An Investigation of Feature Engineering Approaches and Strategies

Andrew Lentz, Matthew A. Lanham

Purdue University Krannert School of Management

lentz3@purdue.edu; lanhamm@purdue.edu

Abstract

This focus of this study is to test the effectiveness of several feature engineering approaches and strategies on multi classification datasets. The motivation for this investigation is to determine if specific feature engineering approaches are beneficial for a variety of modeling techniques. Understanding best practices for feature engineering can help professionals and researchers alike extract the most out of their data in classification settings. We will set up a workflow to systematically test the effect of 5 feature engineering approaches on 18 multi classification datasets using 4 predictive modeling techniques. The workflow will be done in R using the h2o package to control the variability in the models and to parallelize the workload. After preprocessing the data, a loop will be run to test each model on each data set in a controlled setting followed by applying each feature engineering approach to each data set and testing the models again in the same environment. We will then assess the effectiveness of each engineering approach on each predictive modeling technique. We propose this method will allow us to generalize "industry standard" approaches for specific feature engineering approaches on specific predictive models.

Introduction

The rise of data-driven decision making and predictive analytics in nearly every corner of industry has created a need for a more standardized approach to data modeling. Having an "Industry Standard" approach to particular machine learning algorithms regarding data processing and feature engineering can result in streamlining the workflow of modeling problems and increasing accuracy. This can save time and money, allowing for these resources to be used in other areas of a project. More accurately diagnosing disease, more accurate revenue predictions, and more correct classifications of malicious software are all examples of benefitting from maximizing results from data. Maximizing model performance has the potential to increase revenues, margins, and save lives.

Our aim was to answer the following questions:

Are there universal feature engineering approaches that should be tried on classification problems?

Are there approaches that should be tried on specific modeling techniques?

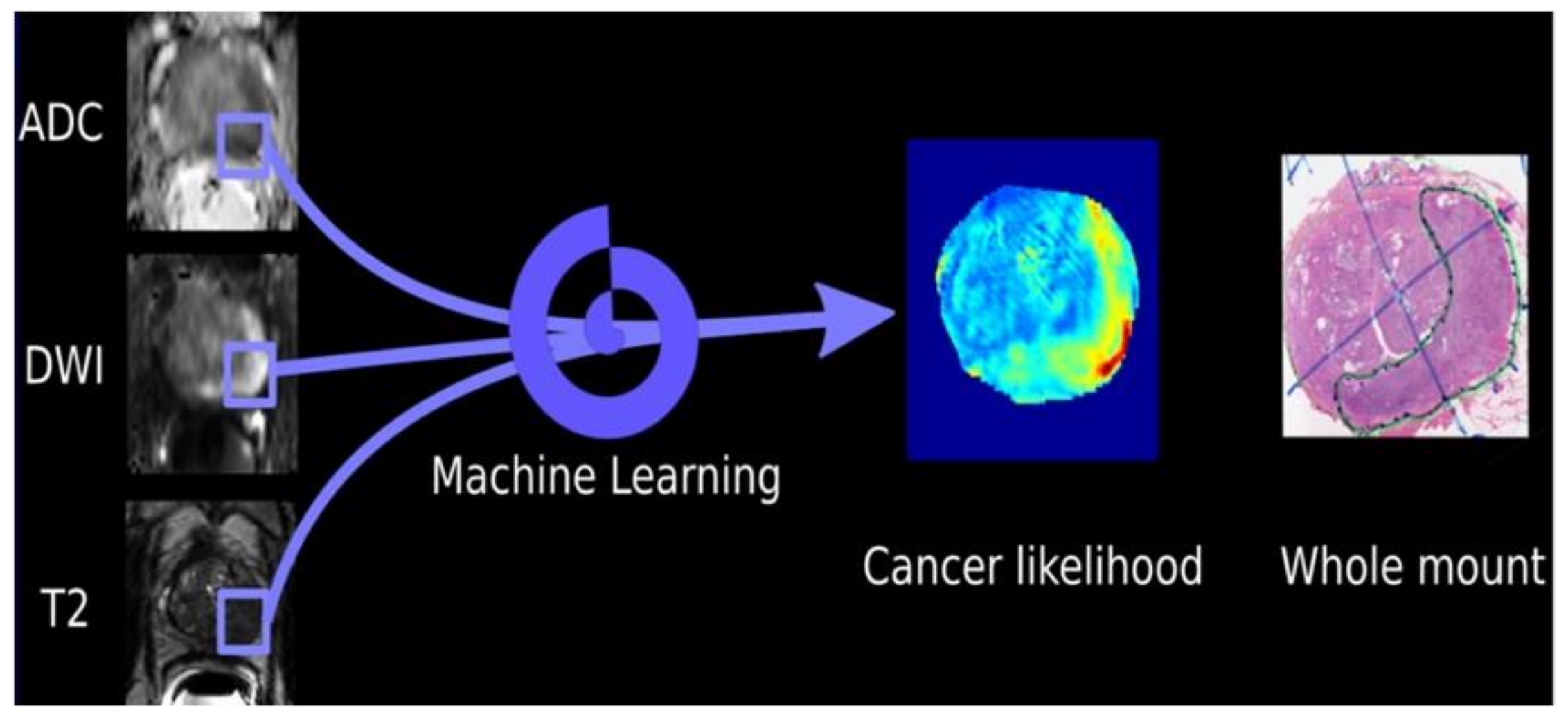

Figure 1. Machine learning classifier for cancer patients can be improved from feature engineering to save lives

Literature Review

Current research in this area is focused mostly on speed or on improving results on specific individual data sets, rather than discovering a standardized approach across data sets.

"Materialization Optimizations for Feature Selection Workloads," a research study done by faculty at University of Wisconsin-Madison and Stanford focused on uncovering approaches to speed up selection of features. The study uncovered methods to increase the selection process by nearly two-fold.

Students from the National Taiwan University's "Feature Engineering and Classifier Ensemble for KDD Cup 2010" focused on finding the optimal feature engineering approaches for the KDD Cup 2010.

Additionally, "Brainwash: A Data System for Feature Engineering," a study done by faculty at University of Wisconsin-Madison, and University of Michigan focused on minimizing the time between iterations of feature engineering processes.

Methodology

Figure 2 outlines our study design, starting from data collection, data cleaning, data pre-processing, feature creation and selection, model/approach selection, cross-validation design, and model assessment/performance measures.

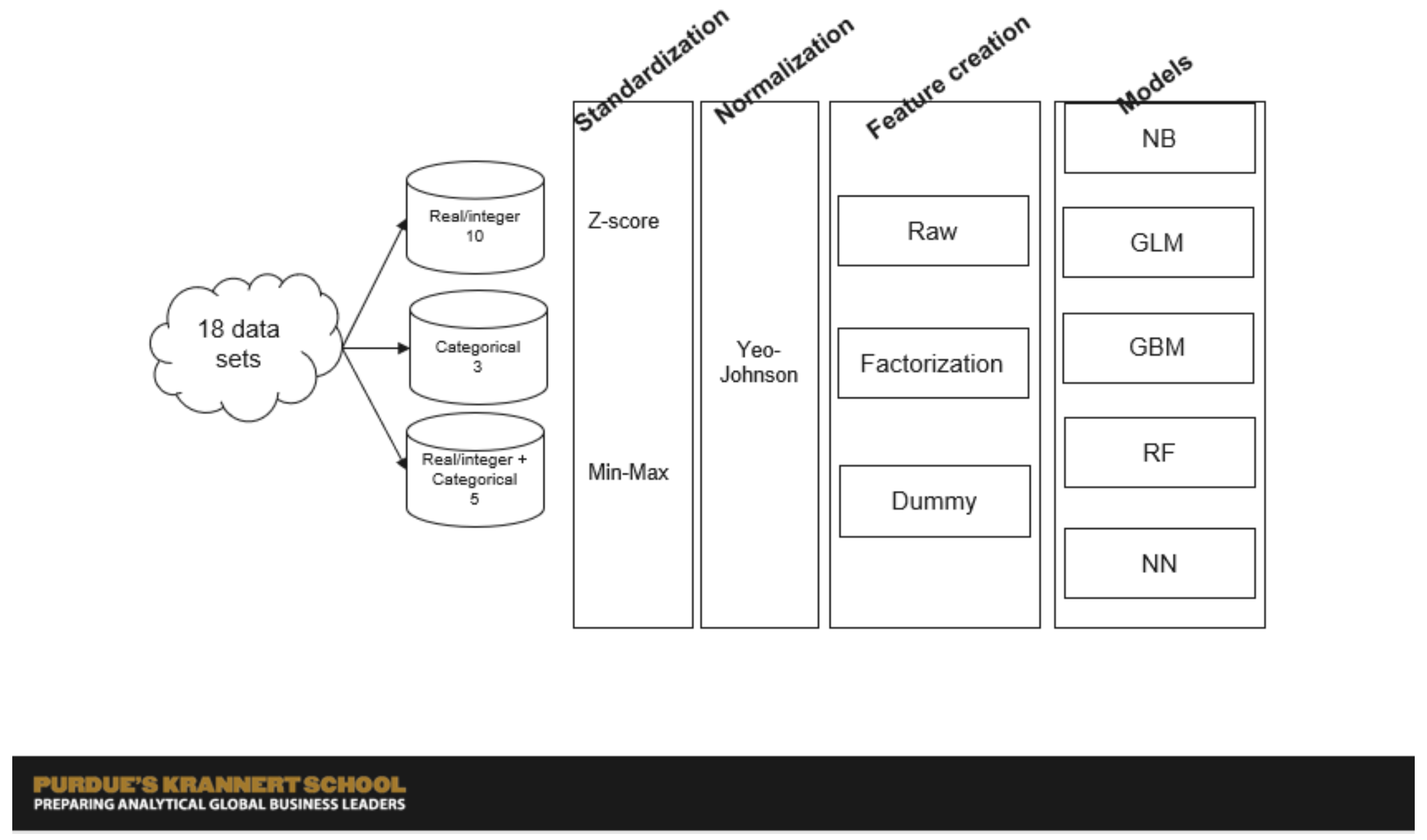

Figure 2. Study Design

Data

The data are several sets of data from the UCI Machine Learning Repository. It has been scraped into R.

Data Cleaning & Pre-Processing

The data was cleaned by removing observations with missing data as well as removing outliers in the data.

Methodology (Approach) Selection

The focus of this analysis was to systematically test the effect of several feature engineering approaches on several machine learning algorithms. The data from the 18 UCI repositories underwent similar feature engineering approaches, such as normalization, standardization, a mix of normalization and standardization, and creating dummy features from categorical features along with maintaining the raw data sets as a control. This created 126 data sets to be tested with five machine learning algorithms which included Neural Network, Random Forest, Gradient Boosting Machine, and Naïve Bayes.

Model Evaluation / Statistical & Business Performance Measures

The performance of the models was measured by MSE, RMSE, and Log-Loss. These results were added to a data frame to track the performance of each data set/feature engineering/model combination to compare against the control.

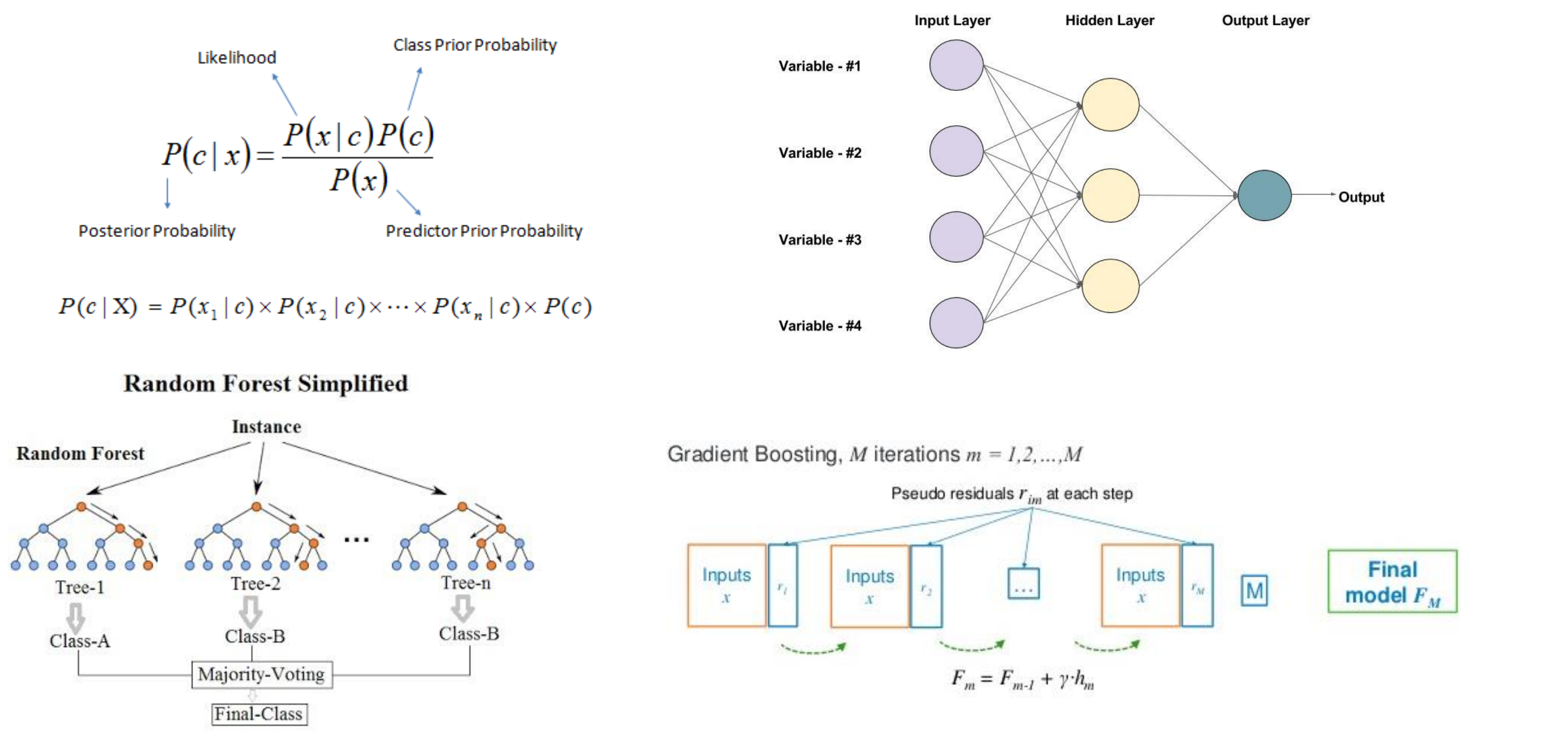

Figure 3. Visual examples of the models being used

Results

The results are visualized below. The random forest algorithm benefitted from dummy variable creation across all performance metrics. Interestingly enough, standardization/normalization of numeric features seemed to harm results for the random forest when compared against the control data sets.

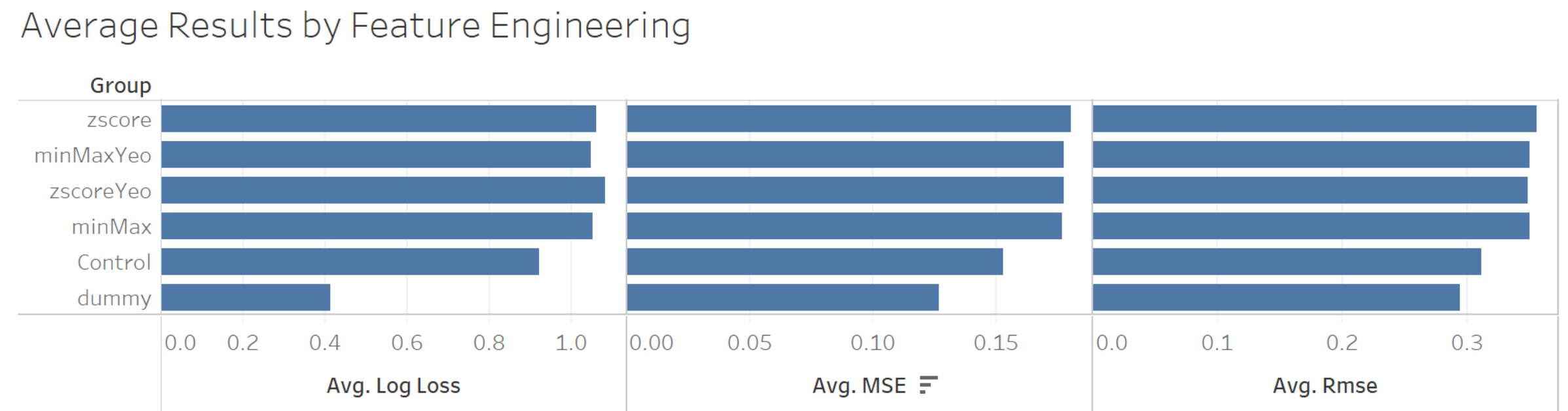

Figure 4. Random Forest Results

Similar results were found for gradient boosting machines.

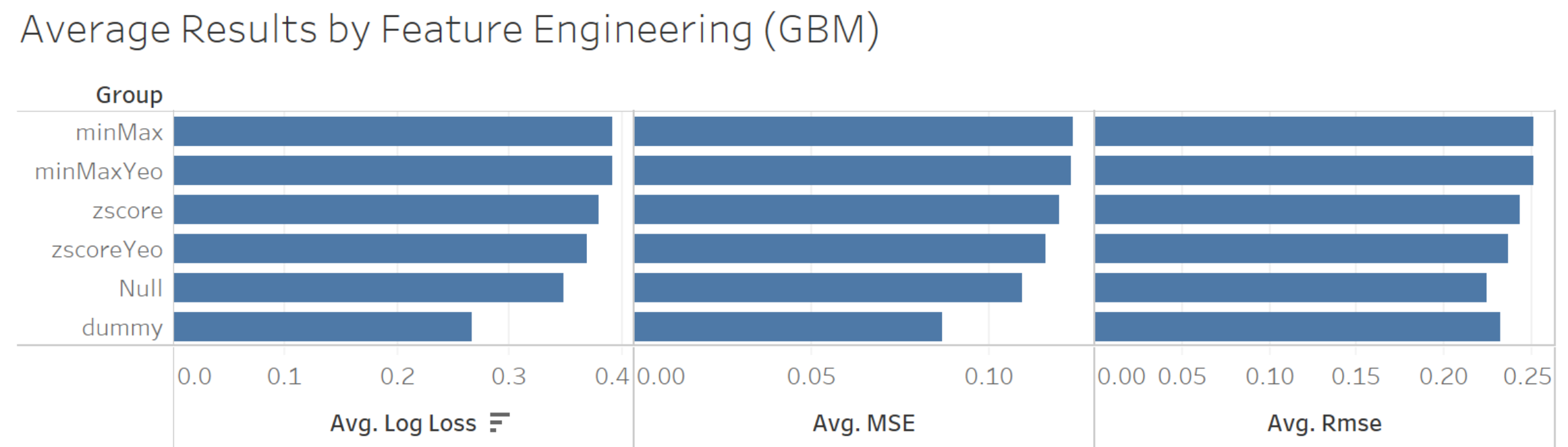

Figure 5. GBM Results

The results for the neural network appear to be mixed. There was not a consistent feature engineering approach that benefitted the performance metrics across all data sets. In actuality, each approach both improved results and harmed results compared to controls depending on the data set. The results also suggest naïve bayes modeling does not consistently benefit from a feature engineering approach with the raw data sets producing the best performance on average.

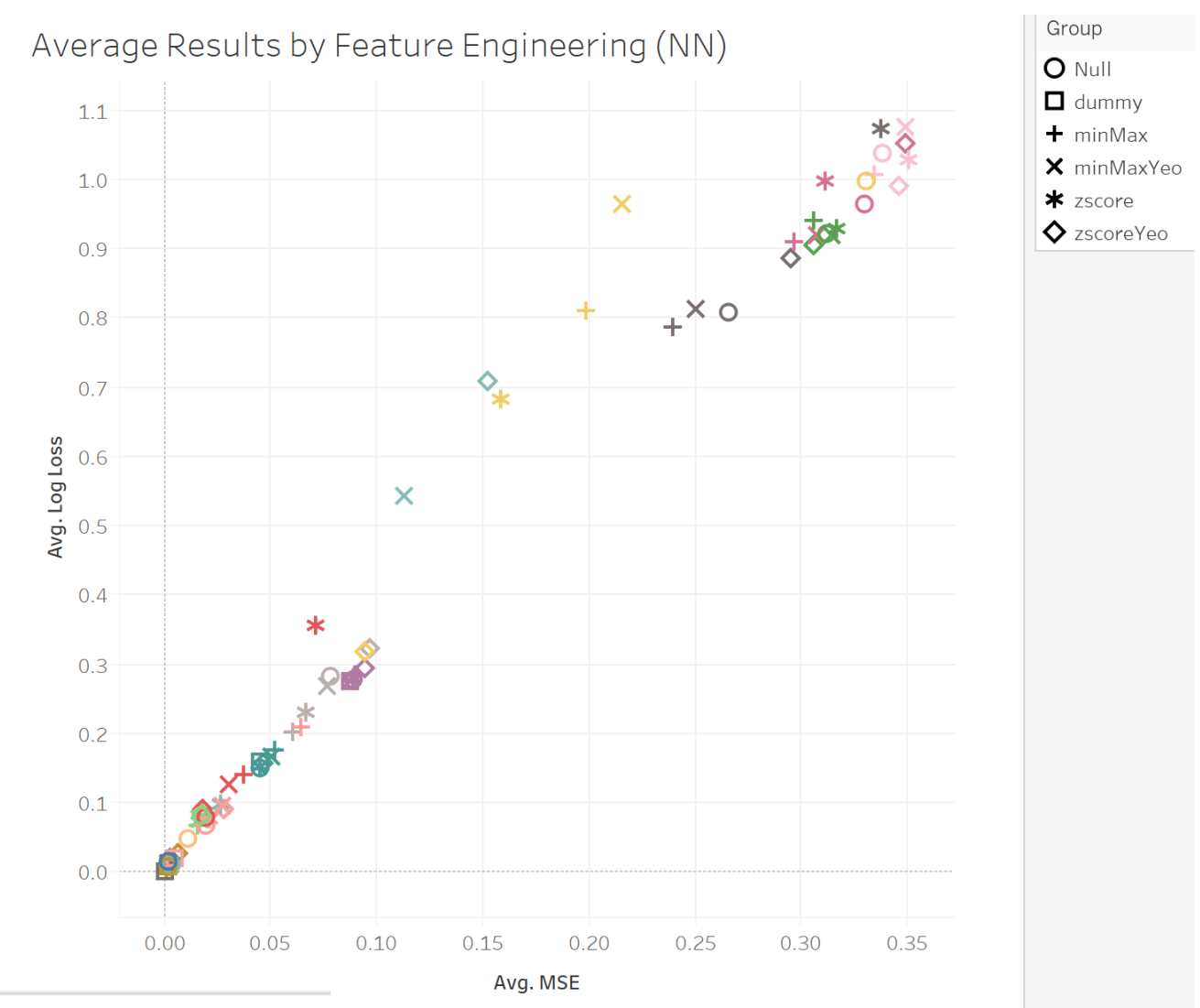

Figure 6. NN Results

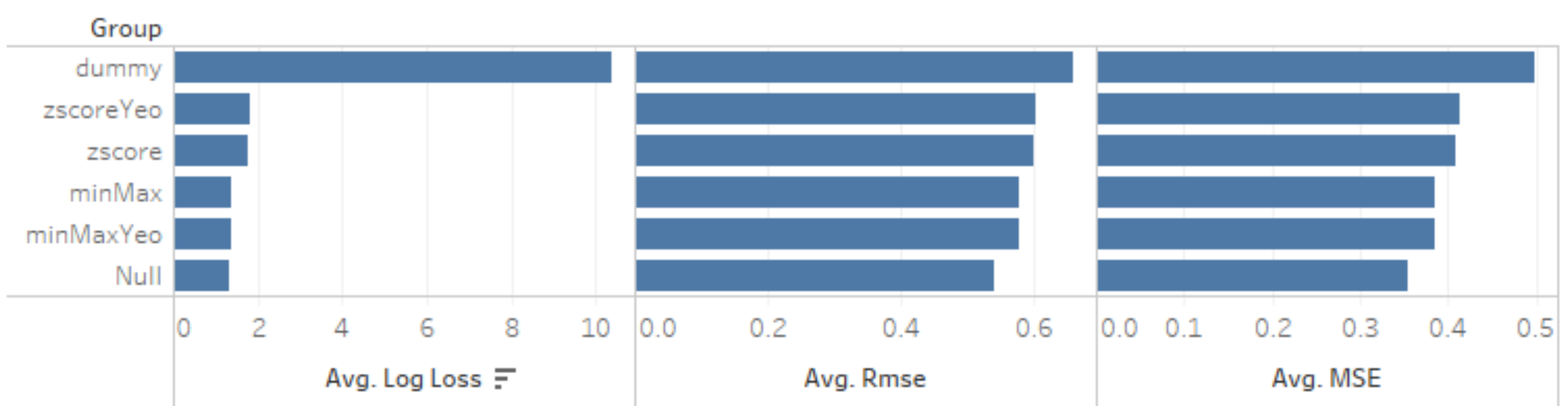

Figure 7. NB Results

Conclusions

The results show that in most situations, creating a dummy-1 encoding of categorical variables can improve results for random forest and gbm modeling techniques. Neural Networking and naïve bayes algorithms did not seem to have a consistent favorable feature engineering approach.

Our research suggests when modeling using random forest or gradient boosting machines in a classification setting, creating dummy variables from categorical features can improve accuracy metrics and model performance. Our research also suggests naïve bayes as well as neural networks do not consistently benefit from a specific feature engineering approach.

In future research, we recommend studying the effects on regression based data as well as testing additional modeling techniques.

Acknowledgements

We thank Professor Matthew Lanham for guidance on this project.