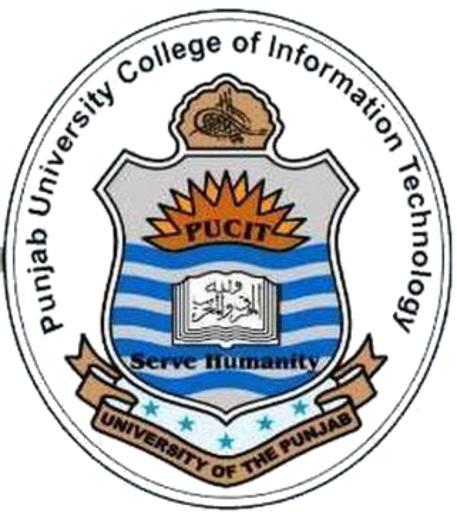
**Machine Learning Project**

****

**Group Members:**

**BCSF19A026-ABDUL AZIZ**

**BCSF20A012-MUHAMMAD UMAR**

**BCSF20A013-YASIR KHAN BAHADUR**

**BCSF20A016-ZARNAB**

**BCSF20A025-MARYAM**

**BCSF20A046-FATIMA**

**Submitted to: Adeel Nisar**

# **REPORT**

# **Introduction:**

This report presents the experiments and results of a project aimed at developing a classification model for sensor data. The **goal** was to extract **meaningful features from the data** and train a machine-learning model to **accurately classify the sensor readings.**

# **Data Preprocessing:**

The project started with data preprocessing, where the provided dataset was processed to extract relevant features.

## **FEATURES:**

Maximum, minimum, mean, and standard deviation, zero crossing, spectral energy, ,spectral energy ,percentiles (20th, 50th, and 80th), first-order mean, second-order mean, interquartile range, skewness, kurtosis, auto-correlation, norm of the first-order mean, norm of the second-order mean

# **Handling Missing Values**:

Missing values were handled using the **Simple Imputer strategy**.

# **Feature Scaling and Selection:**

To prepare the data for modeling, **feature scaling** was performed using the **StandardScaler.**

To further enhance the model's performance and reduce dimensionality, **feature selection** was conducted using **SelectKBest.** The top **275 features** with the highest scores were selected.

# **Model Training and Evaluation:**

Two classification algorithms, **Random Forest and Support Vector Machine (SVM),** were experimented with to evaluate their performance.

# **Experimentation and Results:**

## **Results of Random Forest Classifier:**

It gave an accuracy of **68% on test data** and **100% on train data**

## **Applied min-max scaling:**

After applying min-max scaling accuracy of test data increased to 69.5%

## **SVM Classifier:**

It gave an accuracy of **11% on test data** and **14% on train data**

**Applied Standard Scaling, Selecting Best features and Tuning Hyperparameters:**

After applying these preprocessing techniques, the accuracy of test data increased to 71% and trained data accuracy increased to 98%.

**Applying Min-Max scaling instead of Standard Scaling:**

When we applied min-max scaling instead of standard scaling. The accuracy of test data was achieved by up to 50%.

# **Libraries Used In the Project:**

* import os
* import numpy as np
* from scipy.stats import skew, kurtosis
* from sklearn.svm import SVC
* from sklearn.ensemble import RandomForestClassifier
* from sklearn.impute import SimpleImputer
* from sklearn.metrics import accuracy\_score
* from scipy.stats import entropy
* from sklearn.metrics import confusion\_matrix, f1\_score
* from sklearn.preprocessing import MinMaxScaler

# **Contribution of the members:**

**Bcsf20a012, BCSF20A013, BCSf19a026:** Programming

**BCSF20A016, BCSF20A025, BCSF20A046**: Documentation, Checking the literature for ideas