Case Study 1

# Case Study Summary

This exploration presented in chapter 7 of the Data Mining Application with R (DMAR) is all about customer profiling. The current customers are the one being profiled by a machine learning algorithm in hopes of producing an accurate model for deployment of said algorithm into production.

The company Caravan Insurance wants to use a predictive model to help identify customers who do not yet have policies but could potentially need one. Once this is done then they will use a more targeted marketing campaign to better reach the clients that they want.

# Technical Discussion

The data that was used in the exploratory analysis was about an insurance company. It contained 9822 rows and 86 columns. The data contained information about current customers like income, age range, other policies held, and it was in binary format. The researchers performed correlations on both the training data and the validation data so if there was anything that did not necessarily have a big impact on anything. They found that all the religion fields were of no significant correlation. Since these fields has missing values, they were just removed. All fields have a binary format, meaning that their values are either 0 or 1, of the data which is perfect for multiple models that work well at predicting a yes or no answer. Logistic Regression, Bagging, Support Vector Machines, and a Rpart Models are the ones that were used, and ROC Curves were used to show performance for these models on the data.

To get an idea of how the transformed data check out, they used a logistic regression with all the variables to briefly look at which ones were significant. Firstly, this model was very quick to complete. Now, diving into the data we see the use of a Rpart model, like a decision tree, the algorithm places the data into a yes or no category based on certain criteria. It keeps doing this until there is no data left. The important variables here are:

1. PPERSAUT
2. APERSAUT
3. MOSTYPE
4. MSHOOFD
5. MKOOPKLA
6. MBERHOOG
7. PBRAND
8. MZFONDS
9. MZPART
10. ARACTOR
11. PTRACTOR
12. PWAPART

In plotting a histogram of the overall importance of the variable results in an extremely right skewed distribution.

The next model used was the bagging, or bootstrap aggregating, model. This model was similarly quick to complete but did take slightly longer than the Linear Regression model. The model used 25 trees during modeling. The median of the data predictions was 0.024 with a max value of 0.38, this tells me that there could be 1 very important variable. The histogram plotted was like the Rpart model in that it is has an extremely right skewed distribution.

The third model used was an SVM took the longest time to compute. This model assigns weights to the variables, which is one reason why it takes so long to calculate. In the weights there are both negative and positive values, this simply describes where the point is in reference to the discriminant line or the separator. The bigger the number, positive or negative, the farther away from the discriminant line and therefore easier to classify. Having these positive and negative numbers means that the date is separable. The model had a similarly right skewed distribution, but not as extreme as the first two.

The last model use was linear regression, which was the same model used to preview the data correlations and significant variables. Here we see that the best variables are:

1. PPERSAUT
2. PBRAND
3. ALEVEN
4. APLEZIER
5. PLEVEN

This model is straightforward in terms of complexity. The histogram still shows and extreme right skewed distribution.

The performance of the predictions for each of the models used were plotted on a ROC graph. All of the models performed quite poorly according to scoring criteria and based on the line from point (0,0) to point (1,1). The model that performed the best would be the logistic regression model, albeit still poor performance.

# Suggestions

First, I would like to suggest the use of the tidyverse. I have utilized this before, and it is quite difficult to go back. There are so many tools that make transforming data and getting it to the point where it can be modeled so painless. Second, I would probably suggest going back and reducing the number of variables by feature engineering. There are several reasons for this suggestion:

1. There are over 80 variables, and this is a way to reduce the number of variables because maybe 5 of them were significant
2. The models would run faster with less variables

Thirdly, it might be beneficial to go back and pick different models if the chosen ones are still not performing very well, even after some feature engineering takes place.