**Experiment: 2.2**

**Student Name:** ASHISH CHETRY **UID:** 21BCS-3155

**Branch:** CSE **Section/Group:** 21BCS-IOT-602B

**Semester:** 5th **Date:** 28/09/23

**Subject Name**: AIML Lab **Subject Code:** 21CSH-316

1. **AIM:** Implementing Linear Regression and Logistic Regression models
2. **Objective:**

* To learn about different functions.
* To learn About Different Linear Regression Techniques.
* To Learn about Linear Regression Model or algorithms.

1. **Tools/Resource Used:**

1. Python programming language.

2. Jupyter Notebook.

1. **Description:**

**Problem Statement:**

The growth of supermarkets in most populated cities is increasing and market competitions are also high. The dataset is one of the historical sales of Supermarket Company which has recorded in 3 different branches for 3 months data. Predictive data analytics methods are easy to apply with this dataset.

1. **Program Code:**

import matplotlib.pyplot as plt

import numpy as np

from sklearn import datasets, linear\_model

from sklearn.metrics import mean\_squared\_error, r2\_score

# Load the diabetes dataset

diabetes = datasets.load\_diabetes()

# Use only one feature

diabetes\_X = diabetes.data[:, np.newaxis, 2]

# Split the data into training/testing sets

diabetes\_X\_train = diabetes\_X[:-20]

diabetes\_X\_test = diabetes\_X[-20:]

# Split the targets into training/testing sets

diabetes\_y\_train = diabetes.target[:-20]

diabetes\_y\_test = diabetes.target[-20:]

# Create linear regression object

regr = linear\_model.LinearRegression()

# Train the model using the training sets

regr.fit(diabetes\_X\_train, diabetes\_y\_train)

# Make predictions using the testing set

diabetes\_y\_pred = regr.predict(diabetes\_X\_test)

# The coefficients

print('Coefficients: \n', regr.coef\_)

# The mean squared error

print("Mean squared error: %.2f" % mean\_squared\_error(diabetes\_y\_test, diabetes\_y\_pred))

# Explained variance score: 1 is a perfect prediction

print('Variance score: %.2f' % r2\_score(diabetes\_y\_test, diabetes\_y\_pred))

# Plot outputs

plt.scatter(diabetes\_X\_test, diabetes\_y\_test, color='black')

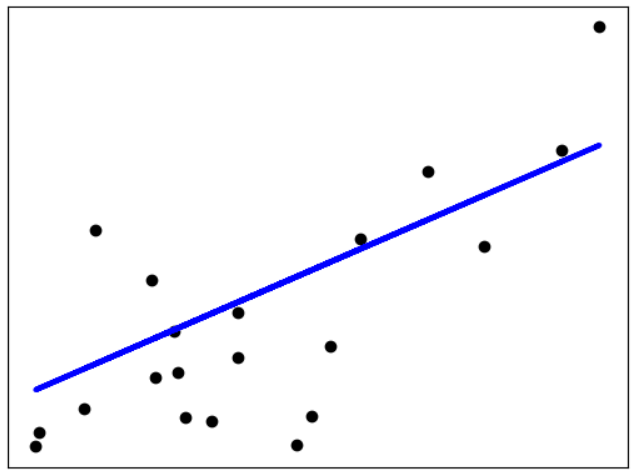
plt.plot(diabetes\_X\_test, diabetes\_y\_pred, color='blue', linewidth=3)

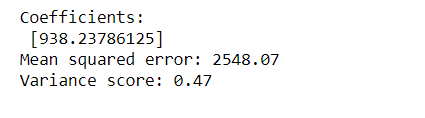
plt.xticks(())

plt.yticks(())

plt.show()

1. **Output/Result:**

****

****

1. **Learning Outcomes:**
2. Implement to implement different python library.
3. Understand the concept of numpy, pandas, SciPy library.
4. Understand the concept of linear regression