DAT 640- Predictive Analytics

Final Project

Southern New Hampshire University

Timothy Harrison-Reyes

March 6, 2022

I. Organizational Background

a. General

The Insurance Company (TIC) is a business that has been insuring drivers for over thirty years. TIC offers varying insurance packages from person to property and vehicle options. The company's overall mission is to offer their customers a peace of mind when it comes to life's uncertainties (i.e. fire, burglary, long term disability, etc.). To meet their goal, TIC offers customizable insurance policies to meet their clients' needs. The company's customer base consists of wildly varying races in all socio-economic statuses.

To remain competitive, meet their mission, and give the best options to their customers, TIC wishes to begin offering caravan insurance under their vehicle insurance branch. TIC's marketing department has used direct mailings to potential customers with a high level of success in the past. Currently, the major limitation of this strategy is that customers are throwing away the direct mailings because it is viewed as 'junk mail' (Putten, n.d.). With most of these mailings now thrown away, this is an inefficient use of resources and marketing budget. The head of the marketing department wishes to still use direct mailings as a marketing tool, but to create a targeted list to have a higher probability of success. TIC has hired the SNHU Data Analytics teams to answer their research question: "Can we predict who would be interested in buying a caravan insurance policy and why?" (Putten, n.d.).

In preparation of this analysis, TIC has released three data sets from Sentient Machine Research that already have done extensive datamining. The data set labeled 'TICDATA.txt' has 5,822 descriptions of customers and "…includes product usage data and socio-demographic data" (Putten, n.d.). The team will use this data set to evaluate, analyze, and validate prediction models. The data set's socio-demographic variables are, but not limited to, religion, education, and income. The data set labeled 'TICEVAL2000.txt' is the testing data once the model is created and can be applied. The data set labeled 'TICTGT2000.txt' is used as part of the evaluation set to allow for

validation of current models and revise as needed. All three data sets that were given are highly dense and provide a plethora of information about their customers.

b. Potential Value:

As stated earlier, the current method of marketing (i.e. sending out mail to as many people as possible for a potential return) is no longer working as well as it once did. For TIC to increase the potential of a sale, specifically a sale of caravan insurance, a strategic marketing plan is needed. The strategic marketing campaign would benefit highly from a predictive analytical strategy. The potential value is compounded because this strategy can be adapted for marketing other types of insurance in the future, customizing individual insurance plans for higher customer satisfaction, and so much more. The predictive analytical strategy will help the marketing team identify future trends and customer needs, keeping TIC a proactive force on a highly competitive market, where reacting to trends can lead to significant customer and revenue loss. Another benefit of the predictive analytical strategy is the identification and mitigation of risk, especially as new economic and social variables arise (i.e. COVID-19 and change in insurance laws due to political change in the U.S.). Although TIC is an insurance company, it is predominately a customer service company. A predictive analytical strategy will help TIC identify individual buying behaviors to create a more customizable insurance option for the customer thus leading to higher customer satisfaction and profits.

II. Predictive Algorithms

a. Specifications:

The Insurance Company (TIC) has tasked the SNHU Data Analytics Team to use analytical methodology to predict potential customers that would most likely purchase a caravan insurance policy rather than blindly mailing out flyers. To increase current levels of caravan policy holders,

TIC has shared historical data from current policy holders to find trends, similarities, etc. to predict whether those customers would most likely add caravan policies.

The historical data allows for the basis for a comprehensive analysis using predictive methodologies. To start, a descriptive analysis will be applied that "…primarily build models based on Regression and Decision Trees" ("How To Choose An Algorithm For Predictive Analytics?", 2018). These algorithms focus on finding the relationship between the variables and attributes ("How To Choose An Algorithm For Predictive Analytics?", 2018). The initial descriptive analysis will focus on the variables and attributes of current policy holders compared to policy holders that do not have caravan insurance. This analysis will create a clear picture of what variables and attributes have a significant impact on probability of purchase.

Data mining and analysis offer significant avenues of model creation for TIC, but the ones that will best serve the company's needs are cluster analysis and regression analysis. Regression analysis is widely used at the descriptive analysis stage and "…is useful for evaluating multiple independent variables" (Regression Analysis, 2020). These independent variables can be used to identify trends. Cluster analysis would fit TIC's needs because it creates "…a number of clusters based on the observed values of several variables for each individual" ("Cluster Analysis", 2018). Cluster analysis would be highly beneficial because each policy holder with single or multiple policies has several variables that need to be analyzed (i.e., age, location, income status, current policies, etc.). Once these analyses are completed, current policy holders that hold low probability of buying the caravan policy can be deleted from the mailing list, but new individuals will be added later.

b. Predictive Algorithm Recommendation:

To conduct the analysis, the SNHU Analytics team will adopt a random forest algorithm. A random forest algorithm will allow the team to quickly build hundreds or thousands, depending on what is needed, of decision trees to create a high level of accuracy predictive model to meet TICs needs. To create the random algorithm, the team will use the original data set (ticdata.txt) to create

the training data set, which will be used to identify which customers will be most likely to adopt a new caravan insurance policy. Once the model is created, it will be applied to the evaluation data set (ticeval2000.txt) that was provided by TIC to develop a working list of customers who will most likely buy a new caravan insurance policy with a high level of accuracy.

III. Data Analytic Tools

The SNHU Data Analytics team will be using two popular and open-source data analysis tools: RStudio and the Rattle library that is built into the R statistical programming language. RStudio is an integrated development environment (IDE) made specifically for R. It runs on Windows, Mac and Linux desktop environments ("RStudio", 2022). It also includes tools for plotting and direct code execution. Part of the model, as will be seen below, will be executed in RStudio using direct code execution that is written in Appendix A. Rattle is a graphical user interface for data mining using R ("Togaware: Rattle: A Graphical User Interface for Data Mining using R", 2022). Rattle allows analysts to "…present statistical and visual summaries of data, transforms data so that it can be readily modelled, builds both unsupervised and supervised machine learning models from the data, presents the performance of models graphically, and scores new datasets for deployment into production" ("Togaware: Rattle: A Graphical User Interface for Data Mining using R", 2022). Rattle and RStudio have powerful and complementary tools that will allow the team to create, evaluate, and revise the predictive model before full implementation.

IV. Model Optimization

a. Model Evaluation

As part of the overall analysis, the team must evaluate our current model to ensure results have not been over-fitted or have a high rate of error. Evaluating the model will also allow the team to best understand how the model will work with new data input in the future so that the model is not a one-time use application, but an evolving application. There are many ways to evaluate models, but

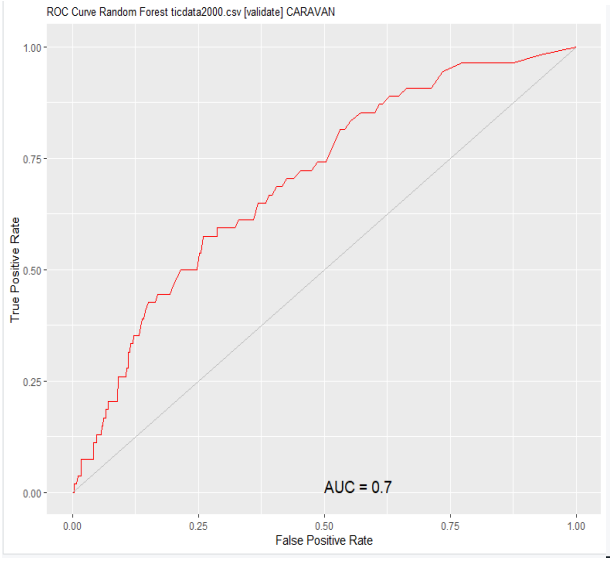
the team has chosen the confusion matrix, Receiver Operating Characteristic (ROC) curve, and Area Under Curve (AUC) techniques.

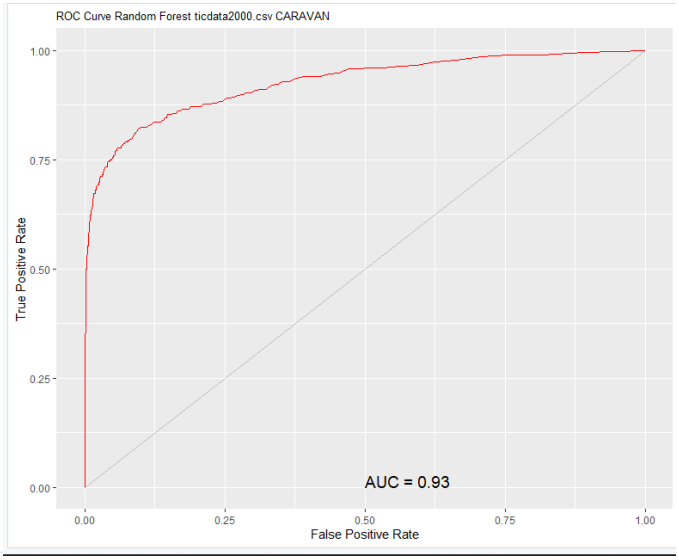
A confusion matrix takes the form of a square matrix where the columns represent the actual values, and the rows represent the predicted value of the model (Tyagi, 2021). Its purpose is to present "…the ways in which a classification model becomes confused while making predictions" (Tyagi, 2021). This technique will allow the SNHU Data Analysis Team and others to see what types are being created: True Positive, True Negative, False Positive and False Negative. The types of errors and the level of severity of said errors will help the team determine the accuracy of the model and, if low accuracy, how to strengthen the model.

```
Error matrix for the Random Forest model on ticdata2000.csv [validate] (counts):

      Predicted
Actual   0 1 Error
     0 816 3   0.4
     1  54 0 100.0

Error matrix for the Random Forest model on ticdata2000.csv [validate] (proportions):

      Predicted
Actual    0   1 Error
     0 93.5 0.3   0.4
     1  6.2 0.0 100.0

Overall error: 6.5%, Averaged class error: 50.2%

Rattle timestamp: 2022-02-27 19:34:03 Profe
======================================================================
```

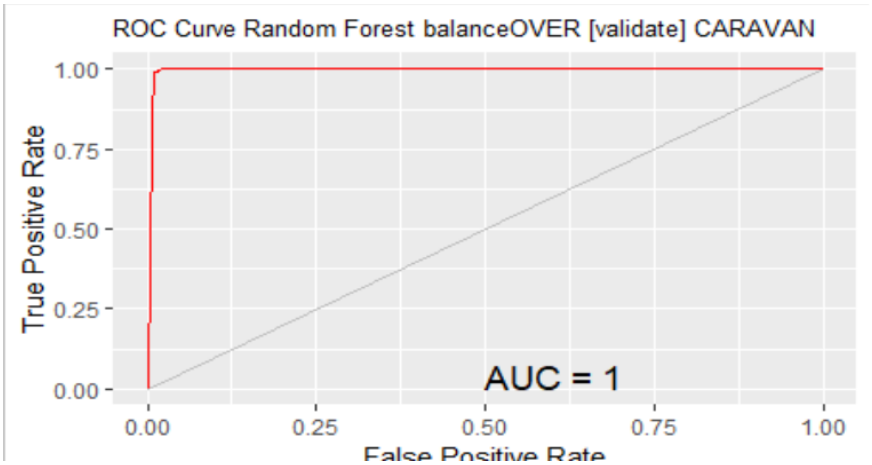
As we see above, the overall error is 6.5%, which is a good indicator that the model is working accurately as planned.

A Receiver Operating Characteristic (ROC) curve is a graph that depicts a curve to depict the false positive rate (FPR) against the true positive rate (TPR) at various threshold settings (Williams, 2013). Another indicator is the Area Under Curve (AUC) which is the area under the ROC curve and the closer the AUC is closer to 1 the more accurate the TPR is.





As shown above, the ROC on the left has an AUC of 0.7 using the validation data, which the ROC on the right has an AUC of 0.93 using the full data set. These numbers are not ideal thus, the model needs to be adjusted to improve accuracy. To do this, the team, will apply the oversampling technique to adjust for imbalances within the classes. The results of the oversampling technique show a drastic improvement of the AUC level of 1 rather than 0.7 or 0.93.



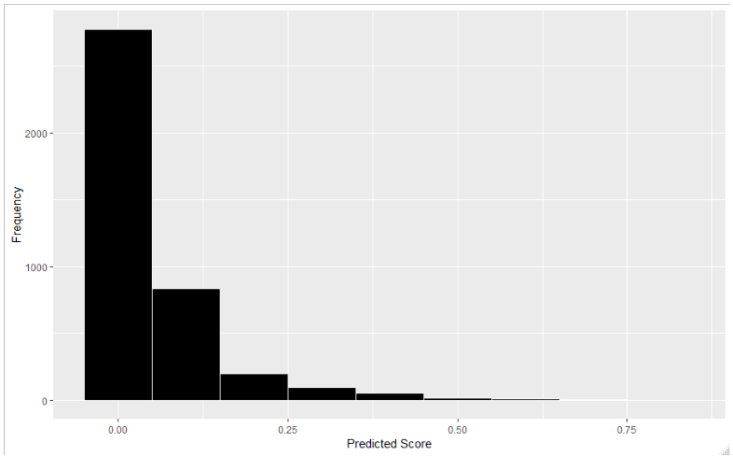
With the new model updated with the oversampling technique, the team notes that this model is now the most accurate possible. With this high level of accuracy, TIC can predict which customers would most likely purchase a caravan insurance policy with a high degree of accuracy.

The team wishes to note that the random forest can be used to understand the overall key variables the influence the customer's purchasing capability and probability. In the random forest, it outputs the variables in order of importance. Different variables do make the customer more likely

to buy a caravan insurance policy such as individuals who have auto and fire policies who are at the higher contribution range with TIC are more likely to purchase more insurance including caravan policies than other existing policy holders.

b. Implementation Steps

Now that the model has been evaluated and tested to better suite TIC's needs, the team can now create the implementation steps needed to apply the model. As stated earlier, the model needs to be able to continue accurately predicting who would purchase a new caravan policy as time goes on rather than the model being a one-time use application. Rattle can evaluate the ability to predict with new data sets using a scoring report. The scores of the report range from 0 to 1 with scores closet to 1 being the most likely to occur. In TIC search for customers to buy caravan insurance policies, the customers who score closest to 1 are the customers that will most likely buy caravan insurance. With the team's model loaded, the following report was created using Rattle's scoring method:



As show above, most of the data falls in line with our initial evaluation made earlier that a small percentage (22 out of 4000) would buy caravan insurance. Unfortunately, none of the entries were scored as an absolute one or close to it, which means our model will need adjustments before final implementation. To maintain the model and refine it over time, a feedback system can be implemented. TIC will have to determine appropriate thresholds for selecting customers from the

model and, depending on whether those thresholds are met, the model can be refined until the correct thresholds are found (Williams, 2013).

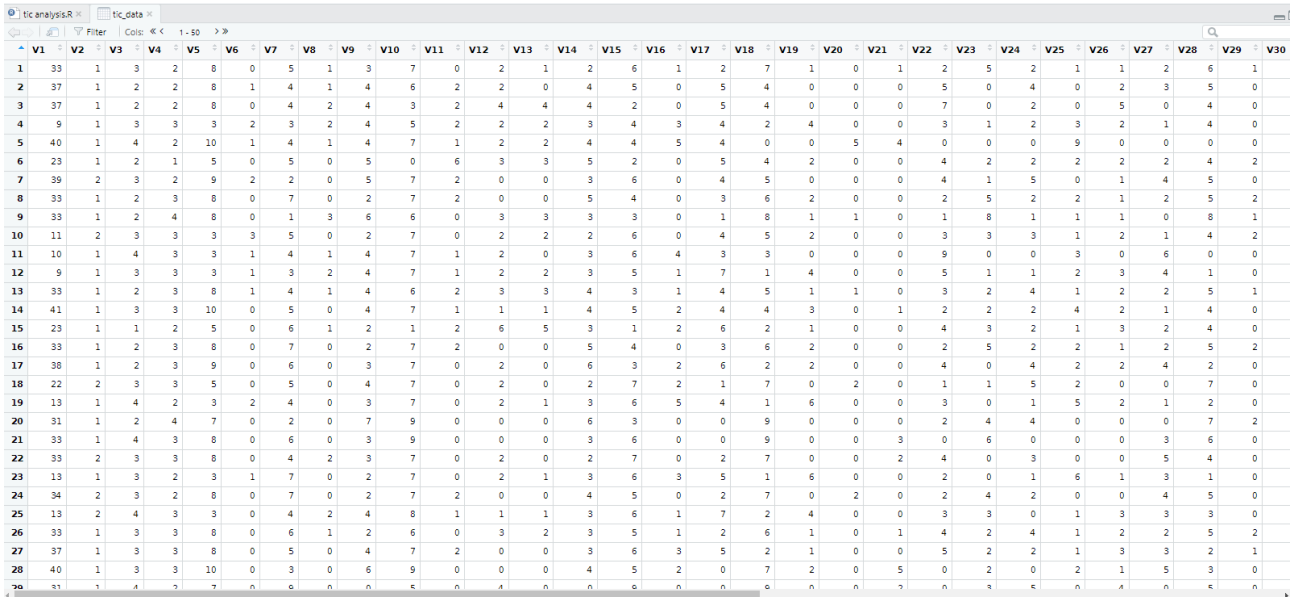
To implement this model throughout the company and with various systems and servers, TIC can implement the use of Predictive Modeling Markup Language (PMML), which can be done when needed so that it is less costly and easy for non-analysts to use (PMML. Definition, 2021). PMML works very well with very well with existing servers, which will save the company money because it will integrate existing data warehouses into the model (Williams, 2013). While this is a less expensive option, it can have issues with performance. Another way for the model to have a feedback system is to have sales personnel and sales managers to give feedback on the customers that were identified so that a better understanding of the level of interest, success rate, and other relevant identifiers as time goes on.

V. Reproducible Research

The SNHU Data Analytics team will be using RStudio to create the random forest analysis. As with all great research, a standard methodology of the SNHU Data Analytics team is to document procedures for it to be easily reproducible in various scenarios. If you wish to read the R script in its entirety, we have provided it in Appendix A at the end of the document.

We will load the data set into RStudio to create our training data and ensure it loaded without error.

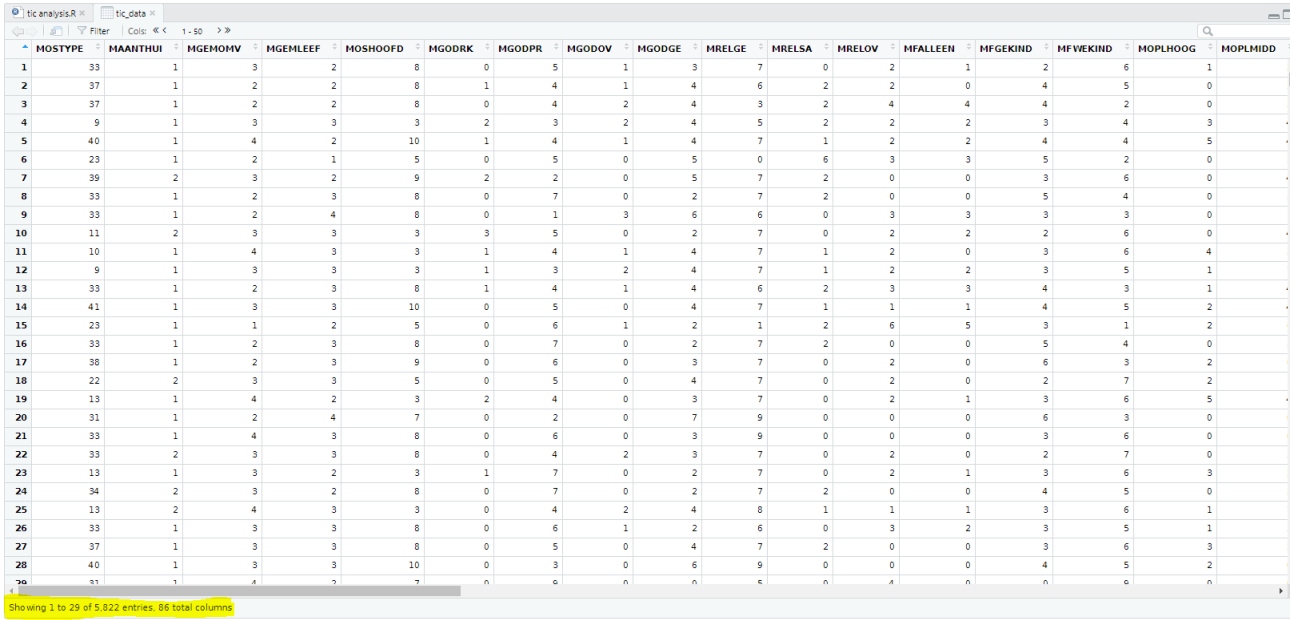
```
#import data
tic_data<-read.delim('C:/Users/Profe/Documents/ticdata2000.txt', header=FALSE)

#view data
View(tic_data)
```



	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
1	33	1	3	2	8	0	5	1	3	7	0	2	1	2	6	1	2	7	1	0	1	2	5	2	1	1	2	6	1	
2	37	1	2	2	8	1	4	1	4	6	2	2	0	4	5	0	5	4	0	0	0	5	0	4	0	2	3	5	0	
3	37	1	2	2	8	0	4	2	4	3	2	4	4	4	2	0	5	4	0	0	0	7	0	2	0	5	0	4	0	
4	9	1	3	3	3	2	3	2	4	5	2	2	2	3	4	3	4	2	4	0	0	3	1	2	3	2	1	4	0	
5	40	1	4	2	10	1	4	1	4	7	1	2	2	4	4	5	4	0	0	5	4	0	0	0	9	0	0	0	0	
6	23	1	2	1	5	0	5	0	5	0	6	3	3	5	2	0	5	4	2	0	0	4	2	2	2	2	2	4	2	
7	39	2	3	2	9	2	2	0	5	7	2	0	0	3	6	0	4	5	0	0	0	4	1	5	0	1	4	5	0	
8	33	1	2	3	8	0	7	0	2	7	2	0	0	5	4	0	3	6	2	0	0	2	5	2	2	1	2	5	2	
9	33	1	2	4	8	0	1	3	6	6	0	3	3	3	3	0	1	8	1	1	0	1	8	1	1	1	0	8	1	
10	11	2	3	3	3	3	5	0	2	7	0	2	2	2	6	0	4	5	2	0	0	3	3	3	1	2	1	4	2	
11	10	1	4	3	3	1	4	1	4	7	1	2	0	3	6	4	3	3	0	0	0	9	0	0	3	0	6	0	0	
12	9	1	3	3	3	1	3	2	4	7	1	2	2	3	5	1	7	1	4	0	0	5	1	1	2	3	4	1	0	
13	33	1	2	3	8	1	4	1	4	6	2	3	3	4	3	1	4	5	1	1	0	3	2	4	1	2	2	5	1	
14	41	1	3	3	10	0	5	0	4	7	1	1	1	4	5	2	4	4	3	0	1	2	2	2	4	2	1	4	0	
15	23	1	1	2	5	0	6	1	2	1	2	6	5	3	1	2	6	2	1	0	0	4	3	2	1	3	2	4	0	
16	33	1	2	3	8	0	7	0	2	7	2	0	0	5	4	0	3	6	2	0	0	2	5	2	2	1	2	5	2	
17	38	1	2	3	9	0	6	0	3	7	0	2	0	6	3	2	6	2	2	0	0	4	0	4	2	2	4	2	0	
18	22	2	3	3	5	0	5	0	4	7	0	2	0	2	7	2	1	7	0	2	0	1	1	5	2	0	0	7	0	
19	13	1	4	2	3	2	4	0	3	7	0	2	1	3	6	5	4	1	6	0	0	3	0	1	5	2	1	2	0	
20	31	1	2	4	7	0	2	0	7	9	0	0	0	6	3	0	0	9	0	0	0	2	4	4	0	0	0	7	2	
21	33	1	4	3	8	0	6	0	3	9	0	0	0	3	6	0	0	9	0	0	0	3	0	6	0	0	0	3	6	0
22	33	2	3	3	8	0	4	2	3	7	0	2	0	2	7	0	2	7	0	0	2	4	0	3	0	0	5	4	0	
23	13	1	3	2	3	1	7	0	2	7	0	2	1	3	6	3	5	1	6	0	0	2	0	1	6	1	3	1	0	
24	34	2	3	2	8	0	7	0	2	7	2	0	0	4	5	0	2	7	0	2	0	2	4	2	0	0	4	5	0	
25	13	2	4	3	3	0	4	2	4	8	1	1	1	3	6	1	7	2	4	0	0	3	3	0	1	3	3	3	0	
26	33	1	3	3	8	0	6	1	2	6	0	3	2	3	5	1	2	6	1	0	1	4	2	4	1	2	2	5	2	
27	37	1	3	3	8	0	5	0	4	7	2	0	0	3	6	3	5	2	1	0	0	5	2	2	1	3	3	2	1	
28	40	1	3	3	10	0	3	0	6	9	0	0	0	4	5	2	0	7	2	0	5	0	2	0	2	1	5	3	0	

The data does not have the necessary columns to allow for further analysis, so the team manually added the columns to the data set.

```
colnames(tic_data) <- c("MOSTYPE","MAANTHUI","MGEMOMV","MGEMLEEF","MOSHOOFD","MGODRK","MGODPR","MGODOV","MGODGE","MRELGE","MRELSA","MRELOV","MFALLEEN","MFGEKIND","MFWEKIND","MOPLHOOG"
,"MOPLMIDD","MOPLLAAG","MBERHOOG","MBERZELF","MBERBOER","MBERMIDD","MBERARBG","MBERARBO","MSKA","MSKB1","MSKB2","MSKC","MSKD","MHHUUR","MHKOOP","MAUT1","MAUT2","MAUT0","MZFONDS","MZPART"
,"MINKM30","MINK3045","MINK4575","MINK7512","MINK123M","MINKGEM","MKOOPKLA","PWAPART","PWABEDR","PWALAND","PPERSAUT","PBESAUT","PMOTSCO","PVRAAUT","PAANHANG","PTRACTOR","PWERKT","PBROM"
,"PLEVEN","PPERSONG","PGEZONG","PWAOREG","PBRAND","PZEILPL","PPLEZIER","PFIETS","PINBOED","PBYSTAND","AWAPART","AWABEDR","AWALAND","APERSAUT","ABESAUT","AMOTSCO","AVRAAUT","AAANHANG",
"ATRACTOR","AWERKT","ABROM","ALEVEN","APERSONG","AGEZONG","AWAOREG","ABRAND","AZEILPL","APLEZIER","AFIETS","AINBOED","ABYSTAND","CARAVAN")
```



	MOSTYPE	MAANTHUI	MGEMOMV	MGEMLEEF	MOSHOOFD	MGODRK	MGODPR	MGODOV	MGODGE	MRELGE	MRELSA	MRELOV	MFALLEEN	MFGEKIND	MFWEKIND	MOPLHOOG	MOPLMIDD
1	33	1	3	2	8	0	5	1	3	7	0	2	1	2	6	1	
2	37	1	2	2	8	1	4	1	4	6	2	2	0	4	5	0	
3	37	1	2	2	8	0	4	2	4	3	2	4	4	4	2	0	
4	9	1	3	3	3	2	3	2	4	5	2	2	2	3	4	3	
5	40	1	4	2	10	1	4	1	4	7	1	2	2	4	4	5	
6	23	1	2	1	5	0	5	0	5	0	6	3	3	5	2	0	
7	39	2	3	2	9	2	2	0	5	7	2	0	0	3	6	0	
8	33	1	2	3	8	0	7	0	2	7	2	0	0	5	4	0	
9	33	1	2	4	8	0	1	3	6	6	0	3	3	3	3	0	
10	11	2	3	3	3	3	5	0	2	7	0	2	2	2	6	0	
11	10	1	4	3	3	1	4	1	4	7	1	2	0	3	6	4	
12	9	1	3	3	3	1	3	2	4	7	1	2	2	3	5	1	
13	33	1	2	3	8	1	4	1	4	6	2	3	3	4	3	1	
14	41	1	3	3	10	0	5	0	4	7	1	1	1	4	5	2	
15	23	1	1	2	5	0	6	1	2	1	2	6	5	3	1	2	
16	33	1	2	3	8	0	7	0	2	7	2	0	0	5	4	0	
17	38	1	2	3	9	0	6	0	3	7	0	2	0	6	3	2	
18	22	2	3	3	5	0	5	0	4	7	0	2	0	2	7	2	
19	13	1	4	2	3	2	4	0	3	7	0	2	1	3	6	5	
20	31	1	2	4	7	0	2	0	7	9	0	0	0	6	3	0	
21	33	1	4	3	8	0	6	0	3	9	0	0	0	3	6	0	
22	33	2	3	3	8	0	4	2	3	7	0	2	0	2	7	0	
23	13	1	3	2	3	1	7	0	2	7	0	2	1	3	6	3	
24	34	2	3	2	8	0	7	0	2	7	2	0	0	4	5	0	
25	13	2	4	3	3	0	4	2	4	8	1	1	1	3	6	1	
26	33	1	3	3	8	0	6	1	2	6	0	3	2	3	5	1	
27	37	1	3	3	8	0	5	0	4	7	2	0	0	3	6	3	
28	40	1	3	3	10	0	3	0	6	9	0	0	0	4	5	2	

Showing 1 to 29 of 5,822 entries, 86 total columns

As shown above, it created a total of 86 labeled columns and the data set has 5,822 entries, which is a decent size data set for the team to utilize in analysis. To create the model for the random forest, the team must now identify the target variable, which is a dependent variable labeled ‘CARAVAN’.

```
#setting target variable as categorical and assigned to dependent variable CARAVAN
tic_data$CARAVAN <-as.factor(tic_data$CARAVAN)
```

This will also make the target variable to be categorical. Next the team will use a common methodology by splitting the data set into a training data set and a validation data set. The training data set will be made of 70% of the original data set and the validation data set will be made of the remaining 30%. While this methodology was conducted only in RStudio, it can be applied using Rattle as well.

```
#split our data set into training and validation data sets. 70% of original will be for training and 30% will be for validation
set.seed(100)
train <- sample(nrow(tic_data), 0.7*nrow(tic_data), replace = FALSE)
TrainSet <- tic_data[train,]
ValidSet <- tic_data[train,]
```

To ensure that the data sets were split correctly and see how randomly partitioned it was, the team can run a quick summary output using the summary() function.

```
CARAVAN        CARAVAN
0:3817         0:1657
1: 258         1:   90
```

Our training set shows 258/3817 or 6.7% of the entries already have customers with an insurance policy. The validation data shows 90/1657 or 5.43% of the entries have an insurance policy. The reason to keep these numbers low is we do not want the policy holders to hold majority of either data set, which can skew the output, and the point is to identify individuals that would buy a new insurance policy not ones that already have them.

Now that the data sets are created and categorical with the correct target variable, the team can begin creating the appropriate models. We will first create a model with no limitations and, if needed we can add the appropriate limitations to reduce the error rate.

```
#Create first model (modelA) using random forest and training set, this is a general model with no limitations
modelA <- randomForest(Caravan ~ ., data = TrainSet, importance = TRUE)
```

Model A has an error rate of 7.31%, which means that any predictions created from the model have a 92.69% accuracy, but this error rate could be reduced to create higher accuracy from the model's

output. We will create a second model called ‘modelB’ to see if this possible using 1000 decision trees and 6 variables.

```
#create second model (modelB) by running the data through 1000 decision trees and only 6 variables
modelB <- randomForest(Caravan ~ ., data = TrainSet,ntree = 1000, mtry = 6, importance = TRUE)
```

Model B produces a lower error rate at 7.12%, which means that predictions made from this model have a level of 92.88% accuracy. Since this is a lower error rate, the team will adopt modelB as the model to apply to the training set.

```
#create predication based on training set and applying model B
PTrain <- predict(modelB, TrainSet, type = "class")
table(PTrain, TrainSet$CARAVAN)
```

```
PTrain    0    1
     0 3815   96
     1    2  162
> |
```

Next, the team will apply modelB to the validation set to create predictions based on the mean of the target variable CARAVAN. The validated data was sent to its own table.

```
#Create validation model predictions using model b and valid se
PValid <- predict(modelB, ValidSet, type = "class")
mean(PValid == ValidSet$CARAVAN)

#Create a table of the validated set
table(PValid, ValidSet$CARAVAN)
```

```
> PValid <- predict(modelB, ValidSet, type = "class")
> mean(PValid == ValidSet$CARAVAN)
[1] 0.9759509
> table(PValid, ValidSet$CARAVAN)

PValid    0    1
     0 3815   96
     1    2  162
> |
```

It should be noted that both the PTrain (predictions from the training data set) and the PValid (predictions from the validation data set) have the exact same output, because the model the team applied had a high level of accuracy.

The team feels that it is now appropriate to apply modelB to the evaluation data set (ticeval2000.txt) that was given by TIC in order to identify the predictions of possible new caravan

insurance buyers. To do this, the team had to import the evaluation data set and create the columns similar to that of the training data set. The evaluation data set has 85 columns without CARAVAN, so the team had to add the column and input the data as 'NA' before manually adding the column names.

```
#Next Level of Analysis: adding eval data to apply to our model, first we need to import and assign columns like before
tic_eval<-read.delim('C:/Users/Profe/Documents/ticeval2000.txt', header=FALSE)
tic_eval$x86 <- NA
colnames(tic_eval) <- c("MOSTYPE","MAANTHUI","MGEMOMV","MGEMLEEF","MOSHOOFD","MGODRK","MGODPR","MGODOV","MGODGE","MRELGE",  "MRELSA","MRELOV","MFALLEEN","MFGEKIND","MFWEKIND"
,"MOPLHOOG","MOPLMIDD","MOPLLAAG","MBERHOOG", "MBERZELF","MBERBOER","MBERMIDD","MBERARBG","MBERARBO","MSKA","MSK81","MSKB2","MSKC","MSKD","MHHUUR", "MHKOOP","MAUT1","MAUT2"
,"MAUTO","MZFONDS","MZPART","MINKM30","MINK3045","MINK4575","MINK7512","MINK123M", "MINKGEM","MKOOPKLA","PWAPART","PWABEDR","PWALAND","PPERSAUT","PBESAUT","PMOTSCO","PVRAAUT"
,"PAANHANG","PTRACTOR","PWERKT","PBROM","PLEVEN","PPERSONG","PGEZONG","PWAOREG","PBRAND","PZEILPL","PPLEZIER","PFIETS", "PINBOED","PBYSTAND","AWAPART","AWABEDR","AWALAND"
,"APERSAUT","ABESAUT","AMOTSCO","AVRAAUT","AAANHANG", "ATRACTOR","AWERKT","ABROM","ALEVEN","APERSONG","AGEZONG","AWAOREG","ABRAND","AZEILPL","APLEZIER","AFIETS", "AINBOED"
,"ABYSTAND","CARAVAN")
```

tic analysis.R* | tic_eval | tic_data

Filter | Cols: | 1 - 50

	MOSTYPE	MAANTHUI	MGEMOMV	MGEMLEEF	MOSHOOFD	MGODRK	MGODPR	MGODOV	MGODGE	MRELGE	MRELSA	MRELOV	MFALLEEN	MFGEKIND	MFWEKIND	MOPLHOOG
1	33	1	4	2	8	0	6	0	3	5	0	4	1	1	8	
2	6	1	3	2	2	0	5	0	4	5	2	2	1	4	5	
3	39	1	3	3	9	1	4	2	3	5	2	3	2	3	6	
4	9	1	2	3	3	2	3	2	4	5	4	1	2	4	4	
5	31	1	2	4	7	0	2	0	7	9	0	0	0	6	3	
6	30	1	2	4	7	1	4	2	3	5	0	4	4	3	2	
7	35	1	2	4	8	2	5	1	2	8	0	1	2	5	3	
8	6	1	3	3	2	3	4	2	2	9	0	0	0	5	4	
9	4	1	2	4	1	0	7	2	0	9	0	0	1	7	2	
10	10	1	4	2	3	0	7	0	2	9	0	0	0	2	7	
11	34	1	3	3	8	0	9	0	0	5	0	4	2	3	4	
12	36	1	3	3	8	2	1	0	6	8	0	1	1	1	7	
13	38	1	3	2	9	0	6	3	0	9	0	0	0	4	5	
14	11	1	2	3	3	1	3	1	5	4	2	4	4	2	3	
15	40	1	2	4	10	0	6	0	3	6	0	3	3	4	3	
16	25	1	1	4	6	0	5	2	3	3	1	6	6	3	1	
17	41	1	3	3	10	1	7	1	2	8	1	1	1	3	6	
18	39	2	5	3	9	0	6	0	3	9	0	0	0	3	6	
19	23	1	1	1	5	1	4	2	3	4	1	4	5	4	1	
20	22	1	2	4	5	0	5	2	2	9	0	0	0	6	3	
21	33	1	2	2	8	0	4	2	4	3	2	4	4	4	2	
22	13	1	3	3	3	0	3	0	6	7	0	2	2	2	6	
23	5	1	2	4	1	0	5	3	2	5	1	4	3	4	3	
24	39	1	4	2	9	2	4	2	2	8	0	1	1	3	6	
25	32	1	3	3	7	1	4	2	3	7	1	2	2	3	4	
26	24	1	2	5	5	1	5	1	3	4	1	5	5	3	2	
27	12	1	4	2	3	0	7	2	0	9	0	0	0	0	8	
28	11	1	3	3	3	1	4	1	4	7	1	2	2	4	4	

The evaluation data only has 4000 entries, which is only slightly less than the training data set.

To apply modelB to the evaluation data set, the team create a table with the predictions called 'PEval'.

```
#Use Model B on the evaluation data set created above
PEval <- predict(modelB, tic_eval, type = "class")
tic_eval$CARAVAN <- PEval
View(tic_eval)
table(tic_eval$CARAVAN)
```

```
   0    1
3978   22
>
```

The output of the PEval table shows 22 out of the 4000 individuals have a high level of accuracy to buy a caravan insurance policy. While this is not a high level of the overall data set, these 22 entries are the most likely to buy a new policy. Once the predictions are created, the outputs are saved as an external file.

```
#Save as external CSV file
write.csv(PEval, file = "C:/Users/Profe/Documents/final_prediction_analysis_output.csv")
```

References:

Cluster Analysis. (2018). Retrieved 9 February 2022, from https://www.sciencedirect.com/topics/medicine-and-dentistry/cluster-analysis

Docs.intersystems.com. 2021. *PMML.Definition*. [online] Available at: <https://docs.intersystems.com/irislatest/csp/documatic/%25CSP.Documatic.cls?LIBRARY=%25SYS&CLASSNAME=%25DeepSee.PMML.Definition> [Accessed 28 February 2022].

How To Choose An Algorithm For Predictive Analytics?. (2018). Retrieved 9 February 2022, from https://www.bistasolutions.com/resources/blogs/how-to-choose-an-algorithm-for-predictive-analytics/

Putten, P. van der. (n.d.). *CoIL Challenge 2000 Report*. Coil Challenge 2000 Report. Retrieved January 17, 2022, from https://liacs.leidenuniv.nl/~puttenpwhvander/library/cc2000/

Regression Analysis. (2020). Retrieved 9 February 2022, from https://sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/BS704_Multivariable/BS704_Multivariable6.html\

RStudio. (2022). Retrieved 7 March 2022, from https://www.rstudio.com/products/rstudio/

Togaware: Rattle: A Graphical User Interface for Data Mining using R. (2022). Retrieved 7 March 2022, from https://rattle.togaware.com/

Tyagi, N., 2021. *What is Confusion Matrix? | Analytics Steps*. [online] Analyticssteps.com. Available at: <https://www.analyticssteps.com/blogs/what-confusion-matrix> [Accessed 26 February 2022].

Williams, G.J. (2013). Data Mining with Rattle and R: The art of excavating data for knowledge discovery. New York: Springer.

Appendix A: R Source Code

```
#import data
tic_data<-read.delim('C:/Users/Profe/Documents/ticdata2000.txt', header=FALSE)

#view data
View(tic_data)

#create columns for analysis

colnames(tic_data) <-
        c("MOSTYPE","MAANTHUI","MGEMOMV","MGEMLEEF","MOSHOOFD","MGODR
        K","MGODPR","MGODOV","MGODGE","MRELGE",
        "MRELSA","MRELOV","MFALLEEN","MFGEKIND","MFWEKIND","MOPLHOOG","
        MOPLMIDD","MOPLLAAG","MBERHOOG",
        "MBERZELF","MBERBOER","MBERMIDD","MBERARBG","MBERARBO","MSKA","
        MSKB1","MSKB2","MSKC","MSKD","MHHUUR",
        "MHKOOP","MAUT1","MAUT2","MAUT0","MZFONDS","MZPART","MINKM30","MI
        NK3045","MINK4575","MINK7512","MINK123M",
        "MINKGEM","MKOOPKLA","PWAPART","PWABEDR","PWALAND","PPERSAUT","P
        BESAUT","PMOTSCO","PVRAAUT","PAANHANG","PTRACTOR","PWERKT","PBRO
        M","PLEVEN","PPERSONG","PGEZONG","PWAOREG","PBRAND","PZEILPL","PPLE
        ZIER","PFIETS",
        "PINBOED","PBYSTAND","AWAPART","AWABEDR","AWALAND","APERSAUT","A
        BESAUT","AMOTSCO","AVRAAUT","AAANHANG",
        "ATRACTOR","AWERKT","ABROM","ALEVEN","APERSONG","AGEZONG","AWAO
        REG","ABRAND","AZEILPL","APLEZIER","AFIETS",
        "AINBOED","ABYSTAND","CARAVAN")

#view formatted data with new column headings
View(tic_data)

#setting target variable as categorical and assigned to dependent variable CARAVAN
tic_data$CARAVAN <-as.factor(tic_data$CARAVAN)

#split our data set into training and validation data sets. 70% of original will be for training and
        30% will be for validation
set.seed(100)
train <- sample(nrow(tic_data), 0.7*nrow(tic_data), replace = FALSE)
TrainSet <- tic_data[train,]
ValidSet <- tic_data[train,]

#summary analysis
summary(TrainSet)
summary(ValidSet)

#Create first model (modelA) using random forest and training set, this is a general model with no
        limitations
modelA <- randomForest(CARAVAN ~ ., data = TrainSet, importance = TRUE)
```

```
#create second model (modelB) by running the data through 1000 decision trees and only 6
        variables
modelB <- randomForest(CARAVAN ~ ., data = TrainSet,ntree = 1000, mtry = 6, importance =
        TRUE)

#create predication based on training set and applying model B
PTrain <- predict(modelB, TrainSet, type = "class")
table(PTrain, TrainSet$CARAVAN)

#Create validation model predictions using model b and valid set
PValid <- predict(modelB, ValidSet, type = "class")
mean(PValid == ValidSet$CARAVAN)

#Create a table of the validated set
table(PValid, ValidSet$CARAVAN)

#Next Level of Analysis: adding eval data to apply to our model, first we need to import and assign
        columns like before
tic_eval<-read.delim('C:/Users/Profe/Documents/ticeval2000.txt', header=FALSE)
tic_eval$x86 <- NA
colnames(tic_eval) <-
        c("MOSTYPE","MAANTHUI","MGEMOMV","MGEMLEEF","MOSHOOFD","MGODR
        K","MGODPR","MGODOV","MGODGE","MRELGE",
        "MRELSA","MRELOV","MFALLEEN","MFGEKIND","MFWEKIND","MOPLHOOG","
        MOPLMIDD","MOPLLAAG","MBERHOOG",
        "MBERZELF","MBERBOER","MBERMIDD","MBERARBG","MBERARBO","MSKA","
        MSKB1","MSKB2","MSKC","MSKD","MHHUUR",
        "MHKOOP","MAUT1","MAUT2","MAUT0","MZFONDS","MZPART","MINKM30","MI
        NK3045","MINK4575","MINK7512","MINK123M",
        "MINKGEM","MKOOPKLA","PWAPART","PWABEDR","PWALAND","PPERSAUT","P
        BESAUT","PMOTSCO","PVRAAUT","PAANHANG","PTRACTOR","PWERKT","PBRO
        M","PLEVEN","PPERSONG","PGEZONG","PWAOREG","PBRAND","PZEILPL","PPLE
        ZIER","PFIETS",
        "PINBOED","PBYSTAND","AWAPART","AWABEDR","AWALAND","APERSAUT","A
        BESAUT","AMOTSCO","AVRAAUT","AAANHANG",
        "ATRACTOR","AWERKT","ABROM","ALEVEN","APERSONG","AGEZONG","AWAO
        REG","ABRAND","AZEILPL","APLEZIER","AFIETS",
        "AINBOED","ABYSTAND","CARAVAN")

View(tic_eval)

#Use Model B on the evaluation data set created above
PEval <- predict(modelB, tic_eval, type = "class")
tic_eval$CARAVAN <- PEval
View(tic_eval)
table(tic_eval$CARAVAN)

#Save as external CSV file
```

```
write.csv(PEval, file =
        "C:/Users/Profe/Documents/final_prediction_analysis_output.csv")diction_analysis_output.c
        sv")
```