Use of Machine Learning Techniques to Estimate System Performance

15th Annual NATO Operations Research and Analysis Conference

Mr. James Randall Way

Computer Scientist

U.S. Army Futures Command, DEVCOM Data & Analysis Center

18 Oct 2021

Approved for public release

PROBLEM STATEMENT #1

DEVCOM DATA & ANALYSIS CENTER

- Entity-level combat simulations, such as the Infantry Warrior Simulation (IWARS), require probability of incapacitation (P(I)) estimates for fragmenting munitions that depend on target posture, target body armor, casualty criterion, firer-to-target range, burst height, burst-to-target azimuth angle, and burst-to-target range.

- This data is provided in large lookup tables, but using such large tables can slow down the simulation, and sometimes the tables are too large to fit into the simulation's database.

Munition (about 11)
M67
M384
M397A1
M406
etc…

×

Posture
Standing
Crouching
Kneeling
Prone
Foxhole

×

Clothing, Vest, Helmet
Winter, Interceptor #3, ACH
Winter, PASGT, PASGT
Winter, none, none
Winter, none, PASGT

×

Casualty Criterion
30SA
30SD
5MA
5MD

×

Grid Size (Firer-to-Target Ranges, Burst Heights, Burst-to-Target Az. Angles, Burst-to-Target Ranges)
5,000 to 250,000

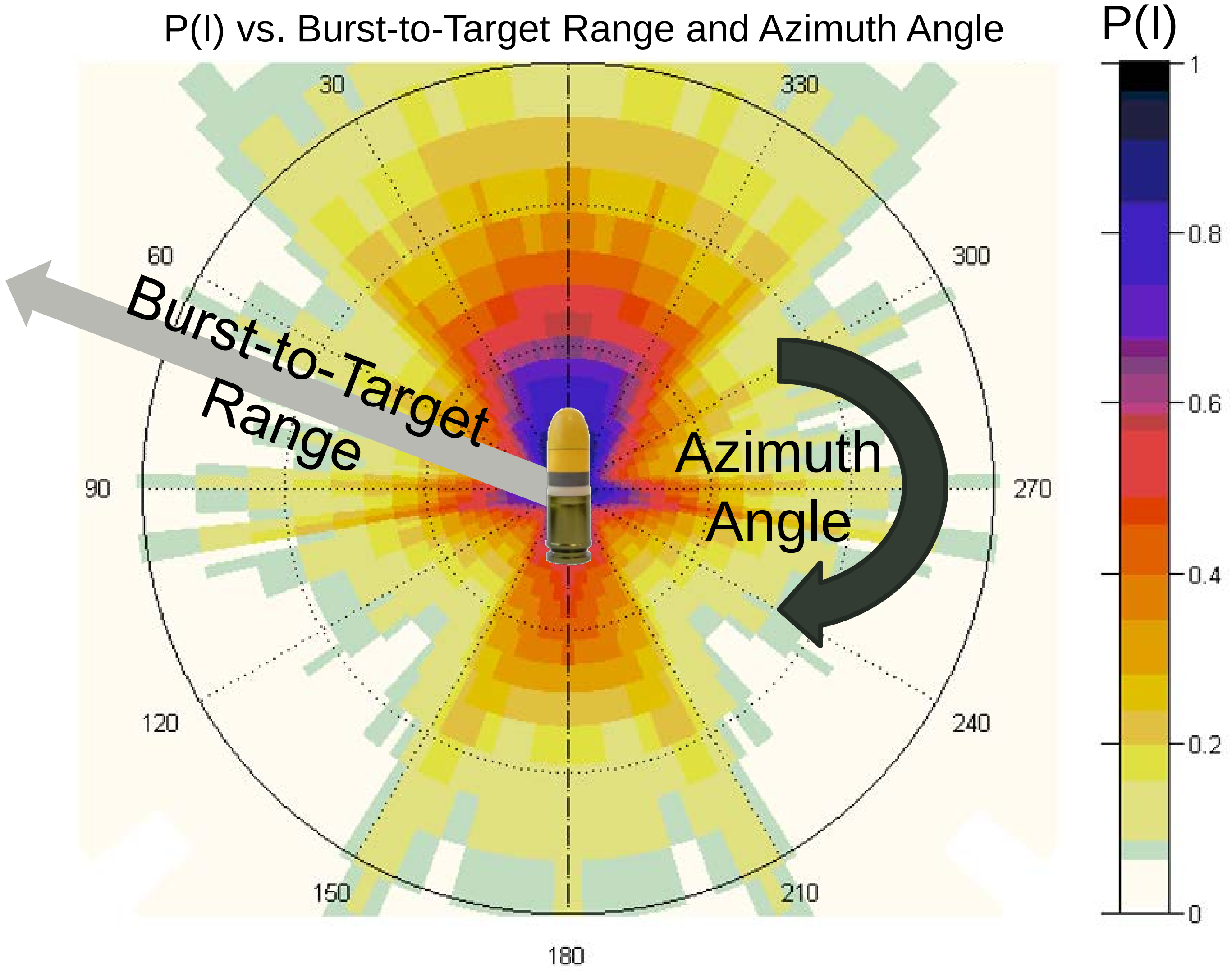
=

Number of P(I) Estimates
76,000,000

Munition	# Firer-to Target Ranges	# Burst Heights	# Burst-to-Target Azimuth Angles	# Burst-to-Target Ranges	Grid Size
Munition #1	7	9	91	41	235,053
Munition #2	1	1	91	61	5,551
Munition #3	8	1	91	35	25,480
Munition #4	3	2	91	41	22,386

DATA VISUALIZATION

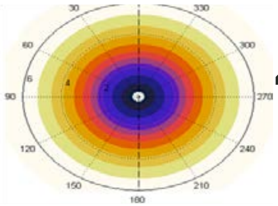
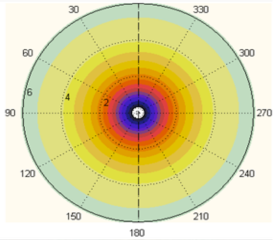
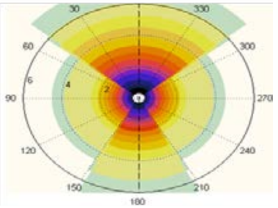
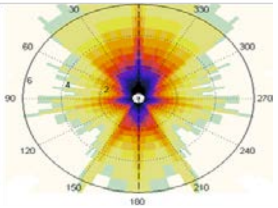
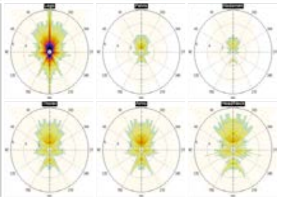
- The P(I) data for just one firer-to-target range and burst height can be complicated.



CURRENT SOLUTIONS

- None of these solutions is an optimal balance of fidelity and compression.

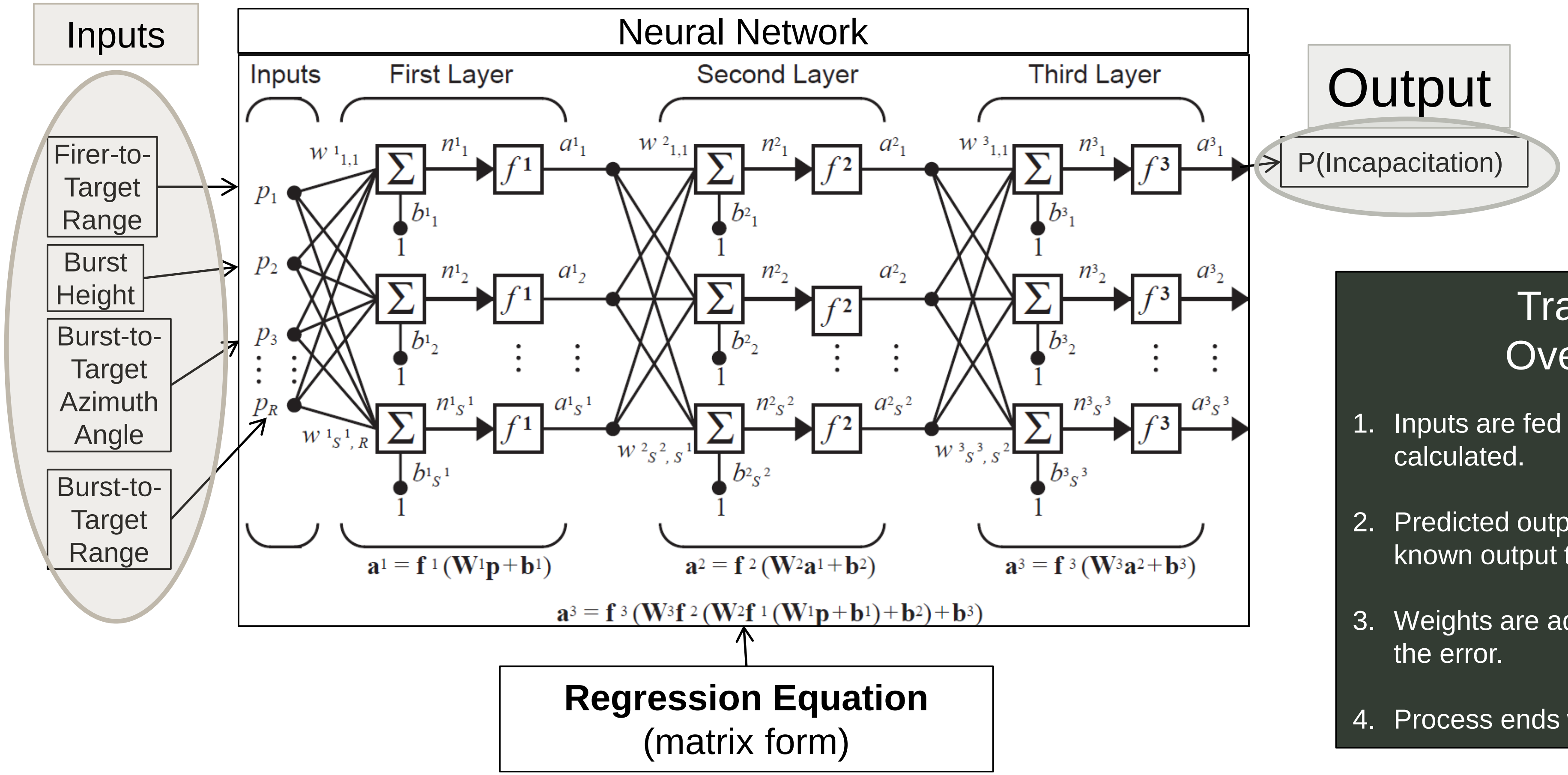
Increasing Fidelity

Name	Description	Example
Carleton Damage Function	Assumes P(I) drops off exponentially: $P(I) = D_0 * \exp(-\pi * D_0 * r^2 / A_L)$, where D_0 is zero-range P(I), A_L is lethal area, and r is range.	
PIVR (Probability of Incapacitation vs. Range)	P(I) vs. miss-distance (burst-to-target range). Does not model azimuth angle.	
PIVR_Angle (P(I) vs. Range and Angle)	P(I) vs. miss-distance and 3 angular zones. Requires judgment in defining zones.	
Whole-Body Grid	P(I) vs. miss-distance and 91 azimuth angles.	
By-Body-Part Grid	P(I) vs. miss-distance and 91 azimuth angles and 6 body-parts. Can be used to improve the modeling of partial cover.	

Increasing Compression

NEURAL NETWORK SOLUTION

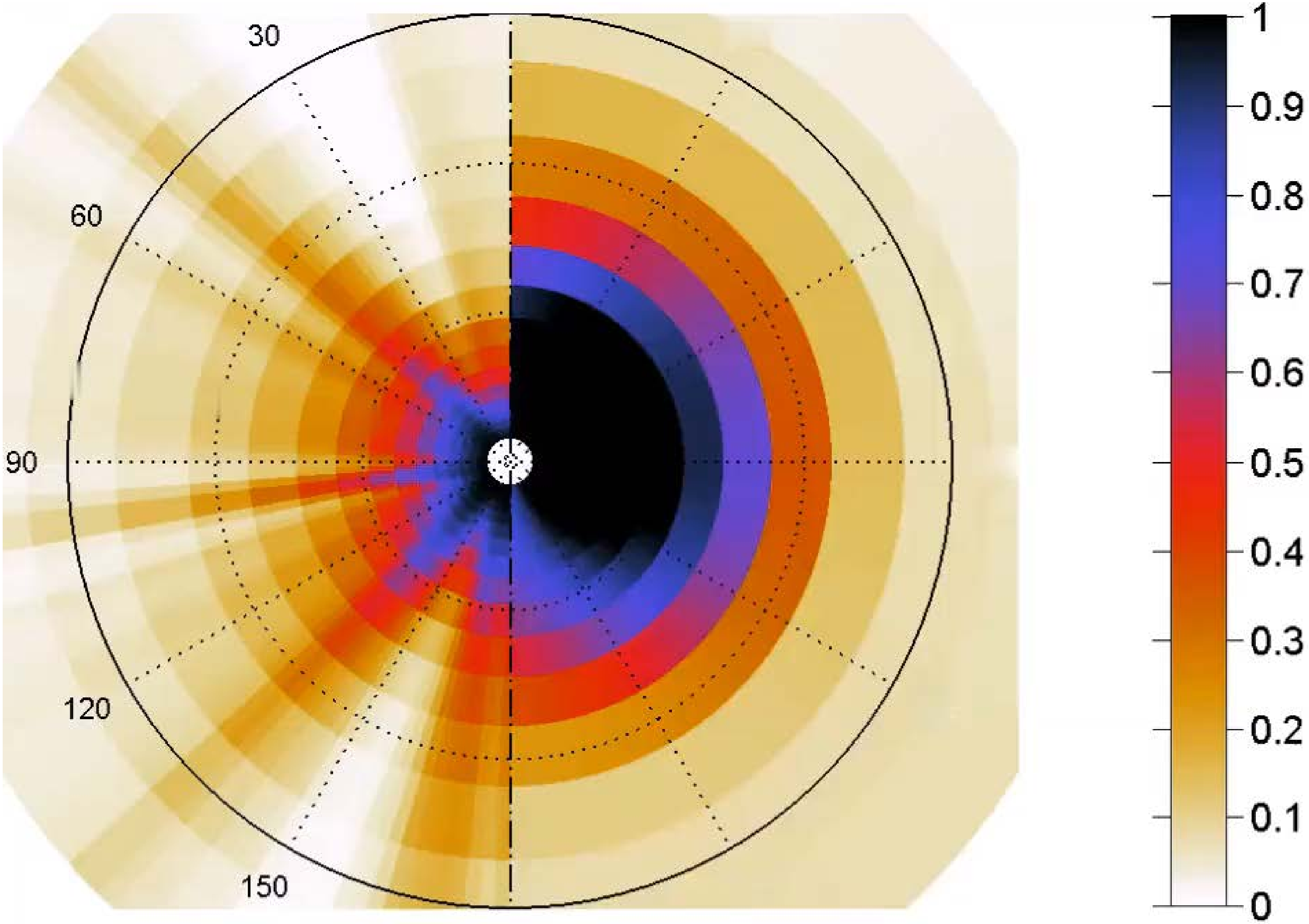
- A neural network is a ***regression equation*** with many free parameters.
- A neural network can, in *theory*, approximate any reasonable function arbitrarily well.
- If the size of the network is less than the size of the training data, the result is data compression.



Training Overview

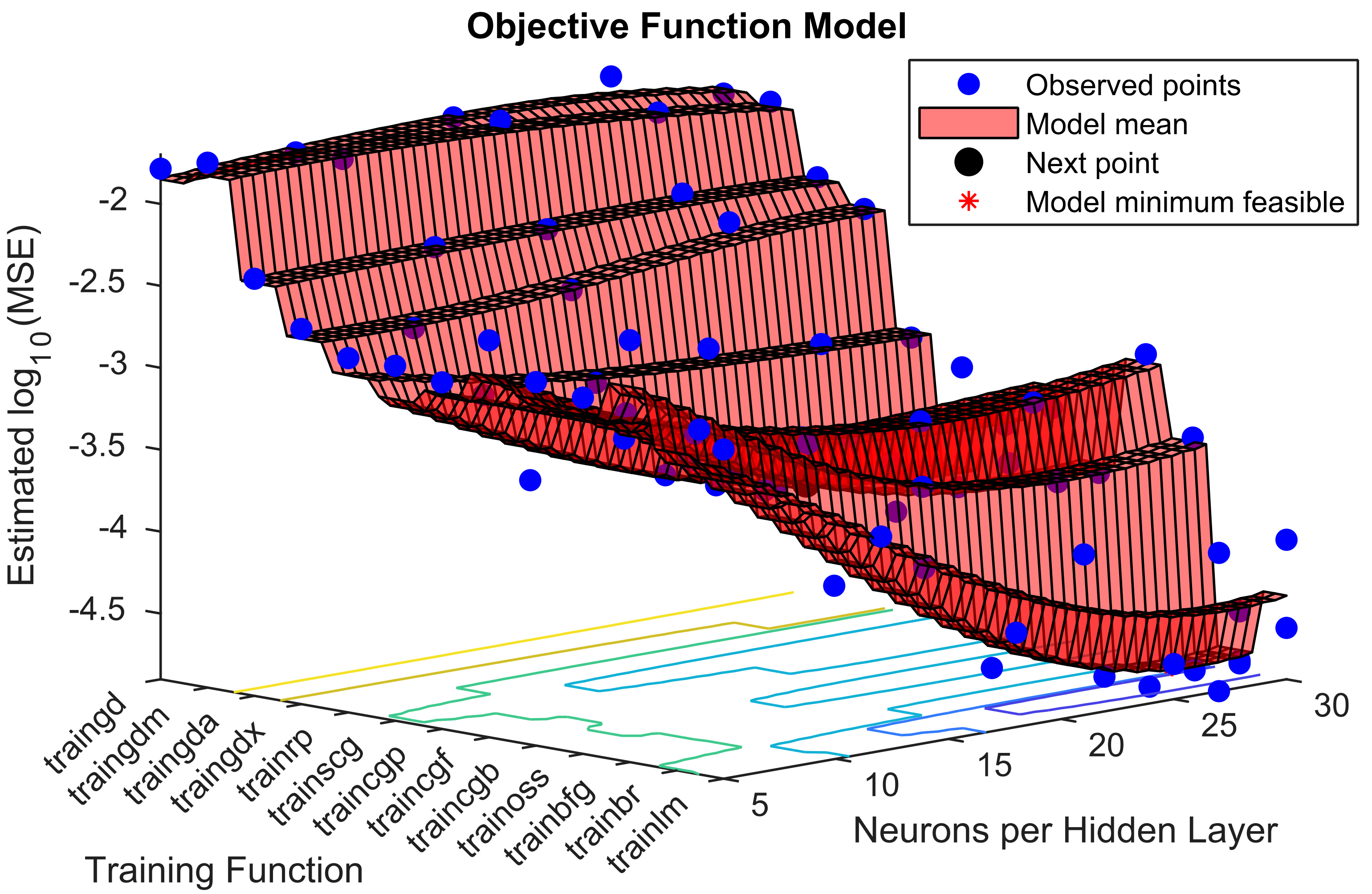
1. Inputs are fed into net and output is calculated.
2. Predicted output is compared to known output to calculate the error.
3. Weights are adjusted to decrease the error.
4. Process ends when error is "small".

TRAINING IN ACTION



BAYESIAN OPTIMIZATION

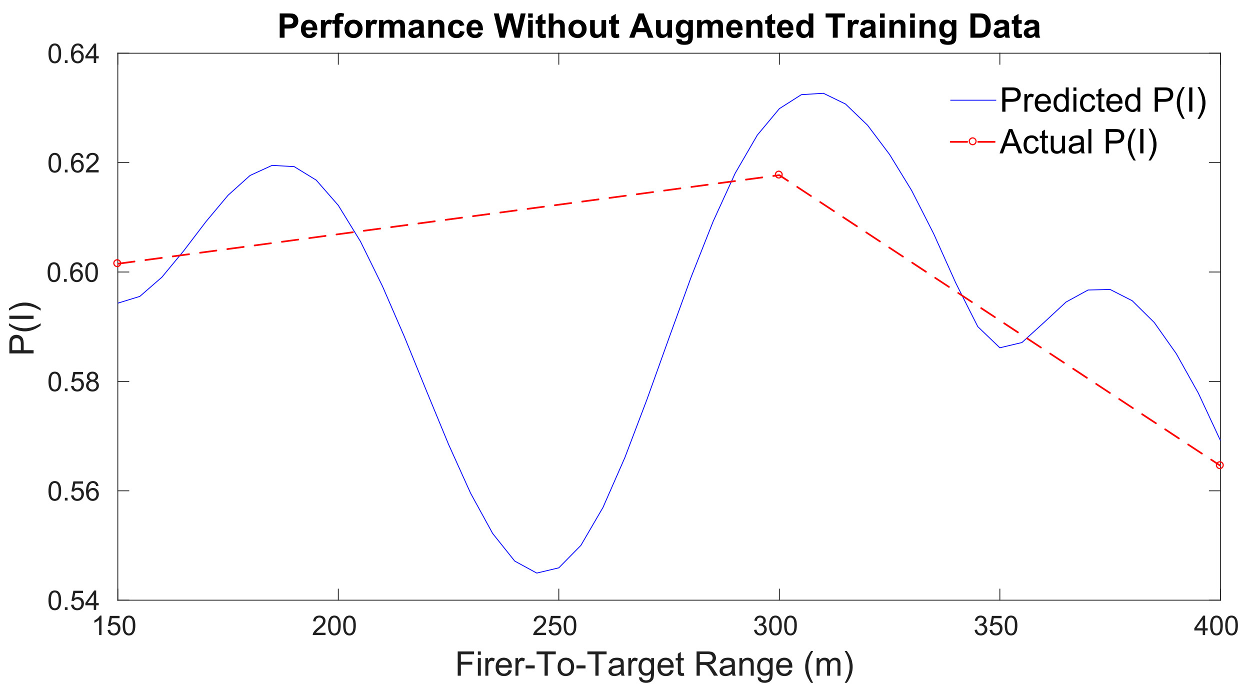
- Mean squared error (MSE) can vary significantly depending on the training function and network parameters.

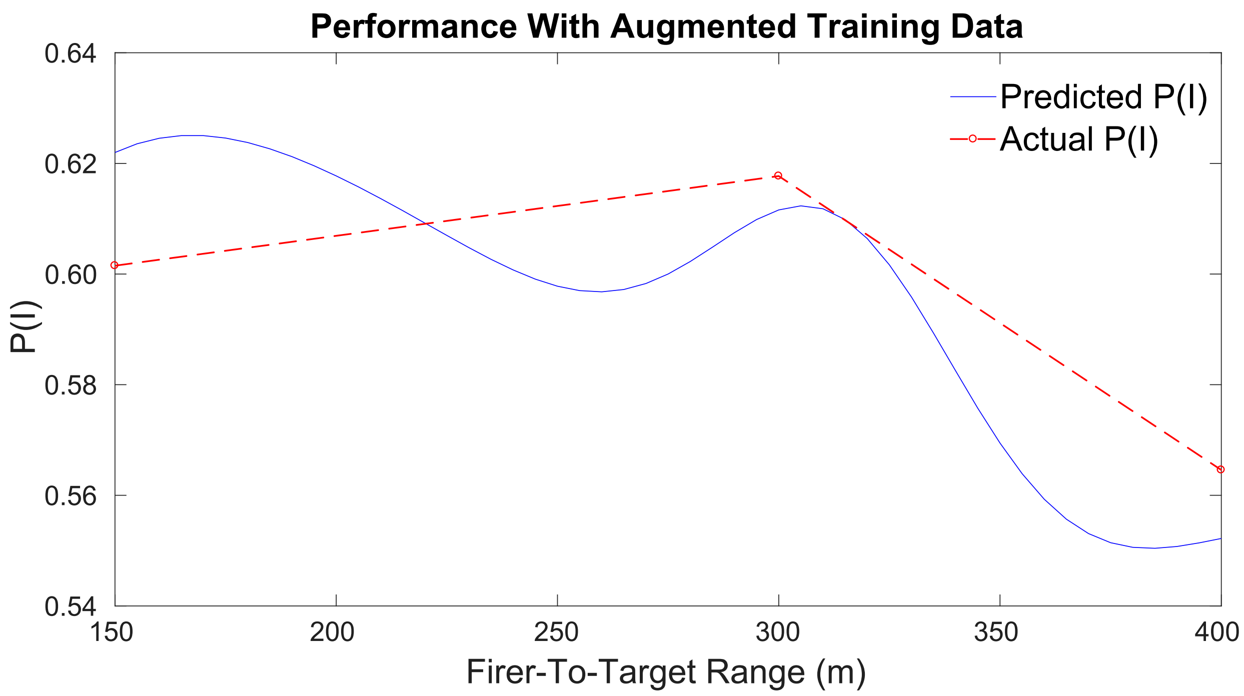


GENERALIZATION AND DATA AUGMENTATION

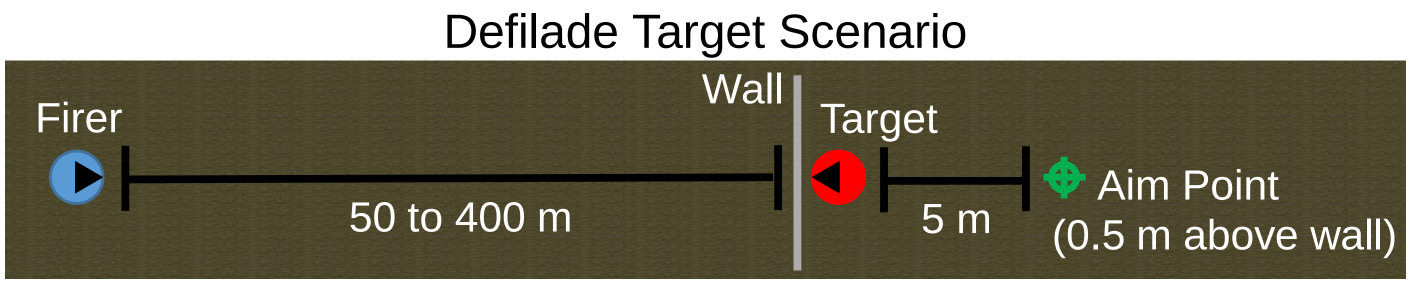
- The 66,339 P(I) values used for training in this example were for 3 firer-to-target ranges, 9 burst heights, 27 burst-to-target ranges, and 91 azimuth angles.

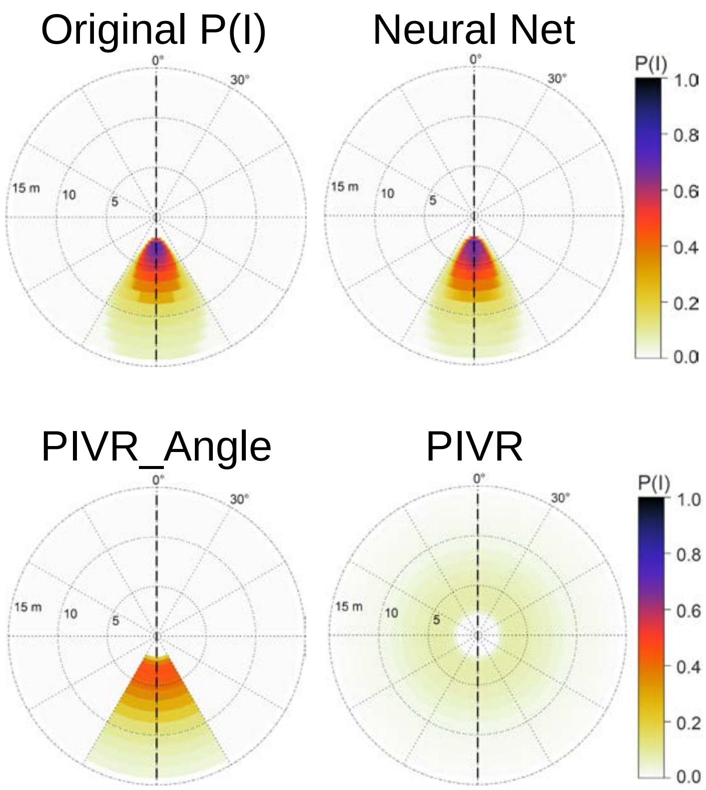
- To improve generalization, a new network was trained on linearly interpolated P(I) values for 18 firer-to-target ranges and 18 burst heights.

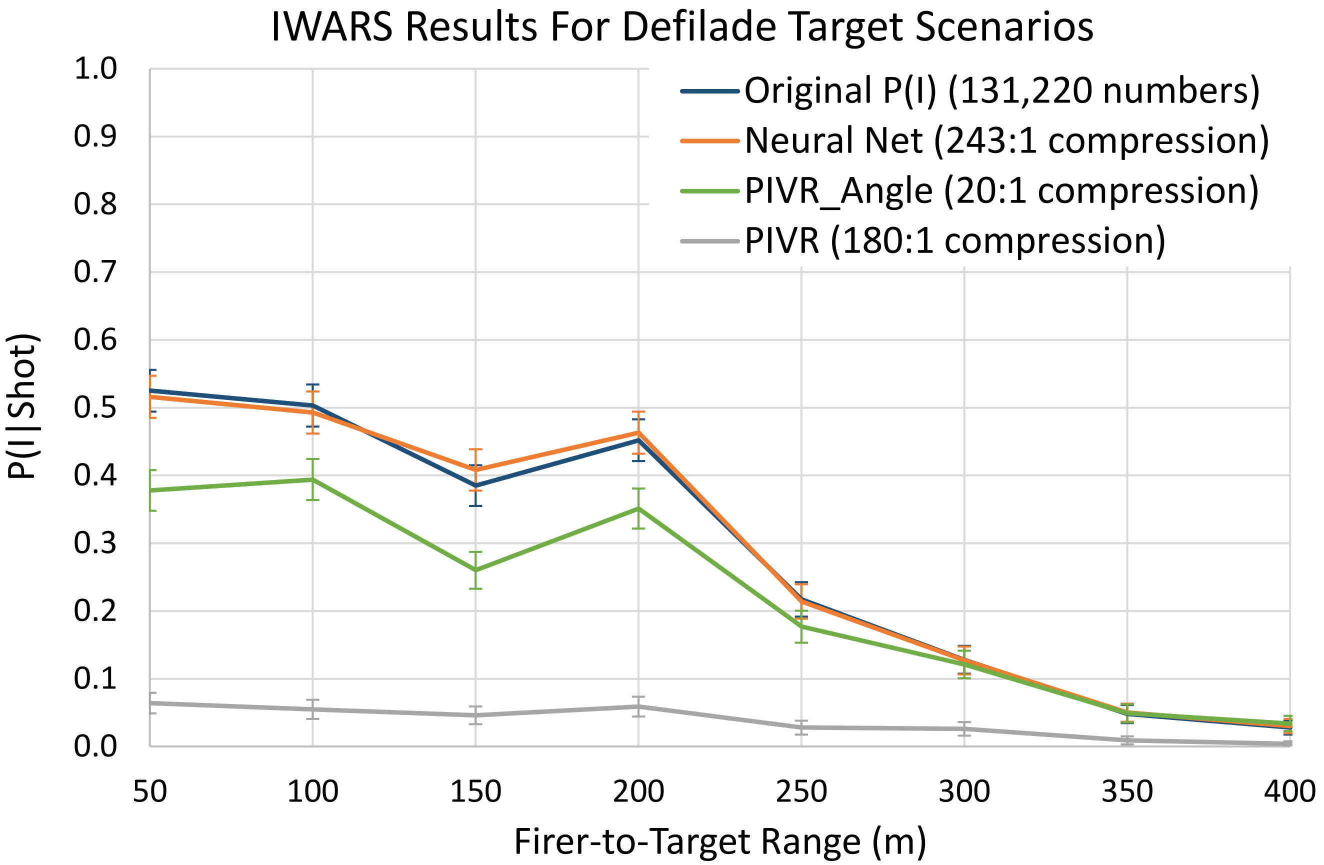




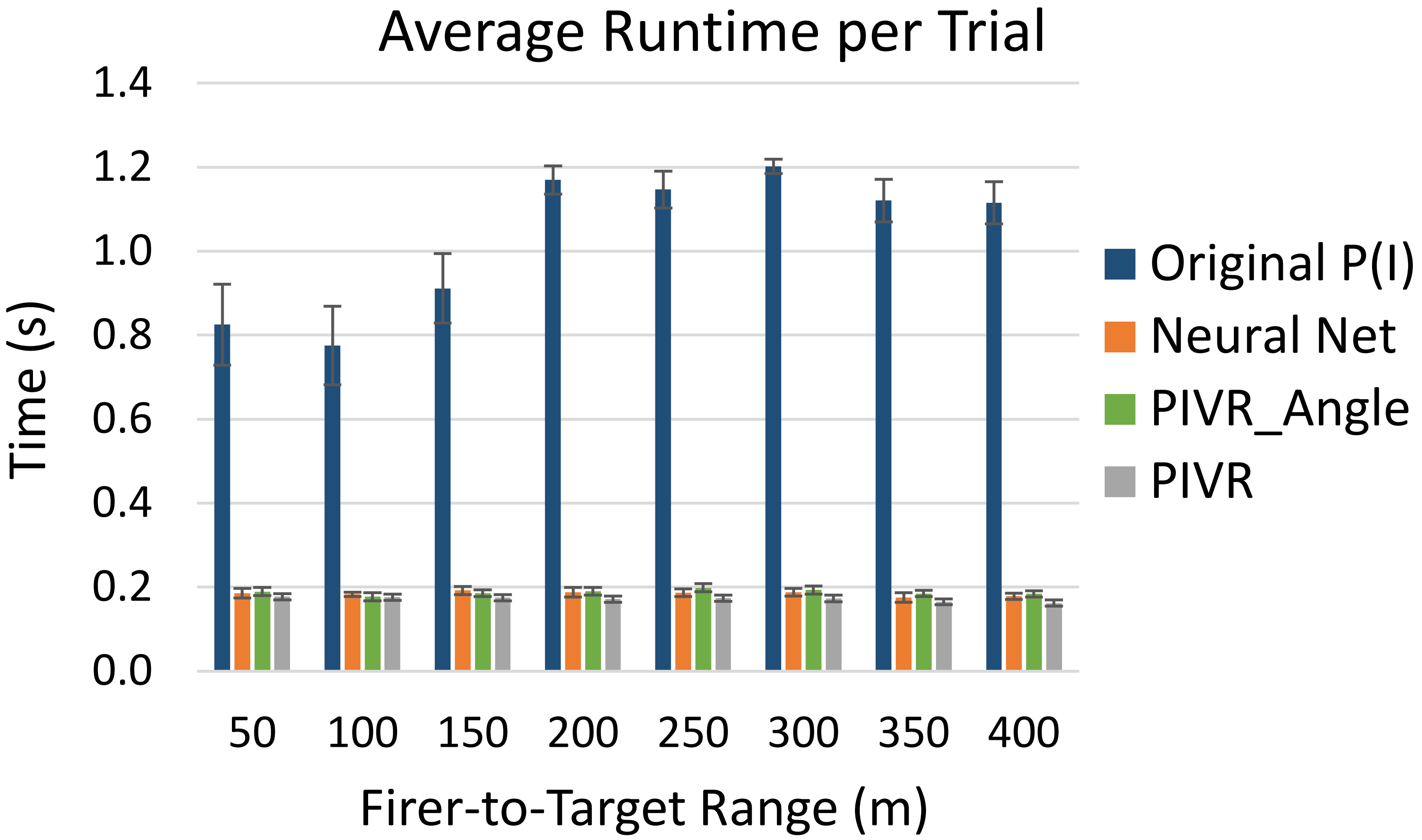
EXAMPLE SCENARIO RESULTS







EXAMPLE SCENARIO RESULTS



PROBLEM STATEMENT #2

- When performance data is not available for a system, subject-matter experts (SMEs) may provide data for another, similar system as a surrogate.

- SMEs may have to surrogate thousands of data items for just one data request.
  This can be tedious and error-prone.

SOLUTION

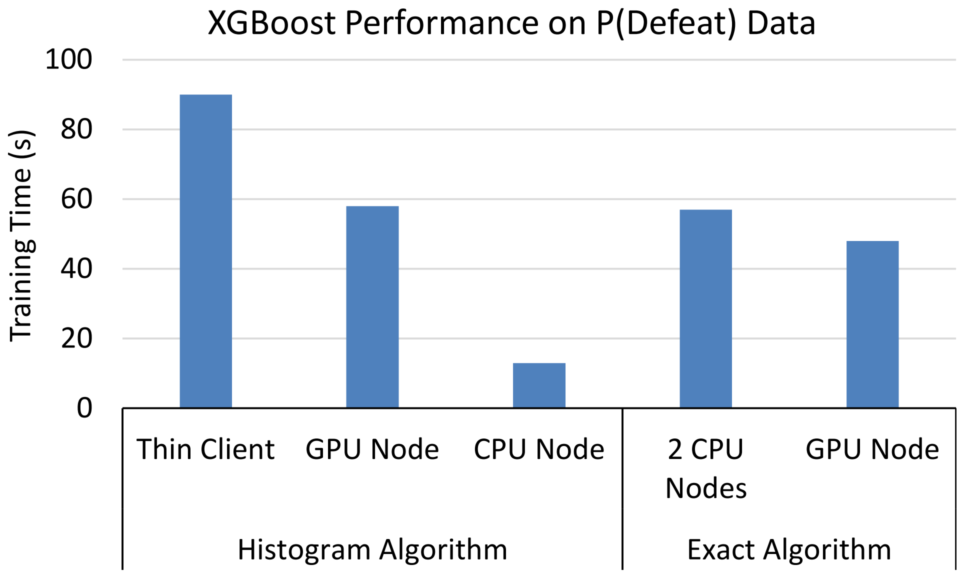
- Trained a gradient boosted decision tree model to predict the best available system/weapon/mount/munition/target pairing to use as a surrogate for a desired system/weapon/mount/munition/target pairing.
  - Used LightGBM with one classifier per output (e.g., available system and available munition are two different outputs).
  - Distinct output values were treated as classes (e.g., M4, M16, and M16A2 rifles were treated as separate classes).
  - Tuned model hyperparameters using randomized search with stratified 3-fold cross-validation

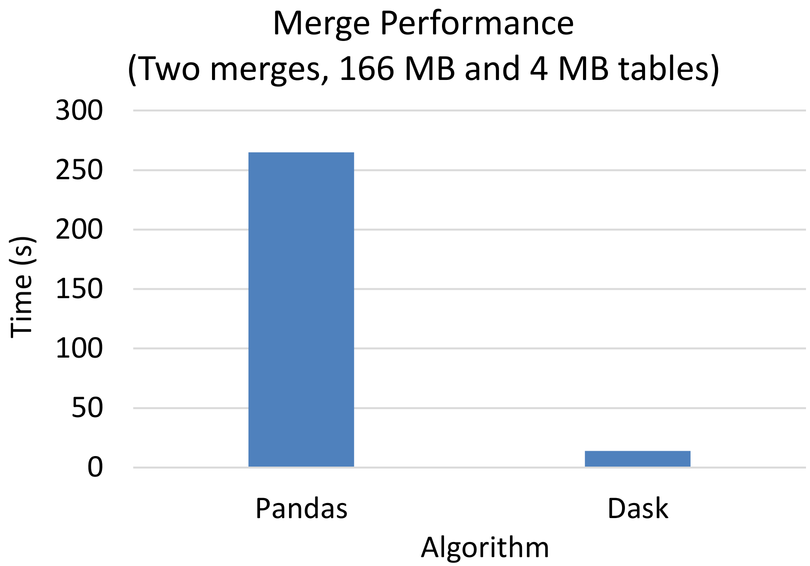
- Achieved high prediction accuracy as shown in the table below. These results are for a small arms “Probability of Kill given a Shot” (PKS) data request with 1,013 pairings.

Feature	Top-1 Accuracy	Top-3 Accuracy	Training Time (min)
System	90%	96.0%	2.2
Weapon	96%	99.4%	1.6
Mount	94%	99.6%	0.4
Munition	96%	100.0%	2.1
Target	87%	96.0%	7.6

LESSONS LEARNED

- HPC speedup enables cross-validation and hyperparameter optimization.





- High dimensionality, sparseness, missing values, and “messiness” make the Equipment Characteristics Database (ECDB) a difficult dataset to learn from.
  - 103 munition characteristics ranging from ‘Cone_Liner_Angle’ to ‘Frequency_Band’.
  - No M855A1 munition; several other fielded munitions were missing.
  - Different names describe the same characteristic, such as ‘Num_Submunitions’ and ‘Number_Submunitions’.
  - Typographical errors such as ‘0’ instead of ‘O’.
  - Multi-value fields such as ‘Altitude_Band’ in the format ‘MinHeight – MaxHeight’.
  - Inconsistent units such as some ‘Explosive_Weight’ values in pounds and others in kilograms.
  - Missing units, inconsistent labeling of ‘n/a’ and unknown data, values assigned to the wrong field, etc…