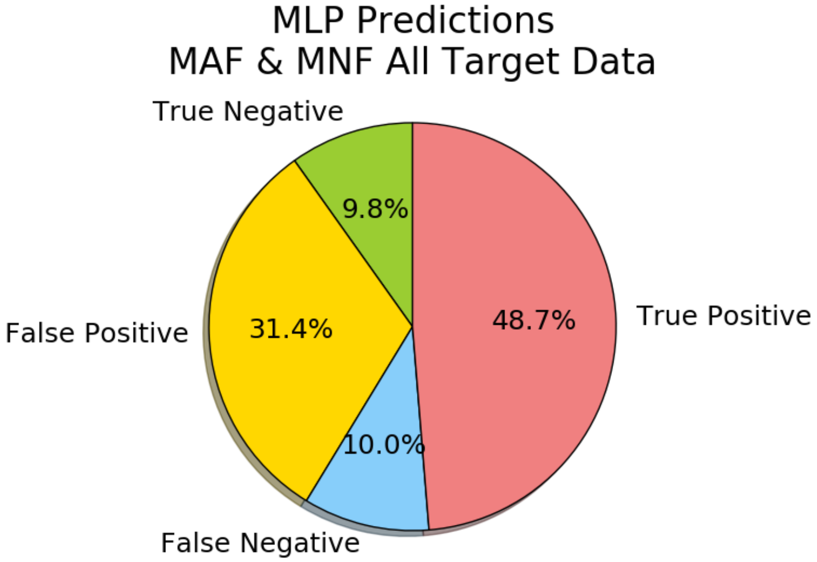
In Figure 6.12 the bins are runtime in milliseconds and the percentage is the percent of runs that had runtimes that fell into that bin. As an example, in Figure 6.12a there were 343 out of 579 MLP runs that took less than 0.25 ms for a value of 59% and there were 62 out of 2442 Segletes model runs that took between 1.75 ms and 1.0 ms for a value of 11%. The average runtime for the MLP model against MAF and MNF targets was 1.46 ms and the average runtime for the Segletes model was 6.78 ms. Figure 6.12b provides the MAF and MNF runtimes averaged by the number of elements in the target. The average runtime per target element for the MLP model against MAF and MNF targets was 0.49 ms and the average runtime for the Segletes model was 2.26 ms.

## 6.4 Summary

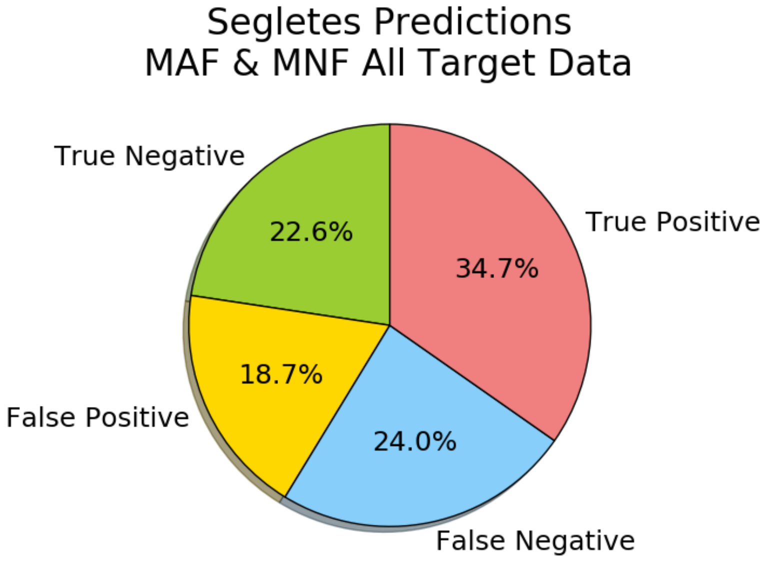
This chapter provided results that can be used to answer RQ1.3, RQ1.4, and RQ1.5. By applying the MLP iteratively it was possible for the MLP to model the terminal ballistics of KEPs against multiple element targets. The model did well against targets with air gaps between plates (“Spaced” targets) but did not do as well on those that had no air gaps between plates (“NonSpaced” targets). Overall, the MLP performed well and can be used for modeling of KEPs against multiple element targets (RQ1.3).

The MLP performed slightly better than the Segletes model at predicting perforation of multiple element targets. The MLP predicted perforation correctly 58.5% of the time and the Segletes model was correct 57.3% of the time. The MLP was slightly better at predicting residual mass than the Segletes model, but the Segletes model was slightly better at predicting residual velocity. The MLP was better at making predictions for multiple element targets that were “Spaced”, but the Segletes model was better at making predictions for “NonSpaced” targets. Overall, both models had areas of strength and weakness, but overall they both performed about the same on multiple element targets. The MLP performed similar to the Segletes model in accuracy (RQ1.4).

The MLP performed as well as the Segletes model at predicting perforation and residual, but it did it with a 78% decrease in runtime. Given the similar prediction performance of both models and the drastic reduction in runtime of the MLP, the MLP is the better model to use for multiple element targets. The MLP was drastically faster than the Segletes model when predicting the terminal ballistics of KEPs against multiple element targets (RQ1.5).

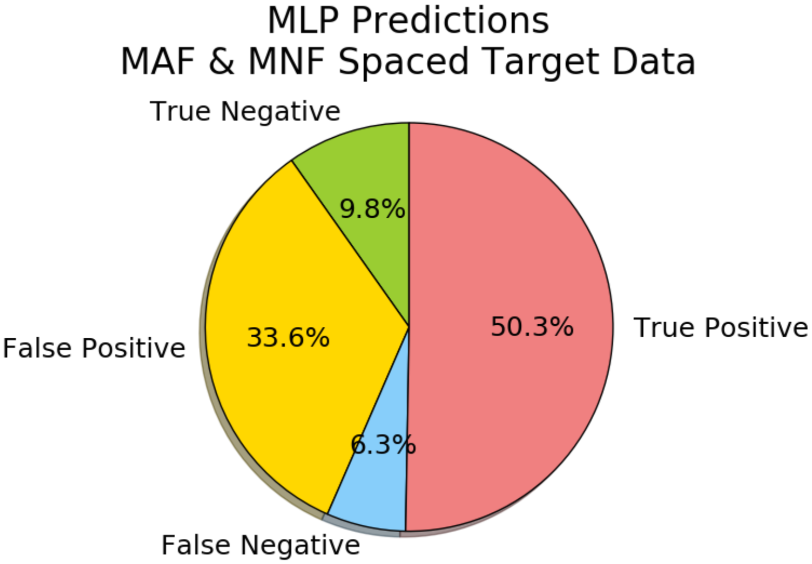


(a)

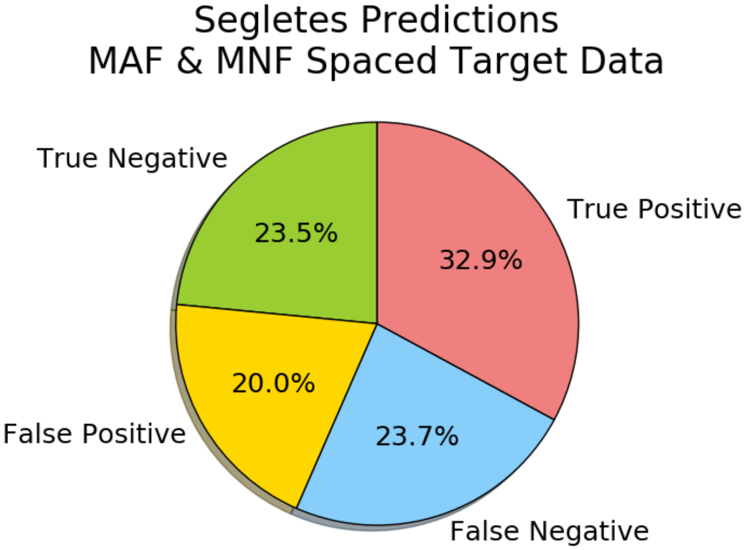


(b)

Figure 6.1: Distributions of classification outcomes of the MLP and the Segletes model for multiple element targets - All Data

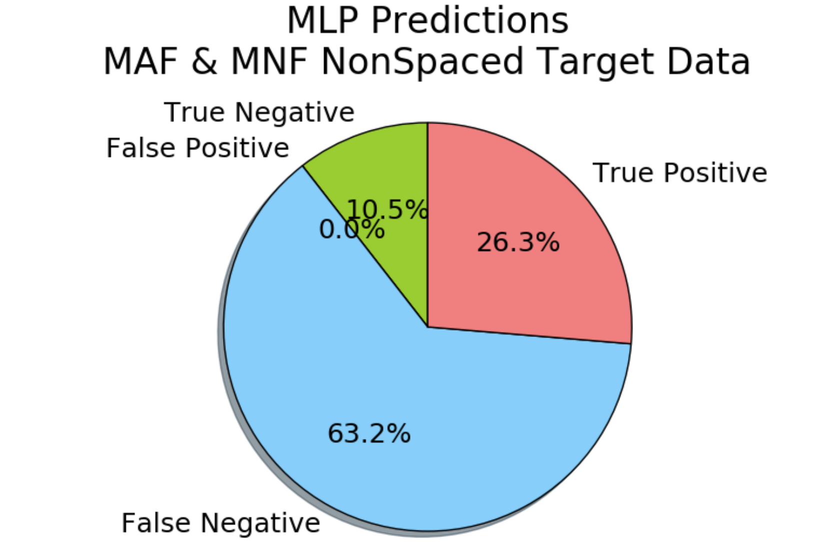


(a)

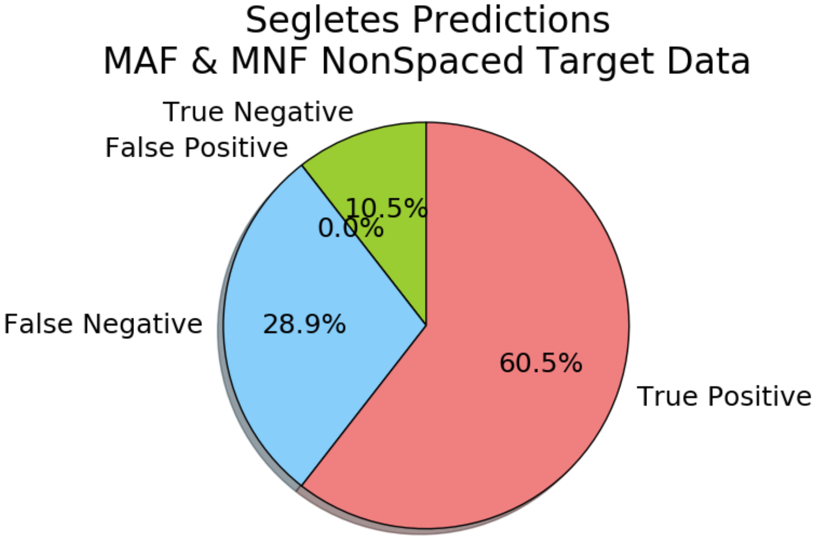


(b)

Figure 6.2: Distributions of classification outcomes of the MLP and the Segletes model for multiple element targets - Spaced Data

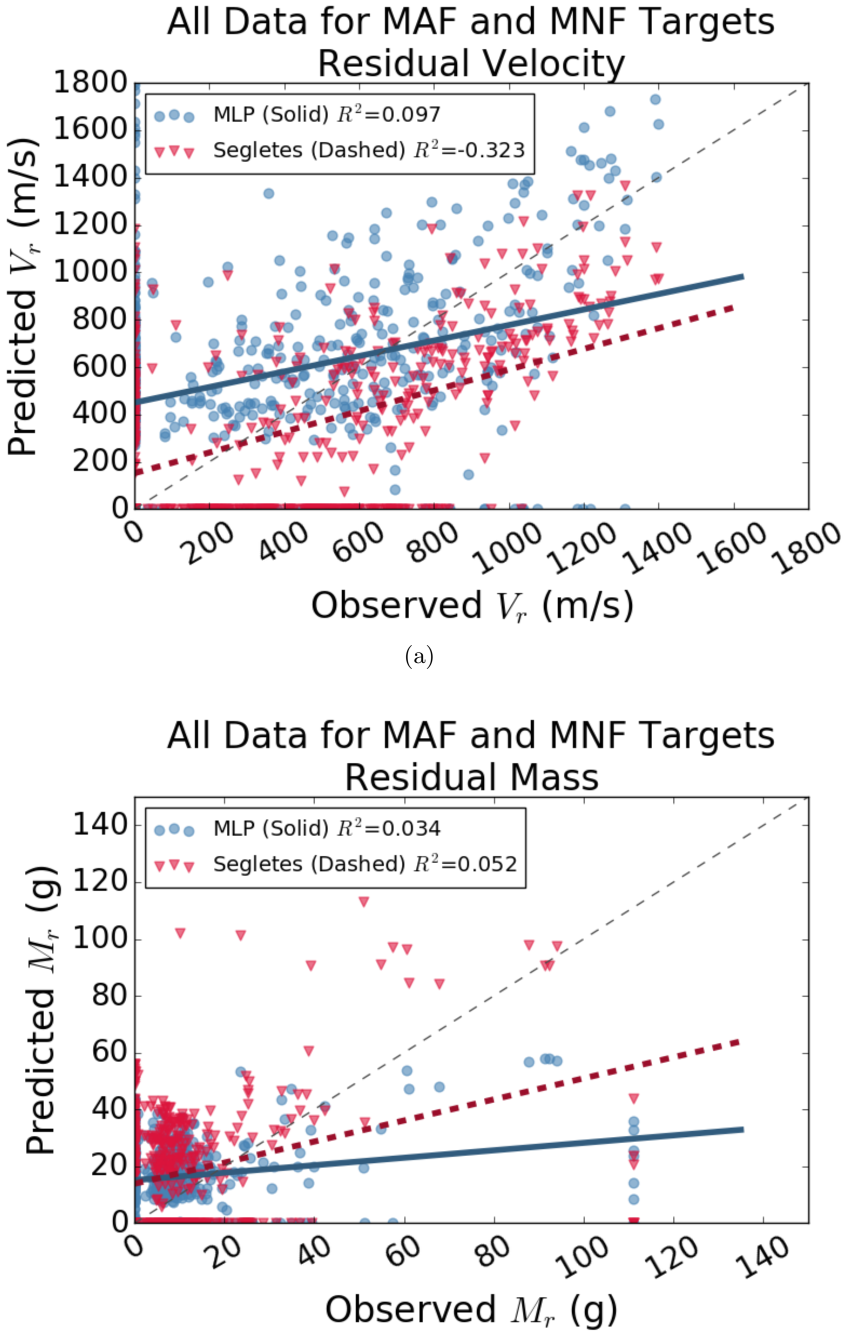


(a)



(b)

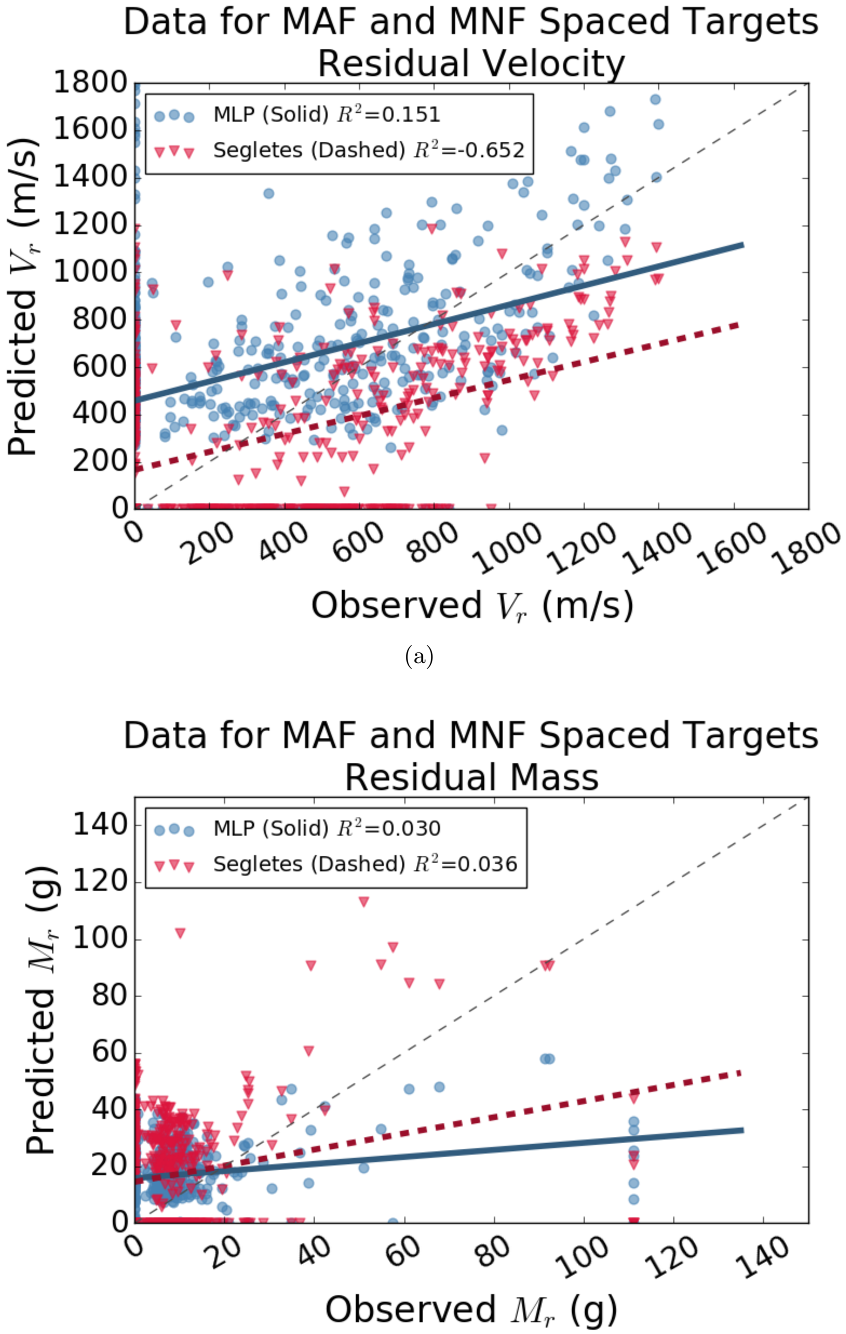
Figure 6.3: Distributions of classification outcomes of the MLP and the Segletes model for multiple element targets - NonSpaced Data



(a)

(b)

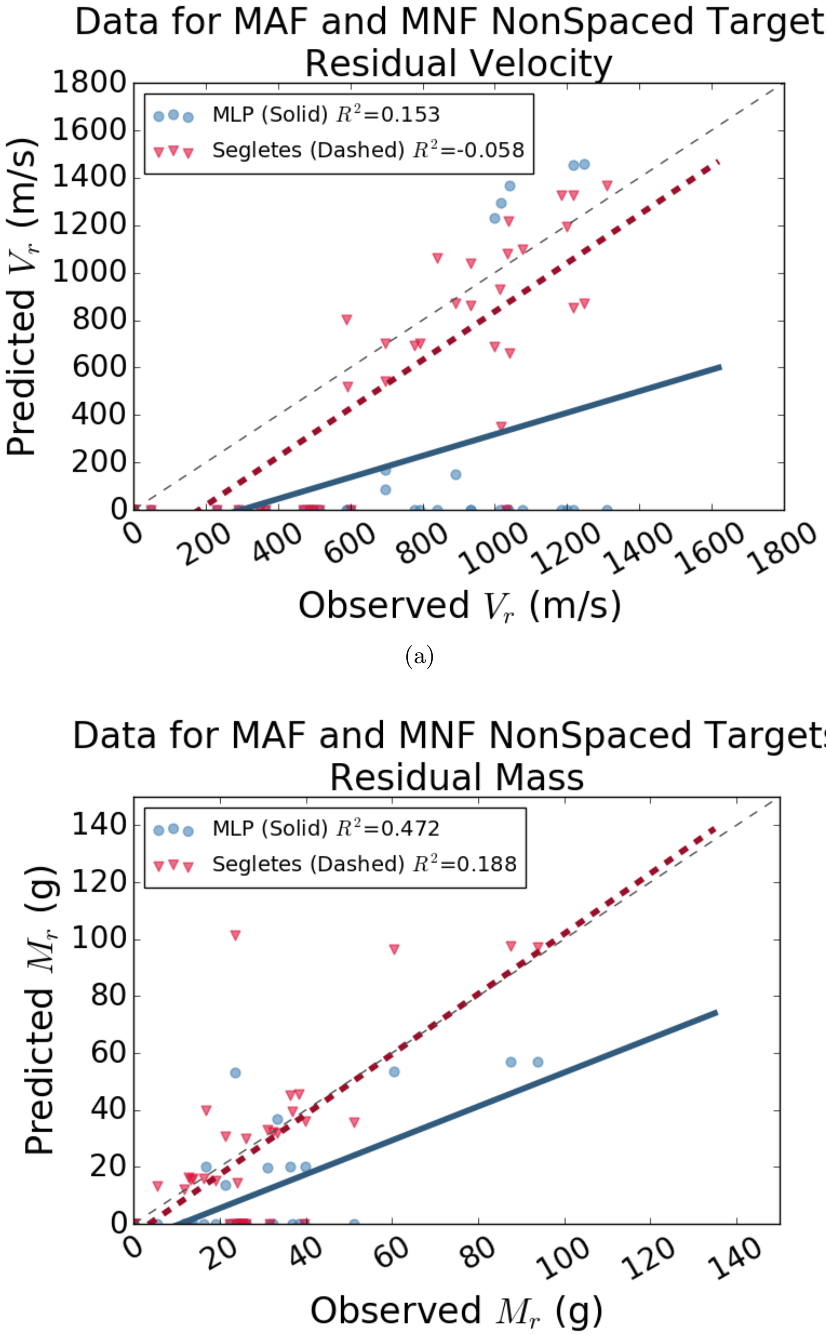
Figure 6.4: Scatter plots of observed vs predicted values - All Data



(a)

(b)

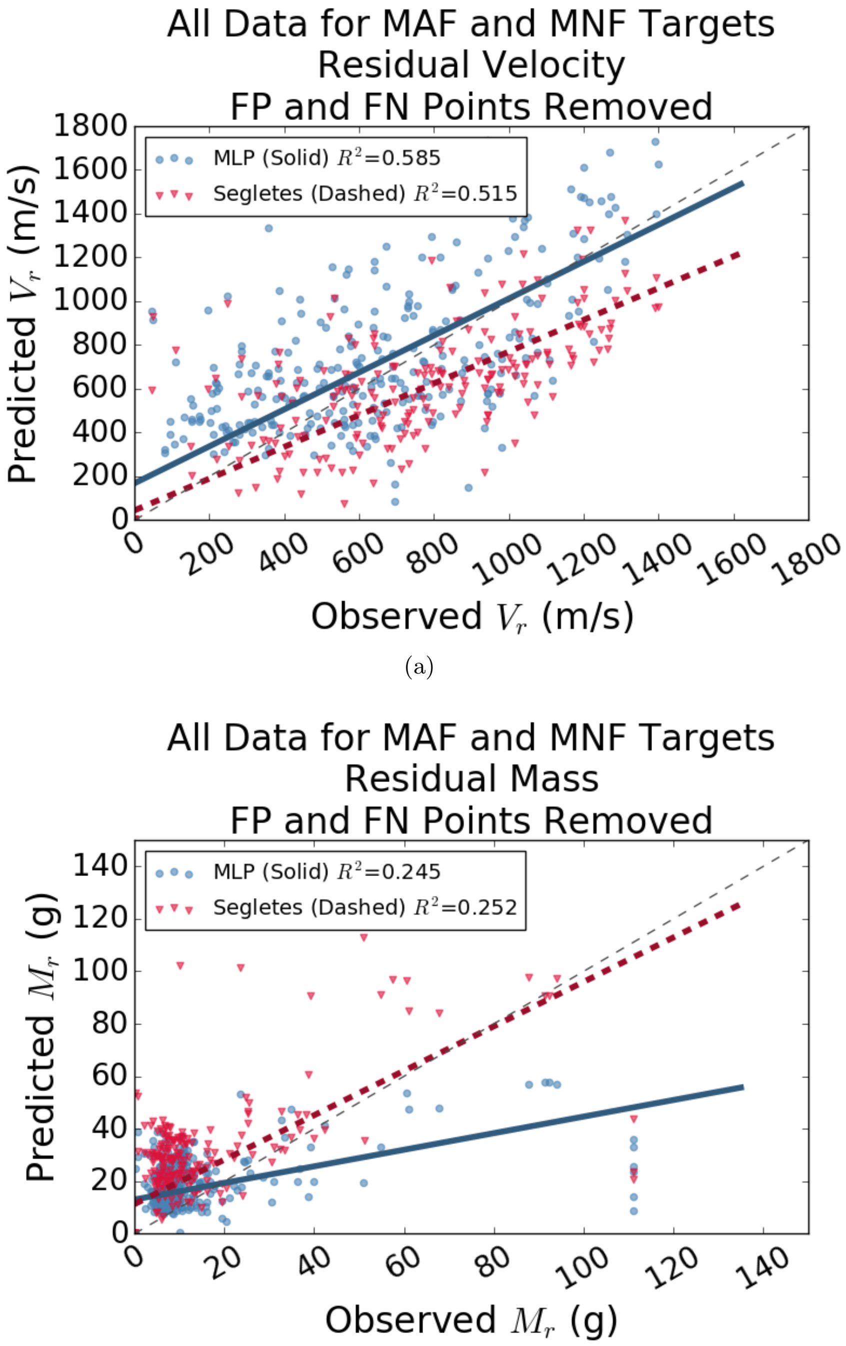
Figure 6.5: Scatter plots of observed vs predicted values - Spaced Data



(a)

(b)

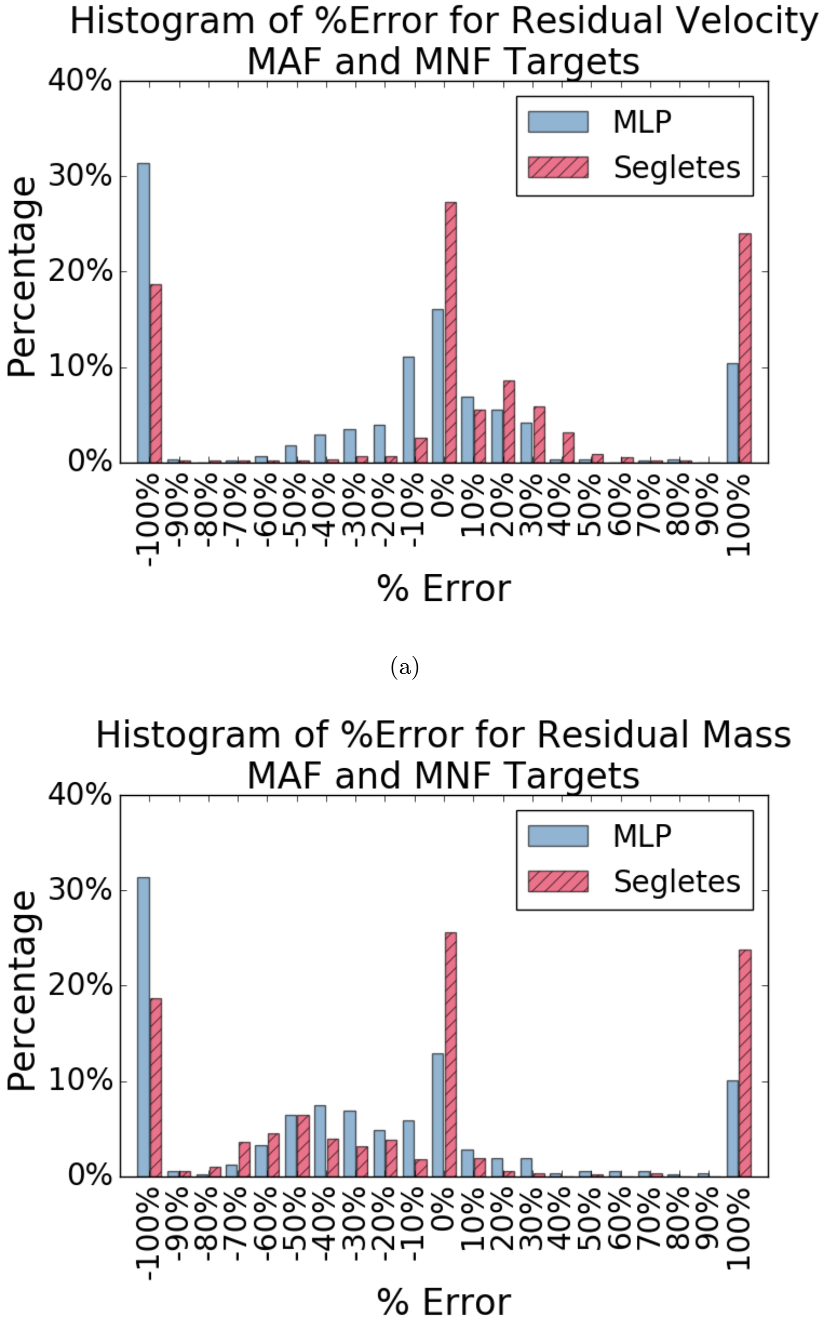
Figure 6.6: Scatter plots of observed vs predicted values - NonSpaced Data



(a)

(b)

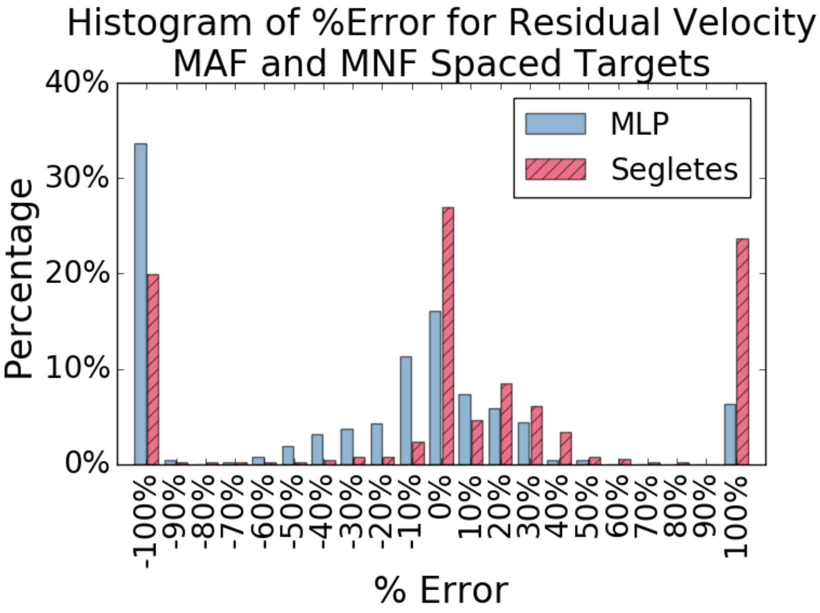
Figure 6.7: Scatter plots of observed vs predicted values, with false positives and false negatives not plotted



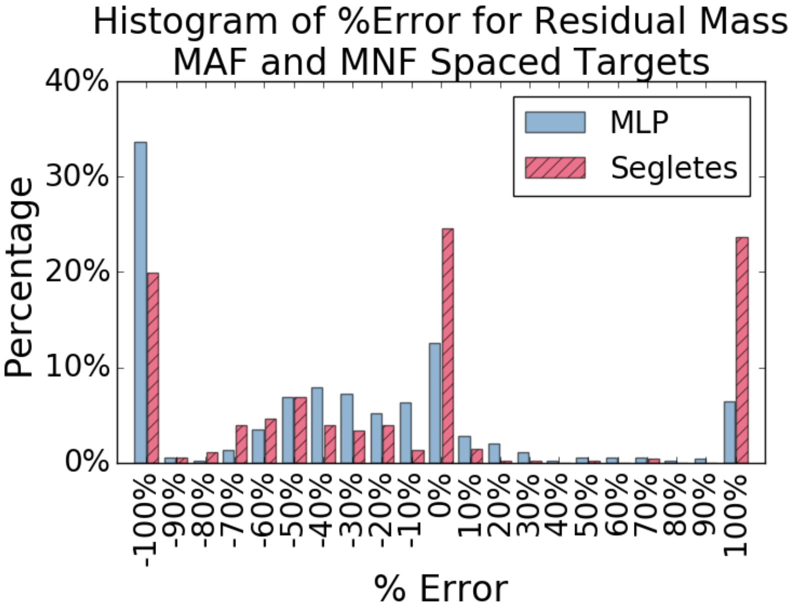
(a)

(b)

Figure 6.8: Histograms of percent error for MLP and the Segletes model predictions of residual velocity and residual mass - All Data

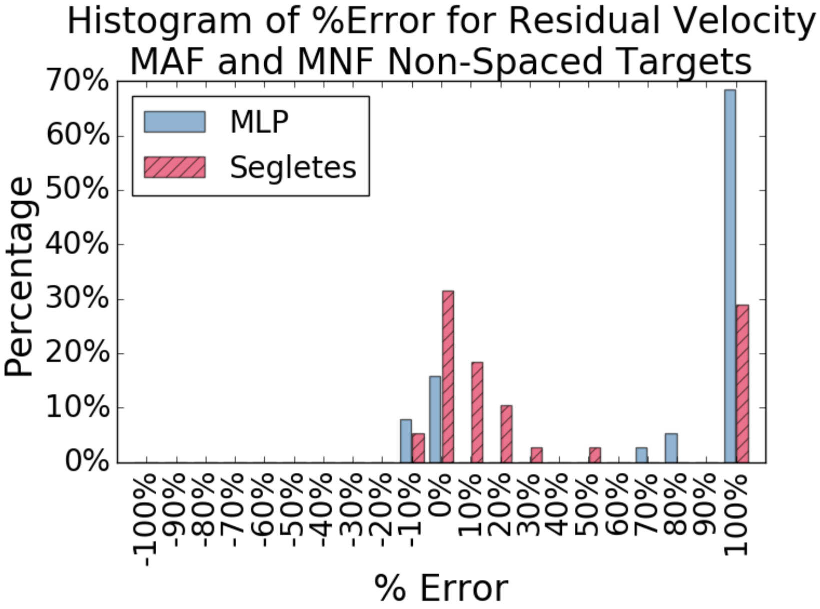


(a)

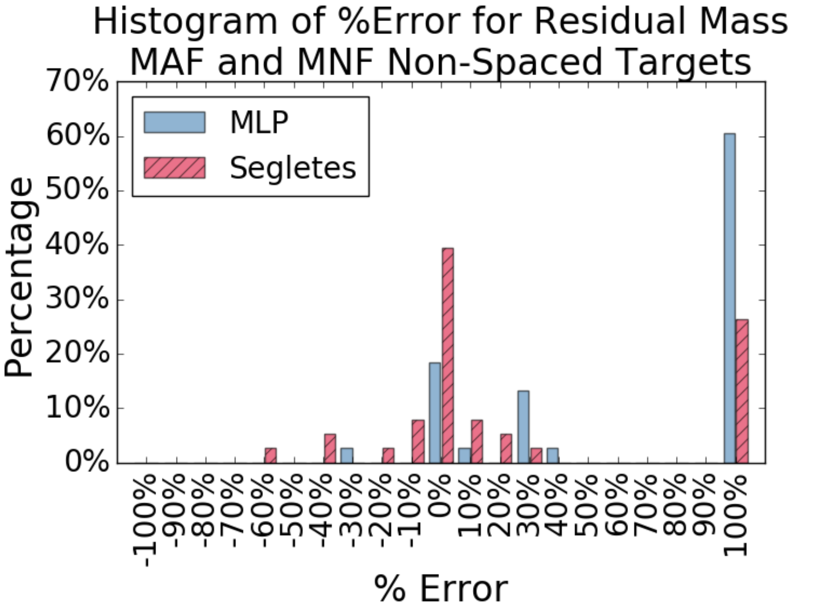


(b)

Figure 6.9: Histograms of percent error for MLP and the Segletes model predictions of residual velocity and residual mass - Spaced Data

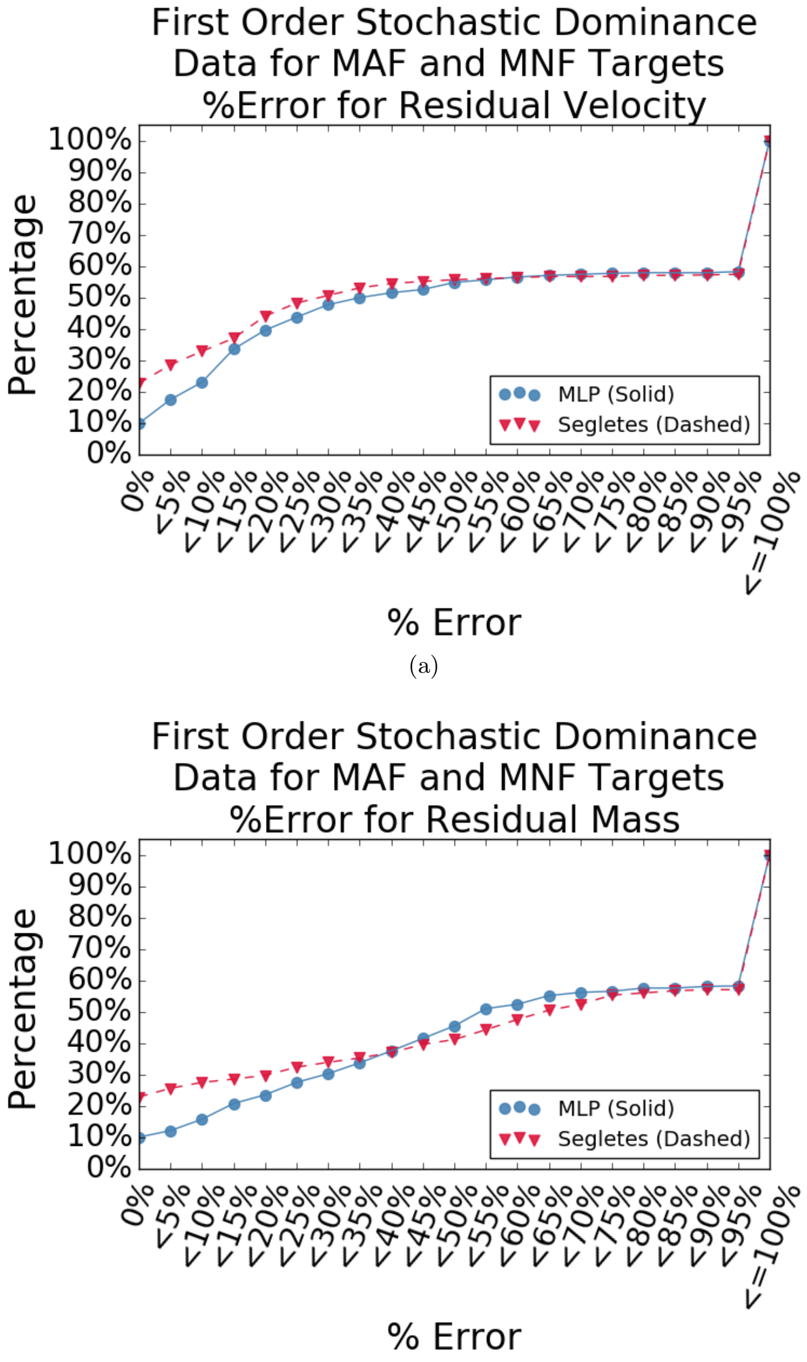


(a)



(b)

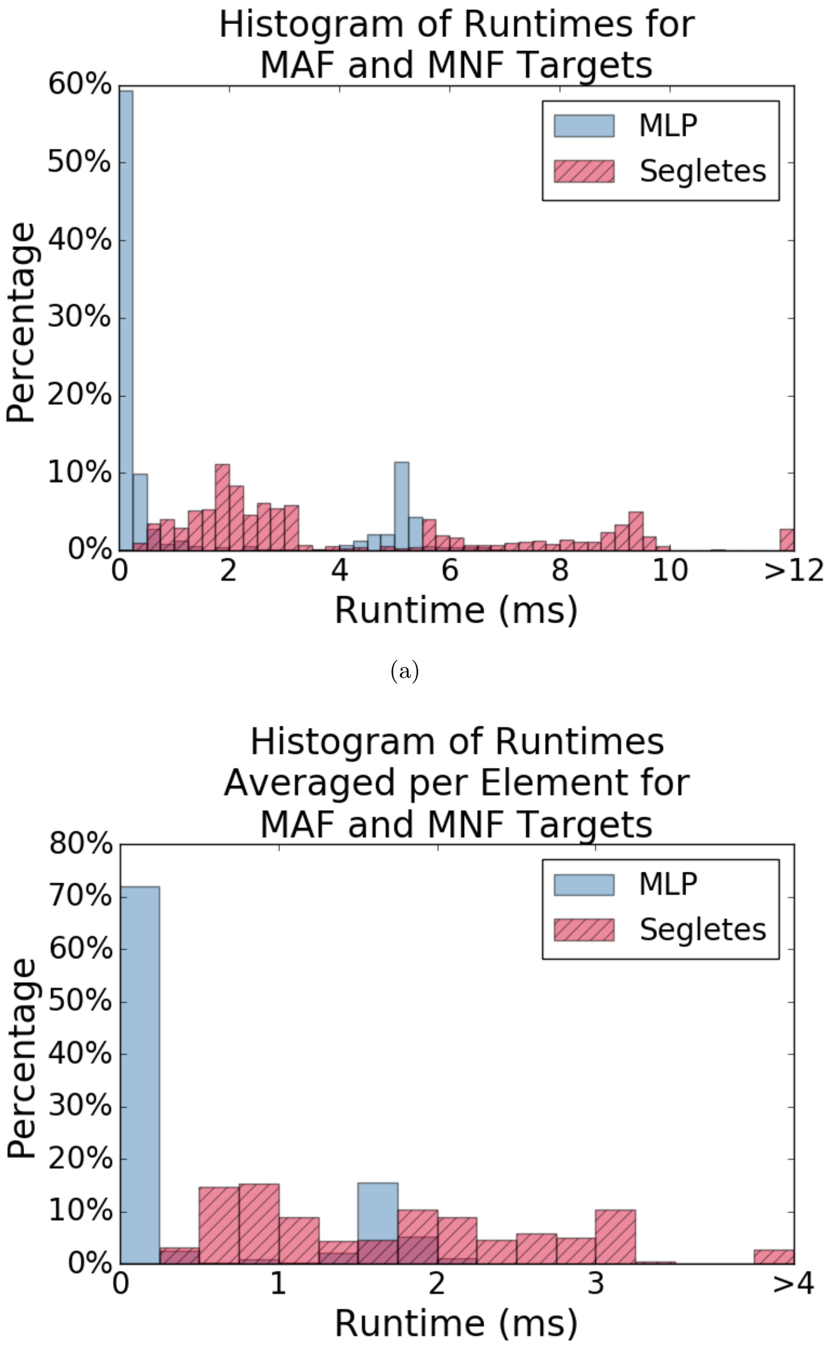
Figure 6.10: Histograms of percent error for MLP and the Segletes model predictions of residual velocity and residual mass - NonSpaced Data



(a)

(b)

Figure 6.11: First order stochastic dominance plots



(a)

(b)

Figure 6.12: Model runtimes for multiple element targets

# Chapter 7

# Conclusion

At the very beginning of this research, when the use of an Artificial Neural Network (ANN) to model the terminal ballistics of a Kinetic Energy Projectile (KEP) was being proposed, it was not clear if it would be effective. This led to the primary research question of: *RQ1: Can an ANN, used to model the terminal ballistics of KEPs, be developed that is both fast, accurate, and generalized?*

The following section will answer RQ1, by addressing each of the subquestions that define the primary research question. The answers to the subquestions will support the final answer of yes, an ANN can be used to model the terminal ballistics of a KEP effectively.

## 7.1 Revisit of Research Questions

**RQ 1.1 – What target or threat parameters have the most influence on terminal ballistics results?**

Through lengthy discussions with ballistics experts of the U.S. Army Research Laboratory (ARL) and trial and error during the early designs for the Multi-Layered Perceptron (MLP), 11 parameters were selected for use with the MLP. What follows is a brief account of why those parameters are important.

Through basic physics analysis, we know that at high velocities ($\approx 10\,{}^{\mathrm{km}}\!/\!{}_{\mathrm{s}}$) the dominate material property for the projectile and the target is density [6]. As the velocities drop down lower ($\approx 2\,{}^{\mathrm{km}}\!/\!{}_{\mathrm{s}}$) other material properties begin to become important (i.e. hardness, tensile strength, etc...). Obviously, since the importance of the material properties are dependent on velocity, the striking velocity of the projectile was a very important parameter. The total yaw of the projectile was an important parameter because it could be used to capture the drop off in penetration capability as the yaw increased beyond the theoretical critical yaw angle. The projectile length is important because it represents how much of the projectile is available for erosion during the penetration process. The length in conjunction with the diameter also provides information about the presented area of the projectile making contact with the target. The target thickness in conjunction with obliquity provides information on how much target material is available to resist the penetration of the projectile. The Young's Modulus provides information on the target's ability to resist compressive forces during the penetration process. The final list of parameters selected for use, because they have the most influence on the terminal ballistics results, are provided below:

- Striking Velocity
- Total Yaw
- Projectile Length
- Projectile Diameter
- Projectile Density
- Projectile Hardness
- Target Thickness
- Target Obliquity
- Target Hardness
- Target Density
- Target Young's Modulus

**RQ 1.2 – Can an ANN be used to produce a generalized, accurate model of the terminal ballistics of a KEP against monolithic metallic targets?**

The implementation of an MLP (a type of ANN) for prediction of terminal ballistics of a KEP against monolithic metallic targets is documented in Chapter 5. In summary, the MLP performed exceptionally well at predicting perforation, residual velocity, and residual mass against monolithic metallic targets. The input parameters in the training data set have a good spread over the values for where this model could be expected to be used. And the input parameters in the test data set also have a good spread over the values for where this model could be expected to be used, so the model has demonstrated that it is able to generalize over all of the 11 parameters. When the MLP was compared to the Segletes model (the current state-of-the-art model) it beat it in nearly every category. Here is a recap of the various statistics for the two models:

- Perforation prediction statistics
    - The MLP had a better Mean Squared Error (MSE) than the Segletes model, **0.8078** compared to 1.2205.
    - The MLP had a better accuracy than the Segletes model, **75.1%** compared to 62.3%.
    - The MLP had a better false positive rate than the Segletes model, **32.5%** compared to 64.9%.
    - The MLP had a better true negative rate than the Segletes model, **67.5%** compared to 35.1%.
    - The MLP had a better Matthews Correlation Coefficient (MCC) than the Segletes model, **0.48** compared to 0.19.
- Residual prediction statistics
    - The MLP had a better MSE than the Segletes model for residual velocity, **0.0726** compared to 0.2339.
    - The MLP had a better MSE than the Segletes model for residual mass, **0.0232** compared to 0.113.
    - The MLP had a better Symmetric Mean Absolute Percentage Error (SMAPE) than the Segletes model for residual velocity, **33.4%** compared to 47.9%.
    - The MLP had a better SMAPE than the Segletes model for residual mass, **35.6%** compared to 55.1%.

The MLP was clearly the better model than the Segletes model for use against monolithic metallic targets. The MLP is generalized and accurate, and can be used to effectively model the terminal ballistics of a KEP against monolithic metallic targets.

**RQ 1.3 – Can the ANN from RQ1.2 be used to model the terminal ballistics of a KEP against multi-element metallic targets?**

The application of the MLP from RQ 1.2 for prediction of terminal ballistics of a KEP against multi-element metallic targets is documented in Chapter 6. When the MLP was compared to the Segletes model (the current state-of-the-art model) for predictions against multiple element targets, it beat it in several categories. Here is a recap of the various statistics for the two models:

- Perforation prediction statistics
  - The MLP had a better MSE than the Segletes model, **1.343** compared to 1.3822.
  - The MLP had a better accuracy than the Segletes model, **58.5%** compared to 57.3%.
  - The MLP had a better false negative rate than the Segletes model, **17.1%** compared to 40.9%.
  - The MLP had a better true positive rate than the Segletes model, **82.9%** compared to 59.1%.
- Residual prediction statistics
  - The MLP had a better MSE than the Segletes model for residual mass, **0.0886** compared to 0.1226.
  - The MLP had a comparable SMAPE to the Segletes model for residual velocity, 51.8% compared to **49.5%**.
  - The MLP had a comparable SMAPE to the Segletes model for residual mass, 58.8% compared to **57.7%**.

In summary, the MLP performed as well as (and sometimes better than) the current state-of-the-art model at predicting perforation, residual velocity, and residual mass against multi-element metallic targets. Therefore, the MLP can be used to model the terminal ballistics of a KEP against multi-element metallic targets.

**RQ 1.4 – How does the speed (execution time and elapsed time) of an ANN based terminal ballistics model compare to the Segletes model?**

The average runtime of the MLP for monolithic targets was 0.06 ms and for the Segletes model the average runtime was 1.61 ms. The MLP takes on average 96% less time than the Segletes model to calculate the results of a single element target. This also means that the MLP is roughly 27 times faster than the Segletes model for single element targets. For multiple element targets the average runtime per element of the MLP was 0.49 ms and for the Segletes model it was 2.26 ms. The MLP takes on average 78% less time than the Segletes model to calculate the results of one element of a multiple element target. This also means that the MLP is roughly 5 times faster than the Segletes model for multiple element targets. The difference in runtime for monolithic targets means that the MLP finishes roughly 1.55 ms sooner than the Segletes model and for multiple element targets the MLP finished each element roughly 1.77 ms sooner.

An example Vulnerability/Lethality (V/L) analysis will be used to demonstrate the dramatic impact of the difference in runtimes. Imagine a main battle tank with rough dimensions of 2.4 m height, 3.7 m width, and 9.8 m length. Breaking the presented area of the side of that tank into cells that are 50 mm x 50 mm could result in a total of 7611 cells total. Doing the same for the front of the tank would result in roughly 2854 total cells. Those two sets of cells are called views; averaging them would give a rough approximation of how many cells to expect for any particular view. Assume that in each cell the simulation will make 10 sample runs and that each one of those runs may need to call the penetration model 10 times. The final assumption to be made is the number of views that will be run for the analysis; this assumption will be 109 total views. The azimuths for those view start at 0 ° and increment by 30 ° until 330 °. When an analysis is requested the requirements for the views can change based on the planned use for the results; here the assumption will be that the results will be used in a force on force model and some encounters could happen at higher elevation angles and in some cases even negative angles. For that type of requirement the following angles could be ran to meet those needs; start at -45 ° elevation and increment by 15 ° until 75 ° elevation. That results in 108 total views, and with the addition of a final view of 0 ° azimuth and 90 ° elevation, the total comes to 109 views. The average runtime for the MLP per element of a target was 0.142 ms and for the Segletes model it was 1.73 ms. A total of 109 views with 5232 cells per view, 10 samples per cell, and 10 model calls per sample amounts to 57 028 800 total calls per model for this sample scenario run. That amounts to 2.25 h total for the MLP and 27.47 h for the Segletes model. Those totals are for just one sample run of the simulation; many times an analyst will not catch an error in their inputs and the simulation can run for most of the required time before a problem is noted. This leads to an analyst rerunning the analysis multiple times so those runtime totals may be multiplied by any number of times for debugging purposes. In the end this can amount to a savings of days for an analyst.

The other time saving benefit of the MLP is that because it is generalized it does not require Subject Matter Expert (SME) fitting prior to an analyst using it in a V/L model run. The only part an SME may be involved in is checking that the model is predicting accurately for materials not seen before, which could lead to a retraining of the model at some point. Other models in use require ballistic SMEs to fit a new KEP threat to whatever material of is required (typically Rolled Homogeneous Armor (RHA) steel) and then for other materials the thicknesses are scaled based on density. The time required for SME work on input files for V/L analyses can vary from days to weeks, but a fair assumption is typically 2 to 3 weeks. That can add a significant amount of time to the completion of a V/L analysis.

In summary, the MLP execution time is extremely fast, performing 27 times faster than the Segletes model for single element targets and 5 times faster than the Segletes model for multiple element targets. The elapsed time for the MLP is also faster than the Segletes model because it requires less fitting by SMEs prior to use.

**RQ 1.5 – How does the accuracy of an ANN based terminal ballistics model compare to the Segletes model?**

For monolithic targets (SNF) the MLP far surpassed the predictive capabilities of the Segletes model for both the classification problem of determination of perforation and the regression problem of predicting the residual velocity and mass. For perforation prediction the MLP had better scores than the Segletes model for MSE, ACC, FPR, TNR, and MCC. For both residual velocity and residual mass the MLP had better MSE scores than the Segletes model. The MSE scores for the MLP were 0.0726 for $V_r$ and 0.0232 for $M_r$. For the Segletes model they were 0.2339 for $V_r$ and 0.113 for $M_r$. For both residual velocity and residual mass the MLP had better SMAPE scores than the Segletes model. The SMAPE scores for the MLP were 33.4% for $V_r$ and 35.6% for $M_r$. For the Segletes model they were 47.9% for $V_r$ and 55.1% for $M_r$. In test by test comparison the MLP was as good or better than the Segletes model for 68% of the tests for residual velocity and 76% of the tests for residual mass. The MLP was also first order stochastic dominate to the Segletes model for both residual velocity and mass.

The results for multiple element targets (MAF and MNF) were a little more ambiguous than those for single element targets. The MLP had a better MSE for perforation prediction and residual mass, but not residual velocity. The MLP had a better ACC score and TPR score, but not FPR, FNR, TNR, or MCC. The SMAPE scores for the MLP were nearly identical to that of the Segletes model. The SMAPE scores for the MLP were 51.8% for $V_r$ and 58.8% for $M_r$. For the Segletes model they were 49.5% for $V_r$ and 57.7% for $M_r$. For residual velocity, the MLP had more spread in the distribution of percent errors, but they were more balanced than the Segletes model. The Segletes model tended to under predict both perforations and residual estimates. The MLP was unbalanced in the prediction of perforation though, there were more false positives than false negatives. For residual mass, both models were skewed to negative percent errors (over predictions) and performed similarly.

Even though the Segletes model performed better in some areas (such as MSE scores) for multiple element targets, there were areas were the MLP also performed better (such as test-by-test comparison). Overall, the errors that the MLP had were comparable to the Segletes model and are acceptable because of the planned use for the MLP and the drastically better runtimes provided by using it.

In summary, the accuracy of the MLP is significantly superior to the Segletes model for single element targets and is similar to the Segletes model for multiple element targets.

## 7.2 Summary of Contributions

### 7.2.1 Publications

There were a total of four papers written and accepted for publication during the span of time of this research project:

- "Predicting the Terminal Ballistics of Kinetic Energy Projectiles Using Artificial Neural Networks", *Journal of Information Systems Applied Research*, Volume 7, Issue 1, February 2014. [97]

  The first publication was originally published and presented at the 2013 Conference on Information Systems Applied Research (CONISAR) and was selected for subsequent inclusion in the above journal. At the time of the conference, this research was in the first phase of the approach (see Section 1.5) and was focused on dealing with data. The key topic of the conference was "Big Data", so the conference presented a great opportunity to publish the current work related to data and to get good feedback on how to handle the data issues that were present at that time.

- "Predicting the Perforation of Kinetic Energy Projectiles using Artificial Neural Networks", *2014 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES 2014)*, Volume 1, 2014. [96]

  The second publication was written for the 2014 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES). At the time of that conference, the research was in the very beginning of the second phase of the approach (see Section 1.5) and there were many questions about how to properly set up the MLP. This conference presented a great opportunity to get feedback from experts in the field of computational intelligence, as well as present the current progress of the MLP implementation. A paper was also written for a doctoral consortium at that conference. Participation in the consortium proved valuable because some feedback from attendees led to important changes in the approach of topology selection.

- "Comparing the Prediction Capabilities of an Artificial Neural Network vs a Phenomenological Model for Predicting the Terminal Ballistics of Kinetic Energy Projectiles", *Proceedings of the Conference on Information Systems Applied Research (CONISAR)*, 2015. [98]

  The third paper was presented at the 2015 Conference on Information Systems Applied Research (CONISAR). That conference was selected because it posed an opportunity to present the results from comparing the MLP to the Segletes model and to get feedback from some of the same people that had seen the early research publication from that conference in 2013.

- "Comparing the Iterative Application of an Artificial Neural Network vs. a Phenomenological Model for Predicting the Terminal Ballistics of Kinetic Energy Projectiles Against Multiple Element Targets", Submitted for publication to *2017 IEEE Computing Conference.* [99]

  The fourth paper was submitted and accepted (acceptance notification received on October 3, 2016) to the 2017 IEEE Computing Conference. The conference is an international conference and has an "Intelligent Systems" track; the conference will provide a great venue to present the final results of this research.

In addition to the aforementioned publications, this research was also presented at two events. The first was at the Towson University Computer Sciences Department $30^{th}$ Anniversary celebration. The second was at the $21^{st}$ Annual United States Military Academy and U.S. Army Research Laboratory Symposium (AUTS). The presentation at the AUTS was a great opportunity to present this research to military and ballistics SMEs providing information about this research and receiving valuable feedback for experts in the field.

### 7.2.2 Impact

The work to collect and clean the terminal ballistics data for KEPs is of such importance that there are already opportunities being presented to receive funding to complete the cleaning and preparation of the QBasic database for inclusion in the Penetration Database Markup Language (PDML) database. The PDML database will be able to serve as a foundation for many other research projects in the U.S. ARL.

The results from this research can provide programs like MUVES-S2, Visual Simulation Lab (VSL), and System-of-Systems Survivability Simulation (S4) with an accurate and fast model for modeling KEPs. In particular, the MLP should allow VSL to model KEPs in real-time as a vehicle geometry is rotated on the screen. With the MLP VSL will be able to provide real-time information on vulnerability of a vehicle to KEPs. The benefit to MUVES-S2 will be faster overall runtimes, but the bigger impact will be in the end-to-end analysis time. There is a lot of time spent preparing inputs for KEP models and the use of the MLP will drastically reduce the time spent waiting on inputs from SMEs. All of those improvements enable the use of MUVES-S2 in larger force-on-force models such as S4, providing better vulnerability information to the decision makers of the U.S. Army which is critical to the survivability of our combat forces.

## 7.3 Discussion of Limitations

The MLP was only trained on SNF targets, which had air behind the only target element. So the effect of the plasticly deformed volume in the target reaching the back of the target element should be inherently modeled. This leads to a couple of limitations for the MLP. The first limitation is that the MLP does not know how the KEP will behave when impacting another target element after leaving an air gap. The second limitation is that the MLP also does not know how to handle two target elements that do not have an air gap between them. Those limitations were known while doing this research, but it was of interest to understand how the MLP would perform against those types of targets anyway. Given those limitations, the iterative approach still performed well.

The MLP is limited to use with KEPs that are static and solid. So specialized KEP designs such as telescoping and segmented Kinetic Energy (KE) rods were not tested and the performance of the MLP for those designs is unknown. As with any empirical model great care should be taken when attempting to use data that falls outside of the bounds of the data used for training. A lot of effort went into making sure the MLP did not overfit the data, but extrapolating outside of the bounds of the data could lead to unforeseen results. Nose shape was not used in the MLP in any other way than to add an approximate equivalent mass cylinder to the length of the KEPs. So any effects that would occur from interaction of the nose of the projectile and the target at higher obliquities is likely not captured. Pitch and yaw were not recorded in all of the experimental test data used so total yaw was used instead. That means that for any interactions that had pitch or yaw that would “turn” the projectile into the target at obliquity is also not captured. Any experimental tests that resulted in ricochet or severe shattering of the projectile were not included in the test data because there were not enough of them to properly represent the phenomena.

Even though the data used in this research was sufficient for answering the research questions, there were still gaps in the data. Another limitation of the MLP is the need for more data and data with a broader range of values. The MLP is highly dependent on good data and the database was lacking data for target materials other than steel and aluminum as well as for projectile materials other than steel, tungsten, and depleted uranium.

## 7.4 Future Work

As noted earlier, the MLP is limited by the fact that it was only trained on SNF data. More research can be done to improve the performance of the MLP for multiple element target data by leveraging the work by Gruss and Hirsch [39]. They have an approach that may allow for the MLP to be trained on SNF, MNF, and MAF data. In addition to that approach, future work may involve designing the MLP as a Recurrent Neural Network (RNN) so that later elements in the target can feedback into the calculation of earlier elements. Another potential design would allow for more input parameters, where the additional parameters would be the additional elements in the target. Work will need to be done to determine the best approach to ensure good fitment to the data.

Future work will need to be done to further develop the ballistics database to cover a broader scope of values for the MLP's parameters. By doing that the MLP will be able to generalize better over the full range of possible input values. The data for this research was unclassified, since it was being used for dissertation research. The results have shown that a MLP can be used to model KEPs, so further work can now be done at the U.S. ARL using classified test data

Many scientists and analysts like to be able to see an easy to understand equation when using models, so it is possible that a move from using an ANN to the use of symbolic regression using genetic programming may provide value in future research.

# Chapter 8

# References

[1] U.S. Department of Defense, "Operation of the Defense Acquisition System," Department of Defense Instruction DoDI 5000.02, U.S. Department of Defense, Washington, DC: Government Printing Office, December 2008.

[2] P. H. Deitz, H. L. Reed Jr., J. T. Klopcic, and J. N. Walbert, *Fundamentals Of Ground Combat System Ballistic Vulnerability/Lethality*. Progress In Astronautics And Aeronautics, AIAA (American Institute of Aeronautics & Astronautics), 2009.

[3] R. E. Ball, *The Fundamentals of Aircraft Combat Survivability Analysis and Design*. AIAA Education Series, AIAA (American Institute of Aeronautics & Astronautics), 2003.

[4] International Ballistics Society, "Ballistic Fields," 2012.

[5] J. A. Zook, K. Frank, and G. F. Silsby, "Terminal Ballistics Test and Analysis Guidelines for the Penetration Mechanics Branch," Memorandum Report BRL-MR-3960, Ballistic Research Laboratory, 1992.

[6] L. Magness, "Antiarmor Kinetic Energy Penetrators," in *Advanced Ballistics Science and Engineering* (B. Burns, ed.), no. ARL-SR-168, special report Lethality. Antiarmor Kinetic Energy Projectiles, pp. 386–416, U.S. Army Research Laboratory, November 2008.

[7] S. Price and J. Robertson, “Best business practices for conducting muves-s2 vulnerability/lethality (V/L) analyses,” Technical Report ARL-TR-5409, U.S. Army Research Laboratory, 2010.

[8] S. Moulsdale, “MUVES-S2 Study: Comparison of Vulnerability Results When Varying Grid Cells Size and Number of Views,” Technical Report ARL-TR-5835, U.S. Army Research Laboratory, June 2012.

[9] P. J. Hanes, S. L. Henry, G. S. Moss, K. R. Murray, and W. A. WInner, “Modular Unix-Based Vulnerability Estimation Suite (MUVES) Analyst’s Guide,” Memorandum Report BRL-MR-3954, Ballistic Research Laboratory, December 1991.

[10] R. Bernstein Jr., R. Flores, and M. Starks, “Objectives and capabilities of the system of systems survivability simulation (S4),” Technical Note ARL-TN-260, U.S Army Research Laboratory, 2006.

[11] Ballistics and NBC Division, Survivability/Lethality Analysis Directorate, U.S. Army Research Laboratory, “MUVES 3 Project Goals and Rationale.” Version 260210, March 2010.

[12] L. A. Butler, E. Kerzner, and C. Wyman, “High-Performance Vulnerability/Lethality Analysis,” Technical Report ARL-TR-6109, U.S.Army Research Laboratory, September 2012.

[13] J. R. Auten Sr., “A Comparison of Kinetic Energy Rod Algorithm Predictions to Test Data,” Technical Report ARL-TR-6192, U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, September 2012.

[14] S. B. Segletes, R. Grote, and J. Polesne, “Improving the Rod-Penetration Algorithm for Tomorrow’s Armors,” Reprint Report ARL-RP-23, U.S. Army Research Laboratory, June 2001.

[15] L. Tarassenko, *A Guide to Neural Computing Applications*. New York: Arnold, 1998.

[16] P. H. Deitz and A. Ozolins, “Computer Simulations of the Abrams Live-Fire Field Testing,” Memorandum Report BRL-MR-3755, Ballistic Research Laboratory, May 1989.

[17] R. E. Dibelka, "MUVES-S2 Configuration Management," June 2004.

[18] A. G. Williams, "The Search for High Velocity," *Guns Review International*, May 1996.

[19] W. Lanz and W. Odermatt, "Minimum Impact Energy For KE-Penetrators in RHA-Targets," in *Proceedings of the European Forum on Ballistics of Projectiles* (M. Giraud and V. Fleck, eds.), pp. 349–365, ISL, French-German Research Institute of Saint-Louis, April 2000.

[20] SBWiki, "M829," August 2012.

[21] W. Leonard and L. Magness, "The terminal ballistic performance of microstructural oriented tungsten heavy alloy penetrators," Technical Report ARL-TR-881, Army Research Laboratory, October 1995.

[22] W. Lanz and W. Odermatt, "Penetration Limits of Conventional Large Caliber Anti-tank Guns/Kinetic Energy Projectiles," in *Proceedings of the 13$^{th}$ International Symposium of Ballistics*, vol. 3, pp. 225–233, 1992.

[23] W. Odermatt, "Long rod penetrators - perforation equation," 2014.

[24] R. Jeanquartier and W. Odermatt, "Post-perforation length and velocity of ke projectiles with single oblique targets," in *15$^{th}$ International Symposium on Ballistics, Proceedings, Volume1*, pp. 245–252, 1995.

[25] W. A. Gooch, M. Burkins, H. J. Ernst, and T. Wolf, "Ballistic penetration of titanium alloy ti-6al-4v," in *Proceedings of the Lightweight Armour Systems Symposium*, 1995.

[26] R. Saucier, "Simple Line-of-Sight Subtraction Method for KE Penetrators: Using the Semi-Empirical Lanz-Odermatt Function," 2003. Army Research Laboratory. MUVES 3 Methodology Research Board. KELOS Algorithm Description.

[27] J. R. Auten Sr., "Algorithm Description for Segletes Hybrid Rod Model of Frank-Zook and Walker-Anderson." MUVES 3 Methodology Review Board Algorithm Description, March 2012.

[28] V. P. Alekseevskii, "Penetration of a Rod into a Target at High Velocity," *Comb., Expl., and Shock Waves*, vol. 2, pp. 63–66, 1966.

[29] A. Tate, "A Theory for the Deceleration of Long Rods after Impact," *Journal of Mechanics and Physics of Solids*, vol. 15, pp. 387–399, 1967.

[30] K. Frank and J. Zook, "Energy-Efficient Penetration of Targets," Memorandum Report BRL-MR-3885, U.S. Ballistics Research Laboratory, February 1991.

[31] J. D. Yatteau, R. H. Zernow, G. W. Recht, K. T. Edquist, and D. L. Dicksinson, "FATEPEN, a model to predict terminal ballistic penetration and damage to military targets," tech. rep., 1999.

[32] J. D. Yatteau, R. H. Zernow, G. W. Recht, and D. L. Dickinson, "FATEPEN, a model to predict terminal ballistic penetration and damage to military targets model improvements 1999-2005,"

[33] J. D. Walker and C. E. Anderson Jr., "A Time-Dependent Model for Long-Rod Penetration," *International Journal of Solids and Structures*, vol. 16, no. 1, pp. 19–48, 1995.

[34] S. B. Segletes, "An Adaptation of Walker-Anderson Model Elements Into the Frank-Zook Penetration Model for Use in MUVES," Technical Report ARL-TR-2336, U.S. Army Research Laboratory, September 2000.

[35] T. W. Bjerke, G. F. Silsby, D. R. Scheffler, and R. M. Mudd, "Yawed Long-Rod Armor Penetration," *International Journal of Impact Engineering*, vol. 12, no. 2, pp. 281–292, 1992.

[36] W. A. Gooch and M. Burkins, "A Ballistic Evaluation of Ti-6Al-4V vs. Long Rod Penetrators," in *RTO Meeting Proceedings 69(II); Low Cost Composite Structures and Cost Effective Application of Titanium Alloys in Military Platforms*, May 2001. NATO RTO-MP-069 (II).

[37] W. S. McCulloch and W. Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity," *Bulletin of Mathematical Biology*, vol. 5, pp. 115–133, 1943.

[38] A. Canas, E. Ortigosa, E. Ros, and P. Ortigosa, "FPGA Implementation of a Fully and Partially Connected MLP," in *FPGA Implementations of Neural Networks* (A. R. Omondi and J. C. Rajapakse, eds.), pp. 271–296, Springer US, 2006.

[39] E. Gruss and E. Hirsch, "Approximating the Ballistic Penetration Function of a Jet in a Multi-Cassette Target by the use of Neural Networks," in *Proceedings 19$^{th}$ International Symposium of Ballistics 7–11 May 2001*, pp. 1069–1075, May 2001.

[40] I. Gonzalez-Carrasco, A. Garcia-Crespo, B. Ruiz-Mezcua, and J. L. Lopez-Cuadrado, "Dealing with limited data in ballistic impact scenarios: an empirical comparison of different neural network approaches," *Applied Intelligence*, vol. 35, no. 1, pp. 89–109, 2011.

[41] I. Gonzalez-Carrasco, A. Garcia-Crespo, B. Ruiz-Mezcua, and J. L. Lopez-Cuadrado, "Neural Network Application for High Speed Impacts Classification," in *Proceedings of the World Congress on Engineering*, vol. 1, 2008.

[42] L.-M. Fu, *Neural Networks in Computer Intelligence.* McGraw-Hill Education, 2003.

[43] S. Walczak and N. Cerpa, "Heuristic principles for the design of artificial neural networks," *Information and software technology*, vol. 41, no. 2, pp. 107–117, 1999.

[44] J. E. Dayhoff, *Neural Network Architectures.* International Thomson Computer Press, 1996.

[45] D. Fernández-Fdz and R. Zaera, "A new tool based on artificial neural networks for the design of lightweight ceramic-metal armour against high-velocity impact of solids," *International Journal of Solids and Structures*, vol. 45, no. 25-26, pp. 6369–6383, 2008.

[46] C. M. Bishop, *Neural Networks for Pattern Recognition.* New York, NY, USA: Oxford University Press, Inc., 1995.

[47] S. Lawrence, C. L. Giles, and S. Fong, "Natural language grammatical inference with recurrent neural networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 1, pp. 126–140, 2000.

[48] K. Swingler, *Applying Neural Networks: A Practical Guide.* Academic Press, 1996.

[49] W. S. Sarle, “Neural Network FAQ,” 2002. Compilation of Usenet newsgroup comp.ai.neural-nets.

[50] A. Garcia-Crespo, B. Ruiz-Mezcua, D. Fernandez-Fdz, and R. Zaera, “Prediction of the response under impact of steel armours using a multilayer perceptron,” *Neural Computing & Applications*, vol. 16, no. 2, pp. 147–154, 2007.

[51] Y. LeCun, L. Bottou, G. Orr, and K. Müller, “Efficient Backprop,” in *Neural Networks—Tricks of the Trade* (G. Orr and K. Mller, eds.), vol. 1524 of *Lecture Notes in Computer Science*, pp. 5–50, Springer, 1998.

[52] J. A. Snyman, *Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms*, vol. 97. Springer, 2005.

[53] S. Danaher, S. Datta, I. Waddle, and P. Hackney, “Erosion modelling using Bayesian regulated artificial neural networks,” *Wear*, vol. 256, no. 9-10, pp. 879–888, 2004.

[54] M. McInerney and A. P. Dhawan, “Use of genetic algorithms with backpropagation in training of feedforward neural networks,” in *IEEE International Conference on Neural Networks*, vol. 1, pp. 203–208, IEEE, 1993.

[55] R. Kapoor, D. Pal, and J. Chakravartty, “Use of artificial neural networks to predict the deformation behavior of Zr-2.5 Nb-0.5 Cu,” *Journal of Materials Processing Technology*, vol. 169, no. 2, pp. 199–205, 2005.

[56] P. Koehn, “Combining genetic algorithms and neural networks: The encoding problem,” Master’s thesis, The University of Tennesse, Knoxville, 1994.

[57] R. A. Wildman and D. S. Weile, “Greedy search and a hybrid local optimization/genetic algorithm for tree-based inverse scattering,” *Microwave and optical Technology Letters*, vol. 50, pp. 822–825, March 2008.

[58] T. Weise, *Global Optimization Algorithms – Theory and Application.* it-weise.de (self-published): Germany, 2009.

[59] P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-Validation," in *Encyclopedia of Database Systems* (L. LIU and M. ZSU, eds.), pp. 532–538, Springer US, 2009.

[60] G. Zhang, M. Y Hu, B. Eddy Patuwo, and D. C Indro, "Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis," *European journal of operational research*, vol. 116, no. 1, pp. 16–32, 1999.

[61] G. B. Ramaiah, R. Y. Chennaiah, and G. K. Satyanarayanarao, "Investigation and modeling on protective textiles using artificial neural networks for defense applications," *Materials Science and Engineering: B*, vol. 168, no. 1-3, pp. 100–105, 2010.

[62] D. Fernández-Fdz, J. López-Puente, and R. Zaera, "Prediction of the behaviour of CFRPs against high-velocity impact of solids employing an artificial neural network methodology," *Composites Part A: Applied Science and Manufacturing*, vol. 39, no. 6, pp. 989–996, 2008.

[63] I. Gonzalez-Carrasco, A. Garcia-Crespo, B. Ruiz-Mezcua, and J. L. Lopez-Cuadrado, "A neural network-based methodology for the recreation of high-speed impacts on metal armours," *Neural Computing & Applications*, vol. 21, pp. 91–107, 2012.

[64] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," *Journal of Machine Learning Research*, vol. 10, no. Jan, pp. 1–40, 2009.

[65] Y. Bengio, "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[66] T. Taheri, "Benchmarking and Comparing Encog, Neuroph and JOONE Neural Networks," June 2010.

[67] J. Heaton, "Encog machine learning framework," 2016.

[68] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, “Duplicate record detection: A survey,” *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 1, pp. 1–16, 2007.

[69] R. Parekh, J. Yang, and V. Honavar, “Constructive neural-network learning algorithms for pattern classification,” *Neural Networks, IEEE Transactions on*, vol. 11, no. 2, pp. 436–451, 2000.

[70] M. A. Costa, A. P. Braga, and B. R. de Menezes, “Improving neural networks generalization with new constructive and pruning methods,” *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 13, no. 2-4, pp. 75–83, 2002.

[71] L. Prechelt, “Early stopping - But when?,” in *Neural Networks: Tricks of the trade*, pp. 53–67, Springer, 2012.

[72] N. Mohamad, F. Zaini, A. Johari, I. Yassin, and A. Zabidi, “Comparison between levenberg-marquardt and scaled conjugate gradient training algorithms for breast cancer diagnosis using mlp,” in *Signal Processing and Its Applications (CSPA), 2010 6th International Colloquium on*, pp. 1–7, IEEE, 2010.

[73] S. Perez, “Apply genetic algorithm to the learning phase of a neural network,” 2008.

[74] J. M. Hellerstein, “Quantitative Data Cleaning for Large Databases,” *United Nations Economic Commission for Europe (UNECE)*, 2008.

[75] D. Collins, “Data quality: Types of error in database data,” 2008. Accessed 23 February 2015.

[76] P. Oliveira, F. Rodrigues, and H. Pedro, “A formal definition of data quality problems,” in *Proceedings of International Conference on Information Quality*, 2005.

[77] P. Oliveira, F. Rodrigues, and H. Pedro, “A taxonomy of data quality problems,” in *Proceedings of 2nd International Workshop on Data and Information Quality*, 2005.

[78] J. Wright and M. Manic, “Neural network architecture selection analysis with application to cryptography location,” in *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pp. 1–6, IEEE, July 2010.

[79] Y. Liu, J. Cheng, C. Yan, X. Wu, and F. Chen, “Research on the matthews correlation coefficients metrics of personalized recommendation algorithm evaluation,” *International Journal of Hybrid Information Technology*, vol. 8, no. 1, pp. 163–172, 2015.

[80] D. M. W. Powers, “Evaluation: From precision, recall and f-factor to roc, informedness, markedness & correlation,” *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

[81] D. A. Swanson, J. Tayman, and T. Bryan, “Mape-r: a rescaled measure of accuracy for cross-sectional subnational population forecasts,” *Journal of Population Research*, vol. 28, no. 2-3, pp. 225–243, 2011.

[82] S. Kolassa and R. Martin, “Percentage errors can ruin your day (and rolling the dice shows how).,” *Foresight: The International Journal of Applied Forecasting*, no. 23, 2011.

[83] C. Tofallis, “A better measure of relative prediction accuracy for model selection and model estimation,” *Journal of the Operational Research Society*, vol. 66, no. 8, pp. 1352–1362, 2015.

[84] T. Foss, E. Stensrud, B. Kitchenham, and I. Myrtveit, “A simulation study of the model evaluation criterion mmre,” *IEEE Transactions on Software Engineering*, vol. 29, no. 11, pp. 985–995, 2003.

[85] B. P. Helms and W. H. Jean, “An algorithm for $N$th degree stochastic dominance,” vol. 2, pp. 71 – 81, 1986.

[86] R. Davidson, *Stochastic Dominance.* 2nd ed., 2006.

[87] C. E. Anderson Jr., B. L. Morris, and D. L. Littlefield, “A Penetration Mechanics Database,” Technical Report SwRI Report 3593/001, Southwest Research Institute, 1992.

[88] T. G. Farrand, "Terminal Ballistic Evaluation of a Candidate 0.60-Caliber Electromagnetically Launched Long-Rod Penetrator," Memorandum Report ARL-MR-104, U.S. Army Research Laboratory, September 1993.

[89] S. B. Segletes, "The Erosion Transition of Tungsten-Alloy Long Rods into Aluminum Targets," *International Journal of Solids and Structures*, vol. 44, pp. 2168–2191, 2007.

[90] M. Burkins, J. Paige, and J. Hansen, "A Ballistic Evaluation of Ti-6Al-4V vs. Long Rod Penetrators," Technical Report ARL-TR-1146, U.S. Army Research Laboratory, 1996.

[91] J. A. Zook and K. Frank, "Comparative Penetration Performance of Tungsten Alloy L/D=10 Long Rods with Different Nose Shapes Fired at Rolled Homogeneous Armor," Memorandum Report BRL-MR-3480, U.S. Army Ballistic Research Laboratory, November 1985.

[92] Z. Rosenberg and M. J. Forrestal, "Perforation of Aluminum Plates with Conical-Nosed Rods - Additional Data and Discussion," *Journal of Applied Mechanics*, vol. 55, pp. 236–238, 1988.

[93] M. Burkins, W. A. Gooch, and D. Weeks, "Performance of a 30-mm armor piercing discarding sabot (apds) munition into steel and aluminum alloy armors," Memorandum Report BRL-MR-385, Army Research Laboratory, 1998.

[94] J. I. Maletic and A. Marcus, "Data Cleansing - A Prelude to Knowledge Discovery," in *The Data Mining and Knowledge Discovery Handbook* (O. Maimon and L. Rokach, eds.), pp. 21–36, Springer, 2005.

[95] J. W. Graham, *Missing data: Analysis and design*, ch. Chapter 2: Analysis of Missing Data, pp. 47–69. Springer, 2012.

[96] J. R. Auten Sr. and R. J. Hammell II, "Predicting the perforation of kinetic energy projectiles using artificial neural networks," in *2014 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES 2014)*, vol. 1, pp. 132–139, 2014.

[97] J. R. Auten Sr. and R. J. Hammell II, “Predicting the terminal ballistics of kinetic energy projectiles using artificial neural networks,” *Journal of Information Systems Applied Research*, vol. 7, pp. 23–32, February 2014.

[98] J. R. Auten Sr. and R. J. Hammell II, “Comparing the prediction capabilities of an artificial neural network vs a phenomenological model for predicting the terminal ballistics of kinetic energy projectiles,” in *Proceedings of the Conference on Information Systems Applied Research (CONISAR)*, 2015.

[99] J. R. Auten Sr. and R. J. Hammell II, “Comparing the iterative application of an artificial neural network vs. a phenomenological model for predicting the terminal ballistics of kinetic energy projectiles against multiple element targets,” in *Proceedings of the 2017 IEEE Computing Conference*, 2017. Accepted.

# Chapter 9

# Bibliography

[1] *Neural Networks Library In Java*, 5th ed.

[2] "Department of defense handbook design of projectiles for terminal ballistic effects," U.S. DEPARTMENT OF DEFENSE, Handbook MIL-HDBK-1226(AR), May 1997.

[3] *Materials Data Book.* Cambri, 2003.

[4] "Armor plate, steel, wrought, homogeneous (for use in combat-vehicles and for ammunition testing)," U.S. DEPARTMENT OF THE ARMY, Specification MIL-DTL-12560J(MR), July 2009.

[5] M. Abdella and T. Marwala, "The use of genetic algorithms and neural networks to approximate missing data in database," *Computing and Informatics*, vol. 24, pp. 577–589, 2005.

[6] S. Abdullah, J. G. Hetherington, and D. W. Leeming, "Penetration Performance of Segmented Rods – Comparison with Continuous Rods at High Velocity," *Journal of Battlefield Technology*, vol. 1, no. 1, pp. 4–8, March 1998.

[7] A. C. Acock, "Working With Missing Values," *Journal of Marriage and Family*, vol. 67, pp. 1012–1028, November 2005.

[8] S. Adam, D. A. Karras, and M. N. Vrahatis, “Revisiting the problem of weight initialization for multi-layer perceptrons trained with back propagation,” in *International Conference on Neural Information Processing.* Springer, 2008, pp. 308–315.

[9] B. Adams, “Simulation of ballistic impacts on armored civil vehicles,” *Eindhoven University of Technology*, 2006.

[10] P. Adriaans and D. Zantinge, *Data Mining*, illustrated, reprint ed. Addison-Wesley, 1996.

[11] V. P. Alekseevskii, “Penetration of a Rod into a Target at High Velocity,” *Comb., Expl., and Shock Waves*, vol. 2, pp. 63–66, 1966.

[12] Y. Ali and A. Riad, “Perforation of thick plates by high-speed projectiles,” in *13th International Conference on AEROSPACE SCIENCES & AVIATION TECHNOLOGY, ASAT*, vol. 13, 2009.

[13] W. A. Allen and J. W. Rogers, “Penetration of a Rod into Semi-Infinite Target.” *Journal of the Franklin Institute*, vol. 272, pp. 272–284, 1961.

[14] P. D. Allison, “Missing Data Techniques for Structural Equation Modeling,” *Journal of Abnormal Psychology*, vol. 112, no. 4, pp. 545–557, 2003.

[15] ——, “Handling Missing Data by Maximum Likelihood,” in *Proceedings of the SAS Global Forum 2012 Conference*, no. 312-2012. SAS Institute Inc., 2012.

[16] U. Anders and O. Korn, “Model selection in neural networks,” *Neural Networks*, vol. 12, pp. 309 – 323, 1999.

[17] T. Andersen and T. Martinez, “Cross validation and MLp architecture selection,” in *Neural Networks, 1999. IJCNN’99. International Joint Conference on*, vol. 3. IEEE, 1999, pp. 1614–1619.

[18] T. Andersen, M. Rimer, and T. Martinez, “Optimal artificial neural network architecture selection for bagging,” in *Neural Networks, 2001. Proceedings. IJCNN’01. International Joint Conference on*, vol. 2. IEEE, 2001, pp. 790–795.

[19] R. L. Anderson, “Fracture of grooved kinetic energy rods subject to oblique impact,” Army Research Laboratory, Technical Report ARL-TR-3923, 2006.

[20] C. E. Anderson Jr, T. Behner, and V. Hohler, “Penetration efficiency as a function of target obliquity and projectile pitch,” *Journal of Applied Mechanics*, vol. 80, no. 3, pp. 031 801–1–031 801–11, 2013.

[21] C. E. Anderson Jr., T. Behner, T. J. Holmquist, and D. L. Orphal, “Penetration Response of Silicon Carbide as a Function of Impact Velocity,” *International Journal of Impact Engineering*, vol. 38, pp. 892–899, 2011.

[22] C. E. Anderson Jr., S. Chocron, and R. P. Bigger, “Time-Resolved Penetration into Glass: Experiments and Computations,” *International Journal of Impact Engineering*, vol. 38, pp. 723–731, 2011.

[23] C. E. Anderson Jr. and et al., “Long Rod Penetration, Target Resistance, and Hypervelocity Impacts,” in *Proceedings of Hypervelocity Impact Symposium (HVIS Symp.)*, 1992.

[24] C. E. Anderson Jr., B. L. Morris, and D. L. Littlefield, “A Penetration Mechanics Database,” Southwest Research Institute, Technical Report SwRI Report 3593/001, 1992.

[25] C. E. Anderson Jr., S. A. Mullin, A. J. Piekutowski, N. W. Blaylock, and K. L. Poormon, “Scale Model Experiments with Ceramic Laminte Targets,” *International Journal of Impact Engineering*, vol. 18, no. 1, pp. 1–22, 1996.

[26] C. E. Anderson Jr. and D. L. Orphal, “An Examination of Deviations from Hydrodynamic Penetration Theory,” *International Journal of Impact Engineering*, vol. 35, pp. 1386–1392, 2008.

[27] C. E. Anderson Jr., D. L. Orphal, T. Behner, and D. W. Templeton, “Failure and Penetration Response of Borosilicate Glass During Short-Rod Impact,” *International Journal of Impact Engineering*, vol. 36, pp. 789–798, 2009.

[28] C. E. Anderson Jr. and S. A. Royal-Timmons, “Ballistic Performance of Confined $99.5\% - Al_2O_3$ Ceramic Tiles,” *International Journal of Impact Engineering*, vol. 19, no. 8, pp. 703–713, 1997.

[29] C. E. Anderson Jr., R. Subramanian, J. D. Walker, M. Normandia, and T. R. Sharron, “Penetration Mechanics of Seg-Tel Penetrators,” *International Journal of Impact Engineering*, vol. 20, pp. 13–26, 1997.

[30] C. E. Anderson Jr., J. D. Walker, S. Bless, and Y. Partom, “On the L/D Effect for Long-Rod Penetrators,” *International Journal of Impact Engineering*, vol. 18, no. 3, pp. 247–264, 1996.

[31] C. E. Anderson Jr., J. D. Walker, S. J. Bless, and T. R. Sharron, “On the Velocity Dependence of the L/D Effect For Long-Rod Penetrators,” *International Journal of Impact Engineering*, vol. 17, pp. 13–24, 1995.

[32] C. E. Anderson Jr., J. D. Walker, and G. E. Hauver, “Target Resistance for Long-Rod Penetration into Semi-Infinite Targets,” *Nuclear Engineering and Design*, vol. 138, pp. 93–104, 1992.

[33] C. E. Anderson Jr., V. Hohler, J. D. Walker, and A. J. Stilp, “Time-Resolved Penetration of Long Rods Into Steel Targets,” *International Journal of Impact Engineering*, vol. 16, no. 1, pp. 1–18, 1995.

[34] ——, “The Influence of Projectile Hardness on Ballistic Performance,” *International Journal of Impact Engineering*, vol. 22, pp. 619–632, 1999.

[35] C. E. Anderson Jr. and D. L. Littlefield, “Pretest Predictions of Long-Rod Interactions with Armor Technology Targets,” Southwest Research Institute, Technical Report, April 1994.

[36] C. E. Anderson Jr, D. L. Littlefield, and J. D. Walker, “Long-rod penetration, target resistance, and hypervelocity impact,” *International Journal of Impact Engineering*, vol. 14, no. 1, pp. 1–12, 1993.

[37] C. E. Anderson Jr., D. L. Orphal, R. R. Franzen, and J. D. Walker, "On the Hydrodynamic Approximation for Long-Rod Penetration," *International Journal of Impact Engineering*, vol. 22, pp. 23–43, 1999.

[38] C. E. Anderson Jr, T. Sharron, J. D. Walker, and C. J. Freitas, "Simulation and analysis of a 23-mm hei projectile hydrodynamic ram experiment," *International journal of impact engineering*, vol. 22, no. 9, pp. 981–997, 1999.

[39] C. E. Anderson Jr and J. D. Walker, "An examination of long-rod penetration," *International Journal of Impact Engineering*, vol. 11, no. 4, pp. 481–501, 1991.

[40] ——, "An analytical model for dwell and interface defeat," *International Journal of Impact Engineering*, vol. 31, no. 9, pp. 1119–1132, 2005.

[41] A. V. Andrade, L. d. Errico, A. L. L. d. Aquino, L. P. d. Assis, and C. H. N. d. R. Barbosa, "Analysis of selection and crossover methods used by genetic algorithm-based heuristic to solve the lsp allocation problem in mpls networks under capacity constraints," in *Proceedings of 2008 International Conference on Engineering Optimization*, 2008.

[42] D. Anguita, A. Ghio, S. Ridella, and D. Sterpi, "K-Fold Cross Validation for Error Rate Estimate in Support Vector Machines." in *DMIN*, R. Stahlbock, S. F. Crone, and S. Lessmann, Eds. CSREA Press, 2009, pp. 291–297. [Online]. Available: http://dblp.uni-trier.de/db/conf/dmin/dmin2009.html#AnguitaGRS09

[43] J. E. Angus, "Criteria for Choosing the Best Neural Network: Part I," Naval Health Research Center, Report NHRC R 91-16, September 1991.

[44] M. Arad, D. Touati, I. Latovitz, P. IMI, and M. Arad, "The relation between initial yaw and long rod projectile shape after penetrating an oblique thin plate," in *$4^{th}$ European LSDYNA Users Conference*.

[45] O. Aran, O. T. Yildiz, and E. Alpaydin, “An incremental framework based on cross-validation for estimating the architecture of a multilayer perceptron,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 02, pp. 159–190, 2009.

[46] S. Arlot, A. Celisse *et al.*, “A survey of cross-validation procedures for model selection,” *Statistics surveys*, vol. 4, pp. 40–79, 2010.

[47] “Standard Test Method for Brinell Hardness of Metallic Materials,” ASTM International, West Conshohocken, PA, Standard, 2001.

[48] A. Atiya, “Learning algorithms for neural networks,” Ph.D. dissertation, Caltech, 1991. [Online]. Available: http://resolver.caltech.edu/CaltechETD:etd-09232005-083502

[49] P. Auerkari, *Mechanical and physical properties of engineering alumina ceramics.* Technical Research Centre of Finland Finland, 1996.

[50] J. R. Auten Sr. and R. J. Hammell II, “Predicting the perforation of kinetic energy projectiles using artificial neural networks,” in *2014 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES 2014)*, vol. 1, 2014, pp. 132–139.

[51] ——, “Comparing the iterative application of an artificial neural network vs. a phenomenological model for predicting the terminal ballistics of kinetic energy projectiles against multiple element targets,” in *Proceedings of the 2017 IEEE Computing Conference*, 2017, accepted.

[52] J. R. Auten Sr., “A Comparison of Penetration Algorithms: Predictions vs. Test Data for Kinetic Energy Rods,” in *Proceedings of the 26$^{th}$ International Symposium of Ballistics*, E. Baker and D. Templeton, Eds., vol. 2. Lancaster, PA.: DEStech Publications, September 2011, pp. 1522–1532.

[53] ——, “A Comparison of Kinetic Energy Rod Algorithm Predictions to Test Data,” U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, Technical Report ARL-TR-6192, September 2012.

[54] ——, “Algorithm Description for Segletes Hybrid Rod Model of Frank-Zook and Walker-Anderson,” March 2012, MUVES 3 Methodology Review Board Algorithm Description.

[55] ——, “Kinetic energy rod analysis tool user’s guide (version 1.0.0),” U.S. Army Research Laboratory, Technical Report ARL-TR-6886, April 2014.

[56] J. R. Auten Sr. and R. J. Hammell II, “Predicting the terminal ballistics of kinetic energy projectiles using artificial neural networks,” *Journal of Information Systems Applied Research*, vol. 7, no. 1, pp. 23–32, February 2014.

[57] ——, “Comparing the prediction capabilities of an artificial neural network vs a phenomenological model for predicting the terminal ballistics of kinetic energy projectiles,” in *Proceedings of the Conference on Information Systems Applied Research (CONISAR)*, 2015.

[58] V. V. Ayuyev, J. Jupin, P. W. Harris, and Z. Obradovic, “Dynamic Clustering-Based Estimation of Missing Values in Mixed Type Data,” in *Proceedings of the 11$^{th}$ International Conference on Data Warehousing and Knowledge Discovery*, ser. DaWaK ’09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 366–377.

[59] A. D. Back and T. P. Trappenberg, “Selecting inputs for modeling using normalized higher order statistics and independent component analysis,” *IEEE Transactions on Neural Networks*, vol. 12, no. 3, pp. 612–617, 2001.

[60] J. R. Baker, “Rod Lethality Studies,” Naval Research Laboratory, Report NRL R 6920, 1969.

[61] J. R. Baker and A. Williams, “Hypervelocity Penetration of Plate Targets by Rod and Rod-Like Projectiles,” *International Journal of Impact Engineering*, vol. 5, pp. 101–110, 1987.

[62] W. E. Baker and P. S. Westine, “Model Analysis for Penetration of Spaced Armor,” Ballistic Research Laboratory, Aberdeen Proving Ground, MD, Contract Report BRL-CR-327, January 1977.

[63] W. E. Baker, R. Saucier, T. M. Muehl, and R. L. Grote, “Comparison of MUVES-SQuASH with Bradley Fighting Vehicle Live-Fire Test Results,” U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, Technical Report ARL-TR-1846, November 1998.

[64] W. E. Baker, J. H. Smith, and W. A. WInner, “Vulnerability/Lethality Modeling of Armored Combat Vehicles Status and Recommendations,” U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, Technical Report ARL-TR-42, February 1993.

[65] R. E. Ball, *The Fundamentals of Aircraft Combat Survivability Analysis and Design*, ser. AIAA Education Series, J. A. Schetz, Ed. AIAA (American Institute of Aeronautics & Astronautics), 2003.

[66] Ballistics and NBC Division, Survivability/Lethality Analysis Directorate, U.S. Army Research Laboratory, “MUVES 3 Project Goals and Rationale,” March 2010, Version 260210.

[67] S. D. Bartus, “Evaluation of Titanium-5Al-5Mo-5V-3Cr (Ti-5553) Alloy Against Fragment and Armor-Piercing Projectiles,” U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, Technical Report ARL-TR-4996, September 2009.

[68] O. H. Basquin, *Tangent modulus and the strength of steel columns in tests.* Govt. Print. Off., 1924, no. 263.

[69] T. Behner, C. E. Anderson Jr., D. L. Orphal, V. Hohler, M. Moll, and D. W. Templeton, “Penetration and Failure of Lead and Borosilicate Glass Against Rod Impact,” *International Journal of Impact Engineering*, vol. 35, pp. 447–456, 2008.

[70] T. Behner, D. L. Orphal, V. Hohler, C. E. Anderson Jr., R. L. Mason, and D. W. Templeton, “Hypervelocity Penetration of Gold Rods into SiC-N for Impact Velocities from 2.0 to 6.2 km/s,” *International Journal of Impact Engineering*, vol. 33, pp. 68–79, 2006.

[71] T. Behner, C. E. Anderson Jr., T. J. Holmquist, D. L. Orphal, M. Wickert, and D. W. Templeton, “Penetration Dynamics and Interface Defeat Capability of Silicon Carbide Against Long Rod Impact,” *International Journal of Impact Engineering*, vol. 38, pp. 419–425, 2011.

[72] J. T. Behrens, “Principles and Procedures of Exploratory Data Analysis,” *Psycollogical Methods*, vol. 2, no. 2, pp. 1131–160, 1997.

[73] C. Beleites, U. Neugebauer, T. Bocklitz, C. Krafft, and J. Popp, “Sample size planning for classification models,” *Analytica Chimica Acta*, vol. 760, pp. 25–33, 2013.

[74] Y. Bengio, “Learning deep architectures for ai,” *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[75] M. D. Bennett, “Long Rod Penetrator Performance,” *Journal of Battlefield Technology*, vol. 1, no. 3, pp. 1–6, November 1998.

[76] ——, “Long rod penetrator performance,” *Journal of Battlefield Technology*, vol. 1, no. 3, November 1998.

[77] S. Berlemont, N. Burrus, D. Lesage, F. Maes, J.-B. Mouret, B. Perrot, M. Rey, N. Tisserand, and A. Wang, “Neural Networks: Multi-Layer Perceptron and Hopfield Network,” March 2012. [Online]. Available: http://www.lrde.epita.fr/~david/nn.pdf

[78] R. S. Bernard and D. C. Creighton, “Projectile penetration in soil and rock: analysis for non-normal impact,” DTIC Document, Tech. Rep., 1979.

[79] R. Bernstein Jr., R. Flores, and M. Starks, “Objectives and capabilities of the system of systems survivability simulation (S4),” U.S Army Research Laboratory, Technical Note ARL-TN-260, 2006.

[80] J. Bertini, M. do Carmo Nicoletti, and L. Zhao, “Imputation of missing data supported by complete p-partite attribute-based decision graphs,” in *Neural Networks (IJCNN), 2014 International Joint Conference on.* IEEE, 2014, pp. 1100–1106.

[81] H. Billon, "A model for ballistic impact on soft armour," Aeronautical and Maritime Research Laboratory, Technical Report DSTO-TR-0730, 1998.

[82] P. Bilski, "Data set preprocessing methods for the artificial intelligence-based diagnostic module." *Measurement*, vol. 54, pp. 180 – 190, 2014. [Online]. Available: http://proxy-tu.researchport.umd.edu/login?ins=tu&url=http://search.ebscohost.com.proxy-tu.researchport.umd.edu/login.aspx?direct=true&db=edselp&AN=S0263224114001274&site=eds-live&scope=site

[83] C. M. Bishop and C. M. Roach, "Fast Curve Fitting Using Neural Networks," *Review of Scientific Instruments*, vol. 63, no. 10, pp. 4450–4456, October 1992.

[84] C. M. Bishop, *Neural Networks for Pattern Recognition.* New York, NY, USA: Oxford University Press, Inc., 1995.

[85] M. Bisi and N. Kumar Goyal, "Software reliability prediction using neural network with encoded input," *International Journal of Computer Applications*, vol. 47, no. 22, pp. 46–52, 2012.

[86] T. W. Bjerke, G. F. Silsby, D. M. Scheffler, and R. M. Mudd, "Yawed Long Rod Armor Penetration at Ordance and Higher Velocities," Ballistic Research Laboratory, Technical Report BRL-TR-3221, March 1991.

[87] T. W. Bjerke, G. F. Silsby, D. R. Scheffler, and R. M. Mudd, "Yawed Long-Rod Armor Penetration," *International Journal of Impact Engineering*, vol. 12, no. 2, pp. 281–292, 1992.

[88] T. W. Bjerke, J. A. Zukas, and K. D. Kimsey, "Penetration Performance of Tungsten Alloy Penetrators with L/D Ratios of 1 to 1/32," Ballistic Research Laboratory, Technical Report BRL-TR-3246, June 1991.

[89] S. Bless, B. A. Pedersen, J. M. Campos, M. Normandia, and R. Subramanian, "Penetration of Oblique Plates," Institute for Advanced Technology (IAT), Technical Report IAT.R 0149, December 1997.

[90] S. Bless, Z. Rosenberg, and B. Yoon, “Hypervelocity Penetration of Ceramics,” *International Journal of Impact Engineering*, vol. 5, no. 1–4, pp. 165–171, 1987.

[91] S. J. Bless, “Interior and Terminal Ballistics of 25g Long Rod Penetrators,” Ballistic Research Laboratory, Contract Report ARBRL-CR-00398, April 1979.

[92] T. Blickle and L. Thiele, “A Comparison of Selection Schemes used in Genetic Algorithms,” Gloriastrasse 35, CH-8092 Zurich: Swiss Federal Institute of Technology (ETH) Zurich, Computer Engineering and Communications Networks Lab (TIK, Tech. Rep., 1995.

[93] V. Boljanovic, *Applied Mathematical & Physical Formulas*, J. Carleo, Ed. Industrial Press, 2007.

[94] B. M. Bolker, *Ecological Models and Data in R*. Princeton University Press, 2011.

[95] S. Borman, “The expectation maximization algorithm-a short tutorial,” January 2009. [Online]. Available: http://ftp.csd.uwo.ca/faculty/olga/Courses/Fall2006/Papers/EM_algorithm.pdf

[96] T. Børvik, A. H. Clausen, O. S. Hopperstad, and M. Langseth, “Perforation of AA5083-H116 Aluminum Plates with Conical-Nose Steel Projectiles–Experimental Study,” *International Journal of Impact Engineering*, vol. 30, pp. 367–384, 2004.

[97] T. Børvik, M. J. Forrestal, O. S. Hopperstad, T. L. Warren, and M. Langseth, “Perforation of AA5083-H116 Aluminium Plates with Conical-Nose Steel Projectiles – Calculations,” *International Journal of Impact Engineering*, vol. 36, pp. 426–437, 2009.

[98] T. Børvik, O. S. Hopperstad, M. Langseth, and K. A. Malo, “Effect of Target Thickness in Blunt Projectile Penetration of Weldox 460 E Steel Plates,” *International Journal of Impact Engineering*, vol. 28, pp. 413–464, 2003.

[99] T. Børvik, M. Langseth, O. S. Hopperstad, and K. A. Malo, “Ballistic Penetration of Steel Plates,” *International Journal of Impact Engineering*, vol. 22, pp. 855–886, 1999.

[100] A. Bose, J. Lankford Jr., and H. Couque, “Development and Characterization of Adiabatic Shear Prone Tungsten Heavy Alloys,” U.S. Army Research Laboratory, Contractor Report ARL-CR-60, July 1993.

[101] R. A. Bowers, “Overview of muves 3 and the muves 3 v/l service,” in *Proceedings of the 26$^{th}$ International Symposium of Ballistics*, E. Baker and D. Templeton, Eds., vol. 2. Lancaster, PA.: DEStech Publications, September 2011, pp. 2035–2046.

[102] P. N. Brooks, “On the Prediction of Crater Profiles Produced in Ductile Targets by the Impact of Rigid Penetrators at Ballistic Velocities,” Defence Research Establishment Valcartier, Technical Report DREV R-686/73, January 1973.

[103] N. Bruchey and K. Kimsey, “Penetration of a Highly Oblique Steel Plate by a Thin Disk,” U.S. Army Research Laboratory, Technical Report ARL-TR-2828, September 2002.

[104] W. J. Bruchey, E. J. Horwath, and W. R. Rowe, “The Effect of Crystallographic Orientation on the Performance of Single Crystal Tungsten Subscale Penetrators,” U.S. Army Research Laboratory, Memorandum Report ARL-MR-89, August 1993.

[105] M. Burkins, W. A. Gooch, and D. Weeks, “Performance of a 30-mm armor piercing discarding sabot (apds) munition into steel and aluminum alloy armors,” Army Research Laboratory, Memorandum Report BRL-MR-385, 1998.

[106] M. Burkins, J. Paige, and J. Hansen, “A Ballistic Evaluation of Ti-6Al-4V vs. Long Rod Penetrators,” U.S. Army Research Laboratory, Technical Report ARL-TR-1146, 1996.

[107] P. Burman, “A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods,” *Biometrika*, vol. 76, no. 3, pp. 503–514, 1989.

[108] F. Busetti, “Genetic algorithms overview,” 2007. [Online]. Available: http://citeseerx.ist.psu.edu

[109] L. A. Butler, E. Kerzner, and C. Wyman, “High-Performance Vulnerability/Lethality Analysis,” U.S.Army Research Laboratory, Technical Report ARL-TR-6109, September 2012.

[110] W. D. Callister Jr, *Fundamentals of materials science and engineering.* John Wiley & Sons, 2001.

[111] J. M. Campos, W. Reinecke, and S. Bless, “A Parameter that Combines the Effects of Bend and Angle of Attack on Penetration Degradation of Long Rods,” in *Proceedings 19th International Symposium of Ballistics 7–11 May 2001*, I. R. Crewther, Ed., vol. 3, May 2001, pp. 1297–1304.

[112] A. Canas, E. Ortigosa, E. Ros, and P. Ortigosa, “FPGA Implementation of a Fully and Partially Connected MLP,” in *FPGA Implementations of Neural Networks*, A. R. Omondi and J. C. Rajapakse, Eds. Springer US, 2006, pp. 271–296.

[113] J. G. Carney and P. Cunningham, “The NeuralBAG Algorithm: Optimizing Generalization Performance in Bagged Neural Networks,” in *Proceedings of the 7th European Symposium on Artificial Neural Networks*, 1999.

[114] C. Carrasco, “Modeling Hypervelocity Impact for Kill Enhancement of Ballistic Missile Warheads,” U.S. Air Force Research Laboratory, Technical Report AFRL-SR-AR-TR-04-0595, October 2004.

[115] R. L. Carter, “Solutions for Missing Data in Structural Equation Modeling,” *Research & Practice in Assessment*, vol. 1, no. 1, March 2006.

[116] G. Castellano and A. M. Fanelli, “Variable selection using neural-network models,” *Neurocomputing*, vol. 31, pp. 1–13, 2000.

[117] G. Castellano, A. M. Fanelli, and M. Pelillo, “An iterative pruning algorithm for feedforward neural networks,” *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 519–531, 1997.

[118] P. Castillo, J. Carpio, J. Merelo, A. Prieto, V. Rivas, and G. Romero, “Evolving multilayer perceptrons,” *Neural Processing Letters*, vol. 12, no. 2, pp. 115–128, 2000.

[119] G. C. Cawley and N. L. Talbot, “On over-fitting in model selection and subsequent selection bias in performance evaluation,” *The Journal of Machine Learning Research*, vol. 99, pp. 2079–2107, 2010.

[120] H. Cecotti and A. Gräser, “Neural network pruning for feature selection-application to a p300 Brain-computer interface.” in *ESANN*, 2009.

[121] W. W.-Y. Chan, “A survey on multivariate data visualization,” *Department of Computer Science and Engineering. Hong Kong University of Science and Technology*, vol. 8, no. 6, pp. 1–29, 2006.

[122] A. L. Chang, “In-Situ Measurement of Penetrator Erosion Rate and Dynamic Flow Stress During Long-Rod Penetration,” U.S. Army Research Laboratory, Technical Report ARL-TR-1187, August 1996.

[123] J. Chang and H. K. H. Lee, “Choosing the number of nodes for a neural network via the graphical jump method,” University of California Santa Cruz, Tech. Rep. UCSC-SOE-12-19, October 2012.

[124] A. D. Chapman, *Principles and methods of data cleaning.* GBIF, 2005.

[125] S. Chaudhuri, V. Ganti, and R. Motwani, “Robust identification of fuzzy duplicates,” in *Data Engineering, 2005. ICDE 2005. Proceedings. 21$^{st}$ International Conference on.* IEEE, 2005, pp. 865–876.

[126] Z.-G. Che, T.-A. Chiang, and Z.-H. Che, “Feed-forward neural networks training: a comparison between genetic algorithm and back-propagation learning algorithm,” *Int J Innov Comput Inf*, vol. 7, pp. 5839–5850, 2011.

[127] D. Chen and M. T. Hagan, “Optimal use of regularization and cross-validation in neural network modeling,” in *Neural Networks, 1999. IJCNN’99. International Joint Conference on*, vol. 2. IEEE, 1999, pp. 1275–1280.

[128] T. Chen, C. Zhang, X. Chen, and L. Li, “An input variable selection method for the artificial neural network of shear stiffness of worsted fabrics,” *Statistical Analysis and Data Mining*, vol. 1, no. 5, pp. 287–295, 2009.

[129] X. W. Chen and Q. M. Li, “Perforation of a Thick Plate by Rigid Projectiles,” *International Journal of Impact Engineering*, vol. 28, pp. 743–759, 2003.

[130] X. W. Chen, Y. B. Yang, Z. H. Lu, and Y. Z. Chen, “Perforation of Metallic Plates Struck by a Blunt Projectile with a Soft Nose,” *International Journal of Impact Engineering*, vol. 35, pp. 549–558, 2008.

[131] Y.-C. Cheng, Y. H. Chiu, H.-C. Wang, F.-M. Chang, K.-C. Chung, C.-H. Chang, and K.-S. Cheng, “Using akaike information criterion and minimum mean square error mode in compensating for ultrasonographic errors for estimation of fetal weight by new operators,” *Taiwanese Journal of Obstetrics and Gynecology*, vol. 52, no. 1, pp. 46–52, 2013.

[132] K. Cho, L. Kecskes, R. Dowding, B. Schuster, Q. Wei, and R. Z. Valiev, “Nanocrystalline and Ultra-Fine Grained Tungsten for Kinetic Energy Penetrator and Warhead Liner Applications,” U.S. Army Research Laboratory, Reprint Report ARL-RP-180, June 2007.

[133] I. S. Chocron, C. E. Anderson Jr., T. Behner, and V. Hohler, “Lateral Confinement Effects in Long-Rod Penetration of Ceramics at Hypervelocity,” *International Journal of Impact Engineering*, vol. 33, pp. 169–179, 2006.

[134] S. Chocron, C. E. Anderson Jr., and J. D. Walker, “Analytical Model of Long Rod Interaction with Spaced-Plate Targets,” in *Proceedings 19th International Symposium of Ballistics 7–11 May 2001*, 2001.

[135] S. Chocron, C. E. Anderson Jr., J. D. Walker, and M. Ravid, “A Unified Model for Long-Rod Penetration in Multiple Metallic Plates,” *International Journal of Impact Engineering*, vol. 28, pp. 391–411, 2003.

[136] M. Chris, J. Neville, and S. Prabhakar, “A Statistical Method for Integrated Data Cleaning and Imputation,” Purdue University Computer Science, Technical Report CSD TR 09-008, September 2009.

[137] D. R. Christman and J. W. Gehring, “Analysis of High-Velocity Projectile Penetration Mechanics,” *Journal of Applied Physics*, vol. 37, no. 4, pp. 1579–1588, 1966.

[138] G. Chryssolouris, M. Lee, and A. Ramsey, “Confidence interval prediction for neural network models,” *Neural Networks, IEEE Transactions on*, vol. 7, no. 1, pp. 229–232, 1996.

[139] t. L. J. D. Cinnamon, “Further One-Dimensional Analysis of Long-Rod Penetration of Semi-Infinite Targets,” Master’s thesis, University of Texas, 1992.

[140] C. F. Cline, R. P. Gogolewski, and J. E. Reaugh, “Low Fineness Ratio Kinetic Energy Penetrators,” in *Proceedings of the 11$^{th}$ International Symposium on Ballistics*, 1989.

[141] R. Cody and R. W. Johnson. Data Cleaning 101. Accessed 29 December 2012. [Online]. Available: http://www.ats.ucla.edu/stat/sas/library/nesug99/ss123.pdf

[142] D. Collins. (2008) Data quality: Types of error in database data. Accessed 23 February 2015. [Online]. Available: http://dataquality.origma.co.uk/uploads/fckeditor/file/200805/20080520155902.pdf

[143] L. M. Collins, J. L. Schafer, and C.-M. Kam, “A Comparison of Inclusive and Restrictive Strategies in Modern Missing Data Procedures,” *Psychological Methods*, vol. 6, no. 4, pp. 330–351, 2001.

[144] J. J. Condon and J. R. Baker, “Annual Technical Progress Report on Rod Lethality Studies,” Naval Research Laboratory, Memorandum Report NRL MR 18092, September 1967.

[145] S. Conrady and L. Jouffe. (2011, December) Missing Values Imputation: A New Approach to Missing Values Processing with Bayesian Networks. Conrady Applied Science, LLC. [Online]. Available: http://library.bayesia.com/download/attachments/3441122/missing_values_v14.pdf

[146] A. Copland, T. W. Bjerke, and D. Weeks, "Semi-Infinite Target Design Effects on Terminal Ballistics Performance - Influence of Plate Spacing on Penetrator Energy Partitioning," U.S. Army Research Laboratory, Technical Report ARL-TR-3688, December 2005.

[147] A. Copland and D. Scheffler, "Influence of Air Gaps on Long Rod Penetrators Attacking Multi-Plate Target Arrays," U.S. Army Research Laboratory, Technical Report ARL-TR-2906, February 2003.

[148] K. P. Corporation, *Material Science DOE Fundamentals Handbook: Volumes 1 And 2*, ser. DOE Material Science. Knowledge Publications Corporation, 2006. [Online]. Available: http://books.google.com/books?id=VBMlnwEACAAJ

[149] M. A. Costa, A. P. Braga, and B. R. de Menezes, "Improving neural networks generalization with new constructive and pruning methods," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 13, no. 2-4, pp. 75–83, 2002.

[150] G. Crowell, "The descriptive geometry of nose cones," 1996. [Online]. Available: http://www.if.sc.usp.br

[151] M. Cubedo and J. Oller, "Hypothesis testing: a model selection approach," *Journal of statistical planning and inference*, vol. 108, no. 1, pp. 3–21, 2002.

[152] I. G. Cullis and N. J. Lynch, "The Influence of Aerodynamic Heating on the Terminal Ballistic Performance of Scale Size KE-Projectiles," *International Journal of Impact Engineering*, vol. 20, pp. 185–196, 1997.

[153] I. Cullis and N. J. Lynch, "Performance of Model Scale Long Rod Projectiles Against Complex Targets Over The Velocity Range 1700–2200 m/s," *International Journal of Impact Engineering*, vol. 17, pp. 263–274, 1995.

[154] W. Dai, I. Wardlaw, Y. Cui, K. Mehdi, Y. Li, and J. Long, *Information Technology: New Generations: 13th International Conference on Information Technology.* Cham: Springer International Publishing, 2016, ch. Data Profiling Technology of Data Governance Regarding Big Data: Review and Rethinking, pp. 439–450. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-32467-8_39

[155] S. Danaher, S. Datta, I. Waddle, and P. Hackney, "Erosion modelling using Bayesian regulated artificial neural networks," *Wear*, vol. 256, no. 9-10, pp. 879–888, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0043164803006355

[156] K. Daneshjou and M. Shahravi, "Penetrator Strength Effect in Long-Rod Critical Ricochet Angle," *Journal of Mechanical Science and Technology*, vol. 22, pp. 2076–2089, 2008.

[157] P. Daniušis and P. Vaitkus, "Neural network with matrix inputs," *Informatica*, vol. 19, no. 4, pp. 477–486, 2008.

[158] V. N. Dao and V. Vemuri, "A performance comparison of different back propagation neural networks methods in computer network intrusion detection," *Differential equations and dynamical systems*, vol. 10, no. 1&2, pp. 201–214, 2002.

[159] I. G. N. Darmawan, "NORM software review: handling missing values with multiple imputation methods," *Evaluation Journal of Australasia*, vol. 2, no. 1, pp. 51–57, 2002.

[160] R. Davidson, *Stochastic Dominance*, 2nd ed., 2006.

[161] R. Davitt, "A comparison of the advantages and disadvantages of depleted uranium and tungsten alloy as penetrator materials," Large Caliber Waepons System Laboratory, Technical Report Tank Ammo Section Report No. 107, June 1980.

[162] J. E. Dayhoff, *Neural Network Architectures.* International Thomson Computer Press, 1996.

[163] W. S. de Rosset, "Optimum Velocity for High-Velocity Penetrators," Ballistic Research Laboratory, Technical Report BRL-TR-3377, July 1992.

[164] ——, "An Overview of Novel Penetrator Technology," U.S. Army Research Laboratory, Technical Report ARL-TR-2395, February 2001.

[165] R. Decker and H.-J. Lenz, Eds., *Advances in Data Analysis, [Proceedings of the 30$^{th}$ Annual Conference of the Gesellschaft für Klassifikation e.V., Freie Universität Berlin, March 8-10, 2006].* Springer, 2007.

[166] J. T. Dehn, "A Unified Theory of Penetration," Ballistic Research Laboratory, Technical Report BRL-TR-2770, December 1986.

[167] P. H. Deitz, W. E. Baker, D. C. Bely, B. A. Bodt, A. LaGrange, R. W. Lauzze II, J. T. K. Scott L. Henry, J. A. Morrissey, L. K. Roach, S. M. Robinson, J. H. Smith, and J. N. Walbert, "Evaluation Strategies for Live-Fire Planning, Analysis, and Testing," U.S. Army Research Laboratory, Technical Report ARL-TR-1273, December 1996.

[168] P. H. Deitz and A. Ozolins, "Computer Simulations of the Abrams Live-Fire Field Testing," Ballistic Research Laboratory, Memorandum Report BRL-MR-3755, May 1989.

[169] ——, "High-Resolution Vulnerability Methods and Applications," Ballistic Research Laboratory, Memorandum Report BRL-MR-3876, November 1990.

[170] P. H. Deitz, H. L. Reed Jr., J. T. Klopcic, and J. N. Walbert, *Fundamentals Of Ground Combat System Ballistic Vulnerability/Lethality*, ser. Progress In Astronautics And Aeronautics, E. Edwards, W. Hacker, W. Kincheloe, and D. Bely, Eds. AIAA (American Institute of Aeronautics & Astronautics), 2009.

[171] W. H. Delashmit and M. T. Manry, "Recent developments in multilayer perceptron neural networks," in *Proceedings of the seventh Annual Memphis Area Engineering and Science Conference, MAESC*, 2005.

[172] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[173] L. Deng, G. Hinton, and B. Kingsbury, "New types of deep neural network learning for speech recognition and related applications: an overview," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 8599–8603.

[174] F. M. Dias, A. Antunes, J. Vieira, and A. M. Mota, "Implementing the levenberg-marquardt algorithm on-line: A sliding window approach with early stopping," in *$2^{nd}$ IFAC Workshop on Advanced Fuzzy/Neural Control*, 2004.

[175] R. E. Dibelka, "MUVES-S2 Configuration Management," June 2004.

[176] D. L. Dickinson and L. T. Wilson, "The Impact of Long Rods into Spaced-Plate Arrays," *International Journal of Impact Engineering*, vol. 23, pp. 193–204, 1999.

[177] J. Diebolt and E. Ip, "A Stochastic EM algorithm for approximating the maximum likelihood estimate," Stanford University, Technical Report 301, September 1994.

[178] R. Dittrich, B. Francis, R. Hatzinger, and W. Katzenbeisser, "Missing Observations in Paired Comparison Data," *Statistical Modelling*, vol. 12, no. 2, pp. 117–143, 2012.

[179] C. B. Do and S. Batzoglou, "What is the expectation maximization algorithm?" *Nature Biotechnology*, vol. 26, no. 8, pp. 897–899, August 2008.

[180] A. R. T. Donders, G. J. van der Heijden, T. Stijnen, K. G. Moons *et al.*, "Review: a gentle introduction to imputation of missing values," *Journal of clinical epidemiology*, vol. 59, no. 10, pp. 1087–1091, 2006.

[181] R. E. Dorsey, R. O. Edmister, and J. D. Johnson, *Bankruptcy prediction using artificial neural systems.* Research Foundation of The Institute of Chartered Financial Analysts, 1995.

[182] R. J. Dowding, "Tungsten Alloy Technology," U.S. Army Research Laboratory, Memorandum Report ARL-MR-57, January 1993.

[183] R. J. Dowding, K. J. Tauer, P. Woolsey, and F. S. Hodi, "The Metallurgical and Ballistic Characterization of Quarter-Scale Tungsten Alloy Penetrators," Materials Technology Laboratory, Technical Report MTL TR 90-31, May 1990.

[184] P. Dragicevic, “My technique is 20% faster: problems with reports of speed improvements in hci,” 2012.

[185] W. H. Drysdale, R. D. Kirkendall, and L. D. Kokinakis, “Sabot design for a 105mm apfsds kinetic energy projectile,” DTIC Document, Tech. Rep., 1978.

[186] W. Dwinnell, “Modeling methodology 2: model input selection,” vol. 12, p. 32, 1998.

[187] B. Efron, “Missing data, imputation, and the bootstrap,” Stanford University, Technical Report 153, July 1992.

[188] E. Elizalde, S. Gomez, and A. Romeo, “Encoding strategies in multilayer neural networks,” *Journal of Physics A: Mathematical and General*, vol. 24, no. 23, p. 5617, 1991.

[189] C. Elkan, “Evaluating classifiers,” 2012.

[190] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, “Duplicate record detection: A survey,” *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 1, pp. 1–16, 2007.

[191] C. K. Enders, “A Primer on Maximum Likelihood Algorithms Available for Use With Missing Data,” *Structural Equation Modeling*, vol. 8, no. 1, pp. 128–141, 2001.

[192] ——, “The Performance of the Full Information Maximum Likelihood Estimator in MultipleRegression Models with Missing Data,” *Education and Psychological Measurement*, vol. 61, pp. 713–740, 2001.

[193] C. K. Enders and D. L. Bandalos, “The Relative Performance of Full Information Maximum Likelihood Estimation for Missing Data in Structural Equation Models,” *Structural Equation Modeling*, vol. 8, no. 3, pp. 430–457, 2001.

[194] C. M. Ennett, M. Frize, C. R. Walker *et al.*, “Influence of missing values on artificial neural network performance,” *Studies in health technology and informatics*, no. 1, pp. 449–453, 2001.

[195] D. C. Erlich, L. Seaman, and D. A. Shockey, "Computational Model for Armor Penetration," Ballistic Research Laboratory, Contract Report BRL-CR-584, October 1987.

[196] T. F. Erline, "Dispersion analysis of the xm881 armor-piercing, fin-stabilized, discarding sabot (apfsds) projectile," Army Research Laboratory, Memorandum Report ARL-MR-433, 1999.

[197] H. J. Ernst, W. Lanz, and T. Wolf, "Penetration Comparison of L/D=20 and 30 Mono-Block Penetrators with L/D=40 Jacketed Penetrators in Different Target Materials," in *Proceedings 19$^{th}$ International Symposium of Ballistics 7–11 May 2001*, I. R. Crewther, Ed., vol. 3, May 2001, pp. 1151–1157.

[198] H. D. Espinosa, N. S. Brar, G. Yuan, Y. Xu, and V. Arrieta, "Enhanced Ballistic Performance of Confined Multi-Layered Ceramic Targets Against Long Rod Penetrators through Interface Defeat," *International Journal of Solids and Structures*, vol. 37, pp. 4893–4913, 2000.

[199] M. Ester, H.-P. Kriegal, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *Proceedings of 2$^{nd}$ International Conference on Knowledge Discovery and Data Mining*, 1996.

[200] H. Fang, J. Hölzle, R. Knobel, and W. Lanz, "Joining jacket and core in jacketed steel/tungsten penetrators," in *Proceedings of the 19$^{th}$ Int. Symp. on Ballistics*, 2001.

[201] Y. Fang, "Asymptotic equivalence between cross-validations and akaike information criteria in mixed-effects models," *Journal of data science*, vol. 9, no. 1, pp. 15–21, 2011.

[202] T. G. Farrand, "Terminal Ballistic Evaluation of a Candidate 0.60-Caliber Electromagnetically Launched Long-Rod Penetrator," U.S. Army Research Laboratory, Memorandum Report ARL-MR-104, September 1993.

[203] D. Fernández-Fdz, J. López-Puente, and R. Zaera, "Prediction of the behaviour of CFRPs against high-velocity impact of solids employing an artificial neural network methodology," *Composites Part A: Applied Science and Manufacturing*, vol. 39, no. 6, pp. 989–996, 2008.

[204] D. Fernández-Fdz and R. Zaera, “A new tool based on artificial neural networks for the design of lightweight ceramic-metal armour against high-velocity impact of solids,” *International Journal of Solids and Structures*, vol. 45, no. 25-26, pp. 6369–6383, 2008.

[205] T. Fernando, H. R. Maier, G. C. Dandy, and R. May, “Efficient selection of inputs for artificial neural network models,” in *Proc. of MODSIM 2005 International Congress on Modelling and Simulation: Modelling and Simulation Society of Australia and New Zealand*, 2005, pp. 1806–1812.

[206] E. A. Ferriter, I. McCulloh, and W. DeRosset, “Techniques Used to Estimate Limit Velocity in Ballistics Testing with Small Sample Size,” in *Proceedings of the 13$^{th}$ Annual U.S. Army Research Laboratory/United States Military Academy Technical Symposium*, 2005, pp. 72–95.

[207] P. A. Fishwick, “Neural Network Models in Simulation: A Comparison with Traditional Modeling Approaches,” in *Proceedings of the 1989 Winter Simulation Conference*, 1989.

[208] M. J. Forrestal, N. S. Brar, and Luk, “Penetration of Strain Hardening Targets with Rigid Spherical-Nose Rods,” *ASME Journal of Applied Mechanics*, vol. 58, pp. 7–10, 1991.

[209] M. J. Forrestal and S. J. Hanchak, “Perforation Experiments on HY-100 Steel Plates with 4340$R_c$38 and Maraging T-250 Steel Rod Projectiles,” *International Journal of Impact Engineering*, vol. 22, pp. 923–933, 1999.

[210] M. J. Forrestal, K. Okajima, and V. K. Luk, “Penetration of 6061-T651 Aluminum Targets with Rigid Long Rods,” *ASME Journal of Applied Mechanics*, vol. 55, pp. 755–760, 1988.

[211] M. J. Forrestal and A. J. Piekutowski, “Penetration Experiments with 6061-T6511 Aluminum Targets and Spherical-Nose Projectiles at Striking Velocities Between 0.5 and 3.0 km/s,” *International Journal of Impact Engineering*, vol. 24, pp. 57–67, 2000.

[212] M. J. Forrestal and T. L. Warren, “Penetration Equations for Ogive-Nose Rods into Aluminum Targets,” *International Journal of Impact Engineering*, vol. 35, pp. 727–730, 2008.

[213] ——, “Perforation Equations for Conical and Ogival Nose Rigid Projectiles into Aluminum Target Plates,” *International Journal of Impact Engineering*, vol. 36, pp. 220–225, 2009.

[214] T. Foss, E. Stensrud, B. Kitchenham, and I. Myrtveit, “A simulation study of the model evaluation criterion mmre,” *IEEE Transactions on Software Engineering*, vol. 29, no. 11, pp. 985–995, 2003.

[215] K. Frank, “Armor-Penetrator Performance Measures,” U.S. Army Ballistic Research Laboratory, Memorandum Report ARBRL-MR-03097, March 1981.

[216] K. Frank and J. Zook, “Energy-Efficient Penetration of Targets,” U.S. Ballistics Research Laboratory, Memorandum Report BRL-MR-3885, February 1991.

[217] D. Y. Frank Seide, Gang Li, “Conversational speech transcription using context-dependent deep neural networks,” in *Interspeech 2011*. International Speech Communication Association, August 2011. [Online]. Available: https://www.microsoft.com/en-us/research/publication/conversational-speech-transcription-using-context-dependent-deep-neural-networks/

[218] R. R. Franzen, D. L. Orphal, and C. E. Anderson Jr., “The Influence of Experimental Design on Depth-of-Penetration (DOP) Test Results and Derived Ballistic Efficiencies,” *International Journal of Impact Engineering*, vol. 19, no. 8, pp. 727–737, 1997.

[219] J. A. Freeman, *Simulating Neural Networks with Mathematica.* Reading, MA: Addison-Wesley, 1994.

[220] J. H. Friedman and J. W. Tukey, “A Projection Pursuit Algorithm for Exploratory Data Analysis,” *IEEE Transactions on Computers*, vol. c-23, pp. 881–889, 1974.

[221] L.-M. Fu, *Neural Networks in Computer Intelligence.* McGraw-Hill Education, 2003.

[222] E. Fugelso, “Further Experimental Studies on the Effect of Combined Obliquity and Yaw on the Perforation of Stell Plates by Long-Rod U-0.75 WT% Ti Penetrators,” Los Alamos Scientific Laboratory, Los Alamos, NM, Informal Report LA-8357-MS, May 1980.

[223] E. Fugelso and J. W. Taylor, “Evaluation of Combined Obliquity and Yaw for U 0.75 wt% Ti Penetrators,” Los Alamos Scientific Laboratory, Los Alamos, NM, Informal Report LA-7402-MS, July 1978.

[224] B. A. Galanov, S. M. Ivanov, and V. V. Kartuzov, “On One New Modification of Alekseevskii-Tate Model for NonStationary Penetration of Long-Rods into Targets,” *International Journal of Impact Engineering*, vol. 26, pp. 201–210, 2001.

[225] T. D. Ganchev, K. E. Parsopoulos, M. N. Vrahatis, and N. D. Fakotakis, *Partially Connected Locally Recurrent Probabilistic Neural Networks.* I-Tech, 2008, pp. 377–400. [Online]. Available: http://www.i-techonline.com

[226] M. Ganesh Babu, R. Velmurugan, and N. K. Gupta, “Energy Absorption and Ballistic Limit of Targets Struck by Heavy Projectile,” *Latin American Journal of Solids and Structures*, vol. 3, pp. 21–39, 2006.

[227] E. R. Gansner, E. Koutsofios, S. C. North, and G.-P. Vo, “A technique for drawing directed graphs,” *Software Engineering, IEEE Transactions on*, vol. 19, no. 3, pp. 214–230, 1993.

[228] V. Ganti, R. Ramakrishnan, J. Gehrke, A. Powell, and J. French, “Clustering large datasets in arbitrary metric spaces,” in *Proceedings of the 15th International Conference on Data Engineering*, March 1999, pp. 502–511.

[229] V. Ganti and A. D. Sarma, *Data Cleaning: A Practical Perspective.*, ser. Synthesis Lectures on Data Management. San Rafael : Morgan & Claypool Publishers, 2013., 2013. [Online]. Available: http://proxy-tu.researchport.umd.edu/login?ins=tu&url=http://search.ebscohost.com.proxy-tu.researchport.umd.edu/login.aspx?direct=true&db=cat01451a&AN=towson.004358610&site=eds-live&scope=site

[230] A. Garcia-Crespo, B. Ruiz-Mezcua, D. Fernandez-Fdz, and R. Zaera, “Prediction of the response under impact of steel armours using a multilayer perceptron,” *Neural Computing & Applications*, vol. 16, no. 2, pp. 147–154, 2007.

[231] H. Gavin, “The levenberg-marquardt method for nonlinear least squares curve-fitting problems,” *Department of Civil and Environmental Engineering, Duke University*, 2011.

[232] D. J. Gee, “Oblique Plate Perforation by Slender Rod Projectiles,” in *Proceedings 19$^{th}$ International Symposium of Ballistics 7–11 May 2001*, 2001.

[233] ——, “Plate Perforation by Eroding Rod Projectiles,” *International Journal of Impact Engineering*, vol. 28, pp. 377–390, 2003.

[234] D. J. Gee and D. L. Littlefield, “Yaw Impact of Rod Projectiles,” *International Journal of Impact Engineering*, vol. 26, pp. 211–220, 2001.

[235] P. Geiger and C. Dellago, “Neural networks for local structure detection in polymorphic systems,” *The Journal of chemical physics*, vol. 139, no. 16, p. 164105, 2013.

[236] A. Gelman and J. Hill., *Data Analysis Using Regression and Multilevel/Hierarchical Models.* Cambridge University Press, 2006, ch. 25, pp. 529–543.

[237] C. Gershenson, “Artificial Neural Networks for Beginners,” 2003. [Online]. Available: http://arxiv.org/abs/cs.NE/0308031

[238] A. Ghaffari, H. Abdollahi, M. Khoshayand, I. S. Bozchalooi, A. Dadgar, and M. Rafiee-Tehrani, “Performance comparison of neural network training algorithms in modeling of bimodal drug delivery,” *International journal of pharmaceutics*, vol. 327, no. 1, pp. 126–138, 2006.

[239] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Aistats*, vol. 9, 2010, pp. 249–256.

[240] R. P. Godwin and E. J. Chapyak, “Apparent Target Strength in Long-Rod Penetration,” *International Journal of Impact Engineering*, vol. 21, no. 1–2, pp. 77–88, 1998.

[241] K. Goh, R. Aspden, K. Mathias, and D. Hukins, “Finite–element analysis of the effect of material properties and fibre shape on stresses in an elastic fibre embedded in an elastic matrix in a fibre–composite material,” in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 460, no. 2048. The Royal Society, 2004, pp. 2339–2352.

[242] L. Gong, C. Liu, Y. Li, and F. Yuan, “Training feed-forward neural networks using the gradient descent method with the optimal stepsize,” *Journal of Computational Information Systems*, vol. 8, no. 4, pp. 1359–1371, 2012.

[243] I. Gonzalez-Carrasco, A. Garcia-Crespo, B. Ruiz-Mezcua, and J. L. Lopez-Cuadrado, “Neural Network Application for High Speed Impacts Classification,” in *Proceedings of the World Congress on Engineering*, vol. 1, 2008.

[244] ——, “Dealing with limited data in ballistic impact scenarios: an empirical comparison of different neural network approaches,” *Applied Intelligence*, vol. 35, no. 1, pp. 89–109, 2011.

[245] ——, “A neural network-based methodology for the recreation of high-speed impacts on metal armours,” *Neural Computing & Applications*, vol. 21, pp. 91–107, 2012.

[246] ——, “An Optimization Methodology for Machine Learning Strategies and Regression Problems in Ballistic Impact Scenarios,” *Applied Intelligence*, vol. 36, pp. 424–441, 2012.

[247] W. A. Gooch and M. Burkins, “A Ballistic Evaluation of Ti-6Al-4V vs. Long Rod Penetrators,” in *RTO Meeting Proceedings 69(II); Low Cost Composite Structures and Cost Effective Application of Titanium Alloys in Military Platforms*, May 2001, nATO RTO-MP-069 (II).

[248] W. A. Gooch, M. Burkins, H. J. Ernst, and T. Wolf, “Ballistic penetration of titanium alloy ti-6al-4v,” in *Proceedings of the Lightweight Armour Systems Symposium*, 1995.

[249] C. Grabarek, “Penetration of Armor by Steel and High Density Penetrators,” U.S. Army Ballistic Research Laboratories, Memorandum Report BRL-MR-2134, 1971.

[250] ——, “An armor penetration predictive scheme for small arms ap ammunition,” Ballistic Research Laboratory, Memorandum Report BRL-MR-2620, April 1976.

[251] C. Grabarek and L. Herr, “X-Ray Multi-Flash System for Measurement of Projectile Performance at the Target,” Ballistic Research Laboratories, Technical Note BRL-TN-1634, September 1966.

[252] C. L. Grabarek and A. J. Ricchiazzi, “Long Rod Impacts on Finite Targets,” in *Proceedings of the Army Symposium on Solid Mechanics*, no. AMMRC MS 68-09. Applied Mechanics Research Laboratory, September 1968, pp. 95–108.

[253] F. I. Grace, “Long-Rod Penetration Into Targets of Finite Thickness at Normal Impact,” *International Journal of Impact Engineering*, vol. 16, no. 3, pp. 419–433, 1995.

[254] F. Grace, “Nonsteady penetration of long rods into semi-infinite targets,” *International Journal of Impact Engineering*, vol. 14, no. 1, pp. 303–314, 1993.

[255] ——, “Long-rod penetration into targets of finite thickness at normal impact,” *International Journal of Impact Engineering*, vol. 16, no. 3, pp. 419–433, 1995.

[256] F. Grace and N. Rupert, “Mechanisms for ceramic/metal, bi-element target,” in *Proceedings 14th Int. Symp. on Ballistics, Quebec City, Quebec, Canada*, 1993.

[257] F. I. Grace and N. L. Rubert, “Analysis of Long Rods Impacting Ceramic Targets at High Velocity,” U.S. Army Research Laboratory, Technical Report ARL-TR-1493, September 1997.

[258] J. W. Graham, *Missing data: Analysis and design.* Springer, 2012, ch. Chapter 2: Analysis of Missing Data, pp. 47–69.

[259] ——, “Missing Data Analysis: Making It Work in the Real World,” *The Annual Review of Psychology*, vol. 60, pp. 549–576, 2009.

[260] J. Green and P. Moy, “Large strain compression of two tungsten alloys at various strain rates,” U.S. Army Materials Technology Laboratory, Technical Report MTL-TR-92-66, September 1992.

[261] M. Green and M. Ohlsson, "Comparison of standard resampling methods for performance estimation of artificial neural network ensembles," in *Third International Conference on Computational Intelligence in Medicine and Healthcare*, 2007.

[262] R. C. Grubinskas and R. J. Squillacioti, "The development of an automated armor data base - phase 1," U.S. Army Research Laboratory, Technical Report ARL-TR-218, September 1993.

[263] E. Gruss and E. Hirsch, "Approximating the Ballistic Penetration Function of a Jet in a Multi-Cassette Target by the use of Neural Networks," in *Proceedings 19th International Symposium of Ballistics 7–11 May 2001*, May 2001, pp. 1069–1075.

[264] M. R. Gupta and Y. Chen, "Theory and Use of the EM Algorithm," *Foundations and Trends in Signal Processing*, vol. 4, no. 3, pp. 223–296, 2010.

[265] N. K. Gupta, M. A. Iqbal, and G. S. Sekhon, "Effect of Projectile Nose Shape, Impact Velocity, and Target Thickness on the Deformation Behavior of Layered Plates," *International Journal of Impact Engineering*, vol. 35, pp. 37–60, 2008.

[266] N. K. Gupta and V. Madhu, "An Experimental Study of Normal and Oblique Impact of Hard-Core Projectile on Single and Layered Plates," *International Journal of Impact Engineering*, vol. 19, no. 5–6, pp. 395–414, 1997.

[267] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[268] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.

[269] L. Hagemo. (2007, July) Java Coding Practices for Improved Application Performance. Candle. [Online]. Available: http://www.capitalware.biz/dl/docs/WhitePaperJavaCodingPractices.pdf

[270] Y. Haitovsky, "Missing Data in Regression Analysis," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 30, no. 1, pp. 67–82, 1968.

[271] A. Hameed, J. G. Hetherington, and R. D. Brown, “Design trends in the development of large-calibre kinetic-energy rounds,” *Journal of Battlefield Technology*, vol. 7, no. 3, p. 9, 2004.

[272] J. Han and J. Gao, “Research challenges for data mining in science and engineering,” *Next Generation of Data Mining*, pp. 1–18, 2009.

[273] P. J. Hancock, “An empirical comparison of selection methods in evolutionary algorithms,” in *Evolutionary Computing.* Springer, 1994, pp. 80–94.

[274] P. J. Hanes, S. L. Henry, G. S. Moss, K. R. Murray, and W. A. WInner, “Modular Unix-Based Vulnerability Estimation Suite (MUVES) Analyst’s Guide,” Ballistic Research Laboratory, Memorandum Report BRL-MR-3954, December 1991.

[275] J. Hansen and A. Krogh, “A General Method for Combining Predictors Tested on Protein Secondary Structure Prediction,” in *In Proceedings of Artificial Neural Networks in Medicine and Biology.* Springer-Verlag, 2000, pp. 259–264.

[276] G. E. Hauver, N. Huffington, K. Kimsey, L. Magness, M. Raftenberg, G. Randers-Pehrson, M. Scheidler, S. Segletes, J. Walter, and T. Wright, “Material modeling for terminal ballistic simulation,” Ballistics Research Laboratory, Technical Report BRL-TR-3392, September 1992.

[277] G. E. Hauver and A. Melani, “Behavior of Segmented Rods During Penetration,” Ballistic Research Laboratory, Technical Report BRL-TR-3129, July 1990.

[278] G. E. Hauver, E. J. Rapacki Jr., P. H. Netherwood., and R. F. Benck, “Interface Defeat of Long-Rod Projectiles by Ceramic Armor,” U.S. Army Research Laboratory, Technical Report ARL-TR-3590, September 2005.

[279] S. He and J. Li, *Confidence intervals for neural networks and applications to modeling engineering materials.* INTECH Open Access Publisher, 2011.

[280] J. Heaton, *Introduction to Neural Networks with Java*, 2nd ed. Heaton Research, Inc., 2008.

[281] ——, *Programming Neural Networks with Encog3 in Java*, WordsRU.com, Ed. Heaton Research, Inc, 2011.

[282] ——. (2016) Encog machine learning framework. [Online]. Available: http://www.heatonresearch.com/encog/

[283] M. Held, "A tutorial on the penetration of kinetic-energy (ke) rounds," *Journal of Battlefield Technology*, vol. 7, no. 1, p. 1, 2004.

[284] J. M. Hellerstein, "Quantitative Data Cleaning for Large Databases," *United Nations Economic Commission for Europe (UNECE)*, 2008.

[285] B. P. Helms and W. H. Jean, "An algorithm for $N$th degree stochastic dominance," vol. 2, pp. 71 – 81, 1986.

[286] W. HeMing, H. Yu, and L. Bin, "Analytical Model for Cratering of Semi-Infinite Metallic Targets by Long Rod Penetrators," *Science China Technological Sciences*, vol. 53, no. 12, pp. 3189–3196, December 2010.

[287] M. A. Hernández and S. J. Stolfo, "The merge/purge problem for large databases," in *ACM SIGMOD Record*, vol. 24, no. 2. ACM, 1995, pp. 127–138.

[288] ——, "Real-world data is dirty: Data cleansing and the merge/purge problem," *Data mining and knowledge discovery*, vol. 2, no. 1, pp. 9–37, 1998.

[289] E. L. Herr and C. Grabarek, "Standardizing the Evaluation of Candidate Materials for High L/D Penetrators," Ballistic Research Laboratory, Memorandum Report ARBRL-MR-02860, September 1978.

[290] L. Herr and C. Grabarek, "Ballistic performance and beyond armor data for rods impacting steel armor plates," Ballistics Research Laboratory, Memorandum Report BRL-MR-2575, January 1976.

[291] V. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85–126, 2004.

[292] V. Hohler, E. Schneider, A. J. Stilp, and R. Tham, "Length and Velocity Reduction of High Density Rods Perforating Mild Steel and Armor Steel Plates," in *Proceedings of the 4th International Symposium on Ballistics*, 1978.

[293] V. Hohler and A. J. Stilp, "Penetration of Steel and High Density Rods in Semi–Infinite Steel Targets," in *Proceedings 3rd International Symposium of Ballistics 1977*, 1977.

[294] ——, "Influence of the Length-to-Diameter Ratio in the Range from 1 to 32 on the Penetration Performance of Rod Projectiles," in *Proceedings of the 8th International Symposium on Ballistics*, 1984.

[295] ——, "Hypervelocity Impact of Rod Projectiles with L/D from 1 to 32," *International Journal of Impact Engineering*, vol. 5, no. 1–4, pp. 323–331, 1987.

[296] ——, "Penetration Performance of Segmented Rods at Different Spacing - Comparison with Homogeneous Rods at 2.5 - 3.5 km/s," in *Proceedings of the 12th International Symposium on Ballistics*, 1990.

[297] V. Hohler, A. J. Stilp, and K. Weber, "Hypervelocity Penetration of Tungsten Sinter-Alloy Rods into Aluminum," *International Journal of Impact Engineering*, vol. 17, pp. 409–418, 1995.

[298] T. Holmquist, D. Templeton, K. Bishnoi *et al.*, "A ceramic armor material database," U.S. Army Tank-Automotive Research, Development, and Engineering Center, Technical Report 13754, 1999.

[299] N. J. Horton and K. P. Kleinman, "Much Ado About Nothing: A Comparison of Missing Data Methods and Software to Fit Incomplete Data Regression Models," *The American Statistician*, vol. 61, no. 1, pp. 79–90, February 2007.

[300] N. J. Horton and S. R. Lipsitz, "Multiple Imputation in Practice: Comparison of Software Packages for Regression ModelsWith Missing Variables," *The American Statistician*, vol. 55, no. 3, pp. 244–254, August 2001.

[301] R. M. Hristev, "Matrix techniques in artificial neural networks," Master's thesis, 2000.

[302] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," 2010. [Online]. Available: http://www.csie.ntu.edu.tw

[303] F.-L. Huang and L.-S. Zhang, "Investigation on Ballistic Performance of Armor Ceramics against Long-Rod Penetration," *Metallurgical and Materials Transactions A*, vol. 38A, pp. 2891–2895, December 2007.

[304] J. Huang, Y.-F. Li, and M. Xie, "An empirical analysis of data preprocessing for machine learning-based software cost estimation." *Information and Software Technology*, vol. 67, pp. 108 – 127, 2015. [Online]. Available: http://proxy-tu.researchport.umd.edu/login?ins=tu&url=http://search.ebscohost.com.proxy-tu.researchport.umd.edu/login.aspx?direct=true&db=edselp&AN=S0950584915001275&site=eds-live&scope=site

[305] J. A. Huguet, S. Reichwein, S. E. Jones, and W. P. Walters, "Small parameter analysis of the modified tate equations," Army Research Laboratory, Reprint Report ARL-RP-199, 2008.

[306] J. E. Hunt, R. E. Dibelka, W. A. WInner, M. D. Milam, and C. J. Gillich, "MUVES-S2 Accreditation Support Package: Volume I," U.S. Army Research Laboratory, Technical Report ARL-TR-3025, July 2003.

[307] T. Q. Huynh and R. Setiono, "Effective neural network pruning using cross-validation," in *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, vol. 2. IEEE, 2005, pp. 972–977.

[308] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International journal of forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[309] T. Iliou, C.-N. Anagnostopoulos, I. M. Stephanakis, and G. Anastassopoulos, "A novel data preprocessing method for boosting neural network performance: A case study in osteoporosis prediction." *Information Sciences*, 2015. [On-

line]. Available: http://proxy-tu.researchport.umd.edu/login?ins=tu&url=http://search.ebscohost.com.proxy-tu.researchport.umd.edu/login.aspx?direct=true&db=edselp&AN=S0020025515007549&site=eds-live&scope=site

[310] International Ballistics Society. (2012) Ballistic Fields. [Online]. Available: http://www.ballistics.org

[311] M. A. Iqbal and N. K. Gupta, "Energy Absorption Characteristics of Aluminum Plates Subjected to Projectile Impact," *Latin American Journal of Solids and Structures*, vol. 5, pp. 259–287, 2008.

[312] M. J. Islam, M. Ahmadi, M. A. Sid-Ahmed, and Y. M. Alginahi, "Optimal Parameter Selection Technique for a Neural Network Based Local Thresholding Method," *Journal of Pattern Recognition Research*, vol. 1, pp. 69–94, 2010.

[313] J. Jantzen, "Introduction to perceptron networks," *Technical University of Denmark, Lyngby, Denmark, Technical Report*, 1998.

[314] R. Jeanquartier and W. Odermatt, "Post-perforation length and velocity of ke projectiles with single oblique targets," in *15th International Symposium on Ballistics, Proceedings, Volume1*, 1995, pp. 245–252.

[315] L. Jing, "Missing Data Imputation," Master's thesis, University of California, 2012.

[316] G. R. Johnson, "Lagrangian computational approaches for penetration and perforation of solids," *ASME Applied Mechanics Division Advances in Numerical Simulation*, vol. 171, pp. 49–49, 1993.

[317] G. R. Johnson and W. H. Cook, "A constitutive model and data for metals subjected to large strains, high strain rates and high temperatures," in *Proceedings of the 7th International Symposium on Ballistics*, vol. 21. The Netherlands, 1983, pp. 541–547.

[318] N. Jones and J. K. Paik, "Impact Perforation of Aluminum Alloy Plates," *International Journal of Impact Engineering*, vol. 48, pp. 46–53, 2012.

[319] J. Joo, C. Lee, and J. Choi, “A Numerical Research on the Penetration into a Semi-Infinite Rolled Homogeneous Armor by a Medium-Caliber Kinetic Energy Projectile,” *Materials Science Forum*, vol. 673, pp. 197–202, 2011.

[320] I. Jordanov and N. Petrov, “Sets with incomplete and missing data—nn radar signal classification,” in *Neural Networks (IJCNN), 2014 International Joint Conference on.* IEEE, 2014, pp. 218–224.

[321] R. Kapoor, D. Pal, and J. Chakravartty, “Use of artificial neural networks to predict the deformation behavior of Zr-2.5 Nb-0.5 Cu,” *Journal of Materials Processing Technology*, vol. 169, no. 2, pp. 199–205, 2005.

[322] T. Kärkkäinen, “On cross-validation for mlp model evaluation,” in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR).* Springer, 2014, pp. 291–300.

[323] I. H. Kazi, H. H. Chen, B. Stanley, and D. J. Lilja, “Techniques for obtaining high performance in java programs,” *ACM Computing Surveys (CSUR)*, vol. 32, no. 3, pp. 213–240, 2000.

[324] M. J. Keele, E. J. Rapacki Jr., and W. J. Bruchey Jr., “High Velocity Performance of a Uranium Alloy Long Rod Penetrator,” Ballistic Research Laboratory, Technical Report BRL-TR-3236, May 1991.

[325] E. W. Kennedy, M. N. Raftenberg, and D. L. Diehl, “Perforation of Rolled Homogeneous Armor Steel by 91W-6Ni-3Co Penetrators Impacting at Three Velocity Regimes,” U.S. Army Research Laboratory, Technical Report ARL-TR-1787, September 1998.

[326] K. S. Kim and J. H. Lee, “Simplified Vulnerability Assessment Procedure for a Warship Based on the Vulnerable Area Approach,” *Journal of Mechanical Science and Technology*, vol. 26, no. 7, pp. 2171–2181, 2012.

[327] Ö. Kişi and E. Uncuoglu, “Comparison of three back-propagation training algorithms for two case studies,” *Indian journal of engineering & materials sciences*, vol. 12, no. 5, pp. 434–442, 2005.

[328] J. T. Klopcic and H. L. Reed, "Historical Perspectives on Vulnerability/Lethality Analysis," U.S. Army Research Laboratory, Special Report ARL-SR-90, 1999.

[329] J. T. Klopcic, "The Vulnerability/Lethality Taxonomy as a General Analytical Procedure," U.S. Army Research Laboratory, Technical Report ARL-TR-1944, May 1999.

[330] T. Kobayashi and D. L. Simon, "A Hybrid Neural Network-Genetic Algorithm Technique for Aircraft Engine Performance Diagnostics," U.S. Army Research Laboratory, Technical Report ARL-TR-1266, July 2001.

[331] P. Koehn, "Combining genetic algorithms and neural networks: The encoding problem," Master's thesis, The University of Tennesse, Knoxville, 1994.

[332] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *IJCAI*, vol. 14, no. 2, 1995, pp. 1137–1145.

[333] S. Kolassa and R. Martin, "Percentage errors can ruin your day (and rolling the dice shows how)." *Foresight: The International Journal of Applied Forecasting*, no. 23, 2011.

[334] P. G. Korning, "Training neural networks by means of genetic algorithms working on very long chromosomes," *International Journal of Neural Systems*, vol. 6, no. 03, pp. 299–316, 1995.

[335] M. J. Koslow, "Ballistic Flash Characterization: Penetration and Back-Face Flash," Master's thesis, Air Force Institute of Technology, March 2012.

[336] K. Koutroumbas and Y. Bakopoulos, "On the Approximation Capabilities of Hard Limiter Feedforward Neural Networks," in *Proceedings of the 6$^{th}$ Hellenic conference on Artificial Intelligence: theories, models and applications*, ser. SETN'10. Springer-Verlag, 2010, pp. 163–172.

[337] V. V. Kovalishyn, I. V. Tetko, A. I. Luik, V. V. Kholodovych, A. E. P. Villa, and D. J. Livingstone, "Neural Network Studies. 3. Variable Selection in the Cascade-Correlation Learning Architecture," *Journal of Chemical Information and Computer Sciences*, vol. 38, pp. 651–659, 1998.

[338] V. Kucher, “Two-Dimensional Computations of Obliquity Effects on the Penetration Process,” Ballistic Research Laboratory, Technical Report ARBRL-TR-02064, May 1978.

[339] P. S. Kulkarni and J. Bakal, “Survey on data cleaning,” *structure*, vol. 3, 2014.

[340] J. P. Lambert and G. H. Jonas, “Towards Standardization in Terminal Ballistics Testing: Velocity Representation,” Ballistics Research Laboratory, Report BRL-R-1852, January 1976.

[341] J. P. Lambert and B. E. Ringers, “Standardization of Terminal Ballistics Testing, Data Storage and Retrieval,” Ballistic Research Laboratory, Technical Report ARBRL-TR-02066, May 1978.

[342] J. P. Lambert, “The Terminal Ballistics of Certain 65 Gram Long Rod Penetrators Impacting Steel Armor Plate,” Ballistics Research Laboratory, Technical Report ARBRL-TR-02072, May 1978.

[343] S. Lampert, J. R., and B. Lehmann, “Penetration Efficiency of Tungsten Penetrators into Glass Fiber Reinforced Resin/Steel Composites as a Function of Aspect Ratio and Impact Velocity,” in *Proceedings 19th International Symposium of Ballistics 7–11 May 2001*, I. R. Crewther, Ed., vol. 3, May 2001, pp. 1313–1319.

[344] B. Lan and H. Wen, “Alekseevskii-tate revisited: An extension to the modified hydrodynamic theory of long rod penetration,” *Science China Technological Sciences*, vol. 53, no. 5, pp. 1364–1373, 2010.

[345] W. Lanz and W. Odermatt, “Penetration Limits of Conventional Large Caliber Anti-tank Guns/Kinetic Energy Projectiles,” in *Proceedings of the 13th International Symposium of Ballistics*, vol. 3, 1992, pp. 225–233.

[346] ——, “Minimum Impact Energy For KE-Penetrators in RHA-Targets,” in *Proceedings of the European Forum on Ballistics of Projectiles*, M. Giraud and V. Fleck, Eds. ISL, French-German Research Institute of Saint-Louis, April 2000, pp. 349–365.

[347] W. Lanz, W. Odermatt, and D. G. Weihrauch, “Kinetic energy projectiles: Development history, state of the art, trends,” in *Proceedings 19th International Symposium of Ballistics 7–11 May 2001*, 2001.

[348] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, “Exploring strategies for training deep neural networks,” *Journal of Machine Learning Research*, vol. 10, no. Jan, pp. 1–40, 2009.

[349] S. Lawrence, C. L. Giles, and S. Fong, “Natural language grammatical inference with recurrent neural networks,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 1, pp. 126–140, 2000.

[350] P. Lay and S. Lüttringhaus-Kappel, “Transforming xml schemas into java swing guis.” *GI Jahrestagung (1)*, vol. 50, pp. 271–276, 2004.

[351] P. Leahy, G. Kiely, and G. Corcoran, “Structural optimisation and input selection of an artificial neural network for river level prediction,” *Journal of Hydrology*, vol. 355, no. 1, pp. 192–201, 2008.

[352] B. Leavy, C. Krauthauser, J. Houskamp, and J. LaSalvia, “Fundamental Investigation of High-Velocity Impact of Ductile Projectiles on Confined Ceramic Targets,” in *Proceedings of the 25th Army Science Conference*, Orlando, FL, November 2006.

[353] Y. LeCun, L. Bottou, G. Orr, and K. Müller, “Efficient Backprop,” in *Neural Networks—Tricks of the Trade*, ser. Lecture Notes in Computer Science, G. Orr and K. Mller, Eds. Springer, 1998, vol. 1524, pp. 5–50. [Online]. Available: http://citeseer.ist.psu.edu/lecun98efficient.html

[354] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[355] H. Ledbetter and R. P. Reed, “Elastic properties of metals and alloys, i. iron, nickel, and iron-nickel alloys,” *Journal of Physical and Chemical Reference Data*, vol. 2, no. 3, pp. 531–618, 1973.

[356] H. K. Lee, “Model selection for neural network classification,” *Journal of classification*, vol. 18, no. 2, pp. 227–243, 2001.

[357] M. Lee, “Analysis of Jacketed Rod Penetration,” *International Journal of Impact Engineering*, vol. 24, pp. 891–905, 2000.

[358] M. Lee and S. J. Bless, “A discreet impact model for effect of yaw angle on penetration by rod projectiles,” in *The tenth American Physical Society topical conference on shock compression of condensed matter*, vol. 429, no. 1. AIP Publishing, 1998, pp. 929–932.

[359] M. Lee and M. J. Normandia, “Successive Impact of Segmented Rods at High–Velocity,” *Korean Society of Mechanical Engineers International Journal*, vol. 13, no. 4, pp. 312–320, 1999.

[360] W. Lee, H.-J. Lee, and H. Shin, “Ricochet of a Tungsten Heavy Alloy Long-Rod Projectile from Deformable Steel Plates,” *Journal of Physics D: Applied Physics*, vol. 35, pp. 2676–2686, 2002.

[361] J.-F. Legendre, M. Giraud, and M. Henner, “Ram-Accelerator: A New Hypervelocity Launcher for Ballistic Studies,” *International Journal of Impact Engineering*, vol. 23, pp. 533–545, 1999.

[362] P. Lehmann, V. Schirm, H. Peter, and J. Wey, “Rail Launchers to Reach Hypervelocity,” *International Journal of Impact Engineering*, vol. 17, pp. 509–515, 1995.

[363] H. F. Lehr, E. Wollman, and G. Koerber, “Experiments with Jacketed Rods of High Fineness Ratios,” *International Journal of Impact Engineering*, vol. 15, pp. 517–526, 1995.

[364] H. F. Lehr, E. Wollmann, W. Lanz, and K. Sterzelmeier, “On the Behavior of Long-Rod Penetrators Undergoing Lateral Accelerations,” in *Proceedings 19$^{th}$ International Symposium of Ballistics 7–11 May 2001*, 2001, pp. 1141–1150.

[365] F. Leisch, L. C. Jain, K. Hornik *et al.*, “Cross-validation with active pattern selection for neural-network classifiers,” *IEEE Transactions on Neural Networks*, vol. 9, no. 1, pp. 35–41, 1998.

[366] A. Lendasse, V. Wertz, and M. Verleysen, "Model selection with cross-validations and bootstraps - application to time series prediction with RBFN models," in *Artificial Neural Networks and Neural Information Processing ICANN/ICONIP 2003*. Springer, 2003, pp. 573–580.

[367] W. Leonard and L. Magness, "The terminal ballistic performance of microstructural oriented tungsten heavy alloy penetrators," Army Research Laboratory, Technical Report ARL-TR-881, October 1995.

[368] W. Leonard, "The Effect of Nose Shape on Depleted Uranium (DU) Long-Rod Penetrators," U.S. Army Research Laboratory, Technical Report ARL-TR-1505, September 1997.

[369] P. Leray, P. Gallinari, P. Gallinari, and P. Gallinari, "Feature Selection with Neural Networks," *Behaviormetrika*, vol. 26, no. 1, pp. 145–166, 1999.

[370] E. Lidén, J. Ottoson, and L. Holmberg, "WHA Long Rods Penetrating Stationary and Moving Oblique Steel Plates," in *Proceedings of the 16th International Symposium on Ballistics*, 1996.

[371] R. L. Lindsey, "Function Prediction Using Recurrent Neural Networks," Ph.D. dissertation, Air Force Institute of Technology, December 1991.

[372] Y. Ling, T. Mullen, and X. Lin, "Analysis of optimal thread pool size," *ACM SIGOPS Operating Systems Review*, vol. 34, no. 2, pp. 42–55, 2000.

[373] P. G. J. Lisboa, Ed., *Neural Networks: Current Applications*. London: Chapman & Hall, 1992.

[374] R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data*. Wiley, 1987.

[375] R. J. Little and N. Zhang, "Subsample ignorable likelihood for regression analysis with missing data," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 60, no. 4, pp. 591–605, 2011.

[376] R. J. A. Little, “A Test of Missing Completely at Random for Multivariate Data with Missing Values,” *journal of the American Statistical Association*, vol. 83, no. 404, pp. 1198–1202, December 1988.

[377] R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data.* New York, NY, USA: John Wiley & Sons, Inc., 1986.

[378] D. L. Littlefield, C. E. Anderson Jr., Y. Partom, and S. Bless, “The Penetration of Steel Targets Finite in Radial Extent,” *International Journal of Impact Engineering*, vol. 19, no. 1, pp. 49–62, 1997.

[379] D. L. Littlefield and W. Reinecke, “Penetration mechanics of extending hemicylindrical rods,” in *Proceedings of the 19th International Symposium of Ballistics*, E. Baker and D. Templeton, Eds., vol. 2. Lancaster, PA.: DEStech Publications, September 2001, pp. 1522–1532.

[380] Y. Liu, J. Cheng, C. Yan, X. Wu, and F. Chen, “Research on the matthews correlation coefficients metrics of personalized recommendation algorithm evaluation,” *International Journal of Hybrid Information Technology*, vol. 8, no. 1, pp. 163–172, 2015.

[381] Y. Liu, “Neural network model selection using asymptotic jackknife estimator and cross-validation method,” in *Advances in Neural Information Processing Systems*, 1992, pp. 599–606.

[382] ——, “Create stable neural networks by cross-validation,” in *Neural Networks, 2006. IJCNN’06. International Joint Conference on.* IEEE, 2006, pp. 3925–3928.

[383] A. Loureiro, L. Torgo, and C. Soares, “Outlier detection using Clustering methods: A data cleaning Application,” in *Proceedings of KDNet Symposium on Knowledge-based systems for the Public Sector*, 2004.

[384] V. K. Luk and A. J. Piekutowski, “An Analytical Model on Penetration of Eroding Long Rods into Metallic Targets,” *International Journal of Impact Engineering*, vol. 11, no. 3, pp. 323–340, 1991.

[385] P. Lundberg, L. Westerling, and B. Lundberg, “Influence of Scale on the Penetration of Tungsten Rods into Steel-Backed Alumina Targets,” *International Journal of Impact Engineering*, vol. 18, no. 4, pp. 403–416, 1996.

[386] Luttwak, “Oblique and Yawed Rod Penetration,” in *Proceedings 5th International Symposium on Behaviour of Dense Media under High Dynamic Pressures 23–27 June 2003*, 2003.

[387] G. Luttwak, “Yawed Rod Penetration,” in *Proceedings of 20th International Symposium of Ballistics*, 2002.

[388] N. J. Lynch, “Constant Kinetic Energy Impacts of Scale Size KE Projectiles at Ordance and Hypervelocity,” *International Journal of Impact Engineering*, vol. 23, pp. 573–584, 1999.

[389] N. J. Lynch, R. Subramanian, C. Brissenden, and P. Shears, “Terminal Ballistic Performance of Novel KE Projectiles,” in *Proceedings 15th International Symposium of Ballistics 21–24 May 1995*, 1995, pp. 35–42.

[390] N. J. Lynch, R. Subramanian, S. Brown, and J. Alston, “The Influence of Penetrator Geometry and Impact Velocity on the Formation of Crater Volume in Semi-Infinite Targets,” in *Proceedings 19th International Symposium of Ballistics 7–11 May 2001*, 2001.

[391] C. MacLeod and G. M. Maxwell, “Incremental evolution in anns: Neural nets which grow,” *Artificial Intelligence Review*, vol. 16, no. 3, pp. 201–224, 2001.

[392] V. Madhu, T. Balakrishna Bhat, and N. K. Gupta, “Normal and Oblique Impacts of Hard Projectile on Single and Layered Plates-An Experimental Study,” *Defense Science Journal*, vol. 53, no. 2, pp. 147–156, April 2003.

[393] V. Madhu, K. Ramanjaneyulu, T. Balakrishna Bhat, and N. K. Gupta, “An Experimental Study of Penetration Resistance of Ceramic Armour Subjected to Projectile Impact,” *International Journal of Impact Engineering*, vol. 32, pp. 337–350, 2005.

[394] L. Magness and W. Leonard, "Scaling Issues for Kinetic Energy Penetrators," in *Proceedings of the 14$^{th}$ International Symposium on Ballistics*, 1993.

[395] L. Magness, "Antiarmor Kinetic Energy Penetrators," in *Advanced Ballistics Science and Engineering*, B. Burns, Ed. U.S. Army Research Laboratory, November 2008, no. ARL-SR-168, special report Lethality. Antiarmor Kinetic Energy Projectiles, pp. 386–416.

[396] L. S. Magness and T. Farrand, "Deformation Behavior and Its Relationship to the Penetration Performance of High-Density KE Penetrator Materials," in *Army Science Conference Proceedings*, vol. 2, June 1990, pp. 465–479.

[397] L. Magness and D. Scheffler, "The influence of penetrator material and projectile nose shape on the onset of penetrator deformation and erosion," in *Proceedings of the second Australian Conference on Applied Mechanics*, 1999.

[398] O. Maimon and L. Rokach, Eds., *The Data Mining and Knowledge Discovery Handbook*. Springer, 2005.

[399] J. I. Maletic and A. Marcus, "Data cleansing: Beyond integrity analysis." in *IQ*. Citeseer, 2000, pp. 200–209.

[400] ——, "Data Cleansing - A Prelude to Knowledge Discovery," in *The Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, Eds. Springer, 2005, pp. 21–36.

[401] R. S. Mariano, "Testing forecast accuracy," *A companion to economic forecasting*, pp. 284–298, 2002.

[402] M. K. Markey, G. D. Tourassi, M. Margolis, and D. M. DeLong, "Impact of missing data in evaluating artifical neural networks trained on complete data," *Computers in Biology and Medicine*, vol. 36, pp. 516–525, 2006.

[403] Massachusetts Institute of Technology Lexington Lincoln Laboratory, "DARPA Neural Network Study," U.S. Defense Advanced Research Projects Agency/Tactical Technology Office (DARPA/TTO), Technical Report ESD-TR-88-311, March 1989.

[404] M. Matteucci and D. Spadoni, “Evolutionary Learning of Rich Neural Networks in the Bayesian Model Selection Framework,” *International Journal of Applied Math and Computer Science*, vol. 14, no. 3, pp. 423–440, 2004.

[405] W. S. McCulloch and W. Pitts, “A Logical Calculus of the Ideas Immanent in Nervous Activity,” *Bulletin of Mathematical Biology*, vol. 5, pp. 115–133, 1943.

[406] M. McDermeit, R. Funk, and M. Dennis, “Data Cleaning and Replacement of Missing Values.”

[407] J. R. McDonnell and D. Waagen, “Determining Neural Network Connectivity Using Evolutionary Programming,” in *Conference Record of The Twenty-Sixth Asilomar Conference on Signals, Systems and Computers*, vol. 2, October 1992, pp. 786–790.

[408] ——, “Determining Neural Network Hidden Layer Size Using Evolutionary Programming,” in *World Congress on Neural Networks: 1993 International Neural Network Society Annual Meeting*, vol. 3, 1993, pp. 564–567.

[409] M. R. McHenry, Y. Choo, and D. L. Orphal, “Numerical Simulations of Low L/D Rod Aluminum Into Aluminum Impacts Compared to the Tate Cratering Model,” *International Journal of Impact Engineering*, vol. 23, pp. 621–628, 1999.

[410] M. McInerney and A. P. Dhawan, “Use of genetic algorithms with backpropagation in training of feedforward neural networks,” in *IEEE International Conference on Neural Networks*, vol. 1. IEEE, 1993, pp. 203–208.

[411] Y. Me-Bar, “A Method for Scaling Ballistic Penetration Phenomena,” *International Journal of Impact Engineering*, vol. 19, no. 9–10, pp. 821–829, 1997.

[412] J. J. Misey, “Analysis of Long Rod Penetration at Hypervelocity Impact,” Ballistic Research Laboratory, Technical Report BRL-R-1982, April 1977.

[413] N. Mohamad, F. Zaini, A. Johari, I. Yassin, and A. Zabidi, “Comparison between levenberg-marquardt and scaled conjugate gradient training algorithms for breast cancer diagnosis using mlp,” in *Signal Processing and Its Applications (CSPA), 2010 6th International Colloquium on*. IEEE, 2010, pp. 1–7.

[414] S. A. Mojarad, S. S. Dlay, W. L. Woo, and G. Sherbet, “Breast Cancer prediction and cross validation using multilayer perceptron neural networks,” in *Communication Systems Networks and Digital Signal Processing (CSNDSP), 2010 7$^{th}$ International Symposium on.* IEEE, 2010, pp. 760–764.

[415] M. F. Møller, “Exact calculation of the product of the hessian matrix of feed-forward network error functions and a vector in 0 (n) time,” *DAIMI Report Series*, vol. 22, no. 432, 1993.

[416] D. J. Montana and L. Davis, “Training feedforward neural networks using genetic algorithms,” in *Proceedings of the 11$^{th}$ international joint conference on Artificial intelligence - Volume 1*, ser. IJCAI’89. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1989, pp. 762–767. [Online]. Available: http://dl.acm.org/citation.cfm?id=1623755.1623876

[417] B. L. Morris and C. E. Anderson Jr., “The Ballistic Performance of Confined Ceramic Tiles,” in *Proceedings of the 1991 TACOM Combat Vehicle Survivability Symposium*, 1991, pp. 235–244.

[418] R. W. Morrison and K. A. De Jong, “Measurement of population diversity,” in *Artificial Evolution.* Springer, 2002, pp. 31–41.

[419] S. Moulsdale, “MUVES-S2 Study: Comparison of Vulnerability Results When Varying Grid Cells Size and Number of Views,” U.S. Army Research Laboratory, Technical Report ARL-TR-5835, June 2012.

[420] R. Muldoon, “MTL Armor Data Base Program,” U.S. Army Materials Technology Laboratory, Tech. Rep., 1991.

[421] H. Müller and J.-C. Freytag, *Problems, methods, and challenges in comprehensive data cleansing.* Professoren des Inst. Für Informatik, 2005.

[422] N. Murata, S. Yoshizawa, and S.-i. Amari, “Network information criterion-determining the number of hidden units for an artificial neural network model,” *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 865–872, 1994.

[423] K. A. Myers, “Lethal Area Description,” Ballistic Research Laboratories, Technical Note BRL-TN-1510, July 1963.

[424] N. M. Nawi, N. A. Hamid, R. Ransing, R. Ghazali, and M. N. M. Salleh, “Enhancing back propagation neural network algorithm with adaptive gain on classification problems,” *networks*, vol. 4, no. 2, 2011.

[425] J. Nazari and O. K. Ersoy, “Implementation of Back-Propagation Neural Networks with MatLab,” Purdue University Electrical and Computer Engineering, Technical Report TR-RR 92-39, 1992.

[426] M. Neale, S. Boker, G. Xie, and H. Maes, “Mx: Statistical modeling . richmond, va: Department of psychiatry,” *Virginia Institute for Psychiatric and Behavior Genetics, Virginia Commonwealth University*, 2003.

[427] P. E. Nebolsine, N. D. Humer, N. F. Harmon, and J. R. Baker, “Statistical Analysis of NRL 1964-1969 Hypervelocity Rod-Plate Impact Data and Comparison to Recent Data,” *International Journal of Impact Engineering*, vol. 23, pp. 639–649, 1999.

[428] V. F. Nesterenko, S. S. Indrakanti, S. Brar, and G. YaBei, “Long Rod Penetration Test of Hot Isostatically Pressed Ti-Based Targets,” in *American Institute of Physics Conference Proceedings*, vol. 505, 2000, pp. 419–422.

[429] P. H. Netherwood Jr., “Rate of Penetration Measurements,” Ballistic Research Laboratory, Memorandum Report ARBRL-MR-02978, December 1979.

[430] M. J. Normandia, “Eroded Length Model for Yawed Penetrators Impacting Finite Thickness Targets at Normal and Oblique Incidence,” *International Journal of Impact Engineering*, vol. 23, pp. 663–674, 1999.

[431] D. Norris, W. McMaster, and M. Wilkins, “Long-rod projectiles against oblique targets: analysis and design recommendations,” California Univ., Livermore (USA). Lawrence Livermore Lab., Tech. Rep., 1976.

[432] R. K. Nowicki, B. Nowak, J. T. Starczewski, K. Cpalka *et al.*, “The learning of neuro-fuzzy approximator with fuzzy rough sets in case of missing features,” in *Neural Networks (IJCNN), 2014 International Joint Conference on.* IEEE, 2014, pp. 3759–3766.

[433] G. Ochoa, I. Harvey, and H. Buxton, “Optimal Mutation Rates and Selection Pressure in Genetic Algorithms.” in *GECCO.* Citeseer, 2000, pp. 315–322.

[434] W. Odermatt. (2014) Long rod penetrators - perforation equation. [Online]. Available: http://www.longrods.ch/perfeq.php

[435] P. Oliveira, F. Rodrigues, and H. Pedro, “A formal definition of data quality problems,” in *Proceedings of International Conference on Information Quality*, 2005.

[436] ——, “A taxonomy of data quality problems,” in *Proceedings of 2nd International Workshop on Data and Information Quality*, 2005.

[437] T. Onoda, “Neural network information criterion for the optimal number of hidden units,” in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 1. IEEE, 1995, pp. 275–280.

[438] D. L. Orphal and C. E. Anderson Jr., “The Dependence of Penetration Velocity on Impact Velocity,” *International Journal of Impact Engineering*, vol. 33, pp. 564–554, 2006.

[439] D. L. Orphal and R. R. Franzen, “Penetration Mechanics and Performance of Segmented Rods Against Metal Targets,” *International Journal of Impact Engineering*, vol. 10, no. 1–4, pp. 427–438, 1990.

[440] ——, “Penetration of Confined Silicon Carbide Targets by Tungsten Long Rods at Impacts Velocities from 1.5 to 4.6 km/s,” *International Journal of Impact Engineering*, vol. 19, no. 1, pp. 1–13, 1997.

[441] D. L. Orphal, R. R. Franzen, A. C. Charters, T. L. Menna, and A. J. Piekutowski, “Penetrations of Confined Boron Carbide Targets by Tungsten Long Rods at Impact Velocities From 1.5 to 5.0 km/s,” *International Journal of Impact Engineering*, vol. 19, no. 1, pp. 15–29, 1997.

[442] D. L. Orphal, R. R. Franzen, A. J. Piekutowski, and M. J. Forrestal, “Penetration of Confined Aluminum Nitrade Targets by Tungsten Long Rods at 1.5–4.5 km/s,” *International Journal of Impact Engineering*, vol. 18, no. 4, pp. 355–368, 1996.

[443] D. Orphal, C. Anderson, R. Franzen, J. Walker, P. Schneidewind, and M. Majerus, “Impact and penetration by l/d 1 projectiles,” *International Journal of Impact Engineering*, vol. 14, no. 1-4, pp. 551–560, 1993.

[444] J. W. Osborne and A. Overbay, *Best practices in data cleaning.* Sage, 2012.

[445] G. Panchal, A. Ganatra, Y.P.Kosta, and D. Panchal, “Article: Searching Most Efficient Neural Network Architecture Using Akaike’s Information Criterion (AIC),” *International Journal of Computer Applications*, vol. 1, no. 5, pp. 41–44, February 2010, Published By Foundation of Computer Science.

[446] R. Parekh, J. Yang, and V. Honavar, “Constructive neural-network learning algorithms for pattern classification,” *Neural Networks, IEEE Transactions on*, vol. 11, no. 2, pp. 436–451, 2000.

[447] L. E. Parker, “Notes on multilayer, feedforward neural networks,” 2007. [Online]. Available: http://web.eecs.utk.edu

[448] Y. Partom, “On the Hydrodynamic Limit of Long Rod Penetration,” *International Journal of Impact Engineering*, vol. 20, pp. 617–625, 1997.

[449] D. V. Patil and R. S. Bichkar, “Multiple Imputation of Missing Data with Genetic Algorithm based Techniques,” *International Journal of Computer Applications*, vol. 2, pp. 74–79, 2010.

[450] C. C. Peck, A. P. Dhawan, and C. M. Meyer, "Genetic algorithm based input selection for a neural network function approximator with applications to ssme health monitoring," in *Neural Networks, 1993., IEEE International Conference on.* IEEE, 1993, pp. 1115–1122.

[451] B. A. Pedersen, S. J. Bless, and J. U. Cazamias, "Hypervelocity Jacketed Penetrators," *International Journal of Impact Engineering*, vol. 26, pp. 603–611, 2001.

[452] H. Peng, H. Yan, and W. Zhang, "The connection between cross-validation and akaike information criterion in a semiparametric family," *Journal of Nonparametric Statistics*, vol. 25, no. 2, pp. 475–485, 2013.

[453] K. Pepperdine, "Tuning the Size of Your Thread Pool," 2013. [Online]. Available: http://www.infoq.com

[454] E. Perez, "Experimental and Theoretical Study on the Penetration of Semi-Infinite Metal Targets by Great-Length Metal Projectiles with a Velocity Higher than 2000 m/s," *Science et Techniques de l'Armement*, vol. 56, pp. 1–155, 1982.

[455] S. Perez, "Apply genetic algorithm to the learning phase of a neural network," 2008. [Online]. Available: http://www.ics.uci.edu/~dramanan/teaching/ics273a_winter08/projects/sperez1_GANN.pdf

[456] G. J. J. M. Peskes and W. Lanz, "Evaluation of Replica Scale Jacketed Penetrators for Tank Ammunition," in *Proceedings 19$^{th}$ International Symposium of Ballistics 7–11 May 2001*, I. R. Crewther, Ed., vol. 3, May 2001, pp. 1223–1229.

[457] E. Pesonen, M. Eskelinen, and M. Juhola, "Treatment of missing data values in a neural network based decision support system for acute abdominal pain," *Artificial Intelligence in Medicine*, vol. 13, no. 3, pp. 139–146, 1998.

[458] H. Peyre, A. Leplège, and J. Coste, "Missing data methods for dealing with missing items in quality of life questionnaires. A comparison by simulation of personal mean score, full information maximum likelihood, multiple imputation, and hot deck techniques applied to the SF-36 in the French 2003 decennial health survey," *Quality of Life Research*, vol. 20, no. 2, pp. 287–300, 2011.

[459] A. J. Piekutowski, M. J. Forrestal, K. L. Poormon, and T. L. Warren, "Perforation of Aluminum Plates with Ogive-Nose Steel Rods at Normal and Oblique Impacts," *International Journal of Impact Engineering*, vol. 18, no. 7–8, pp. 877–887, 1996.

[460] ——, "Penetration of 6061-T6511 Aluminum Targets by Ogive-Nose Steel Projectiles with Striking Velocities Between 0.5 and 3.0 km/s," *International Journal of Impact Engineering*, vol. 23, pp. 723–734, 1999.

[461] A. J. Piekutowski and C. E. Anderson Jr., "Post-Perforation Characteristics of Yawed Long Rods," in *Proceedings of 14th International Symposium of Ballistics*, September 1993.

[462] T. D. Pigott, "A Review of Methods for Missing Data," *Educational Research and Evaluation*, vol. 7, no. 4, pp. 353–383, 2001.

[463] P. Plostins, "Launch dynamics of apfsds ammunition," US Army Ballistic Research Laboratory, Technical Report BRL-TR-2595, 1984.

[464] D. M. W. Powers, "Evaluation: From precision, recall and f-factor to roc, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

[465] S. S. Prabhune and S. Sathe, "Reconstruction of a Complete Dataset from an Incomplete Dataset by Expectation Maximization Technique: Some Results," *International Journal of Computer Science and Network Security*, vol. 10, no. 11, pp. 141–144, 2010.

[466] L. Prechelt, "Automatic early stopping using cross validation: quantifying the criteria," *Neural Networks*, vol. 11, no. 4, pp. 761–767, 1998.

[467] ——, “Early stopping - But when?” in *Neural Networks: Tricks of the trade.* Springer, 2012, pp. 53–67.

[468] S. Price and J. Robertson, “Best business practices for conducting muves-s2 vulnerability/lethality (V/L) analyses,” U.S. Army Research Laboratory, Technical Report ARL-TR-5409, 2010.

[469] W. G. Proud, N. J. Lynch, A. Marsh, and J. E. Field, “Instrumented Small Scale Rod Penetration Studies: The Effect of Pitch,” in *Proceedings 19th International Symposium of Ballistics 7–11 May 2001*, 2001.

[470] J. S. Przemieniecki, *Mathematical Methods in Defense Analyses*, 3rd ed. AIAA (American Institute of Aeronautics & Astronautics), 2000.

[471] M. J. Puma, R. B. Olsen, S. H. Bell, and C. Price, “What to Do when Data Are Missing in Group Randomized Controlled Trials. NCEE 2009-0049.” *National Center for Education Evaluation and Regional Assistance*, no. NCEE 2009-0049, October 2009.

[472] Y. Qian and K. Zhang, “The Role of Visualization in Effective Data Cleaning,” in *Proceedings of the 2005 ACM symposium on Applied computing*, 2005, pp. 1239–1243.

[473] X. Quan, R. A. Clegg, M. S. Cowler, N. K. Birnbaum, and C. J. Hayhurst, “Numerical Simulation of Long Rods Impacting Silicon Carbide Targets Using JH-1 Model,” *International Journal of Impact Engineering*, vol. 33, pp. 634–644, 2006.

[474] M. Raftenberg, “A shear banding model for penetration calculations,” Army Research Laboratory, Technical Report ARL-TR-2221, April 2000.

[475] M. N. Raftenberg and E. W. Kennedy, “Perforation of Rolled-Homogeneous-Armor Plates by Tungsten Rods of Small Length-to-Diameter Ratios,” U.S. Army Research Laboratory, Technical Report ARL-TR-739, 1995.

[476] ——, “Steel Plate Perforation by Tungsten Rods of Small Length-to-Diameter Ratios,” in *Proceedings from the 15th International Symposium on Ballistics*, M. Mayseless and S. R. Bodner, Eds., vol. 1, 1995, pp. 315–321.

[477] M. N. Raftenberg, “Strength, Equation of State, and Drag Flare Effects on Steel Plate Perforation by Small Length-to-Diameter Ratio Tungsten Rods,” U.S. Army Research Laboratory, Technical Report ARL-TR-1052, April 1996.

[478] ——, “Damage and Strength Effects in Steel Plates Perforated by Tungsten Rods,” U.S. Army Research Laboratory, Technical Report ARL-TR-1578, December 1997.

[479] E. Rahm and H. H. Do, “Data cleaning: Problems and current approaches,” *IEEE Data Eng. Bull.*, vol. 23, no. 4, pp. 3–13, 2000.

[480] A. M. Rajendran, “Penetration of Tungsten Alloy Rods Into Shallow-Cavity Steel Targets,” *International Journal of Impact Engineering*, vol. 21, no. 6, pp. 451–460, 1998.

[481] A. M. Rajendran and P. Woolsey, “Penetration of Tungsten Alloy Rods Into Shallow-Cavity Steel Targets,” U.S. Army Research Laboratory, Technical Report ARL-TR-216, August 1993.

[482] G. B. Ramaiah, R. Y. Chennaiah, and G. K. Satyanarayanarao, “Investigation and modeling on protective textiles using artificial neural networks for defense applications,” *Materials Science and Engineering: B*, vol. 168, no. 1-3, pp. 100–105, 2010.

[483] A. Ranganathan, “The levenberg-marquardt algorithm,” *Tutoral on LM Algorithm*, pp. 1–5, 2004.

[484] E. J. Rapacki Jr., K. Frank, R. B. Leavy, M. J. Keele, and J. J. Prifti, “Armor Steel Hardness Influence on Kinetic Energy Penetration,” in *Proceedings 15th International Symposium of Ballistics 21–24 May 1995*, 1995, pp. 323–330.

[485] N. M. Razali and J. Geraghty, “Genetic Algorithm Performance with Different Selection Strategies in Solving TSP,” in *Proceedings of the World Congress on Engineering*, vol. 2, 2011.

[486] J. E. Reaugh, A. C. Holt, M. L. Wilkins, B. J. Cunningham, B. L. Hord, and A. S. Kusubov, “Impact Studies of Five Ceramic Materials and Pyrex,” *International Journal of Impact Engineering*, vol. 23, pp. 771–782, 1999.

[487] R. Recht and T. Ipson, “The dynamics of terminal ballistics. ballistic evaluation procedures for armored grille designs,” Denver Research Institute, Tech. Rep., 1962.

[488] R. F. Recht, “Ballistic perforation dynamics of armor-piercing projectiles,” Naval Weapons Center, Technical Report NWC TP 4532, 1967.

[489] R. Reed, “Pruning algorithms-a survey,” *IEEE Transactions on Neural Networks*, vol. 4, no. 5, pp. 740–747, 1993.

[490] P. Refaeilzadeh, L. Tang, and H. Liu, “Cross-Validation,” in *Encyclopedia of Database Systems*, L. LIU and M. ZSU, Eds. Springer US, 2009, pp. 532–538.

[491] C. G. L. Reinhart, “A fortran based learning system using multilayer back-propagation neural network techniques,” Master’s thesis, Department of the Air Force Air University, 1994.

[492] C. C. Renahan, W. S. Andrews, and K. M. Jaansalu, “Modeling the Depth of Penetration of Ceramic Targets Using Artificial Neural Networks,” in *Proceedings for the $23^{rd}$ International Symposium of Ballistics*, 2007, pp. 1487–1494. [Online]. Available: http://books.google.com/books?id=QOWxpwAACAAJ

[493] D. Riha, B. Thacker, J. Pleming, J. Walker, S. Mullin, C. Weiss, E. Rodriguez, and P. Leslie, “Verification and validation for a penetration model using a deterministic and probabilistic design tool,” *International Journal of Impact Engineering*, vol. 33, no. 1, pp. 681–690, 2006.

[494] B. E. Ringers, “Numerical Simulation of Yawed Rod Impacts,” Ballistic Research Laboratory, Memorandum Report ARBRL-MR-03155, February 1982.

[495] B. D. Ripley, “Pattern recognition via neural networks,” 1996. [Online]. Available: http://citeseerx.ist.psu.edu

[496] D. Rittel and G. Weisbrod, “Dynamic fracture of tungsten base heavy alloys,” *International journal of fracture*, vol. 112, no. 1, pp. 87–98, 2001.

[497] I. Rivals and L. Personnaz, “On cross validation for model selection,” *Neural Computation*, vol. 11, no. 4, pp. 863–870, 1999.

[498] J. M. Robins, A. Rotnitzky, and L. P. Zhao, “Analysis of Semiparametric Regression Models for Repeated Outcomes in the Presence of Missing Data,” *Journal of the American Statistical Association*, vol. 90, no. 429, pp. 106–121, March 1995.

[499] A. Roche, “EM algorithm and variants: an informal tutorial,” *ArXiv e-prints*, May 2011.

[500] J. D. Rodríguez, A. Perez, and J. A. Lozano, “Sensitivity analysis of k-fold cross validation in prediction error estimation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 3, pp. 569–575, 2010.

[501] I. V. Roisman, K. Weber, A. L. Yarin, V. Hohler, and M. B. Rubin, “Oblique Penetration of a Rigid Projectile into a Thick Elastic–Plastic Target: Theory and Experiment,” *International Journal of Impact Engineering*, vol. 22, pp. 707–726, 1999.

[502] I. V. Roisman, A. L. Yarin, and M. B. Rubin, “Oblique Penetration of a Rigid Projectile Into an Elastic-Plastic Target,” *International Journal of Impact Engineering*, vol. 19, no. 9–10, pp. 769–795, 1997.

[503] R. Rojas, *Second Order Backpropagation: Efficient Computation of the Hessian Matrix for Neural Networks.* International Computer Science Institute, 1993.

[504] Z. Rosenberg, Y. Ashuach, and E. Dekel, “More on the ricochet of eroding long rods - validating the analytical model with 3D simulations,” *International Journal of Impact Engineering*, vol. 34, no. 5, pp. 942–957, 2007.

[505] Z. Rosenberg and E. Dekel, “A Computational Study of the Influence of Projectile Strength on the Performance of Long-Rod Penetrators,” *International Journal of Impact Engineering*, vol. 18, no. 6, pp. 671–677, 1996.

[506] ——, “A Computational Study of the Relations Between Material Properties of Long-Rod Penetrators and their Ballistic Performance,” *International Journal of Impact Engineering*, vol. 21, no. 4, pp. 283–296, 1998.

[507] ——, "On the Role of Nose Profile in Long-Rod Penetration," *International Journal of Impact Engineering*, vol. 22, pp. 551–557, 1999.

[508] ——, "Further Examination of Long Rod Penetration: The Role of Penetrator Strength at Hypervelocity Impacts," *International Journal of Impact Engineering*, vol. 24, pp. 85–102, 2000.

[509] ——, "Material Similarities in Long-Rod Penetration Mechanics," *International Journal of Impact Engineering*, vol. 25, pp. 361–372, 2001.

[510] ——, "More on the Secondary Penetration of Long Rods," *International Journal of Impact Engineering*, vol. 26, pp. 639–649, 2001.

[511] ——, "On the Role of Material Properties in the Terminal Ballistics of Long Rods," *International Journal of Impact Engineering*, vol. 30, pp. 835–851, 2004.

[512] ——, "On the Deep Penetration and Plate Perforation by Rigid Projectiles," *International Journal of Solids and Structures*, vol. 46, pp. 4169–4180, 2009.

[513] ——, "The Penetration of Rigid Long Rods – Revisted," *International Journal of Impact Engineering*, vol. 36, pp. 554–564, 2009.

[514] ——, *Terminal Ballistics*. Springer, 2012.

[515] Z. Rosenberg, E. Dekel, V. Hohler, A. J. Stilp, and K. Weber, "Hypervelocity Penetration of Tungsten Alloy Rods into Ceramic Tiles: Experiments and 2-D Simulations," *International Journal of Impact Engineering*, vol. 20, pp. 675–683, 1997.

[516] Z. Rosenberg and M. J. Forrestal, "Perforation of Aluminum Plates with Conical-Nosed Rods - Additional Data and Discussion," *Journal of Applied Mechanics*, vol. 55, pp. 236–238, 1988.

[517] Z. Rosenberg, R. Kreif, and E. Dekel, "A Note on the Geometric Scaling of Long-Rod Penetration," *International Journal of Impact Engineering*, vol. 19, no. 3, pp. 277–283, 1997.

[518] Z. Rosenberg, E. Marmor, and M. Mayseless, “On the Hydrodynamic Theory of Long-Rod Penetration,” *International Journal of Impact Engineering*, vol. 10, no. 1–4, pp. 483–486, 1990.

[519] Z. Rosenberg, Y. Partom, and Y. Yeshurun, “Determination of the hugoniot elastic limits of differently treated 2024 Al specimens,” *Journal of Physics D: Applied Physics*, vol. 15, no. 7, p. 1137, 1982.

[520] Z. Rosenberg, Y. Yeshurun, and M. Mayseless, “On the ricochet of long rod projectiles,” in *Proc. 11th Int. Symp. Ballistics*, vol. 501, 1989.

[521] G. A. Rovithakis, M. Maniadakis, and M. Zervakis, “A hybrid neural network/genetic algorithm approach to optimizing feature extraction for signal classification,” *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 34, no. 1, pp. 695–703, 2004.

[522] S. Roweis, “Gaussian identities,” *University of Toronto*, 1999.

[523] ——, “Levenberg-marquardt optimization,” *Notes, University Of Toronto*, 1996.

[524] ——, “Matrix identities,” *University of Toronto*, 1999.

[525] D. Roylance, “Mechanical properties of materials,” 2008. [Online]. Available: http://web.mit.edu/course/3/3.225/book.pdf

[526] P. Royston, “Multiple imputation of missing values,” *Stata Journal*, vol. 4, pp. 227–241, 2004.

[527] P. Royston and I. R. White, “Multiple Imputation by Chained Equations (MICE): Implementation in Stata,” *Journal of Statistical Software*, vol. 45, no. 4, December 2011.

[528] D. B. Rubin, “Inference and Missing Data,” *Biometrika*, vol. 63, no. 3, pp. 581–592, December 1976.

[529] N. L. Rupert and F. I. Grace, "Penetration of Semi-Infinite, Bi-Element Targets by Long Rod Penetrators," U.S. Army Research Laboratory, Technical Report ARL-TR-666, January 1995.

[530] A. Rusinek, J. A. Rodríguez-Martínez, R. Zaera, J. R. Klepaczko, A. Arias, and C. Sauvelet, "Experimental and Numerical Study on the Perforation Process of Mild Steel Sheets Subjected to Perpendicular Impact by Hemispherical Projectiles," *International Journal of Impact Engineering*, vol. 36, pp. 565–587, 2009.

[531] K. F. Ryan and R. J. Dowding, "Yield Properties of Tungsten and Tungsten Heavy Alloys," U.S. Army Research Laboratory, Technical Report ARL-TR-143, June 1993.

[532] W. S. Sarle, "Neural Network FAQ," 2002, Compilation of Usenet newsgroup comp.ai.neural-nets. [Online]. Available: ftp://ftp.sas.com/pub/neural/FAQ.html

[533] R. Saucier, "Simple Line-of-Sight Subtraction Method for KE Penetrators: Using the Semi-Empirical Lanz-Odermatt Function," 2003, Army Research Laboratory. MUVES 3 Methodology Research Board. KELOS Algorithm Description.

[534] SBWiki, "M829," August 2012. [Online]. Available: http://www.steelbeasts.com/sbwiki/index.php/Image:M829.jpg

[535] J. L. Schafer and J. W. Graham, "Missing Data: Our View of the State of the Art," *Psychological Methods*, vol. 7, no. 2, pp. 147–177, 2002.

[536] J. L. Schafer and M. K. Olsen, "Multiple imputation for multivariate missing-data problems: A data analyst's perspective," *Multivariate Behavioral Research*, vol. 33, no. 4, pp. 545–571, 1998.

[537] C. Schaffer, "Selecting a classification method by cross-validation," *Machine Learning*, vol. 13, no. 1, pp. 135–143, 1993.

[538] D. R. Scheffler, "Modeling Non-Eroding Perforation of an Oblique Aluminum Target Using the Eulerian CTH Hydrocode," *International Journal of Impact Engineering*, vol. 32, pp. 461–472, 2005.

[539] ——, “Modeling the Effect of Penetrator Nose Shape on Threshold Velocity for Thick Aluminum Targets,” U.S. Army Research Laboratory, Technical Report ARL-TR-1417, July 1997.

[540] ——, “Modeling Threshold Velocity of Hemispherical and Ogival-Nose Tungsten-Alloy Penetrators Perforating Finite Aluminum Targets,” U.S. Army Research Laboratory, Technical Report ARL-TR-1583, January 1998.

[541] D. R. Scheffler and T. M. Sherrick, “Large-Scale Simulations of Monolithic and Segmented Projectiles Impacting Spaced Armor,” Ballistic Research Laboratory, Technical Report ARL-TR-3080, February 1990.

[542] P. R. Schlegel, R. E. Shear, and M. S. Taylor, “A Fuzzy Set Approach to Vulnerability Analysis,” U.S. Army Ballistic Research Laboratory, Technical Report BRL-TR-2697, December 1985.

[543] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, vol. 61, pp. 85 – 117, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608014002135

[544] B. E. Schuster and L. S. Magness, “A Comparison of the Deformation, Flow, and Failure of Two Tungsten Heavy Alloys in Ballistic Impacts,” U.S. Army Research Laboratory, Reprint Report ARL-RP-112, December 2006.

[545] B. E. Schuster, B. P. Peterson, and L. S. Magness, “A Comparison of the Deformation Flow and Failure of Two Tungsten Heavy Alloys in Ballistic Impacts,” U.S. Army Research Laboratory, Reprint Report ARL-RP-122, May 2006.

[546] S. Segletes, “Homogenized penetration calculations,” Army Research Laboratory, Technical Report ARL-TR-1075, 1996.

[547] ——, “Further development of a model for rod ricochet,” Army Research Laboratory, Reprint Report ARL-RP-164, 2007.

[548] S. Segletes and W. P. Walters, “Efficient solution of the long-rod penetration equations of alekseevskii-tate,” Army Research Laboratory, Technical Report ARL-TR-2855, 2002.

[549] S. B. Segletes, “A Rod Ricochet Model,” U.S. Army Research Laboratory, Technical Report ARL-TR-3257, August 2004.

[550] ——, February 2010, a Small Compilation of Material Data for Penetration Modeling.

[551] S. B. Segletes and W. P. Walters, “A Note on the Application of the Extended Bernoulli Equation,” U.S. Army Research Laboratory, Technical Report ARL-TR-1895, February 1999.

[552] S. B. Segletes, “An Adaptation of Walker-Anderson Model Elements Into the Frank-Zook Penetration Model for Use in MUVES,” U.S. Army Research Laboratory, Technical Report ARL-TR-2336, September 2000. [Online]. Available: http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA385339

[553] ——, “Analysis of the Noneroding Penetration of Tungsten Alloy Long Rods Into Aluminum Targets,” U.S. Army Research Laboratory, Technical Report ARL-TR-3075, September 2003.

[554] ——, “Analysis of Erosion Transition in Tungsten-Alloy Rods Into Aluminum Targets,” U.S. Army Research Laboratory, Technical Report ARL-TR-3153, March 2004.

[555] ——, “The Erosion Transition of Tungsten-Alloy Long Rods into Aluminum Targets,” *International Journal of Solids and Structures*, vol. 44, pp. 2168–2191, 2007.

[556] S. B. Segletes, R. Grote, and J. Polesne, “Improving the Rod-Penetration Algorithm for Tomorrow’s Armors,” U.S. Army Research Laboratory, Reprint Report ARL-RP-23, June 2001.

[557] S. B. Segletes and W. P. Walters, “Extensions to the exact solution of the long-rod penetration/erosion equations,” *International Journal of Impact Engineering*, vol. 28, no. 4, pp. 363–376, 2003.

[558] H. J. Seltman, *Experimental Design and Analysis.* Carnegie Mellon University, 2012.

[559] R. Setiono and H. Liu, "Neural-Network Feature Selector," *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 654–662, May 1997.

[560] P. K. Sharpe and R. J. Solly, "Dealing with Missing Values in Neural Network-Based Diagnostic Systems," *Neural Computing and Applications*, vol. 3, pp. 73–77, 1995.

[561] D. A. Shockey and D. R. Curran, "Examination of Steel Specimens Impacted at Hypervelocity," Stanford Research Institute, Technical Report SRI-PYU-3754, June 1975.

[562] E. Shorohov, A. Gornovoi, and A. Denisenko, "Elastic precursor attenuation in iron and steels," *Le Journal de Physique IV*, vol. 4, no. C8, pp. C8–409, 1994.

[563] G. F. Silsby, "Penetration of Semi-Infinite Steel Targets by Tungsten Long Rods at 1.3 to 4.5 km/s," in *Proceedings of the 8th International Symposium on Ballistics*, 1984.

[564] ——, "Terminal Ballistics of a Reduced-Mass Penetrator," U.S. Army Research Laboratory, Memorandum Report ARL-MR-320, July 1996.

[565] M. Skrbek, "Fast neural network implementation," *Neural Network World*, vol. 9, no. 5, pp. 375–391, 1999.

[566] J. A. Snyman, *Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms*, P. M. Pardalos and D. W. Hearn, Eds. Springer, 2005, vol. 97.

[567] S. S. Sodhi, P. Chandra, and S. Tanwar, "A new weight initialization method for sigmoidal feedforward artificial neural networks," in *Neural Networks (IJCNN), 2014 International Joint Conference on.* IEEE, 2014, pp. 291–298.

[568] J. Sola and J. Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems," *Nuclear Science, IEEE Transactions on*, vol. 44, no. 3, pp. 1464–1468, 1997.

[569] B. R. Sorensen, K. D. Kimsey, and B. M. Love, “High-Velocity Impact of Low-Density Projectiles on Structural Aluminum Armor,” U.S. Army Research Laboratory, Reprint Report ARL-RP-271, September 2009, Reprint from International Journal of Impact Engineering, Vol. 35, pp. 18081815, 2008.

[570] B. R. Sorensen, K. D. Kimsey, G. F. Silsby, D. R. Scheffler, T. M. Sherrick, and W. S. De Rosset, “High Velocity Penetration of Steel Targets,” *International Journal of Impact Engineering*, vol. 11, no. 1, pp. 107–119, 1991.

[571] B. Sorensen, “Design, Analysis, and Verification of a 50-mm Push/Traction Sabot,” U.S. Army Research Laboratory, Technical Report ARL-TR-2292, August 2000.

[572] B. R. Sorensen, K. D. Kimsey, J. A. Zukas, and K. Frank, “Numerical analysis and modeling of jacketed rod penetration,” *International Journal of Impact Engineering*, vol. 22, no. 1, pp. 71–91, 1999.

[573] D. F. Specht, “A general regression neural network,” *IEEE Transactions on Neural Networks*, vol. 2, no. 6, pp. 568–576, 1991.

[574] C. Sporleder, M. Van Erp, T. Porcelijn, and A. Van Den Bosch, “Spotting the ’odd-one-out’: Data-driven error detection and correction in textual databases,” in *Proc. EACL 2006 Workshop Adaptive Text Extraction and Mining*, 2006, pp. 40–47.

[575] M. Squire, *Clean data : save time by discovering effortless strategies for cleaning, organizing, and manipulating your data.*, ser. Community experience distilled. Birmingham, UK : Packt Publishing, 2015., 2015. [Online]. Available: http://proxy-tu.researchport.umd.edu/login?ins=tu&url=http://search.ebscohost.com.proxy-tu.researchport.umd.edu/login.aspx?direct=true&db=cat01451a&AN=towson.004560555&site=eds-live&scope=site

[576] S. A. Stanhope and J. M. Daida, “Optimal mutation and crossover rates for a genetic algorithm operating in a dynamic environment,” in *International Conference on Evolutionary Programming.* Springer, 1998, pp. 693–702.

[577] M. W. Starks, “Vulnerability science: A response to a criticism of the ballistic research laboratorys vulnerability modeling strategy,” Ballistic Research Laboratory, Technical Report BRL-TR-3113, 1990.

[578] B. Steelman, “Vulnerability of Four Red Targets to Single 60-Grain Flechette Impacts,” U.S. Army Research Laboratory, Memorandum Report ARL-MR-199, November 1994.

[579] S. W. Stepniewski and A. J. Keane, “Pruning backpropagation neural networks using modern stochastic optimisation techniques,” *Neural Computing & Applications*, vol. 5, no. 2, pp. 76–98, 1997.

[580] J. Stubberfield, N. Lynch, and I. Wallis, “Comparisons of unitary and jacketed rod penetration into semi-infinite and oblique plate targets at system equivalent velocities,” in *Proceedings of the 22nd International Symposium on Ballistics*, 2005.

[581] P. Stubbings, “A comparative study of the backpropagation learning algorithm for neural networks & decision tree induction.”

[582] A. T. V. Subgroup, “Penetration equations handbook for kinetic-energy penetrators. with modifications to reflect the implementation in covart 4,” Joint Technical Coordinating Group for Munitions Effectiveness (Anti-Air), Tech. Rep. 61 JTCG/ME-77-16, October 1985.

[583] R. Subramanian and S. Bless, “Penetration of Semi-Infinite AD995 Alumina Targets by Tungsten Long Rod Penetrators from 1.5 to 3.5 km/s,” *International Journal of Impact Engineering*, vol. 17, pp. 807–816, 1995.

[584] R. Subramanian, S. Bless, J. U. Cazamias, and D. Berry, “Reverse Impact Experiments Against Tungsten Rods and Results for Aluminum Penetration Between 1.5 and 4.2 km/s,” *International Journal of Impact Engineering*, vol. 17, pp. 817–824, 1995.

[585] R. Subramanian and S. Bless, “Reference correlations for tungsten long rods striking semi-infinite steel targets,” in *Proc. $19^{th}$ Int. Symp. of Ballistics*, 2001.

[586] N. R. Suri, D. Deodhare, and P. Nagabhushan, "Parallel levenberg-marquardt-based neural network training on linux clusters-a case study." in *ICVGIP*, 2002.

[587] K. Suzuki, I. Horiba, and N. Sugie, "A simple neural network pruning algorithm with application to filter synthesis," *Neural Processing Letters*, vol. 13, no. 1, pp. 43–53, 2001.

[588] D. A. Swanson, J. Tayman, and T. Bryan, "Mape-r: a rescaled measure of accuracy for cross-sectional subnational population forecasts," *Journal of Population Research*, vol. 28, no. 2-3, pp. 225–243, 2011.

[589] K. Swingler, *Applying Neural Networks: A Practical Guide.* Academic Press, 1996. [Online]. Available: http://books.google.com/books?id=bq0YnP4BNKsC

[590] T. Taheri. (2010, June) Benchmarking and Comparing Encog, Neuroph and JOONE Neural Networks. [Online]. Available: http://www.codeproject.com/Articles/85487/Benchmarking-and-Comparing-Encog-Neuroph-and-JOONE

[591] N. L. Talbot, *Using LaTeX to Write a PhD Thesis.* Dickimaw Books, 2007.

[592] W. Talloen, "Nested loop cross validation for classification using nlcv," 2011.

[593] P. J. Tanenbaum, "A Guide for Technical Authors: Creating Texts That Are Clear and Effective," U.S. Army Research Laboratory, Special Report ARL-SR-0198, 2010.

[594] L. Tarassenko, *A Guide to Neural Computing Applications.* New York: Arnold, 1998.

[595] A. Tate, "A Theory for the Deceleration of Long Rods after Impact," *Journal of Mechanics and Physics of Solids*, vol. 15, pp. 387–399, 1967.

[596] ——, "Further Results in the Theory of Long Rod Penetration," *Journal of Mechanics and Physics of Solids*, vol. 17, pp. 141–150, 1969.

[597] ——, "A simple estimate of the minimum target obliquity required for the ricochet of a high speed long rod projectile," *Journal of physics. D. Applied physics*, vol. 12, no. 11, pp. 1825–1829, 1979.

[598] ——, “Long rod penetration modelspart ii. extensions to the hydrodynamic theory of penetration,” *International Journal of mechanical sciences*, vol. 28, no. 9, pp. 599–612, 1986.

[599] A. Tate, K. E. B. Green, P. G. Chamberlain, and R. G. Baker, “Model Scale Experiments on Long Rd Penetration,” in *Proceedings of the $4^{th}$ International Symposium on Ballistics*, 1978.

[600] K. J. Tauer, R. J. Dowding, and P. Woolsey, “Ballistic Performance of a Coated Powder Tungsten Alloy,” in *Tungsten Alloy Technology*, R. J. Dowding, Ed., no. ARL-MR-57, January 1993, pp. 57–64.

[601] I. V. Tetko, A. E. P. Villa, and D. J. Livingstone, “Neural Network Studies. 2. Variable Selection,” *Journal of Chemical Information and Computer Sciences*, vol. 36, pp. 794–803, 1996.

[602] G. Thimm and E. Fiesler, “Pruning of neural networks,” IDIAP, Tech. Rep., 1997.

[603] C. Tofallis, “A better measure of relative prediction accuracy for model selection and model estimation,” *Journal of the Operational Research Society*, vol. 66, no. 8, pp. 1352–1362, 2015.

[604] V. Tresp, S. Ahmad, and R. Neuneier, “Training Neural Networks with Deficient Data,” in *NIPS*, 1993, pp. 128–135.

[605] C. Truxillo, “Maximum Likelihood Parameter Estimation with Incomplete Data,” in *Proceedings of the Thirtieth Annual SAS Users Group International Conference.* SAS Institute Inc., 2005.

[606] K. Tumer and J. Ghosh, “Estimating the bayes error rate through classifier combining,” in *Pattern Recognition, 1996., Proceedings of the $13^{th}$ International Conference on*, vol. 2. IEEE, 1996, pp. 695–699.

[607] N. Ueda and R. Nakano, "Estimating expected error rates of neural network classifiers in small sample size situations: a comparison of cross-validation and bootstrap," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 1. IEEE, 1995, pp. 101–104.

[608] U.S. Department of Defense, "Operation of the Defense Acquisition System," U.S. Department of Defense, Washington, DC: Government Printing Office, Department of Defense Instruction DoDI 5000.02, December 2008. [Online]. Available: http://www.dtic.mil/whs/directives/corres/pdf/500002p.pdf

[609] K. Vahedi, H. Zohoor, A. Nezamabadi, and M. Zolfaghari, "Performance optimization of a long rod penetrator penetrating into a semi-infinite target considering bending characteristics," *Turkish Journal of Engineering & Environmental Sciences*, vol. 33, pp. 9–20, 2009.

[610] F. Valafar and O. K. Ersoy, "A parallel implementation of backpropagation neural network on maspar mp-1," 1993.

[611] J. Van Gorp, J. Schoukens, and R. Pintelon, "Adding Input Noise to Increase the Generalization of Neural Networks is a Bad Idea," in *Proceedings of the International Workshop on Advanced Black-Box Techniques for Nonlinear Modeling, Leuven, Belgium*, 1998, pp. 1–7.

[612] G. Vanwinckelen and H. Blockeel, "On estimating model accuracy with repeated cross-validation," in *BeneLearn 2012: Proceedings of the 21$^{st}$ Belgian-Dutch Conference on Machine Learning*, 2012, pp. 39–44.

[613] C. Vasios, G. Matsopoulos, E. Ventouras, K. Nikita, and N. Uzunoglu, "Cross-validation and neural network architecture selection for the classification of intracranial current sources," in *Neural Network Applications in Electrical Engineering, 2004. NEUREL 2004. 2004 7$^{th}$ Seminar on.* IEEE, 2004, pp. 151–158.

[614] V. R. Vemuri, *Artificial Neural Networks: Concepts and Control Applications.* Los Alamitos, CA: IEEE Computer Society Press, 1992.

[615] A. Verikas and M. Bacauskiene, “Feature selection with neural networks,” *Pattern Recognition Letters*, vol. 23, no. 11, pp. 1323–1335, 2002.

[616] J. Vetrovec, R. Padfield, D. Schwab, P. Nebolsine, R. Hayami, M. Zwiener, J. Huntington, and S. L. Hancock, “Analysis and Testing of Rod-Like Penetrators in the 4–5 km/s Velocity Regime,” *International Journal of Impact Engineering*, vol. 26, pp. 797–808, 2001.

[617] R. Šindelář and R. Babuška, “Input Selection for Nonlinear Regression Models,” *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 5, pp. 688 – 696, October 2004.

[618] Y. Wada and M. Kawato, “Estimation of generalization capability by combination of new information criterion and cross validation,” in *Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on*, vol. 2. IEEE, 1991, pp. 1–6.

[619] E.-J. Wagenmakers and S. Farrell, “AIc model selection using akaike weights,” *Psychonomic Bulletin & Review*, vol. 11, no. 1, pp. 192–196, 2004.

[620] S. Walczak and N. Cerpa, “Heuristic principles for the design of artificial neural networks,” *Information and software technology*, vol. 41, no. 2, pp. 107–117, 1999.

[621] J. D. Walker, “Hypervelocity penetration modeling: momentum vs. energy and energy transfer mechanisms,” *International Journal of Impact Engineering*, vol. 26, no. 1, pp. 809–822, 2001.

[622] J. D. Walker and C. E. Anderson Jr, “The influence of initial nose shape in eroding penetration,” *International Journal of Impact Engineering*, vol. 15, no. 2, pp. 139–148, 1994.

[623] J. D. Walker and C. E. Anderson Jr., “A Time-Dependent Model for Long-Rod Penetration,” *International Journal of Solids and Structures*, vol. 16, no. 1, pp. 19–48, 1995.

[624] J. D. Walker, C. E. Anderson Jr, and D. L. Goodlin, “Tungsten into steel penetration including velocity, l/d, and impact inclination effects,” in *Proceedings of the 19$^{th}$ international symposium on ballistics, Interlaken Switzerland*, 2001, pp. 1133–1139.

[625] J. D. Walker, S. A. Mullin, C. E. Weiss, and P. O. Leslie, “Penetration of Boron Carbide, Aluminum, and Beryllium Alloys by Depleted Uranium Rods: Modeling and Experimentation,” *International Journal of Impact Engineering*, vol. 33, pp. 826–836, 2006.

[626] E. N. Waltenburg, W. Mclauchlan, and S. Wiest, *Exploratory Data Analysis: A Primer for Undergraduates.* Kendall Hunt Publishing, 2012.

[627] W. P. Walters and J. N. Majerus, “Impact Models for Penetration and Hole Growth,” Ballistic Research Laboratory, Technical Report ARBRL-TR-02069, May 1978.

[628] W. P. Walters and S. B. Segletes, “An exact solution of the long rod penetration equations,” *International Journal of Impact Engineering*, vol. 11, no. 2, pp. 225–231, 1991.

[629] W. P. Walters and C. L. Williams, “The Influence of Armor Material Parameters on the Penetration by Long-Rod Projectiles,” U.S. Army Research Laboratory, Reprint Report ARL-RP-129, September 2006.

[630] D. Wang and S. X. Chen. (2006) Nonparametric imputation of missing values for estimating equation based inference. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.71.7288&rep=rep1&type=pdf

[631] T. L. Warren and M. J. Forrestal, “Comment on ”Penetration Equations for Ogive-Nose Rods into Aluminum Targets” (Int J Impact Eng 2008; 35: 727–730),” *International Journal of Impact Engineering*, vol. 36, p. 1079, 2009.

[632] T. L. Warren and K. L. Poormon, “Penetration of 6061-T6511 Aluminum Targets by Ogive-Nosed VAR 4340 Steel Projectiles at Oblique Angles: Experiments and Simulations,” *International Journal of Impact Engineering*, vol. 25, pp. 993–1022, 2001.

[633] M. Watson, *Practical Artifical Intellegence Programming with Java*, 2008.

[634] J. C. Wayman, “Multiple imputation for missing data: What is it and how can I use it?” in *Annual Meeting of the American Educational Research Association, Chicago, IL*, 2003, pp. 2–16.

[635] K. Weber, T. J. Holmquist, and D. W. Templeton, “The Response of Layered Aluminum Nitride Targets Subjected to Hypervelocity Impact,” *International Journal of Impact Engineering*, vol. 26, pp. 831–841, 2001.

[636] G. R. Weckman, H. W. Paschold, J. D. DOWLER, H. S. WHITING, and W. A. YOUNG, “Using neural networks with limited data to estimate manufacturing cost,” 2010.

[637] T. Weerasooriya and J. Clayton, “Failure behavior of a tungsten heavy alloy at different strain rates under tensile loading,” in *Proceedings of Int Conf on Tungsten, Refractory & Hardmetals VI*, 2006.

[638] A. Weigend, D. Rumelhart, and B. Huberman, “Generalization by Weight-Elimination with Application to Forecasting,” in *Advances in Neural Information Processing Systems 3*, R. Lippmann, J. Moody, and D. Touretzky, Eds. Morgan Kaufmann, November 1991, pp. 875–882.

[639] T. Weise, *Global Optimization Algorithms – Theory and Application.* it-weise.de (self-published): Germany, 2009. [Online]. Available: http://www.it-weise.de/projects/book.pdf

[640] H. M. Wen, Y. He, and B. Lan, “A Combined Numerical and Theoretical Study on the Penetration of a Jacketed Rod into Semi-Infinite Targets,” *International Journal of Impact Engineering*, vol. 38, pp. 1001–1010, 2011.

[641] I. Wesley-Smith, “A Parallel Artificial Neural Network Implementation,” in *Proceedings of The National Conference On Undergraduate Research*, April 2006.

[642] V. L. Wiggins, S. K. Engquist, and L. T. Looper, “Neural Network Applications: A Literature Review,” Armstrong Laboratory, Technical Paper AL-TP-1992-0044, November 1992.

[643] B. Wilamowski, “How to not get frustrated with neural networks,” in *Industrial Technology (ICIT), 2011 IEEE International Conference on*, 2011, pp. 5–11.

[644] B. M. Wilamowski, Y. Chen, and A. Malinowski, “Efficient algorithm for training neural networks with one hidden layer,” in *Proc. IJCNN*, vol. 3, 1999, pp. 1725–728.

[645] B. M. Wilamowski and H. Yu, “Improved computation for levenberg–marquardt training,” *IEEE Transactions on Neural Networks*, vol. 21, no. 6, pp. 930–937, 2010.

[646] R. A. Wildman and D. S. Weile, “Greedy search and a hybrid local optimization/genetic algorithm for tree-based inverse scattering,” *Microwave and optical Technology Letters*, vol. 50, no. 3, pp. 822–825, March 2008.

[647] G. Wilensky, N. Manukian, J. Neuhaus, and N. Rivetti, “Neural Network Studies,” Logicon RDA, Technical Report RDA-TR-43-0005-001, July 1993.

[648] M. Wilkins, J. Gibbons, V. Hohler, A. J. Stilp, and M. Cozzi, “Ballistic Performance of AIN, SiC, and Al2O3 Ceramic Tiles IMpacted by Tungsten Alloy Long Rod Projectiles,” in *Proceedings of the U. S. Army TACOM Combat Survivability Symposium*, vol. 2, 1991, pp. 75–95.

[649] A. G. Williams, “The Search for High Velocity,” *Guns Review International*, May 1996. [Online]. Available: http://www.quarryhs.co.uk/highvel.htm

[650] B. E. Williams and J. J. Stiglich Jr., “Hafnium- and Titanium-Coated Tungsten Powders for Kinetic Energy Penetrators, Phase I, SBIR,” U.S.Army Materials Technology Laboratory, Technical Report MTL TR 92-36, May 1992.

[651] B. E. Williams, J. J. Stiglich Jr., and R. B. Kaplan, “Coated Tungsten Powders for Advanced Ordance Applications, Phase II, SBIR,” U.S. Army Materials Technology Laboratory, Technical Report MTL TR 92-35, May 1992.

[652] D. R. Wilson and T. R. Martinez, “Combining cross-validation and confidence to measure fitness,” in *Neural Networks, 1999. IJCNN’99. International Joint Conference on*, vol. 2. IEEE, 1999, pp. 1409–1414.

[653] L. L. Wilson, J. C. Foster Jr., S. E. Jones, and P. P. Gillis, “Experimental Rod Impact Results,” *International Journal of Impact Engineering*, vol. 8, no. 1, pp. 15–25, 1989.

[654] L. L. Wilson, “Armor penetration dynamics,” Air Force Armament Laboratory, Technical Report AFATL-TR-84-43, 1984.

[655] R. L. Woodward, “Modelling Penetration by Slender High Kinetic Energy Projectiles,” Materials Research Laboratories, Report MRL-R-811, April 1981.

[656] P. Woolsey, “Residual Penetration Ballistic Testing of Armor Ceramics,” in *Proceedings of the Second Annual TACOM Combat Vehicle Survivability Symposium*, 1991.

[657] P. Woolsey, R. Dowding, K. J. Tatter, and F. S. Hodi, “Performance-Property Relationsips in Tungsten Alloy Penetrators,” in *Tungsten Alloy Technology*, R. J. Dowding, Ed., no. ARL-MR-57, January 1993, pp. 73–80.

[658] P. Woolsey, D. Kokidko, and S. Mariano, “Progress Report on Ballistic Test Methodology for Armor Ceramics,” in *Proceedings of TACOM Combat Vehicle Survivability Symposium*, 1990.

[659] P. Woolsey, D. Kokidko, and S. A. Mariano, “Alternative Test Methodology for Ballistic Performance Ranking of Armor Ceramics,” Materials Technology Laboratory, Technical Report MTL TR 89-43, April 1989.

[660] P. Woolsey, S. Mariano, and D. Kokidko, “Alternative Test Methodology for Ballistic Performance Ranking of Armor Ceramics,” in *Proceedings of the Fifth Annual TACOM Armor Coordinating Conference*, 1989.

[661] J. Wright and M. Manic, “Neural network architecture selection analysis with application to cryptography location,” in *Neural Networks (IJCNN), The 2010 International Joint Conference on.* IEEE, July 2010, pp. 1–6.

[662] T. W. Wright, “Penetration with Long Rods: A Theoretical Framework and Comparison with Instrumented Impacts,” U.S. Army Ballistic Research Laboratory, Technical Report ARBRL-TR-02323, May 1981.

[663] ——, “A Survey of Penetration Mechanics for Long Rods,” Ballistic Research Laboratory, Aberdeen Proving Ground, MD, Technical Report ARBRL-TR-02496, June 1983.

[664] T. W. Wright and K. Frank, “Approaches to penetration problems,” Ballistic Research Laboratory, Technical Report BRL-TR-2957, 1988.

[665] T. W. Wright, *The physics and mathematics of adiabatic shear bands.* Cambridge University Press, 2002.

[666] J.-K. Wu, *Neural Networks and Simulation Methods.* New York: Marcel Dekker, 1994.

[667] H. Yakuwa, T. Narita, M. Kawasaki, M. Miyasaka, C. Fang, T. Go, and S. Nakahama, “Development of a sulfidation-corrosion resistant nickel-base superalloy for fcc power recovery turbine rotors.”

[668] Y. Yang and J. Xin, “Penetration Mechanisms and Ballistic Performance of TiC-TiB Composite Armour under the Impact of Long-Rod Kinetic-Energy Projectiles,” *Advanced Materials Research*, vol. 391–392, pp. 236–241, 2012.

[669] ——, “The Influences of Lateral Confinement on Ballistic Performance of TiC-TiB Composites under the Impact of Long-Rod Kinetic-Energy Projectiles,” *Advanced Materials Research*, vol. 391–392, pp. 230–235, 2012.

[670] A. L. Yarin, M. B. Rubin, and J. V. Roisman, “Penetration of a Rigid Projectile into an Elastic-Plastic Target of Finite Thickness,” *International Journal of Impact Engineering*, vol. 16, no. 5/6, pp. 801–831, 1995.

[671] J. D. Yatteau, “High velocity multiple plate penetration model,” Naval Surface Weapons Center, Technical Report NSWC TR 82-123, 1982.

[672] J. D. Yatteau, R. H. Zernow, G. W. Recht, and D. L. Dickinson, “FATEPEN, a model to predict terminal ballistic penetration and damage to military targets model improvements 1999-2005.”

[673] J. D. Yatteau, R. H. Zernow, G. W. Recht, K. T. Edquist, and D. L. Dicksinson, “FATEPEN, a model to predict terminal ballistic penetration and damage to military targets,” Tech. Rep., 1999.

[674] D. Yaziv, Y. Reifen, and G. Gans, “The penetration process of long rods at interfaces between materials,” in *Proc. of 21$^{th}$ Int. Symp. Ballistics*, 2004.

[675] D. S. Yeung and X.-q. Zeng, “Hidden neuron pruning for multilayer perceptrons using a sensitivity measure,” in *Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference on*, vol. 4. IEEE, 2002, pp. 1751–1757.

[676] Y. H. Yoo and M. Lee, “Protection Effectiveness of an Oblique Plate Against a Long Rod,” *International Journal of Impact Engineering*, vol. 33, pp. 872–879, 2006.

[677] Y.-H. Yoo and H. Shin, “Protection capability of dual flying plates against obliquely impacting long-rod penetrators,” *International Journal of Impact Engineering*, vol. 30, no. 1, pp. 55–68, 2004.

[678] M. M. R. Yousefi, M. Mirmomeni, and C. Lucas, “Input Variables Selection Using Mutual Information for Neuro Fuzzy Modeling with the Application to Time Series Forecasting,” in *International Joint Conference on Neural Networks*, 2007, pp. 1121–1126.

[679] H. Yu and B. Wilamowski, “Neural network training with second order algorithms,” in *Human–Computer Systems Interaction: Backgrounds and Applications 2*. Springer, 2012, pp. 463–476.

[680] H. Yu and B. M. Wilamowski, “Levenberg–marquardt training,” *Industrial Electronics Handbook*, vol. 5, no. 12, p. 1, 2011.

[681] Y. C. Yuan, “Multiple Imputation for Missing Data: Concepts and New Development,” in *Proceedings of the Twenty-Fifth Annual SAS Users Group International Conference*. SAS Institute Inc., 2000.

[682] C. V. Zabielski and M. Levy, “Corrosion of High Density Kinetic Energy Penetrator Materials,” U.S. Army Materials Technology Laboratory, Technical Report MTL TR 92-17, April 1992.

[683] A. Zapranis and E. Livanis, “Prediction intervals for neural network models,” in *Proceedings of the 9$^{th}$ WSEAS International Conference on Computers.* World Scientific and Engineering Academy and Society (WSEAS), 2005, p. 76.

[684] C. Zener and J. H. Hollomon, “Effect of strain rate upon plastic flow of steel,” *Journal of Applied Physics*, vol. 15, no. 1, pp. 22–32, 1944. [Online]. Available: http://scitation.aip.org/content/aip/journal/jap/15/1/10.1063/1.1707363

[685] G. Zhang, M. Y Hu, B. Eddy Patuwo, and D. C Indro, “Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis,” *European journal of operational research*, vol. 116, no. 1, pp. 16–32, 1999.

[686] J. Zhang, Q. Jin, and Y. Xu, “Inferential Estimation of Polymer Melt Index Using Sequentially Trained Bootstrap Aggregated Neural Networks,” *Chemical Engineering & Technology*, vol. 29, no. 4, pp. 442–448, 2006.

[687] P. Zhang, “Model selection via multifold cross validation,” *The Annals of Statistics*, pp. 299–313, 1993.

[688] X.-d. Zhang, “A master-slave neural network for precise diagnosis of large rotating machinery,” in *Innovative Computing, Information and Control, 2007. ICICIC’07. Second International Conference on.* IEEE, 2007, pp. 11–11.

[689] J. Zhao, X. W. Chen, F. N. Jin, and Y. Xu, “Depth of Penetration of High-Speed Penetrator with Including the Effect of Mass Abrasion,” *International Journal of Impact Engineering*, vol. 37, pp. 971–979, 2010.

[690] Z.-H. Zhou, J. Wu, and W. Tang, “Ensembling neural networks: Many could be better than all,” *Artificial Intelligence*, vol. 137, no. 12, pp. 239–263, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000437020200190X

[691] J. A. Zook, K. Frank, and G. F. Silsby, “Terminal Ballistics Test and Analysis Guidelines for the Penetration Mechanics Branch,” Ballistic Research Laboratory, Memorandum Report BRL-MR-3960, 1992.

[692] J. A. Zook and K. Frank, “Comparative Penetration Performance of Tungsten Alloy L/D=10 Long Rods with Different Nose Shapes Fired at Rolled Homogeneous Armor,” U.S. Army Ballistic Research Laboratory, Memorandum Report BRL-MR-3480, November 1985.

[693] J. A. Zook, K. Frank, and G. F. Silsby, “Terminal ballistics handbook of the penetration mechanics branch,” Ballistic Research Laboratory, Special Report BRL-SP-78, 1989.

[694] W. Zucchini, “An introduction to model selection,” *Journal of Mathematical Psychology*, vol. 44, no. 1, pp. 41–61, 2000.

[695] J. Zukas and D. Scheffler, “Impact effects in multilayered plates,” Army Research Laboratory, Technical Report ARL-TR-2223, April 2000.

[696] J. A. Zukas, *High velocity impact dynamics.* Wiley-Interscience, 1990.

[697] J. Zupan, “Introduction to artificial neural network (ann) methods: what they are and how to use them,” *Acta Chimica Slovenica*, vol. 41, pp. 327–327, 1994.

[698] J. Zurada, A. Malinowski, and I. Cloete, “Sensitivity analysis for minimization of input data dimension for feedforward neural network,” in *Circuits and Systems, 1994. ISCAS ’94., 1994 IEEE International Symposium on*, vol. 6, 1994, pp. 447–4506.

# Appendix A

# Vitae

## John Robert Auten Sr.

### Contact Information

Towson University
Fisher College of Science & Mathematics
Department of Computer and Information Sciences
7800 York Road
Towson, Maryland 21252 USA

(410)704-2633
jauten@towson.edu

### Research Interests

Terminal ballistics, computational intelligence, cloud computing, machine learning, modeling & simulation, methodology development, and software development.

### Education

**Towson University**

D.Sc. in Information Technology, December 2016

- Dissertation Topic: Predicting the Terminal Ballistics of Kinetic Energy Projectiles Using Artificial Neural Networks
- Advisor: Dr. Robert J. Hammell II

M.S. in Applied Information Technology, August 2007

- Case Topic: TRAIL – Technical Report and Information Library
- Advisor: Al Skudzinskas

B.S. in Computer Information Systems, December 2002

- Degree Honors: Cum Laude
- Minor: Business Administration

### Current Positions

**Operations Research Analyst**
Survivability/Lethality Analysis Directorate, U.S. Army Research Laboratory
Aberdeen Proving Ground, MD
December 2005 – Present
Key Work: Methodology development, software development, project management, and modeling & simulation.

**Adjunct Professor**
Department of Computer & Information Sciences, Towson University
Towson, MD
January 2014 – Present
Courses Taught:

- COSC109: Computers and Creativity
- ITEC427: Cloud Computing for Enterprises

### Selected Previous Positions

**Senior Analyst**
Quantum Research International
Bel Air, MD
August 2004 – December 2005
Key Tasks: Methodology development, methodology review, and modeling & simulation.

**Analyst**
Regal Decision Systems, Inc.
Severna Park, MD
July 2003 – August 2004
Key Tasks: Project management, data collection, training coordinator, and modeling & simulation.

**Munitions Systems Specialist**
Maryland Air National Guard, Air Force
Warfield Air National Guard Base, MD
December 1998 – December 2004
Key Tasks: Handling, storing, transporting, arming, and disarming non-nuclear weapons systems.

Publications

J. R. Auten Sr. and R. J. Hammell II, "Comparing the Iterative Application of an Artificial Neural Network vs. a Phenomenological Model for Predicting the Terminal Ballistics of Kinetic Energy Projectiles Against Multiple Element Targets," *Proceedings of the IEEE Computing Conference*, 2017, accepted.

J. R. Auten Sr. and R. J. Hammell II, "Comparing the Prediction Capabilities of an Artificial Neural Network vs a Phenomenological Model for Predicting the Terminal Ballistics of Kinetic Energy Projectiles," in *Proceedings of the Conference on Information Systems Applied Research (CONISAR)*, 2015.

J. R. Auten Sr. and R. J. Hammell II, "Predicting the Perforation of Kinetic Energy Projectiles using Artificial Neural Networks," in *IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES)*, vol. 1., 2014, pp. 132–139.

J. R. Auten Sr., "Kinetic Energy Rod Analysis Tool User's Guide (version 1.0.0)," U.S. Army Research Laboratory, Technical Report ARL-TR-6886, April 2014.

J. R. Auten Sr. and R. J. Hammell II, "Predicting the Terminal Ballistics of Kinetic Energy Projectiles using Artificial Neural Networks," *Journal of Information Systems Applied Research*, vol. 7., no. 1, pp. 23–32, February 2014.

J. R. Auten Sr., "A Comparison of Kinetic Energy Rod Algorithm Predictions to Test Data," U.S. Army Research Laboratory, Technical Report ARL-TR-6192, April 2012.

J. R. Auten Sr., "A Comparison of Penetration Algorithms: Predictions vs. Test Data for Kinetic Energy Rods," in *Proceedings of the 26th International Symposium of Ballistics*, vol. 2., September 2011, pp. 1522–1532.

T. Farrand, R. Summers, J. Angel, N. Reichenbach, and J. R. Auten Sr., "Flight Test Data and Analysis for the Guided Multiple Launch Rocket System-Alternative Warhead: A Concept," U.S. Army Research Laboratory, Technical Report ARL-SR-246, July 2012.

T. Farrand, R. Summers, J. Angel, N. Reichenbach, and J. R. Auten Sr., "Flight Test Data and Analysis for the Guided Multiple Launch Rocket System-Alternative Warhead: B Concept," U.S. Army Research Laboratory, Technical Report ARL-SR-247, July 2012.

T. Farrand, R. Summers, J. Angel, N. Reichenbach, and J. R. Auten Sr., "Flight Test Data and Analysis for the Guided Multiple Launch Rocket System-Alternative Warhead: C Concept," U.S. Army Research Laboratory, Technical Report ARL-SR-248, July 2012.

T. Farrand, R. Summers, and J. R. Auten Sr., "Fragment Characterization of the Guided Multiple Launch Rocket System-Alternative Warhead for Engineering Development Test-1," U.S. Army Research Laboratory, Technical Report ARL-SR-0313, March 2015.

R. Summers, T. Farrand, and J. R. Auten Sr., “Fragment Characterization of the Guided Multiple Launch Rocket System-Alternative Warhead for Engineering Development Test-2,” U.S. Army Research Laboratory, Technical Report ARL-SR-0314, March 2015.

R. Summers, T. Farrand, J. R. Auten Sr., and C. Inmon, “Fragment Characterization of the Guided Multiple Launch Rocket System-Alternative Warhead for Engineering Development Test-3,” U.S. Army Research Laboratory, Technical Report ARL-SR-0315, March 2015.

R. Summers, T. Farrand, J. R. Auten Sr., and C. Inmon, “Fragment Characterization of the Guided Multiple Launch Rocket System-Alternative Warhead for Production Qualification Test-1,” U.S. Army Research Laboratory, Technical Report ARL-SR-0316, March 2015.

S. Snead and J. R. Auten Sr., “Ballistic Vulnerability and Lethality Analyses – With a Discussion on Developing Methodology for Multiple Explosively Formed Penetrators,” in *Advanced Ballistics Science and Engineering,* B. Burns, Ed. U.S. Army Research Laboratory, November 2008, no. ARL-SR-168, Special Report, Analyses: Ballistic Survivability and Lethality Analyses, pp. 581–604.

## Conference Talks

*Comparing the Prediction Capabilities of an Artificial Neural Network vs a Phenomenological Model for Predicting the Terminal Ballistics of Kinetic Energy Projectiles*, 21st Annual United States Military Academy and U.S. Army Research Laboratory Symposium (AUTS), Aberdeen Proving Ground. (October 2015)

## Other Talks

*Predicting the Terminal Ballistics of Kinetic Energy Projectiles Using Artificial Neural Networks (ANNs)*, Towson University Computer Sciences Department 30th Anniversary Celebration, Towson University. (April 2014)

*Developing Methodology for Multiple Explosively Formed Penetrators*, U.S. Army Research Laboratory’s Ballistics Symposium, Aberdeen Proving Ground. (November 2008)

## Honors and Awards

| | |
|---|---|
| 2012 | Army Materiel Command, Award for Meritorious Achievement in Systems Analysis (Large Group Category) |
| 2011 | Department of the Army Commendation |
| 2003 | Air Force Achievement Medal |
| 2003 | National Defense Service Medal |
| 2003 | Air Force Expeditionary Service Ribbon with Gold Border |
| 2003 | Global War on Terrorism Expeditionary Medal |
| 2000 | Airman of the Year, 175th Logistics Group |
| 1999 | Distinguished Graduate, Air Force Munitions School |
| 1996 | Eagle Scout |

# Appendix B

# Penetration Database Markup Language

## B.1 PDML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:jxb="http://java.sun.com/xml/ns/jaxb" jxb:version="2.0">
 <xsd:annotation>
  <xsd:documentation>
Description: Generated for PDML
Owner: John R Auten Sr.
Date: 2013-07-01
Current Version: 0.11

Version: 0.11
-Changed reported boolean to a ENUM
Version: 0.10
-Added element for ogive nose CRH value
Version: 0.9
-Added releasability information for source data
Version: 0.8
-Added extensive nose characterization elements
Version: 0.7
-Added UTS as a element and made units an optional attribute
Version: 0.6
-Removed required value for velocity since limit velocity test do not need it
Version: 0.5
-Added three new test types: SNL, MNL, MAL
Version: 0.4
-Added suspect attribute to test
Version: 0.3
-Corrected elements and attribute types
Version: 0.2
-Corrected elements and attribute requirement status
Version: 0.1
-Initial version

References:
```

```
RefID0087
</xsd:documentation>
 </xsd:annotation>
 <xsd:element name=”PDML”>
  <xsd:annotation>
   <xsd:documentation>Penetration Database Markup Language</xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
   <xsd:sequence>
    <xsd:element name=”Test” minOccurs=”1” maxOccurs=”unbounded”>
     <xsd:annotation>
      <xsd:documentation>This element represents a ballistic test event.</xsd:documentation>
     </xsd:annotation>
     <xsd:complexType>
      <xsd:sequence>
       <xsd:element name=”Source” minOccurs=”1” maxOccurs=”1”>
        <xsd:annotation>
         <xsd:documentation>This element represents the source for the test event data.</xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
         <xsd:simpleContent>
          <xsd:extension base=”xsd:string”>
           <xsd:attribute name=”BibtexRefID” type=”xsd:string” use=”required”>
            <xsd:annotation>
             <xsd:documentation>RefID used in Bibtex to reference this source.</xsd:documentation>
            </xsd:annotation>
           </xsd:attribute>
           <xsd:attribute name=”TestNumber” type=”xsd:string” use=”optional”>
            <xsd:annotation>
             <xsd:documentation>Test number used in the source document.</xsd:documentation>
            </xsd:annotation>
           </xsd:attribute>
           <xsd:attribute name=”Distribution” use=”required”>
            <xsd:annotation>
             <xsd:documentation>What is the releasability (Public, Limited, etc ...)?</xsd:documentation>
            </xsd:annotation>
            <xsd:simpleType>
             <xsd:restriction base=”xsd:string”>
              <xsd:enumeration value=”Unknown” />
              <xsd:enumeration value=”A. Approved for Public Release” />
              <xsd:enumeration value=”B. U.S. Government Agencies Only” />
              <xsd:enumeration value=”C. U.S. Government Agencies and Their Contractors” />
              <xsd:enumeration value=”D. DoD and DoD Contractors Only” />
              <xsd:enumeration value=”E. DoD Components Only” />
              <xsd:enumeration value=”F. Further Dissemination Only as Directed by the DoD Controlling Office or Higher DoD Authority” />
             </xsd:restriction>
            </xsd:simpleType>
           </xsd:attribute>
           <xsd:attribute name=”ExportControl” type=”xsd:boolean” use=”required”>
            <xsd:annotation>
             <xsd:documentation>Is the document export controlled?</xsd:documentation>
            </xsd:annotation>
           </xsd:attribute>
          </xsd:extension>
         </xsd:simpleContent>
        </xsd:complexType>
       </xsd:element>
       <xsd:element name="Impact" minOccurs="1" maxOccurs="1">
        <xsd:annotation>
         <xsd:documentation>This element defines the conditions at impact.</xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
         <xsd:sequence>
          <xsd:element name="Velocity" minOccurs="0" maxOccurs="1" type="Metric">
           <xsd:annotation>
            <xsd:documentation>Striking velocity of projectile.</xsd:documentation>
           </xsd:annotation>
          </xsd:element>
          <xsd:element name="TotalYaw" minOccurs="0" maxOccurs="1" type="Metric">
           <xsd:annotation>
            <xsd:documentation>Yaw of projectile at impact.</xsd:documentation>
           </xsd:annotation>
          </xsd:element>
          <xsd:element name="Yaw" minOccurs="0" maxOccurs="1" type="Metric">
           <xsd:annotation>
            <xsd:documentation>Yaw of projectile.</xsd:documentation>
           </xsd:annotation>
          </xsd:element>
          <xsd:element name="Pitch" minOccurs="0" maxOccurs="1" type="Metric">
           <xsd:annotation>
            <xsd:documentation>Pitch of projectile.</xsd:documentation>
           </xsd:annotation>
          </xsd:element>
         </xsd:sequence>
        </xsd:complexType>
       </xsd:element>
       <xsd:element name="Projectile" minOccurs="1" maxOccurs="1">
        <xsd:annotation>
         <xsd:documentation>This element defines the properties of the projectile in this test event.</xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
         <xsd:complexContent>
          <xsd:extension base="MaterialProperties">
           <xsd:sequence>
            <xsd:element name="TotalLength" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="EffectiveLength" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="CoreLength" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="CoreDiameter" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="Mass" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="Nose" minOccurs="0" maxOccurs="1" type="xsd:string" />
            <xsd:element name="Fineness" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="EffectiveFineness" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="NoseLength1" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="NoseConeAngle1" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="NoseDiameter" minOccurs="0" maxOccurs="1" type="Metric" />
            <xsd:element name="NoseLength2" minOccurs="0" maxOccurs="1" type="Metric" />
```

```
             <xsd:element name="NoseConeAngle2" minOccurs="0" maxOccurs="1" type="Metric" />
             <xsd:element name="CRH" minOccurs="0" maxOccurs="1" type="Metric" />
             <xsd:element name="NoseType" minOccurs="0" maxOccurs="1">
              <xsd:simpleType>
               <xsd:restriction base="xsd:string">
                <xsd:enumeration value="Flat" />
                <xsd:enumeration value="Conical" />
                <xsd:enumeration value="Bi-Conic" />
                <xsd:enumeration value="Frustrum" />
                <xsd:enumeration value="Blunt" />
                <xsd:enumeration value="Ogive" />
                <xsd:enumeration value="Hemispherical" />
               </xsd:restriction>
              </xsd:simpleType>
             </xsd:element>
            </xsd:sequence>
           </xsd:extension>
          </xsd:complexContent>
         </xsd:complexType>
        </xsd:element>
        <xsd:element name="Target" minOccurs="1" maxOccurs="1">
         <xsd:annotation>
          <xsd:documentation>This element defines the target array.</xsd:documentation>
         </xsd:annotation>
         <xsd:complexType>
          <xsd:sequence>
           <xsd:element name="Element" minOccurs="1" maxOccurs="unbounded">
            <xsd:annotation>
             <xsd:documentation>Defines the individual elements in the target array.</xsd:documentation>
            </xsd:annotation>
            <xsd:complexType>
             <xsd:complexContent>
              <xsd:extension base="MaterialProperties">
               <xsd:sequence>
                <xsd:element name="Thickness" minOccurs="1" maxOccurs="1" type="Metric" />
                <xsd:element name="Obliquity" minOccurs="1" maxOccurs="1" type="Metric" />
               </xsd:sequence>
               <xsd:attribute name="Number" type="xsd:int" use="required">
                <xsd:annotation>
                 <xsd:documentation>Which element in the array is this (starts at 1 and goes up)?</xsd:documentation>
                </xsd:annotation>
               </xsd:attribute>
               <xsd:attribute name="Type" type="xsd:string" use="required">
                <xsd:annotation>
                 <xsd:documentation>What type of material is this element (Ceramic, Metallic, Composite)?</xsd:documentation>
                </xsd:annotation>
               </xsd:attribute>
              </xsd:extension>
             </xsd:complexContent>
            </xsd:complexType>
           </xsd:element>
          </xsd:sequence>
          <xsd:attribute name="Elements" type="xsd:int" use="required">
           <xsd:annotation>
            <xsd:documentation>How many elements are in this target array?</xsd:documentation>
           </xsd:annotation>
```

```
  </xsd:attribute>
 </xsd:complexType>
</xsd:element>
<xsd:element name="Results" minOccurs="1" maxOccurs="1">
 <xsd:annotation>
  <xsd:documentation>This element defines the results from the test event.</xsd:documentation>
 </xsd:annotation>
 <xsd:complexType>
  <xsd:sequence>
   <xsd:element name="Penetration" minOccurs="0" maxOccurs="1" type="Metric">
    <xsd:annotation>
     <xsd:documentation>Defines the total penetration of the projectile through the entire target array.</xsd:documentation>
    </xsd:annotation>
   </xsd:element>
   <xsd:element name="ResidualVelocity" minOccurs="0" maxOccurs="1" type="Metric">
    <xsd:annotation>
     <xsd:documentation>Defines the final residual velocity of the projectile after perforating the entire target array.</xsd:documentation>
    </xsd:annotation>
   </xsd:element>
   <xsd:element name="ResidualLength" minOccurs="0" maxOccurs="1" type="Metric">
    <xsd:annotation>
     <xsd:documentation>Defines the final residual length of the projectile after perforating the entire target array.</xsd:documentation>
    </xsd:annotation>
   </xsd:element>
   <xsd:element name="ResidualMass" minOccurs="0" maxOccurs="1" type="Metric">
    <xsd:annotation>
     <xsd:documentation>Defines the final residual mass of the projectile after perforating the entire target array.</xsd:documentation>
    </xsd:annotation>
   </xsd:element>
   <xsd:element name="CraterDiameter" minOccurs="0" maxOccurs="1" type="Metric">
    <xsd:annotation>
     <xsd:documentation>Defines the diameter of the crater in the target after penetration of the projectile.</xsd:documentation>
    </xsd:annotation>
   </xsd:element>
   <xsd:element name="CraterVolume" minOccurs="0" maxOccurs="1" type="Metric">
    <xsd:annotation>
     <xsd:documentation>Defines the volume of the crater in the target after penetration of the projectile.</xsd:documentation>
    </xsd:annotation>
   </xsd:element>
   <xsd:element name="LimitVelocity" minOccurs="0" maxOccurs="1" type="Metric">
    <xsd:annotation>
     <xsd:documentation>Defines the limit velocity of the projectile against the target.</xsd:documentation>
    </xsd:annotation>
   </xsd:element>
  </xsd:sequence>
 </xsd:complexType>
</xsd:element>
```

```
      <xsd:element name="Notes" minOccurs="0" maxOccurs="1" type="xsd:string">
       <xsd:annotation>
        <xsd:documentation>Use this element to document any important notes about this test event data.</xsd:documentation>
       </xsd:annotation>
      </xsd:element>
     </xsd:sequence>
     <xsd:attribute name="ID" type="xsd:string" use="required">
      <xsd:annotation>
       <xsd:documentation>This is the identifier for this Test.</xsd:documentation>
      </xsd:annotation>
     </xsd:attribute>
     <xsd:attribute name="Type" use="required">
      <xsd:annotation>
       <xsd:documentation>This defines the type of test that this is.</xsd:documentation>
      </xsd:annotation>
      <xsd:simpleType>
       <xsd:restriction base="xsd:string">
        <xsd:enumeration value="SNI" />
        <xsd:enumeration value="SNF" />
        <xsd:enumeration value="MAI" />
        <xsd:enumeration value="MAF" />
        <xsd:enumeration value="MNI" />
        <xsd:enumeration value="MNF" />
        <xsd:enumeration value="SNL" />
        <xsd:enumeration value="MNL" />
        <xsd:enumeration value="MAL" />
       </xsd:restriction>
      </xsd:simpleType>
     </xsd:attribute>
     <xsd:attribute name="Suspect" type="xsd:boolean" use="required">
      <xsd:annotation>
       <xsd:documentation>This is used to mark the test as a suspect.</xsd:documentation>
      </xsd:annotation>
     </xsd:attribute>
    </xsd:complexType>
   </xsd:element>
  </xsd:sequence>
 </xsd:complexType>
</xsd:element>
<!-- Parent complexTypes -->
<xsd:complexType name="MaterialProperties">
 <xsd:sequence>
  <xsd:element name="Material" minOccurs="1" maxOccurs="1" type="xsd:string" />
  <xsd:element name="Hardness" minOccurs="0" maxOccurs="1" type="Metric" />
  <xsd:element name="Density" minOccurs="1" maxOccurs="1" type="Metric" />
  <xsd:element name="Ductility" minOccurs="0" maxOccurs="1" type="Metric" />
  <xsd:element name="YoungsModulus" minOccurs="0" maxOccurs="1" type="Metric" />
  <xsd:element name="YieldStrength" minOccurs="0" maxOccurs="1" type="Metric" />
  <xsd:element name="UltimateTensileStrength" minOccurs="0" maxOccurs="1" type="Metric" />
  <xsd:element name="PoissonRatio" minOccurs="0" maxOccurs="1" type="Metric" />
  <xsd:element name="Toughness" minOccurs="0" maxOccurs="1" type="Metric" />
 </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Metric">
 <xsd:simpleContent>
  <xsd:extension base="xsd:double">
   <xsd:attribute name="Units" type="xsd:string" use="optional">
    <xsd:annotation>
     <xsd:documentation>Units of measure for metric.</xsd:documentation>
     </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="Pedigree" use="required">
     <xsd:annotation>
      <xsd:documentation>What is the pedigree of the data metric (As Reported, Unit Conversion, etc...)?</xsd:documentation>
     </xsd:annotation>
     <xsd:simpleType>
      <xsd:restriction base="xsd:string">
       <xsd:enumeration value="Reported" />
       <xsd:enumeration value="Converted" />
       <xsd:enumeration value="Estimated" />
       <xsd:enumeration value="Imputed" />
       <xsd:enumeration value="Surrogated" />
       <xsd:enumeration value="NA" />
      </xsd:restriction>
     </xsd:simpleType>
    </xsd:attribute>
   </xsd:extension>
  </xsd:simpleContent>
 </xsd:complexType>
</xsd:schema>
```

## B.2 PDML Design Diagrams

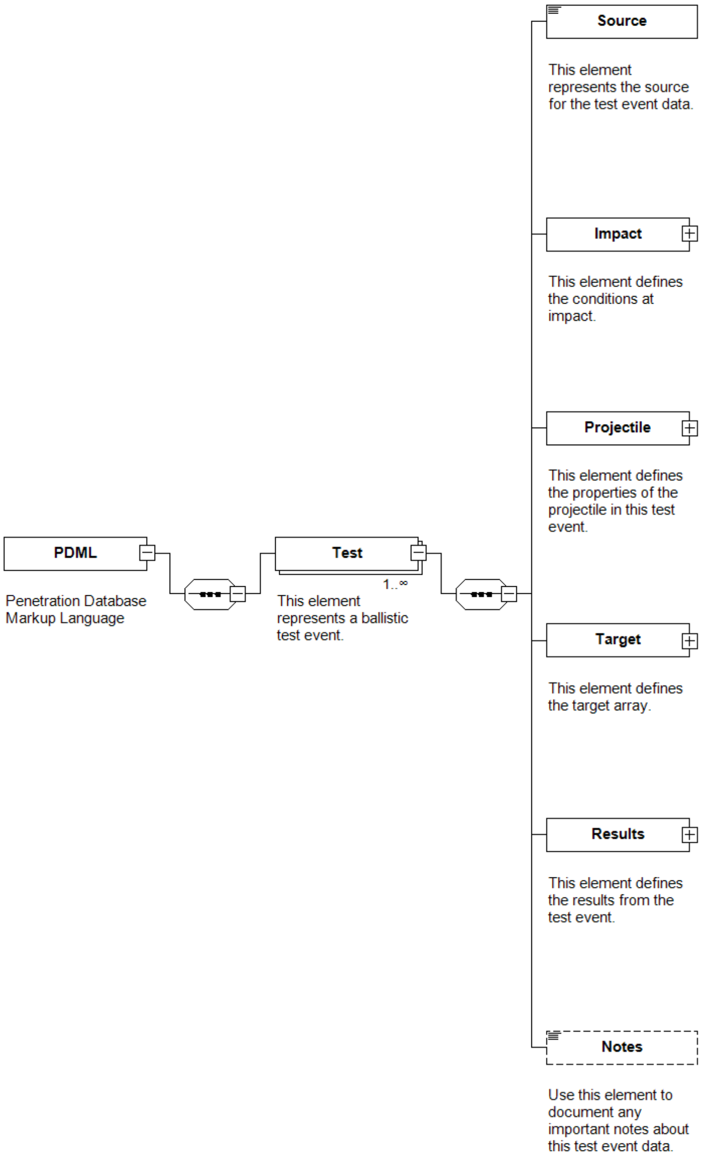

Source
This element represents the source for the test event data.
Impact
This element defines the conditions at impact.
Projectile
This element defines the properties of the projectile in this test event.
PDML
Penetration Database Markup Language
Test
1..∞
This element represents a ballistic test event.
Target
This element defines the target array.
Results
This element defines the results from the test event.
Notes
Use this element to document any important notes about this test event data.

Figure B.1: A top level diagram of the PDML

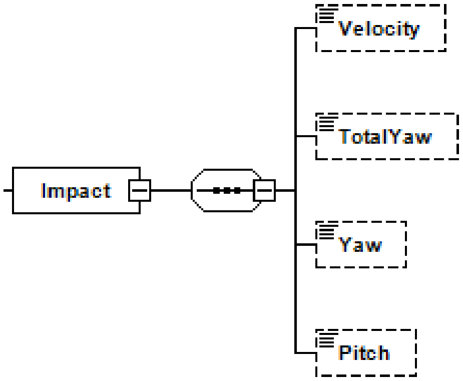


Figure B.2: A diagram of the Impact element from the PDML

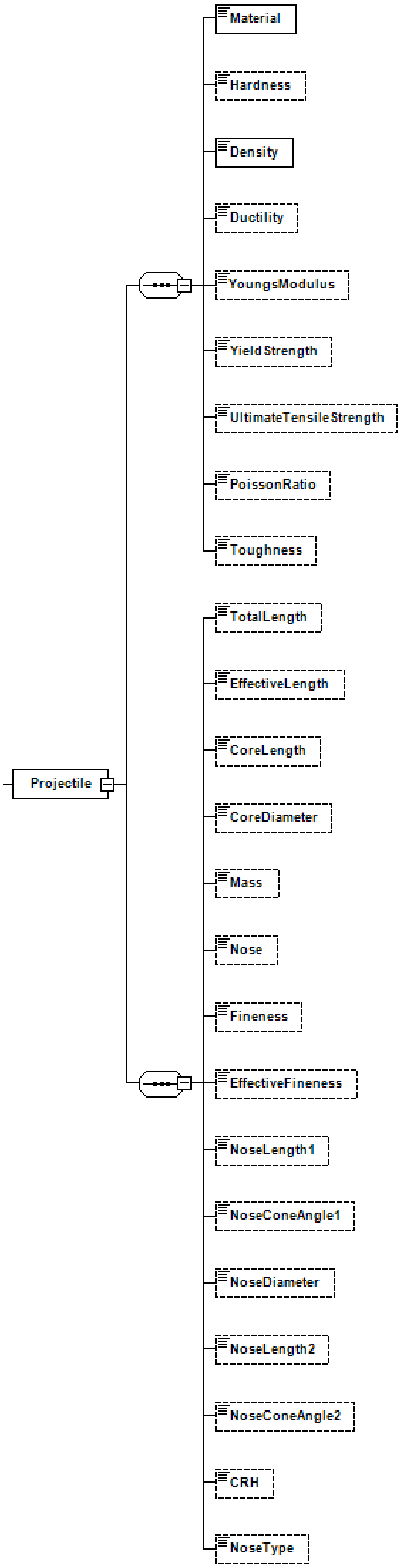


Figure B.3: A diagram of the Projectile element from the PDML

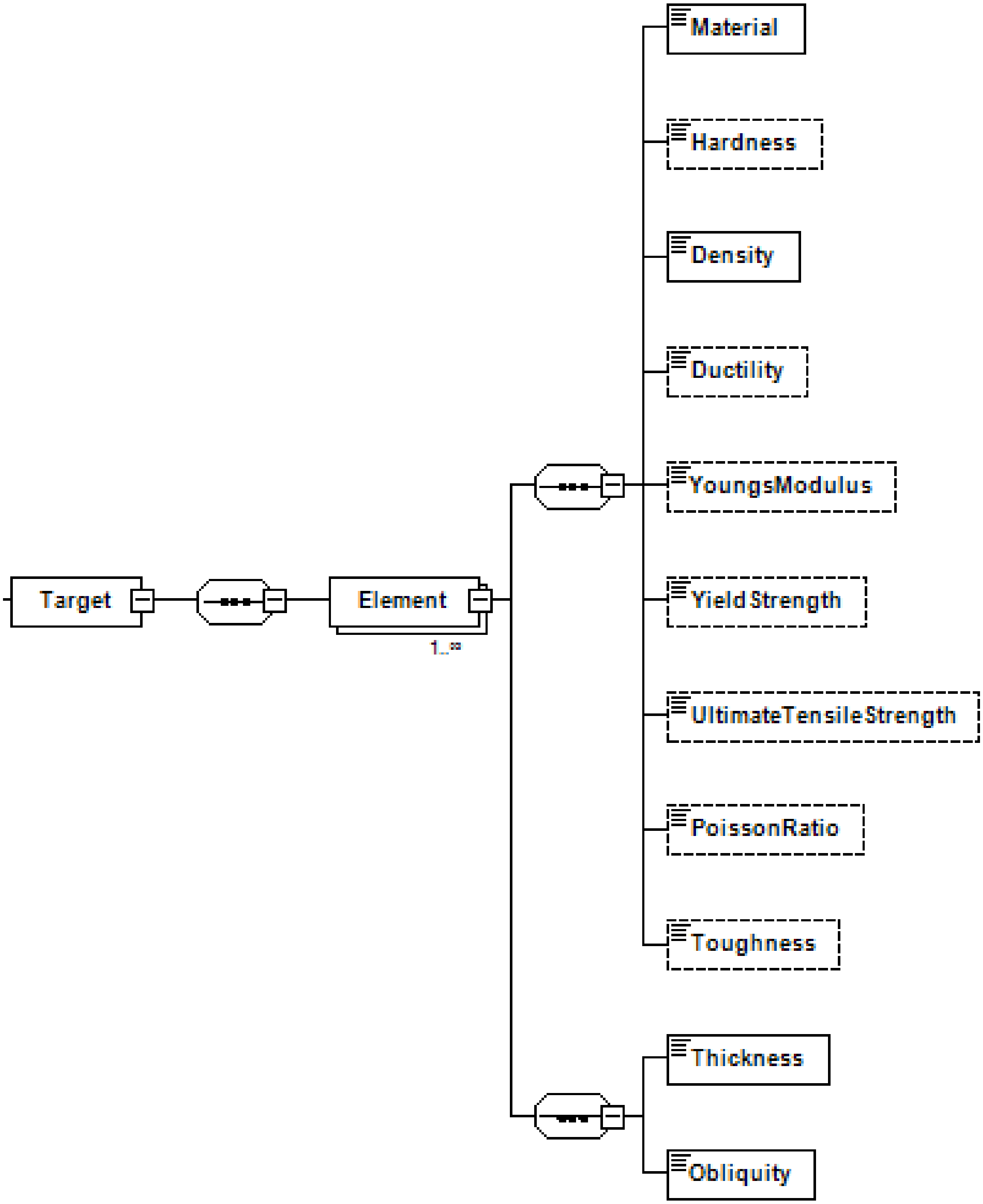


Figure B.4: A diagram of the Target element from the PDML

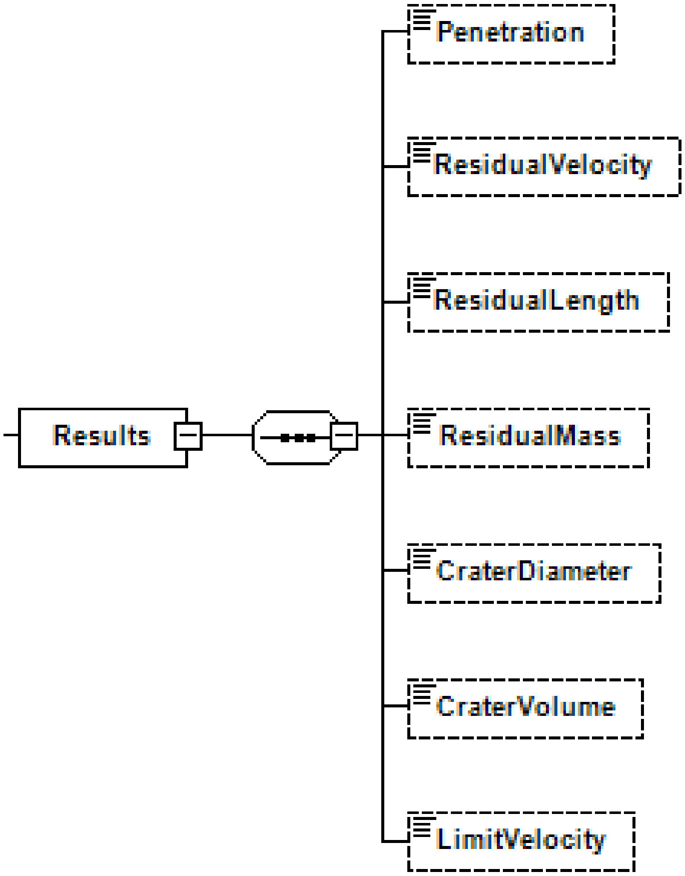


Figure B.5: A diagram of the Results element from the PDML