INTRODUCTION TO DATA SCIENCE (Fall'24)

Mini-Project: Human Activity Recognition Using WISDM Dataset

Submission Date: 19th-November'2024(on GCR)
Total Marks: 100

Project Overview

In this project, you'll use sensor data from a smartphone to recognize human activities like jogging and walking. You'll use both classification and clustering methods to analyze the data, build models, and compare their effectiveness. This project will guide you through the entire data science pipeline, from data exploration to modeling and evaluation.

Dataset Information

The Wireless Sensor Data Mining (WISDM) dataset contains data collected from smartphones placed in users' pockets. The sensors recorded acceleration along three axes: X, Y, and Z. The dataset includes:

- User_ID: Unique identifier for each user.
- Activity_Label: The activity being performed (e.g., Jogging).
- Timestamp: Time of the sensor reading.
- X, Y, Z: Acceleration values along each axis.

The goal is to use this data to:

1. **Build a model that can classify activities based on the accelerometer data.**
2. **Explore clustering to see if we can discover natural groupings of activities.**

Project Tasks

Task 1: Data Loading and Preprocessing

1. **Load the Dataset:** Load the WISDM dataset and inspect its structure.
2. **Data Cleaning:** Check for and handle any missing values, and make sure all data types are appropriate for analysis. You may need to convert columns to the correct types.
3. **Scaling:** Normalize or standardize the acceleration columns (X, Y, Z) for consistency. This will help the models perform better since they'll work with data that's on the same scale.

Task 2: Exploratory Data Analysis (EDA)

1. **Visualize Data:** Plot the values of the X, Y, and Z accelerations over time for different activities (e.g., jogging vs. walking). This will help you understand the patterns and distinguish between activities.
2. **Calculate Summary Statistics:** Calculate the mean, median, and standard deviation of X, Y, and Z values for each activity. This will give you an idea of how the data varies by activity.
3. **Create Additional Features:**
    - Compute the magnitude of the acceleration vector using

      magnitude= $\sqrt{X^2 + Y^2 + Z^2}$. This feature can capture the overall intensity of movement .

Task 3: Model 1 - K-Nearest Neighbors (KNN) Classification

1. **Split the Data:** Divide the dataset into training and testing sets.
2. **Train a KNN Model:**
    - Use KNN to classify activities based on the X, Y, and Z acceleration values.
    - Experiment with different values of K (the number of neighbors) to see which value gives the best results.
3. **Evaluate the Model:**
    - Use metrics like accuracy, precision, and recall to evaluate the model's performance on the test data.

Task 4: Model 2 - K-Means or K-Medoids Clustering

1. **Prepare Data for Clustering:**
    - Use only the X, Y, and Z values for clustering. Ignore the activity labels for this step since clustering is an unsupervised method.
2. **Apply K-Means or K-Medoids:**
    - Set K (the number of clusters) to the number of unique activities in the dataset.
    - Run the algorithm to group data points into clusters based on the acceleration values.
3. **Compare Clusters to True Labels:**
    - Once you have clusters, compare them to the actual activity labels to see if similar activities were grouped together.
    - Hint: Use a confusion matrix to visualize how well the clusters match the true activities.
4. **Analyze Clustering Results:**
    - Discuss whether K-Means/K-Medoids grouped similar activities correctly. Were jogging and walking grouped together, or were there mixed clusters? Reflect on why clustering might or might not work well here.

Task 5: Model 3 - Support Vector Machine (SVM) Classification

1. **Train an SVM Model:**
    - Train an SVM model on the training data to classify activities.
    - Experiment with different kernels (like linear, polynomial, or RBF) to see which gives the best accuracy.
2. **Tune Hyperparameters:**
    - Adjust parameters like C (regularization parameter) and gamma (for RBF kernel) to improve performance.
3. **Evaluate the Model:**
    - Calculate metrics such as accuracy, precision, and recall for the SVM model on the test data.

Task 6: Comparison and Reflection

1. **Compare All Models:**
    - Summarize the performance of KNN, K-Means/K-Medoids, and SVM. Which model had the best accuracy? Which model was fastest to train?

2.Reflection Questions:

- Which approach do you think is best suited for this dataset and why?
- Did the clustering results match your expectations? What challenges did you face?
- How could you improve these models or try other techniques to get better results?

Expected Deliverables

1. **Code:** Submit a Notebook with clear, commented code for each task.
2. **Visualizations:** Include relevant plots in your code, as well as any key visualizations saved as images.
3. **Summary Report:** Submit a short report summarizing your results and reflections. The report should include:
    - Overview of the data
    - Key findings from each model
    - Comparison of model performances
    - Final reflections on the best approach for this dataset

    Your reports must be original and should truly reflect what you have been able to grasp about classification of WISDM dataset(If not justifiably original,they would be straight away canceled out).

Grading Breakdown

1. **Data Preprocessing and EDA - 30%**
2. **KNN Classification - 15%**
3. **K-Means/K-Medoids Clustering - 15%**
4. **SVM Classification - 10%**
5. **Report and Reflection - 30%**

Note: Be sure to start early and do your work in an orderly manner. Good luck, and enjoy working on this project!