PROJECT REPORT

(Group Project on Python)

CREDIT RISK

SUBMITTED TOWARDS THE PARTIAL FULFILLMENT OF THE CRITERIA FOR AWARD OF GENPACT DATA SCIENCE PRODEGREE BY IMARTICUS

Submitted By:

MR. RAHUL MEHTA
MR. MANOJ KHILARI
MR. HITEN PATEL
MR. OMKAR KUNDE

COURSE AND BATCH: DSP -14 2018

ABSTRACT

People often save their money in the banks which offer security but with lower interest rates. Lending Club operates an online lending platform that enables borrowers to obtain a loan, and investors to purchase notes backed by payments made on loans. It is transforming the banking system to make credit more affordable and investing more rewarding. But this comes with a high risk of borrowers defaulting the loans. Hence there is a need to classify each borrower as defaulter or not using the data collected when the loan has been given.

ACKNOWLEDGEMENTS

We are using this opportunity to express my gratitude to everyone who supported us throughout the course of this group project. We are thankful for their aspiring guidance, invaluably constructive criticism and friendly advice during the project work. I am sincerely grateful to them for sharing their truthful and illuminating views on a number of issues related to the project.

She/he has readily shared his immense knowledge in data analytics and guide us in a manner that the outcome resulted in enhancing our data skills.

We wish to thank all the faculties, as this project utilized knowledge gained from every course that formed the DSP program.

We certify that the work done by us for conceptualizing and completing this project is original and authentic.

Mr. Hiten Patel
Mr. Rahul Mehta
Mr. Manoj Khilari
Mr. Omkar Kunde

CERTIFICATE OF COMPLETION

I hereby certify that the project titled **" Credit Risk "** was undertaken and completed under my supervision by **Mr. Rahul Mehta , Mr. Manoj Khilari , Mr. Hiten Patel** and **Mr. Omkar Kunde** from the batch of DSP 14 (Jul 2018)

Date: September 11, 2018
Place – Mumbai

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 TITLE & OBJECTIVE OF THE STUDY

Title of the project is XYZ Corporation Lending Data Project. In this project, we work on the simpler problem that is to predict loan defaults. To classify if the borrower will default the loan using borrower's finance history. That means, given a set of new predictor variables, we need to predict the target variable as 1 -> Defaulter or 0 -> Non-Defaulter.

1.2 NEED OF THE STUDY

In this project, ourpropose to predict whether a borrower will default or not, so that investors can avoid those borrowers using manual investing feature provided by lending club. This, however, does not necessarily lead to highest return on investment (ROI) because by completely avoiding potential defaults, one also avoid riskier loans that may lead to higher ROI even though they default at some point in the future. In order to maximize ROI, one needs to optimize ROI instead. In this project, we work on the simpler problem that is to predict loan defaults.

1.3 BUSINESS OR ENTERPRISE UNDER STUDY

XYZ Corporation Lending Data is under the study. Data of Loans issued by XYZ Corp. through 2007-2015 is used for analysis. The data contains the indicator of default, payment information, credit history, etc.

1.4 BUSINESS MODEL OF ENTERPRISE

Selecting the relevant variables from the dataset and arranging their values in order of importance to create models to predict the probability of default of an individual in the future by performing different types of algorithms on the data.

1.5 DATA SOURCES

XYZ _Corp_Lending Data

Data contains the information about the status of the loan defaulter. The dataset contains the information like age, gender, annual income, grade of the customer paying capacity.

Data Set Description:

Contains 855969 rows and 73 columns

1.6 TOOLS & TECHNIQUES

Tools: Anaconda Navigator Spyder version 3.2.8

Techniques:

Logistic Regression

In regression Applied logistic regression algorithm in managing the credit risk to predict loan default. Analysis, logistic regression or logit regression is estimating the parameters of a logistic model. More formally, a logistic model is one where the log-odds of the probability of an event is a linear combination of independent or predictor variables. The two possible dependent variable values are often labeled as "0" and "1", which represent outcomes such as pass/fail, win/lose, alive/dead or healthy/sick. The binary logistic regression model can be generalized to more than two levels of the dependent variable: categorical outputs with more than two values are modeled by multinomial logistic regression, and if the multiple categories are ordered, by ordinal logistic regression, for example the proportional odds ordinal logistic model.

Extra-trees classifier

This class implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting.

1.7 INFRASTRUCTURE CHALLENGES

- Inefficient data management.
- Limited group wide risk modeling infrastructure
- Lacking risk tools
- Less than intuitive reporting and visualization

CHAPTER 2

DATA PREPARATION AND UNDERSTANDING

One of the first steps we engaged in was to outline the sequence of steps that we will be following for our project. Each of these steps are elaborated below

2.1 PHASE I – DATA EXTRACTION AND CLEANING:

2.1.1 MISSING VALUE ANALYSIS AND TREATMENT

Finding the missing value in the data with the isnull.sum () function

The dataset consists of around 73 features out of which there are a few features which consists missing values. By setting a threshold of 50%, if more than 50% or records are null, then they have been dropped.

The following are a few techniques for handling missing values:

· Forward Fill

· Backward Fill

· Replace by Mean value

· Replace by Maximum occurring value

· Drop the record

- To check Missing Values and its output:

```
application_type                 0
annual_inc_joint            855527
dti_joint                   855529
verification_status_joint   855527
acc_now_delinq                   0
tot_coll_amt                 67313
tot_cur_bal                  67313
open_acc_6m                 842681
open_il_6m                  842681
open_il_12m                 842681
open_il_24m                 842681
mths_since_rcnt_il          843035
total_bal_il                842681
il_util                     844360
open_rv_12m                 842681
open_rv_24m                 842681
max_bal_bc                  842681
all_util                    842681
total_rev_hi_lim             67313
inq_fi                      842681
total_cu_tl                 842681
inq_last_12m                842681
default_ind                      0
Length: 73, dtype: int64
```

Figure 1: Missing value in Data

- To remove Columns with more than 50% missing values

```
Console 1/A

In [41]: half_count=len(df)/2
    ...: df1=df.dropna(thresh=half_count,axis=1)# del va
values as na
    ...: print(df1.isnull().sum())
id                                0
member_id                         0
loan_amnt                         0
funded_amnt                       0
funded_amnt_inv                   0
term                              0
int_rate                          0
installment                       0
grade                             0
sub_grade                         0
emp_title                     49443
emp_length                    43061
home_ownership                    0
annual_inc                        0
verification_status               0
issue_d                           0
pymnt_plan                        0
purpose                           0
title                            33
zip_code                          0
addr_state                        0
dti                               0
delinq_2yrs                       0
earliest_cr_line                  0
inq_last_6mths                    0
```

Figure 2: Remove columns Missing value

2.1.2 FEATURE EXTRACTION

In financial risk, credit risk management is one of the most important issues in financial decision-making. Reliable credit scoring models are crucial for financial agencies to evaluate credit applications and have been widely studied in the field of machine learning and statistics. Deep learning is a powerful classification tool which is currently an active research area and successfully solves classification problems in many domains.

Deep Learning provides training stability, generalization, and scalability with big data. Deep Learning is quickly becoming the algorithm of choice for the highest predictive accuracy. Feature selection is a process of selecting a subset of relevant features, which can decrease the dimensionality, reduce the running time, and improve the accuracy of classifiers. In this study, we constructed a credit scoring model based on deep learning and feature selection to evaluate the applicant's credit score from the applicant's input features. Two public datasets, Australia and German credit ones have been used to test our method.

The experimental results of the real world data showed that the proposed method results in a higher prediction rate than a baseline method for some certain datasets and also shows comparable and sometimes better performance than the feature selection methods widely used in credit scoring.

2.2 PHASE II - FEATURE ENGINEERING

A **credit risk** is the **risk** of default on a debt that may arise from a borrower failing to make required payments. The goal of **credit risk** evaluation is to estimate the probability that a borrower will default on a debt.

2.3 DATA PROCESSING

In this section, some steps of data preprocessing are documented.

Convert ordinal variables to numeric variables

```
#%%
df1.grade.value_counts()
grade_final={'A':6,'B':5,'C':4,'D':3,'E':2,'F':1,'G':0}
df1.grade=[grade_final[item]for item in df1.grade]
print(df1.grade)
#%%
```

Combine similar classes in categorical variables and convert to numeric variables


```
#%%
print(df1['application_type'].unique())
App_type={'INDIVIDUAL':1,'JOINT':2}
df1.application_type=[App_type[item] for item in df1.application_type]
print(df1.application_type)
#%%
```

Extract the dtring and apply numerical variable

```
#%%
df1.term.value_counts()
df1.term=df1.term.str.extract('(\d+)')
term_final={'36':0,'60':1}
df1.term=[term_final[item]for item in df1.term]
print(df1.term)
```

Extract the string and apply range

```
#%%
df1['earliest_cr_line'].nunique()
df1.earliest_cr_line=df1.earliest_cr_line.str.extract('(\d+)')
print(df1.earliest_cr_line)
df1.earliest_cr_line = [int(x) for x in df1.earliest_cr_line]
print(df1.earliest_cr_line.dtype)
#%%
def earliest_cr_line_final(i):
    if i in range(44,79):
        return(0)
    elif i in range(80,90):
        return(1)
    elif i in range(90,0):
        return(2)
    else:
        return(3)
```

Dropping Columns


```
print(df1.last_pymnt_d)
#%%
df2=pd.DataFrame(df1)
df2.drop(['member_id','funded_amnt','addr_state','emp_title','sub_grade','inq_last_6mths','funded_amnt_inv','py
df2.shape
df2.isnull().sum()
```

2.4 EXPLORATORY DAT ANALYSIS:

We can see the relationship between grade and Default _ind

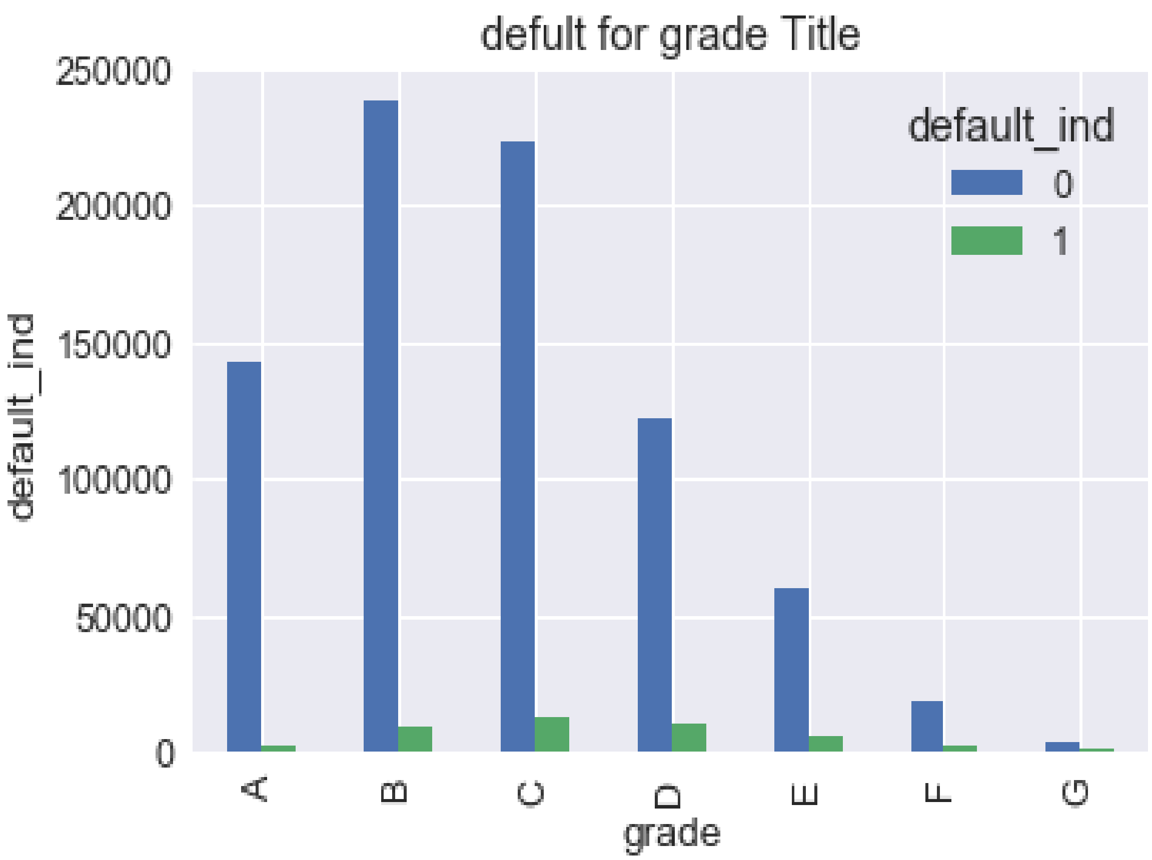

Figure 3: Grade relationship with Default

We can see the relationship between Employments length and default credit or not on the basis of Corp_Lending Data.

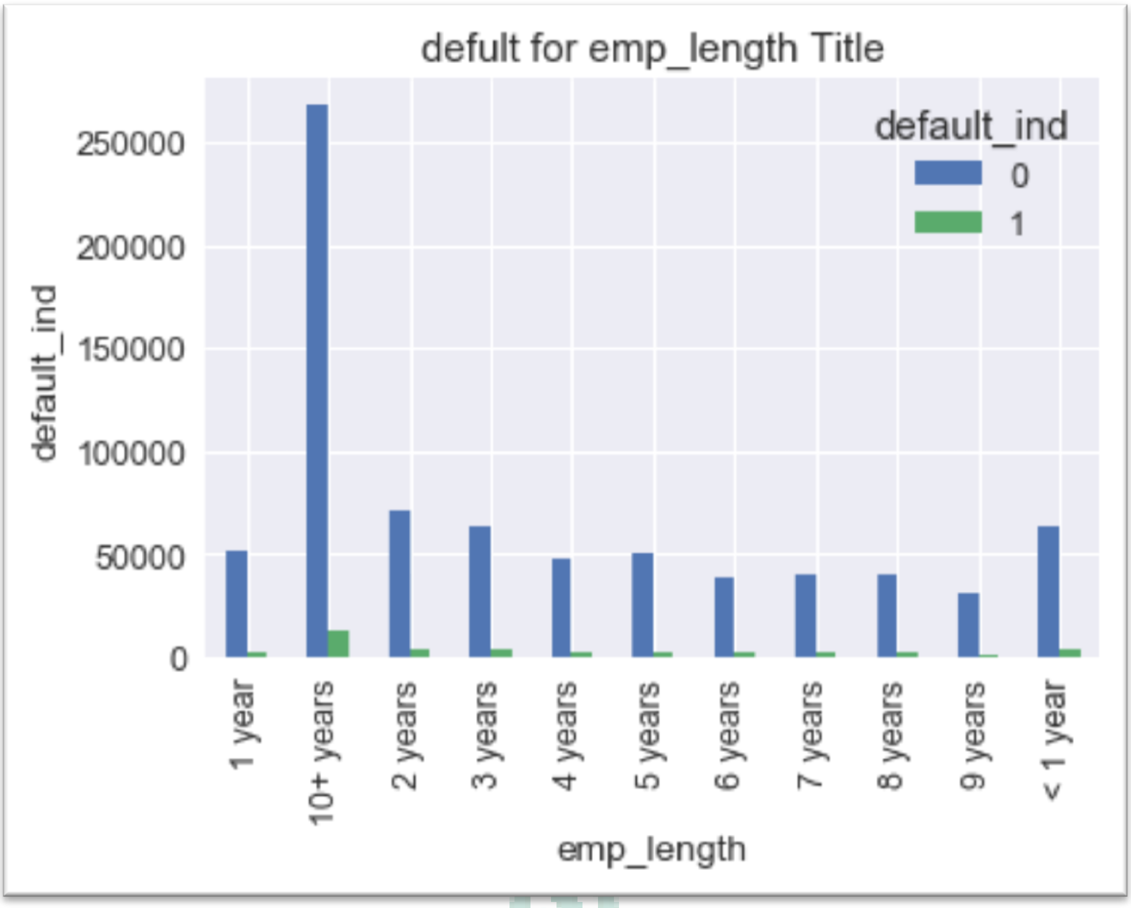

Figure4:emp_length vs. Default_ind

We can see the relationship between purpose and default Credit or not on the basis of Corp_Lending Data.

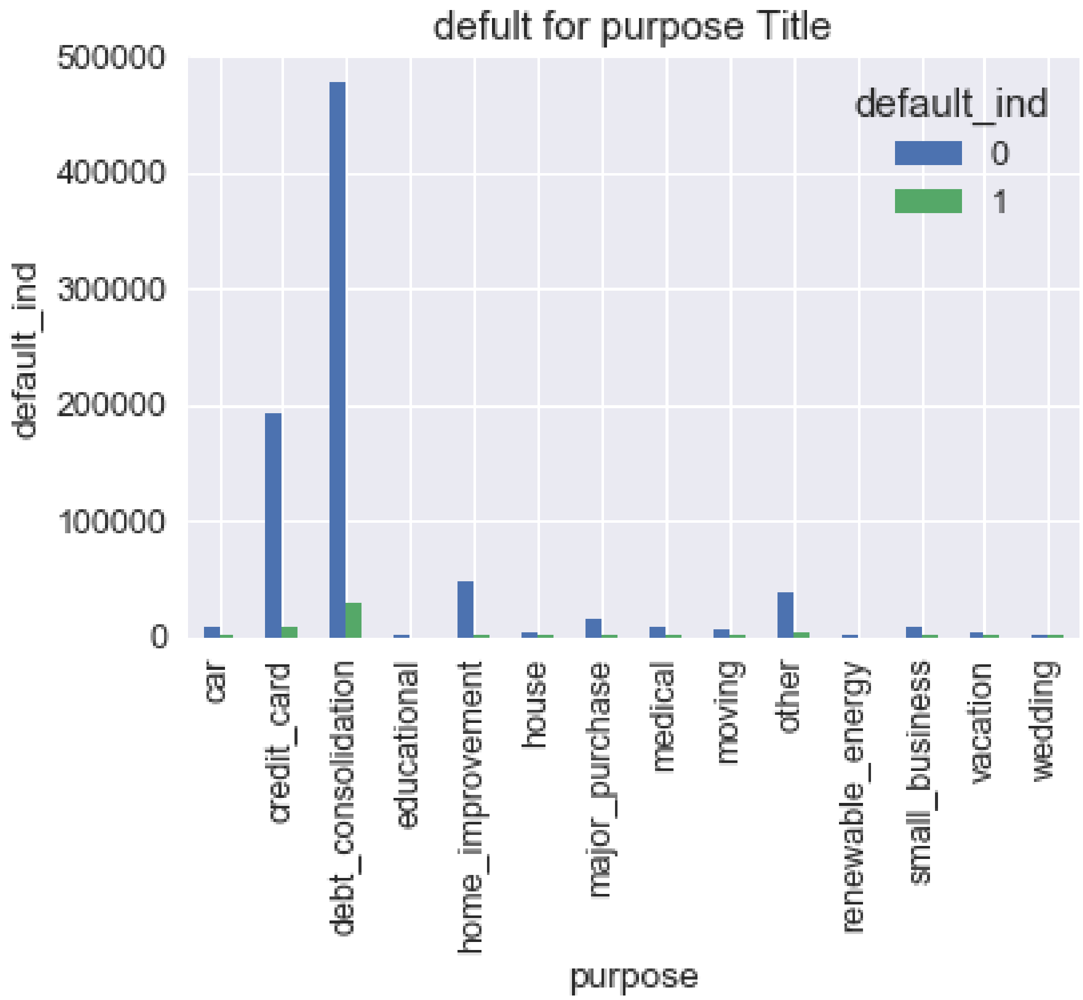

Figure 5: purpose of with loan

CHAPTER 3

FITTING MODELS TO DATA

3.1 CHECKING FOR VARIANCE INFLATION (MULTICOLLINEARITY)

First checking multicollinearity in data long correlation coefficients among independent variables are less than 0.90 the assumption is met.it keep and above it remove .It will check heat map function graph.

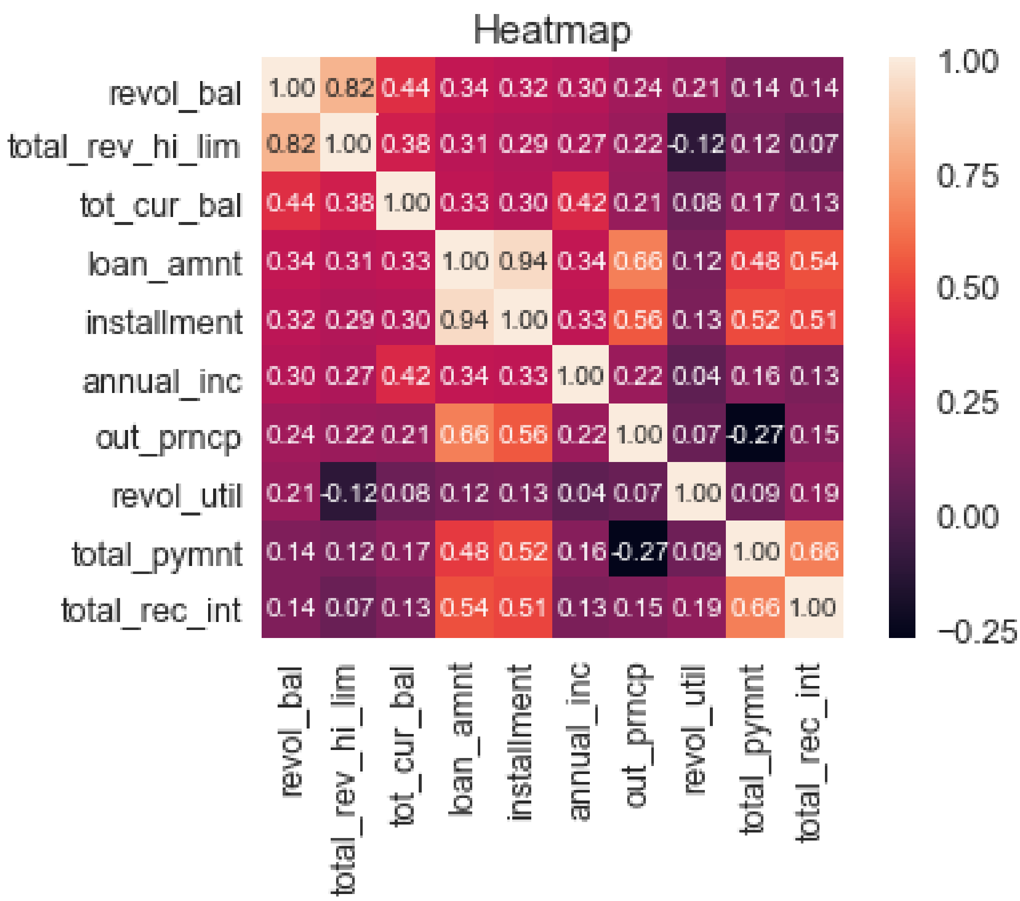

Figure 6 : Heatmap

There should be no high correlations (multicollinearity) among the predictors. This can be assessed by a correlation matrix among the predictors. Tabachnick and Fidell (2013) suggest that as long correlation coefficients among independent variables are less than 0.90 the assumption is met.

3.2. MODELS APPLIED AND MOTIVATION

Logistic Regression: With logistic regression, outputs have a nice probabilistic interpretation, and the algorithm can be regularized to avoid overfitting. Hence, we choose to build logistic regression classifier.

Extra-trees classifier: This class implements a Meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting.

3.3. SPLITTING OF THE DATA

```
#%%
df3['issue_d'].head()
test_data=df3[df3['issue_d'].isin(['Jun-15', 'Jul-15','Aug-15','Sep-15','Oct-15','Nov-15','Dec-15'])]
train_data=df3[df3['issue_d'].isin(['Jun-15', 'Jul-15','Aug-15','Sep-15','Oct-15','Nov-15','Dec-15'])==False]
#%%
train_data=train_data.drop('issue_d',axis=1)
test_data=test_data.drop('issue_d',axis=1)
#%%
#creating training and testing datasets
X_train=train_data.values[:,1:-1]
Y_train=train_data.values[:,-1]
#%%
X_test=test_data.values[:,1:-1]
Y_test=test_data.values[:,-1]
# In[25]:
```

CHAPTER 4

CLASSIFICATION MODEL

4.1 LOGISTIC REGRESSION MODEL

Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variable.

Type of questions that a binary logistic regression can examine.

How does the probability of getting lung cancer (yes vs. no) change for every additional pound a person is overweight and for every pack of cigarettes smoked per day?

Do body weight, calorie intake, fat intake, and age have an influence on the probability of having a heart attack (yes vs. no)?

Binary logistic regression major assumptions:

The dependent variable should be dichotomous in nature (e.g., presence vs. absent).

There should be no outliers in the data, which can be assessed by converting the continuous predictors to standardized scores, and removing values below -3.29 or greater than 3.29.

There should be no high correlations (multicollinearity) among the predictors. This can be assessed by a correlation matrix among the predictors. Tabachnick and Fidell (2013) suggest that as long correlation coefficients among independent variables are less than 0.90 the assumption is met.

At the center of the logistic regression analysis is the task estimating the log odds of an event. Mathematically, logistic regression estimates a multiple linear regression function defined as:

logit(p)

$$= \log\left(\frac{p(y=1)}{1-(p=1)}\right) = \beta_0 + \beta_1 \cdot x_{i1} + \beta_2 \cdot x_{i2} + \_ + \beta_p \cdot x_{in}$$

Figure 7: Logistic Equation

The logistic function is defined as:

transformed = 1 / (1 + e^-x)

for i = 1…n .

Overfitting: - When selecting the model for the logistic regression analysis, another important consideration is the model fit. Adding independent variables to a logistic regression model will always increase the amount of variance explained in the log odds. However, adding more and more variables to the model can result in over fitting, which reduces the generalizability of the model beyond the data on which the model is fit

4.2. LOGISTIC REGRESSION MODEL

4.2.1. PERFORMANCE OF LOGISTIC REGRESSION MODEL

To evaluate the performance of a logistic regression model, we must consider few metrics. Irrespective of tool (SAS, R, Python) you would work on, always look for:

1. Null Deviance and Residual Deviance – Null Deviance indicates the response predicted by a model with nothing but an intercept. Lower the value, better the model. Residual deviance indicates the response predicted by a model on adding independent variables. Lower the value, better the model.

2. Confusion Matrix: It is nothing but a tabular representation of Actual vs. Predicted values. This helps us to find the accuracy of the model and avoid overfitting. This is how it looks like:

		Predicted	
		Good	Bad
Actual	Good	True Positive (d)	False Negative (c)
	Bad	False Positive (b)	True Negative (a)

Figure 8: Confutation Matrix Representation

You can calculate the accuracy **of your model with:**

$$\frac{\text{True Positive} + \text{True Negatives}}{\text{True Positive} + \text{True Negatives} + \text{False Positives} + \text{False Negatives}}$$

From confusion matrix, Specificity and Sensitivity can be derived as illustrated below:

$$\left.\begin{aligned} \text{True Negative Rate (TNR), specificity} &= \frac{A}{A+B} \\ \text{False Positve Rate (FPR), } 1-\text{specificity} &= \frac{B}{A+B} \end{aligned}\right\}\text{sum to 1}$$

$$\left.\begin{aligned} \text{True Positive Rate (TPR), sensitivity} &= \frac{D}{C+D} \\ \text{False Negative Rate (FNR)} &= \frac{C}{C+D} \end{aligned}\right\}\text{sum to 1}$$

4.2.2. CONFUSION MATRIX MODEL LOGISTIC REGRESSION

```
In [58]: from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
    ...: cfm=confusion_matrix(Y_test,Y_pred)
    ...: print(cfm)
    ...: print("Classification report:")
    ...: print(classification_report(Y_test,Y_pred))
    ...: acc=accuracy_score(Y_test,Y_pred)
    ...: print("Accuracy of the model:",acc)
[[256639     34]
 [    63    248]]
Classification report:
             precision    recall  f1-score   support

        0.0       1.00      1.00      1.00    256673
        1.0       0.88      0.80      0.84       311

avg / total       1.00      1.00      1.00    256984

Accuracy of the model: 0.999622544594216
```

Figure 9: Confutation Matrix Logistic Regression

4.2.4. ROC CURVE LOGISTIC REGRESSION

Receiver Operating Characteristic (ROC) summarizes the model's performance by evaluating the tradeoffs between true positive rate (sensitivity) and false positive rate(1- specificity). For plotting ROC, it is advisable to assume $p > 0.5$ since we are more concerned about success rate. ROC summarizes the predictive power for all possible values of $p > 0.5$. The area under curve (AUC), referred to as index of accuracy(A) or concordance index, is a perfect performance metric for ROC curve. Higher the area under curve, better the prediction power of the model. Below is a sample ROC curve. The ROC of a perfect predictive model has TP equals 1 and FP equals 0. This curve will touch the top left corner of the graph.

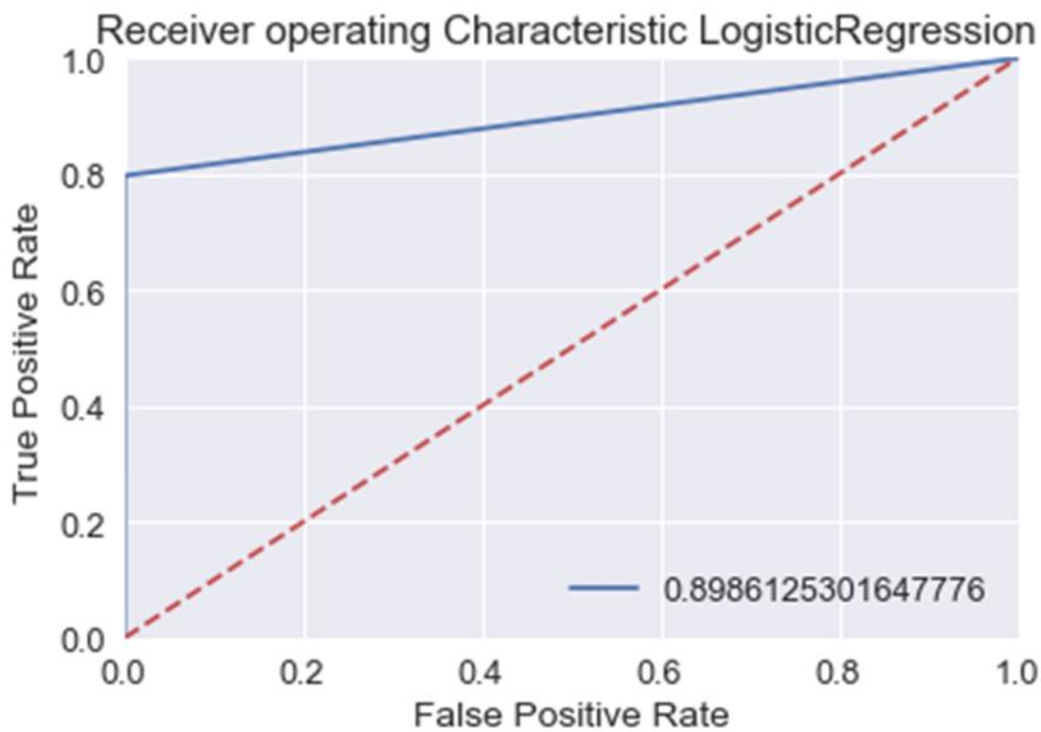

Figure 10: ROC Logistic Regression Model

4.3 EXTRA TREES CLASSIFIER MODEL

This class implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting.

```
#%%
# predicting using the extratreesclassifier
from sklearn.ensemble import ExtraTreesClassifier
model=(ExtraTreesClassifier(51))
#fit the model on data and predict values
model=model.fit(X_train,Y_train)
Y_pred=model.predict(X_test)
```

4.3.1. CONFUSION MATRIX MODEL EXTRA TREES CLASSIFIER

```
   ...: print(classification_report(Y_test,Y_pred))
[[189197  67476]
 [    23    288]]
0.7373416243812845
             precision    recall  f1-score   support

        0.0       1.00      0.74      0.85    256673
        1.0       0.00      0.93      0.01       311

avg / total       1.00      0.74      0.85    256984
```

Figure 11: Confutation Matrix Extra Trees Classifier

4.3.2. ROC CURVE EXTRA TREES CLASSIFIER

Figure 12: ROC Logistic Extra Trees Classifier

CHAPTER 5

KEY FINDINGS

Significant Variables identified in logistic models are also used in Random forest Below table provides a snapshot of the various models which the business can choose from based on the pros and cons of each model.

Below are some of the key findings

Sr. No	Model Name	Accuracy	Type1 Error	Type2 Error
1	Logistic Regression Model	0.99	34	63
2	Extratreesclassifier Model	0.73	67476	23

CHAPTER 6

RECOMMENDATIONS AND CONCLUSION

We recommend that if type 1 error is low then use logistic regression model & If type 2 error is low but type 1 error is high use extra trees classifier model. Here we visualize the ROC curve for the final model. The x-axis for the ROC curve is FPR (False Positive) rate and the y-axis is TPR (True positive) rate. The plot shows that the model prediction is better than random guess, which is the diagonal line running from (0, 0) to (1, 1).

CHAPTER 7

REFERENCES

https://stackoverflow.com

https://www.investopedia.com

http://scikit-learn.org

http://seaborn.pydata.org/