Ronald Pacheco

State Farm Exercise

Executive Summary

In this exercise, we are looking to examine the generated data provided by State Farm, as well as create GLM and Non-GLM classification models to be applied on the test data. The data is segmented into a Train data, containing 40,000 rows and 101 columns, including the target variable, and a Test set, containing 10,000 rows and 100 columns. The models will be evaluated based on AUC score, and the best model for each family will be chosen based on these scores.

Data Cleaning

For the most part, the dataset was complete. Only four features were dropped as one of them contained only one category, and the other three were missing over 80% of the values.

The dataset was composed of categorical and numerical data. Some of the numerical data was presented as a string (interpreted by the computer as letters rather than numbers).

One of the categorical features presented inconsistencies. More specifically, it contained two types of formats for weekdays (i.e. Monday and Mon).

Exploratory Data Analysis

The dataset was highly imbalanced, with only 14.5% of the target variable as positive label (figure 1). This was very important when we were creating and evaluating our models.

There we no notable outliers, and the variable distributions were within the requirements of the models we used (for visualization refer to the notebook).

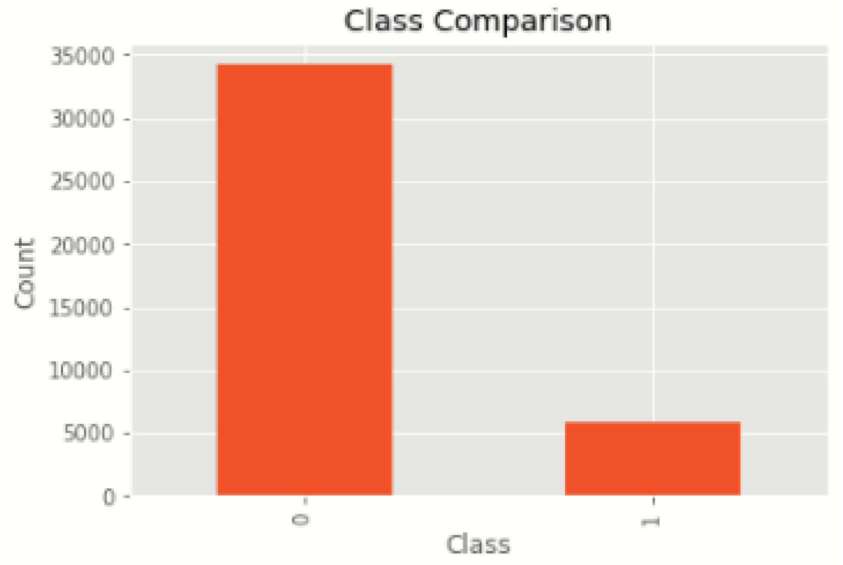

Figure 1 – Class Imbalance

Modeling

Logistic Regression

Logistic Regression is one of the supervised Machine Learning algorithms used for classification i.e. to predict discrete valued outcome. It is a statistical approach that is used to predict the outcome of a dependent variable based on observations given in the training set.

- Advantage: Logistic Regression is one of the simplest machine learning algorithms and is easy to implement yet provides great training efficiency in some cases. Also due to these reasons, training a model with this algorithm doesn't require high computation power. Also, this algorithm allows models to be updated easily to reflect new data, unlike decision trees or support vector machines.
- Disadvantage: The training features are known as independent variables. Logistic Regression requires moderate or no multicollinearity between independent variables. This means if two independent variables have a high correlation, only one of them should be used. Repetition of information could lead to wrong training of parameters (weights) during minimizing the cost function. To overcome this issue, we selected the features that best described the outcome using SelectKBest.

Logistic Regression Pre-processing

We tried three models:

- **Model 1** - baseline: variables with missing values were dropped.
- **Model 2** - balanced data: the positive label was resampled using bootstrap.
- **Model 3** - imputing the data: The missing data was imputed using KNN.

Model Iteration	AUC Score	Accuracy
1	**0.6809**	**85.95%**
2	**0.6759**	**85.93%**
3	**0.7575**	**86.24%**

Table 1 – AUC and Accuracy scores for Logistic Regression models

Random Forest Classifier

Random forest is a supervised learning algorithm. It builds a forest with an ensemble of decision trees. It is an easy-to-use machine learning algorithm that produces a great result most of the time even without hyperparameter tuning.

- Advantage: Random Forests implicitly perform feature selection and generate uncorrelated decision trees. It does this by choosing a random set of features to build each decision tree. This also makes it a great model when you have to work with a high number of features in the data. As well as being robust against outliers.
- Disadvantage: Random Forests are not easily interpretable. They provide feature importance but it does not provide complete visibility into the coefficients as linear regression. Random forest is like a black box algorithm, you have very little control over what the model does.

Random Forest Classifier Pre-processing

- **Model 1** - baseline: variables with missing values were dropped.
- **Model 2** – fine-tuned baseline model: the baseline model was fine-tuned using GridSearchCV.
- **Model 3** - imputing the data: a pipeline was created using the KNN imputer.

Model Iteration	AUC Score	Accuracy
1	**0.7285**	**86.03%**
2	**0.7606**	**86.32%**
3	**0.7630**	**86.33%**

Table 2 - AUC and Accuracy scores for Random Forest models

Business Recommendation

Random Forest is better because the type of data we are working with is harder for Logistic Regression to work with. Random Forest, as its name implies, is, basically, a bunch of decision trees that are created by randomly selecting features. Then the results of each tree are combined to make a final decision. This is what allows the random forest model to perform better with our data, which has high dimensionality as well as being highly imbalanced. Further work will improve the model's performance even more.

Conclusions

- RandomForestClassifier is believed to perform better on the test data, since the nature of the data makes it be high-dimension and imbalanced, and random forest are robust against these types of datasets.
- We were able to improve the models' AUC score from as low as 0.67, to 0.76.
- We were also able to select the best threshold for each model based on accuracy. It is important to note that, with highly imbalanced datasets, it is better to use precision and recall to determine the threshold at which the model performs at its best based on the business case goad.
- For these models, we are using data we do not fully understand. Therefore, we used accuracy as our main metric to determine model performance. In a real-world scenario, I would want to understand the business goal of the model to select more adequate metrics. Precision/Recall are usually better, depending on the use case. For example: If this model was to be used to determined best candidates for email ads, then I'd say we'd want a higher Recall, given that this ad campaign would not cost too much money, and if we have false positives, it wouldn't really matter. On the other hand, if the ad campaign was more expensive, then we'd want more True Positives, therefore, Precision would be a better metric. After knowing the use case, a threshold for the best result can be selected.
- Our EDA was very basic, we only checked for outliers and distributions. This was mostly due to not knowing what the variables represented. Some variables contained state names, car brans, sex, day of the week and month. A comparation could have been made to understand which cars tend to be insured in what state and whether it was a woman or a man. Knowing the other variables would have allowed for much deeper analysis of the data.
- Other than creating dummy variables for categories, Feature Engineering was not performed for the same reasons stated above.
- It is important to keep in mind that these models are far from perfect. Good models are the result of constant improvement, and this is just the baseline for future research.

Future Research

- Imputing our missing data gave us a significant increase in our AUC scores, trying a SMOTE method might better the scores even further.
- We tried different approaches, and we also cut corners. We upsampled our data after dropping the missing values. Trying to upsample, or SMOTE, after imputing might give us better results.
- Applying GridSearchCV with different number of neighbors for the imputer, as well as different K values for SelectKBest, might improve the LogisticRegression model as well as the RandomForestClassifier. And then, running another GridSearchCV to make sure the imputed data didn't significantly change the model and its optimal parameters.