**Predicting Santander Customer Transaction**

Priyanka Chettri

28-Jul-2019

Contents

1. **Introduction**
   1. Problem Statement 3
   2. Data 3
2. **Methodology**
   1. Exploratory Data Analysis 4
   2. Distribution of dependent variables 4
   3. Distribution of independent variables 5
   4. Collinearity check between the variables 7
   5. Sampling from the dataset 9
3. **Modelling**
   1. Model Selection 10
   2. Logistic Regression 11
   3. Decision Tree 14
   4. Naïve Bayes 16
4. **Conclusion** 19

**Chapter 1**

**Introduction**

* 1. **Problem Statement**

The aim of this project is to predict whether the customer of Santander bank will make a particular transaction in the future or not, irrespective of the money they spend on transactions. Santander is a wholly owned subsidiary of Spanish Santander group. It is based in Boston and its principal market is in Northeastern United States. It operates about 650 retail banking offices and over 2000 ATMs and employs approximately 9800 people. For such a big and prestigious organisation it is imperative that they provide the best services to their customers. This would help them with their continued growth.

Thus, this project aims to help decide the customer transaction which in turn will help the bank focus their time and resources on the customers with highest transaction probabilities.

* 1. **Data**

Our goal is to build a classification model that will help in predicting whether the customer will carry out a specific transaction or not. The data is considerably large, with 200 predictors and 200000 data points. It has not been specified what the predictor names are, rather they are named var\_0 till var\_200. The dependent variable is named as the target and has two distinct values of 0 signifying ‘No’ and 1 signifying ‘Yes’ for the transactions not made and made respectively.

Since the importance of variables has not been signified or quantified. The values of all variables lie close to each other, so we will use all 200 variables to build the model.

**Chapter 2: Methodology**

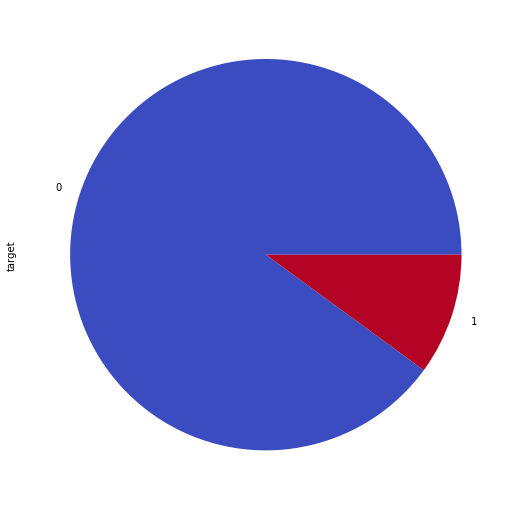
**2.1 Exploratory Data Analysis**

The first and foremost activity that is carried out before any data modelling is exploratory data analysis (EDA). EDA refers to the critical process of performing initial investigation on data so as to discover patterns, spot anomalies, test hypotheses, and check assumptions with the help of summary statistics and graphical representation.

**2.2 Distribution of dependent variable**

Since our problem is a case of classification, it is only fair we check how our target variable is distributed. There are only 2 values for the target, i.e., 0 and 1. The outcome we are interested is in 1, because it denotes a positive scenario where the customer made a specific transaction. As can be seen from the figure the output is highly uneven, with only about 20% of the data contributing to the desired outcome. From the pie chart we can clearly tell that there are a lot of instances of the negative case rather than the desired one. This information will be very helpful in case we decide to take samples from the data for further use.

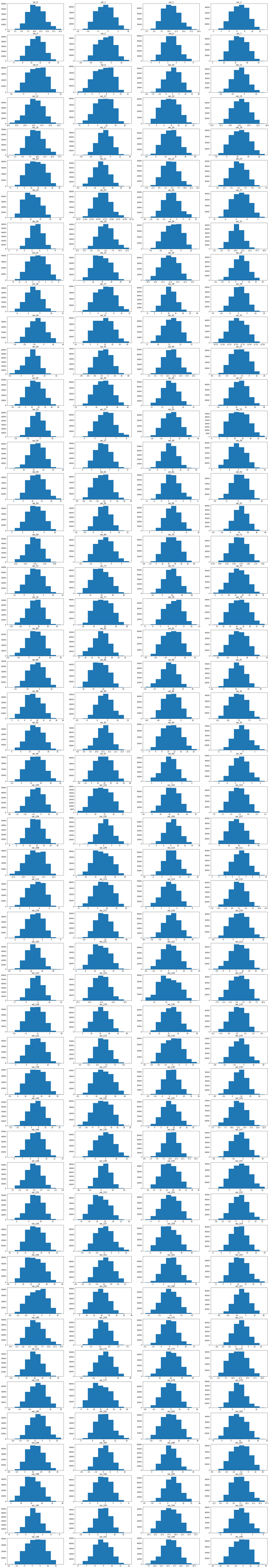
Fig 2.1: Distribution of target variable using a pie chart



**2.3 Distribution of independent variables**

There is a total of 200 independent variables named var\_0 through var\_200. We will check how each of them is distributed using a histogram.

Fig 2.2 Distribution of independent variables using histogram



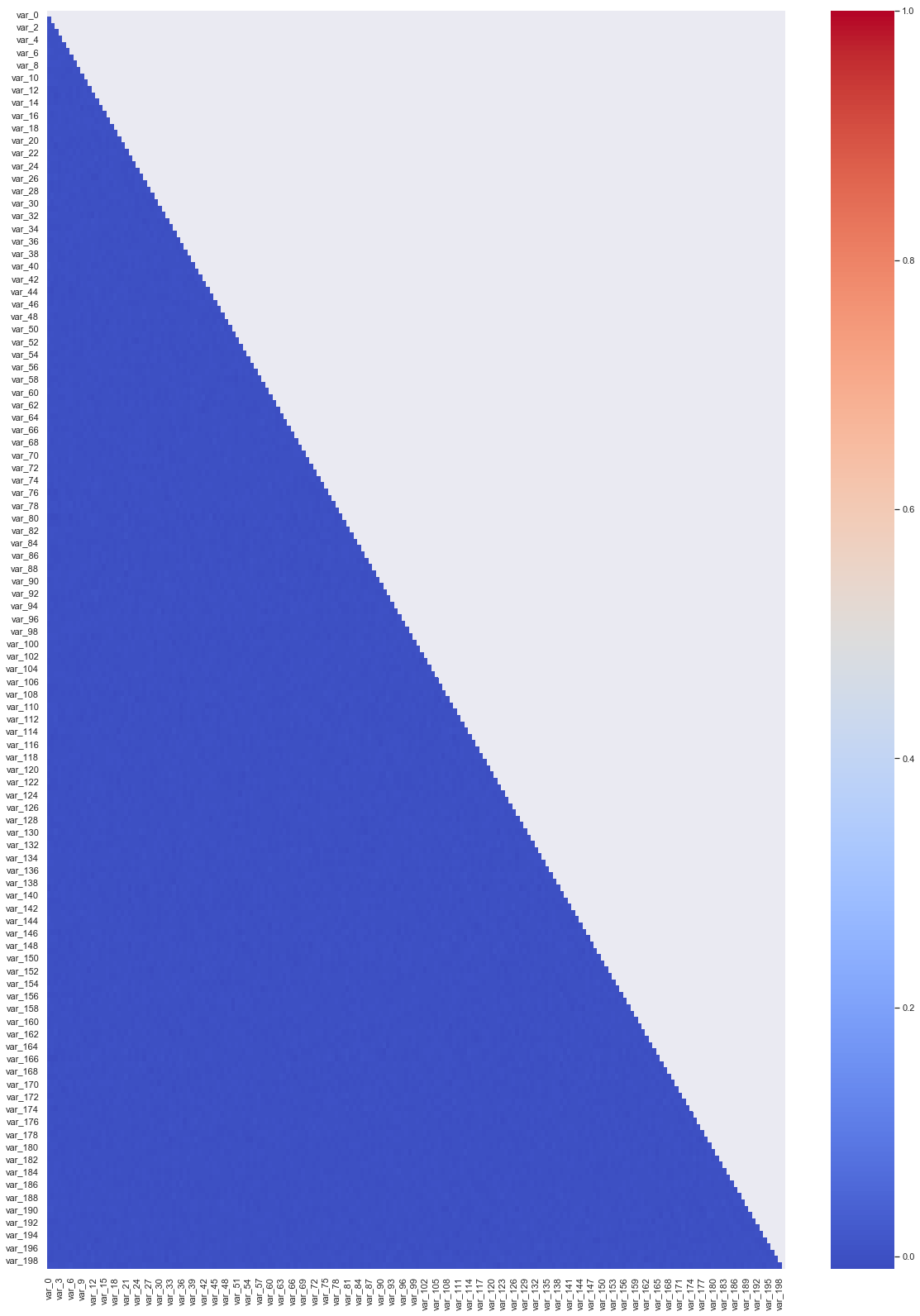
As we can see from the figure above most of the data is normally distributed. It is very important for the data to be normally distributed because then 68% of the data would be explained by 1 standard deviation, while 95% of data can be explained by 2 standard deviations. So, we would be running our models to predict for a majority of the population. If there were skewness it would suggest the presence of outliers which would have to be removed. However, since the data is normally distributed there are no outliers.

**2.4 Collinearity check between the variables**

After checking the datatypes of the variables and their distribution plots we will now check if there’s any correlation between the variables. Correlation signifies the relationship among the predictor variables. Correlation ranges from -1 and +1, -1 signifying high negative correlation, +1 signifying high positive correlation while 0 signifying no correlation at all. It is highly desirable to have variables with no correlation, because each variable would then be contributing to a different set of information. Variables with high negative correlation are also desirable, meaning one variable would be inversely related to the other. But in case of high positive correlation, the variable would simply be giving the information, and the combined effect of both may affect the result in a negative way. Therefore, it is very important that we remove of the highly positively correlated variables.

As we can see from the figure below, all the variables are independent of each other, meaning they have no correlation with one another. In this case we can safely assume that each variable makes a significant contribution in predicting the result.

Fig 2.3 Collinearity plot of the independent variables



**2.5 Sampling from the Dataset**

The dataset is considerably huge with 200000 records and 200 variables, running a model on all these data especially, random forest, KNN etc would require a lot of processing power and may end up taking a lot of time. One way of dealing with this problem is by working with the samples. Data sampling is a statistical analysis technique used to select, manipulate and analyse a representative subset of data points to identify patterns and trends in the larger dataset being examined. There are multiple sampling techniques like simple random sampling, stratified sampling, cluster sampling, multistage sampling, systematic sampling etc. For the purpose of this project Stratified Sampling is used.

This technique divides the elements of the population into small subgroups (strata) based on the similarity in such a way that the elements within the group are homogeneous and heterogeneous among the other subgroups formed. And then the elements are randomly selected from each of these strata. We need to have prior information about the population to create subgroups.

For this case the sampling was done based on the ‘target’ variable. Since there are two values 0 and 1, two strata were created. Since the distribution of the 0 and 1 values of target variable is 4:1, therefore, the samples also had same ratio. The samples taken were 10% of the total population which is equal to 20000 out of which roughly 2000 records had target value 1, while 18000 of them had target value of 0.

The code below was used to get the stratified sample from the dataset.

stratas = sampling:::strata(train, stratanames = 'target', size = c(17990, 2010), method = 'srswor')

table(stratas)

stratified\_data = getdata(train,stratas)

**Chapter 3: Modelling**

**3.1: Model Selection**

The dependent variable, i.e., ‘target’ is a categorical variable, with values as 0 or 1. Therefore, the models we use must tend to classification problems. I chose Logistic Regression, Decision Tree and Naïve Bayes algorithms to build a model to fit the data and come to accurate conclusions.

The error metrics used was Confusion matrix and AUC (Area under the curve) or ROC (Receiver Operating Characteristics).

Confusion matrix is a performance measurement for machine learning classification problem where output can be two or more classes. It is a table with 4 different combinations of predicted and actual values.

Fig 3.1 Confusion Matrix



AUC - ROC curve is a performance measurement for classification problem at various thresholds settings. ROC is a probability curve and AUC represents degree or measure of separability. It tells how much model is capable of distinguishing between classes.

Fig 3.2 AUC – ROC curve



**3.1: Logistic Regression**

Logistic regression is used for classification problems with binary outputs. Since our output also has two possible outcomes, logistic regression would fit the data properly. Logistic regression is based on the sigmoid function. If all the input variables result in a value higher than the threshold value the output will be considered 1, while if all the input variables result in a value lower than the threshold value the result will be considered as 0.

Our model also works on the same principle.

Fig 3.3 Logistic Regression sigmoid function



Confusion Matrix: Based on the figures from the confusion matrix in the figure below, various model predictors were calculated as follows.

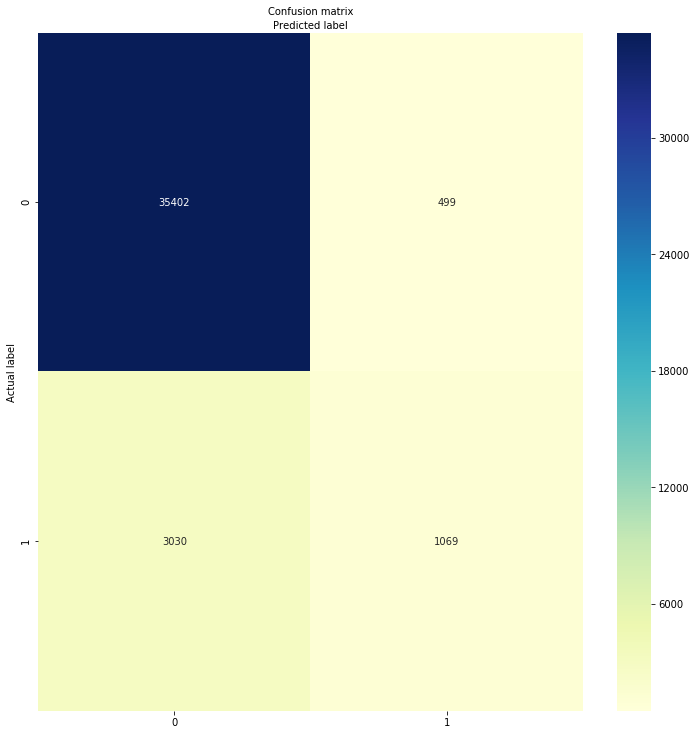
Accuracy: 0.911775

Precision: 0.6817602040816326

Recall: 0.2607953159307148

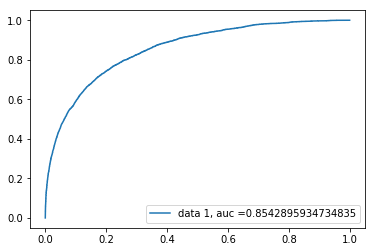
The accuracy of the model is 91.1% which is very good, however the precision is 68% meaning that the model predicted correct for 68% of the predictions it made.

Fig 3.2 Confusion Matrix for logistic regression



ROC -AUC curve: As can be seen from the figure below the AUC is 85.4, which is a pretty good number. We can say that our model is performing well.

Fig 3.3 AUC for logistic regression model



**3.1: Decision Tree**

Decision tree as the name suggests is a tree – structured algorithm based on the decision made at every node. The leaf node is reached when no further branching is possible. We will use all of the 200 predictor variables to come at the decision of whether the customer will perform the transaction or not.

Confusion Matrix: Based on the figures from the confusion matrix in the figure below, various model predictors were calculated as follows.

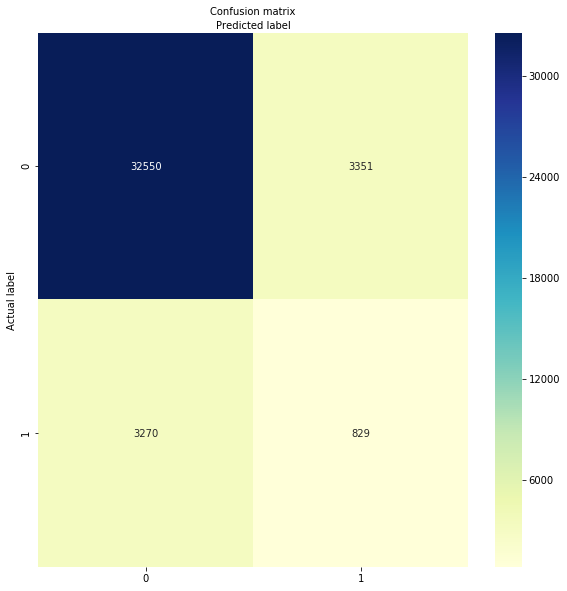
Accuracy: 0.834475

Precision: 0.19832535885167465

Recall: 0.20224444986582094

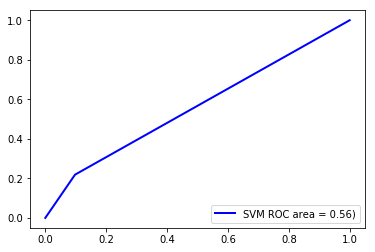
The accuracy is pretty good at 83.4%. However, if we have a look at the precision and recall values, both of them are extremely low indicating that the model is not up to the mark. The model has probably over fitted the data. This can be expected due to the large number of predictor variables. It is advisable not to use decision tree for such problems.

Fig 3.4 Confusion Matrix for decision tree model



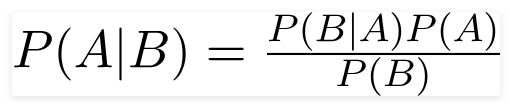
ROC – AUC curve: As can be seen from the figure below the AUC is 56, which is not a good number. We can say that our model is under performing and decision tree is not fit for this type of classification problem.

Fig 3.5 ROC – AUC for decision tree model



**3.1: Naive Bayes Model**

A Naive Bayes classifier is a probabilistic machine learning model that’s used for classification task. The crux of the classifier is based on the Bayes theorem.



Using Bayes theorem, we can find the probability of **A** happening, given that **B** has occurred. Here, **B** is the evidence and **A** is the hypothesis. The assumption made here is that the predictors/features are independent. That is presence of one particular feature does not affect the other. Hence it is called naive.

Confusion Matrix: Based on the figures from the confusion matrix in the figure below, various model predictors were calculated as follows.

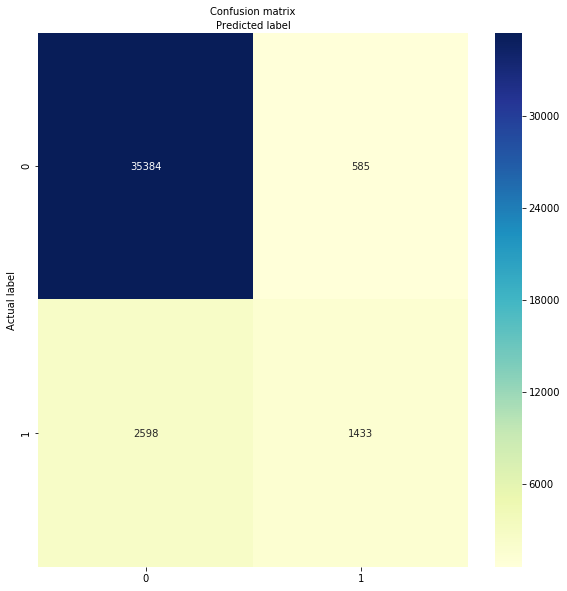
Accuracy: 0.920425

Precision: 0.7101090188305252

Recall: 0.35549491441329695

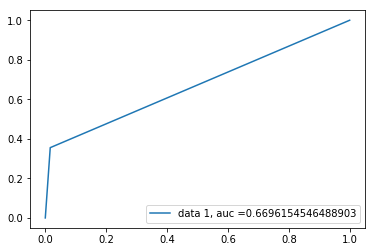
The accuracy is very good at 92%, so is the precision at 71%. Recall value is lower but however, we desire a higher precision value rather than recall. So based on that the model is performing pretty well.

Fig 3.6 Confusion Matrix for Naïve Bayes model



ROC – AUC curve: As can be seen from the figure below the AUC is 67, which is good compared to Decision tree model. We can say that our model is performing fairly.

Fig 3.7 ROC – AUC curve for Naïve Bayes model



Chapter 4: Conclusion

Now that we have a few models for predicting the target variable, we need to decide which one to choose. There are several criteria that exist for evaluating and comparing models. We can compare the models using any of the following criteria:

1. Predictive Performance
2. Interpretability
3. Computational Efficiency

In our case of predicting Santander Customer transaction, Computation Efficiency and predictive performance hold much higher significance than interpretability. Therefore, we will use Predictive performance and Computational efficiency as the criteria to compare and evaluate models.

Predictive performance can be measured by comparing Predictions of the models with real values of the target variables, and calculating some average error measure. For our case we have used Confusion matrix and ROC as an error metric measure.

For our model it is highly important that it be precise, because correct prediction will help the customer grow their business. Considering all the attributes from confusion matrix and ROC curve, I choose Naïve Bayes to be the final model.

Naïve Bayes has a very high precision and accuracy. Though the AUC value is lower compared to Logistic regression model, for our model precision holds a higher weightage.