 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

Aim: To understand the logistic Regression that includes non-linearity to linear regression

IDE: Google Colab

Theory:

Within machine learning, logistic regression belongs to the family of supervised machine learning models. It is also considered a discriminative model, which means that it attempts to distinguish between classes (or categories). Unlike a generative algorithm, such as naïve bayes, it cannot, as the name implies, generate information, such as an image, of the class that it is trying to predict (e.g. a picture of a cat). This type of statistical model (also known as logit model) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables. Since the outcome is a probability, the dependent variable is bounded between 0 and 1. In logistic regression, a logit transformation is applied on the odds—that is, the probability of success divided by the probability of failure. This is also commonly known as the log odds, or the natural logarithm of odds, and this logistic function is represented by the following formulas:

$$\text{Logit}(\pi) = 1/(1 + \exp(-\pi))$$


$$\ln(\pi/(1-\pi)) = \beta_0 + \beta_1 X_1 + \dots + \beta_k X_k$$

In this logistic regression equation, $\text{logit}(\pi)$ is the dependent or response variable and x is the independent variable. The beta parameter, or coefficient, in this model is commonly estimated via maximum likelihood estimation (MLE). This method tests different values of beta through multiple iterations to optimize for the best fit of log odds. All of these iterations produce the log likelihood function, and logistic regression seeks to maximize this function to find the best parameter estimate. Once the optimal coefficient (or coefficients if there is more than one independent variable) is found, the conditional probabilities for each observation can be calculated, logged, and summed together to yield a predicted probability. For binary classification, a probability less than .5 will predict 0 while a probability greater than 0 will predict 1. After the model has been computed, it's best practice to evaluate the how well the model predicts the dependent variable, which is called goodness of fit.

Logistic regression can also be prone to overfitting, particularly when there is a high number of predictor variables within the model. Regularization is typically used to penalize parameters large coefficients when the model suffers from high dimensionality.

Linear Regression VS Logistic Regression:

Both linear and logistic regression are among the most popular models within data science, and open-source tools, like Python and R, make the computation for them quick and easy.

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

Linear regression models are used to identify the relationship between a continuous dependent variable and one or more independent variables. When there is only one independent variable and one dependent variable, it is known as simple linear regression, but as the number of independent variables increases, it is referred to as multiple linear regression. For each type of linear regression, it seeks to plot a line of best fit through a set of data points, which is typically calculated using the least squares method.


Similar to linear regression, logistic regression is also used to estimate the relationship between a dependent variable and one or more independent variables, but it is used to make a prediction about a categorical variable versus a continuous one. A categorical variable can be true or false, yes or no, 1 or 0, et cetera. The unit of measure also differs from linear regression as it produces a probability, but the logit function transforms the S-curve into a straight line.

While both models are used in regression analysis to make predictions about future outcomes, linear regression is typically easier to understand. Linear regression also does not require as large of a sample size as logistic regression needs an adequate sample to represent values across all the response categories. Without a larger, representative sample, the model may not have sufficient statistical power to detect a significant effect.

Types of Logistics Regression:

There are three types of logistic regression models, which are defined based on categorical response.

- 1. Binary logistic regression:** In this approach, the response or dependent variable is dichotomous in nature—i.e. it has only two possible outcomes (e.g. 0 or 1). Some popular examples of its use include predicting if an e-mail is spam or not spam or if a tumor is malignant or not malignant. Within logistic regression, this is the most commonly used approach, and more generally, it is one of the most common classifiers for binary classification.
- 2. Multinomial logistic regression:** In this type of logistic regression model, the dependent variable has three or more possible outcomes; however, these values have no specified order. For example, movie studios want to predict what genre of film a moviegoer is likely to see to market films more effectively. A multinomial logistic regression model can help the studio to determine the strength of influence a person's age, gender, and dating status may have on the type of film that they prefer. The studio can then orient an advertising campaign of a specific movie toward a group of people likely to go see it.
- 3. Ordinal logistic regression:** This type of logistic regression model is leveraged when the response variable has three or more possible outcomes, but in this case, these values do have a defined order. Examples of ordinal responses include grading scales from A to F or rating scales from 1 to 5.

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

Methodology:

1. Load the basic libraries and packages
2. Load the dataset
3. Analyze the dataset
4. Normalize the data
5. Pre-process the data
6. Visualize the Data
7. Separate the feature and prediction value columns
8. Write the Hypothesis Function
9. Write the Cost Function
10. Write the Gradient Descent optimization algorithm
11. Apply Feature Normalization technique over the data
12. Apply the training over the dataset to minimize the loss
13. Observe the cost function vs iterations learning curve

Program (Code):

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

```
from google.colab import drive
```

```
drive.mount('/content/drive')
```

```
cd /content/drive//content/ex2data1 (1).csv"
```

```
dataset = pd.read_csv("/content/ex2data1 (1).csv")
```


```
#add column in dataset
dataset.columns=["X","Y","Z"]
dataset
```

```
dataset.shape
```

```
dataset
```

```
dataset.describe()
```

```
x=dataset.iloc[:, :-1].values
y=dataset.iloc[:, -1].values
```

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

```
pos , neg = (y==1).reshape(100,1) , (y==0).reshape(100,1)
plt.scatter(x[pos[:,0],0],x[pos[:,0],1],c="r",marker="+")
plt.scatter(x[neg[:,0],0],x[neg[:,0],1],marker="o",s=10)
plt.xlabel("X")
plt.ylabel("Y")
plt.legend(["X","Y"],loc=0)
```

```
#hypothesis function
def sigmoid(p):
    return (1/(1+np.exp(-p)))
```


```
sigmoid(0)
```

```
def featurenormalization(x):
    mean=np.mean(x,axis=0)
    std=np.std(x,axis=0)
    x_norm=(x-mean)/std
    return x_norm, mean, std
```

```
#define cost function
def costfunction(theta,x,y):
    m=len(y)
    diff = 0
    predict=[]
    for i in range(m):
        predict_value=sigmoid(np.dot(x[i],theta))
        predict.append(predict_value)
        diff = diff+(-y[i]*np.log(predict_value)-((1-y[i])*np.log(1-predict_value)))
    cost=(1/m)*diff
    grad=(1/m)*np.dot(x.transpose(),(np.array(predict)-y))
    return cost,grad
```

```
def gradientdescent(theta,x,y,alpha,num_iters):
    cost_values=[]
    for i in range(num_iters):
        cost,grad=costfunction(theta,x,y)
        theta=theta-(alpha*grad)
        cost_values.append(cost)
    return theta,cost_values
```

```
#feature normalization block
m,n=x.shape[0],x.shape[1]
x,x_mean,x_std=featurenormalization(x)
```

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

```
x= np.append(np.ones((m,1)),x,axis=1)
```

```
y = y.reshape(m,1)
```

```
# y
```

```
#training process
```

```
initial_theta=np.zeros((n+1,1))
```

```
cost,grad=costfunction(initial_theta,x,y)
```

```
grad
```

```
theta,cost_values=gradientdescent(initial_theta,x,y,0.55,500)
```

```
plt.plot(cost_values)
```

```
plt.xlabel("Iterations")
```

```
plt.ylabel("cost value")
```

```
plt.title("Cost function curve")
```

```
plt.scatter(x[pos[:,0],1],x[pos[:,0],2],c="r",marker="+",label="True")
```

```
plt.scatter(x[neg[:,0],1],x[neg[:,0],2],c="b",marker="o",label="False")
```

```
x_value= np.array([np.min(x[:,1]),np.max(x[:,1])])
```

```
y_value=-(theta[0] +theta[1]*x_value)/theta[2]
```

```
plt.plot(x_value,y_value, "g")
```

```
plt.xlabel("X")
```


```
plt.ylabel("Y")
```

```
plt.legend(loc=0)
```

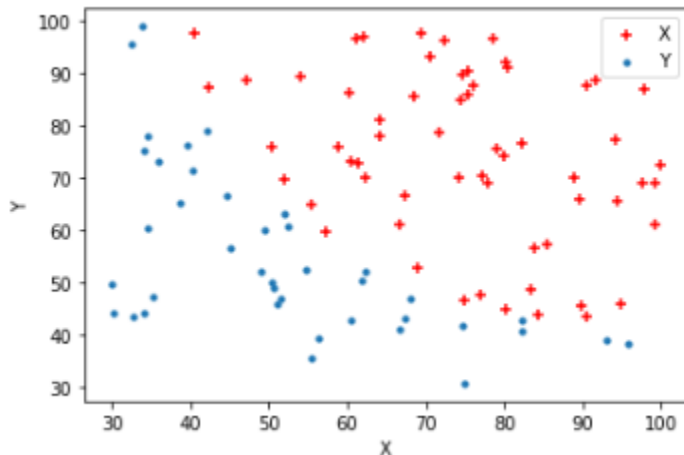
Results:

To be attached with

- a. Datapoints scattering

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018


<matplotlib.legend.Legend at 0x7f6eca66e5d0>



b. Data Statistics before Normalization

	X	Y	Z
count	100.000000	100.000000	100.000000
mean	65.644274	66.221998	0.600000
std	19.458222	18.582783	0.492366
min	30.058822	30.603263	0.000000
25%	50.919511	48.179205	0.000000
50%	67.032988	67.682381	1.000000
75%	80.212529	79.360605	1.000000
max	99.827858	98.869436	1.000000

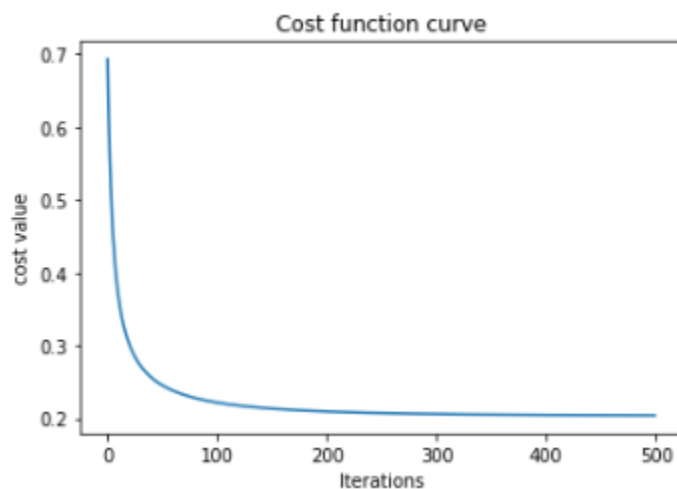
c. Data Statistics after Normalization

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

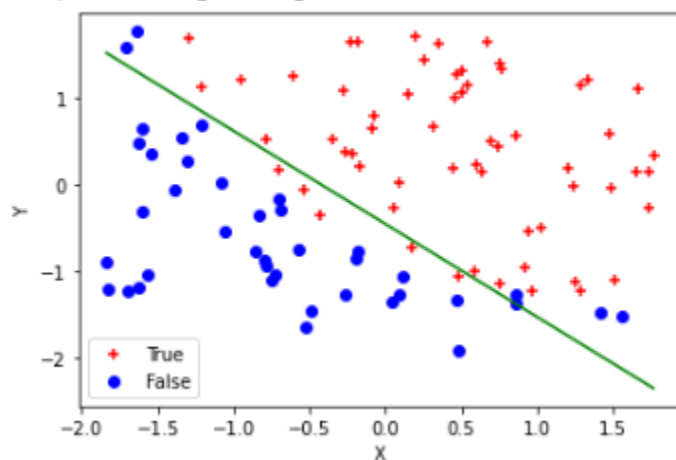
	X	Y	Z
count	100.000000	100.000000	100.000000
mean	65.644274	66.221998	0.600000
std	19.458222	18.582783	0.492366
min	30.058822	30.603263	0.000000
25%	50.919511	48.179205	0.000000
50%	67.032988	67.682381	1.000000
75%	80.212529	79.360605	1.000000
max	99.827858	98.869436	1.000000


d. Learning Curve (Cost function vs iterations)

```
Text(0.5, 1.0, 'Cost function curve')
```



```
<matplotlib.legend.Legend at 0x7f6eca090790>
```




 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

Observation and Result Analysis:

a. Nature of the dataset

b. During Training Process

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

c. After the training Process


d. Observation over the Learning Curve

Post Lab Exercise:

a. What is the difference between Linear Regression and Logistic Regression

b. What are the data challenges during model development?

c. What is the meaning of Maximum Likelihood?

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: Machine Learning (01CT0519)	Aim: To understand the logistic Regression that includes non-linearity to linear regression	
Experiment No: 04	Date:	Enrolment No:92000133018

d. What does odds-ratio signify?

e. How do you decide the cut-off for the output of logistic regression?

f. What are the key matrices used to check the performance of logistic regression?

g. How do you handle the missing values?

Post Lab Activity:

Consider any dataset from <https://archive.ics.uci.edu/ml/datasets.php> and perform the logistics regression. Make sure that the dataset is not matching with your classmates. You can also select the dataset from other ML repositories with prior permission from your concerned subject faculty.