**Name = Nihar**

**Sap ID = 500091867**

**Roll No = R2142210525**

**Experiment No -3**

**Title:- Implementation of Logistics Regression Analysis**

Logistic regression is a statistical method used to analyze the relationship between a binary dependent variable (such as yes/no, success/failure) and one or more independent variables (also called predictor variables or features). It is a type of regression analysis that is commonly used in the field of machine learning and statistics for classification tasks.

The logistic regression model uses a logistic function, also called the sigmoid function, to estimate the probability of a binary outcome. The output of the logistic function is a value between 0 and 1, which can be interpreted as the probability of a positive outcome (e.g., the probability of a customer buying a product, a patient having a disease, etc.) given the input features.

To build a logistic regression model, the algorithm estimates the coefficients of the independent variables that maximize the likelihood of the observed data. The model is trained on a set of labeled data, where the outcome variable is known, and the goal is to learn the relationship between the input features and the outcome.

Code:-

import pandas as pd

import matplotlib.pyplot as plt

df=pd.read\_csv("Iris.csv")

print(df.head())

X = df.drop("Species", axis = 1)

print(X)

y = df["Species"]

print(y)

from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

y = le.fit\_transform(y)

print(y)

from sklearn.linear\_model import LogisticRegression

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=1)

a=LogisticRegression(max\_iter=1000)

a.fit(X\_train, y\_train)

print("Accuracy :", a.score(X\_test, y\_test))

predictions = a.predict(X\_test)

residuals = y\_test - predictions

plt.scatter(range(len(residuals)), residuals)

plt.show()

Output:-



