School of Computer Science Engineering and Technology

Course-BTech	**Type - AI Core-1**
Course Code - CSET211	**Course Name - Statistical Machine Learning**
Year - Second	**Semester - ODD**
Date - 04/11/2024	**Batch - CSE 3rd Semester**

Lab Assignment – 10: K-Means Clustering

CO- Mapping

Section	CO1	CO2	CO3
Section 1: Q1-Q5	√	√	
Section 2: Q1-Q5	√	√	√

Section 1: K-means clustering on Diabetes Dataset

Q1: Load the ***Diabtes.csv*** dataset and implement K-means clustering on the dataset. The code is given on LMS for your reference.

Section 2: K-means clustering on Loan Dataset

Q1. Now, implement K-means clustering on ***Loan.csv*** dataset. The dataset contains following features: *Loan_ID, Gender, Married, Dependents, Education, Self_Employed, ApplicantIncome, CoapplicantIncome, LoanAmount, Loan_Amount_Term, Credit_History, Property_Area*

And the output column *Loan_Status* having *'**Y**' and '**N**' values.*

Apply the following steps to implement K-means clustering on the dataset:

- Load the Loan dataset.
- Drop rows with any missing values using dropna() function
- Encode all categorical features using Label Encoder **except** the target *Loan_Status* column.
- Now, separately encode the target *Loan_Status* column using Label Encoder.

```
# Encode the target column "Loan_Status"
loan_status_encoder = LabelEncoder()
data['Loan_Status'] = loan_status_encoder.fit_transform(data['Loan_Status'])
```

- Separate the features and target into X and y values as shown below:

```
# Separate features and target
X = data.drop(columns=['Loan_Status'])
y = data['Loan_Status']  # This is 0 for "N" and 1 for "Y" after encoding
```

- Scale the data using Standard Scaler.
- Implement K-means clustering with 2 clusters.

```
# KMeans Clustering
kmeans = KMeans(n_clusters=2, random_state=0)
kmeans.fit(X_scaled)
```

- Get the cluster labels

```
# Get the cluster labels
data['Cluster'] = kmeans.labels_
```

- Use PCA for 2D visualization and plot the clusters