Companion Read Me to Paper: Validating Machine Learned Diagnostic Classifiers

Daniel Wade and Andrew Wilson, PhD

US Army AMRDEC, AED

# Abstract

Data science techniques such as machine learning are rapidly becoming available to engineers building models from system data, such as aircraft operations data. These techniques require validation for use in fielded systems providing recommendations to operators or maintainers. The methods for validating and testing machine learned algorithms generally focus on model performance metrics such as accuracy or F1-score. Many aviation datasets are highly imbalanced, which can invalidate some underlying assumptions of machine learning models. Two simulations are performed to show how some common performance metrics respond to imbalanced populations. The results show that each performance metric responds differently to a sample depending on the imbalance ratio between two classes. The results indicate that traditional methods for repairing underlying imbalance in the sample may not provide the rigorous validation necessary in safety critical applications.

# Objective

The objective of this document is to inspire other researchers and engineers to examine and understand the underlying assumptions made when a method is chosen for selecting a machine-learned model for an application.

What is being released is a python notebook version of the following two procedures described in the companion paper:

The imbalance simulation follows this procedure:

1. Define the initial imbalance ratio, RI, humans to zombies, e.g. 250:1.
2. Define the fixed sample size, Ns, the number of total subjects in a sample.
3. Define an imbalance change function to iterate over, e.g. RIn = RIn-2 + RIn-1.
4. Determine which model validation metrics should be computed, e.g. informedness and weighted accuracy.
5. Call a simulation function that can be iterated.
6. Pull subjects into the sample with random diagnostic values (Dvz and Dvh) such that the number of zombies (Nz) and humans (Nh) summed equals the sample size (Ns).
7. Determine all possible threshold values for the sample.
8. Find each threshold value that maximizes each of the metrics.
9. Save the threshold and confusion matrix for each of the metrics.
10. Repeat the process 1,000 times for the same imbalance ratio (Step 4).
11. Increment the imbalance ratio (Step 3) until the imbalance ratio goes to 1:1.
12. Plot mean TPR, FPR, Total Error Rate, and Threshold Value as a function of imbalance.

The sample size simulation follows the same procedure with the following underlined changes:

1. Define the initial sample size, Ns.
2. Define the fixed imbalance ratio, RI.
3. Define a sample size increase function to iterate over.
4. Determine which model validation metrics should be computed.
5. Call a simulation function that can be iterated.
6. Pull subjects into the sample with random diagnostic values such that Nz + Nh = Ns.
7. Determine all possible threshold values for the sample.
8. Find each threshold value that maximizes each of the metrics.
9. Save the threshold and confusion matrix for each of the metrics.
10. Repeat the process 1,000 times for the same sample size (Step 4).
11. Increment the sample size (Step 3) up to 72,000.
12. Plot mean TPR, FPR, Total Error Rate, and Threshold Value as a function of sample size.