DME Project: The Caravan Insurance Data

Abhijit Singh
s1788323

Abstract

Performing binary classifications is a standard task in machine learning. However the Coil 2000 Data Mining Challenge is a tricky task because of the heavy class imbalance. In this report, we explore the dataset and try different sampling techniques to make meaningful predictions. We compare various classifiers and feature selection techniques, and evaluate our results. We show that using just 7 features is enough to make good predictions, and this means that data collection for future analysis will be an easier and cheaper task to perform. Our best model performed as well on the test set as the mean performance in the competition.

1 Introduction

1.1 Description of the task

This dataset was used in the Coil 2000 data mining competition. It contains customer data of a Dutch insurance company. Each customer has 86 attributes associated with them, and the goal is to classify whether an individual has an insurance policy covering caravans. Out of these 86 attributes, 43 are socio-demographic variables associated via the customer's ZIP area code, and the other 43 contain information about ownership of other insurance policies of that individual. It is assumed that all individuals in a particular ZIP area code have the same socio-demographic attributes.

The training set contains information of 5,822 customers, of which 348 are customers holding an insurance policy for caravans. This means that only about 6% of the training examples belong to the positive class (holding caravan insurance), and the rest are in the negative class (not holding caravan insurance). The test set consists of 4,000 other individuals, who are drawn from the same population. The main task is to predict whether a given individual is likely to buy a caravan insurance policy or not.

1.2 Relevant background and related previous work

Many participants of the Coil 2000 data mining competition published materials explaining their work. The common theme across the most successful methods was a detailed analysis of feature selection, and use of statistical significance to judge which features were most important for this classification task. Huang et al. mention in their study that several important features for the classification could be guessed intuitively, such as a rich person who owns both a caravan and a car would be extremely likely to buy a caravan insurance. Thus they chose features which would help identify such traits, and discarded the rest. They also created some new binary features using the existing ones, such as owning a fire insurance policy and third party insurance policies. They used under-sampling (in a 50-50 ratio) to solve the class imbalance in the training set, and used classification trees as their learning model. They also implemented a logistic regression model, and for this, they eliminated useless or insignificant variables based on domain knowledge and p-values for each variable.

Amr in his work also used statistical tools to reduce the number of features used for the

final classification task. He used Information Gain and Chi-squared method to select the top 3,7 and 14 features. He tested across the different sizes of features, and also among the different sampling techniques such as Random Under Sampling (RUS), Random Over Sampling (ROS) and Synthetic Minority Over Sampling Technique (SMOTE). He used a Multi Layer Perceptron (MLP) and decision trees as his two learning models. For both the models, SMOTE did not work well.

The winner of the competition, Elkan, also used similar methods for feature selection. He argued that the data was limited, and checking for statistical correlations threw up obvious features to look for while classifying customers. Hence it was needless to use complicated models to achieve a good performance on this challenge. He used a Naive Bayes classifier, with new attributes derived from existing attributes, so as to improve the performance of the classifier, as it relaxes the conditional independence assumptions that are used in the Naive Bayes model.

This background gave us a good starting point for our work, as we could judge that the crucial aspect for this task would not be to use the most complicated model, but to do a detailed and thorough statistical analysis of the features, and discard the ones which have little contribution. Another key insight was to look at different sampling methods depending on which classifier was being used.

1.3 Explanation of the significance of the objective

The objective of this task is to identify which individuals are most likely to buy caravan insurance. This challenge was a part of an open competition, so that the Dutch insurance company which provided the customer data could benefit from the insights provided by the competitors. A few examples of the benefits enjoyed by the company, as a result of insights gained after the competition, would be the possibility to implement **area-focused marketing** (focus on strategies that are most suitable and most likely to induce customers in each specific area to buy the insurance policy) and creating **targeted advertisements** for people who are most likely to buy the policy. Another benefit could be **holding joint marketing events with caravan makers** to target potential customers and arouse their awareness of risks, which would make them consider buying an insurance policy.

Apart from these, the feature selection methods used would also give them an indication of which pieces of information are being used for the predictions. This would in turn mean that they can collect less data to follow the same prediction process in the future, saving the high costs involved in data collection.

2 Data preparation

Before exploring the dataset, it is necessary to have an initial overview of the given data. This is crucial because data preparation methods should be applied, if required, before moving to the analytical tasks.

At the beginning we are given a dataset of 5,822 entries, where each entry has 86 features. All features contain different pieces of information regarding each customer, and the last one is used to indicate whether the corresponding customer had bought a caravan insurance or not (1 or 0 respectively). We also verified that our data does not have any outliers using a nearest neighbours search, and checked for any missing values as well.

An important task was to split our dataset into training, validation and test sets. The given data already had two different files corresponding to the training and test sets. Therefore, our job was to split the initial training set file with a ratio of 4:1 . Thus, 80 % of the examples were used for training, while the rest 20 % were used as the validation set (1084 examples of negative class, 81 examples of positive class) that would help in fine-tuning the hyper-parameters of each of the classification algorithms used.

Since our dataset is unbalanced, with approximately 6% (or 267) of the training examples (after splitting training and validation sets) belonging to the positive class, we need to use some techniques to overcome this hurdle. The most common ones are RUS, ROS and SMOTE. RUS is basically choosing only a few examples from the dominant class, so that the final ratio is as desired. In our case, we use a 1:1 ratio, and since the positive class has only 267 examples, we randomly choose

267 examples from the negative class. In ROS, we duplicate the minority class examples until we reach the desired ratio. We had 4,390 examples from the negative class, and 4,390 examples from the positive class. SMOTE (Bowyer et al., 2011) is a variation of ROS, where instead of simply duplicating the minority class, we synthetically create new examples from the existing ones. The Adaptive Synthetic sampling approach, or ADASYNHe et al. (2008) algorithm, is similar to SMOTE. The main difference is shifting the importance of the classification boundary to those minority classes which are difficult to learn.

Consequently, through the rest of our analysis our models are trained and tuned with the training set and validation set respectively. The remaining test set is used only for reporting the performance of our best model, so that a comparison with other solutions (including the winning one) would be meaningful. Note that oversampling and under-sampling is carried out solely on the training set. This is crucial since with the sampling techniques we just want to prevent a possible bias in our classifiers. The performance on either the validation or the test set should be evaluated on the unsampled data.

3 Exploratory Data Analysis

In this section we do an exploratory analysis of data with the prospect of increasing our understanding of the data and its properties. Before splitting our dataset, as mentioned in Section 2, we explored the values contained in each of the 85 features (86^{th} feature represents the label). In Figure 1 we present how many zero values each feature contained amongst all the entries.

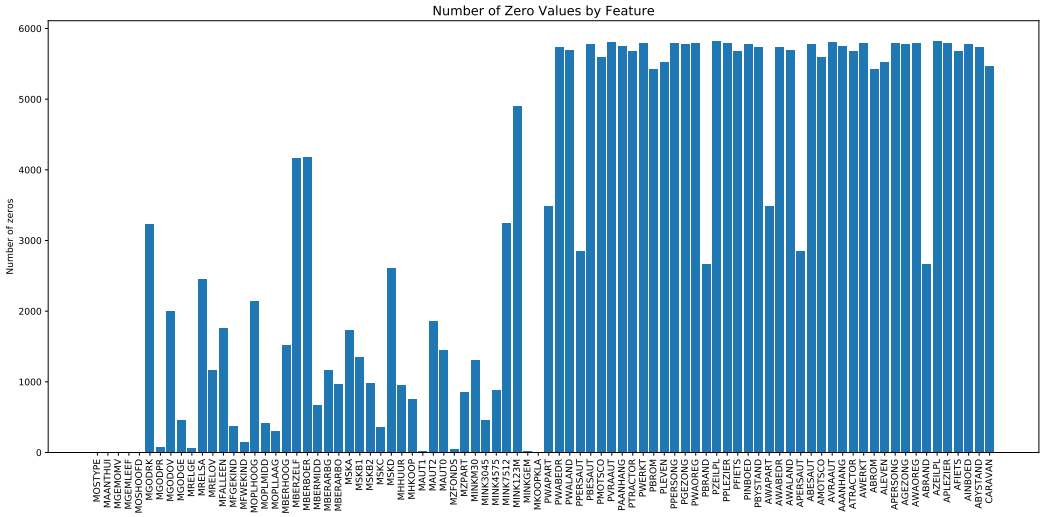
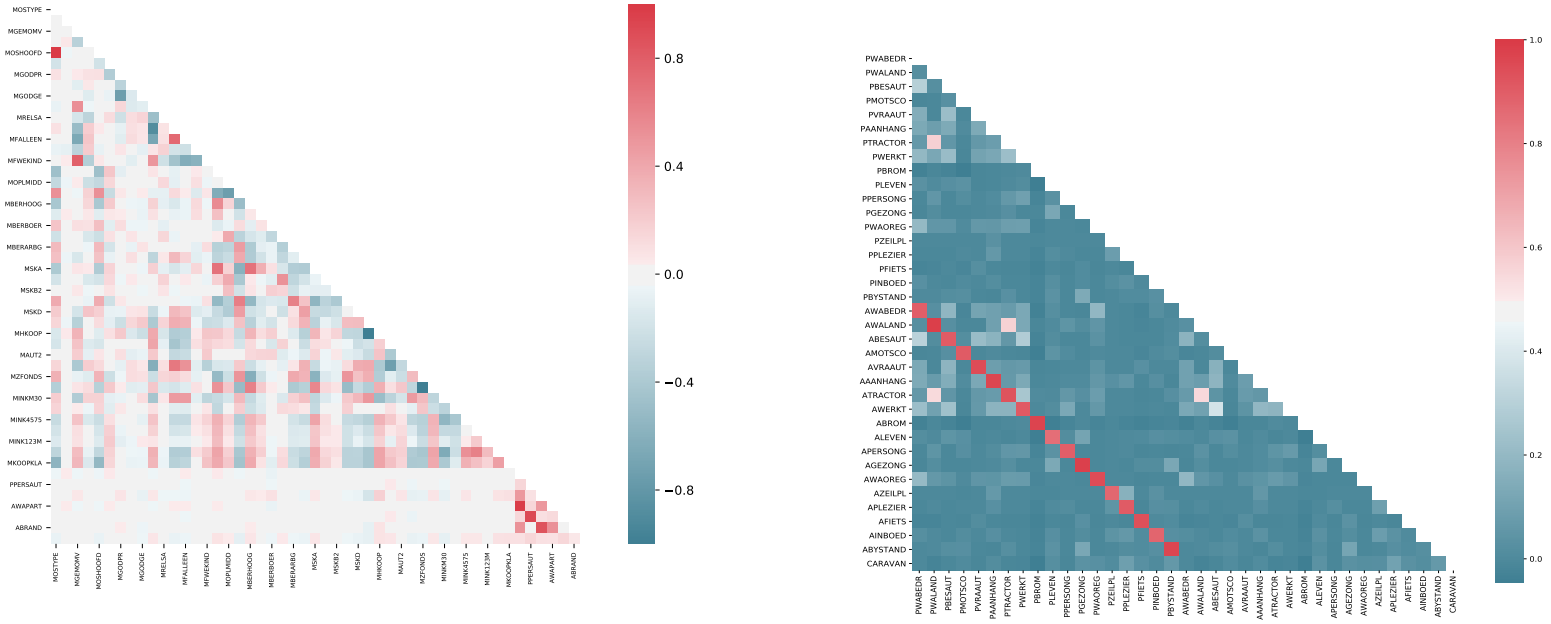


Figure 1: Total number of zero values for each one of the features.

As we can see, many features (36 out of the 86) have a zero value for almost every single customer. This is an interesting finding. This either enables us to make good classifications by exploiting these non zero entries, or it allows us to remove those features, thus reducing the amount of information that needs to be collected about a customer in the future. In order to prove this, we checked whether these 36 features were correlated with the output class. The last row in Figure 2(b) shows how correlated the features are to the output class. Dense red colour shows high correlation, and it is clear this is not the case. We can see that none of these 36 features is correlated with the output class, and thus we can remove this unnecessary extra information.

Having reduced the number of features to 50 (including the output class), we decided to identify which features contribute the most in the classification problem, and which may be less relevant, and hence should be merged with others or even discarded. The first approach was to discover which attributes were closely correlated with each other. Therefore, a correlation matrix was computed, as shown in Figure 2 (a), where we obtain a broad view of this information.

This matrix is symmetric (as expected) and that is why we present only the lower part of it. In general, we can see that most of the features are not highly correlated with others, except only for a



(a) Correlation matrix for the 50-feature data.

(b) Correlation matrix for the 36 removed feature data.

Figure 2: Correlation Matrices for different parts of Data

few specific cases. After identifying which features had a significant correlation between them (a coefficient greater than 70 % or lower than -70 %) we tried to interpret the information that they specifically represented.

An important finding was that the features representing the number of a specific type of policy and the one representing the contribution of this policy in terms of actual value were quite correlated. Fire policies, car policies and private third party policies had the maximum positive correlation value with their respective contributions. However, from this level of analysis it is not yet clear to decide if it is best to combine those features or just keep one of the pairs. Elkan the winner of the competition, used the combination of car policies and the contribution of each car policy rather than keeping only one of them.

Other significant correlations were spotted between features that correspond to demographic information. For the case of features MOSTYPE and MOSHOOFD that represent the customer's Subtype and Main Type respectively, the correlation was nearly 100 %, thus indicating that once we know the value of the first feature, we can easily decide the value of the other one.

There were also features that were negatively intercorrelated to a significant extent. Example cases are the ones representing whether the customer is married or has other relations, or the ones indicating whether the customer owns a private health insurance or a national one. The feature-pairs extracted this way informed us of relationships that were quite obvious and could be explained by human reasoning.

4 Learning Methods

For the classification task, we experimented with 4 different algorithms, namely Logistic Regression, Support Vector Machines (SVM), Neural Networks (NN) and Naive-Bayes (NB). Each one of them is well-known and widely used in practice, with different advantages and disadvantages.

4.1 Logistic Regression

Logistic Regression is a regression model for binary classification where the dependent variable is categorical, instead of numerical. The goal of this algorithm is to find a boundary in the feature space that could separate the training instances (represented as data points) into 2 distinct groups. A major drawback of this approach comes when there is no way to transform the data into separable and distinct groups as efficiently as required.

4.2 SVM

SVM shares a lot with Logistic Regression, however the optimization problem is somewhat different. Here the goal is considered a geometrical one, where we try to find a particular, optimal separating hyperplane in the context of the support vectors. However, in the case of inseparable data the problems would be much worse compared to Logistic Regression.

4.3 Neural Networks

NN are quite popular and offer promising results when dealing with non-linear classification boundary and they are widely used for multi-class problems. A very common issue with NN is the increased complexity and the amount of weights to be learned and hyper-parameters to be tuned. For our task NN could be a needlessly complex architecture, based on the fact that Logistic Regression is the simplest form of a NN and that our case does not involve a multi-class problem. Also, NN does not tend to work well with limited data.

4.4 Naive Bayes

Finally, the Naive Bayes classifier belongs to a family of simple probabilistic classifiers that make use of the Bayes theorem. Their simplicity stems from the strong (or "naive") independence assumption between the given features. However, they offer quite satisfactory results and with appropriate pre-processing, they manage to be competitive than more sophisticated methods like SVM.

5 Results & Evaluation

After proving the validity of our choice to keep 49 features (plus the output class) in total, we moved on to comparing the aforementioned techniques. Given that we have an unbalanced dataset, we are not interested in validation accuracy and our focus is more on the precision and recall metrics.

Our first concern was related to the imbalance of the data. Therefore our initial experiments were targeted at identifying the most suitable sampling approach. As we have mentioned, there were many alternatives, namely oversampling, under-sampling and different variations of them. So in order to have a valid comparison, we firstly ran experiments with a logistic regression model and all possible choices of oversampling listed in Table 1.

In the table, 'Original' refers to the data which has all 85 features (except the output class) with ROS applied to it, while all the rest have only 49 features. 'NOS' refers to the training set without any sampling techniques applied to it. 'Val Acc' refers to validation accuracy.

Table 1: Comparison of different oversampling methods on the classification task with Logistic Regression

Method	Val Acc	Recall	Precision
ROS	0.713	**0.691**	0.153
ADASYN	0.702	0.630	0.139
SMOTE	0.715	0.630	0.144
Original	0.740	0.654	0.162
NOS	0.940	0.000	0.000

We are more interested in recall and precision scores because this dataset is unbalanced, so the validation accuracy will not give us the right picture. We can see this in Table 1, NOS has an accuracy of 94%, but this is due to it classifying all examples as being part of the negative class. This is why

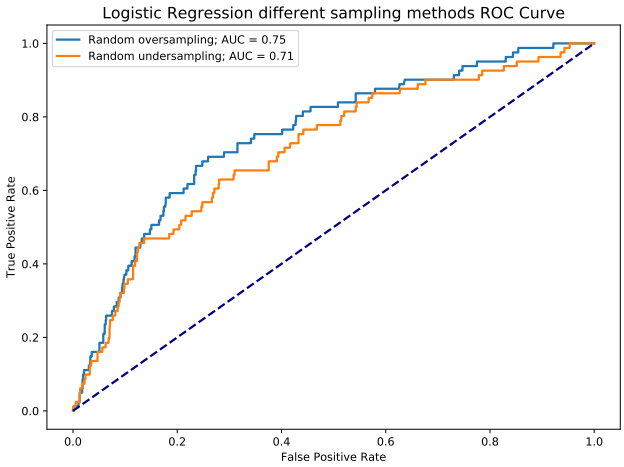


Figure 3: Comparison of ROS and RUS with respect to their ROC curves.

the recall and precision scores are 0. Further, we want to maximize the number of people who are likely to buy the policy, and thus recall and precision give us a better idea of this. Based on results shown in Table 1, we found that ROS was the best oversampling technique, as it had the highest recall and good precision.

After determining which oversampling procedure was the most effective, we compared oversampling with under-sampling, and did this for all four different classifiers. Table 2 summarizes the results of these experiments. We can see that ROS outperforms RUS in all values of precision and validation accuracy, and is better at recall for the Logistic Regression and Naive Bayes models. The best performing model was Logistic Regression with ROS, as it had a recall of 69.1% and precision of 15.3%, which was the best of all the different models. Another illustration of the ROS superiority is given with the Receiver Operating Characteristic (ROC) curve in Figure 3, where we observe a greater Area Under Curve (AUC) for the ROS, compared to RUS. For the next set of experiments, where we do feature selection, we use only this model.

Table 2: Comparison of different algorithms on the classification task.

Classifier	Val Acc ROS	Recall ROS	Precision ROS	Val Acc RUS	Recall RUS	Precision RUS
Logistic	0.713	**0.691**	**0.153**	0.672	0.654	0.130
SVM	0.856	0.198	0.134	0.659	0.605	0.118
Naive Bayes	0.673	0.679	0.134	0.663	0.617	0.121
NN	0.791	0.407	0.145	0.584	0.679	0.107

5.1 Feature Selection

Feature or attribute selection is a process where we select those features that contribute most to the final prediction. If our data contains many features that do not really contribute to the final prediction, then the performance of our model may not be optimal. Another reason is that feature selection can help avoid over-fitting.

The first thing we tried was using some statistical measure to judge which features were most important for the prediction task. We used the chi-squared (x^2) method (Kaushik, 2016), which evaluates the correlation or association between a group of categorical features using their frequency distribution. The selection of the features is independent of the machine learning algorithm that is used for classification. We used chi-squared method to select the 25, 14, 7 and 3 most important features.

Next we use wrapper methods, which use a subset of features to train the classification model to find out which features have most influence on the prediction. Recursive feature elimination (RFE) (Bakharia, 2016) recursively removes attributes and builds a new model on the attributes that remain.

The identification of the attributes which contribute the most is done based on accuracy. We used RFE to select the 25, 14, 7 and 3 most important features.

We then use Principal Component Analysis (PCA). PCA can be used for dimensionality reduction of the input data. The data is represented by its first k principle component scores. PCA is a linear transformation and results in a lower-dimensional projection of the data that preserves the maximal data variance. In our experiments, we project the input data in 39, 25, 14, 7 and 3 most varying dimensions.

Bagged decision tree algorithms like Random Forests can be used as estimators of the features' importance. In this feature selection method, we fit a decision tree model and then we select the 25, 14, 7 and 3 most important features.

So far we have examined feature selection from an analytical point of view. In addition to this, we would like to compare a more intuitive approach with respect to a PCA alternative. Therefore, we decided to discard one of the highly correlated features for every pair we found, thus obtaining a different way of doing feature selection. The attributes to be discarded were 10 in total and were selected based on our own intuition. Having done that, we decided to compare our logistic regression model that utilized PCA for a total of 39 features versus our own solution of choosing 39 features. Table 3 summarizes the results of all the experiments described above, and we can observe the superiority of the analytical solution (PCA 39) against our intuition (Human 39), both on precision and recall, albeit by a small amount.

Table 3: Different Feature Selection Methods.

#Most important features (25,14,7,3)	Val acc	Recall	Precision
Chi-squared 25	0.700	0.716	0.151
Chi-squared 14	0.687	0.704	0.143
Chi-squared 7	0.667	0.691	0.134
Chi-squared 3	0.663	0.679	0.130
RFE 25	0.688	0.691	0.149
RFE 14	0.687	0.716	0.149
RFE 7	0.623	0.531	0.104
RFE 3	0.605	0.494	0.095
PCA 39	0.709	0.704	0.153
PCA 25	0.700	0.704	0.149
PCA 14	0.692	0.704	0.145
PCA 7	0.652	0.691	0.128
PCA 3	0.632	0.543	0.101
Random Forest 25	0.698	0.704	0.148
Random Forest 14	0.677	0.704	0.139
Random Forest 7	0.662	**0.741**	0.139
Random Forest 3	0.663	0.679	0.130
Human 39	0.699	0.691	0.147

Amongst the different trials of chi-squared method, we get the best results when the top 25 features are being used, both in terms of recall as well as precision. However, the performance while using just the top 7 features is not much worse, and hence the company can decide if they want a slightly better performance by using many features, or save costs by collecting much lesser data and getting almost similar results. For RFE, we see that using top 14 features give best results, and there is a sharp drop off when the number of features is reduced further. For PCA, the best recall values are seen for 3 different feature combinations: the best 39, 25 and 14 features. The precision reduces slightly as we decrease the number of features. Again, using just 7 features gives almost the same performance as using many more features. Random forests had the best performance amongst all the

different techniques, when 7 features were used. It had a recall of 74.1%, which is 2.5% more than the next best model, although the precision was slightly lower compared to others.

Depending on how many features we want to use, we can accordingly choose the best model. For 3 features, Random Forests and chi-squared method both had the exact same (and best) performance, and a deeper investigation revealed that they both chose the same three features. For using top 7 features, Random Forests was the best model. It was also the best of any model we had tested, and thus if we use only one of these four classifiers, our experiments show that we only need the 7 features used by the Random Forest model. A final demonstration of our best solutions on feature selection are given in Figure 4. Here we can observe the AUC of the ROC curves and although the no feature selection approach has the maximum value, the differences with the others are not that significant. This means that the company can save costs on data collection in the future, without compromising on the best performance.

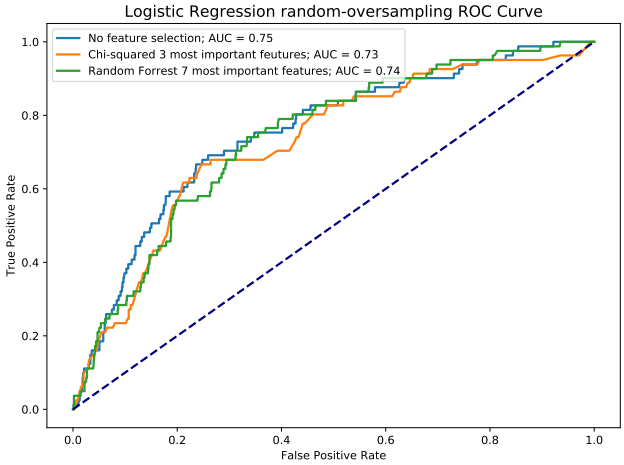


Figure 4: Feature Selection effect comparison with respect to the different ROC curves.

The final 7 features being used pointed towards the fact that the profile of a potential customer is a rich person who owns their own house, and has good contributions to their car policies, and own fire insurance policies as well.

As the last step in the study, we wanted to compare our performance with that of the competitors in the actual challenge. We ran our best model on the test set, and correctly identified 145 individuals who actually bought the policy (true positives), and incorrectly identified another 1387 (false positives). However, our 800 most likely candidates to buy the policy only had 93 true positives, which was the mean performance in the competition.

6 Conclusion

Our task was to use the given dataset and identify which individuals in the test set would be likely to buy the caravan insurance policy. A brief review of the literature guided us on which sampling techniques we should try. We played around with the data to learn more about it, and using a correlation matrix of the features along with an analysis of the values of each feature allowed us to eliminate 36 of the 85 features. We compared different sampling techniques, followed by a comparison of different classifiers. We chose the best model up to that point, and compared different feature selection methods on it. We found that using top 7 features using a Random Forest to train a logistic regression classifier gave us the best performance, in terms of recall. Since we are concerned with maximizing the number of true positives in our predictions, recall was the most important metric for us throughout our analysis. Our best model had a performance comparable to other competitors.

This was a small study, and it is likely that this is not the optimal solution for this task. In the future, we can look towards trying a few more classifiers, and more sophisticated feature selection methods. We can also experiment with creating new attributes derived from combinations of existing ones, and perhaps converting some attributes into binary attributes.

References

Huang, C.; Mabuchi, M.; Songchitruksa, M.; Visitrattakul, N. Understanding Characteristics of Caravan Insurance Policy Buyer. 2007; http://www.galitshmueli.com/sites/galitshmueli.com/file/Caravan%20Insurance%20REPORT.pdf.

Amr, T. Mining Caravan Insurance Database Technical Report. 2010.

Elkan, C. Magical thinking in data mining: lessons from CoIL challenge 2000. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. 2001; pp 426–431.

Bowyer, K. W.; Chawla, N. V.; Hall, L. O.; Kegelmeyer, W. P. *CoRR* **2011**, *abs/1106.1813*.

He, H.; Bai, Y.; Garcia, E. A.; Li, S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence). 2008; pp 1322–1328.

Kaushik, S. Introduction to Feature Selection methods with an example (or how to select the right variables?). 2016; https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables

Bakharia, A. Recursive Feature Elimination with Scikit Learn. 2016; https://medium.com/@aneesha/recursive-feature-elimination-with-scikit-learn-3a2cbdf23fb7.