Ahmedabad University

Mid Sem Report

Group : MLcops

Course: Machine Learning

Instructor: Prof. Mehul Raval

Project: Classification of Drivers Based on Driving Patterns

Group Members

Roll Number	Name
AU2140164	Deeprajsinh Gohil
AU2140111	Raiyan Diwan
AU2140101	Tejas Pansuriya
AU2140186	Vats Patel

Abstract

This report presents a methodology for driver classification based on their driving patterns using unsupervised learning techniques. The study utilizes data related to various driving parameters such as velocity, warning signals, and collision anticipation. The dataset includes frame number, coordinates, velocities, warning signal distances, warning reception anticipation, and collision types. The analysis employs KMeans clustering and K Nearest Neighbors (KNN) algorithms to categorize drivers based on their driving behaviors. Results demonstrate the effectiveness of the proposed approach in classifying drivers and identifying distinct driving patterns.

Keywords

Unsupervised Learning, Driver Classification, Driving Patterns, KMeans Clustering, K Nearest Neighbors.

Introduction

Understanding driver behavior is crucial for enhancing road safety and developing intelligent transportation systems. Analyzing driving patterns can provide insights into individual driving styles, which can be utilized for various applications such as driver assistance systems, insurance risk assessment, and traffic management. Unsupervised learning techniques offer a data-driven approach to classify drivers based on their driving behaviors without the need for labeled training data. In this report, we present a methodology for driver classification using unsupervised learning algorithms applied to a dataset containing various driving parameters.

Methodology

The methodology involves several steps:

Data Preprocessing: The dataset, comprising parameters such as velocity, warning signal distances, and collision anticipation, is preprocessed to handle missing values, scale numeric features, and prepare the data for analysis.

Feature Engineering: Relevant features are selected, and additional preprocessing steps are performed to extract meaningful information from the raw data. This step involves aggregating data into time windows to capture driving patterns effectively.

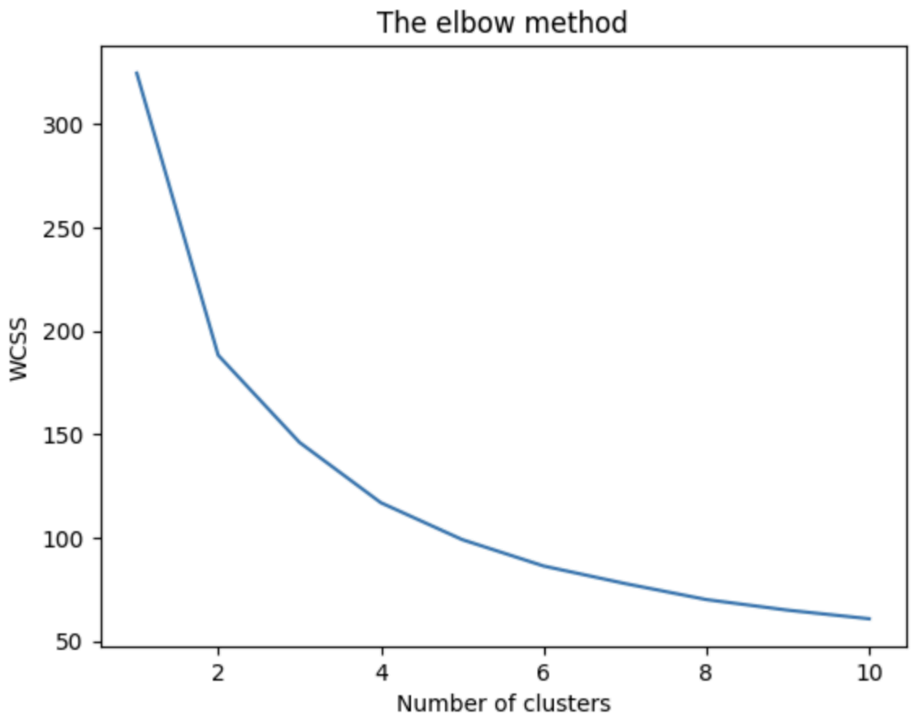
Clustering Analysis: KMeans clustering algorithm is applied to the preprocessed data to group drivers into distinct clusters based on similarity in driving patterns. The optimal number of clusters is determined using the elbow method.

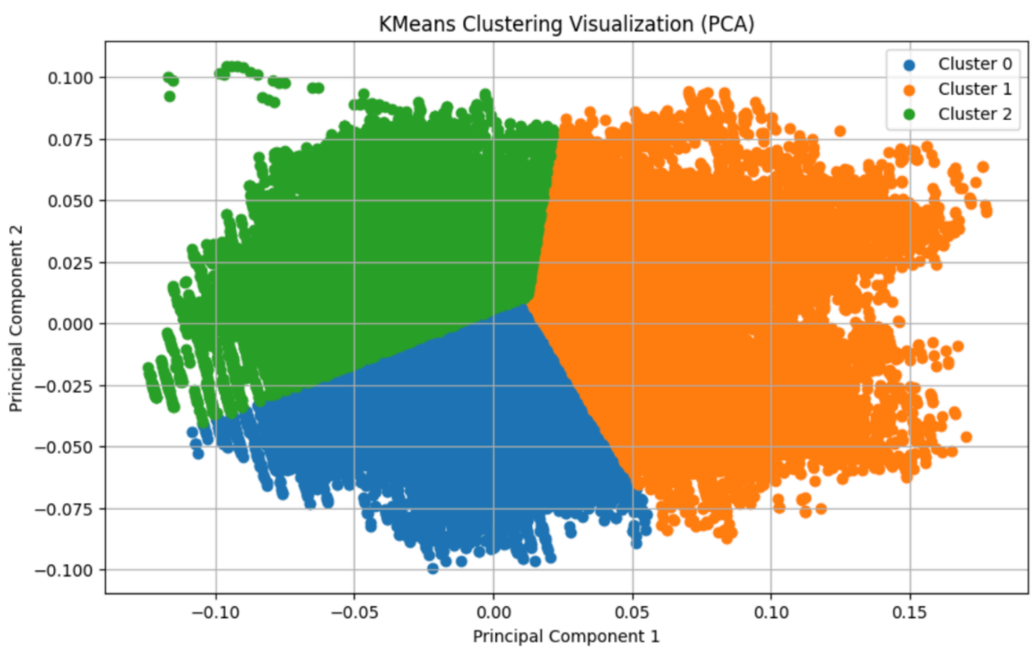
Dimensionality Reduction: Principal Component Analysis (PCA) is employed to reduce the dimensionality of the data for visualization purposes while preserving the variance.

Classification: K Nearest Neighbors (KNN) algorithm is utilized to classify drivers based on their driving patterns. Both dynamic time warping (DTW) and Euclidean distance metrics are explored for classification.

Results

The results reveal distinct clusters of driving behaviors, indicating varying patterns in velocity, warning signal distances, and collision types among drivers. The elbow method suggests an optimal number of clusters for K-means clustering. Visualization using PCA demonstrates separability among clustered data points. Classification accuracy rates using different algorithms are reported, with KNN achieving notable performance in driver classification. Time series-based classifiers also show promising results, particularly when considering the temporal nature of driving patterns.


The elbow method
WCSS
Number of clusters
300
250
200
150
100
50
2
4
6
8
10


KMeans Clustering Visualization (PCA)
Cluster 0
Cluster 1
Cluster 2
Principal Component 1
Principal Component 2
0.100
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.075
−0.100
−0.10
−0.05
0.00
0.05
0.10
0.15

[32]:

```python
# Nearest neighbor classification
knn_clf = KNeighborsTimeSeriesClassifier(n_neighbors=3, metric="dtw")
knn_clf.fit(X_train, y_train)
predicted_labels = knn_clf.predict(X_test)
print("\n2. Nearest neighbor classification using DTW")
print("Correct classification rate:", accuracy_score(y_test, predicted_labels))
```

```
2. Nearest neighbor classification using DTW
Correct classification rate: 0.7
```

[33]:

```python
knn_clf = KNeighborsTimeSeriesClassifier(n_neighbors=3, metric="euclidean")
knn_clf.fit(X_train, y_train)
predicted_labels = knn_clf.predict(X_test)
print("\n3. Nearest neighbor classification using L2")
print("Correct classification rate:", accuracy_score(y_test, predicted_labels))
```

```
3. Nearest neighbor classification using L2
Correct classification rate: 0.7
```

Discussions

The results highlight the importance of unsupervised learning techniques in driver classification based on driving patterns. Clustering analysis provides valuable insights into the diversity of driving behaviors present in the dataset. Additionally, classification accuracy reflects the ability of the proposed methodology to generalize and classify unseen driver data effectively.

Conclusion

The study demonstrates the feasibility of utilizing unsupervised learning techniques for driver classification based on driving patterns. By analyzing various driving parameters, such as velocity, warning signals, and collision anticipation, distinct driving behaviors among drivers are identified. The proposed methodology offers a data-driven approach to categorize drivers, which can be beneficial for various applications in transportation and automotive industries.