**Analysis of Employee Retention**

|  |  |
| --- | --- |
| [COMP 596-002 Top:Intro to Data Science](https://bridgew.blackboard.com/webapps/blackboard/execute/launcher?type=Course&id=_42619_1&url=" \t "_top) | Project execution and implementation document |
| Submitted To: Dr. Yiheng Liang | Submission Date: 20th December 2017 |

Contents

[Tool Used 2](#_Toc501552383)

[Project Team and Contribution: 2](#_Toc501552384)

[Project Execution: 3](#_Toc501552385)

[1. Getting Started 3](#_Toc501552386)

[2. View Data 3](#_Toc501552387)

[3. Data Wrangling 4](#_Toc501552388)

[4. Data Visualization 4](#_Toc501552389)

[5. Correlation: 11](#_Toc501552390)

[6. Splitting the data 12](#_Toc501552391)

[7. Classification and Prediction Models: 13](#_Toc501552392)

[How to Download RStudio: 20](#_Toc501552393)

[References: 21](#_Toc501552394)

## Tool Used

R studio

## Project Team and Contribution:

1. Saurav P. Shrestha – Banner ID: 00369895
   1. Import data from CSV file into R
   2. Verify the structure of the data and its summary
   3. Attribute wise data visualization using
      1. pie() – for pie chart
      2. hist() – for histograms
      3. barplot() – for bar plots
2. Bina Maria – Banner ID: 00366107
   1. Project Proposal documentation
   2. Finding the correlation between the attributes
      1. Numeric Attributes: Package used – “corrplot”
      2. Categorical Attributes: chisq.test()
3. Saumya Bhatnagar – Banner ID: 00370168
   1. Splitting data set into training data and test data
      1. Using sample()
   2. Applying classification models on the data and checking the accuracy
      1. Package used – “caret”, “e1071”, “rpart”, “rattle”, “randomForest”
      2. Function used – naiveBayes(), rpart(), randomForest(), prune(), predict()
      3. Accuracy check using – confusionMatrix()
   3. Project presentation and execution documentation

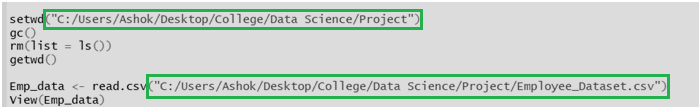
## Project Execution:

For executing this project, we need RStudio. To download RStudio on your desktop, please follow the instructions available [here](#_How_to_Download).

### Getting Started

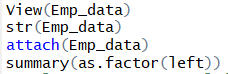
Once RStudio is available, we start with setting the directories.

setwd(“”) commands help us set the working directory. We will mention the path to the folder where we want our work to be saved. We can check the path using getwd() command. Next, let us import the file (Employee\_Dataset.csv) from our system into R, as shown below. We need to provide the path of where our file is stored on the system. Press ctrl+Enter after each line of code to execute it.

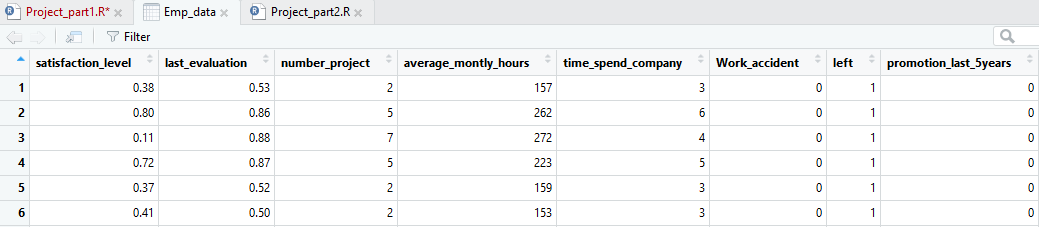


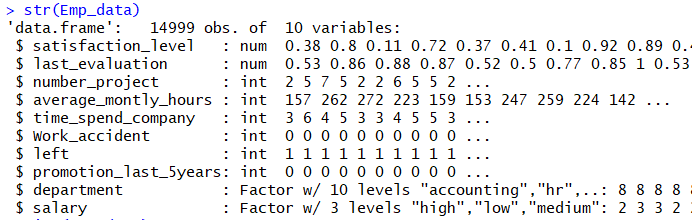
### View Data

Check the data, its structure and summary as below.



Output:







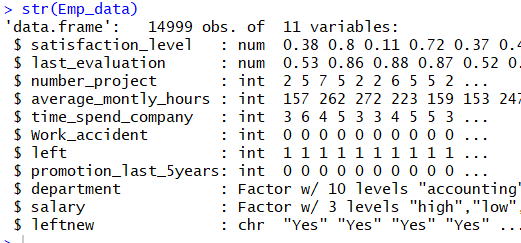
Our data set consists of a total of 14999 observation, and 10 attributes. The attribute Left is our target variable.

### Data Wrangling

We see that our target variable “left” is of type int, and has values 0 and 1. 00 means not left, 1 means left. For the ease of use in fitting the data set in functions ahead, let us convert this variable into categorical variable with values ‘Yes’ and ‘No’, as below.



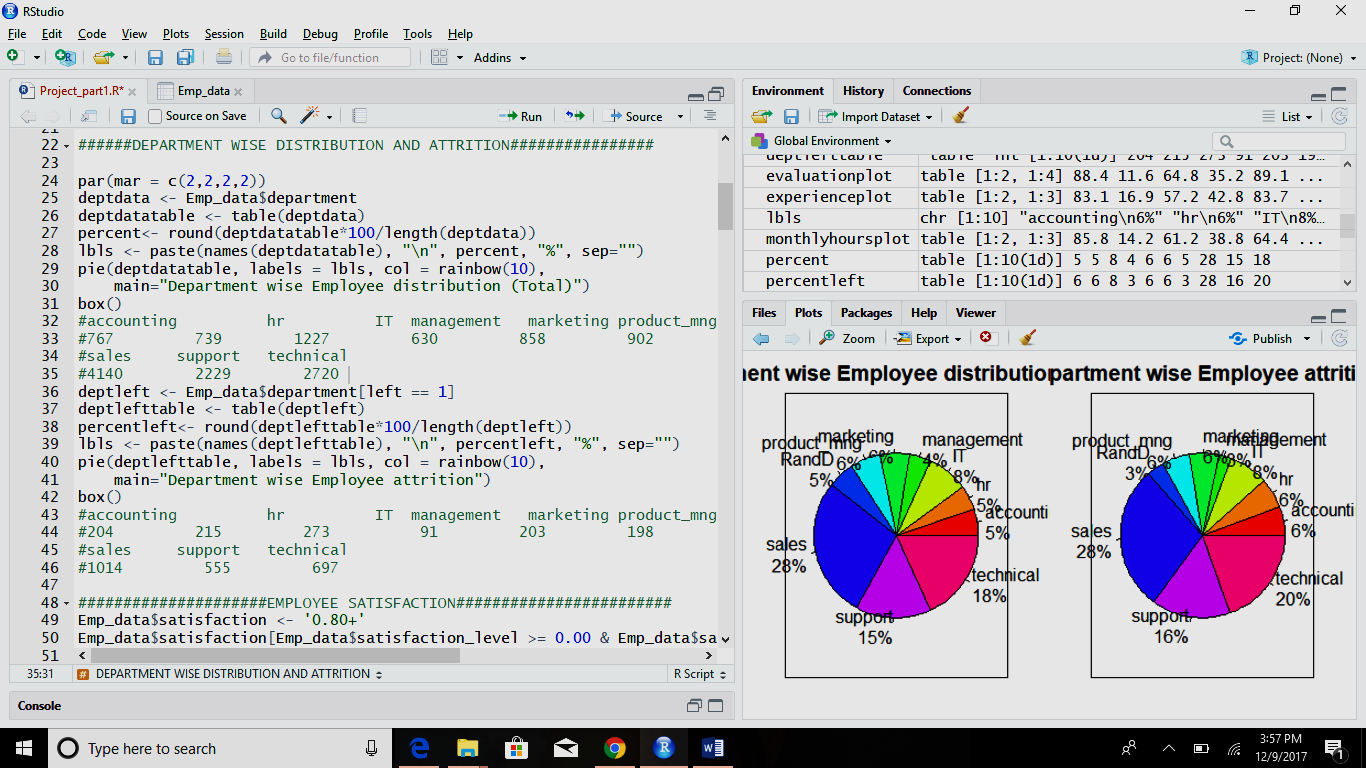
Thus we see our data set has a new column, leftnew.

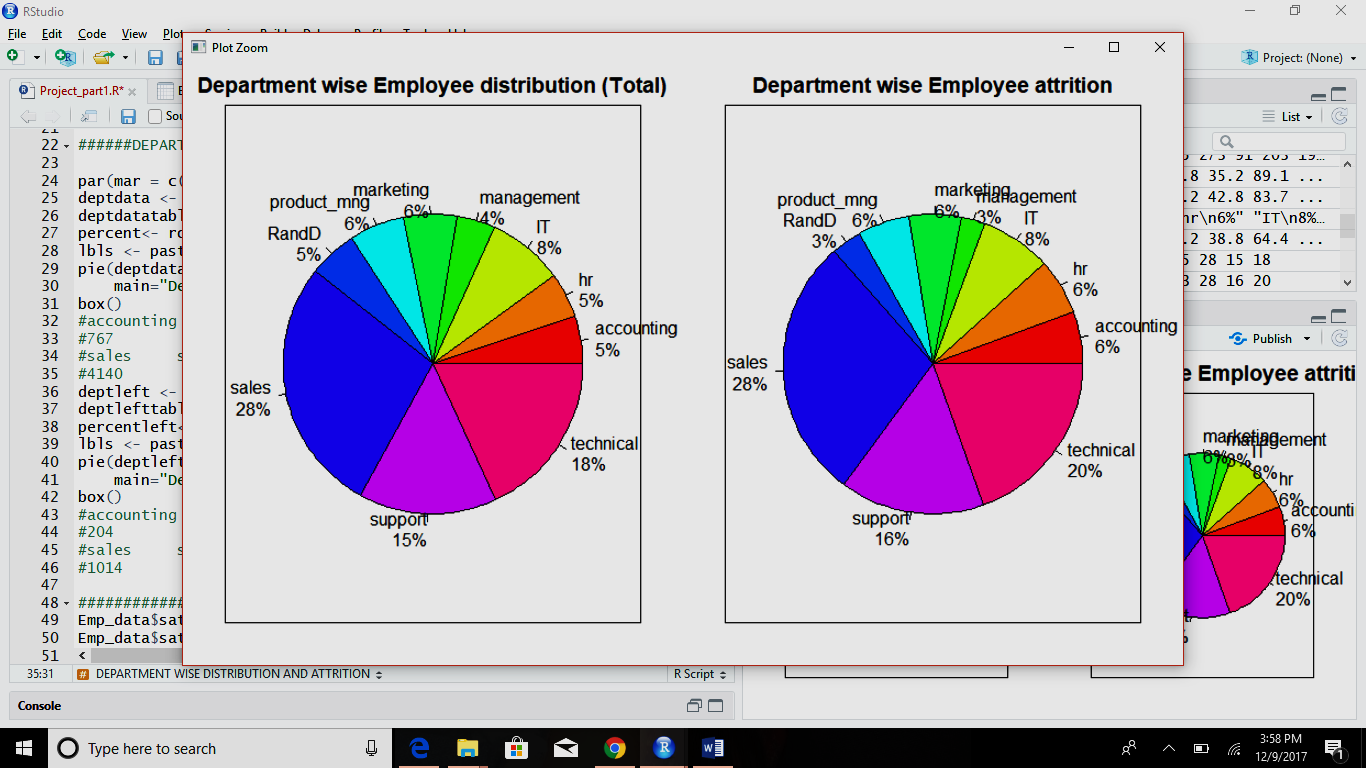


### Data Visualization

Now let us visualize our data.

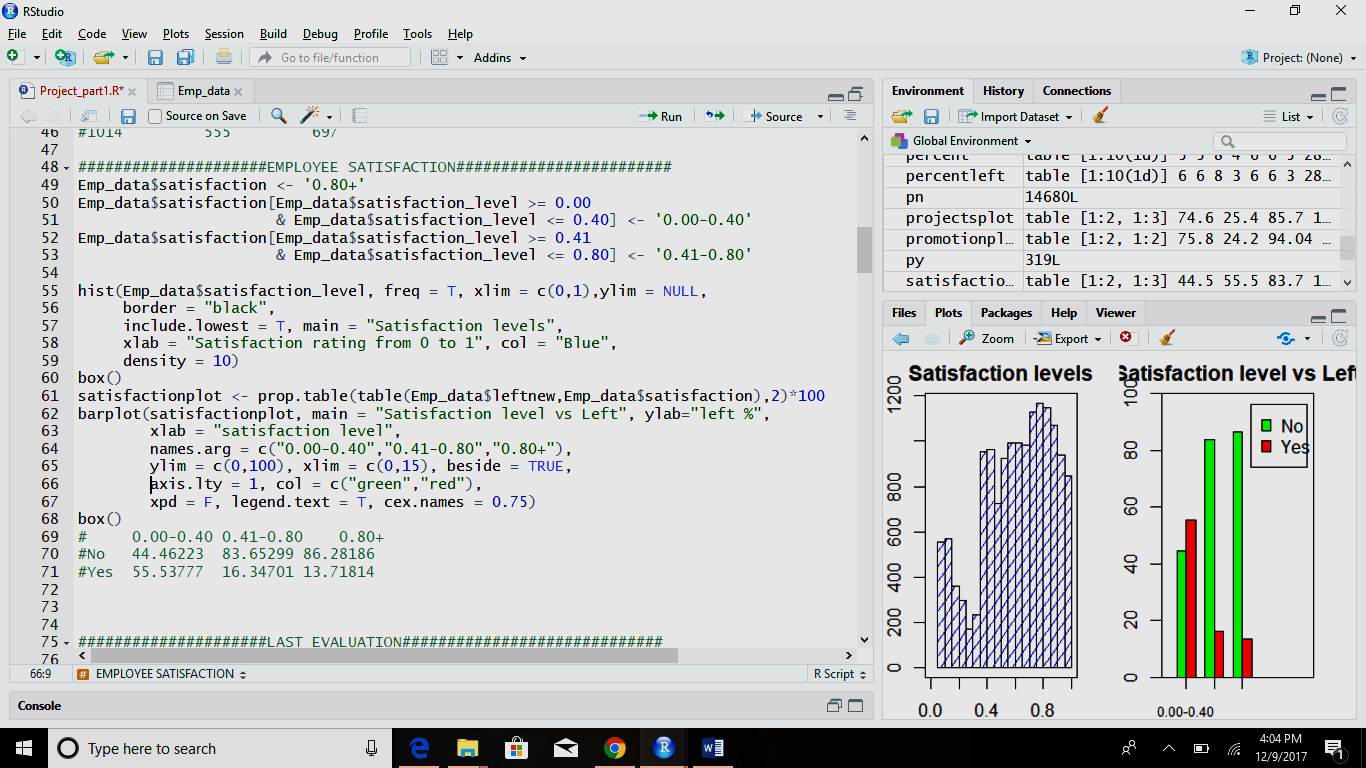
* 1. Analyze the department wise turnouts and find out the percentage of employees leaving from each department

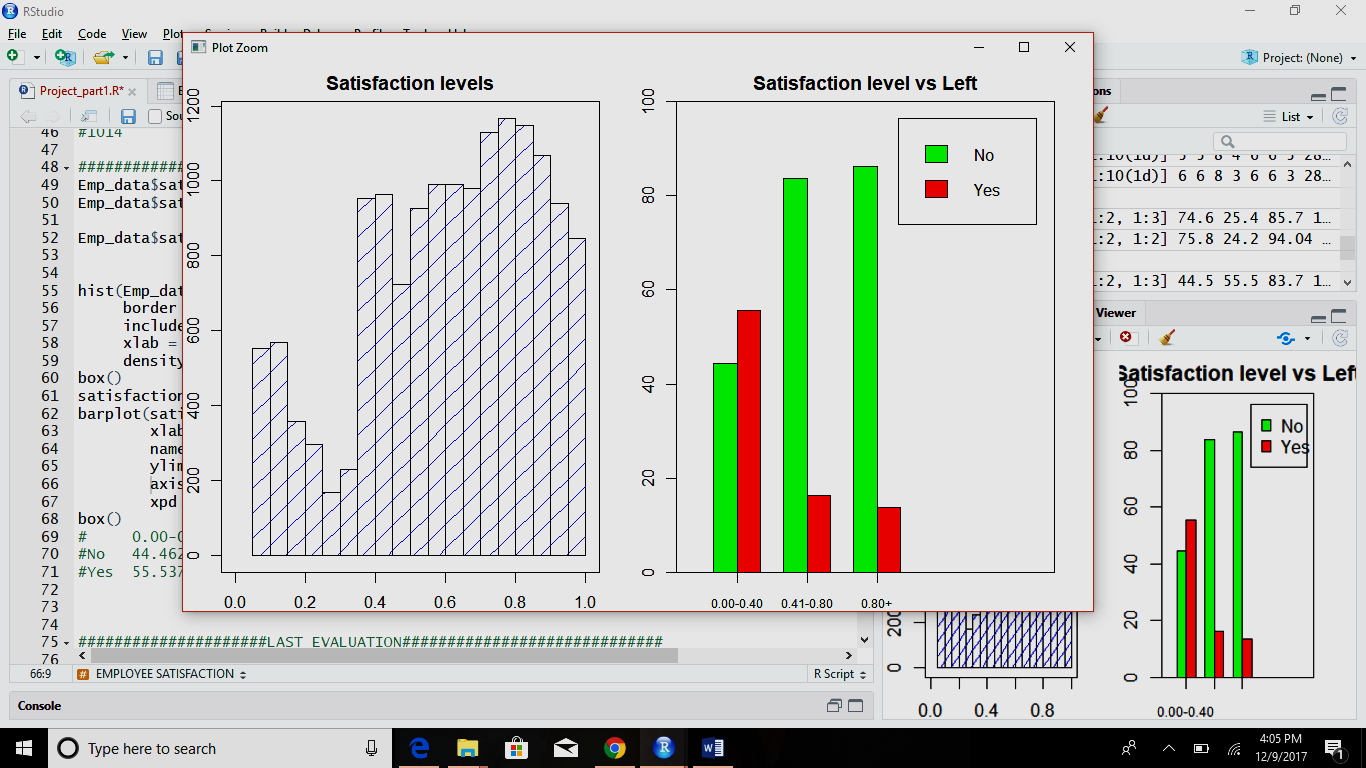




* 1. Employee satisfaction level

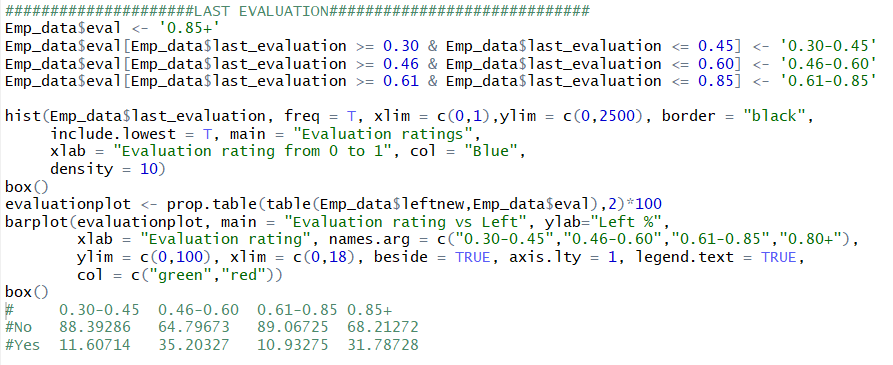
Since satisfaction level is a continuous value attribute, let us make categories for this attribute to better visualize the data.

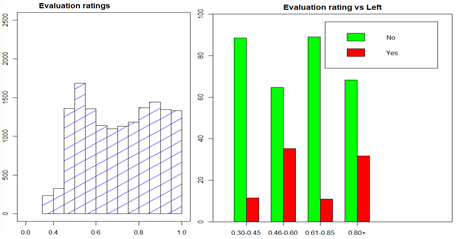




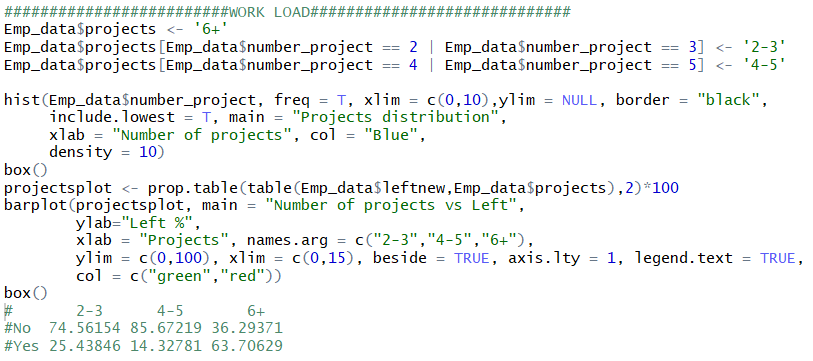
* 1. Ratings from last evaluation

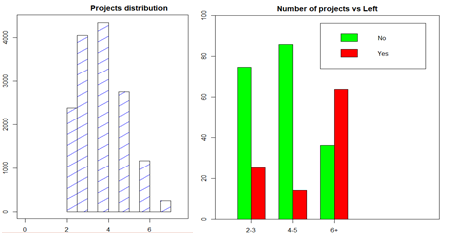
Same as explained above, let us divide the evaluation rating attribute values into categories and see the result.



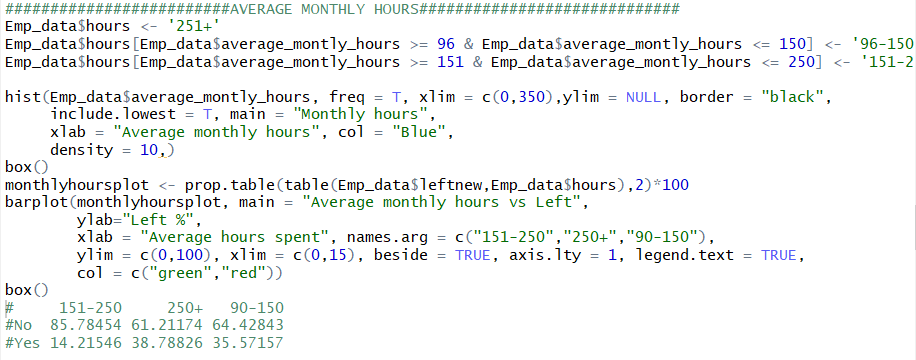


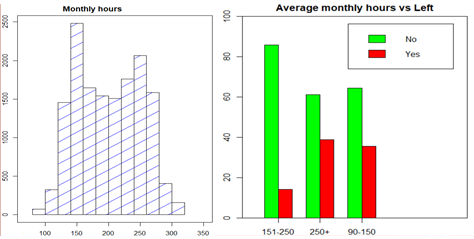
* 1. Work load



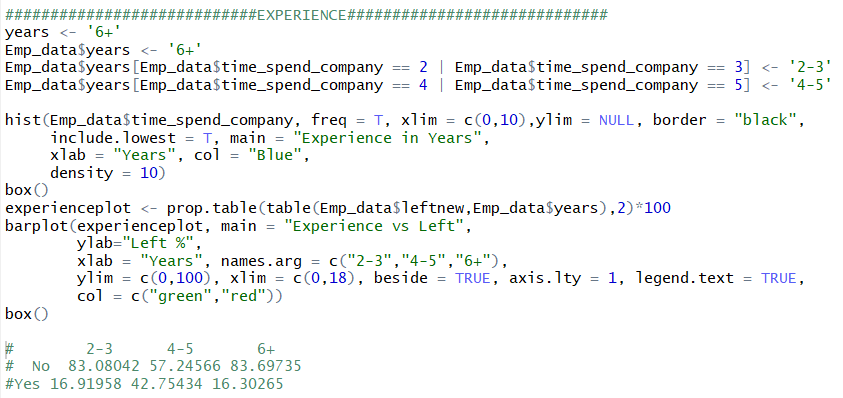


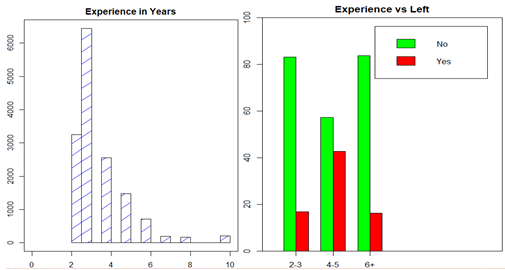
* 1. Average Monthly hours



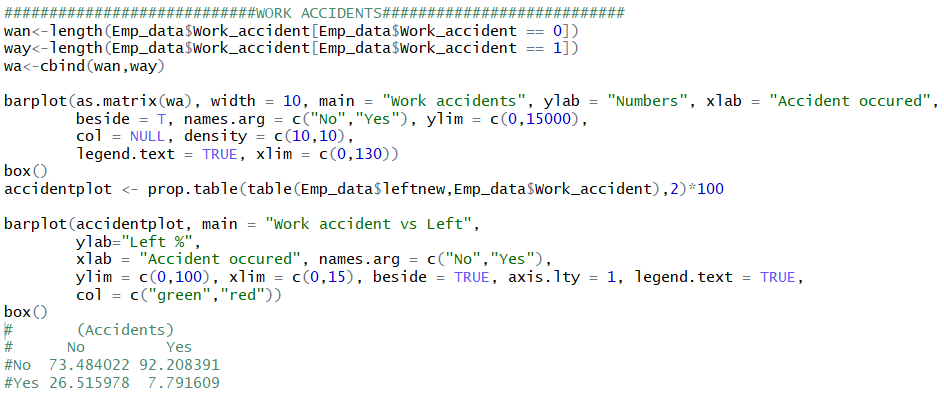


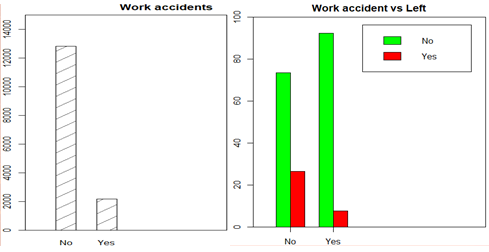
* 1. Experience in the company





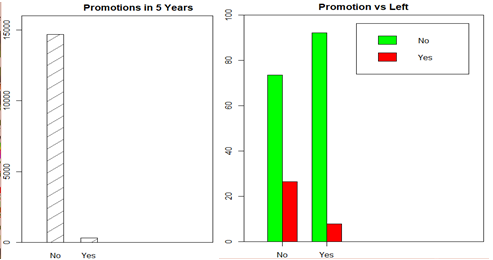
* 1. Work accidents



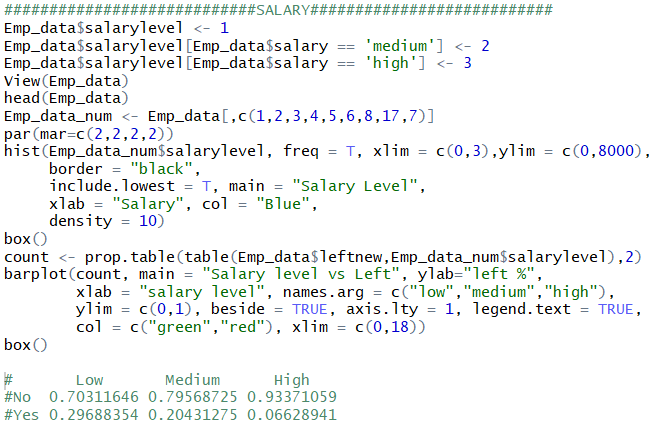


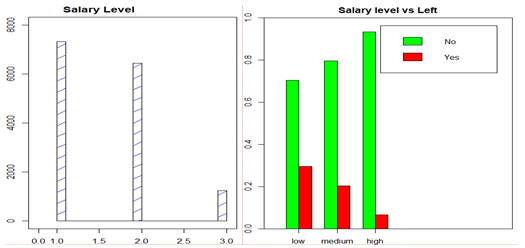
* 1. Promotion in last 5 years





* 1. Salary



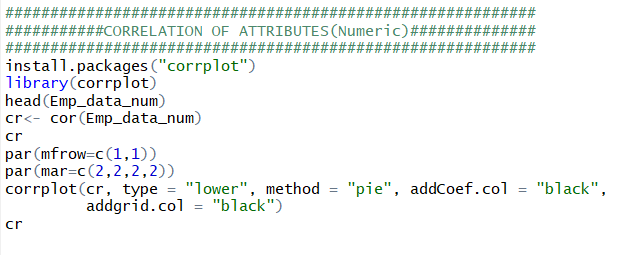


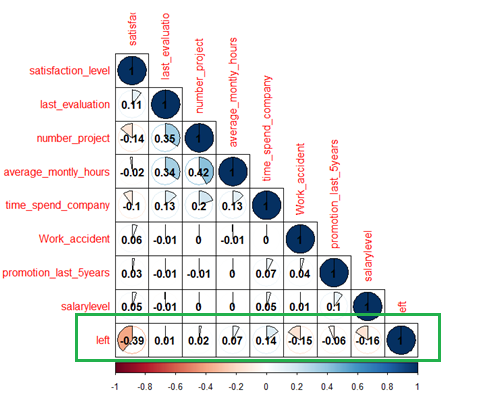
Now, we have a fair idea of what our attributes are and how they are related to our target variable.

### Correlation:

Now, to find the correlation of these attributes with our target variable, we perform the following functions on our data set.

* 1. To find the correlation between all the numeric attributes, we use the package “corrplot”

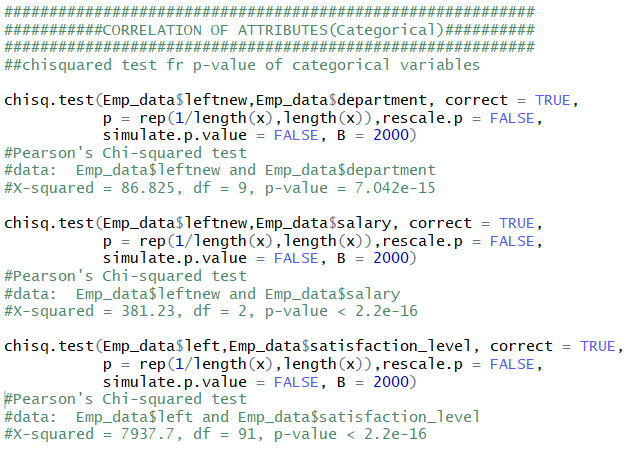




Thus we see, that most related value to our target variable is “satisfaction”, which has the inverse relationship with the target. Lesser the satisfaction level, more the probability of employee to leave.

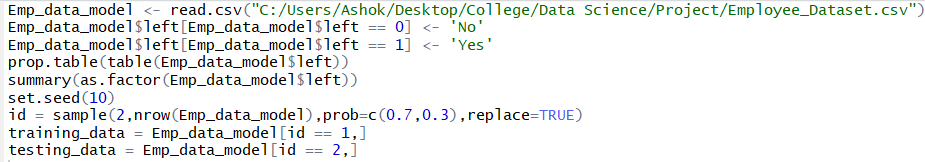
Next comes “salarylevel” and so on.

* 1. To find the correlation between categorical attributes, we use the chi-squared-test. Greater the X-squared value, higher the correlation.



### Splitting the data

The simplest way for us to get a handle on the ability of a predictive model to perform on future data is to try to simulate this eventuality. Although we cannot literally gain access to the future before it occurs we can reserve some of our currently available data and treat it as if were data from the future. Hence, we split our existing data into training (70%) and testing (30%) data.



Two datasets are created with below specifications:



### Classification and Prediction Models:

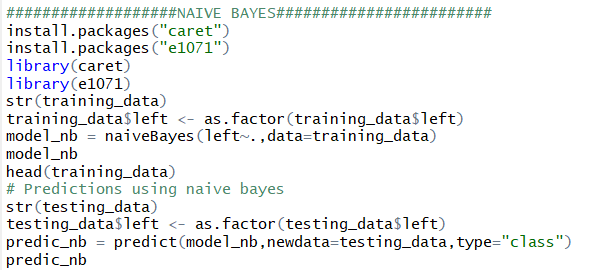
Now, we create the classification models for prediction.

* 1. Naïve Bayes

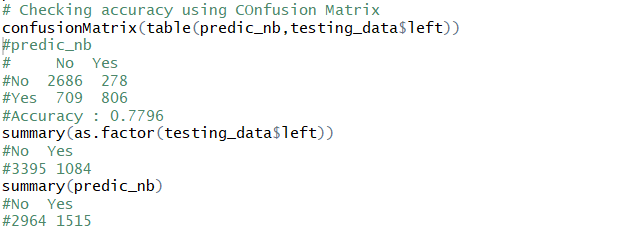
Naive Bayes classifier is a straightforward and powerful algorithm for the classification task. Even if we are working on a data set with millions of records with some attributes, it is suggested to try Naive Bayes approach.

To understand the naive Bayes classifier we need to understand the Bayes theorem.

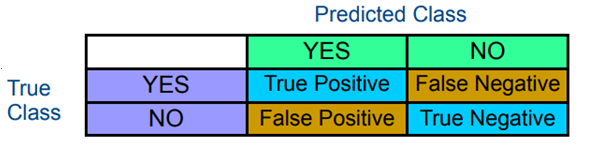
Naive Bayes is a kind of classifier which uses the Bayes Theorem. It predicts membership probabilities for each class such as the probability that given record or data point belongs to a particular class. This models assumes that outcome of all the attributes is independence of each other.



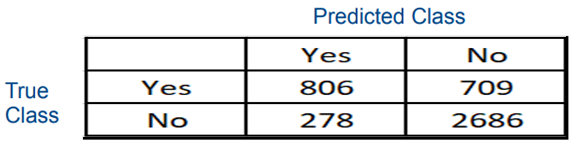
To check the accuracy of our prediction, we use confusion matrix.



As we know the structure of confusion matrix is



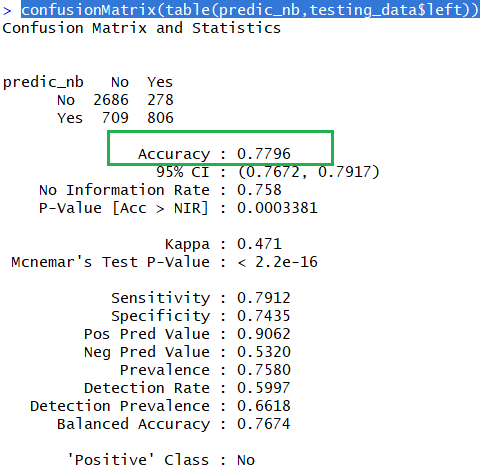
Inputting our values here, we get the below accuracy:



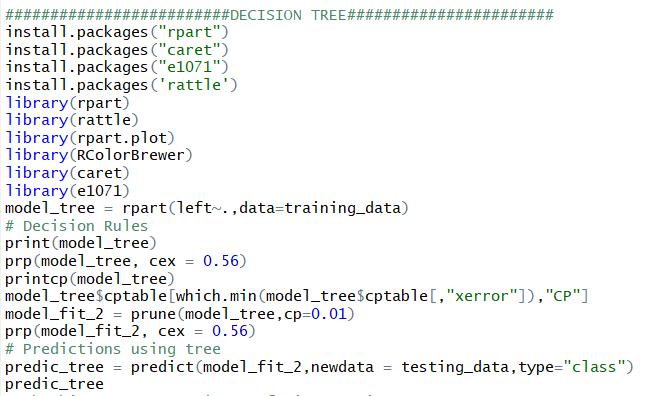
**Accuracy = 77.96%**

The function confusionMatrix(table(predic\_nb,testing\_data$left))

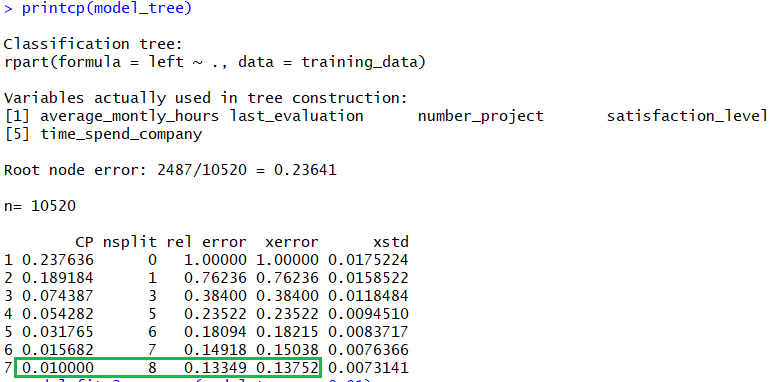
also provides its statistics, as below, confirming our model accuracy.



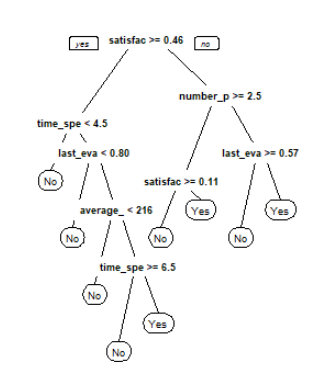
* 1. Decision Tree



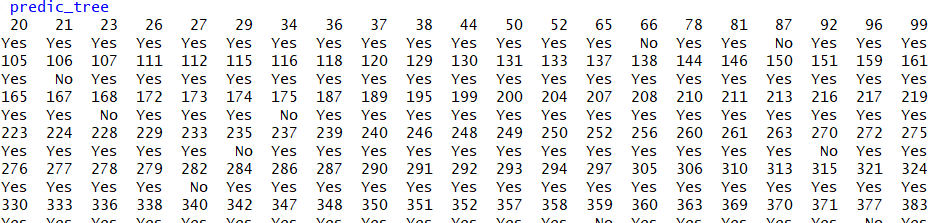
Rpart() function helps us create a decision tree model. This model is pruned to create the best fit model which here is model\_fit\_2. The pruning is done on the basis of complexity parameter(cp). The complexity parameter (cp) is used to control the size of the decision tree and to select the optimal tree size. If the cost of adding another variable to the decision tree from the current node is above the value of cp, then tree building does not continue. We could also say that tree construction does not continue unless it would decrease the overall lack of fit by a factor of cp. The optimal value of cp is the value for which the cross error is zero, as below.



The decision tree created is as below:



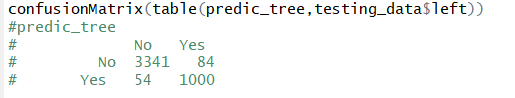
Based on the decision tress, the prediction result on testing dataset is as below:



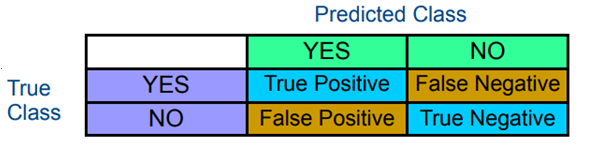
We can see the summary of the above predictions as well.



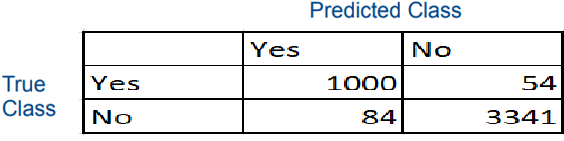
We can check the accuracy of this model using confusion matrix:



As we know the structure of confusion matrix is

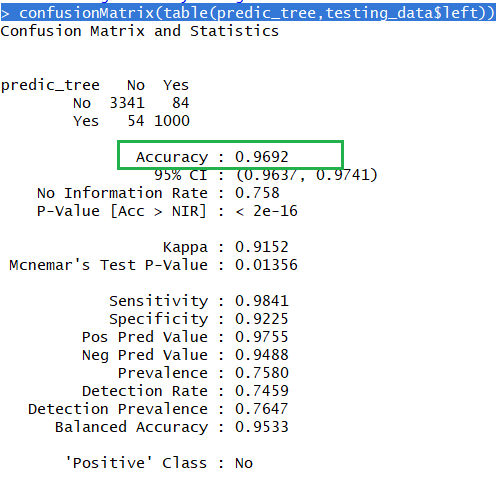


Inputting our values here, we get the below accuracy:



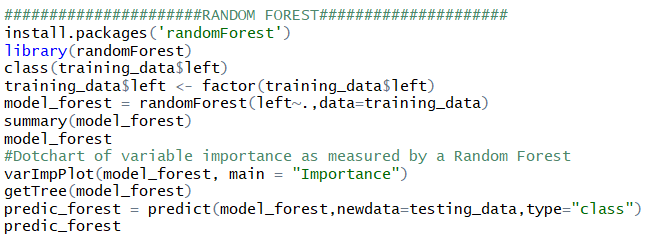
**Accuracy = 96.92%**

The function confusionMatrix(table(predic\_tree,testing\_data$left)) also provides it statistics as below:

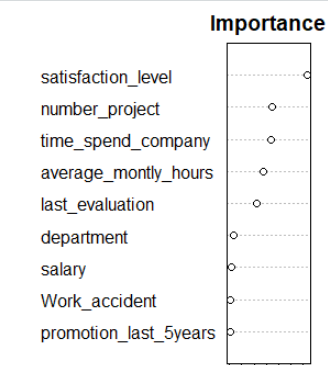


* 1. Random Forest

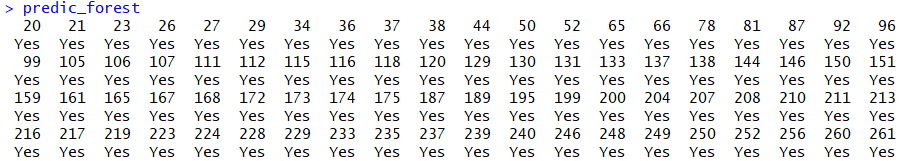
Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.



Using the function varImpPlot(model\_forest, main = "Importance"), we can check the importance of the attributes random forest considers while making the prediction.



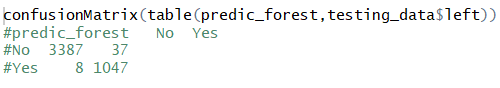
Based on the random forest, the prediction result on testing dataset is as below:



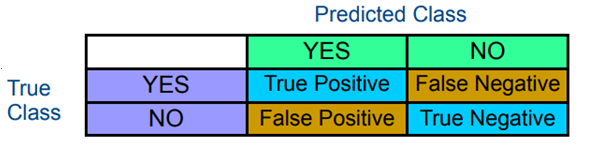
We can see the summary of the above predictions as well.



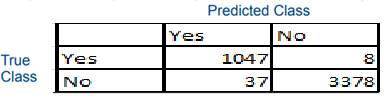
We can check the accuracy of this model using confusion matrix:



As we know the structure of confusion matrix is

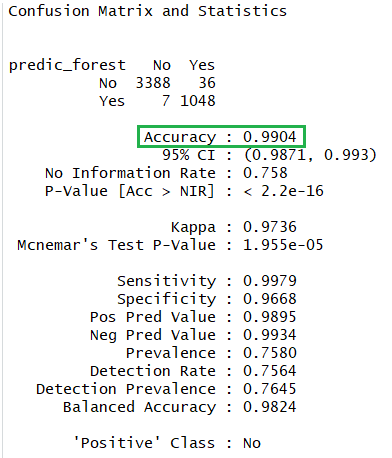


Inputting our values here, we get the below accuracy:



**Accuracy = 99.04%**

The function confusionMatrix(table(predic\_forest,testing\_data$left)) also provides it statistics as below:



Thus we see, for our dataset, Random Forest model provides the maximum accuracy.

## How to Download RStudio:

Windows Users:

To Install R:

1. Open an internet browser and go to [www.r-project.org](http://www.r-project.org/).
2. Click the "download R" link in the middle of the page under "Getting Started."
3. Select a CRAN location (a mirror site) and click the corresponding link.
4. Click on the "Download R for Windows" link at the top of the page.
5. Click on the "install R for the first time" link at the top of the page.
6. Click "Download R for Windows" and save the executable file somewhere on your computer. Run the .exe file and follow the installation instructions.

Now that R is installed, you need to download and install RStudio.

To Install RStudio:

1. Go to [www.rstudio.com](http://www.rstudio.com/)  and click on the "Download RStudio" button.
2. Click on "Download RStudio Desktop."
3. Click on the version recommended for your system, or the latest Windows version, and save the executable file. Run the .exe file and follow the installation instructions.

Mac Users:

To Install R:

1. Open an internet browser and go to [www.r-project.org](http://www.r-project.org/).
2. Click the "download R" link in the middle of the page under "Getting Started."
3. Select a CRAN location (a mirror site) and click the corresponding link.
4. Click on the "Download R for (Mac) OS X" link at the top of the page.
5. Click on the file containing the latest version of R under "Files."
6. Save the .pkg file, double-click it to open, and follow the installation instructions.

Now that R is installed, you need to download and install RStudio.

To Install RStudio:

1. Go to [www.rstudio.com](http://www.rstudio.com/)  and click on the "Download RStudio" button.
2. Click on "Download RStudio Desktop."
3. Click on the version recommended for your system, or the latest Mac version, save the .dmg file on your computer, double-click it to open, and then drag and drop it to your applications folder.

## References:

1. How the Naive Bayes Classifier works in Machine Learning. (2017, February 19). Retrieved December 20, 2017, from <http://dataaspirant.com/2017/02/06/naive-bayes-classifier-machine-learning/>
2. Steinberg, D. (n.d.). Why Data Scientists Split Data into Train and Test. Retrieved December 20, 2017, from <https://info.salford-systems.com/blog/bid/337783/Why-Data-Scientists-Split-Data-into-Train-and-Test>
3. Ray, S., Bansal, S., Srivastava, P., & Gupta, D. (2017, September 13). Understanding Support Vector Machine algorithm from examples (along with code). Retrieved December 20, 2017, from <https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>
4. UTAustinX: UT.7.01x Foundations of Data Analysis. (n.d.). Retrieved December 20, 2017, from <https://courses.edx.org/courses/UTAustinX/UT.7.01x/3T2014/56c5437b88fa43cf828bff5371c6a924/>
5. Random forest. (2017, December 13). Retrieved December 20, 2017, from <https://en.wikipedia.org/wiki/Random_forest>