

School of Computer Science Engineering and Technology  
Assignment-04

|                         |  |
|-------------------------|--|
| <b>Course-</b> B.Tech   | <b>Type-</b> Core  |
| <b>Code-</b> 23CS106    | <b>Course Name-</b> Artificial Intelligence & Machine Learning |
| <b>Year-</b> 2024-2025  | <b>Semester-</b> Even, <b>Instructor:</b> Prof. E.L.N. Kiran   |
| <b>Date-</b> 30-01-2024 | <b>Batch-</b> AIML-A,B   |

## 1 Implement Logistic Regression Model Using Titanic Ship Dataset

### Part 1 – Import the required Python, Pandas, Matplotlib, Seaborn packages

1. Load the Titanic Ship Passengers dataset into a dataframe using `pandas`
2. Check the data types of each feature(column) in the dataset.
3. Generate a summary of the dataset for `min`, `max`, `stddev`, `quartile` vales for 25%,50%,75%,90%,
4. List the names of columns/features in the dataset
5. Perform Exploratory analysis - plot numeric features, check relative size of survived/un-survived, check if any pattern on gender. passenger class, class-wise survival rate, sibilings, overall age distribution, class-wise age distribution - apply bar plot, histogram, box plots to visualize.
6. Perform data wrangling - impute age data, drop unnecessary features, recode the categorical features to a class

### Part 2 – Model training and Fit the data to Model

1. Split the data generated from list created as `X`, `Y` is distributed using `train_test_split` function as `X_train`, `Y_train`, `X_test`, `Y_test`
2. Apply the logistic regression model of `sklearn` package
3. Fit the data to the Logistic Model using `fit`
4. Compute the F1 score as function of penalty , F1 as function of test data. Plot both outcomes using scatter plots.

## 2 Implement the K Nearest Neighbor Classification using Classified Manufacturing Dataset

### Part 1 – Import the required Python, Pandas, Matplotlib, Seaborn packages

1. Load the classified dataset into a dataframe using `pandas`

2. Check the data types of each feature(column) in the dataset.
3. Generate a summary of the dataset for min, max, stddev, quartile vales for 25%,50%,75%,90%,
4. List the names of columns/features in the dataset
5. Scale the features using `StandardScaler` and transform the data

## **Part 2 – Model training and Fit the data to Model**

1. Split the data generated from list created as `X`, `Y` is distributed using *train\_test\_split* function as *X\_train*, *Y\_train*, *X\_test*, *Y\_test*
2. Apply the KNN Classifier model of `sklearn.neighbors` import `KNeighborsClassifier` package
3. Fit the data to the Classier Model using `fit`.

## **Part 3 – Evaluate the Classification Quality**

1. Generate the confusion matrix to estimate the correction among features
2. Generate the classification report using `classification_report`

### 3 Solution - Part 1

---