**Final Project**

**Team:**

1. Dileep Gudena (dxg161730)
2. Gokul Surendra (gxs161530)
3. Chaithra Mysore Nagaraj (cxm161130)

**Purpose:** To build a classification model that will predict whether the customer own a mobile home insurance policy.

**Dataset:** Insurance Company Benchmark

**Approach:**

1. **Pre-Processing:**

* We have 3 dataset files (training & test datasets), after appropriately binding those files we discovered 86 attributes and 9822 observations in total. Looking at the summary of the data, we found out many categorical variables. So, we factor all the attributes and put it the data frame with the following command.

ticcompl <- as.data.frame(ticcompl)

ticcompl <- lapply(ticcompl, as.factor)

* Now, by looking at the summary, we observed attributes with few NA’s. Removed such values, as it adds no information.

ticcompl = na.omit(ticcompl)

* We use nearZeroVar to remove predicators that have large ratio of the frequency of the most common value to the frequency of the second most common value (near-zero variance predictors).

We delete all these attributes, as they don’t add any information to the data.

ticcompl$PWABEDR = NULL

ticcompl$PWALAND = NULL

ticcompl$PBESAUT = NULL

ticcompl$PMOTSCO = NULL

ticcompl$PVRAAUT = NULL

ticcompl$PAANHANG = NULL

ticcompl$PTRACTOR = NULL

ticcompl$PWERKT = NULL

ticcompl$PBROM = NULL

ticcompl$PLEVEN = NULL

ticcompl$PPERSONG = NULL

ticcompl$PGEZONG = NULL

ticcompl$PWAOREG = NULL

ticcompl$PZEILPL = NULL

ticcompl$PPLEZIER = NULL

ticcompl$PFIETS = NULL

ticcompl$PINBOED = NULL

ticcompl$PBYSTAND = NULL

ticcompl$AWABEDR = NULL

ticcompl$AWALAND = NULL

ticcompl$ABESAUT = NULL

ticcompl$AMOTSCO = NULL

ticcompl$AVRAAUT = NULL

ticcompl$AAANHANG = NULL

ticcompl$ATRACTOR = NULL

ticcompl$AWERKT = NULL

ticcompl$ALEVEN = NULL

ticcompl$APERSONG = NULL

ticcompl$AGEZONG = NULL

ticcompl$AWAOREG = NULL

ticcompl$AZEILPL = NULL

ticcompl$APLEZIER = NULL

ticcompl$AFIETS = NULL

ticcompl$AINBOED = NULL

ticcompl$ABYSTAND = NULL

* Following attributes are removed as they add no information to the dataset. Majority of the data is categorized with one value.

ticcompl$MAANTHUI=NULL

ticcompl$MINK123M=NULL

ticcompl$ABROM=NULL

* Following conversions are done as the values comprise of very less number of records.

ticcompl$MAUT2 <- ifelse(ticcompl$MAUT2>4,4, ticcompl$MAUT2)

ticcompl$PWAPART <- ifelse(ticcompl$PWAPART==3,1, ticcompl$PWAPART)

ticcompl$ABRAND <- ifelse(ticcompl$ABRAND>2,2, ticcompl$ABRAND)

ticcompl$APERSAUT <- ifelse(ticcompl$APERSAUT>2,2, ticcompl$APERSAUT)

ticcompl$AWAPART <- ifelse(ticcompl$AWAPART==2,1, ticcompl$AWAPART)

ticcompl$PPERSAUT <- ifelse(ticcompl$PPERSAUT>6,7, ticcompl$PPERSAUT)

ticcompl$MINK7512 <- ifelse(ticcompl$MINK7512>4,4, ticcompl$MINK7512)

ticcompl$MSKD <- ifelse(ticcompl$MSKD>5,5, ticcompl$MSKD)

ticcompl$MSKB1 <- ifelse(ticcompl$MSKB1>5,5, ticcompl$MSKB1)

ticcompl$MBERZELF <- ifelse(ticcompl$MBERZELF>2,5, ticcompl$MBERZELF)

ticcompl$MBERBOER <- ifelse(ticcompl$MBERBOER>5,5, ticcompl$MBERBOER)

ticcompl$MRELSA <- ifelse(ticcompl$MRELSA>4,4, ticcompl$MRELSA)

ticcompl$MGODOV <- ifelse(ticcompl$MGODOV==5,4, ticcompl$MGODOV)

ticcompl$MGODRK <- ifelse(ticcompl$MGODRK>5,5, ticcompl$MGODRK)

ticcompl$MGEMLEEF <- ifelse(ticcompl$MGEMLEEF==6,1, ticcompl$MGEMLEEF)

ticcompl$MGEMOMV <- ifelse(ticcompl$MGEMOMV>5,1, ticcompl$MGEMOMV)

1. **Building the model:**

Now the data is ready for modelling. We create classification models on this processed data.

**Since the predictor variable value is completely skewed, prediction is quite difficult using SVM or random Forest**

1. **Using SVM**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Classifier** | **Parameter1 Cost** | **Parameter2 kernel** | **Parameter3 gamma** | **Parameter4 tolerance** | **Accuracy for 1** | **Accuracy for 0** | **Overall Accuracy** |
| SVM | 1 | linear | - | - | 100 | 0 | 94.13 |
| SVM | 10 | Polynomial | 0.25 | - | 95.77 | 8.47 | 90.65 |
| SVM | 20 | Polynomial | 0.75 | - | 95.77 | 8.47 | 90.65 |
| **SVM** | **20** | **Polynomial** | **0.5** | **-** | **95.77** | **8.47** | **90.65** |
| SVM | 10 | Polynomial | 0.5 | 0.05 | 95.77 | 8.47 | 90.65 |
| SVM | 10 | Radial | 0.5 | 0.05 | 97.88 | 3.39 | 92.34 |
| SVM | 15 | Radial | 0.33 | 0.02 | 97.15 | 6.78 | 91.84 |
| SVM | 25 | Radial | 0.75 | 0.05 | 98.20 | 3.39 | 92.64 |
| SVM | 10 | Radial | 0.2 | 0.01 | 96.72 | 6.78 | 91.44 |
| SVM | 12 | Polynomial | 0.3 | 0.02 | 95.67 | 8.47 | 90.55 |
| SVM | 5 | Radial | 0.1 | 0.01 | 96.19 | 6.78 | 90.95 |

* Commands used to build a model on train and predict on test using SVM:

svm.lin = svm(CARAVAN~., data=train, kernel = "radial", cost=5, gamma= 0.1,tolerance = 0.01)

svm.lin.pred = predict(svm.lin,test)

* The following table shows the prediction accuracy for various parameter values (kernel, gamma, cost, tolerance)
* Even though for **linear kernel,** the overall accuracy is 95.41 but the accuracy for predicting 0 is almost zero. Hence the model for this kernel doesn’t seem feasible.
* Among the various parameter values used, **polynomial kernel** with gamma of 0.5 and cost of 20 gives best possible prediction outcome for both predictor variable values with overall accuracy of 90.65. With the accuracy of 95.77 for 1 & 8.47 for 0, it is best among the values tested.

This accuracy is higher among others. So, it is optimal to consider this set for modelling this data.

1. **Using Random Forest**

* Commands used to build a model on train and predict on test using randomForest:

rf.model = randomForest(CARAVAN~., data= train,mtry=35,ntree = 1000, importance=TRUE)

rf.pred = predict(rf.model,test,type="class")

The following table shows the prediction accuracy for various parameter values (ntrees, Importance, mtry)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Classifier** | **ntree** | **Importance** | **mtry** | **Accuracy for 1** | **Accuracy for 0** | **Overall Accuracy** |
| RF | - | TRUE | 3 | 99.79 | 3.39 | 99.6 |
| **RF** | **500** | **TRUE** | **3** | **99.79** | **3.39** | **99.6** |
| RF | 300 | TRUE | 23 | 100 | 0 | 94.13 |
| RF | 400 | TRUE | 12 | 100 | 0 | 94.13 |
| RF | 1000 | TRUE | 35 | 100 | 0 | 94.13 |

* Among the various parameter values used, classifierwith ntree value of 500 and mtry of 3 gives best possible prediction outcome for both predictor variable values with overall accuracy of 99.6 for RF.
* We can observe increasing the ntree & mtry values, decreases the prediction accuracy for 0.

This accuracy isn’t feasible as accuracy for 0 becomes almost zero. So, it is not optimal to consider this set for modelling this data.

**Summary:**

* We can clearly see that because of skewed data, prediction of output is very uneven. Even though we have a good result using randomForest, the accuracy in predicting **0** is almost zero.
* Out of the randomForest and SVM that we created in predicting the output, SVM gives the best possible results for both predictor values.
* We conclude that **SVM** with **polynomial kernel** is the best classification modelling algorithm for the data of Home Insurance Policy.