Group 8

All State Purchase Prediction

Aji Somaraj

Keerthana Ashok Kumar

Neeraja Anil

Ramkumar Sreeram

Sai Kiran Vamaraju

AGENDA

- Objective
- Dataset Description
- Exploratory & Descriptive Analysis
- Modeling Techniques
- Model Evaluation
- Model Based on Data Pattern
- Conclusion

OBJECTIVE

- Our main objective is to predict whether a customer will opt for a particular insurance policy or not
- Customers receive a number of quotes with different coverage options
- Hence the need to analyze the factors that influence the customers purchase pattern
- If the purchase can be predicted sooner in the shopping window, the quoting process is shortened and becomes more cost and time effective

DATASET DESCRIPTION

- The dataset had **25 features** in total, representing customer demographics and other details.

Details	
Attributes	**25**
Instances	**665249**
Data Set Characteristics	**Multivariate**

Variable	Description
customer_ID	A unique identifier for the customer
shopping_pt	Unique identifier for the shopping point of a given customer
record_type	0=shopping point, 1=purchase point
day	Day of the week (0-6, 0=Monday)
time	Time of day (HH:MM)
state	State where shopping point occurred
location	Location ID where shopping point occurred
group_size	How many people will be covered under the policy (1, 2, 3 or 4)
homeowner	Whether the customer owns a home or not (0=no, 1=yes)
car_age	Age of the customer's car
car_value	How valuable was the customer's car when new
risk_factor	An ordinal assessment of how risky the customer is (1, 2, 3, 4)
age_oldest	Age of the oldest person in customer's group
age_youngest	Age of the youngest person in customer's group
married_couple	Does the customer group contain a married couple (0=no, 1=yes)
C_previous	What the customer formerly had or currently has for product option C (0=nothing, 1, 2, 3,4)
duration_previous	How long (in years) the customer was covered by their previous issuer
A, B, C, D, E, F, G	The coverage options
Cost	Cost of the quoted coverage options

We are analyzing the data to build a model that will predict the coverage option B. B can have 2 options – 0 or 1

EXPLORATORY ANALYSIS

- Overview of the dataset

```
> insurance<-read.csv("insurance.csv",header=TRUE)
> str(insurance)
'data.frame':	665249 obs. of  25 variables:
 $ customer_ID      : int  10000000 10000000 10000000 10000000 10000000 10000000 10000000 10000000 10000000 10000005 ...
 $ shopping_pt      : int  1 2 3 4 5 6 7 8 9 1 ...
 $ record_type      : int  0 0 0 0 0 0 0 0 1 0 ...
 $ day              : int  0 0 0 0 0 0 0 0 0 3 ...
 $ time             : Factor w/ 1204 levels "0:01","0:09",..: 1120 1123 1123 1124 143 145 146 151 155 1141 ...
 $ state            : Factor w/ 36 levels "AL","AR","CO",..: 11 11 11 11 11 11 11 11 11 24 ...
 $ location         : int  10001 10001 10001 10001 10001 10001 10001 10001 10001 10006 ...
 $ group_size       : int  2 2 2 2 2 2 2 2 2 1 ...
 $ homeowner        : int  0 0 0 0 0 0 0 0 0 0 ...
 $ car_age          : int  2 2 2 2 2 2 2 2 2 10 ...
 $ car_value        : Factor w/ 10 levels "","a","b","c",..: 8 8 8 8 8 8 8 8 8 6 ...
 $ risk_factor      : int  3 3 3 3 3 3 3 3 3 4 ...
 $ age_oldest       : int  46 46 46 46 46 46 46 46 46 28 ...
 $ age_youngest     : int  42 42 42 42 42 42 42 42 42 28 ...
 $ married_couple   : int  1 1 1 1 1 1 1 1 1 0 ...
 $ C_previous       : int  1 1 1 1 1 1 1 1 1 3 ...
 $ duration_previous: int  2 2 2 2 2 2 2 2 2 13 ...
 $ A                : int  1 1 1 1 1 1 1 1 1 1 ...
 $ B                : int  0 0 0 0 0 0 0 0 0 1 ...
 $ C                : int  2 2 2 2 2 2 2 2 2 3 ...
 $ D                : int  2 2 2 2 2 2 2 2 2 3 ...
 $ E                : int  1 1 1 1 1 1 1 1 1 1 ...
 $ F                : int  2 2 2 2 2 2 2 2 2 0 ...
 $ G                : int  2 1 1 1 1 1 1 1 1 2 ...
 $ cost             : int  633 630 630 630 630 638 638 638 634 755 ...
```

HANDLING MISSING VALUES

- Risk factor, C-previous and duration-previous had missing values in the dataset
- C-previous and duration-previous NA's were replaced with "0" assuming that the customer is currently not under coverage option "C".
- To impute the missing values of risk factor we built a regression model using those variables that has high correlations to the risk factor

```
#Risk_factor_NA
datanorisk <- insurance[is.na(insurance$risk_factor), ]
datarisk <- insurance[!is.na(insurance$risk_factor), ]
lm.fit <- lm(risk_factor ~ age_youngest*group_size+married_couple+
             homeowner, data=datarisk)
lm.pred <- predict(lm.fit, newdata=datanorisk)
data$risk_factor[is.na(data$risk_factor)] <- round(lm.pred, 0)
```

DESCRIPTIVE ANALYSIS

Customers – 97009 unique customers / 665249 rows

```
> length(unique(insurance$customer_ID))
[1] 97009
```

Record type

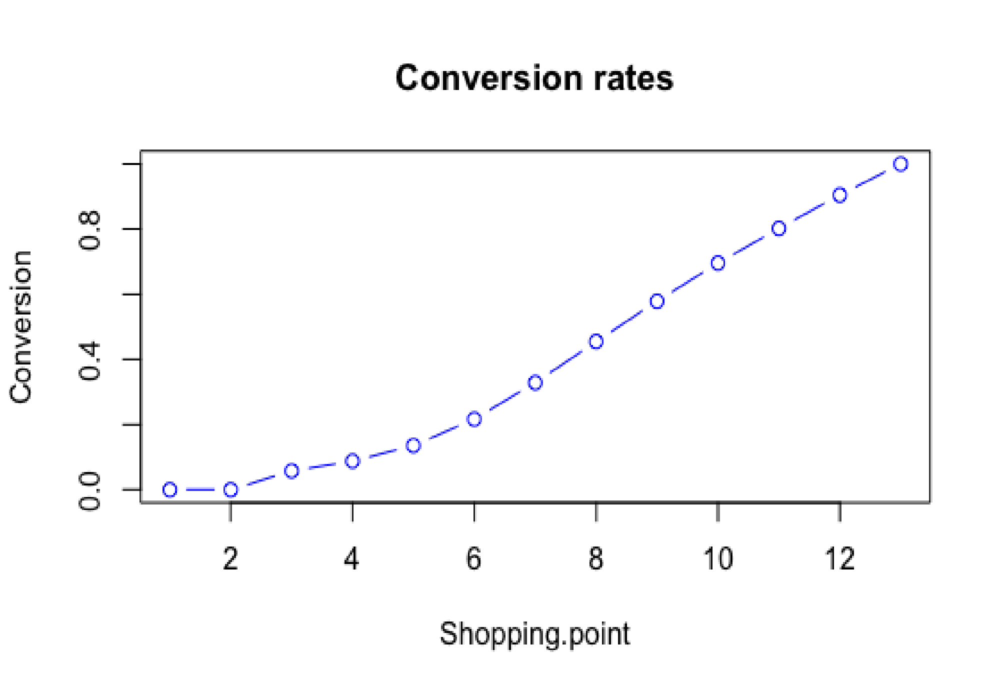
0(Shopping point)

1(Purchase point)

```
> table(insurance$record_type)

     0      1
568240  97009
```

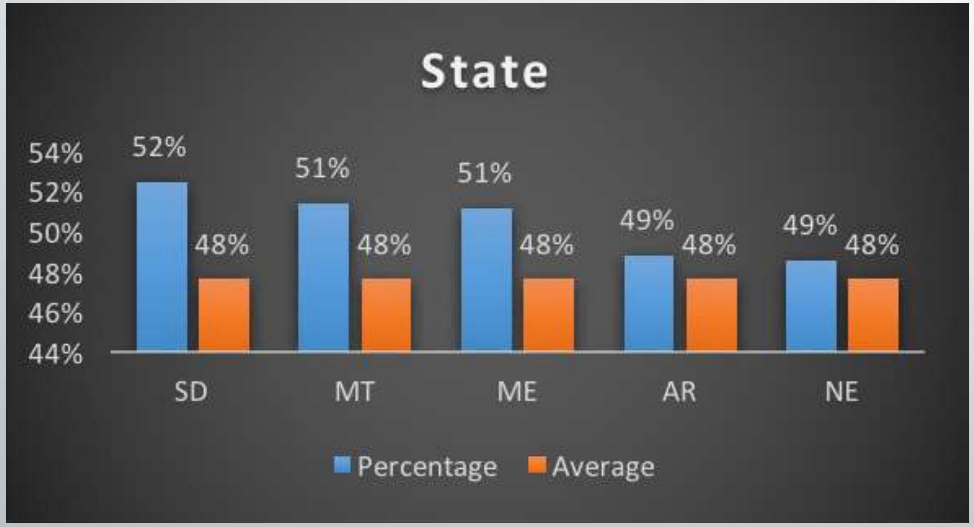
Shopping point – 1 to 13

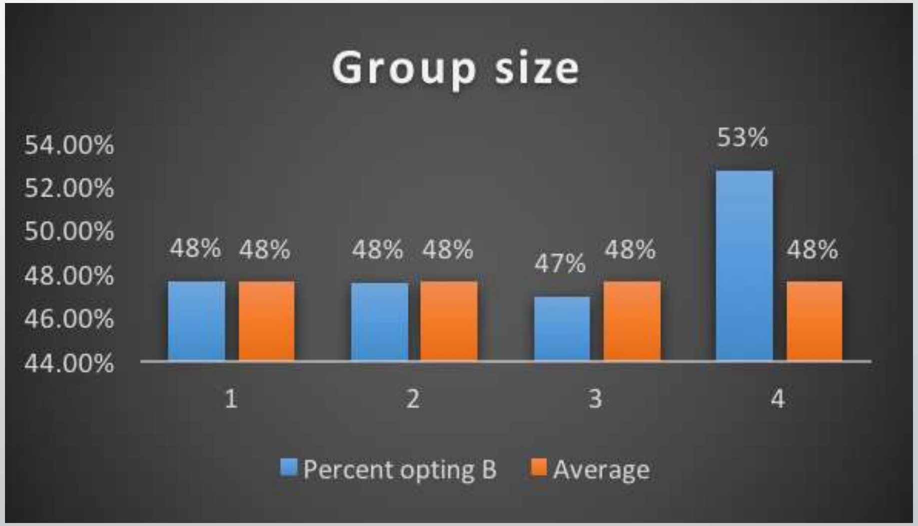
```
> table(insurance$shopping_pt)

    1     2     3     4     5     6     7     8
97009 97009 97009 91441 83440 72171 56548 37958
    9    10    11    12    13
20710  8725  2654   525    50
```



FACTORS AFFECTING COVERAGE POLICY "B"

- Some of the characteristics of people who opt B policy belongs to
- State : SD,MT,ME,AR,NE
- Group-size: People with group size 4 tend to opt for policy B 53% of the time
- Car-age: People with older cars opt for the insurance coverage B





BASELINE MODEL

- For coverage B, it can be seen that more number of customers opt for B=0 over B=1. A baseline model that always predicts that customer will choose B=0 is taken

- This model gives an accuracy of 54.59% just by pure guessing. For other modelling to be effective, the accuracy of prediction should surpass the baseline accuracy.

```
> confusionMatrix(baselinemodel$B,train$B)
Confusion Matrix and Statistics

          Reference
Prediction      0      1
         0 349943 291054
         1      0      0

               Accuracy : 0.5459
                 95% CI : (0.5447, 0.5472)
    No Information Rate : 0.5459
    P-Value [Acc > NIR] : 0.5005

                  Kappa : 0
 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.5459
         Neg Pred Value :    NaN
             Prevalence : 0.5459
         Detection Rate : 0.5459
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 0
```

Modelling Techniques
Logistic Model

- Taking into the factors that influence the purchase decision of the customer, we build models to do the prediction
- The dataset is initially split into training(75%) and test(25%) datasets
- A logistic model is built on the training dataset and it is validated using the test dataset
- We take the state, car-age and risk factors as these have greater influence on the purchase decisions of the customer

```
> LogModel1 <- glm(B ~ state+cost+age_oldest+age_youngest+
+       +car_age+risk_factor+
+       duration_previous, data=insuranceTrain, family=binomial)
```

Logistic Regression Evaluation with Training Dataset

```
> ### confusion matrix for training dataset
> logpredtrain = predict(LogModel1,type = "response")
> table(insuranceTrain$B,logpredtrain >= 0.5)### Assuming our threshold value is 0.5

    FALSE  TRUE
  0 26248 11870
  1 16992 17647
> ### Accuracy
> (26248+17647)/nrow(insuranceTrain)
[1] 0.6033096
```

Accuracy of the model =60.33%

Logistic Regression Evaluation with Test Dataset

```
> ### confusion matrix for testing dataset
> logpredtest = predict(LogModel1,newdata = insuranceTest,type = "response")
> table(insuranceTest$B,logpredtest >= 0.5)### Assuming our threshold value is 0.5

    FALSE TRUE
  0  8696 4010
  1  5677 5869
> ### Accuracy
> (8696+5869)/nrow(insuranceTest)
[1] 0.600569
> |
```

Accuracy of the model =60.06%

Decision tree model

- Decision Tree model is built on the Training data.
- It is evaluated with training and test dataset to compare the performance of model and accuracy.
- In a classification tree the predicted value is one of the possible levels of the response variable. i.e., either 0 or 1 for target variable B.

```
Variables actually used in tree construction:
[1] age_oldest        car_age           cost              duration_previous
[5] risk_factor       shopping_pt       state

Root node error: 291054/640997 = 0.45406

n= 640997

         CP nsplit rel error  xerror      xstd
1 0.0890316      0   1.00000 1.00000 0.0013696
2 0.0102730      1   0.91097 0.91106 0.0013547
3 0.0032434      2   0.90070 0.90075 0.0013524
4 0.0025860      4   0.89421 0.89648 0.0013514
5 0.0025851      7   0.88645 0.88104 0.0013476
6 0.0016200     17   0.85862 0.86158 0.0013424
7 0.0015908     19   0.85538 0.86059 0.0013422
8 0.0015000     20   0.85379 0.85988 0.0013420
```

- The tree has 21 leaves.
- State is the first variable used to split, and it contains 504,494 observations

Decision Tree Evaluation with Training Dataset

```
Error matrix for the Decision Tree model on train.csv [**train**] (counts):

      Predicted
Actual      0      1
     0 258607  91336
     1 157524 133530

Error matrix for the Decision Tree model on train.csv [**train**] (proportions):

      Predicted
Actual    0    1 Error
     0 0.40 0.14  0.26
     1 0.25 0.21  0.54

Overall error  39%, Averaged class error: 40%
```

Decision Tree Evaluation has an overall error 39%
Accuracy for Training Dataset is 61%

Decision Tree Evaluation with Test Dataset

```
Error matrix for the Decision Tree model on test.csv [**train**] (counts):

      Predicted
Actual    0    1
     0 9853 3273
     1 5669 5457

Error matrix for the Decision Tree model on test.csv [**train**] (proportions):

      Predicted
Actual    0    1 Error
     0 0.41 0.13  0.25
     1 0.23 0.23  0.51

Overall error  37%, Averaged class error: 38%
```

Decision Tree Evaluation has an overall error 37%
Accuracy for Training Dataset is 63%

Random Forest to predict coverage option B

- Random forest builds hundreds of decision tress and select those ones with best accuracy.

- Number of Trees=100
  Number of Variables=3

```
Summary of the Random Forest Model
==================================

Number of observations used to build the model: 640997
Missing value imputation is active.

Call:
 randomForest(formula = as.factor(B) ~ .,
              data = crs$dataset[, c(crs$input, crs$target)],
              ntree = 100, mtry = 3, importance = TRUE, replace = FALSE, na.action = randomForest::na.roughfix)

               Type of random forest: classification
                     Number of trees: 100
No. of variables tried at each split: 3

        OOB estimate of  error rate: 14.68%
Confusion matrix:
       0      1 class.error
0 314808  35135   0.1004021
1  58991 232063   0.2026806
```

Random Forest Evaluation with Training Dataset

```
        OOB estimate of  error rate: 14.68%
Confusion matrix:
       0      1 class.error
0 314808  35135   0.1004021
1  58991 232063   0.2026806
```

Overall error=14.68%

Accuracy=85.32%

Random Forest Evaluation with Test Dataset

```
        OOB estimate of  error rate: 24.34%
Confusion matrix:
      0    1 class.error
0 10769 2357   0.1795673
1  3547 7579   0.3188028
```

Overall error=24.34%

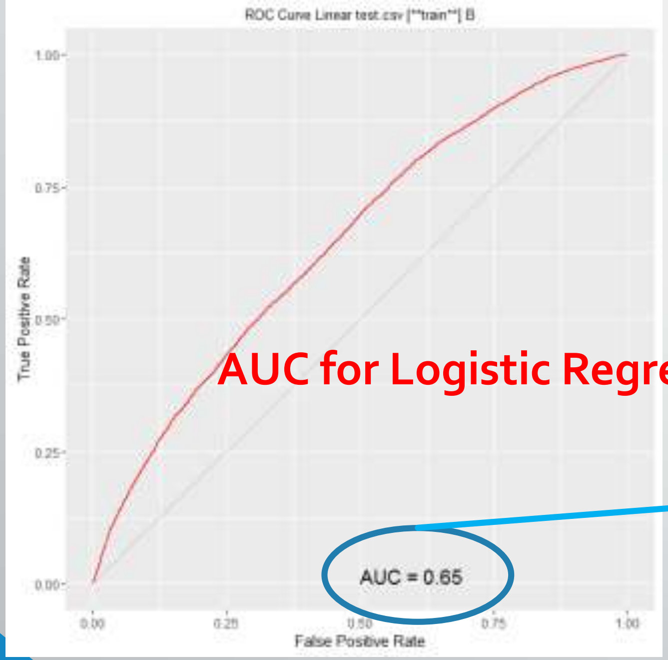
Accuracy=75.66%

MODEL EVALUATION

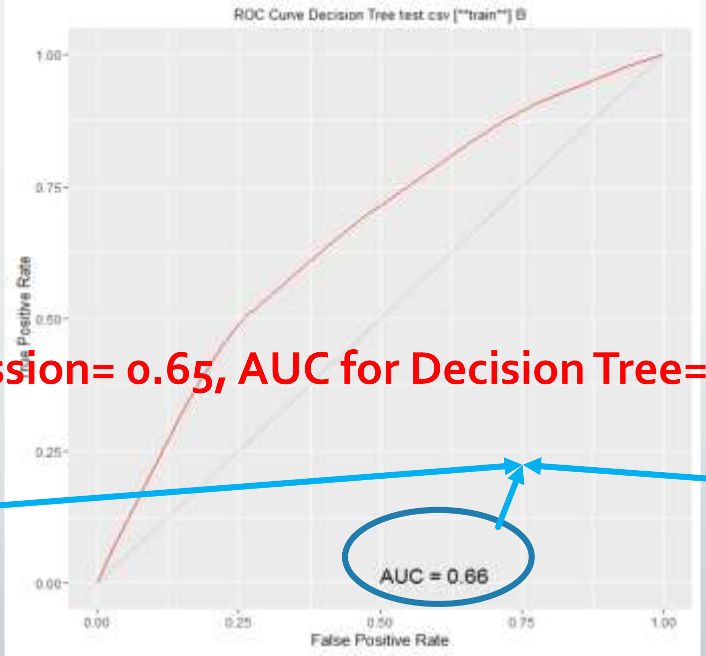
ROC curves for test data

An Receiver Operating Characteristics (ROC) chart plots the true positive rate against the false positive rate. The ROC has a form and interpretation similar to the risk chart, though it plots different measures on the axes.

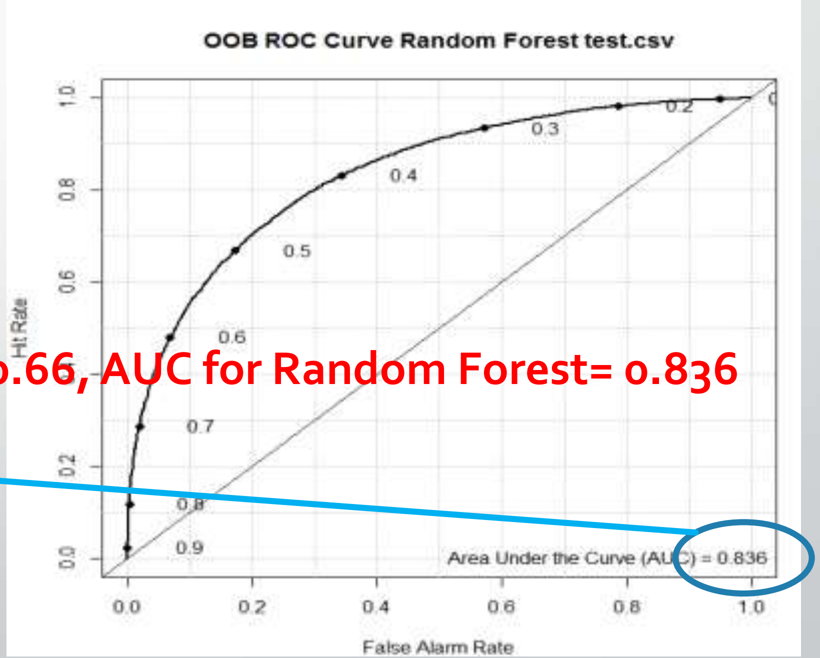
Logistic Regression Evaluation



Decision Tree Evaluation



Random Forest Evaluation



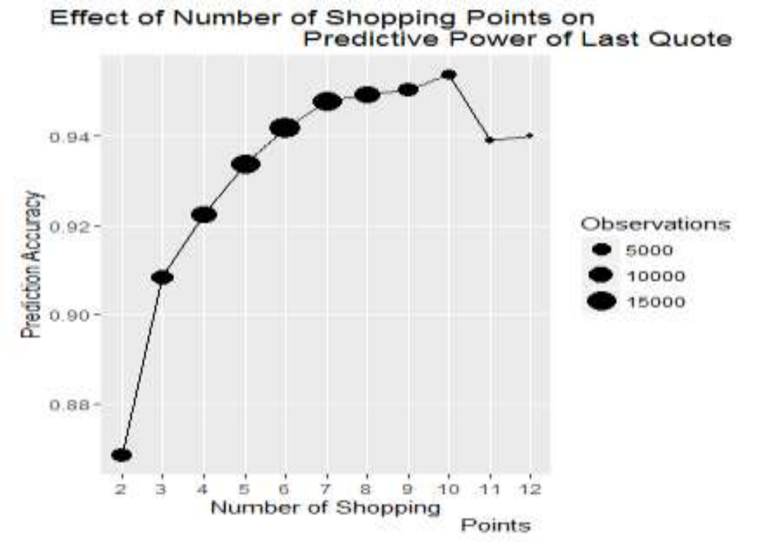
AUC for Logistic Regression= 0.65, AUC for Decision Tree= 0.66, AUC for Random Forest= 0.836

Area under the curve is greatest for Random Forest out of the 3 models, hence it seems to be the best model

MODEL BASED ON DATA PATTERN

Majority of customers purchase the last quote viewed by them.

```
> # changed B from their last quote
> changedB <- ifelse(purchaseddata$B == lastquotedata$B, "No", "Yes")
> table(changedB)
changedB
   No   Yes
90617  6392
> purchaseddata$changedB <- as.factor(changedB)
> lastquotedata$changedB <- as.factor(changedB)
```



The final quote correctly predicted the purchased options 50% to 75% of the time, with that percentage steadily increasing as customers review more quotes.

Model Evaluation with Dataset

```
> confusionMatrix(naivemodel$B,purchaseddata$B)
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 48053  3621
         1  2771 42564

               Accuracy : 0.9341
                 95% CI : (0.9325, 0.9357)
    No Information Rate : 0.5239
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8678
 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9455
            Specificity : 0.9216
         Pos Pred Value : 0.9299
         Neg Pred Value : 0.9389
             Prevalence : 0.5239
         Detection Rate : 0.4953
   Detection Prevalence : 0.5327
      Balanced Accuracy : 0.9335

       'Positive' Class : 0
```

Accuracy for the model=93.41%

CONCLUSION

Key Learnings:

- Customer demographics seem to have little effect on selection of purchased policy.
- The number of quotes received by a customer play a role in selection of purchased policy. Higher the number of quotes, more is the chances of selecting the last quoted policy.
- The last quoted policy has a strong impact on the outcome of the results. More than half of the time, it predicts the purchased policy correctly.

Results:

- Found a pattern in the dataset (I.e.) the last quote viewed by the customer is almost always purchased.
- Used the pattern to build the model. It had the highest predicting power over any other model.

Baseline Model		Logistic Regression		Decision Tree		Random Forest		Model based on Data Pattern	
Train Data	Test Data	Train Data	Test Data	Train Data	Test Data	Train Data	Test Data	Train Data	Test Data
54.59	---	60.33	60.06	61	63	85.32	75.66	93.41	---

Business implementation:

Business can gain an insight into which option a customer is likely to end up choosing.
They could nudge the customer toward that product (to increase their conversion rate), or towards a slightly more expensive product (in order to maximize their profit from that sale).