Predictive Analytics of Institutional Attrition

by

Sindhu Velumula

B.Tech., VNR Vignana Jyothi Institute of Engineering and Technology, India, 2016

A REPORT

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computer Science
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2018

Approved by:

Major Professor
Dr. William H. Hsu

Copyright

Abstract

Institutional attrition refers to the phenomenon of members of an organization leaving it over time - a costly challenge faced by many institutions. This work focuses on the problem of predicting attrition as an application of supervised machine learning for classification using summative historical variables. Raising the accuracy, precision, and recall of learned classifiers enables institutional administrators to take individualized preventive action based on the variables that are found to be relevant to the prediction that a particular member is at high risk of departure. This project focuses on using multivariate logistic regression on historical institutional data with wrapper-based feature selection to determine variables that are relevant to a specified classification task for prediction of attrition.

In this work, I first describe a detailed approach to the development of a machine learning pipeline for a range of predictive analytics tasks such as anticipating employee or student attrition. These include: data preparation for supervised inductive learning tasks; training various discriminative models; and evaluating these models using performance metrics such as precision, accuracy, and specificity/sensitivity analysis. Next, I document a synthetic human resource dataset created by data scientists at IBM for simulating employee performance and attrition.

I then apply supervised inductive learning algorithms such as logistic regression, support vector machines (SVM), random forests, and Naive Bayes to predict the attrition of individual employees based on a combination of personal and institution-wide factors. I compare the results of each algorithm to evaluate the predictive models for this classification task.

Finally, I generate basic visualizations common to many analytics dashboards, comprising results such as heat maps of the confusion matrix and the comparative accuracy, precision, recall and F1 score for each algorithm. From an applications perspective, once deployed, this model can be used by human capital services units of an employer to find actionable ways (training, management, incentives, etc.) to reduce attrition and potentially boost longer-term retention.

Table of Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my committee members William H. Hsu, Dr. Mitchell Nielsen, and Dr. Torben Amtoft for taking time to serve on my committee and their support in the process of this project.

I am especially indebted to my major advisor Dr. William H. Hsu for believing in my abilities and for his constant support from my very first semester here at Kansas State University.

I am also grateful to Dr. Mitchell Nielsen, Dr. Doina Caragea, and Dr. Jorge Valenzuela for being amazing mentors during my Teaching Assistant phases.

I would like to extend a huge thanks to all my family members without whom this journey would not have been possible. I am extremely grateful to my mom, Revati Velumula, dad, BuchiReddy Velumula for their unconditional love, encouragement, and emotional support in my life. I cannot thank my sister, Pooja Velumula enough for her immense love and support. She is my strength and has played a significant role in what I have achieved so far in my life.

And last, but in no way the least, I would like to thank my friends Pavan Manepalli, Poojitha Bikki, Pruthvidhar Dhodda, Ravi Raja Mallala, Sharmila Vegesana, and Sneha Gullapalli for their support and friendship. The past two years would not have been so memorable and fun without all of them.

Chapter 1 - Introduction

This chapter discusses the problem statement, objectives and a synopsis of this project.

1.1 Problem Definition

Attrition in the sense of organizational management refers to a person leaving an organization. In particular, employee attrition refers to the number of employees leaving an organization, and student attrition refers to the number of individuals who leave a program of study before finishing it. Predicting attrition has become an important concern for the institutions in recent days, owing to several reasons. For companies, the costs of employee attrition ranges from calculable numbers to hidden costs, which include costs incurred in job postings, hiring process, paperwork, and new hire training. Also, until the new hires learn the business, overall productivity of the company may be lowered. Student attrition has a different kind of influence on academic institutions, affecting university rankings, school reputation, and financial well-being of the institution. Thus, predicting attrition and its determining factors, is at the forefront of needs of human resources in many organizations. Also, it would enable the institution's management to take individualized and collective preventive actions.

Generally, a certain amount of attrition is inevitable and typically unpredictable because of the limitations of the data along with the human related factors involved in causing attrition. However, usually some noticeable broad patterns in the attrition trend can be observed when attrition over a reasonable period of time is examined. Hence, there is a need for predictive analytics in this domain. This work deals with the task of predicting the attrition in an institution by using machine learning methods.

1.2 Goals and Technical Objectives

The aim of this work is to build a supervised inductive learning models which adopt classification methods to predict the attrition in an institution. I try to predict attrition of an individual employee using various company wide and personal factors. The data set used for this task is “HR Employee Attrition and Performance” data set, created by scientists at IBM. I try to classify the “attrition” attribute to “Yes” or “No” classes. For the task of predicting student attrition in an institution, I use student academic and personal data provided by the Kansas State University. I try to predict attrition of a student owing to factors like Cumulative GPA, Housing, Permanent State Address and others.

Discriminative models such as Logistic Regression, support vector machines (SVM), Random Forests, and Naive Bayes were used for binary classification in this project. The general objective was to classify the attrition attribute and to determine the important features in predicting the rate of attrition in an institution. Also, the results obtained on different classification models are presented.

1.3 Synopsis

In the employee data set, each record has various attributes such as ‘Age’, ‘JobRole’, ‘Educational Field’, ‘Marital Status’, ‘Monthly Rate’, ‘Gender’ etc. In the student data set, each record has attributes such as ‘Housing’, ‘admission date’, ‘Cumulative GPA’ etc. Data is cleaned and preprocessed. Then, feature selection is done to determine the right subset of features that reduce the complexity of a model and improve the accuracy. The data set is split into training and testing data sets. The supervised inductive learning algorithms were trained on the training data set to predict the attrition and testing data set was used to test the performance of these learning models. The results of the algorithms were compared using the performance metrics such as precision, accuracy, recall and ROC curves of these algorithms. *Scikit-learn* (Pedregosa, Varoquaux, Gramfort, & Michel, 2011), a machine learning library written in the Python programming language is used to implement the above-mentioned tasks. Other Python visualization libraries such as *Matplotlib* (Hunter, Dale, Firing, Drothboom, & team, 2012 - 2018) and *Seaborn* (Waskom, 2012 - 2018) were also used for visualizing the results.

Chapter 2 - Background and Related Work

This chapter discusses the background information of the classification techniques used in this project.

2.1 Literature Survey: Classification Problem

Classification is one of the most common tasks in machine learning. It can be referred to as a problem of identifying the items belonging to a predefined set. A classification model is trained on the training data set and the resulting model can be used to classify an unknown object in the test data. This trained model produces predictions of a target variable over formerly unseen data, which represent an employee or a student in this project. The classification of a data object is based on the idea of finding similarity with predefined objects that represent different classes and each new object is classified as a particular class, with a certain probability of accuracy. Also, in a classification task, the number of classes is known ahead and remains unchanged. Since labeled input data is used to train the model, this is considered as a supervised learning technique. Some typical examples of where classification algorithms can be used are: classifying whether a tumor is malignant or benign, classifying whether an email is a spam or not, etc. The main goal of this project is to predict the attrition of individuals in an institution. For this purpose, the idea of classification is used. All the attributes are provided, the classifying model is trained on labeled input data and probability of individuals belonging to the "attrition" or the "no attrition" class is determined.

2.2 Established Classification methods

2.2.1 Logistic Regression

Logistic Regression is one of the most famous machine learning algorithms. This algorithm involves a probabilistic view of classification to measure the relationship between a categorical dependent variable and one or more independent variables. The dependent variable is the target class variable, and the independent variables are the features or attributes used to predict the target class variable. This uses a sigmoid function for building its output to return a probability value which can be mapped to two or more discrete classes.

Depending on the number and ordering of the dependent variables, there are various types of Logistic Regression including:

i. Binary

ii. Multinomial

iii. Ordinal

To use binary logistic regression, the dependent variable should be dichotomous and this is used in this project as we trying to classify the dependent variable “attrition” into one of the two classes: ”yes” and “No”. Logistic function, also called as a sigmoid function, is an S-shaped curve that takes any real-valued number and maps it into a value between 0 and 1, but never exactly at those margins.

The sigmoid function is given by:

$$S(x) = \frac{1}{1 + e^{-x}}$$

Figure 2.1 Sigmoid Function

In the above formula, *S(x)* is the output, *x* is the function's input and *e* is the base for natural log.

To map the output of prediction function to a discrete class, threshold value should be selected. Assuming the two discrete class values possible to be "1" and "2". If the probability score is more than or equal to the threshold, we classify the value into a Class 1, and to class 2 if it less than the threshold value.

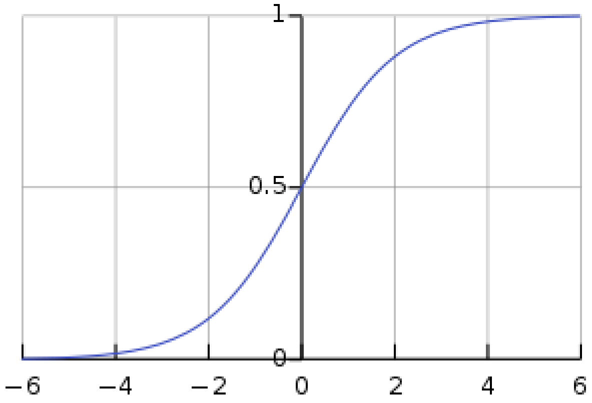


Figure 2.2 Logistic Regression

Using sigmoid function and the threshold values, logistic regression can be used to classify an observation into one of the two possible discrete classes.

2.2.2. Support Vector Machines (Support Vector Classification)

Support Vector Machines (SVM) is one of the popular methods for supervised machine learning for classification, regression and other machine learning tasks. These are a set of supervised learning methods used for classification, regression and outlier's detection. A SVM classifies data by determining the best hyper plane that separates all data points of one class from other class. There can be infinite number of hyperplanes that separate the data points. The best hyperplane is the one that maximizes the margin. Support vectors are the data points that lie on the boundaries of the slab parallel to the hyperplane having no interior data points.

The following figure illustrates these definitions, with + indicating data points of class “Yes” for attrition, and – indicating data points of class “No” for attrition.

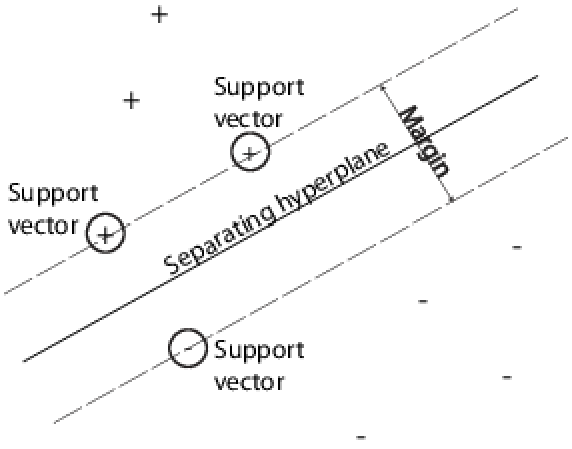


Figure 2.3 SVM

The generalized equation of a hyperplane is:

$$w^T x = 0$$

$w^T x$ is the dot product of the two vectors w and x. The vector w is generally called as the weight vector.

Calculating a hyperplane with no interior data points is not always possible, hence some misclassification of the data points is allowed. This is known as a Soft Margin Classifier or a Support Vector Classifier. The support vector classifier contains a tuning parameter which represents the amount of misclassification allowed.

2.2.3. Naive Bayes

Naive Bayes classifiers are a set of algorithms that are based on Bayes' theorem and work on the assumption that every pair of features in the training data are conditionally independent of each other. It is one of the commonly used basic classifier, because it is easy and fast to implement. Also, it is quite good in classifying real world data sets. It calculates the probabilities of each attribute and selects an outcome with the highest probability.

Bayes Rule is:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}$$

Figure 2.4 Bayes' Rule

Given a data set with n features represented by X_1, X_2, X_3….X_n and classes represented by C_1, C_2, C_3…C_n.

For classification, 'A' represents a class value and 'B' represents the set of features X_1, X_2, X_3….X_n. And since, Since P (B) serves as normalization, and we are often unable to calculate P(X_1, X_2, X_3….X_n). So, by ignoring the term P (B),

$$\begin{aligned} P(c_i|x_0,\ldots,x_n) &\propto P(x_0,\ldots,x_n|c_i)P(c_i) \\ &\propto P(c_i)\prod_{j=1}^{n} P(x_j|c_i) \end{aligned}$$

To determine the class that a given data point falls into, the class with the highest probability value is selected. That is,

$$y = \underset{c_i}{argmax}\ P(c_i) \prod_{j=1}^{n} P(x_j|c_i)$$

This is referred to as the Maximum A Posteriori decision rule.

2.2.4. Random Forests

Random Forests is also one of the widely used algorithms in machine learning, because of its simplicity and the fact that it can be used for both classification and regression tasks. It builds an ensemble of Decision Tree. The decision trees individually may over fit the data set and so they are combined to form a much stronger model. In Random Forests, many decision trees are built during the training, and output is the class that appears the most as output on all single trees. Random Forests select random vector (X, Y) from training set where X = $\{X_1, X_2 \ldots X_n\}$, a set of features and $\forall X \in R+$ contains predictors and $Y \in \{Y_1, Y_2 \ldots Y_n\}$, a set of class labels.

Classifier c is a mapping:

$$c : R+ \in \{ Y_1, Y_2 \ldots Y_n\}$$

One of the important characteristic of the Random forest algorithm is that it is easy to measure the relative importance of each feature on the prediction. *Scikit-learn* provides a great tool that computes features importance automatically for each feature after training and by scaling the results, so that the sum of all importance is equal to 1. Overfitting is not easy to happen to a random forest classifier because the classifier will not over fit the model, if there are enough trees in the forest.

Chapter 3 - Implementation

This chapter includes an overview of the data, data preprocessing methods and implementation steps of this work.

3.1 Overview of the Data

Any machine learning task depends heavily on data. The quality, amount, preparation, and selection of data is important to the success of any machine learning solution.

For this task of predicting attrition in an institution, I use two data sets. For predicting employee attrition, I use a “HR Employee Attrition and Performance data set” provided by IBM. This data set is downloaded from the IBM site. For the task of predicting student attrition, I use student academic and performance data provided by Kansas State University.

The employee data set includes 35 different variables about employee satisfaction, income, seniority, demographics and attrition result. The data set has 1470 rows of records.

Each employee record in the employee data set contains the following fields:

- **Age**: Numerical value denoting Employee age.
- **Attrition**: Employee leaving the company (Yes or No values).
- **Business Travel:** The extent which an employee travels on business purpose (No Travel, Travel Frequently and Travel Rarely values).
- **Daily Rate:** Numerical value denoting employee salary level.
- **Department**: This value denotes employee department (Human Resources, Research & Development, and Sales values).
- **Distance From Home**: Numerical value denoting the distance from work to home.

- **Education**: Numerical value denoting the education level of the employee.
- **Education Field:** Field of education of the employee (Human Resources, Life Sciences, Marketing, Medical, Other and Technical Degree).
- **Employee Count:** Numerical value the number of employees in each row of the data set.
- **Employee Number:** Employee ID number.
- **Environment Satisfaction:** Numerical value denoting an employee satisfaction with the environment.
- **Gender**: Gender of an employee (female and male values)
- **Hourly Rate:** Numerical value denoting employee hourly salary.
- **Job Involvement:** Numerical value denoting employee job involvement.
- **Job Level:** Numerical value denoting employee level of job.
- **Job Role**: Role of employee in the company (HC Rep, Lab Technician, Manager, Managing Director, Research Director, Research Scientist, Sales Executive and Sales Representative values)
- **Job Satisfaction:** Numerical value denoting employee satisfaction with the job.
- **Marital Status:** Indicates whether the employee is single, married or divorced.
- **Monthly Income**: Numerical value denoting employee monthly income.
- **Monthly Rate:** Numerical value denoting employee monthly salary rate.
- **Number of Companied Worked:** Numerical value denoting the number of companies employee worked before joining the current company.
- **Over18:** Yes or No value denoting whether the employee age is over 18 or not.
- **Overtime:** Yes or No value denoting whether the employee works overtime or not.

- **Percentage Salary Hike:** Numerical value denoting a percentage by which employee salary increased.
- **Performance Rating:** Numerical value denoting employee performance rating value.
- **Relations Satisfaction:** Numerical value denoting employee relations satisfaction.
- **Standard Hours:** Numerical value denoting the standard number of hours an employee works biweekly.
- **Stock Option Level:** Numerical value denoting the stock options the employee has in the company.
- **Total Working Years:** Numerical value denoting the number of years the employee has been working in all the previous job(s) and the current job.
- **Training Times Last Year:** Numerical value denoting the number of times the employee has received training last year.
- **Work Life Balance:** Numerical value denoting the work life balance of an employee.
- **Years At Company:** Numerical value denoting the number of years the employee has been working with the current company.
- **Years in Current Role:** Numerical value denoting the number of years the employee has been working in the current job role.
- **Years since Last Promotion:** Numerical value denoting the number of years since the employee has received his last promotion.
- **Years with Current Manager:** Numerical value denoting the number of years the employee has been working with the current manager.

The following bar graph shows the distribution of attrition in the employee data set:

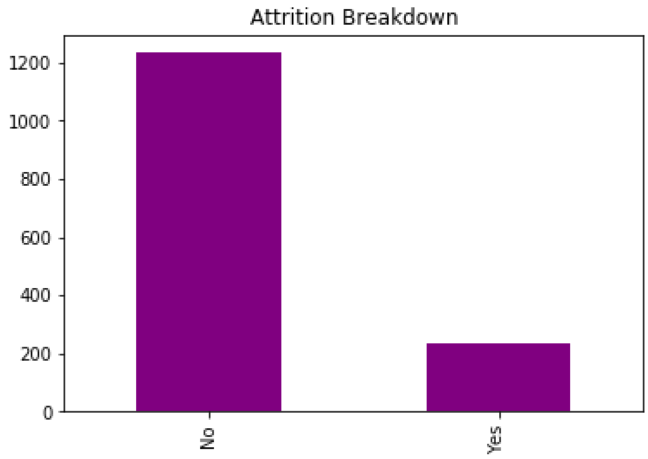


Figure 3.1 Attrition breakdown

Personal and academic student data for the semesters Fall 2012 - Spring 2018 was requested and received from Kansas State University. Prediction of student attrition tasks were performed on this data. Details of the student data set are not included in this report due to data privacy agreement with Kansas State University.

3.2 Data Preparation

Data preparation is such an important step in the machine learning process. In a nutshell, data preparation is a set of procedures that help make your dataset more suitable for machine learning. In broader terms, the data preparation also includes establishing the right data collection mechanism.

3.2.1 Data Preprocessing

It is possible that a data set in its raw form can include missing values, inconsistent entries and noise. These type of data entries may affect the quality of the results. Thus raw content should be processed to eliminate unnecessary data and avoid misleading outcomes. Before passing the data to the classifier, data cleaning and preprocessing steps were applied on each data set. Both the employee and student data sets are searched for missing or null values. No such rows are found in the employee data set. Some student data set rows had some missing values, to handle such situations, the missing fields have either been filled with dummy values or the best possible values, depending on the best suited option for a data column.

Python was used as the major programming language for this project. Using Pandas, which is a Python library often used for data manipulation and analysis tasks, data from a .csv file is loaded onto a Python pandas data frame. Next, the data frame is divided into testing and training data. This is done by using train_test_split from Scikit-learn, a library in Python that provides many unsupervised and supervised learning algorithms. Scikit-learn was also used to train the inductive learners and apply them to the data sets. Also, it is very important to avoid selection bias which occurs when the samples used to produce the model are not fully representative of cases that the model may be used for in the future. From the figure 3.1, which shows attrition breakdown, it is observed that majority of the employees have attrition value “No’. To balance the data set, an oversampling technique called SMOTE is applied on the training data set.

3.2.2 Feature Selection

The following columns are dropped from the employee data set owing to the respective reasons:

1. Employee Count: This attribute value is the count of employee in each row and the value is “1” for all the employees.

2. Employee Number: This feature is an identifier for an employee and hence does not have any significance for classification.

3. Standard Hours: This attribute value is same for all the employees and the value is 80.

4. Over18: This attribute value is also same for all the employees and the value is “Y”.

Feature selection is a process of selection of a subset of features**,** to be used in the predictive modeling process. It can improve accuracy, reduce overall training times, and increase model generalizability. Scikit-learn’s Recursive feature elimination (RFE) is used for feature selection. First, the estimator is trained on the initial set of features and the importance of each feature is obtained either through a `coef_` attribute or through a `feature_importance_` attribute depending on the algorithm used for training. Then, the least important features are removed from current set of features. This procedure is repeated on the new set until the desired number of features to select with a desired accuracy is eventually reached. Some classification models gave better results when they were trained on the selected features, where as other worked best when all the features are used.

3.3 Implementation Steps

The process flow along with implementation steps is shown in the following diagram.

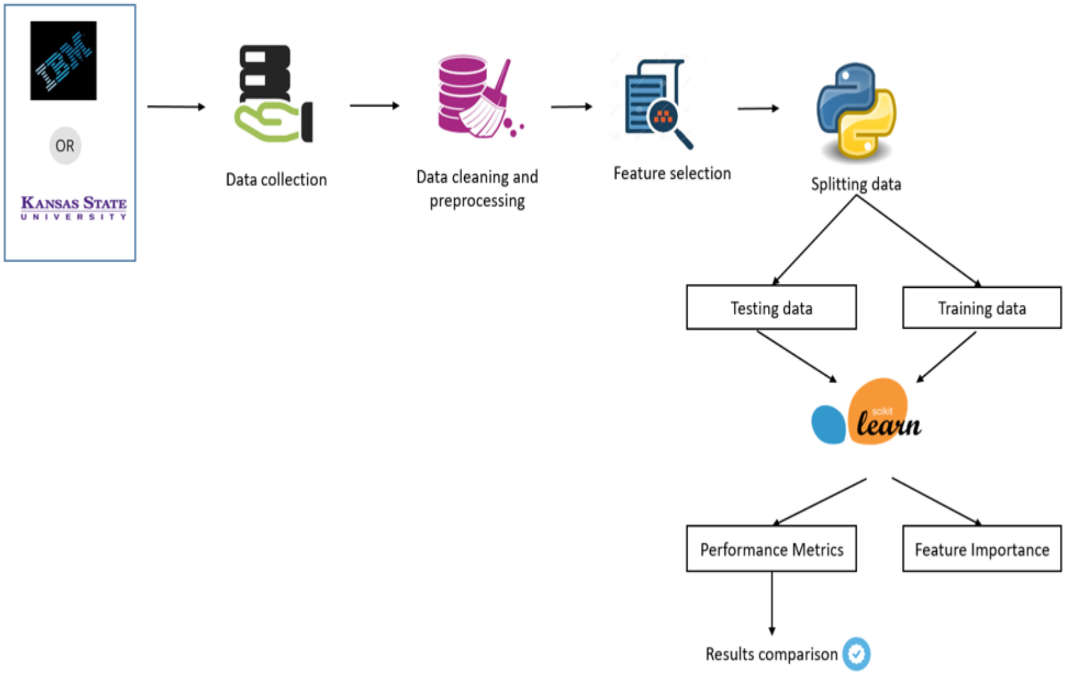


Figure 3.2 Process Flow of Project

Chapter 4 - Experiments

A detailed explanation of the experiments conducted on the data set is given in this section.

4.1 Training and Test Data Sets

An important part of evaluating classification models is separating data into training and testing sets. The training set is used to build the model and the test set is used to measure its performance. I divided both the data sets at the proportion of 70% to 30%, where 70% is training data and 30% is test data.

Table 4-1 Employee Data Set

Total Records	1470
Training Records	1029
Testing Records	4441

Table 4-2 Student Data Set

Total Records	215183
Training Records	150628
Testing Records	64555

4.2 Experiment Design

The approach used to conduct the experiments on the data sets using various machine learning algorithms mentioned in Chapter 2 is discussed in detail in this section. Scikit-learn, a machine learning package in Python has been used to implement all the inductive learning models to predict the attrition of individuals in an institution. The classification models were trained using the training set and the resulting models were evaluated on the test set. The results of the classification were later compared to estimate the best classifier for individual test beds.

In *k-fold* cross validation (Pedregosa et al., 2010), the training set is split into *k* subsets sets of data and for each of the *k* folds, a model is trained using all but the one *(k-1)* of the folds and resulting model is evaluated on the remaining fold of the data. This process is repeated k-times, each time leaving a different fold to be not used for training. In sections 5.3 the mean cross validation scores, and various metrics like accuracy, precision, recall, and F1-score are compared to evaluate the classifier models for individual test beds.

4.2.1 Logistic Regression

Logistic Regression in Scikit-learn is implemented by importing the *LogisticRegression* class in following way: *from sklearn.linear_model import LogisticRegression*. Then the classifier model was fit on the training data using the function *fit*. *predict_proba* function was used to predict the probabilities of the testing data and the function *predict* was used to make the predictions for class labels for individuals in the testing data.

4.2.2 Random Forests

The random forest learning algorithm was implemented by importing the library *sklearn.ensemble.RandomForestClassifier* from Scikit-learn. The depth, estimators and random state fields were specified before fitting the model and predicting the scores.

4.2.3 Naive Bayes

Similar methods were used to import the Gauissian Naïve Bayes library from Scikit-learn, using: *from sklearn.naive_bayes import GaussianNB*. The function fit was used to fit the learning model on the data and the function score was used to find out the F-score of this algorithm and to assess its performance.

4.2.4 SVM

Similar to the above classifiers, I implemented SVM by importing *sklearn.svm.SVC.*

Chapter 5 - Results

This chapter includes the results of the experiments mentioned in Section 4.2.1 - 4.2.4 and the evaluation metrics used to evaluate the performance of those machine learning models.

5.1 Evaluation Metrics

5.1.1 Accuracy

Accuracy can be defined as the number of correct predictions made to the total number of predictions.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$

Equation 5.1 Accuracy Formula

5.1.2 Precision, Recall, and F-Score

Precision can be defined as the ratio of number of true positive values in an evaluation to the total number of true positives and false positives.

$$Precision = \frac{true\ positives}{true\ positives + false\ positives}$$

Equation 5.2 Precision Formula

Recall can be defined as the ratio of number of true positive values in an evaluation to the total number of true positives and false negatives.

$$Recall = \frac{true\ positives}{true\ positives + false\ negatives}$$

Equation 5.3 Recall Formula

F1 Score is the harmonic mean of precision and recall F1. F1 score is a better measure if we are trying to achieve a balance between precision and recall.

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

Equation 5.4 F1 Score Formula

5.1.3 Receiver Operating Characteristic Curve (ROC)

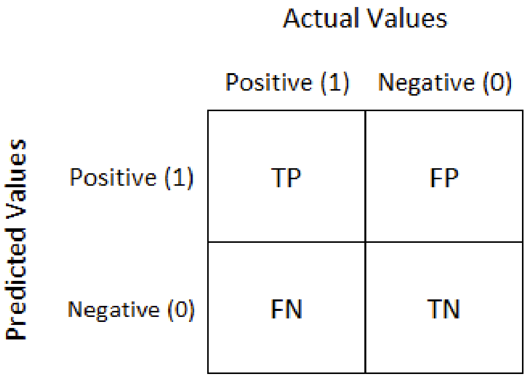
An ROC curve is one of the most widely used methods to for checking any classification model’s performance. It is a graph drawn between the true positive rate and the false positive rate at various threshold settings. Area under the curve (AUC) is a measure of separability which tells how much the model is capable of distinguishing between class labels. The maximum value of AUC is 1 and the higher the AUC, the better the accuracy of prediction of the classifier. *roc_auc_score* and *roc_curve* funtions are imported from *sklearn*.metrics library and are used to plot the ROC curve.

5.1.4 Cross validation

To evaluate each model, I used the *k*-fold cross-validation technique: iteratively training the model on different subsets of the data and testing against the held-out data. This is implemented by importing *cross_val_score* function from *sklearn.model_selection* library. I used *10-fold cross validation* to evaluate the performance of classifier models.

5.1.5 Confusion Matrix

A *confusion matrix* for a model, is a table which has the true positive, true negative, false positive and false negative values for the given test data. The diagonal elements represent the number of objects of the test data which are correctly labeled by the model and off-diagonal elements represent the number of mislabeled objects of the test data by the model. The higher the diagonal values of the confusion matrix that better the classification model is at prediction.



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 5.1 Confusion Matrix

Where TP= True positives, FP = False positives, FN = False negatives, TN = True negatives

5.2 Experimental Results

The results obtained for the algorithms discussed in Chapter 4 are explained here using their performance metrics.

5.2.1 Employee Data

5.2.1.1 Logistic Regression

The accuracy, precision, recall and F1 score values obtained when this model was run on the employee data set are given in the table below:

Table 5-1 Evaluation Metrics - Logistic Regression

Evaluation Metric	Value
Accuracy	85.48
Precision	85.15
Recall	85.48
F1 score	85.31

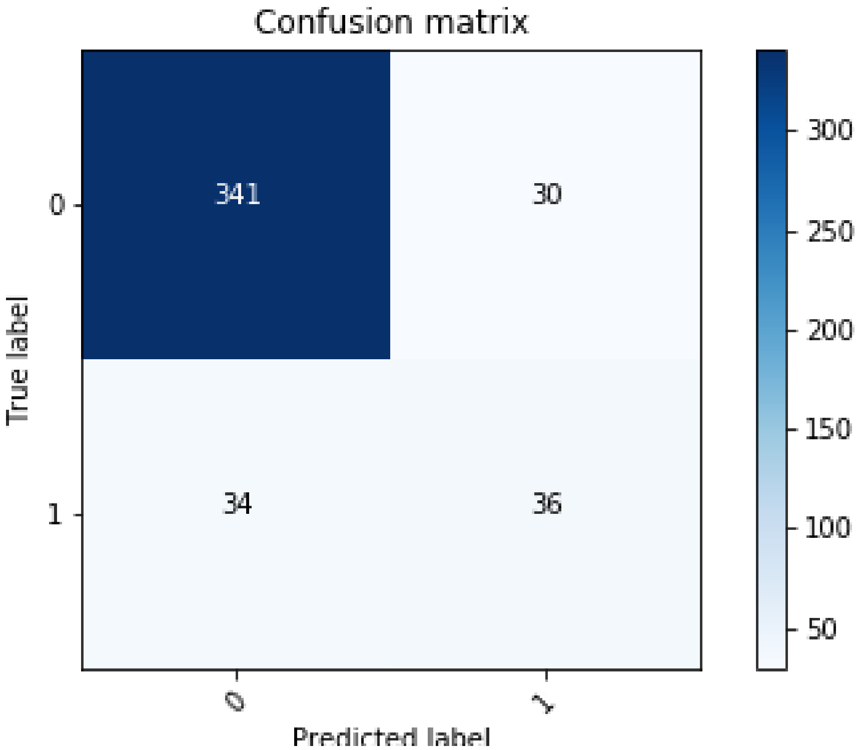


Figure 5.2 Confusion Matrix for Logistic Regression

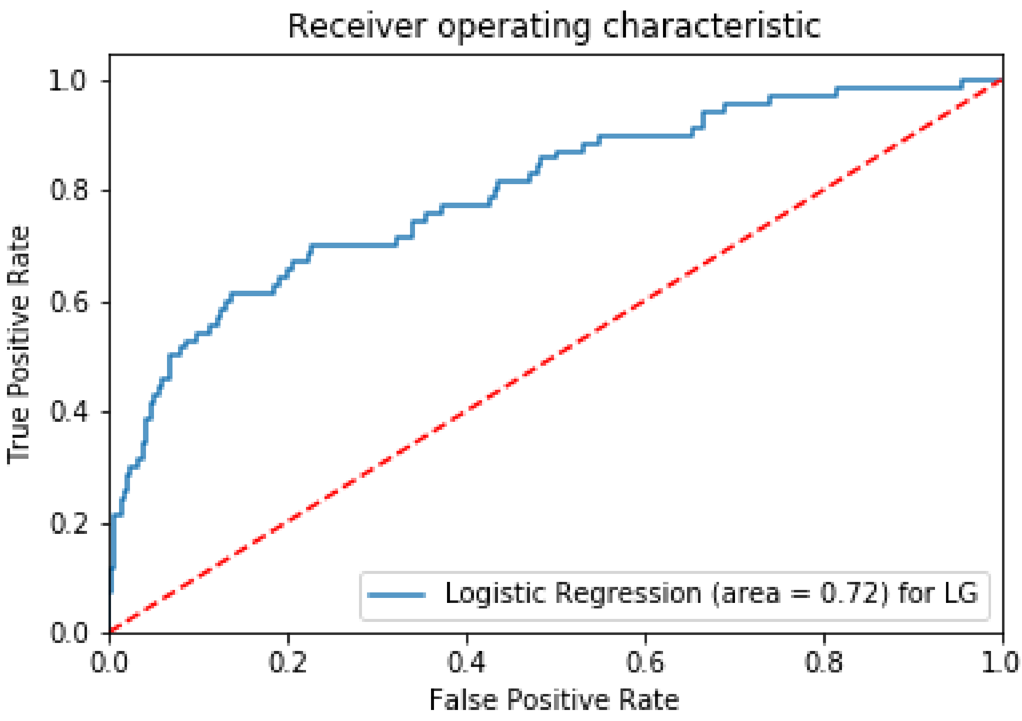


Figure 5.3 Receiver Operating Characteristic Curve for Logistic Regression

5.2.1.2 SVM

The parameters for this algorithm were Kernel = linear and probability = True after a series of experiments. The accuracy, precision, recall and F1 score values obtained when this model was run on the employee data set are given in the table below:

Table 5-2 Evaluation Metrics - SVM

Evaluation Metric	Value
Accuracy	0.7551
Precision	0.7976
Recall	0.7605
F1 score	0.7228

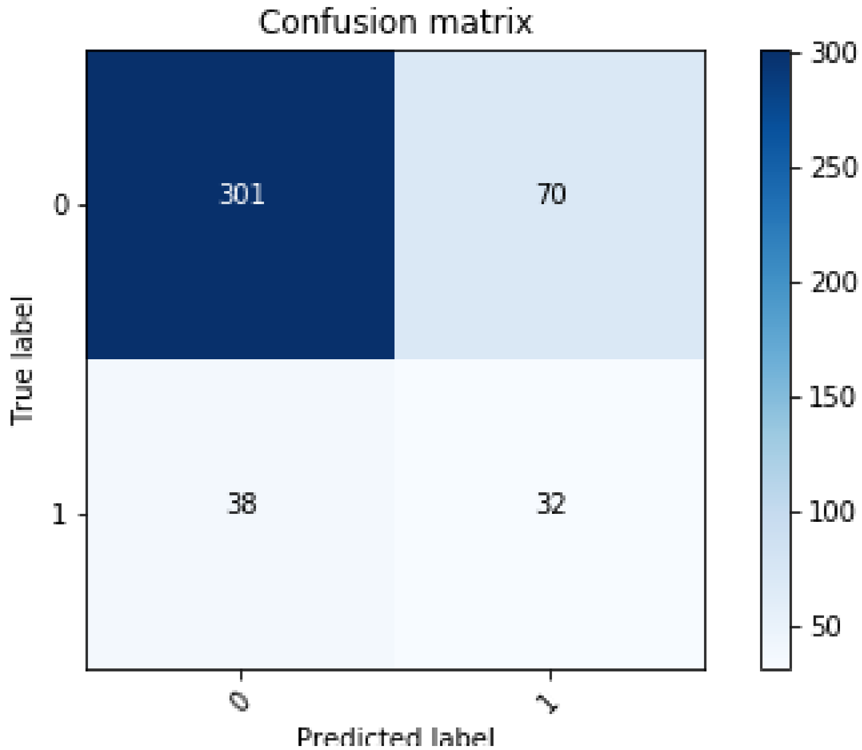


Figure 5.4 Confusion Matrix for SVM

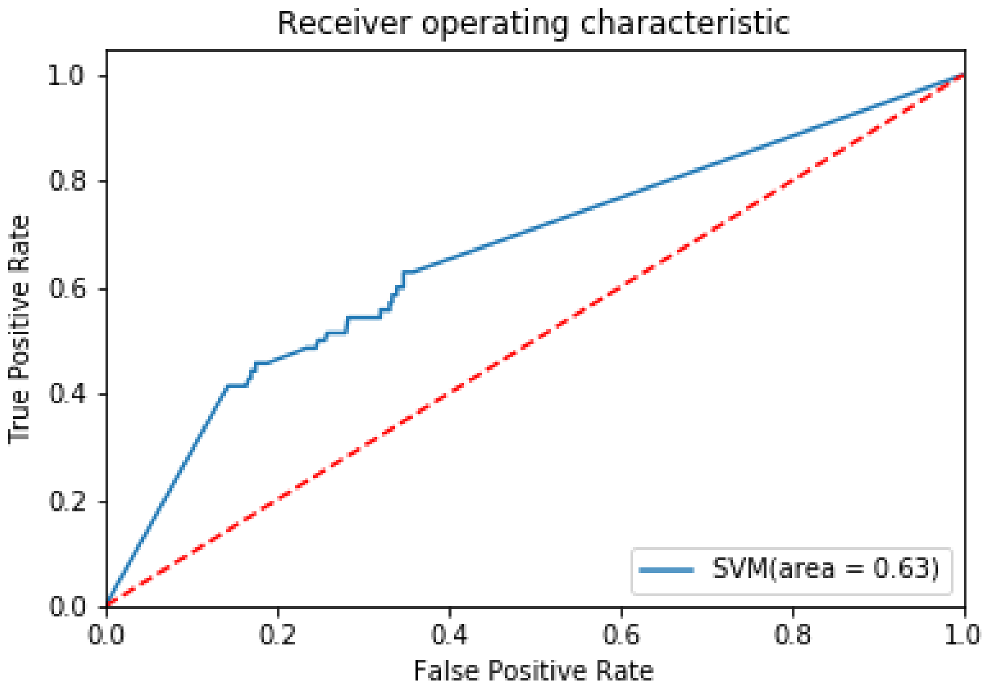


Figure 5.5 Receiver Operating Characteristic Curve for SVM

5.2.1.3 Naive Bayes

Gaussian Naive Bayes classifier is used for this project. The accuracy, precision, recall and F1 score values obtained when this model was run on the employee data set are given in the table below.

Table 5-3 Evaluation Metrics - Naive Bayes

Evaluation Metric	Value
Accuracy	0.6961
Precision	0.7970
Recall	0.6991
F1 score	0.7286

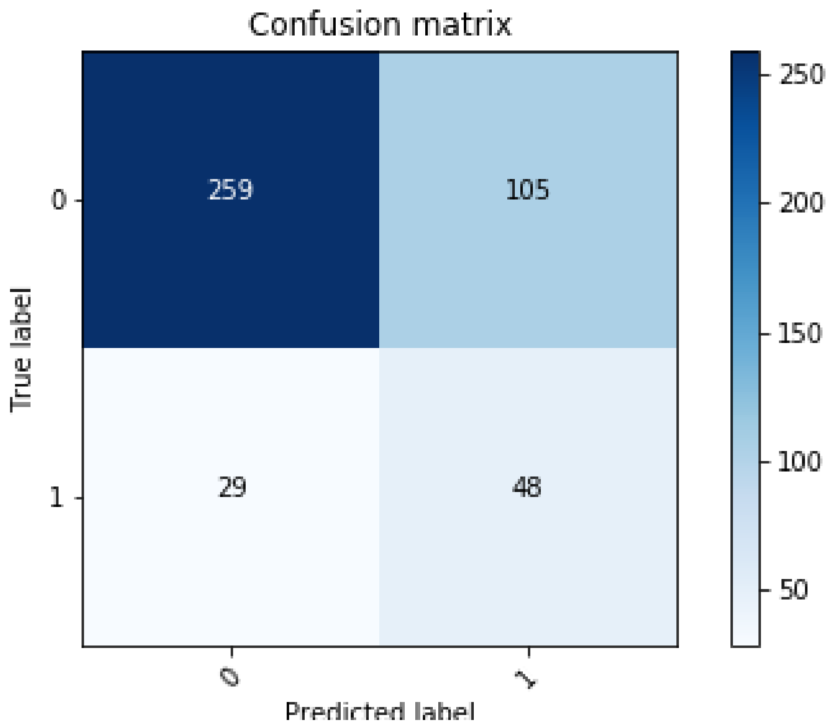


Figure 5.6 Confusion Matrix for Naive Bayes

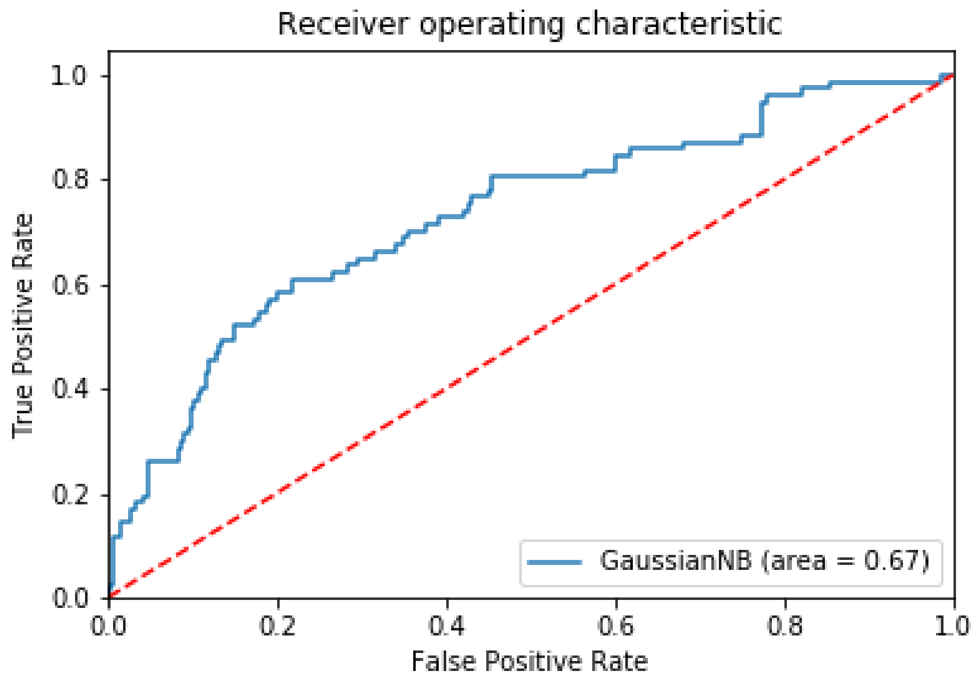


Figure 5.7 Receiver Operating Characteristic Curve for Naive Bayes

5.2.1.4 Random Forests

The parameters for this algorithm were: max_depth = 2, and n_estimators=100. After a series of experiments, these values were chosen as they gave the best prediction accuracy. The accuracy, precision, recall and F1 score values obtained when this model was run on the employee data set are given in the table below.

Table 5-4 Evaluation Metrics - Random Forests

Evaluation Metric	Value
Accuracy	0.7959
Precision	0.8307
Recall	0.7959
F1 score	0.8096

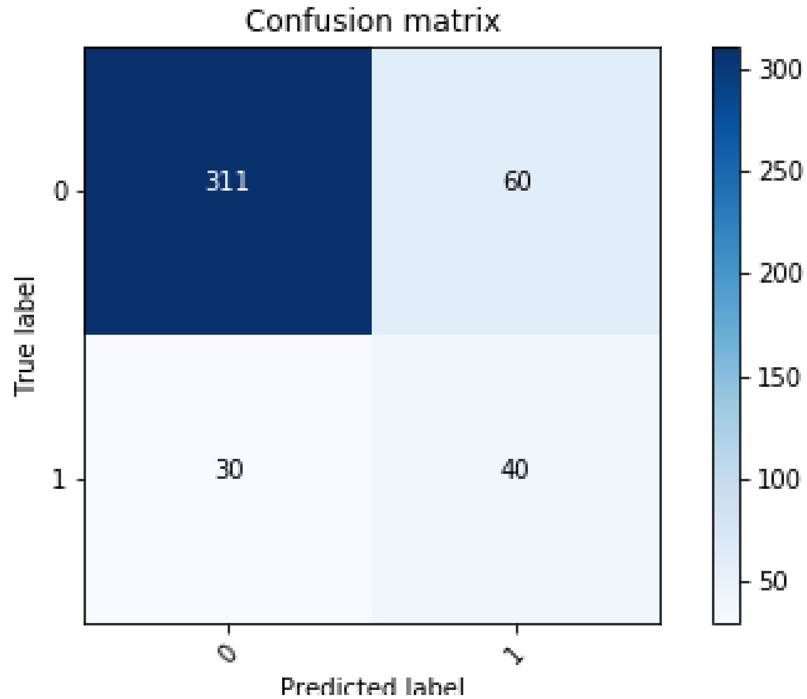


Figure 5.8 Confusion Matrix for Random Forests

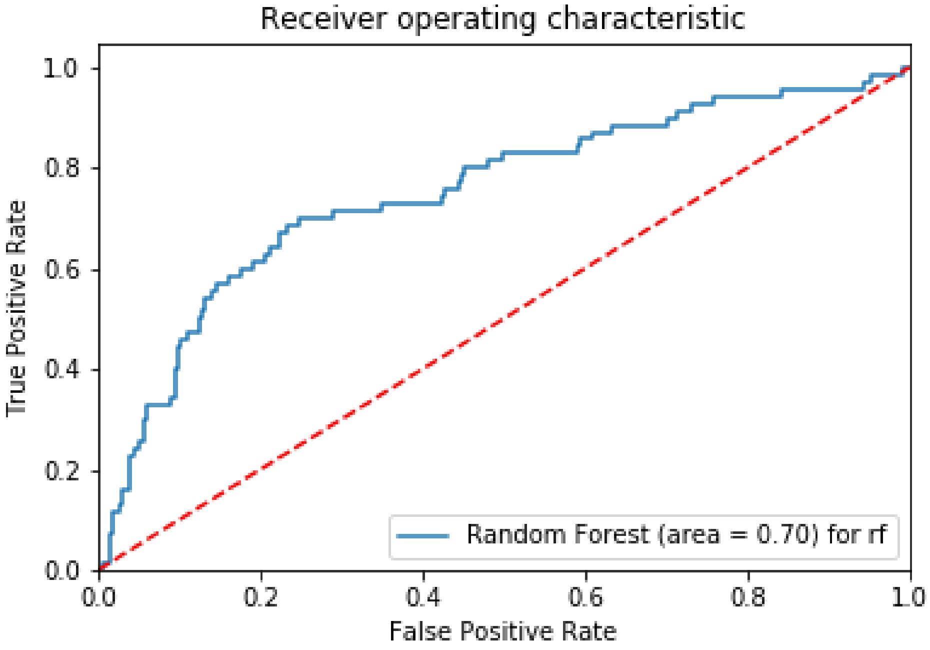


Figure 5.9 Receiver Operating Characteristic Curve for Random Forests

5.2.2 Student Data – Random Forests

The accuracy, precision, recall and F1 score values obtained when this model was run on the student data set are given in the table below.

Table 5-5 Evaluation Metrics for Random Forests - Student Data

Evaluation Metric	Value
Accuracy	0.811
Precision	0.907
Recall	0.811
F1 score	0.849

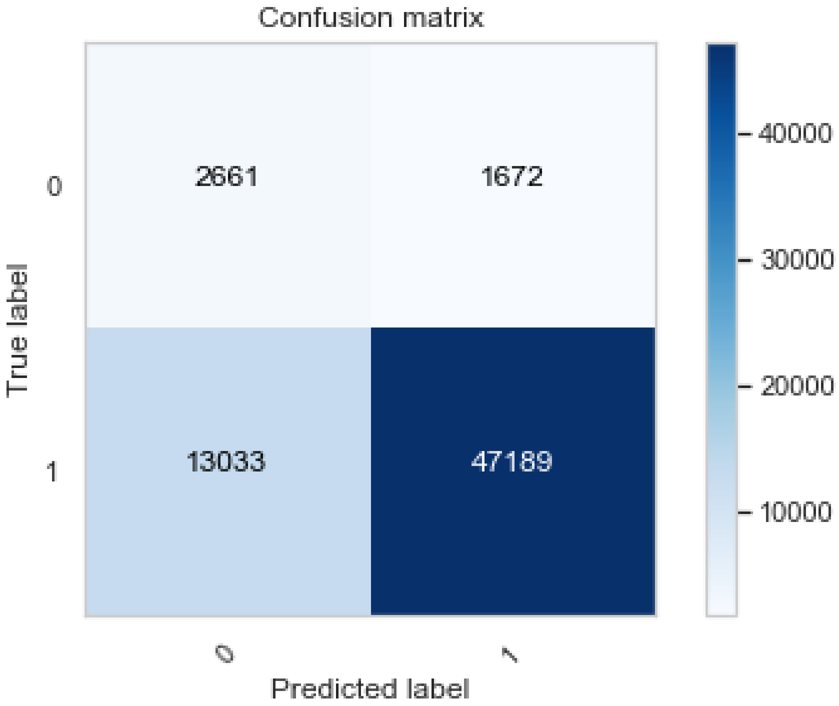


Figure 5.10 Confusion Matrix for Random Forests - Student Data

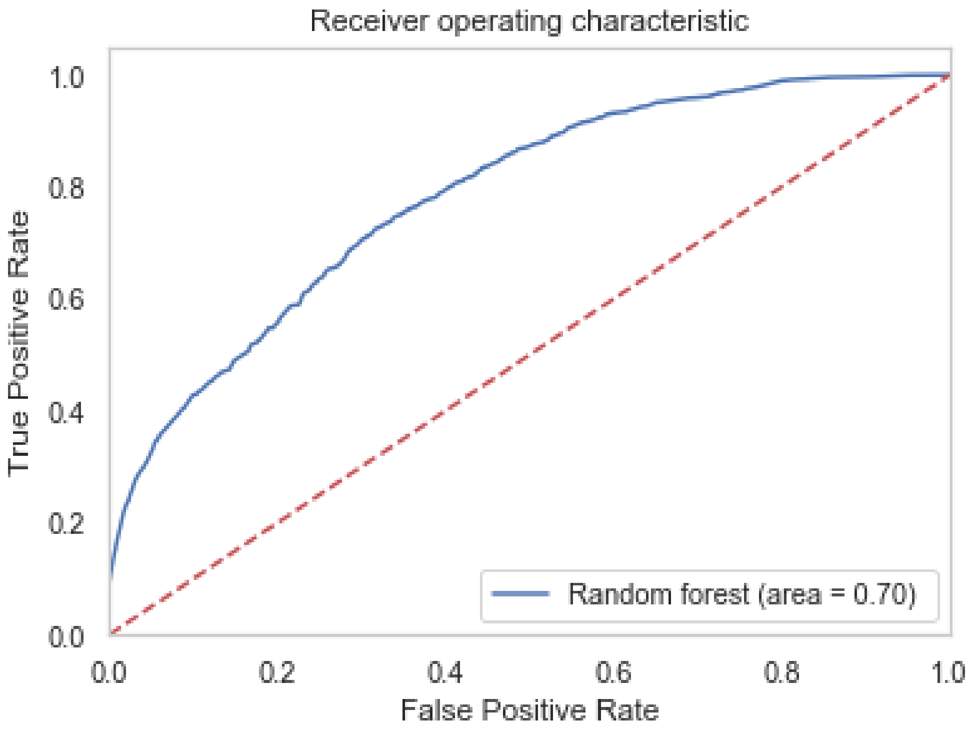


Figure 5.11 Receiver Operating Characteristic Curve for Random Forests - Student Data

5.3 Comparison of all the classifiers

5.3.1 Cross validation:

A comparison of mean accuracy values of 10 fold cross validation of each of the above classifier models is shown in the following table.

Table 5-6 Cross validation mean accuracies for all classifiers

Classification model	Mean accuracy
SVM	0.893
Logistic Regression	0.880
Random Forests	0.782
Naive Bayes	0.741

5.3.2 Accuracy, precision, recall, F1 score values

Table 5-7 Accuracy, precision, recall , F1 Score values for all classifiers

Classification model	Accuracy	Precision	Recall	F1 score
Logistic Regression	0.8548	0.8515	0.8548	0.8531
SVM	0.7551	0.7976	0.7605	0.7228
Naive Bayes	0.6961	0.7970	0.6991	0.7286
Random Forests	0.7959	0.8307	0.7959	0.8096

5.3.3 Receiver Operation Characteristic (ROC) Curve

A graph illustrating all the classifier's performance was compiled.

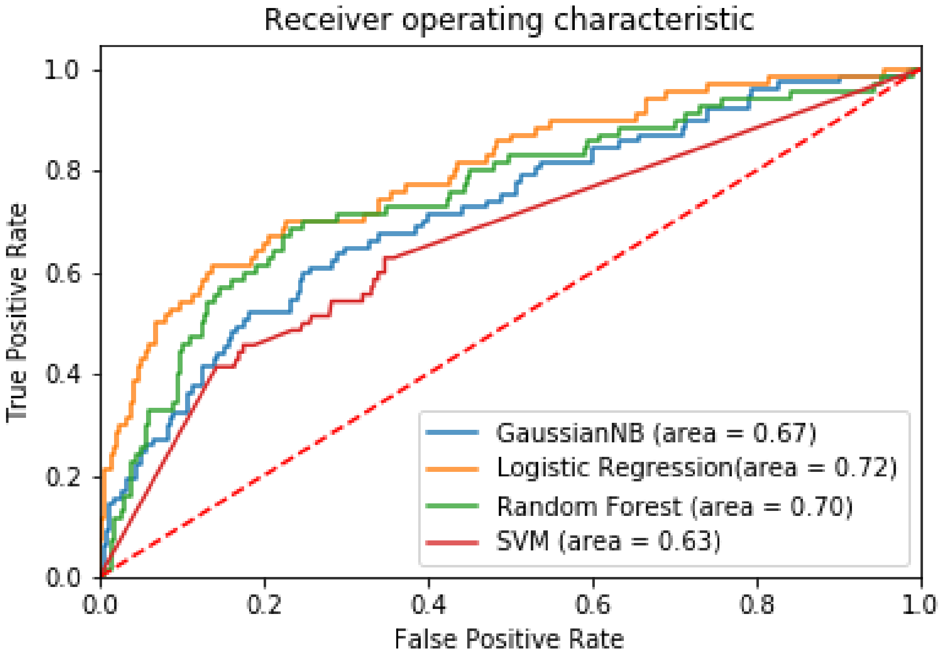


Figure 5.12 Receiver Operating Characteristic Curve for all classifiers

5.4 Feature Importance graph – Employee Data

A relative feature importance graph is plotted based on the `coef_` values for the logistic regression.

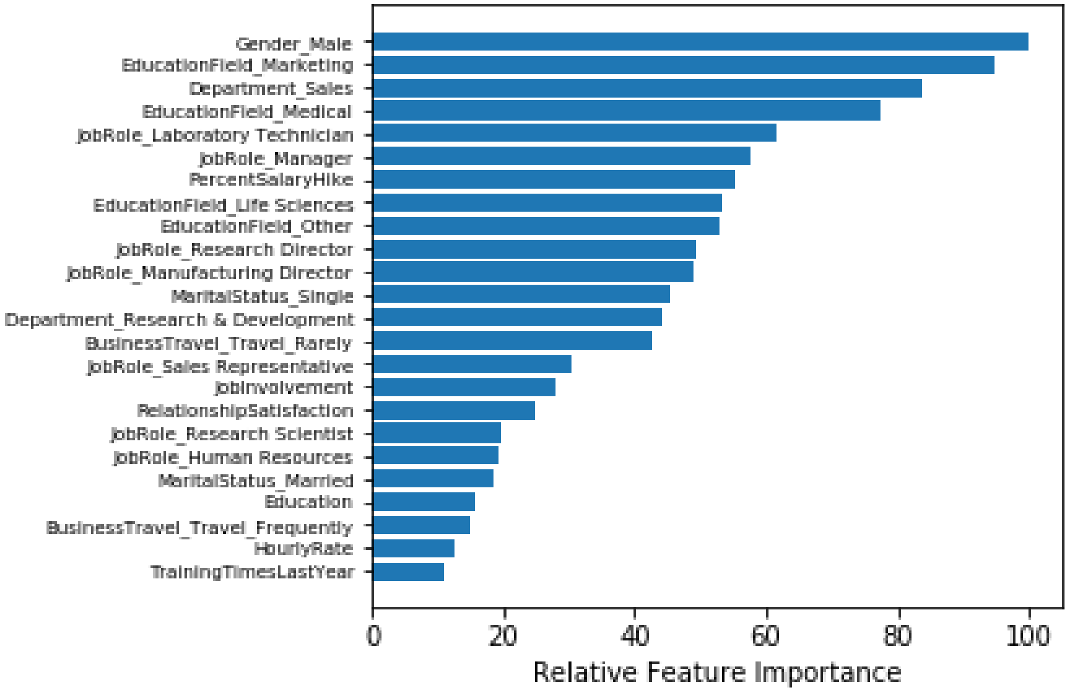


Figure 5.13 Relative feature importance graph

5.5 Feature Importance graph – Student Data

A relative feature importance graph is plotted based on the `feature_importances_` values for the random forests.

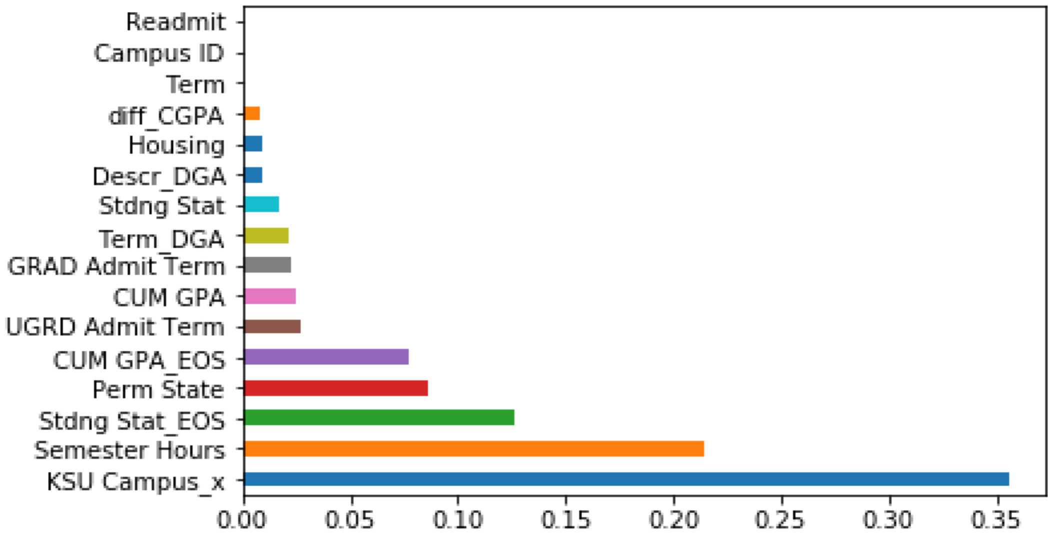


Figure 5.14 Relative Feature Importance Graph - Student Data

Chapter 6 - Summary and Future Scope of Work

6.1 Summary and Interpretation of Results

Based on the experimental results shown in chapter 5, it was observed that Logistic regression out-performed other classification models. Apart from the accuracy value, the values of accuracy, precision, recall, F1 score values of logistic regression are better compared to other models. Though SVM had good accuracy value, the other evaluation matric values were low compared to all other models. Random forests is the next best alternative option as it had similar performance metrics to logistic regression.

6.2 Future Scope of the Project

The data set used for the task of analyzing employee attrition is a fictional data set created by data scientists at IBM. Generally, the real time employee data is not made available by most of the institutions owing to privacy issues. The next step would be to implement the employee attrition prediction task on real world data and also further improve the performance of the classification models. Other feature selection algorithms like LASSO, Elastic Net and Ridge Regression can be applied for feature selection. Regarding the student dropout analysis, financial and other possible data, needs to be requested from the university. Other machine learning techniques can also be applied on the gathered data. The results can be used to analyze campus climate, student success and institutional training.

Chapter 7 - References

Silpa, N. (2015). A Study on Reasons of Attrition and Strategies for Employee Retention. *International Journal of Engineering Research and Applications*, *5*(12), 59-63.

Zhu, Z., Ong, Y. S., & Dash, M. (2007). Wrapper–filter feature selection algorithm using a memetic framework. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *37*(1), 70-76.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *Journal of machine learning research*, *12*(Oct), 2825-2830.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427-437.

Barrett, P., Hunter, J., Miller, J. T., Hsu, J. C., & Greenfield, P. (2005, December). matplotlib--A Portable Python Plotting Package. In *Astronomical data analysis software and systems XIV* (Vol. 347, p. 91).

Nagadevara, V. (2012). Prediction of employee attrition using work-place related variables. *Review of Business Research*, *12*(3), 70-76.

Tinto, V. (1987). *Leaving college: Rethinking the causes and cures of student attrition*. University of Chicago Press, 5801 S. Ellis Avenue, Chicago, IL 60637.

Bean, J. P., & Metzner, B. S. (1985). A conceptual model of nontraditional undergraduate student attrition. *Review of educational Research*, *55*(4), 485-540.

Meyer, D., & Wien, F. T. (2001). Support vector machines. *R News*, *1*(3), 23-26.

Woodley, A., & Parlett, M. (1983). Student drop-out. *Teaching at a Distance*, *24*, 2-23.

Alpaydin, E. (2009). *Introduction to machine learning*. MIT press.

Donges, Niklas. "The Random Forest Algorithm – Towards Data Science." *Towards Data Science*, Towards Data Science, 22 Feb. 2018, towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd.

"Bayesopt." *Reconstructing an Image from Projection Data - MATLAB & Simulink Example*, www.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html.

Shung, Koo Ping. “Accuracy, Precision, Recall or F1? – Towards Data Science.” *Towards Data Science*, Towards Data Science, 15 Mar. 2018, towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9.