**Overview**

The Wells Fargo Campus Analytics Challenge for 2020 was particularly computational in comparison to previous mindsumo competitions put out by the company. The dataset had no obvious correlations to real-world phenomena, focusing the participants on data science methodology alone. As a result, the solution presented has been designed with the intention of acquiring the maximum accuracy, or f1-score, as requested by Wells Fargo. Other components of the challenge were deeply considered but given less importance.

For maximum accuracy, no other method worked better than a logistic regression algorithm coupled with label-based encoding. Support vector machines performed similarly well but showed no major improvements over the basic algorithm.

Unfortunately, no tested combination of encoding, dimensionality reduction, clustering, or machine learning maintained a similar level of predictive capabilities with fewer predictors. While a particular nonlinear reduction technique coupled with a more targeted clustering method may improve the results given a highly nonlinear learning algorithm, such as neural networks, the evaluative metrics conducted thus far do not suggest this to be the case. This will be explored further in the dimensionality reduction and clustering passages.

While logistic regression is a simplistic algorithm, the results collected suggest that it is also the most capable predictive algorithm for this given dataset. PCA is next best utilized to reduce the number of features.

**Review of Data**

The data has 30 regressors, and one categorical predictor. Two methods were employed to digitize the categorical variable: label encoding and one-hot-encoding. They appeared to perform similarly, so label encoding was selected to maintain minimal features.

A box plot revealed that the data was standardized around 0, though a standardizing function was employed to guarantee this for the algorithms. Furthermore, no major correlations were found between any predictors, with the only variance likely being a product of noise. There was no indication as to what this data represented, so no external knowledge could be applied to feature engineering.

**Dimensionality Reduction**

Dimensionality reduction methods often utilize an elbow or percentile to maintain general accuracy while greatly reducing features. Furthermore, they can assist in clustering, as most distance minimizing algorithms perform poorly in higher dimensions. What follows were my attempts at utilizing various methods.

**Missing Value Ratio**

There were no missing values, so no predictors could be eliminated based on this metric.

**Variance Threshold**

As shown by the boxplot within the code, the variances appear to be similar across features. None can be removed based on low variance.

**Univariate Selection and Factor Analysis**

No differentiable correlations appear in the data. As orthogonal features were a criterion for judging, one may utilize the PCA code provided to get entirely uncorrelated features. However, this is mostly unnecessary due to the nature of the data.

**PCA**

Principal Component Analysis was the best dimensional reduction technique, though the explained variance was almost linear. This is likely due to the composition of the data. As a result, there is not a clear threshold as to when one should cutoff the number of features. It is odd that the explained variance graph is not logarithmic in shape.

A piece of code calculates the regression accuracy based on various number of principal components. There is a sharp drop off in accuracy, that generally levels out with progressively fewer features. Perhaps one may get decent accuracy with very few PCs, but it would not come close to that achieved with the 31 regressors.

Still, if one has all 31 features to begin with, and for whatever reason wishes to decrease the number of variables, PCA gave the best results.

**MDS and TSNE**

These nonlinear dimensional scaling methods performed terribly and decreased the accuracy dramatically. Other methods may have more success, or these could be more finely tuned outside of sklearn. However, they are extremely computationally expensive. I spent an hour waiting for an MDS model to loop through all predictors, removing one each time. The results, when trained through the machine learning models, were chaotic and poor.

**Random Forest**

Random Forest gives a small indication for which features are more important than others. Still, the variance is entirely explainable by noise. As a result, it does little to indicate which columns should be removed, as each has a similar weight when utilized by the algorithm.

**Clustering**

Few clustering methods were selected given the boxplot, dimensionality reduction techniques, and correlation matrix. However, a KMeans algorithm was employed for certainty. An elbow method showed that more than one category led to a sharp drop in accuracy, with constant poor performance for each additional category.

Given that KMeans gave 0 additional predictive capability to any of the machine learning models, no further clustering techniques were employed. This algorithm was attempted both with the original data, and dimensionally reduced data using PCA. Even extremely low numbers of features showed no clustering.

**Machine Learning**

A wide selection of machine learning methods was employed to determine the best predictive algorithm. Logistic Regression performed significantly better than the algorithms, excluding Support Vector Machines, which performed similarly. Naïve Bayes, Random Forest, and k-NN worked decently, and can be utilized as desired.

Neural Networks and other complex machines were not utilized due to the failure of more sophisticated algorithms with the dataset. For instance, the greater performance of Logistic Regression over Random Forest suggests that less complexity is desired for prediction.

**Conclusion**

As presented in the flowchart and code, there are plenty of options one may select to process the data. However, the best accuracy is certainly obtained by utilizing the logistic regression method on the original data with a label-encoded XC column.

There were four challenge objectives presented:

1. Use the given dataset.
2. Use the most suitable encoding schemes.
3. Use the least set of feature variables.
4. Use no correlated variables in the set of predictors.

The first objective is impossible to leave incomplete without guessing.

The second is completed by utilizing label encoding, though one-hot-encoding works well and is included as an option.



The third objective is discussed at length in the dimensionality reduction section. Clustering showed no ability to support fewer features in matching a similar accuracy. Furthermore, there is no clear optimal point where number of features and accuracy are maximized.

There are nearly no correlated variables in the original data. One may utilize PCA if they require completely orthogonal features.

Thank you very much for taking the time to read through this document, and for providing us with this challenge. I look forward to hearing from the judges at Wells Fargo and Mindsumo.