INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Driver Profile and Drowsiness Classification

DUARTE FARIA DA MOTA GONÇALVES VALENTE
(Licenciado em Engenharia Informática)

Trabalho de Projeto para obtenção do grau de mestre em Engenharia Informática e Multimédia

Orientadores:
Doutor André Ribeiro Lourenço
Doutor Artur Jorge Ferreira

Júri:
Presidente: Doutor Pedro Viçoso Fazenda
Vogais:
Doutor Hugo Plácido da Silva
Doutor André Ribeiro Lourenço

Setembro, 2025

Driver Profile and Drowsiness Classification

DUARTE FARIA DA MOTA GONÇALVES VALENTE
(Licenciado em Engenharia Informática)

Trabalho de Projeto para obtenção do grau de mestre em Engenharia Informática e Multimédia

Orientadores:
Doutor André Ribeiro Lourenço, ISEL/DEI
Doutor Artur Jorge Ferreira, ISEL/DEI

Júri:
Presidente: Doutor Pedro Viçoso Fazenda, ISEL/DEI
Vogais:
Doutor Hugo Plácido da Silva, IST/UL
Doutor André Ribeiro Lourenço, ISEL/DEI

Setembro, 2025

Acknowledgments

I would like to express my sincere gratitude to my thesis supervisors, Prof. Doctor André Ribeiro Lourenço and Prof. Doctor Artur Jorge Ferreira, for their invaluable guidance, encouragement, and expertise throughout this research. Their mentorship was instrumental in the completion of this work.

I am also grateful to CardioID for providing the datasets and additional technical support, which were essential to this research. Special thanks to Prof. Doutor André Ribeiro Lourenço, whose insights and assistance were crucial in overcoming the challenges encountered.

I extend my appreciation to Luis Loureiro, whose previous thesis laid the foundation for my research. His help in explaining his work was invaluable in guiding my own progress.

Finally, I would like to extend heartfelt thanks to my family for their unwavering support and encouragement throughout this journey. Their belief in me has been a constant source of motivation.

This research was supported by Instituto Politécnico de Lisboa (IPL) under Grant IPL/IDI&CA2024/ML4EP_ISEL.

Duarte Valente

Statement of integrity

I declare that this dissertation is the result of my personal and independent research. Its content is original, and all sources listed in the bibliographic references were consulted and are duly mentioned in the text. I further declare that all scientific and technical references relevant to the development of the work are duly cited and included in the bibliographic references.

The author

Duarte Faria da Mota Gonçalves Valente

Lisbon, June 23, 2025

Resumo

Todos os dias, aproximadamente 3.700 pessoas perdem a vida em acidentes rodoviários, totalizando 1,35 milhões de fatalidades anuais em todo o mundo. As principais causas destes acidentes incluem excesso de velocidade, distração, condução sob efeito de substâncias e fadiga. Todos estes fatores contribuem para comportamentos de condução inseguros.

Mudar os hábitos de condução dos condutores é uma tarefa complexa, uma vez que as campanhas de segurança rodoviária, por si só, têm demonstrado um impacto limitado na redução da sinistralidade. No entanto, apesar da dificuldade em influenciar o comportamento humano, os avanços tecnológicos oferecem soluções promissoras para mitigar riscos e aumentar a segurança rodoviária.

Esta tese investiga comportamentos de condução potencialmente perigosos, com especial foco em estilos de condução ariscados e na sonolência ao volante. Através da análise de dados obtidos por sensores embutidos nos veículos e sinais fisiológicos dos condutores, exploramos métodos para avaliar o estado do condutor e identificar padrões associados a um maior risco de acidentes. Especificamente, analisamos métricas de condução como o excesso de velocidade, travagens bruscas e acelerações súbitas para classificar tendências de condução agressiva. Além disso, foi feita a análise de características extraídas de eletrocardiogramas, registados através de sensores equipados no volante, para classificar o nível de fadiga no condutor.

Os resultados obtidos indicam que tanto os marcadores comportamentais como os fisiológicos podem ser indicadores eficazes de estados de condução potencialmente perigosos. Em particular, os comportamentos agressivos estão fortemente associados ao risco de acidente, enquanto a fadiga e a condução prolongada comprometem significativamente o desempenho do condutor. Estas conclusões contribuem para o desenvolvimento de sistemas inteligentes de monitorização capazes de identificar e mitigar condições de condução inseguras em tempo real, promovendo uma maior segurança rodoviária.

Palavras-chave

Classificação de Perfis de Condução; Condução Sonolenta; Estilos de Condução; i-DREAMS; Variabilidade da Frequência Cardíaca.

Abstract

Every day, approximately 3,700 people die in road accidents, totaling 1.35 million fatalities globally each year. The primary causes of these accidents include speeding, distracted driving, drunk driving, nighttime driving, and drowsy driving.

Changing human driving habits is extremely challenging, as road safety campaigns alone have shown limited impact on reducing fatalities. However, while influencing behavior is challenging, technological advancements offer promising solutions to mitigate risks and enhance road safety.

This thesis explores unsafe driving behaviors, with a particular focus on both dangerous and drowsy driving. Using data from in-car sensors and physiological signals, we investigate methods to assess driver states and identify patterns associated with an increased risk of accidents. Specifically, we analyze driving behavior metrics, such as speeding, harsh braking, and sudden acceleration, to classify risky driving tendencies. Additionally, we leverage heart rate variability features extracted from electrocardiograms recorded via a sensor-equipped steering wheel to detect signs of driver drowsiness.

The results indicate that both behavioral and physiological markers can serve as effective indicators of unsafe driving conditions. In particular, aggressive driving behaviors are strongly linked to accident risk, while prolonged driving and fatigue significantly impair driver performance. Moreover, individual differences in responses to sleep deprivation highlight the need for personalized assessment methods. These insights contribute to the development of intelligent monitoring systems capable of identifying and mitigating unsafe driving conditions in real time, ultimately enhancing road safety.

Keywords

Driver Profile Classification; Driving Style; Drowsiness Classification; Drowsy Driving; i-DREAMS; Heart Rate Variability.

Index

Acknowledgments i

Resumo v

Abstract vii

Symbology and abbreviations xvii

1 Introduction 1
- 1.1 Context . . . 1
- 1.2 Objectives . . . 2
  - 1.2.1 Driver Profile Approach . . . 2
  - 1.2.2 Drowsiness Detection Approach . . . 2
- 1.3 Thesis Contributions . . . 3
- 1.4 Structure . . . 4

2 State Of The Art 5
- 2.1 Introduction . . . 5
- 2.2 Driver Behavior Analysis . . . 6
- 2.3 Physiological Monitoring and Drowsiness Detection . . . 7
  - 2.3.1 Physiological Signals and Their Relation to Internal States . . . 7
  - 2.3.2 Understanding Heart Rate Variability (HRV) . . . 8
  - 2.3.3 HRV Features: Time Domain vs. Frequency Domain . . . 9
  - 2.3.4 HRV in Drowsiness Detection . . . 9
- 2.4 Sleepiness Evaluation Methods . . . 10
  - 2.4.1 Karolinska Sleepiness Scale (KSS) . . . 10
  - 2.4.2 KSS for Machine Learning Models . . . 11
  - 2.4.3 Discussion . . . 11
- 2.5 Normalization Techniques and Their Impact on Performance . . . 12
  - 2.5.1 Driver Profiling: Normalization for Clustering . . . 12
  - 2.5.2 Drowsiness Detection: Normalization for Supervised Learning . . . 12
- 2.6 Feature Engineering and Their Impact on Performance . . . 13
  - 2.6.1 Feature Selection Techniques . . . 13

2.6.2 Feature Reduction Techniques . . . 13
2.7 Machine Learning Methods for Driver Monitoring . . . 14

3 Materials and Methods **15**
3.1 Introduction . . . 15
3.2 Scaling and Normalization of Data . . . 16
3.2.1 Common Techniques . . . 16
3.2.2 Normalization in Trip Data . . . 17
3.3 Feature Selection Techniques . . . 18
3.3.1 Feature Relevance . . . 18
3.3.2 Feature Redundacy . . . 20
3.4 Feature Reduction Techniques . . . 20
3.4.1 Principal Component Analysis (PCA) . . . 20
3.4.2 Singular Value Decomposition (SVD) . . . 21
3.5 Supervised Learning Techniques . . . 22
3.5.1 Neural Networks for Classification . . . 22
3.5.2 Long Short-Term Memory Network for Time-Series Classification . . . 22
3.6 Summary . . . 24

4 Driver Profile Implementation **27**
4.1 Dataset Analysis . . . 27
4.1.1 Overview of the Dataset . . . 27
4.1.2 Data Preprocessing . . . 29
4.1.3 Differences from Luis Loureiro's Dataset . . . 30
4.2 Dataset Normalization . . . 30
4.3 Feature Selection . . . 31
4.4 Feature Reduction . . . 31
4.5 Unsupervised Learning . . . 31
4.5.1 Clustering Evaluation Metrics . . . 32
4.5.2 First Stage Clustering . . . 32
4.5.3 Second Stage Clustering . . . 33
4.5.4 Explainability Analysis . . . 34
4.6 Supervised Learning . . . 35
4.6.1 Machine Learning Pipeline . . . 35
4.6.2 Evaluation Metrics and Model Selection . . . 36

5 Drowsiness Detection Implementation **37**
5.1 Dataset Analysis . . . 37
5.1.1 Overview of the Dataset . . . 37
5.1.2 Data Preprocessing . . . 39
5.1.3 Final Datasets Analysis . . . 43
5.2 Dataset Normalization . . . 44

5.3 Feature Selection . . . 44
5.4 Feature Reduction . . . 45
5.5 Supervised Learning . . . 45
5.5.1 Baseline Classification Models . . . 45
5.5.2 Time Series Classification with LSTM . . . 46
5.5.3 Final Adjustments and Model Optimization . . . 47

6 Evaluation - Driver Profile **49**
6.1 Testing Enviroment . . . 49
6.2 Normalization . . . 50
6.3 Feature Selection . . . 50
6.4 Feature Reduction . . . 53
6.5 Final Datasets . . . 54
6.6 Unsupervised Learning Results . . . 55
6.6.1 First Stage Clustering Results . . . 55
6.6.2 Second Stage Clustering . . . 57
6.7 Driver Profile Supervised Learning . . . 59
6.7.1 Class Imbalance . . . 59
6.7.2 Classifiers Performance Evaluation . . . 59
6.7.3 Supervised Dimensionality Reduction . . . 63
6.7.4 Final Models . . . 64

7 Evaluation - Drowsiness **65**
7.1 Feature Selection . . . 65
7.2 Feature Reduction . . . 65
7.3 Final Datasets . . . 66
7.4 Supervised Learning Results . . . 66
7.4.1 Baseline Classification Models Results . . . 66
7.4.2 Time Series Classification with LSTM Results . . . 68

8 Conclusions **73**
8.1 Future Work . . . 74

Bibliography **79**

Figure index

1.1 Overview of the proposed block diagram for the two approaches: Driver Profiling and Drowsiness Detection. 3

2.1 An ECG graph showing a series of QRS complexes, where the time between heartbeats (R-R interval) varies naturally from beat to beat. Firstbeat Technologies Oy (2025) 8
2.2 Proposed categorical labels for KSS Score. 11

3.1 Example of a Feedforward Neural Network for Drowsiness Detection. 22
3.2 Example of an LSTM Architecture. Roy et al. (2024) 23

4.1 Architecture of the construction of the dataset from the trip event data gathered by the API. Use of unsupervised learning techniques to build a supervised dataset where the best model can be found. 29

5.1 Comparison between the duration of the simulator data with the ECG data. . . . 40
5.2 Comparison between the duration of the simulator data with the ECG data, after filtering. 40
5.3 Evolution of drowsiness state for each participant during all the tests. 43

6.1 Features sorted by Relevance using the Mean-Median metric. 51
6.2 Similarity between the top 10 features pairs. 52
6.3 Distribution of instances based on key features. 56
6.4 Distribution of instances based on key features. 59
6.5 Distribution of instances based on key features. 60
6.6 DT, RF, XGBoost and SVM confusion matrices. 62
6.7 Top features selected by RRFS, using the supervised Fisher ratio as the relevance metric. 63

7.1 Random Forest ROC Curve for each dataset. 68
7.2 LSTM Model Confusion Matrix and ROC curve for the Time Domain 2 min interval dataset with a window size of 10. 69
7.3 Comparison of different interval sizes of HRV features for a window size of 10. . 70
7.4 Models Accuracy Results from Combinations of Different Interval Duration and Window Size. 70

7.5 Best LSTM Models Confusion Matrices and ROC Curves. 72

Table index

2.1 Karolinska Sleepiness Scale 11

5.1 Unsupervised HRV datasets 42
5.2 Supervised Datasets with HRV features 42

6.1 Distance vs Duration Normalization Results 50
6.2 Mean-Median Feature Relevance Clustering Results 51
6.3 Relevance-Redundancy Feature Selection (RRFS) Clustering Results 53
6.4 PCA and SVD Clustering Results 54
6.5 Comparison of Feature Selection and Reduction Methods 54
6.6 First Stage Clustering Results 55
6.7 Top Features Statistical Comparison 57
6.8 Second stage K-Means clustering results 57
6.9 Top Features Statistical Comparison 60
6.10 Performance of supervised classification with DT, RF, XGBoost, and SVM using instance sampling techniques and evaluated by Accuracy (ACC), Precision (PREC), Recall (REC), F1, and AUC scores. 61
6.11 Nested cross-validation experimental results. 62
6.12 Experimental results for the XGBoost classifier, after dimensionality reduction with RFS(FiR) and RRFS(FiR). 63
6.13 Experimental results of the best models for diferent Fisher Ratio best scorring features. 64

7.1 PCA and SVD Supervised Learning Results 66
7.2 Random Forest classification experimental results for all datasets. 67
7.3 LSTM classification experimental results for Time Domain 2 min interval dataset with a window size of 10. 69
7.4 LSTM classification results for Time Domain 2 min interval dataset and with window size of 10. 71

Symbology and abbreviations

Abbreviations

ADASYN	Adaptive Synthetic Sampling
ANS	Autonomic Nervous System
AUC-ROC	Area Under the Receiver Operating Characteristic Curve
CH	Calinski-Harabasz
DBI	Davies-Bouldin Index
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DT	Decision Trees
ECG	Electrocardiography
EEG	Electroencephalography
EOG	Electrooculography
FNN	Feedforward Neural Networks
GMM	Gaussian Mixture Models
HRV	Heart Rate Variability
KSS	Karolinska Sleepiness Scale
LSTM	Long Short-Term Memory Networks
NN	Neural Networks
PCA	Principal Component Analysis
PNN	Probabilistic Neural Networks
PNS	Parasympathetic Nervous System
PVT	Psychomotor Vigilance Tests
RFS	Relevance Feature Selection
RRFS	Relevance-Redundancy Feature Selection
RF	Random Forests
SMOTE	Synthetic Minority Oversampling Technique
SNS	Sympathetic Nervous System
SVD	Singular Value Decomposition
SVM	Support Vector Machines
WCSS	Within-Cluster Sum of Squares

Chapter 1

Introduction

1.1 Context

The previous Master of Science by Loureiro (2023) focused on developing a dataset for trip style assessment using real trip data. Their research aimed to identify different driving profiles through a two-stage clustering approach. The primary goal was to develop a system that takes advantage of machine learning techniques to devise a driver profile identification based on trip data obtained in a non-intrusive way. Their methodology involved performing feature engineering on trip data, preprocessing the dataset, and applying clustering algorithms to classify driving styles into categories such as aggressive, non-aggressive, and risky.

The experimental results showed that distinct driving styles could be identified through clustering techniques. However, the study primarily relied on feature reduction methods rather than feature selection techniques. This approach left room for further exploration to determine the most relevant features contributing to the classification of driving styles. Additionally, the work lacked an in-depth analysis of explainability, making it difficult to understand which features had the most significant impact on determining whether a driver exhibited a safe or aggressive driving style.

Building on this foundation, this thesis aims to address these gaps by exploring feature selection techniques instead of an only feature reduction approach. By identifying the most influential features, this research seeks to improve the model's interpretability and enhance the accuracy of driving style classification. Another key focus of this work is to provide explainability for the results, offering insights into the relevance and impact of different features in distinguishing safe from aggressive driving styles.

Unlike the driving behavior study, which builds upon previous research, the drowsy driving analysis was developed from the ground up. In this case, a dataset was obtained from an experiment conducted in a driving simulator, where multiple participants, some of whom were sleep-deprived, were monitored while driving. During these sessions, physiological signals were recorded using electrooculography (EOG), electrocardiography (ECG), and electroencephalography (EEG). However, for this research, only ECG data was used. Additionally, during the experiment, participants provided subjective drowsiness levels using the Karolinska Sleepi-

ness Scale (KSS).

The objective of this study is to build a supervised dataset using the recorded ECG data and KSS labels. By applying supervised learning models, we aim to develop an approach capable of predicting driver drowsiness in real time, allowing for timely alerts to prevent potential accidents. Furthermore, this research explores whether it is possible to identify specific factors influencing drowsiness while driving and to assess the variability between different drivers. A crucial aspect of this study is determining whether a generalized model can be developed that is effective for a broad population, rather than being overfitted to a small group of individuals.

1.2 Objectives

The main objective of the previous study by Loureiro (2023) was to devise a dataset and to apply clustering techniques to identify different driving styles based on trip data. While the study successfully categorized driving behaviors, it did not delve deeply into the importance of individual features in determining these classifications.

The primary objectives of this thesis are two-fold, as described in the following.

1.2.1 Driver Profile Approach

1. To explore various feature selection techniques and assess their impact on driving style classification.
2. To refine the previous clustering approach by incorporating selected features that contribute the most to distinguishing between driving behaviors.
3. To analyze feature importance and provide explainability for the results, ensuring a better understanding of which factors influence the most the driving style classifications.
4. To evaluate the impact of feature selection techniques on clustering performance and compare the outcomes with those obtained in the previous study.

1.2.2 Drowsiness Detection Approach

1. To explore and understand the dataset obtained from the driving simulator experiment.
2. To compute Heart Rate Variability (HRV) features from the ECG data.
3. To use the recorded KSS values as labels and build a supervised dataset for training predictive models.
4. To train different supervised learning models and identify the most effective one for predicting drowsiness.
5. To investigate whether there are common physiological patterns that influence drowsiness while driving.
6. To analyze individual differences between drivers and determine whether a generalized model can be created that is suitable for a diverse group of people.

By achieving these objectives, this thesis aims to enhance both driver profile analysis and drowsiness detection. The results could provide valuable insights for developing real-time monitoring systems to improve road safety, with applications in areas such as insurance risk assessment, fleet management, and driver assistance technologies.

Figure 1.1 ilustrates the proposed block diagram for both Driver Profiling and Drowsiness Detection approaches.

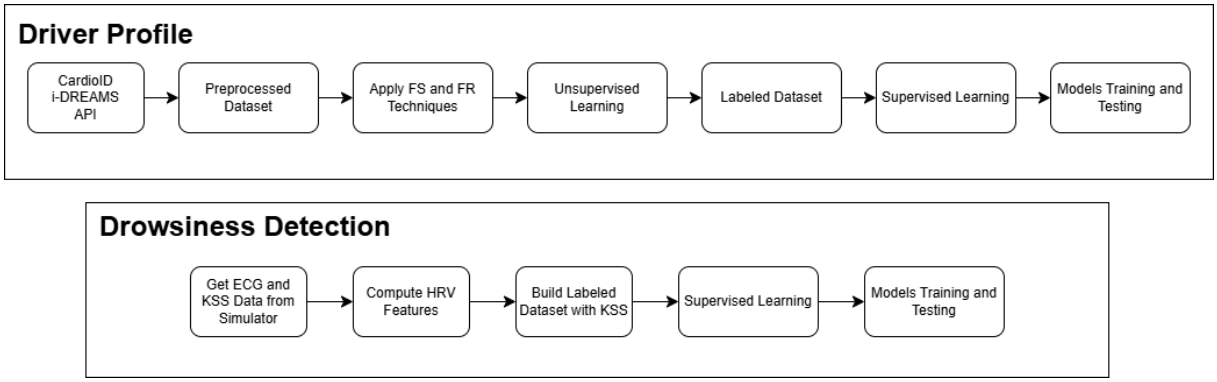

Figure 1.1 Overview of the proposed block diagram for the two approaches: Driver Profiling and Drowsiness Detection.

1.3 Thesis Contributions

This thesis contributed to the development of intelligent driver monitoring systems by addressing two critical aspects of road safety: driver profiling and drowsiness detection. The work presented here led to the implementation of machine learning pipelines for classifying driving styles and detecting drowsiness using physiological data.

In the scope of this thesis, two scientific papers were submitted and are currently under peer review:

- "Assessing Driver Style and Driver Volatility with Machine Learning Techniques" – submitted to INForum 2025.
- "Drowsiness Detection with Time-Series Classification Using HRV Features" – submitted to NCTA 2025.

To promote reproducibility and encourage future research, all code developed during this thesis, including preprocessing, feature extraction, and model training scripts, is available in a public repository: github.com/duartevalente10/TFM-Driver-Behavior-and-Drowsiness-Detection.

1.4 Structure

The remainder of this thesis is structured as follows:

- Chapter 2 - State of the Art – This chapter analyzes the current research and common approaches related to data normalization, feature selection and reduction, Heart Rate Variability (HRV) in drowsiness detection, sleepiness evaluation methods, and supervised learning techniques. It emphasizes how these concepts have been applied in existing literature to address similar problems.
- Chapter 3 - Materials and Methods – This chapter describes the theoretical foundation and methodological choices applied in this work. It details the normalization strategies, feature engineering techniques, HRV-based drowsiness detection methodology, and the supervised learning models implemented, including their rationale and application within this study.
- Chapter 4 - Driver Profile Implementation – This chapter presents the implementation of the driving behavior classification component. It includes the analysis and preprocessing of the driver profile dataset, the use of feature normalization, feature selection, and feature reduction techniques, followed by the development of unsupervised and supervised learning models for driving style classification.
- Chapter 5 - Drowsiness Detection Implementation – This chapter details the implementation of the drowsiness detection system. It covers the analysis and preprocessing of physiological data, the application of normalization, feature selection, and feature reduction techniques, and the development of supervised learning models to predict driver drowsiness based on heart rate variability features.
- Chapter 6 - Driver Profile Experimental Results – The results of experiments for the driver profile classification task are presented and analyzed, comparing the effectiveness of feature selection against the previous feature reduction approach to identify driving behavior.
- Chapter 7 - Drowsiness Detection Experimental Results - The results of experiments for the drowsiness classification task are presented and analyzed, showing the different combinations and the evaluation metrics fdor the best performing models.
- Chapter 8 - Conclusion - The final chapter summarizes the findings, discusses the implications of the research, and suggests directions for future work.

Chapter 2

State Of The Art

Chapter 2 provides a comprehensive review of existing research relevant to driver behavior analysis and drowsiness detection. Section 2.1 introduces the scope and objectives of the literature review. Section 2.2 explores methods and findings related to driver behavior analysis, focusing on behavioral indicators of risk. Section 2.3 shifts to physiological monitoring, with particular emphasis on drowsiness detection. It includes an overview of physiological signals, an explanation of heart rate variability (HRV), a comparison of time and frequency domain HRV features, and their relevance in drowsiness detection. Section 2.4 reviews sleepiness evaluation methods, focusing on the Karolinska Sleepiness Scale (KSS) and its integration into machine learning applications. Section 2.5 discusses normalization techniques and their respective roles in clustering and supervised learning tasks. Section 2.6 addresses the impact of feature engineering, covering both feature selection and dimensionality reduction strategies. Finally, Section 2.7 presents machine learning methods commonly applied in driver monitoring systems, highlighting their strengths and limitations in this domain.

2.1 Introduction

Understanding and categorizing driving behaviors has become a critical area of research in modern transportation, given by advancements in telematics, the Internet of Things (IoT), and machine learning techniques. Identifying unsafe driving conditions, whether due to risky driving behaviors or drowsiness is essential for improving road safety, optimizing fleet management, and enabling more accurate risk assessments. This chapter provides a comprehensive review of the theoretical foundations and methodologies relevant to both driving behavior analysis and drowsy driving detection.

In previous work, Loureiro (2023) conducted an extensive study on driver profiling, focusing on the initial stages of dataset construction and machine learning model development. Their research covered key aspects, including:

- Driver Profiles - Different categorizations of driving styles based on behavioral indicators.
- Parameters for Driver Profiling - Features such as speed, acceleration, braking, lateral maneuvers, and distractions.

- Methods for Driver Profiling - A systematic analysis of clustering techniques, supervised learning models, and feature extraction.
- Machine Learning Techniques - An evaluation of methods such as K-Means, Density-Based Clustering Non-Parametric Algorithm (DBSCAN), Gaussian Mixture Models (GMM), Support Vector Machines (SVM), Decision Trees (DT), and Random Forests (RF).

Their study demonstrated the potential of clustering techniques for categorizing drivers based on real trip data. However, the focus was primarily on feature engineering and initial classification, leaving room for further developments in feature selection, explainability, and model performance. This thesis builds upon their foundation by exploring feature selection techniques, evaluating their impact on driving style classification, and improving the interpretability of the results.

In addition to driving behavior analysis, this research also seeks to extend to drowsy driving detection. Unlike the driving behavior study, which builds upon previous research, the drowsiness detection component was developed from the ground up. The primary goal is to understand how physiological signals, particularly electrocardiography (ECG) and Heart Rate Variability (HRV), can reveal internal states linked to drowsiness. HRV is particularly valuable due to its resistance to noise and its ability to estimate Autonomic Nervous System (ANS) activity, which changes during episodes of stress, fatigue, and drowsiness.

To build a robust drowsiness detection system, this study leverages data collected from a driving simulator experiment, where participants, some of whom were sleep-deprived, were monitored while driving. Physiological signals, including electrooculography (EOG), electroencephalography (EEG), and ECG, were recorded, alongside Karolinska Sleepiness Scale (KSS) scores, which provides subjective sleepiness labels. In this research, ECG data is used to compute HRV features, which are then used as inputs for machine learning models aimed at predicting driver drowsiness.

By integrating these techniques, this research aims to enhance driving behavior classification and develop effective models for drowsiness detection, ultimately contributing to safer and more intelligent driver monitoring systems.

2.2 Driver Behavior Analysis

Understanding and evaluating driver behavior is a key factor in enhancing road safety and developing intelligent driver monitoring systems. Driver behavior can range from cautious and defensive to aggressive and risky, impacting not only the driver but also passengers and other road users. Over the years, several approaches have been developed to profile and classify driving styles based on measurable vehicle and driver parameters.

Ma et al. (2021) proposed a framework to classify driving styles as aggressive, normal, or cautious by analyzing online car-hailing data. Their method involved detecting driving maneuvers using Principal Component Analysis (PCA) and clustering drivers through K-Means algorithms. Their study highlighted that variations in driving tasks, such as turning, acceleration, and deceleration, significantly affect a driver's style, underlining the dynamic nature of driving behavior.

Jurecki and Stańczyk (2021) focused on evaluating driving techniques as safe or unsafe by measuring longitudinal and lateral acceleration values over a 650 km route encompassing various road types. Their findings emphasized that different road conditions influence driving patterns, as drivers tend to adapt their acceleration behaviors based on the environment.

Focusing on critical scenarios, Riahi Samani and Mishra (2022) investigated Commercial Motor Vehicle (CMV) drivers' behavior during take-over conditions using driving simulators. Their analysis, based on Multivariate Dynamic Time Warping and K-Means clustering, categorized driving behaviors into normal, conservative, and risky styles. These findings provided valuable insights for automotive industries and transportation planners dealing with automated or semi-automated driving systems.

Complementary to these studies, Tement et al. (2022) introduced a segmentation approach based on common response strategies and cognitive workload. Their clustering model demonstrated how stress and high cognitive load could influence driver behavior, offering insights into the variability of driver performance under different conditions.

Together, these diverse approaches highlight the richness and complexity of driving behavior analysis. They demonstrate that driver profiling can be effectively achieved through a combination of vehicle dynamics data, physiological state monitoring, and environmental context. However, while behavior-based profiling captures external actions, it often fails to detect internal states like fatigue or drowsiness, which may also be dangerous.

Thus, to build a more comprehensive understanding of driver safety, it is essential to complement behavior analysis with physiological monitoring. This thesis addresses that aspect through the integration of Heart Rate Variability (HRV) analysis for drowsiness detection.

2.3 Physiological Monitoring and Drowsiness Detection

This section aims to provide the theoretical background necessary to understand the physiological basis and significance of Heart Rate Variability (HRV) in the context of drowsiness detection. It outlines the relationship between physiological signals and internal cognitive or emotional states, introduces the concept of HRV, and describes commonly used HRV features.

2.3.1 Physiological Signals and Their Relation to Internal States

The human body continuously responds to various internal and external stimuli through changes in physiological signals. These signals, which include heart rate, brain activity, muscle tone and eye movement, provide valuable insights into a person's mental and physical condition. Among

these, cardiac activity is particularly useful for assessing alertness levels, as it is directly regulated by the Autonomic Nervous System (ANS).

The ANS is composed by two branches:

- Sympathetic Nervous System (SNS) – Activates during stress, exertion, or excitement, increasing heart rate.
- Parasympathetic Nervous System (PNS) – Dominates during rest and recovery, slowing the heart rate.

Fluctuations in heart rate result from the continuous interplay between these two branches, making HRV an adequate tool for assessing fatigue, stress, and drowsiness.

Studies like Matsuzaki et al. (2006) and Cho (2022) have shown that drowsy individuals experience shifts in ANS balance, characterized by increased parasympathetic activity (from PNS) and reduced sympathetic modulation (from SNS). This transition can be identified through changes in HRV features, making it a reliable indicator of drowsiness onset.

2.3.2 Understanding Heart Rate Variability (HRV)

Heart Rate Variability (HRV) refers to the variation in time between consecutive heartbeats, known as RR intervals (measured from one R-peak to the next in an electrocardiogram - ECG). Contrary to common belief, a healthy heart does not beat at a constant rate; instead, it exhibits natural fluctuations (Figure 2.1) influenced by breathing, activity, and emotional states.

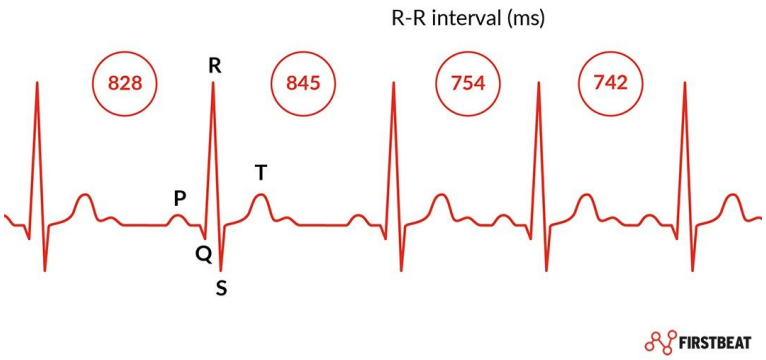

Figure 2.1 An ECG graph showing a series of QRS complexes, where the time between heartbeats (R-R interval) varies naturally from beat to beat. Firstbeat Technologies Oy (2025)

HRV is widely used in medical and neuroscience research because it provides a non-invasive measure of ANS function. A higher HRV is generally associated with a healthy, adaptable nervous system, while a lower HRV may indicate fatigue, stress, or impaired regulatory control, such as in drowsiness.

For driver drowsiness detection, analyzing HRV features helps determine whether an individual is experiencing alertness or fatigue, allowing the development of real-time monitoring

systems to prevent accidents.

2.3.3 HRV Features: Time Domain vs. Frequency Domain

HRV can be analyzed using two primary approaches:

1. Time Domain Analysis – Measures fluctuations in RR intervals over a period.
2. Frequency Domain Analysis – Examines HRV in terms of power spectral density, which helps differentiate between SNS and PNS activity.

The time domain HRV features analysis involves computing statistical measures from RR intervals. Commonly used features include:

- Mean RR Interval (MeanNN) – The average time between heartbeats.
- Standard Deviation of NN intervals (SDNN) – Measures overall HRV. Higher SDNN indicates greater variability and adaptability.
- Root Mean Square of Successive Differences (RMSSD) – Reflects short-term variability, mainly influenced by the parasympathetic system.
- pNN50 – The percentage of successive RR intervals differing by more than 50ms. Higher values suggest increased vagal tone (PNS dominance).

Studies show that increased drowsiness is associated with lower SDNN and RMSSD, reflecting reduced autonomic control over heart rate.

The frequency domain HRV features analysis decomposes HRV signals into different frequency bands:

- Low Frequency (LF: 0.04–0.15 Hz) – Represents both SNS and PNS activity.
- High Frequency (HF: 0.15–0.40 Hz) – Mainly reflects parasympathetic activity and is linked to respiratory influences.
- LF/HF Ratio – A key metric indicating autonomic balance. Higher LF/HF suggests increased sympathetic activity (stress and alertness), while lower LF/HF is associated with fatigue and drowsiness.

Research suggests that during drowsiness, HF power tends to increase, while LF/HF ratio decreases, indicating a shift towards parasympathetic dominance.

2.3.4 HRV in Drowsiness Detection

Heart Rate Variability (HRV) analysis has emerged as a reliable method for detecting fatigue and drowsiness in driving environments. Changes in autonomic nervous system activity reflected in HRV metrics may signal a shift from an alert to a drowsy state. This makes HRV a non-invasive and data-rich source for modeling drowsiness using machine learning techniques.

Recent studies have provided strong evidence for the effectiveness of HRV in detecting drowsiness. For example, Khushaba et al. (2011) demonstrated that HRV monitoring can effectively identify transitions into drowsiness during simulated driving tasks, highlighting its value for real-time fatigue assessment. More recently, Liu et al. (2024) addressed the use of HRV indices as reliable biomarkers for driving-related fatigue, showing that specific HRV patterns correlate with decreasing levels of alertness. Furthermore, research by Widodo and Arifin (2019) confirmed the feasibility of implementing HRV-based drowsiness detection on embedded systems, reinforcing its practicality in real-world automotive applications.

This thesis leverages HRV features extracted from ECG signals to develop a supervised learning approach for drowsiness detection. By training machine learning models on labeled datasets annotated with Karolinska Sleepiness Scale (KSS) scores, the aim is to construct a generalizable model capable of accurately identifying drowsy states across individuals with varying physiological baselines.

In summary, HRV offers valuable insights into driver alertness through both time and frequency domain features. These insights form the basis for intelligent, non-intrusive monitoring systems that can help prevent fatigue-related accidents. This section has provided the theoretical foundation for the HRV-based drowsiness detection strategy adopted in this thesis.

2.4 Sleepiness Evaluation Methods

Understanding how to measure sleepiness is essential for developing reliable drowsiness detection systems. Sleepiness can be assessed through subjective and objective methods. Subjective measures rely on self-reported assessments of drowsiness levels, while objective measures use physiological signals, such as brain activity, eye movements, or Heart Rate Variability (HRV), to determine alertness states.

Among the most commonly used subjective scales, the Karolinska Sleepiness Scale (KSS) Kaid et al. (2016) has gained widespread acceptance due to its simplicity and effectiveness in sleep-related studies.

2.4.1 Karolinska Sleepiness Scale (KSS)

The Karolinska Sleepiness Scale (KSS) is a self-reported, 9-point scale in which individuals assess their current level of sleepiness. Participants are asked to rate their perceived alertness or drowsiness, selecting a value between 1 (very alert) and 9 (very sleepy, fighting sleep). Table 2.1 descrives the KSS scale.

Validation and Correlation with Physiological Measures

Several studies have confirmed the validity of KSS as an effective measure of sleepiness by correlating it with objective physiological indicators such as Electroencephalography (EEG) features, reaction times like Psychomotor Vigilance Tests (PVT) and behavioral performance met-

Table 2.1 Karolinska Sleepiness Scale

Scale	Degree of Alertness/Sleepiness
1	Subject extremely alert
2	Very alert
3	Alert
4	Fairly alert
5	Neither alert nor in sleep mode
6	Few signs of sleepiness
7	Sleepy, no effort to keep alert
8	Sleepy, noticeable effort to keep alert
9	Extremely sleepy, great effort to keep alert

rics.

A study by (Kaid et al., 2016) found an almost linear correlation between KSS scores and physiological sleepiness markers, reinforcing its effectiveness as a ground truth measure for drowsiness detection models.

2.4.2 KSS for Machine Learning Models

To facilitate the use of KSS in automated drowsiness detection, KSS values can be arranged into categorical labels, simplifying classification tasks. Acording to the study by Oliveira et al. (2018) we decided to categorize our KSS scale in a 2-class or 3-class approach, as represented in Figure 2.2:

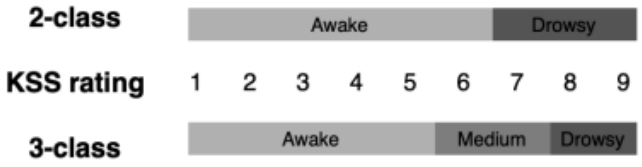

Figure 2.2 Proposed categorical labels for KSS Score.

This binary classification can be used for real-time drowsiness monitoring systems, where a model must quickly determine whether a driver is in a risky drowsy state. The multi-class classification allows for a more granular evaluation of drowsiness levels, capturing transitional states between full alertness and severe drowsiness.

2.4.3 Discussion

The Karolinska Sleepiness Scale (KSS) provides a reliable and validated way to measure sleepiness in real-world scenarios. Its strong correlation with physiological indicators makes

it suitable for training machine learning models in drowsiness detection. By binarizing or categorizing KSS values, it becomes possible to develop effective classification models that can predict driver alertness states using physiological features like HRV.

This section has provided the foundation on how sleepiness will be evaluated in this thesis, forming the basis for the drowsiness classification models implemented in later chapters.

2.5 Normalization Techniques and Their Impact on Performance

In machine learning, data normalization is a crucial preprocessing step that ensures all features are on a similar scale, preventing models from being biased toward variables with larger magnitudes. Many machine learning algorithms, especially those relying on distance-based calculations (K-Means, Support Vector Machines, and Neural Networks), perform better when the data is scaled or normalized. Without this pre-processing thechniques, features with higher numerical ranges can dominate those with smaller ranges, leading to suboptimal clustering and classification results.

2.5.1 Driver Profiling: Normalization for Clustering

In the context of driver profiling, normalization addresses the variability inherent in trip data. Drivers undertake trips of varying lengths and durations, and behaviors such as harsh braking or speeding events must be analyzed proportionally to ensure fair comparisons. To achieve this, trip data is normalized by distance or duration, converting absolute counts into rates (e.g., events per kilometer or per minute).

This approach aligns with the findings of Deepali Virmani (2015), who proposed a normalization-based K-Means clustering algorithm (N-K Means). Their study demonstrated that applying normalization prior to clustering enhances the algorithm's performance by mitigating the influence of features with larger scales, which can skew the Euclidean distance calculations whiche are central to K-Means clustering.

2.5.2 Drowsiness Detection: Normalization for Supervised Learning

For drowsiness detection, Heart Rate Variability (HRV) features derived from Electrocardiogram (ECG) signals are utilized. These physiological signals can vary significantly between individuals due to factors like age, fitness level, and measurement conditions. To ensure that the supervised learning models accurately capture patterns related to drowsiness, it's essential to normalize these features.

Argentina Leite (2020) employed z-score normalization on HRV features before inputting them into Long Short-Term Memory (LSTM) networks for classification tasks. By standardizing the features, subtracting the mean and dividing by the standard deviation, they achieved improved model performance, highlighting the importance of normalization in handling physiological data with inherent variability.

2.6 Feature Engineering and Their Impact on Performance

2.6.1 Feature Selection Techniques

Feature selection is a pivotal step in machine learning and data preprocessing, aiming to identify and retain the most relevant features while eliminating those that are irrelevant or redundant. Unlike feature reduction techniques, such as Principal Component Analysis (PCA), which transform features into a new space, feature selection preserves the original features, thereby enhancing the interpretability of the model's outcomes.

In the realm of driver profiling, feature selection plays a crucial role in refining model accuracy. By discarding unnecessary trip parameters and focusing on the most significant ones, the model becomes more adept at classification tasks. While previous work by Loureiro (2023) primarily emphasized feature reduction, this thesis introduces feature selection as a complementary approach to optimize performance and bolster interpretability.

Three primary benefits of Feature Selection are as follows:

1. Improves Model Performance – By removing noise and irrelevant data, feature selection reduces overfitting and enhances the model's accuracy.
2. Enhances Explainability – Retaining the original features allows for easier interpretation of which driving behaviors influence classification outcomes.
3. Reduces Computational Cost – A streamlined feature set accelerates training times and improves computational efficiency. Moreover, there is no need to acquire some features in the future.

The significance of feature selection is further underscored in the study by Ferreira and Figueiredo (2012), which evaluated both supervised and unsupervised feature selection methods. By computing feature relevance and redundancy to filter data, the study concluded that such methods not only improve model efficiency for high-dimensional datasets but also reduce costs, making them practical for real-world applications.

2.6.2 Feature Reduction Techniques

Feature reduction techniques aim to reduce the dimensionality of a dataset by transforming the original features into a smaller set while preserving as much information as possible. Unlike feature selection, which keeps original features and removes less relevant ones, feature reduction creates new features by combining or transforming existing ones.

In driver profiling, feature reduction is crucial because datasets often contain highly correlated parameters, such as acceleration, braking intensity, and speed variations. Reducing dimensionality helps in:

- Improving computational efficiency – Lowering memory and processing requirements.

- Avoiding the curse of dimensionality – High-dimensional data leads to increased sparsity, making clustering and classification less effective.
- Enhancing model performance – Reducing redundant information can lead to better generalization.

Luis Loureiro's research applied Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) to reduce dimensionality in driver profiling. This thesis also applies the same methods but now aiming to compare the combined impact with feature selection.

2.7 Machine Learning Methods for Driver Monitoring

In this thesis, supervised learning techniques are employed for both driver profiling classification and drowsiness detection. While several methods, including K-Means, DBSCAN, Gaussian Mixtures, Evidence Accumulating Clustering, Support Vector Machines, Decision Trees, Random Forests, and XGBoost, have been thoroughly discussed in the state of the art of Loureiro's thesis, this section focuses on the introduction and justification of two new supervised learning methods: Neural Networks (NN) and Long Short-Term Memory Networks (LSTM).

The application of NN and LSTM is particularly pertinent given the time-series nature of the data involved in driver profiling and drowsiness detection. Traditional methods like K-Means clustering, which rely on engineered features, have limitations in capturing complex temporal patterns inherent in physiological signals. For instance, Balakrishnan et al. (2019) demonstrated that K-Means clustering, when applied to engineered HRV features, failed to identify meaningful structures in the data. In contrast, convolutional and LSTM autoencoders, trained on raw RR interval measurements, successfully identified clusters corresponding to stressed and normal states, as validated by established physiological stress markers.

Furthermore, studies have shown that NN can outperform traditional clustering methods in classification tasks. A study by Izquierdo et al. (2010) found that NN achieved a classification accuracy of 99.1%, surpassing the performance of K-Means and Expectation-Maximization (EM) clustering methods in binary classification tasks.

Additionally, a study by Li et al. (2019) found that the combination of Gaussian Mixture Models and wavelet features, followed by classification using Probabilistic Neural Networks (PNN), has been shown to achieve high classification accuracy of 99.99% in ECG beat classification, outperforming traditional methods that rely solely on Gaussian mixtures.

These findings underscore the necessity of employing NN and LSTM in this thesis to effectively model the complex, nonlinear, and temporal dependencies present in the physiological signals associated with driver behavior and drowsiness. By leveraging these advanced supervised learning techniques, the thesis aims to enhance the accuracy and interpretability of the classification models, thereby contributing to the development of reliable systems for driver profiling and drowsiness detection.

Chapter 3

Materials and Methods

Chapter 3 details the methodological approach adopted in this study. Section 3.1 introduces the structure of the chapter and the rationale behind the chosen methods. Section 3.2 describes the scaling and normalization techniques applied to the data, with emphasis on both general methods and those specifically adapted to trip-based data. Section 3.3 presents the feature selection techniques used to evaluate feature relevance and redundancy. Section 3.4 focuses on feature reduction, highlighting the use of Principal Component Analysis (PCA) and Singular Value Decomposition (SVD). Section 3.5 introduces the supervised learning techniques, including feedforward neural networks and Long Short-Term Memory (LSTM) networks for time-series classification. Finally, Section 3.6 summarizes the key methodologies covered in the chapter.

3.1 Introduction

This chapter presents the methodological framework and techniques employed throughout the development of this work. While the previous chapter focused on analyzing existing literature and commonly adopted approaches, this chapter provides a detailed description of the specific methods used in this study, providing both the theoretical rationale and practical implementation of each one.

The methods are organized into several key topics. Firstly, the normalization techniques applied to the dataset are described, including Min-Max Scaling, Z-Score Normalization, Unit Vector Scaling, Trip Normalization by Distance, and Trip Normalization by Duration. These methods are essential for ensuring data consistency and improving model performance.

Next, the chapter outlines the feature selection strategies used to identify the most relevant attributes for model training. These techniques include Fisher's Ratio, Mutual Information, Mean Absolute Difference, Mean-Median Difference, Arithmetic Mean - Geometric Mean, Variance, Correlation Coefficient, and Absolute Cosine Similarity. Each method is discussed in terms of its ability to evaluate the informativeness and redundancy of features.

Feature reduction methods are also addressed, with particular focus on Principal Component Analysis (PCA) and Singular Value Decomposition (SVD). These techniques are used to

transform the original feature space into a lower-dimensional representation, enhancing computational efficiency and reducing noise.

Finally, this chapter details the supervised learning techniques implemented to model and classify the data. These include traditional neural networks for classification tasks, as well as Long Short-Term Memory (LSTM) networks, which are particularly suited for time-series data.

Together, the methods presented in this chapter form the foundation of the experimental framework and are critical to achive the objectives of this research.

3.2 Scaling and Normalization of Data

This section explores different normalization techniques, including Min-Max Scaling, Z-Score Normalization, and Unit Vector Scaling. It also explains the rationale behind trip-based normalization and physiological data normalization in both driver profiling and drowsiness detection.

3.2.1 Common Techniques

Several standard normalization techniques are widely used in machine learning and data preprocessing.

Min-Max Scaling (Feature Scaling)

Min-Max Scaling rescales features to a fixed range, typically [0,1] or [-1,1], using the formula:

$$X_{\text{scaled}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}, \tag{3.1}$$

where X is the original value, X_{min} is the minimum value of the feature, and X_{max} is the maximum value.

- Advantages - Preserves the original distribution and ensures all features are within the same scale.
- Disadvantages - Sensitive to outliers; extreme values may distort the scale.

Z-Score Normalization (Standardization)

Z-score normalization, also known as standardization, transforms features to have a mean of 0 and standard deviation of 1, using the formula:

$$X_{\text{scaled}} = \frac{X - \mu}{\sigma}, \tag{3.2}$$

where μ is the mean and σ is the standard deviation of the feature.

- Advantages - Works well when features follow a normal distribution and are used in algorithms that assume Gaussian-like data.
- Disadvantages - Does not confine values to a fixed range, making interpretation less intuitive.

Unit Vector Scaling (L2 Normalization)

This technique transforms each feature into a unit vector so that all values have a norm of 1, calculated as:

$$X_{\text{scaled}} = \frac{X}{\|X\|}, \tag{3.3}$$

where $\|.\|$ denotes the norm of the vector.

- Advantages - Useful in text classification and cosine similarity applications.
- Disadvantages - Less common in time-series or tabular datasets.

3.2.2 Normalization in Trip Data

Unlike standard datasets in which features are independent, trip data presents a unique challenge: trips vary in length and duration. This variability affects event-based metrics such as number of harsh braking events, acceleration instances, or speeding occurrences, making it unfair to compare absolute counts across different trips.

Normalization by Distance

One intuitive approach is to normalize all trip-based metrics per kilometer traveled:

$$X_{\text{scaled}} = \frac{X}{TripDistance(km)}, \tag{3.4}$$

where X represents an event count (e.g., number of harsh braking events). This ensures that a trip with 10 braking events over 100 km (0.1 events/km) is fairly compared to a trip with 5 braking events over 50 km (also 0.1 events/km).

- Advantages - Provides a standardized metric that accounts for longer and shorter trips and it is useful in identifying aggressive drivers who exhibit more risky behaviors per kilometer.
- Disadvantages - Less useful for some time-based features, where trip duration is more relevant.

Normalization by Duration

An alternative is to normalize events per second or per minute of driving time:

$$X_{\text{scaled}} = \frac{X}{TripDuration(s)}, \tag{3.5}$$

This is particularly relevant for time-based metrics, where the total driving time is a more meaningful indicator than distance.

- Advantages - Standardizes data for fair comparison between short and long-duration trips.
- Disadvantages - Less useful for distance-dependent parameters.

3.3 Feature Selection Techniques

3.3.1 Feature Relevance

Feature relevance methods assess how important each feature is in distinguishing between driver profiles. These techniques can be supervised (requiring labeled data) or unsupervised (working without labels).

Supervised Feature Selection

Supervised techniques evaluate the relationship between features and the target variable.

Fisher's Ratio (FR)

Fisher's Ratio by Fisher (1936) measures how well a feature separates two classes by comparing the variance within and between classes:

$$FR(X_i) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}, \tag{3.6}$$

where μ_1 and μ_2 are the mean values of feature X_i for each class, and σ_1^2 and σ_2^2 are the variances for each class.

Application in driver profiling - Features with high Fisher's Ratio (e.g., speeding events per km) are better at distinguishing aggressive vs. non-aggressive drivers.

Mutual Information (MI)

Mutual Information by Thomas and Cover (1991) quantifies the dependency between a feature Xi and the target variable Y:

$$I(X_i; Y) = \sum_{x \in X_i} \sum_{y \in Y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)}, \tag{3.7}$$

where $I(X_i; Y)$ represents the mutual information between feature X_i and target Y. $P(x, y)$ is the joint probability distribution, and $P(x)$, $P(y)$ are the marginal probabilities.

A higher MI value means a stronger relationship between the feature and the class label. Leading to identify which features can be more usefull to the classification task, and also provide interpretability.

Unsupervised Feature Selection

Unsupervised techniques assess feature relevance without requiring class labels.

Mean Absolute Difference (MAD)

MAD measures feature variability:

$$MAD(X_i) = \frac{1}{n}\sum_{j=1}^{n} |X_{ij} - \bar{X}_i|, \tag{3.8}$$

where n is the number of samples, X_{ij} are the data points, and $\bar{X}_i$ is the mean (average) of the feature.

A higher MAD value indicates that a feature varies significantly across trips, suggesting that it may be useful for clustering.

Mean-Median Difference

This metric compares the mean and median of a feature:

$$D(X_i) = |mean(X_i) - median(X_i)|, \tag{3.9}$$

A large difference suggests a skewed distribution, which may indicate a key behavioral feature.

Arithmetic Mean - Geometric Mean (AM-GM)

The AM-GM ratio helps identify features with high variability:

$$R(X_i) = \frac{\text{Arithmetic Mean}(X_i)}{\text{Geometric Mean}(X_i)} = \frac{\frac{1}{n}\sum_{j=1}^{n} x_{ij}}{\left(\prod_{j=1}^{n} x_{ij}\right)^{\frac{1}{n}}}, \tag{3.10}$$

A high ratio suggests a non-uniform distribution, which may indicate a critical feature for distinguishing drivers. $R \geq 1$, with higher values meaning that the arithmetic and geometric means are more apart.

Variance

High-variance features typically contain more information and are more valuable for classification tasks. The variance is defined as

$$Var(X_i) = \frac{1}{n}\sum_{j=1}^{n} (X_{ij} - \bar{X}_i)^2, \tag{3.11}$$

where n is the number of samples, X_{ij} are the data points, and $\bar{X}_i$ is the mean of the feature.

For driver profiling, speed variability and acceleration intensity are high-variance features that may be important for classification.

3.3.2 Feature Redundacy

Even if a feature is relevant, it may be redundant if it strongly correlates with another feature. Removing redundant features prevents overfitting and improves model efficiency. We may report some metrices to assess feature similarity.

Correlation Coefficient

The Pearson correlation coefficient measures the linear relationship between two features:

$$p(X_i, X_j) = \frac{\sum(X_i - \bar{X}_i)(X_j - \bar{X}_j)}{\sqrt{\sum(X_i - \bar{X}_i)^2}\sqrt{\sum(X_j - \bar{X}_j)^2}}, \tag{3.12}$$

If $p > 0.9$, the two features are highly correlated, and one with lower relevane can be removed.

Absolute Cosine Similarity

This technique measures how similar two feature vectors are in multi-dimensional space:

$$|cos(\theta)| = \frac{X_i \cdot X_j}{\|X_i\|\|X_j\|}, \tag{3.13}$$

where a value closer to 1 indicates redundancy. Where $< a, b >$ denotes the inner product between vector a and vector b.

3.4 Feature Reduction Techniques

3.4.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) by Strang and Borre (1997) is one of the most widely used dimensionality reduction techniques. It transforms the original dataset into a new coordinate system where the most significant variance is captured in fewer dimensions, known as principal components (PCs).

Given an $n \times d$ dataset X, where n is the number of instances and d is the number of features, PCA follows these steps:

1. Standardization of the dataset – Ensure all features have zero mean and unit variance:

$$X = \frac{X - \mu}{\sigma}. \tag{3.14}$$

2. Compute the covariance matrix C of the dataset:

$$C = \frac{1}{n}(X^T X). \tag{3.15}$$

3. Eigen decomposition – Solve for eigenvalues λ_i and eigenvectors v_i:

$$Cv_i = \lambda_i v_i. \tag{3.16}$$

The eigenvectors correspond to the principal components, and eigenvalues indicate their variance contribution.

4. Select the top-k principal components – Choose components that explain a predefined amount of variance (Say, 90% of the total variance).

5. Project the data onto the new lower-dimensional space:

$$X_{reduced} = XV_k, \tag{3.17}$$

where V_k contains the top k eigenvectors (corresponding to the larger eigenvalues).

This method can capture maximum variance in fewer dimensions, while working well with continuous numerical data and improving model efficiency by reducing the feature space. But on the other hand, interpretability can be lost, since original features are transformed into abstract components, and is also sensitive to outliers, which can distort variance estimation.

3.4.2 Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD) by Strang and Borre (1997) is another adequate dimensionality reduction method. Unlike PCA, which relies on covariance matrices, SVD directly decomposes the dataset into three matrices:

$$X = U\Sigma V^T, \tag{3.18}$$

where U is the left singular vectors (instances), Σ is the diagonal matrix of singular values (importance of each component) and V^T represents right singular vectors (original features).

To reduce dimensionality:

1. Retain only the top-k singular values from Σ

2. Use the corresponding columns from U and V^T

3. Transform the data into a lower-dimensional space using:

$$X_{reduced} = U_k\Sigma_k \tag{3.19}$$

This feature reduction approach can handle sparse datasets, making it useful for real-world sensor data and be more robust to missing values than PCA. Although, it can be computationally more expensive than PCA and also transforms features, reducing interpretability.

3.5 Supervised Learning Techniques

3.5.1 Neural Networks for Classification

Neural Networks (NN) are a class of machine learning models capable of learning complex patterns from data. Generically, a NN is a universal function approximator. Unlike traditional models like Random Forest or SVM, NN are composed of multiple layers of interconnected neurons, allowing them to model highly non-linear relationships.

For drowsiness detection, Feedforward Neural Networks (FNN) can be used to classify HRV features extracted from ECG signals. These networks consist of an input layer to get the data, the hidden layers to perform non-linear transformations through activation functions and the output layer to produce the final classification.

Figure 3.1 illustrates the architecture of a Feedforward Neural Network (FNN) that takes HRV features as input and produces a binary output indicating whether the driver is drowsy or awake.

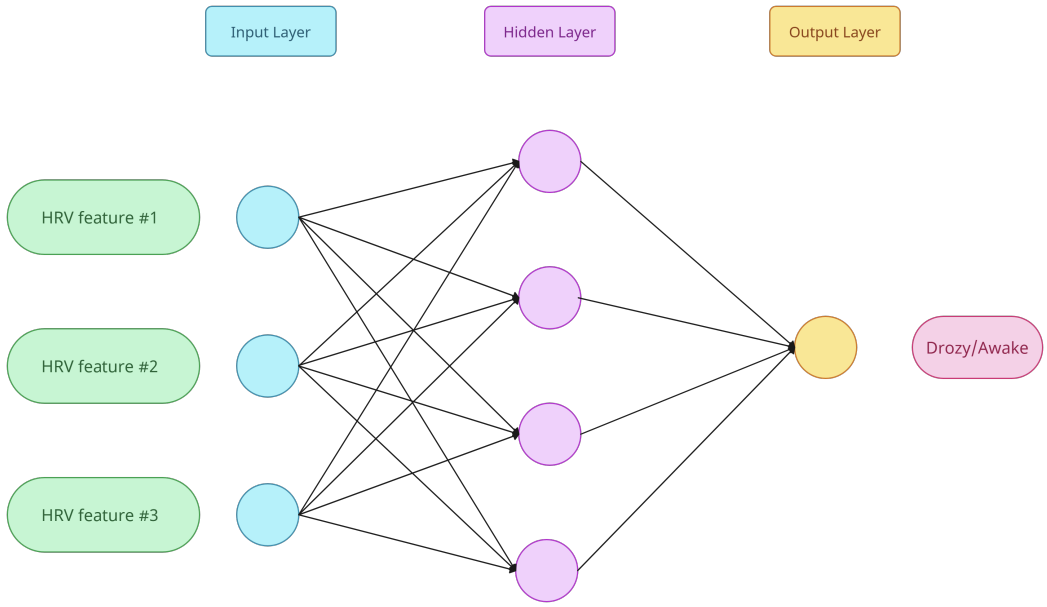

Figure 3.1 Example of a Feedforward Neural Network for Drowsiness Detection.

While traditional neural networks are effective in many classification problems, they are limited when dealing with sequential data, such as time-series information from physiological signals. This is where recurrent architectures like LSTM become more suitable.

3.5.2 Long Short-Term Memory Network for Time-Series Classification

Since drowsiness classification is inherently a time-series problem, traditional machine learning models may struggle to capture temporal dependencies in physiological signals like ECG. LSTM (Long Short-Term Memory networks) are a specialized type of Recurrent Neural Net-

works (RNN) designed to handle sequential data by maintaining information over long time intervals.

LSTM improve upon standard RNN by introducing memory cells to can store and regulate information flow over time. Each LSTM unit consists of a forget gate which decides which past information should be discarded, an input gate to determine which new information should be stored, a cell state to store long-term memory and an output gate to decide what part of the memory should be passed to the next step.

Figure 3.2 presents a basic architecture of an LSTM network and its key components, including the cell state, forget gate, input gate, and output gate.

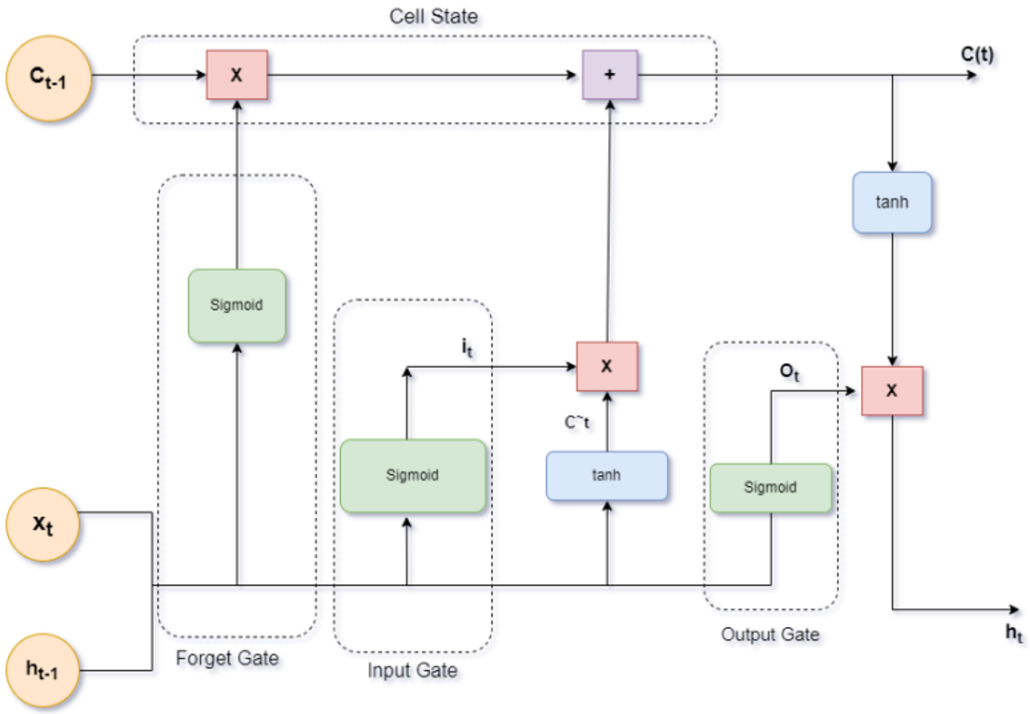

Figure 3.2 Example of an LSTM Architecture. Roy et al. (2024)

Working of LSTM

For a sequence of intervals (int1, int2, int3, ..., intn) and processing the sequence one interval at a time. The state of the LSTM at time step t as (ht, ct), where ht is the hidden state and ct is the cell state. The LSTM works as follows:

- Step 1 - The LSTM receives the input vector (xt) and the previous state (ht-1, ct-1).
- Step 2 - The forget gate decides what information to discard from the cell state. Uses the input vector and the previous hidden state to generate a number between 0 and 1 for each number in the cell state ct-1. A value closer to 1 value represents information to keep and closer to 0 represents information to discard.
- Step 3 - The input gate decides what new information to store in the cell state. It has two parts. A sigmoid layer decides which values to update, and a tanh layer creates a vector

of new candidate values (Ct) that could be added to the state.

- Step 4 - Update the old cell state (ct-1) to the new cell state (ct). The old cell state is multiplied by ft to forget the things decided to forget earlier. Then add the new candidate values, scaled by how much it decided to update each state value.
- Step 5 - Decide the output, that will be based on the cell state filtered version. It passes through a sigmoid layer which decides what parts of the cell state it is going to output. Then, the cell state values are arranged to be in the range from -1 to 1 and multiplied by the output of the sigmoid gate, so only output the parts decided to.

By leveraging this architecture, LSTM can model sequential dependencies in HRV signals, making them highly effective for drowsiness classification based on physiological time-series data.

3.6 Summary

This chapter provided an in-depth review of key concepts and techniques relevant to driver profiling and drowsiness detection. Building upon previous work by Loureiro (2023), which focused on dataset construction and initial classification, this thesis expands the research by optimizing feature selection, feature reduction, normalization techniques, and supervised learning methodologies to enhance classification performance and interpretability.

First, we discussed **normalization techniques**, which play a crucial role in ensuring fair comparisons between trips of varying lengths and durations. We explored different normalization strategies, including Min-Max Scaling, Z-Score Normalization, and Unit Vector Scaling, and justified the trip-specific normalization approach (normalizing events per distance or duration).

Next, we examined feature selection techniques, which aim to identify the most relevant features while eliminating redundant or non-informative ones. We categorized these techniques into:

- Feature Relevance Methods, including Fisher's Ratio, Mutual Information, and unsupervised relevance metrics (Mean Absolute Difference, Variance, and AM-GM Ratio), which help identify the most influential trip parameters.
- Feature Redundancy Reduction, using correlation coefficients and cosine similarity to remove features that convey redundant information.

By applying feature selection, we aim to retain only the most meaningful features, thereby improving model efficiency and interpretability without transforming the original dataset.

We then explored feature reduction techniques, which take a different approach by transforming the dataset into a lower-dimensional space while preserving as much information as possible. We reviewed:

- Principal Component Analysis (PCA), which projects data onto new axes (principal components) based on variance.
- Singular Value Decomposition (SVD), which factorizes data into singular vectors and singular values, providing an alternative method for dimensionality reduction.

By combining feature selection and feature reduction, we aim to enhance clustering and classification accuracy while improving computational efficiency.

While several traditional unsupervised and supervised methods have been used in this research, they were already covered in Luis Loureiro's thesis and thus were not reintroduced here. Instead, we explored new supervised learning techniques, including:

- Neural Networks (NN) for classification, which can capture complex relationships in drowsiness detection.
- Long Short-Term Memory (LSTM) Networks, which are particularly effective for time-series classification and were introduced to better handle the sequential nature of HRV data for drowsiness detection.

This chapter complements the state of the art chapter by integrating both driver profiling and drowsiness detection, introducing new methodologies while maintaining a connection to previous research. By combining machine learning advancements with physiological signal analysis, this thesis lays the groundwork for a more robust and interpretable classification framework that can be applied to real-world driver safety systems.

Chapter 4

Driver Profile Implementation

This chapter provides a comprehensive overview of the methodology used to classify driving behavior based on the available dataset. Section 4.1 begins by analyzing the driver profile dataset, describing its structure, presenting key statistics, and comparing it with the original dataset from Loureiro's work to establish a baseline. In Section 4.2, the focus shifts to feature normalization, where techniques are applied to ensure data consistency across trips of varying lengths, thereby standardizing the dataset for unbiased model training. Section 4.3 delves into feature selection, highlighting the process of identifying the most relevant variables while eliminating redundant ones to enhance model interpretability and computational efficiency. Following this, Section 4.4 discusses feature reduction, examining dimensionality reduction techniques such as Principal Component Analysis (PCA) and Singular Value Decomposition (SVD), which are employed to further streamline the data and improve model performance. Section 4.5 covers the unsupervised learning phase of the driver profiling task, detailing the clustering methods used to uncover distinct driving behavior patterns. A two-stage clustering approach is adopted—first differentiating between safe and risky trips, and then further categorizing the risky trips. Finally, Section 4.6 concludes with the supervised learning phase, where the labeled dataset resulting from the clustering process is used to train classification models, enabling automated driving behavior analysis.

By following this structured methodology, this chapter aims to develop accurate, interpretable, and efficient models, contributing to improved road safety and real-time driver monitoring systems.

4.1 Dataset Analysis

4.1.1 Overview of the Dataset

The dataset used in this study was originally constructed in Luis Loureiro's research and is described in Chapter 3 of his thesis. Therefore, this section provides only a high-level summary of the data and focuses primarily on how it was reprocessed and adapted for this work.

The dataset was obtained from the i-DREAMS API, containing trip records collected from April 1, 2021, to July 20, 2022. Each trip includes a set of recorded events corresponding to

driving behaviors, vehicle movements, and environmental conditions. The dataset was structured per trip, meaning each row corresponds to a unique trip, and its columns represent features extracted from the recorded events.

Each trip contains a set of mandatory features essential for identifying trip details, namely:

- trip_start – Timestamp indicating when the trip began.
- trip_end – Timestamp indicating when the trip ended.
- distance – Total distance traveled during the trip (in kilometers).
- duration – Total duration of the trip (in seconds).

In addition to these mandatory features, other trip characteristics were recorded, resulting in an initial dataset of 17,138 trips and 75 features. However, due to the unavailability of Hands-On and Drowsiness Systems, only 67 features were retained in the dataset.

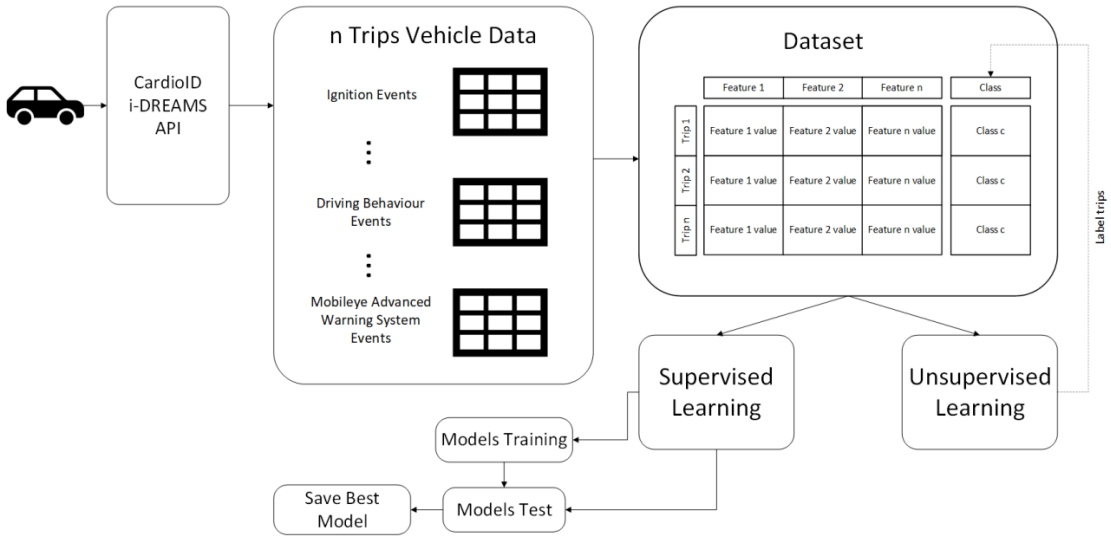

Figure 4.1 Architecture of the construction of the dataset from the trip event data gathered by the API. Use of unsupervised learning techniques to build a supervised dataset where the best model can be found.

4.1.2 Data Preprocessing

To ensure data quality and consistency, the same preprocessing scripts used in the previous work were applied to the dataset. The following steps were performed:

1. Removal of Invalid Trips

Certain trips were removed based on data completeness and trip validity criteria:

- Trips where all values were missing were removed.
- Trips where only the four mandatory features were present, but all other values were missing were also removed.
- Trips with less than one minute of duration were excluded to avoid very short, incomplete trips.
- Trips with less than 1.5 km traveled were removed to ensure valid driving data.

2. Handling Missing Values

For the remaining dataset, missing values were imputed using different strategies based on feature characteristics:

- For numerical features, missing values were filled using the median, ensuring robustness against outliers.
- For categorical features (e.g., light_mode, which indicates whether the trip was during the day, dusk, or night), missing values were imputed using contextual information, such as trip start and end timestamps, to infer the most probable light condition.

3. Final Processed Dataset

   After preprocessing, the final dataset contained:

- 13,207 trips (compared to the initial 17,138 trips).
- 65 features (instead of the original 67, after feature cleaning).

4.1.3 Differences from Luis Loureiro's Dataset

Although the same preprocessing techniques were applied, the final dataset differs slightly from the dataset used in Loureiro's study. The key differences are:

1. Different raw data sources – The dataset for this study contained only 15,213 trips, whereas Loureiro's dataset originally had 17,138 trips.
2. Changes in preprocessing – As a result of missing values and different available trips, the final number of processed trips is slightly lower than in the previous study.

Because of these differences, it is necessary to recompute all previous results using the new dataset to ensure fair comparison between Loureiro's methodology and the improved implementation proposed in this study.

4.2 Dataset Normalization

The Driver Profile Dataset was normalized using the same techniques applied in Loureiro's work, as the scaling of features has a direct impact on clustering performance. Many clustering algorithms, such as K-Means, DBSCAN, and Gaussian Mixture Models, rely on Euclidean distances to group data points, meaning that features with larger numerical ranges can dominate those with smaller ranges.

To avoid bias toward high-magnitude features, two normalization approaches were applied. The first one is distance normalization to normalize the parameters of the dataset by the total distance traveled. The second one is normalization by trip duration.

Since Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) were applied as Feature Reduction techniques, normalization was also essential to ensure that each feature contributes equally to the variance maximization process. Without normalization, features with larger magnitudes would dominate the principal components, leading to biased feature reduction.

The impact of distance normalization vs. duration normalization was evaluated during the modeling phase, and the most effective approach was selected based on performance metrics.

4.3 Feature Selection

This section introduces a new approach that improves the results of Luis Loureiro's work by incorporating unsupervised feature selection techniques before applying clustering algorithms. The goal was to eliminate redundant and less relevant features, ensuring that clustering algorithms work with only the most informative trip parameters.

To achieve this, two feature selection strategies were considered:

1. Mean-Median Feature Relevance
   Mean-Median Feature Relevance is an unsupervised metric that can help us to identify high variability. This metric was calculated for each dataset's feature and then ordered by value. Since a higher mean-median value indicates that a feature has greater variability and is more informative, diferents sets of top-ranked features were created and tested in the clustering algorithms.

2. Relevance-Redundancy Feature Selection (RRFS)
   The Relevance-Redundancy Feature Selection (RRFS) method proposed by Ferreira and Figueiredo (2012) was implemented to improve clustering performance. RRFS evaluates features using:

   - Relevance Measures - Mean-Median Feature Relevance was used to find the most relevant features from the dataset.
   - Redundancy Measures - Absolute cosine similarity was computed for each feature, ensuring that highly correlated features were removed.

   By applying this method, we retained only highly relevant features, removed redundant features that provide duplicate information and improved clustering accuracy.

4.4 Feature Reduction

The Driver Profile Dataset contains a high number of extracted features, making it susceptible to the curse of dimensionality as shown by Altman (2018). Since this dataset is used for unsupervised learning, only unsupervised feature reduction techniques such as Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) were applied.

4.5 Unsupervised Learning

Unsupervised learning plays a crucial role in driver profiling, as it allows for the identification of natural groupings within the dataset without the need for predefined labels. Clustering techniques were applied to detect patterns in driving behavior and segment drivers based on their trip characteristics.

To ensure robust and meaningful clustering, the process was divided into two stages:

1. First Stage Clustering – Applying different clustering techniques to find a separation into aggressive and safe driving styles.
2. Second Stage Clustering – Refining the results by further decomposing clusters to search for more granular driving styles.

Following the clustering process, an Explainability Analysis was conducted to interpret the meaning of the discovered clusters, ensuring that the results were both meaningful and actionable.

4.5.1 Clustering Evaluation Metrics

Evaluating clustering performance is challenging because unsupervised learning does not rely on predefined labels. Therefore, internal validation metrics were used to assess cluster quality by measuring compactness, separation, and structure.

To determine the best clustering method, three widely used clustering evaluation metrics were used:

1. Calinski-Harabasz Index (CH) by Caliński and Harabasz (1974) – To measures cluster compactness and separation.
2. Davies-Bouldin Index (DBI) byDavies and Bouldin (1979) – To evaluate intra-cluster similarity and inter-cluster separation.
3. Silhouette Score by Rousseeuw (1987) - To measure how well each data point fits within its assigned cluster.

Each metric provides complementary insights into cluster quality, allowing for a more comprehensive evaluation of the different clustering approaches in both stages.

4.5.2 First Stage Clustering

The first stage of clustering aimed to identify broad driver profiles by applying multiple clustering techniques. Based on the literature and prior studies, the following clustering methods were used:

1. K-Means Clustering by Hartigan and Wong (1979)
   K-Means is a widely used centroid-based clustering method that groups data points into K clusters by minimizing intra-cluster variance. The algorithm assumes spherical cluster shapes, making it effective for datasets with normally distributed features.

   The first step was to find the Optimal Number of Clusters (K). To determine the best K value, two techniques were used:

   - Elbow Method – Analyzes the Within-Cluster Sum of Squares (WCSS) to find the point where adding more clusters does not significantly reduce variance.

- Silhouette Score – Measures how well each data point fits within its assigned cluster. A higher silhouette score indicates better-defined clusters.

Both methods indicated that K = 2 was the optimal number of clusters, suggesting that the dataset is naturally grouped into two distinct driving profiles.

2. DBScan (Density-Based Spatial Clustering of Applications with Noise) by Sander et al. (1998)
   DBScan is a density-based clustering method that groups together points that are densely packed while marking outliers as noise. Since real-world driving data may contain anomalies, DBScan was applied to identify potential outliers, such as trips with extreme driving behaviors, and to evaluate whether a density-based approach provided a better fit compared to centroid-based methods.

3. Gaussian Mixture Models (GMM) by Reynolds (2009)
   GMM is a probabilistic clustering algorithm that models the dataset as a mixture of multiple Gaussian distributions. Unlike K-Means, which assigns each data point to a single cluster, GMM provides a probability distribution over clusters, offering a more flexible approach for data with non-spherical structures.

   For consistency, the number of Gaussian components was set to K = 2, matching the optimal K-Means cluster count.

4. Evidence Accumulation Clustering (EAC) with K-Means by Fred and Jain (2002)
   To further enhance clustering robustness, Evidence Accumulation Clustering (EAC) was tested using K-Means. EAC operates by:

   (a) Running multiple K-Means clusterings with different initializations.

   (b) Constructing a co-association matrix, where each entry represents how often two data points are clustered together.

   (c) Using hierarchical clustering to identify the final stable clusters.

   EAC was tested to verify whether aggregating multiple cluster partitions led to more stable and reliable clusters.

After clustering, each trip was assigned a cluster label, grouping them based on similarities in driving behavior. These labels were later refined in the second clustering stage.

4.5.3 Second Stage Clustering

While the first stage identified broad driving styles, the second stage aimed to refine these clusters by identifying sub-clusters within the original groups. The best-performing clustering method from the first stage was selected for further decomposition. The objective was to determine whether the clusters could be split further into meaningful sub-groups and if a finer granularity of driver profiles could be achieved.

Since the best result of the first stage resulted in two clusters (agressive and safe trips), this stage only selected the trips labeled as agressive for further decomposition. This approach focused on finding the diference between agressive driving styles and driving styles that can be considered dangerous.

Once the second-stage clustering was completed, the trips were labeled according to their refined profile. These final labels were then used in the Explainability Analysis to interpret the clusters' meaning and characteristics.

4.5.4 Explainability Analysis

After clustering, it was essential to interpret the meaning of each cluster, ensuring that the results are aligned with real-world driving behaviors. Three different explainability techniques were applied:

1. PCA Analysis
   Principal Component Analysis (PCA) was used to identify which features contributed most to variance in the dataset and how clusters were distributed in lower-dimensional space.

   By analyzing the first principal component (PC1), we identified the features that had the strongest influence on clustering, helping to understand which driving behaviors differentiated clusters the most.

2. Decision Tree & Random Forest Analysis
   To further interpret cluster separability, clustering was reframed as a classification problem using Decision Tree (DT) classifiers by Quinlan (1986) and Random Forest (RF) models by Breiman (2001).

   By training these models to predict cluster labels based on trip features, the feature importance rankings and decision rules were extracted, revealing which features best distinguished driver profiles and showing how the clusters could be defined in terms of measurable trip characteristics.

   Since decision trees naturally provide interpretability, they allowed us to see which trip parameters were most critical in separating driver styles.

3. Statistical Analysis of Features
   A statistical approach was used to analyze feature distributions within each cluster, providing a deeper understanding of the behavioral differences between groups. For each key feature, we computed:

   - Mean & Standard Deviation – To measure the average behavior and variability within clusters.
   - Minimum & Maximum Values – To detect extreme cases and outliers.

This analysis helped in validating the clusters and identifying distinct driving behaviors within each group.

4.6 Supervised Learning

After identifying driver profiles through unsupervised learning, the next step was to develop a supervised learning model capable of predicting a driver's profile based on their trip characteristics. This phase aimed to train and evaluate different classification algorithms using the labeled dataset generated from the clustering phase.

To ensure an effective learning procedure, a machine learning (ML) pipeline was developed, incorporating key preprocessing and modeling techniques:

1. Normalization – Standardizing the dataset to ensure fair feature comparisons.
2. Resampling – Addressing class imbalance using instance sampling techniques.
3. Dimensionality Reduction – Reducing feature dimensionality for improved model efficiency.
4. Classification – Training and evaluating different machine learning classifiers.

Each of these steps is explained in detail below.

4.6.1 Machine Learning Pipeline

A standardized ML pipeline was implemented to ensure consistency across different classifiers. The pipeline consisted of the following sequential steps:

1. **Normalization**
    - Since the clustering phase normalized trips by distance, the same distance normalization method was applied to maintain consistency.
    - This ensured that all features contributed equally to model predictions, preventing bias from high-magnitude features.
2. **Resampling (Instance Sampling)**
    - The dataset obtained from clustering was imbalanced, meaning that some driver profiles had significantly more samples than others.
    - To address this, three synthetic oversampling techniques were tested:
        - SMOTE (Synthetic Minority Oversampling Technique) – Generates synthetic samples by interpolating between existing minority class instances.
        - Borderline SMOTE – A variation of SMOTE that focuses on generating synthetic samples near the class boundaries.
        - ADASYN (Adaptive Synthetic Sampling) – Generates synthetic samples by giving more weight to difficult-to-learn minority class examples.
    - The impact of each resampling technique was evaluated based on classification performance.

3. **Dimensionality Reduction**
   Since Singular Value Decomposition (SVD) had the best result on the clustering phase, the same approach was followed here to reduce feature dimensionality and to retain the most informative components while reducing computational complexity.

4. **Classification**

   - After preprocessing, the dataset was fed into different supervised learning classifiers.
   - The following classifiers were selected based on their strong performance in the driver profiling literature:
     - Decision Tree (DT) – A tree-based model that provides interpretability by identifying key decision rules for driver profiling.
     - Random Forest (RF) – An ensemble of decision trees that improves robustness and reduces overfitting.
     - Extreme Gradient Boosting (XGBoost) by Chen and Guestrin (2016) – A boosting algorithm known for high predictive accuracy and computational efficiency.
     - Linear Support Vector Machine (SVM) by Cortes and Vapnik (1995) – A powerful classification model that finds an optimal decision boundary between classes.

4.6.2 Evaluation Metrics and Model Selection

To assess model performance, multiple evaluation metrics were computed:

- Accuracy – To measure the proportion of correctly classified trips.
- Precision, Recall, and F1-Score – To evaluate model performance in handling imbalanced classes.
- Area Under the Receiver Operating Characteristic Curve (AUC-ROC) – To assess the classifier's ability to distinguish between different driver profiles.

Each classifier was trained and tested using nested cross-validation, ensuring reliable and unbiased performance estimates.

Chapter 5

Drowsiness Detection Implementation

This chapter presents the practical implementation of the drowsiness detection component of this research, focusing on the methodology employed for detecting driver drowsiness using physiological data. Section 5.1 begins with an analysis of the dataset used for this task, describing its structure, key characteristics, and the process of labeling data based on subjective sleepiness scores reported by the drivers. In Section 5.2, attention turns to the preprocessing and normalization techniques applied to the physiological signals. These methods ensure consistency across samples and enhance the robustness of the models by minimizing signal variability. Section 5.3 addresses the selection of Heart Rate Variability (HRV) features, focusing on identifying the most informative metrics that contribute to both interpretability and model effectiveness. Building on this, Section 5.4 explores dimensionality reduction techniques aimed at simplifying the feature space while preserving the information necessary for accurate classification. Finally, Section 5.5 details the supervised learning approach used to develop drowsiness detection models. This includes training and evaluating classifiers based on the refined feature set, ultimately enabling the system to distinguish between drowsy and alert states with improved precision.

By implementing these steps, this chapter aims to develop reliable and efficient models for detecting driver drowsiness, contributing to safer and more intelligent driver monitoring systems.

5.1 Dataset Analysis

5.1.1 Overview of the Dataset

The dataset used for drowsiness prediction is called *VALU3S* and was collected in a research project, which aimed to analyze driver behavior under different conditions using a driving simulator. The study recruited 20 participants, evenly divided by gender, all with valid passenger car licenses and a minimum driving experience of at least 50,000 km. The participants completed four 60-minute simulated driving sessions, conducted under varying conditions of alertness and light exposure.

To comprehensively assess driver behavior and physiological responses, data was recorded

using multiple sources:

- A driving simulator, which provided real-time vehicle dynamics and driver performance data.
- Physiological Sensors, to record EOG (eye movements and blinks), ECG (heart activity) and EEG (brain activity).
- The Karolinska Sleepiness Scale (KSS) levels, which participants used to report their subjective sleepiness levels at regular intervals.
- Psychomotor Vigilance Task (PVT) tests, which measured reaction times before and after each drive.

Although a wide range of data was collected in the experiment, only a subset of this data was used for this study. The dataset was structured into three main components, each one stored in a separate folder, described as follows:

PVT Data

The PVT (Psychomotor Vigilance Task) data consists of one file for each PVT test performed by each participant. These tests measured reaction time and vigilance before and after each simulated drive. However, since the focus of this study was on KSS-based driver profiling, this data was not used in the current implementation. Nevertheless, integrating PVT data into future research could provide additional insights for improving the labeling process of supervised models.

Driving Simulator Data

The simulatior data consists of one file for each participant's simulated driving test. Each file consists of multiple rows, where each row represents a timestamp in the simulation, recording various driving behavior parameters. The key features in this dataset include position on the road, vehicle speed and acceleration, brake force applied, steering wheel angle and current participant's KSS response.

From the simulator data, the most important feature to build the supervised datasets was the kss_answer, which records the driver's subjective sleepiness level. The remaining driver behavior features were not utilized, since the driver behavior classification used the i-DREAMS dataset build with real-world driving data.

Vitaport Data

The Vitaport data consists of multiple European Data Format (EDF) files. Where each participant's driving session physiological measurements are stored. Each file contains five physiological signals relative to EOG_v (Vertical Eye Movements), EOG_h (Horizontal Eye Movements),

ECG (Electrocardiogram), ECG (Electrocardiogram), Marker (event synchronization data) and a Bat signal.

Since the focus of this thesis is to explore the capability of the Heart Rate Variability features to predict drowsiness, only the ECG and Marker signals were used.

5.1.2 Data Preprocessing

The raw dataset obtained from the *VALU3S* dataset required significant preprocessing to prepare it for machine learning applications. The data came from multiple sources, including electrocardiogram (ECG) recordings, driving simulator data, and KSS assessments. To create a structured supervised dataset, the preprocessing phase was organized into the following steps:

- Electrocardiogram (ECG) Processing – Extracting and aligning ECG signals with simulator data.
- Heart Rate Variability (HRV) Feature Extraction – Extract HRV metrics using different time window lengths.
- KSS Processing – Merging HRV data with sleepiness scores.
- Supervised Dataset Analysis – Verifying datasets integrity and analyzing sleepiness trends.

Electrocardiogram (ECG) Processing

The ECG data was stored in EDF files, with separate files for each participant and test session. However, no timestamps were available to directly synchronize the ECG recordings with the driving simulator data, making alignment a critical preprocessing step. We have proceeded as follows.

1. **Extract the ECG signal**
   Because each file contained five signals, the first step was to extract the ECG signal from each file.
2. **Duration Mismatch Between ECG and Simulator Data**
   To align the ECG recordings with the corresponding driving session, the total duration of the ECG signal was compared with the duration of the simulator data for each test. This comparison revealed a discrepancy in durations, indicating that the ECG recordings were longer than the simulator data. This mismatch is illustrated in Figure 5.1.
3. **Using the Marker Signal for ECG Filtering**
   To resolve this discrepancy, the Marker signal was used to filter the ECG data. The simulator data contains a feature named vitaport, which was leveraged to align the two datasets. By matching the vitaport values in the simulator data with the marker signal in the ECG recordings, it was possible to extract only the portion of the ECG signal that corresponded to the actual driving session.

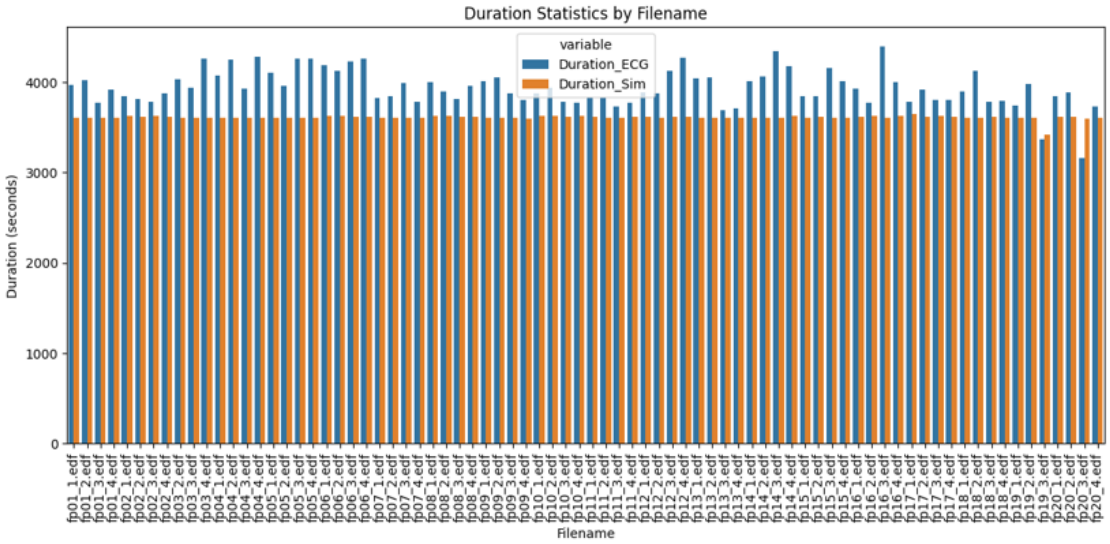

Figure 5.1 Comparison between the duration of the simulator data with the ECG data.

After filtering, a second comparison between the ECG and simulator data durations was conducted. The results, shown in Figure 5.2, confirm that for the most of the files, the durations were successfully aligned after this processing step. However, for some of them it wasn't possible to identify the reason of the mismatch, which led to the removal of these files in the following steps.

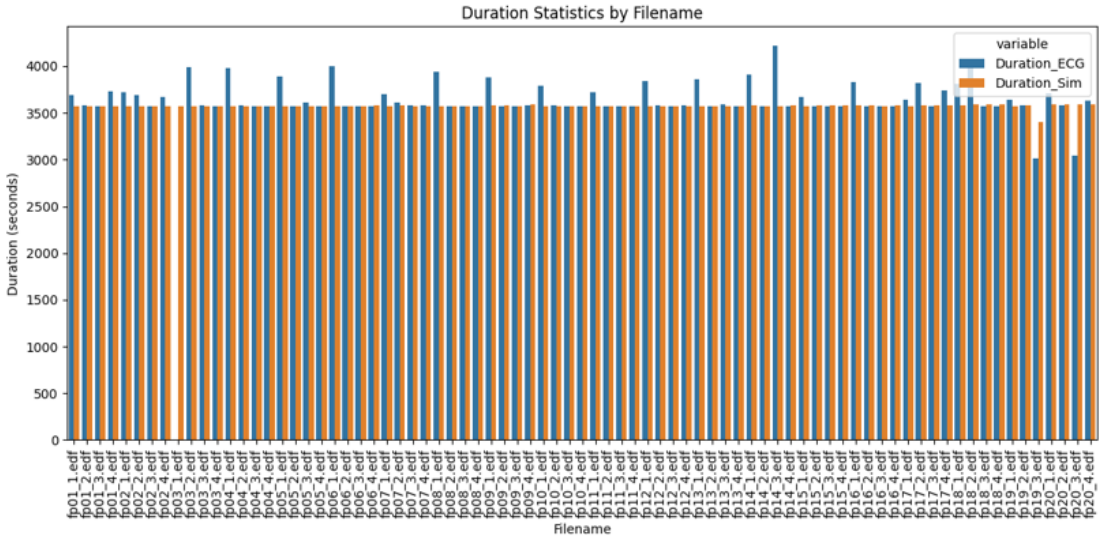

Figure 5.2 Comparison between the duration of the simulator data with the ECG data, after filtering.

4. **Final Filtering of Simulator Data**
   After obtaining the correctly filtered ECG signal, the simulator data was also filtered to ensure it matched the ECG timestamps. Additionally, only three essential features were retained from the simulator dataset for the next steps:

   - timer [s] – The timestamp of each recorded event.
   - kss_answer – The participant's self-reported KSS score.
   - Filename – The identifier linking each record to the corresponding participant and test session.

   This dataset was then used for HRV feature extraction.

HRV Processing

1. **Computing HRV Features**
   After obtaining the filtered ECG for each test, HRV features were extracted using the NeuroKit2 Python library, which provides a robust implementation of both time-domain and frequency-domain HRV metrics.

2. **Selecting Time Windows for HRV Computation**
   Since HRV features are intended for real-time drowsiness prediction, it was essential to select time windows that balance responsiveness with the reliability of the extracted features. Time-domain HRV features can typically be computed over short intervals, while frequency-domain HRV features require longer observation periods to yield meaningful information. With this in mind, two window lengths were chosen for evaluation:

   - 2-minute intervals
   - 5-minute intervals

   he 2-minute window was selected as the shorter interval, as 1 minute was considered potentially too brief even for time-domain metrics to stabilize. On the other hand, the 5-minute window was chosen to provide sufficient duration for frequency-domain features, while still remaining practical for real-world, real-time applications. A window longer than 5 minutes would increase detection latency, reducing the system's responsiveness, which is undesirable in a safety-critical application like drowsiness monitoring.

   For each interval, three different sets of HRV features were computed:

   - Time-domain features only
   - Frequency-domain features only
   - Combined time-domain and frequency-domain features

   This resulted in six different HRV feature datasets, as summarized in Table 5.1:

Table 5.1 Unsupervised HRV datasets

HRV Features	Window Duration	Number of Rows
Time-domain	2 minutes	2179
Time-domain	5 minutes	841
Frequency-domain	2 minutes	2179
Frequency-domain	5 minutes	841
Time and Frequency Domain	2 minutes	2179
Time and Frequency Domain	5 minutes	841

KSS Processing

To create a supervised dataset, the HRV data was merged with the simulator filtered data. Since HRV features were computed over 2-minute and 5-minute intervals, it was necessary to aggregate the corresponding KSS scores from the simulator data. For each HRV interval:

- The mean KSS value was calculated by averaging all KSS responses recorded within that interval.
- The resulting KSS value was then added to the HRV dataset as the ground truth label.

This process was repeated for all six HRV datasets, producing six supervised datasets with labels corresponding to average KSS values. As a result, we have the six final datasets shown in Table 5.2:

Table 5.2 Supervised Datasets with HRV features

Features	Window Duration	Label
HRV Time-domain	2 minutes	KSS value
HRV Time-domain	5 minutes	KSS value
HRV Frequency-domain	2 minutes	KSS value
HRV Frequency-domain	5 minutes	KSS value
HRV Time and Frequency Domain	2 minutes	KSS value
HRV Time and Frequency Domain	5 minutes	KSS value

5.1.3 Final Datasets Analysis

After completing all preprocessing steps, the final dataset consisted of six supervised datasets containing HRV features and their corresponding KSS labels.

A key finding from the dataset analysis was the progressive increase in KSS scores over time, indicating that all participants became drowsier as the tests progressed. This trend is illustrated in Figure 5.3, which shows the change in KSS levels over time for each participant.

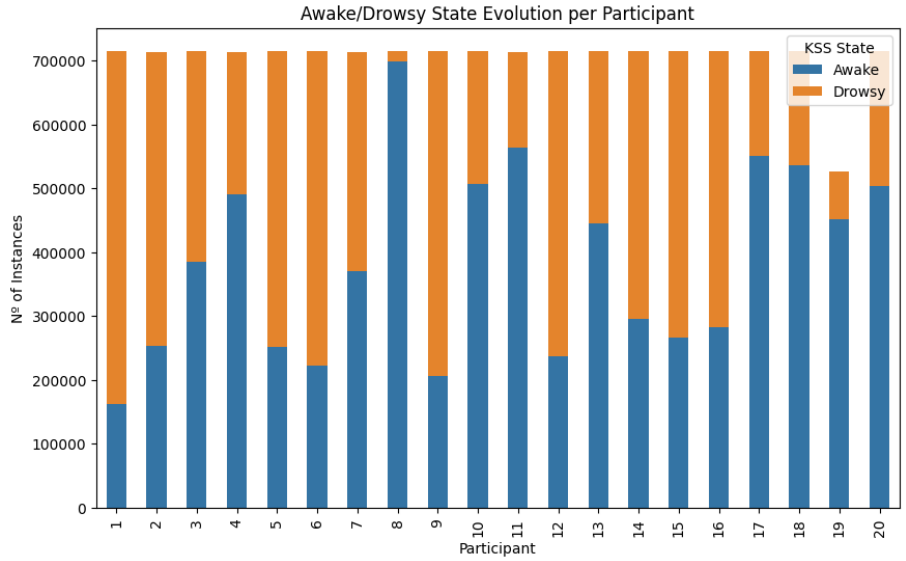

Figure 5.3 Evolution of drowsiness state for each participant during all the tests.

Key Takeaways

- ECG signals were extracted, filtered, and aligned with simulator data using marker signals.
- HRV features were extracted from ECG using 2-minute and 5-minute intervals, resulting in six datasets.
- Supervised datasets were created by merging HRV features with KSS scores.
- The supervised dataset confirms a consistent trend of increasing drowsiness, supporting the reliability of KSS-based classification.

These processed datasets serve as the foundation for the next steps of this thesis, where supervised learning techniques will be applied to predict driver drowsiness levels based on HRV features.

5.2 Dataset Normalization

The Drowsiness Dataset contains only features derived from HRV, which were extracted using the NeuroKit2 library. Since all features are computed from ECG signals, their scales are inherently different based on the HRV computation method (time-domain vs. frequency-domain). To ensure uniformity, StandardScaler normalization was applied to the dataset.

This normalization technique was chosen because:

1. HRV features follow different distributions – Some HRV features have Gaussian-like distributions, while others (especially frequency-domain features) may be skewed. StandardScaler helps standardize all features while preserving their relative distributions.
2. Improves model stability – Many supervised learning models perform better when input features have a zero mean and unit variance, preventing any single feature from dominating predictions.
3. It is essential for distance-based methods – Some machine learing techniques rely on distance metrics, meaning that unscaled features could lead to biased classification.

By applying StandardScaler normalization, the HRV features were prepared for supervised learning, ensuring that machine learning models could effectively learn patterns related to drowsiness classification.

5.3 Feature Selection

Feature selection for the Drowsiness Dataset followed a different approach since the dataset was used for supervised learning instead of clustering. Initially, no feature selection was applied, and all HRV features were used. However, as model development progressed, it became necessary to identify the most important HRV features for drowsiness classification.

Based on findings from the state of the art, a subset of time-domain HRV features was selected, as these features are commonly used for detecting drowsiness. The selected features include:

- MeanNN – Average time interval between heartbeats.
- SDNN – Standard deviation of heartbeat intervals.
- RMSSD – Root mean square of successive heartbeat differences.
- SDSD – Standard deviation of successive heartbeat differences.
- pNN20 & pNN50 – Percentage of heartbeat differences greater than 20 ms and 50 ms.

Finaly, to optimize the drowsiness classification models, different feature selection strategies were also tested. The diferent domains (time-domain, frequency-domain and time and frequency domain combined) datasets, were tested and compared to evaluate which domain provided the best drowsiness detection performance.

5.4 Feature Reduction

Unlike the Driver Profile Dataset, the Drowsiness Detection Dataset already had a small number of HRV features, making feature reduction less critical. However, to assess whether dimensionality reduction could still provide performance improvements, PCA and SVD were tested.

5.5 Supervised Learning

The drowsiness classification task was developed to predict whether a driver is awake or drowsy based on Heart Rate Variability (HRV) features. This phase followed a supervised learning approach, leveraging the six labeled datasets created in the preprocessing stage.

The implementation involved three key stages:

1. Baseline Classification Models – Training and evaluating traditional ML classifiers.
2. Time Series Classification with LSTM – Addressing the temporal dependencies in drowsiness evolution.
3. Final Adjustments – Optimizing the best-performing model for practical usability.

Each of these steps is described in detail below.

5.5.1 Baseline Classification Models

1. **Data Preprocessing**
   Before training the models, the dataset underwent the following preprocessing steps:
   - Removing Non-Relevant Features – features such as filename, interval start, and interval end were removed since they do not contribute to classification.
   - Handling Missing Values – Any remaining missing values were removed to ensure clean training data.
   - Discretization of the KSS Labels – Since the Karolinska Sleepiness Scale (KSS) is a 10-point scale, it was transformed into a binary classification problem as shown in Figure 2.2. Following the state of the art findings the KSS was labeled as Awake for a value less than seven and Drowsy for a value greater than or equal to seven.
   - Normalization – All features were normalized using StandardScaler to ensure they had zero mean and unit variance.
   - Dimensionality Reduction – SVD was applied to reduce dimensionality, while retaining the most significant variance.
2. **Machine Learning Pipeline**
   To maintain consistency across different models, a machine learning pipeline, similar

to the driver profile supervised learning phase, was implemented. It includes StandardScaler normalization, dimensionality reduction with SVD and model training with hyperparameter tuning using nested cross-validation. The same Decision Tree, Random Forest, Extreme Gradient Boosting and Linear Support Vector Machine classifiers were selected and tested on this data, since they showed effectiveness and interpretability on the Driver Profile supervised dataset.

3. **Model Evaluation**
   Each classifier was trained and evaluated using the same performance metrics as in the Driver Profile Supervised Learning phase, namely:
   - **Accuracy**
   - **Precision, Recall, and F1-Score**
   - **AUC-ROC Curve**

4. **Observations from Initial Results**
   While the traditional classifiers achieved reasonable performance, the results suggested that a better approach was needed. The main limitation was that these models treated each HRV instance independently, ignoring the temporal evolution of drowsiness.

   To address this limitation, a time series classification approach was introduced using Long Short-Term Memory (LSTM) networks.

5.5.2 Time Series Classification with LSTM

Since drowsiness is a progressive state, it is influenced by time-dependent patterns rather than isolated HRV measurements. To capture these patterns, an LSTM-based model was implemented, as described in the following.

1. **Reshaping the Data for Sequential Learning**
   The transition from traditional ML to time-series classification required restructuring the dataset. Previously, each instance represented a single 2-minute or 5-minute HRV interval, regardless of other instances.

   For LSTM training, it was necessary to group consecutive time intervals into sequences that captured the temporal progression of drowsiness.

   Sliding Window Approach for Sequence Construction

   To create meaningful time series sequences, a sliding window technique was applied:

   (a) HRV features were recomputed using different interval lengths:
      - 1 min, 2 min, 3 min, 5 min, and 8 min.

   (b) Sequences of consecutive intervals were created, where each sequence contained a window of N previous intervals.

(c) The dataset was reconstructed, where each training sample became a time-dependent sequence of past HRV features instead of an independent instance.

2. **Training the LSTM Model**
   The LSTM architecture was optimized by:

   - Testing different interval durations (1 min, 2 min, 3 min, 5 min, and 8 min) to analyse the impact on model performance.
   - Testing different window sizes (2 to 10) to balance prediction accuracy vs. time efficiency.

5.5.3 Final Adjustments and Model Optimization

1. **Merging Trips for Better Generalization**
   A key observation from data analysis was that each participant completed four consecutive trips, and the largest drowsiness change occurred between the first and last trip.

   To capture this progression, the four trips for each participant were merged into a single time series, improving the model's ability to learn long-term sleepiness patterns.

2. **Model Selection and Final Comparison**
   After training and evaluating different models, the following conclusions were drawn:

   - Traditional ML classifiers (RF, XGBoost and SVM) performed well but lacked time dependency modeling.
   - LSTM outperformed traditional models in capturing time-dependent drowsiness evolution.
   - Adding more time and context of each participant improved the model's ability to learn long-term patterns.

Chapter 6

Evaluation - Driver Profile

Chapter 6 presents the experimental results for the driver profile classification task. Section 6.1 describes the testing environment used to train and evaluate the models. Section 6.2 outlines the normalization process applied to the dataset and K-Means clustering evaluation. In Section 6.3, the results of feature selection are presented, followed by Section 6.4, which explores dimensionality reduction techniques applied to the driver profile dataset. Section 6.5 summarizes the final versions of the datasets used in subsequent experiments. Section 6.6 details the outcomes of the unsupervised learning phase, including results from both the first-stage clustering, which separates safe from risky trips, and the second-stage clustering, which further categorizes risky driving patterns. Finally, Section 6.7 presents the results of supervised learning, including analysis of class imbalance, evaluation of different classifiers, application of supervised dimensionality reduction, and a summary of the final models' performance.

6.1 Testing Enviroment

The experiments were conducted on a desktop PC with the following hardware specifications:

- **Processor -** AMD Ryzen 5 7600X, 6 cores, running at 4.70 GHz
- **Memory (RAM) -** 32.0 GB (31.1 GB usable)
- **System Type -** 64-bit operating system, x64-based processor
- **Operating System -** Windows 11 Home, version 23H2

The code was implemented in the Python programming language, utilizing common libraries for handling datasets and performing machine learning tasks. The environment included packages such as NumPy, Pandas, Scikit-learn, TensorFlow/Keras, and Matplotlib for data processing, model training, and visualization.

6.2 Normalization

To compare the impact of different normalization approaches, a K-Means clustering technique was applied with k = 2, as both the Elbow Method and Silhouette Score indicated that two was the optimal number of clusters. The clustering was performed on both datasets. One with distance normalization and the other with duration normalization without applying any feature reduction or feature selection.

The clustering performance was evaluated using the Calinski-Harabasz (CH) Index, Davies-Bouldin (DB) Index, and Silhouette Score (S). The results, presented in Table 6.1, indicate that the distance-normalized dataset achieved better clustering performance:

Table 6.1 Distance vs Duration Normalization Results

Scores(Euclidean square)	**Distance**	**Duration**
Calinski Harabasz ↑	**6648.037**	3819.919
Davies Bouldin ↓	**1.156**	1.498
Silhouette ↑	**0.485**	0.424
Instances cluster 0	10476	10751
Instances cluster 1	2730	2455
Number of Features	53	53

Higher CH scores, lower DB scores, and higher Silhouette scores suggest better clustering performance. These results confirm that distance normalization leads to superior clustering quality, which aligns with the theoretical discussion presented in the previous chapter.

6.3 Feature Selection

Feature selection was a critical step in reducing the number of features while retaining the most informative and non-redundant ones for clustering and classification.

This section presents the results of feature selection, including the ranking of relevant features, feature reduction statistics, and comparisons between methods.

Two feature selection approaches were applied. Relevance Feature Selection (RFS) to select features based on statistical variance and distribution and Relevance-Redundancy Feature Selection (RRFS) with Absolute Cosine Similarity to select features based on relevance while removing redundant features.

Mean-Median Feature Relevance Results

The Mean-Median Feature Relevance metric was computed for each feature, ranking them based on their absolute mean-median. Figure 6.1 shows the complete 53 features sorted by Mean-Median Feature Relevance.

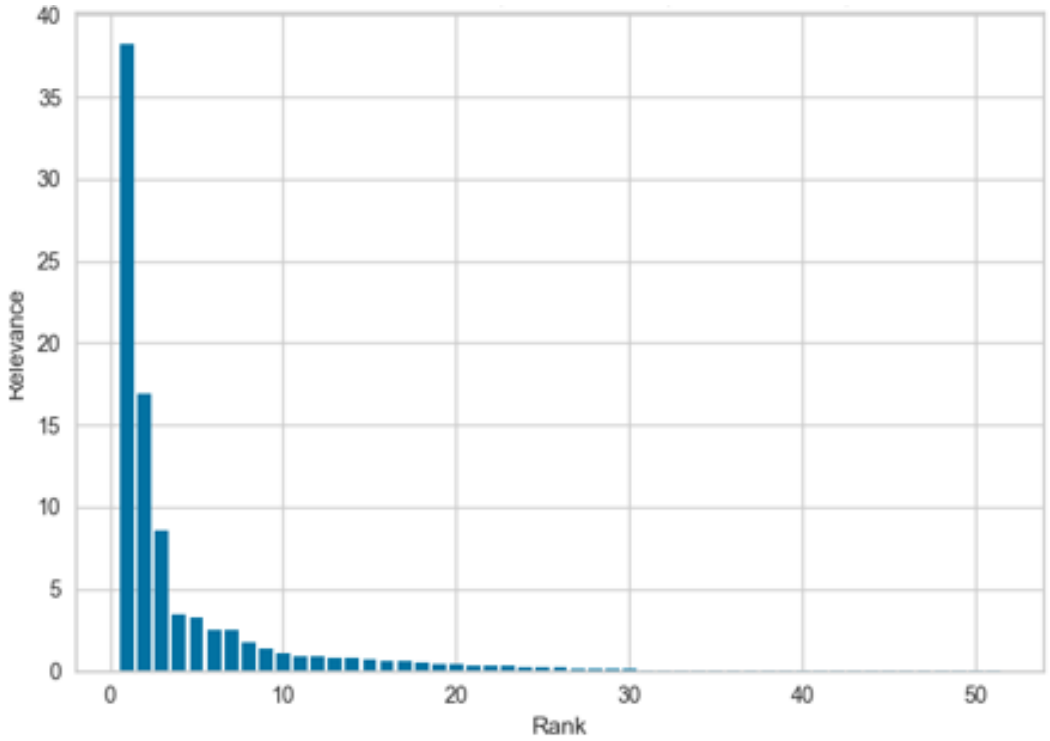

Figure 6.1 Features sorted by Relevance using the Mean-Median metric.

To ensure a balance between dimensionality reduction and information retention, different thresholds were used to retain different percentages of feature relevance. To evaluate what relevance retention percentage should be used, a K-Means clustering approach was used, setting K = 2 and evaluated using the Calinski-Harabasz Index, Davies-Bouldin Index, and Silhouette Score.

The results in Table 6.2 showed that 80% retention of feature relevance achieved the best clustering results.

Table 6.2 Mean-Median Feature Relevance Clustering Results

Scores (Euclidean square)	**99%**	**90%**	**80%**
Calinski Harabasz ↑	6680.274	7372.046	**8290.126**
Davies Bouldin ↓	1.154	1.078	**0.994**
Silhouette ↑	0.485	0.503	**0.530**
Instances cluster 0	10475	10471	10476
Instances cluster 1	2731	2735	2730
Number of Features	27	11	6

This selection reduced the number of features from 53 to 6.

The top 6 Features Ranked by Mean-Median Relevance, are as flollows:

1. **speed -** Mean Speed (km/h).
2. **n_tsr_level -** Number of times the speed limit was exceeded.
3. **n_brakes -** Number of times breaks are ON.
4. **n_hc -** Number of harsh cornering events with low severity.
5. **n_tsr_level_1 -** Number of times 0-5 units over the speed limit.
6. **n_tsr_level_2 -** Number of times 5-10 units over the speed limit.

Relevance-Redundancy Feature Selection (RRFS) with Absolute Cosine Similarity

To further improve feature selection, the Relevance-Redundancy Feature Selection (RRFS) approach was applied to evaluate feature relevance using the same Mean-Median unsupervised metric and to compute feature redundancy using Absolute Cosine Similarity.

Figure 6.2 shows the similarity between the top 10 feature pairs, showing how redundant features were eliminated.

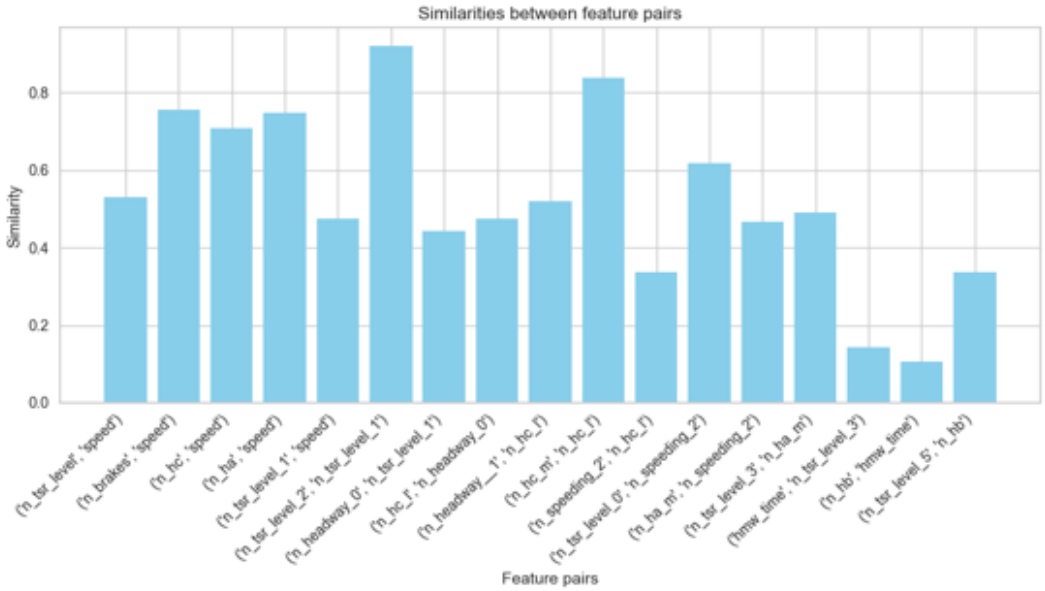

Figure 6.2 Similarity between the top 10 features pairs.

For this approach two redundancy thresholds (M_s=0.5) and (M_s=0.6) were applied, ensuring that redundant features were filtered out, resulting in a more accurate feature subset. Similar to the Mean-Median Feature Relevance a K-Means clustering approach with K=2 was used to evaluate and find the better M_s threshold value.

Table 6.3 shows that M_s=0.5 cleary achieved better results on the clustering task.

With this aproach the feature number was reduced from fifty three to six, similar to the Mean-Median Feature Relevance aporach. Although, the selected features were differente since features like speed and n_brakes have a similarity value above M_s=0.5 so n_brakes will

Table 6.3 Relevance-Redundancy Feature Selection (RRFS) Clustering Results

Scores (Euclidean square)	**0.6**	**0.5**
Calinski Harabasz ↑	7992.497	17830.533
Davies Bouldin ↓	1.045	0.685
Silhouette ↑	0.513	0.612
Instances cluster 0	10484	10636
Instances cluster 1	2722	2570
Number of Features	10	6

be discarted and only speed will be considered since it has a larger Mean-Median Relevance value.

The top 6 Features Ranked by RRFS, are as flollows:

1. **speed -** Mean Speed (km/h).
2. **n_tsr_level_1 -** Number of times 0-5 units over speed limit.
3. **n_headway_0 -** Number of headway level 0 events, vehicle detected.
4. **n_hc_l -** Number of harsh braking events with high severity.
5. **n_speeding_2 -** Number of speeding level 2, events visual speeding warning.
6. **n_ha_m -** Number of harsh braking events with medium severity.

6.4 Feature Reduction

Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) were tested as Feature Reduction techniques. For this dataset PCA and SVD were tested in a clustering aproach, assessing their impact on K-Means clustering quality.

Before applying PCA and SVD, distance normalization was applied, as it was determined, in the Section 4.2.1, to be the best normalization strategy for this dataset.

Both PCA and SVD were then applied to the 53 normalized features, reducing the dataset to 17 features in both cases.

To assess the impact of PCA and SVD, a K-Means clustering approach was used, setting K = 2 as determined in the previous clustering experiments. The cluster quality was evaluated using the Calinski-Harabasz Index, Davies-Bouldin Index, and Silhouette Score.

The results are presented in Table 6.4.

Table 6.4 PCA and SVD Clustering Results

Scores (Euclidean square)	PCA	SVD	No Reduction
Calinski Harabasz ↑	6706.514	**6706.586**	6648.037
Davies Bouldin ↓	**1.150**	**1.150**	1.156
Silhouette ↑	**0.486**	**0.486**	0.485
Instances cluster 0	10477	10478	10476
Instances cluster 1	2729	2728	2730
Number of Features	17	17	53

The results showed that both PCA and SVD achieved the same clustering performance, indicating that both techniques effectively retained the key driving behavior patterns. However, the dimensionality was reduced from 53 to 17, providing a more compact dataset while preserving clustering effectiveness.

Since PCA and SVD performed equally well, either method could be used. However, for computational efficiency, SVD was preferred, as it is generally more efficient for large-scale datasets.

6.5 Final Datasets

For this dataset, the experimental results showed that distance normalization was better then duration normalization, ensuring that features were properly scaled for clustering. Therefore, only distance normalization was applied to the subsequent feature selection and feature reduction steps.

Table 6.5 summarizes the results for different feature selection and feature reduction approaches. The Relevance-Redundancy Feature Selection (RRFS) method achieved the best clustering performance, with a Calinski Harabasz score of 17,830.533, a Davies-Bouldin score of 0.685, and a Silhouette score of 0.612, selecting only six features from the original 53. These results highlight the importance of removing redundant features to enhance the clustering structure, making RRFS the best approach for feature selection.

Table 6.5 Comparison of Feature Selection and Reduction Methods

Scores (Euclidean square)	Distance Normalization				No Reduction
	Feature Selection		Feature Reduction		
	RFS	RRFS	PCA	SVD	
Calinski Harabasz ↑	8290.126	**17830.533**	6706.514	6706.586	6648.037
Davies Bouldin ↓	0.994	**0.685**	1.150	1.150	1.156
Silhouette ↑	0.530	**0.612**	0.486	0.486	0.485
Number of Features	6	6	17	17	53

On the other hand, PCA and SVD for feature reduction both resulted in lower clustering performance than RRFS, with Calinski Harabasz scores around 6,700 and Silhouette scores of 0.486, showing that reducing dimensionality to 17 features was not as effective as selecting six highly relevant and non-redundant features. The non-reduced dataset (53 features) was also tested as a baseline, and its performance was worse than RRFS, reinforcing the prominence of feature selection over feature reduction for this dataset.

Thus, in order to extract the maximum potential of the tested approaches, the best methods were combined in the unsupervised and supervised learning phases. Distance normalization was applied, followed by RRFS Feature Selection and finally the use of SVD to reduce the number of features.

6.6 Unsupervised Learning Results

6.6.1 First Stage Clustering Results

As concluded in the Final Pre-Processed Datasets section, the best pre-processing pipeline for the Driver Profile dataset consisted of distance normalization, Relevance-Redundancy Feature Selection (RRFS), and Singular Value Decomposition (SVD) for feature reduction. This final processed dataset was used in the first stage clustering, where the goal was to identify aggressive and safe driving styles by applying different clustering algorithms.

The clustering methods evaluated were K-Means, DBScan, Gaussian Mixture Models (GMM), and Evidence Accumulation Clustering (EAC). The performance of each algorithm was measured using Calinski-Harabasz, Davies-Bouldin, and Silhouette scores, with results summarized in Table 6.6.

Table 6.6 First Stage Clustering Results

Scores	**K-Means**	**DBScan**	**GMM**	**EAC**
Calinski Harabasz	**18274.235**	4273.523	4528.218	422.600
Davies Bouldin	**0.673**	1.229	1.169	0.933
Silhouette	**0.616**	0.491	0.473	0.560
Instances cluster 0	10632	10165	9524	13111
Instances cluster 1	2574	3041	3682	95

The results showed that K-Means achieved the best clustering performance, DBScan resulted in weaker cluster separability, Gaussian Mixture Models (GMM) performed slightly better than DBScan but still had lower overall scores than K-Means and Evidence Accumulation Clustering (EAC) had the lowest Calinski-Harabasz score, suggesting poor separation of clusters.

Given these results, K-Means was selected as the best clustering approach for this stage.

To better understand the cluster separation, a scatter plot was generated (Figure 4.4), showing the distribution of instances based on four key features: speed, n_tsr_level_1, n_headway_0,

and n_hc_l. The visualization suggests that these features play an important role in distinguishing the two clusters.

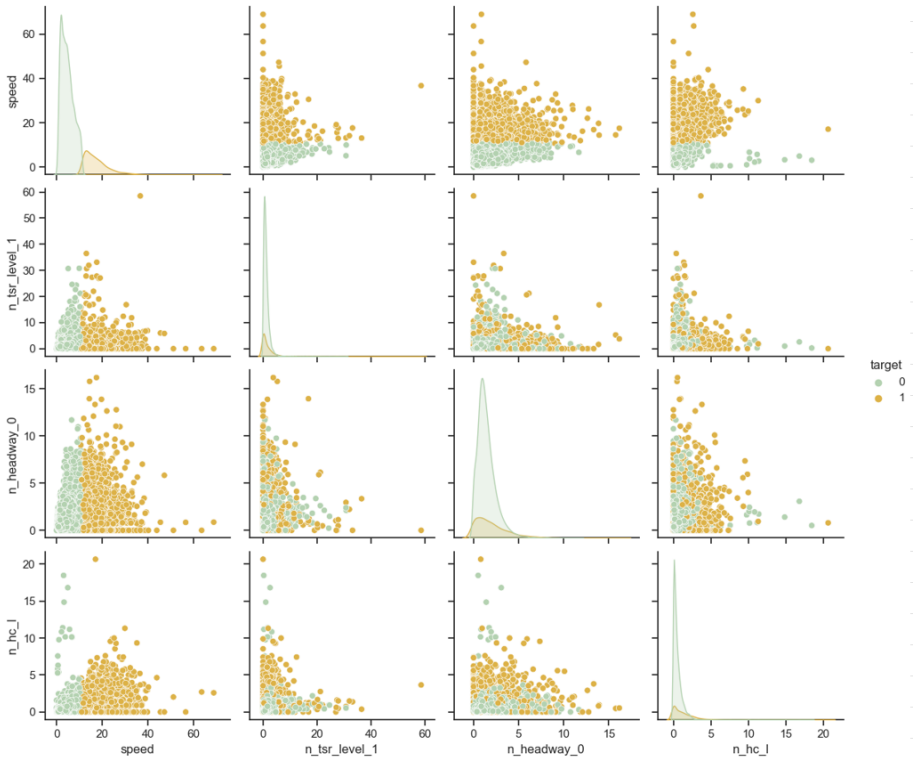

Figure 6.3 Distribution of instances based on key features.

Figure 6.3 shows us that Cluster 0 represents trips with lower speeds (km/h) and fewer speeding events per km, while Cluster 1 represents trips with low to medium speeds but a higher number of braking events per km, indicating a more aggressive driving style.

To further validate and interpret the clusters, two explainable supervised models were trained: a Decision Tree (DT) classifier and a Random Forest (RF) classifier. The top three most important features identified by these models were:

- **Decision Tree (DT) -** speed, n_ha_m, and n_ha.
- **Random Forest (RF) -** speed, n_fatigue_0, and n_overtaking_0.

This reinforces that speed and event-based metrics (e.g., braking, headway, and fatigue-related features) are key discriminators of driving behavior.

Finally, a statistical comparison of the minimum, maximum, average, and standard deviation values for the most important features was conducted (Table 6.7). These statistics further confirm that Cluster 1 exhibits more aggressive driving behavior, as compared to Cluster 0.

Table 6.7 Top Features Statistical Comparison

Feature	Cluster 0				Cluster 1			
	MIN	MAX	MEAN	STD	MIN	MAX	MEAN	STD
speed	0.0	12.47	4.62	2.68	5.12	69.0	17.15	5.85
n_brakes	0.0	23.3	2.44	1.97	0.0	50.85	5.72	4.26
n_tsr_level	0.0	37.31	2.9	2.78	0.0	110.6	4.19	7.16
n_fatigue_0	0.0	1.11	0.13	0.1	0.13	2.8	0.57	0.22
n_overtaking_0	0.0	2.0	0.18	0.15	0.13	5.46	0.64	0.42

In conclusion, Cluster 1 was assigned as aggressive driving, while Cluster 0 was labeled as non-aggressive driving. This categorization serves as the foundation for the next stage of clustering, where a finer-grained analysis of driver behavior will be conducted.

6.6.2 Second Stage Clustering

The second stage clustering was applied only to the aggressive trips identified as Cluster 1 from the first stage of clustering. The objective of this step was to further distinguish different types of aggressive driving behavior. Given that K-Means achieved the best results in the first stage, it was selected again, using k = 2 to refine the categorization of aggressive driving styles.

Since the feature selection and feature reduction techniques had previously been tested for the entire dataset, it was necessary to re-evaluate the best combination of methods specifically for this subset of data. Table 6.8 presents the K-Means clustering performance scores using distance normalization.

Table 6.8 Second stage K-Means clustering results

Dim. reduction	CH	DB	Silhouette	Cluster 1.1	Cluster 1.2
PCA	754.095	1.249	0.519	2527	203
RFS	753.757	1.251	0.5184	2487	205
RFS + PCA	766.422	**0.536**	**0.729**	2655	37
RRFS	1740.735	0.848	0.576	2223	347
RRFS + SVD	**1764.792**	0.837	0.577	2228	346

From these results, RRFS combined with SVD yielded the best clustering performance, under the CH metric. Therefore, this approach was used to analyze and interpret the final clusters.

To understand the clusters obtained in this second stage, the top six features selected through RRFS with M_s=0.5 were:

1. n_tsr_level – Number of times the speed limit was exceeded (Mobileye Advanced Warning System feature).

2. n_hc – Number of harsh cornering events (Driving Behavior feature).
3. n_speeding_2 – Number of overtaking level 2 events (visual and auditory warning, Speeding feature).
4. n_headway_0 – Number of headway level 0 events (vehicle detected, Headway feature).
5. n_tsr_level_3 – Number of times the driver was 10-15 units over the speed limit (Mobileye Advanced Warning System feature).
6. n_pedestrian_dz – Number of times a pedestrian was detected in the danger zone (Mobileye Advanced Warning System feature).

By analyzing these features, it was observed that Cluster 1 recorded a significantly higher number of speeding events per km. Specifically, n_tsr_level_3 was much more frequent in Cluster 1, indicating that this group contained more cases where the driver exceeded the speed limit by 10-15 units. This led to the categorization of Cluster 1 as containing riskier trips than Cluster 0.

To better differentiate the groups, the trips were labeled as "aggressive " in Cluster 0 and as "risky " in Cluster 1, since risky trips had more frequent and severe speeding events compared to aggressive trips.

For better visualization of the Second Stage Clustering, Figure 6.4 presents a scatter plot of the speed, n_tsr_level_1, n_headway_0, and n_hc_l features, displaying the distribution of instances across the three final clusters.

Since RRFS followed by SVD provided the best clustering performance, as indicated in Table 6.9, Figure 6.5 shows one possible visualization of the final clustering results, with the three clusters. The three axis chosen for this plot are the speed, n_brakes, and n_tsr_level features while target 0 represents safe trips, target 1 agressive trips and target 2 risky trips.

Additionally, Table 6.10 shows that the number of trips per cluster remained relatively stable regardless of the dimensionality reduction technique. However, the final dataset is imbalanced, with the majority of trips classified as non-aggressive, a smaller proportion classified as aggressive and a only a few number of trips classified as risky.

To conclude this second stage clustering, Table 6.11 presents the statistical analysis of the most important features across the three clusters, reporting minimum, maximum, average, and standard deviation values. It was observed that Cluster 2 (risky trips) consistently had the highest mean and standard deviation values across all key features, further reinforcing the distinction between aggressive and risky driving behaviors.

With this final three-cluster categorization, the dataset is now labeled and prepared for supervised learning tasks.

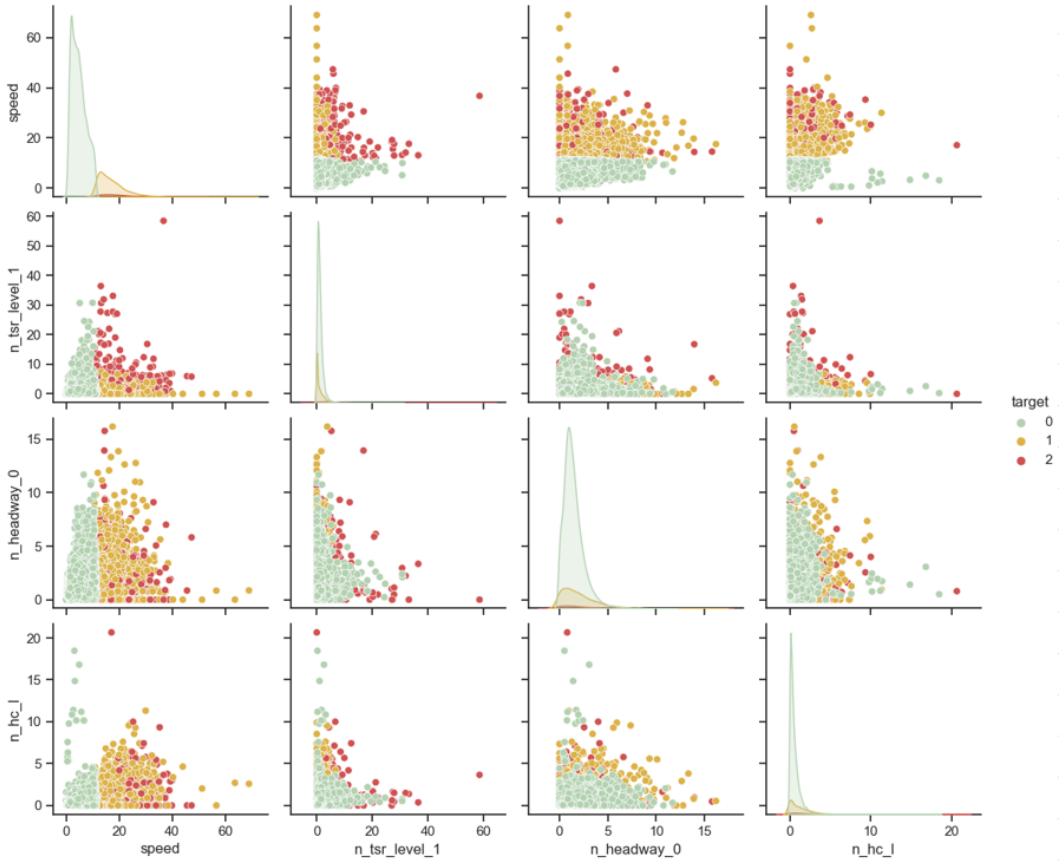

Figure 6.4 Distribution of instances based on key features.

6.7 Driver Profile Supervised Learning

After labeling the dataset through unsupervised clustering, the next step was to apply and evaluate supervised learning techniques for the driver style classification. Given the unbalanced nature of the dataset (c = 3 classes: non-aggressive, aggressive, and risky trips), the main objective was to assess the effectiveness of different ML classifiers and to determine the best-performing model.

6.7.1 Class Imbalance

Since the dataset was imbalanced, the first step was to evaluate the impact of resampling techniques by comparing different instance sampling methods in combination with multiple classifiers. Table 6.11 reports the results of these experiments. Among all the tested sampling strategies, SMOTE (Synthetic Minority Oversampling Technique) obtained the best performance across all classifiers. Given these findings, SMOTE was chosen as the instance sampling technique for balancing the dataset before training the models.

6.7.2 Classifiers Performance Evaluation

Following the selection of SMOTE for data balancing, the next step was to compare different classifiers using nested cross-validation. Table 6.12 presents the average performance metrics

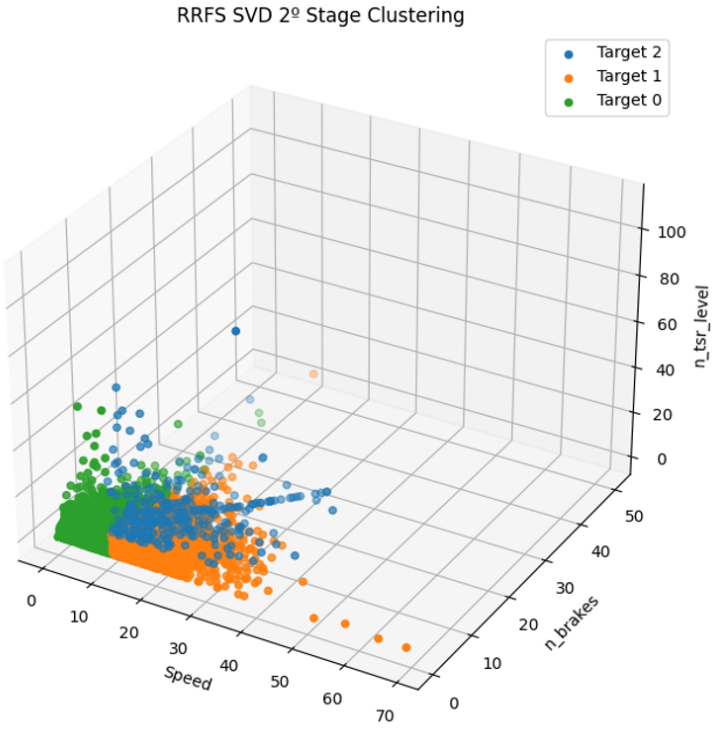

Figure 6.5 Distribution of instances based on key features.

Table 6.9 Top Features Statistical Comparison

Feature	Cluster 0				Cluster 1				Cluster 2			
	MIN	MAX	MEAN	STD	MIN	MAX	MEAN	STD	MIN	MAX	MEAN	STD
speed	0.0	12.47	4.62	2.68	5.12	69.0	17.15	5.85	5.12	69.0	17.15	5.85
n_brakes	0.0	23.3	2.44	1.97	0.0	50.85	5.72	4.26	5.12	69.0	17.15	5.85
n_tsr_level	0.0	37.31	2.9	2.78	0.0	110.6	4.19	7.16	5.12	69.0	17.15	5.85
n_fatigue_0	0.0	1.11	0.13	0.1	0.13	2.8	0.57	0.22	5.12	69.0	17.15	5.85
n_overtaking_0	0.0	2.0	0.18	0.15	0.13	5.46	0.64	0.42	5.12	69.0	17.15	5.85

for all tested classifiers and Figure 6.6 displays the corresponding confusion matrices.

Table 6.10 Performance of supervised classification with DT, RF, XGBoost, and SVM using instance sampling techniques and evaluated by Accuracy (ACC), Precision (PREC), Recall (REC), F1, and AUC scores.

Algorithm	**Random Oversampling**					**SMOTE**				
	ACC	PREC	REC	F1	AUC	ACC	PREC	REC	F1	AUC
DT	0.995	0.995	0.995	0.995	0.995	0.995	0.996	0.995	0.995	0.997
RF	0.994	0.994	0.994	0.994	**1.000**	0.995	0.995	0.995	0.995	**1.000**
XGBoost	0.998	0.998	0.998	0.998	**1.000**	**0.999**	**0.999**	**0.999**	**0.999**	**1.000**
SVM	0.997	0.997	0.997	0.997	0.999	0.997	0.997	0.997	0.997	0.999

Algorithm	**Borderline SMOTE**					**ADASYN**				
	ACC	PREC	REC	F1	AUC	ACC	PREC	REC	F1	AUC
DT	0.992	0.992	0.992	0.992	0.997	0.991	0.991	0.991	0.991	0.995
RF	0.995	0.996	0.995	0.995	**1.000**	0.994	0.994	0.994	0.994	**1.000**
XGBoost	**0.999**	**0.999**	**0.999**	**0.999**	**1.000**	0.997	0.997	0.997	0.997	**1.000**
SVM	0.997	0.997	0.997	0.997	**1.000**	0.997	0.997	0.997	0.997	**1.000**

Table 6.11 Nested cross-validation experimental results.

Algorithm	Accuracy	Precision	Recall	F1	AUC
DT	0.990	0.990	0.990	0.990	0.996
RF	0.994	0.994	0.994	0.994	**1.000**
XGBoost	**0.998**	**0.998**	**0.998**	**0.998**	**1.000**
SVM	0.996	0.996	0.996	0.996	**1.000**

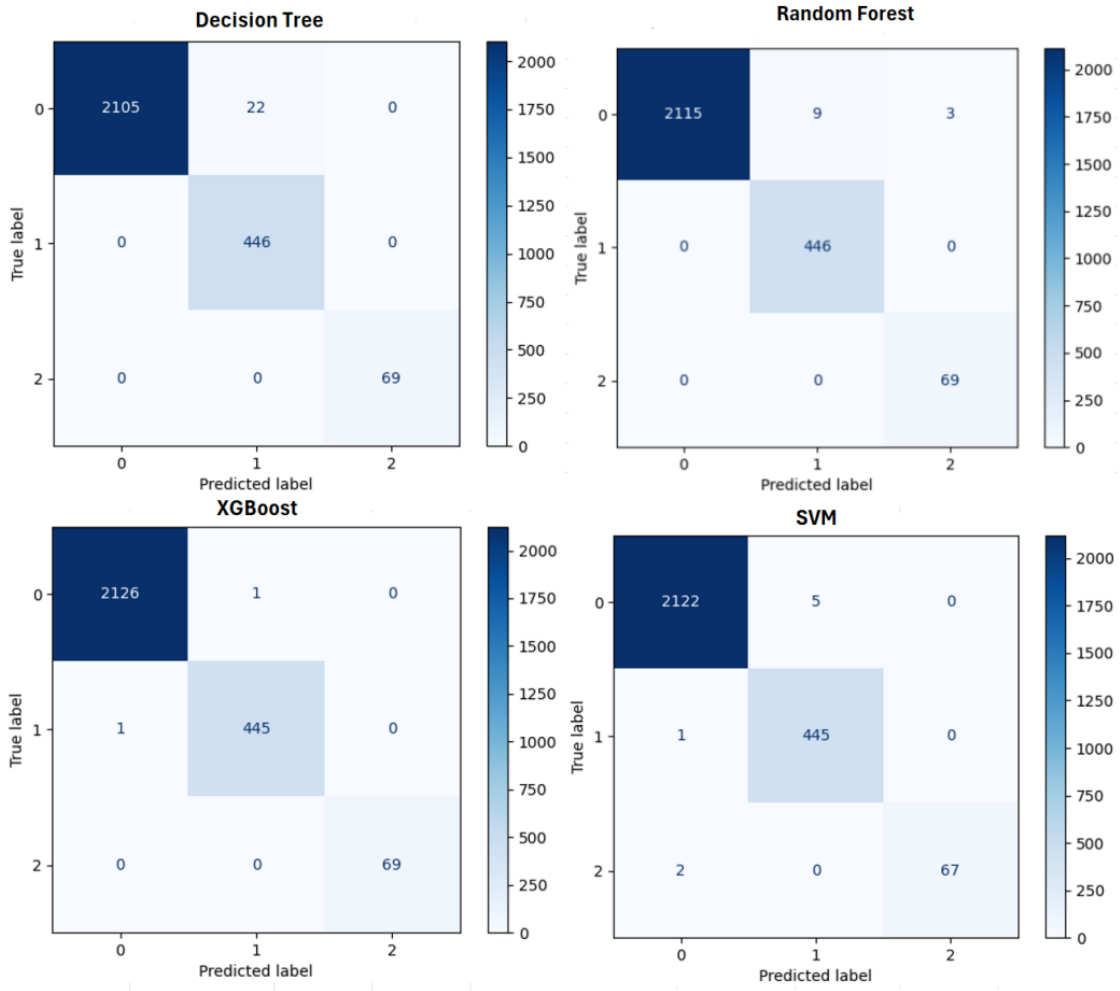

Figure 6.6 DT, RF, XGBoost and SVM confusion matrices.

The results show that XGBoost achieved the best overall performance, with only two misclassification cases. Decision Tree (DT) and Random Forest (RF) misclassified instances from class 0 (non-aggressive trips) but achieved perfect classification for classes 1 (aggressive) and 2 (risky). XGBoost exhibited one error in class 0 and one error in class 1, making it slightly better than DT and RF. SVM show higher misclassification rates across all classes, indicating poorer performance.

These findings highlight XGBoost as the best-performing model for the driver style classification task.

6.7.3 Supervised Dimensionality Reduction

To further improve the classification process, supervised dimensionality reduction techniques were applied to the dataset using Relevance Feature Selection (RFS) and Relevance-Redundancy Feature Selection (RRFS). Both techniques were evaluated with the Fisher's Ratio (FiR) or Fisher Score as the supervised relevance metric.

Table 6.12 presents the classification results of XGBoost (the best-performing model from previous experiments) after applying RFS and RRFS. Figure 6.7 shows the top-ranked features selected by RRFS, sorted by decreasing relevance.

Table 6.12 Experimental results for the XGBoost classifier, after dimensionality reduction with RFS(FiR) and RRFS(FiR).

Class	RFS (FiR), m = 10				RRFS (FiR), m = 6, (M_s=0.5)				RRFS (FiR), m = 10, (M_s=0.6)			
	Precision	Recall	F1	Support	Precision	Recall	F1	Support	Precision	Recall	F1	Support
0	**1.000**	0.997	0.998	2127	**1.000**	**0.998**	**0.999**	2127	**1.000**	0.997	0.998	2127
1	**0.991**	**0.998**	**0.994**	446	0.975	0.953	0.964	446	0.985	**0.998**	0.991	446
2	0.958	**1.000**	0.979	69	0.744	0.884	0.808	69	**0.986**	**1.000**	**0.993**	69
accuracy			**0.997**	2642			0.988	2642			**0.997**	2642
macro avg	0.983	**0.998**	0.990	2642	0.906	0.945	0.923	2642	**0.990**	**0.998**	**0.994**	2642
weighted avg	**0.997**	**0.997**	**0.997**	2642	0.989	0.988	0.988	2642	**0.997**	**0.997**	**0.997**	2642

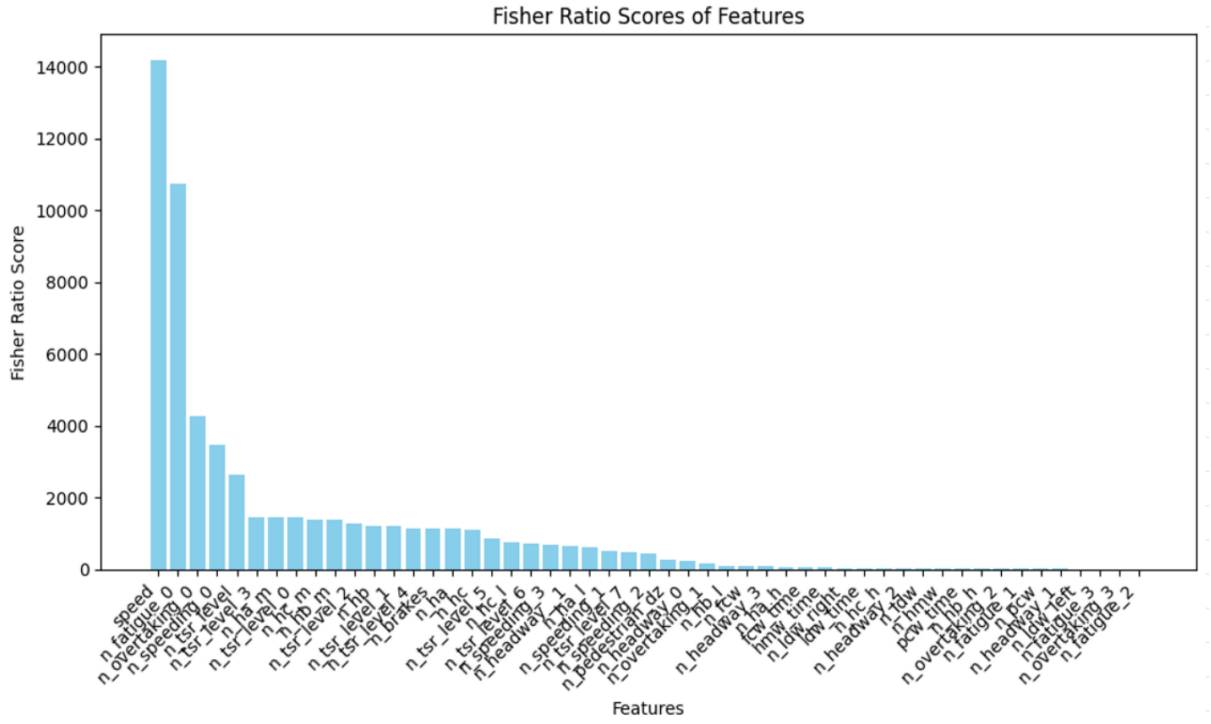

Figure 6.7 Top features selected by RRFS, using the supervised Fisher ratio as the relevance metric.

The top four selected features were speed, n_fatigue_0, n_overtaking_0, and n_speeding_0. Features that show coherence with those identified in the unsupervised learning phase, reinforcing their importance in distinguishing driver behavior.

6.7.4 Final Models

In conclusion, Table 6.13 summarizes the final experimental results of the best models found. Notably the classification scores remained consistent across different methods, indicating that feature selection did not significantly impact overall. Interestingly, the use of dimensionality reduction improved the performance of the SVM classifier, suggesting that feature selection helped mitigate its previous limitations in performance.

Table 6.13 Experimental results of the best models for diferent Fisher Ratio best scorring features.

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1**	**AUC**
SVM (top-20)	0.998	0.998	0.998	0.998	1.000
SVM (top-20 and RRFS (M_s=0.6))	0.998	0.998	0.998	0.998	1.000
SVM (top-10 and RRFS (M_s=0.6))	0.998	0.998	0.998	0.998	1.000
XGBoost (top-20)	0.998	0.998	0.998	0.998	1.000
XGBoost (top-6 and RRFS (M_s=0.6))	0.998	0.998	0.998	0.998	1.000
XGBoost (top-6 and RRFS (M_s=0.6))	0.998	0.998	0.998	0.998	1.000

Chapter 7

Evaluation - Drowsiness

Chapter 7 focuses on the experimental results obtained from the drowsiness detection task using physiological data. Section 7.1 presents the results of the HRV feature selection process, followed by Section 7.2, which covers dimensionality reduction methods applied to the dataset and evaluated by a Random Forest Classifier. Section 7.3 summarizes the final datasets used for training and evaluation. Section 7.4 then details the results of the supervised learning experiments, beginning with baseline classification models and followed by the application of time series models using Long Short-Term Memory (LSTM) networks to classify drowsy and alert states based on sequential physiological data.

7.1 Feature Selection

Unlike the Driver Profile dataset, feature selection for the Drowsiness dataset followed a manual selection approach based on the state-of-the-art research. Since HRV features have documented relationships with drowsiness detection, features were manually selected based on their proven relevance in scientific literature.

Overall, for each of the six supervised HRV datasets, most HRV features were retained, while non-informative features like start and end time of the interval or filenames were removed.

7.2 Feature Reduction

For the Drowsiness Dataset, PCA and SVD were tested in the context of supervised classification rather than clustering. A Random Forest classifier was used to compare the effectiveness of each feature reduction method.

Table 7.1 presents the results obtained from applying PCA, SVD and no reduction before training a Random Forest classifier for the HRV Time and Frequency 2 min dataset.

The results indicate that applying feature reduction (PCA or SVD) led to a decrease in classification performance compared to using the full feature set. While SVD outperformed PCA, it still resulted in lower accuracy and predictive power than the no reduction approach. This suggests that the original feature space already contained an optimal balance of infor-

Table 7.1 PCA and SVD Supervised Learning Results

Random Forest Classifier	PCA	SVD	No Reduction
Accuracy	0.652	0.702	**0.738**
Precision	0.632	0.709	**0.731**
Recall	0.623	0.623	**0.702**
F1	0.627	0.663	**0.716**
Number of Features	14	14	27

mation and dimensionality, and reducing it further led to the loss of valuable signal patterns. Therefore, for the supervised learning phase of the drowsiness detection task, no feature reduction will be applied, ensuring that all relevant HRV features are retained for model training and evaluation.

7.3 Final Datasets

For the Drowsiness dataset, the pre-processing steps were more straightforward, as the dataset originated from HRV features, which are inherently well-structured compared to the diverse feature set of the Driver Profile dataset. A simple StandardScaler normalization was applied to ensure uniform feature scaling.

Unlike the Driver Profile dataset, feature selection was done manually, based on state-of-the-art research on HRV and drowsiness detection. Each of the six datasets had different initial feature counts, but only the most informative HRV features were retained.

For feature reduction, PCA and SVD were evaluated using a Random Forest classifier, but both methods led to worse classification performance compared to using the full feature set (as shown in Table 7.1). Therefore, no feature reduction was applied in the supervised learning tasks for drowsiness detection, ensuring that all relevant physiological information was preserved.

7.4 Supervised Learning Results

The Drowsiness Supervised Learning task was designed to predict whether a driver is awake or drowsy based on Heart Rate Variability (HRV) features. As previously discussed, we worked with six labeled datasets covering time domain, frequency domain, and combined time-frequency domain features with 2-minute and 5-minute intervals.

7.4.1 Baseline Classification Models Results

To initially evaluate the datasets, we trained and tested Baseline Classification Models in a simplified manner. As described in the implementation chapter, the datasets were:

- Preprocessed by removing non-relevant features, handling missing values, and applying normalization.
- Maintained in their original feature space, since dimensionality reduction techniques resulted in worse performance during earlier experiments.

A Machine Learning Pipeline, similar to the one used in the Driver Profile Supervised Learning task, was implemented. For the initial evaluation, a Random Forest classifier was selected and tested on all six datasets.

Random Forest Results

The classification results for Random Forest are reported in Table 7.2, while Figure 7.1 displays the ROC curves for each dataset.

Table 7.2 Random Forest classification experimental results for all datasets.

Dataset Domain	Accuracy	Precision	Recall	F1	AUC
Time - 2 min	**0.761**	0.748	**0.741**	**0.745**	0.82
Time - 5 min	0.702	0.666	0.684	0.675	0.78
Frequency - 2 min	0.563	0.546	0.419	0.474	0.59
Frequency - 5 min	0.672	0.690	0.5	0.58	0.67
Time + Frequency - 2 min	0.753	0.737	0.740	0.738	**0.83**
Time + Frequency - 5 min	0.755	**0.777**	0.644	0.705	**0.83**

These findings highlight that time-domain features consistently outperformed frequency-domain features.Moreover, 2-minute interval datasets achieved higher accuracy than 5-minute intervals, except in the frequency domain, where the 5-minute interval performed better, aligning with the state-of-the-art findings, where frequency-domain features require longer time windows to capture informative patterns effectively. It was also concluded that combining time and frequency domains did not improve classification performance, meaning that using time-domain features alone is preferable as it reduces feature complexity while maintaining high accuracy.

Overall, these results validate the use of HRV features for drowsiness prediction, but at this stage, we believed that further improvements were possible.

As discussed in the implementation chapter, drowsiness is a progressive state rather than an isolated event. This means that it is influenced by time-dependent patterns in HRV measurements, rather than individual snapshots of HRV features. Recognizing this, the next step was to explore time-series classification models, particularly Long Short-Term Memory networks (LSTM), to capture temporal dependencies in the data.

Since it was evident that drowsiness prediction is a time-series problem, other traditional classifiers such as Decision Tree (DT), Extreme Gradient Boosting (XGBoost), and Support Vector Machine (SVM) were not tested, as they do not inherently model temporal dependencies. Instead, we proceeded directly to LSTM-based models for further analysis.

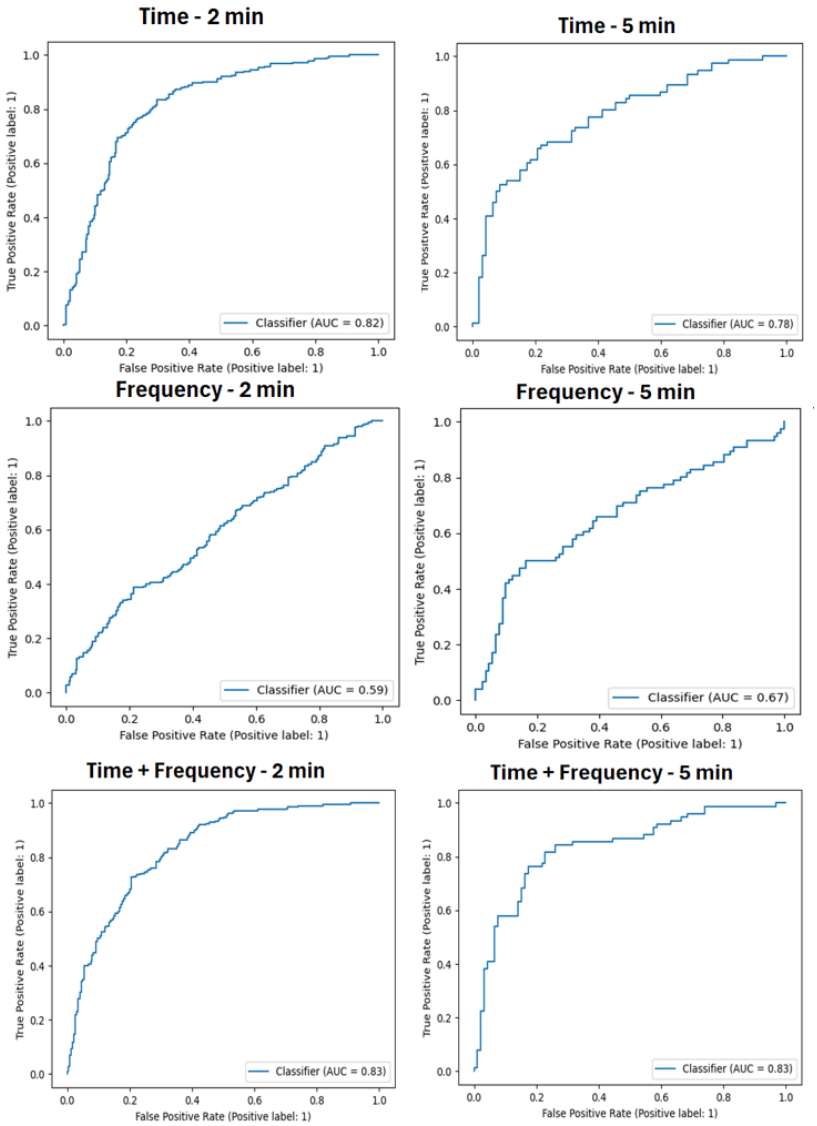

Figure 7.1 Random Forest ROC Curve for each dataset.

7.4.2 Time Series Classification with LSTM Results

In the Baseline Classification Models, each instance represented a single HRV interval (2 min or 5 min), independent of other instances. However, drowsiness is a progressive state, meaning that capturing temporal dependencies in HRV data is crucial for better prediction.

Since the 2-minute time-domain dataset performed best in the Baseline Classification Models, we first tested this dataset under LSTM. However, instead of treating each 2-minute instance independently, we grouped consecutive time intervals into sequences to capture the progression of drowsiness over time.

For sequence creation, we initially set a window size of 10 intervals (each 2 min), meaning a total of 20 min of HRV data per sequence. The preprocessing remained the same as in the Baseline Classification Models, except for the creation of overlapping sequences of size 10.

This initial LSTM model was created and trained. The results are presented in Table 7.3, and Figure 7.2 shows the Confusion Matrix and ROC curve.

Table 7.3 LSTM classification experimental results for Time Domain 2 min interval dataset with a window size of 10.

Evaluation Metrics	Time Domain - 2 min - Window Size of 10
Accuracy	0.838
Precision	0.815
Recall	0.848
F1	0.831
AUC	0.92

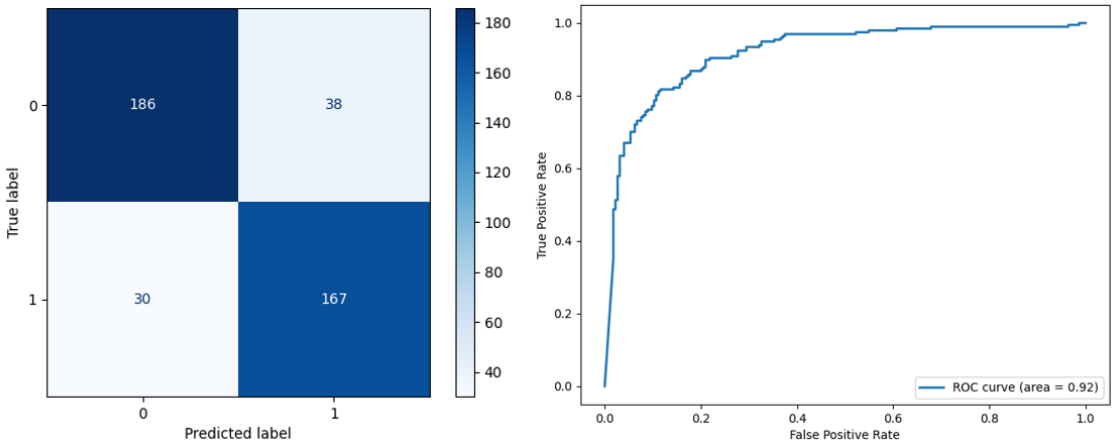

Figure 7.2 LSTM Model Confusion Matrix and ROC curve for the Time Domain 2 min interval dataset with a window size of 10.

This result outperformed all Baseline Classification Models, where the best accuracy was 0.761. This confirmed that treating drowsiness prediction as a time-series problem and incorporating previous HRV measurements significantly improved performance.

Since this initial LSTM model required 20 minutes of ECG data to make a prediction, the next step was to reduce the time needed for classification while maintaining high performance.

Firstly, to explore the impact of interval duration, we tested different interval lengths (1 min, 2 min, 3 min, 5 min, and 8 min), to train 5 different models with a fixed window size of 10. Figure 7.3 presents the models accuracy results.

The results show that shorter intervals performed better than longer intervals. As interval duration increased, accuracy decreased, confirming that shorter time intervals capture drowsiness progression more effectively.

The next step was to find the best intreval duration and window size combination. To do so, the previous intreval durations were combined with different window sizes (2,4,6,8 and 10). Resulting in 25 different LSTM models, which were then trained and evaluated. The results are shown in Figure 7.4.

The results suggest that a larger window size improved performance since it provided more context to the model. Although the best trade-off between performance and practicality was achieved with 1-minute intervals and a window size of 4 or 6, requiring only 4-6 minutes of

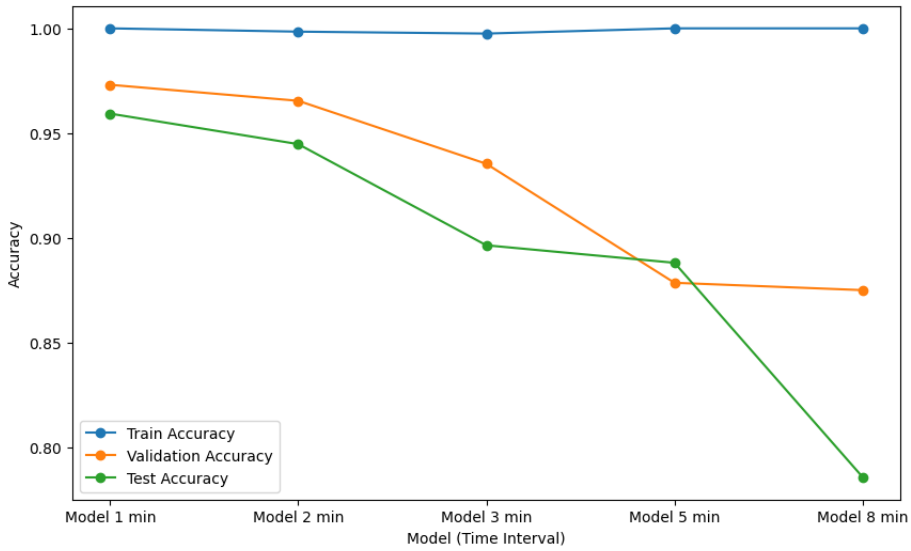

Figure 7.3 Comparison of different interval sizes of HRV features for a window size of 10.

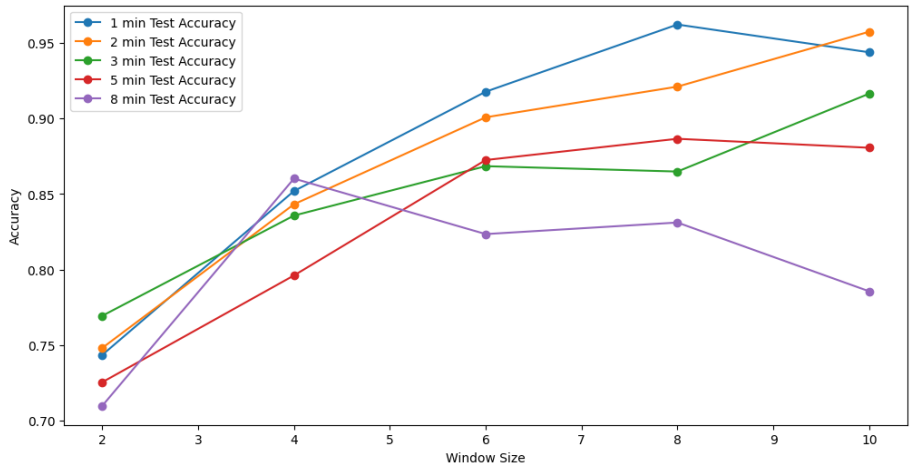

Figure 7.4 Models Accuracy Results from Combinations of Different Interval Duration and Window Size.

ECG data for classification.

In order to optimize these models, we observe that each trip was treated independently, but in the implementation chapter (Figure 4.5), we observed that merging all trips per participant helped to visualize the awake-to-drowsy transition.

Thus, the trips we merged for each driver and retrained the best models (1 min interval, window sizes 2, 4, and 6). The evaluation results are reported in Table 7.4, while in Figure 7.5 we have the respective model Confusion Matrix and ROC Curve.

The results show that with just 6 minutes of ECG data (window size = 6, 1 min intervals), we can predict drowsiness with high accuracy. However, if faster classification is needed, the 4-minute model (window size = 4) still performs very well. The 2-minute model (window size = 2), had a similar accuracy to the best Random Forest baseline, indicating that LSTM truly leverage temporal patterns only when given enough context.

Table 7.4 LSTM classification results for Time Domain 2 min interval dataset and with window size of 10.

Evaluation Metrics	Window Size of 2	Window Size of 4	Window Size of 6
Accuracy	0.763	0.874	**0.914**
Precision	0.766	0.871	**0.936**
Recall	0.823	0.914	**0.914**
F1	0.794	.892	**0.925**
AUC	0.85	0.94	**0.97**

In summary, the drowsiness classification results demonstrated that HRV features effectively predict driver drowsiness, with significant improvements achieved by treating the problem as a time-series task using LSTM. The baseline models confirmed that time-domain features performed better than frequency-domain features, and shorter intervals (2 min) provided higher accuracy. Transitioning to LSTMs with sequential input drastically improved performance, reaching an accuracy of 0.838 with 20-minute windows. Further optimization revealed that shorter interval durations (1 min) yielded better results, and increasing the window size improved performance by providing more context. The final models, trained on merged trip data to capture long-term drowsiness progression, achieved a peak accuracy of 0.914 with a 6-minute window and 0.874 with a 4-minute window, balancing high accuracy with practical real-time applicability. These findings highlight that HRV-based drowsiness detection is not only feasible but can be optimized for real-world use with minimal data collection time.

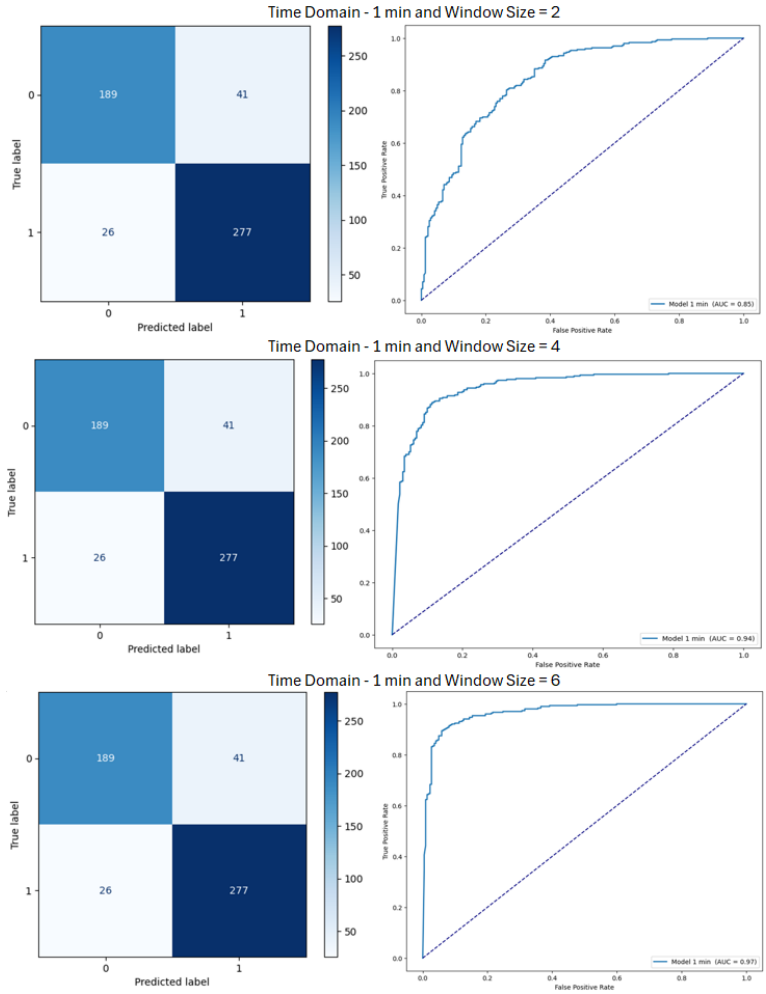

Figure 7.5 Best LSTM Models Confusion Matrices and ROC Curves.

Chapter 8

Conclusions

Road accidents remain a leading cause of death worldwide, with driver-related factors such as aggressive behavior and drowsiness contributing significantly to their occurrence. Addressing these issues through behavioral monitoring and early detection is crucial for improving road safety. This thesis explored two complementary solutions: driver profile classification using vehicle telematics data and drowsiness detection based on physiological signals. For the driver profiling task, an end-to-end machine learning pipeline was developed, incorporating unsupervised and supervised learning to identify driving styles. The approach effectively distinguished between consistent aggressive, risky, and non-aggressive behavior, with XGBoost and SVM yielding the best classification performance. For drowsiness detection, HRV features were extracted from ECG data and used to train both traditional classifiers and sequential models. Time-series classification with LSTM networks proved especially effective, achieving up to 91.4% accuracy while maintaining practical data requirements. These results demonstrate the potential of combining behavioral and physiological data for real-time driver monitoring, offering promising directions for intelligent, safety-focused driving support systems.

The following sections summarize the key findings from each task—driver profile classification and drowsiness detection—highlighting the approaches taken and their respective outcomes.

For the Driver Profile Classification, the objective was to build upon the work of Luis Loureiro by improving feature selection techniques and developing an effective method for profiling drivers based on their behavior using telematics data from the i-DREAMS project. A machine learning pipeline was designed, incorporating data preprocessing, feature extraction, clustering, and classification. The dataset underwent multiple iterations of cleaning, normalization, and dimensionality reduction, which significantly influenced the final results. By normalizing trips by distance and applying feature selection and reduction techniques, we enhanced clustering performance. A two-stage clustering approach was employed, with K-Means proving to be the most effective and stable algorithm. The first stage separated aggressive and non-aggressive trips, while the second stage further divided aggressive trips into aggressive and risky categories. For supervised learning, various classifiers were tested, with Extreme Gradient Boosting (XGBoost) and Support Vector Machines (SVM) achieving the best results.

Instance sampling techniques were also explored to handle dataset imbalance, with ADASYN providing the best performance. The final classification approach successfully identified key features influencing driving behavior, such as speeding violations and harsh cornering events. However, defining a driver's overall profile based solely on individual trips proved challenging, as inconsistent driving behavior impacted profiling accuracy. The method succeeded in cases where drivers exhibited consistent behavior across trips, demonstrating that while machine learning can effectively classify driving styles, a more comprehensive profiling approach may be necessary for inconsistent drivers.

For the Drowsiness Classification, the goal was to develop a machine learning model capable of predicting driver drowsiness based on Heart Rate Variability (HRV) features. This task was framed as a binary classification problem (awake vs. drowsy), using six labeled datasets derived from time-domain and frequency-domain HRV features computed over different time intervals. These datasets were built from the original physiological recordings, requiring the extraction of ECG segments to compute the HRV features. In addition, it was necessary to align the ECG data with the simulator logs to correctly synchronize physiological signals with the corresponding Karolinska Sleepiness Scale (KSS) scores. This alignment process was crucial to ensure accurate labeling of drowsiness states, forming a reliable basis for model training and evaluation. Initially, baseline machine learning models were tested, with Random Forest achieving the best accuracy of 0.761, showing that time-domain features and shorter time intervals performed better. However, since drowsiness is a progressive state influenced by time-dependent patterns, we shifted towards a time-series classification approach using Long Short-Term Memory (LSTM) networks. The first LSTM model, trained on 2-minute intervals grouped into 10-sized windows, significantly improved performance, reaching an accuracy of 0.838, confirming that sequential modeling was the right approach. Further optimization focused on reducing the required classification time while maintaining accuracy. By testing different interval durations and window sizes, we found that a 1-minute interval with a window size of 6 (6 minutes of ECG data) achieved the highest accuracy of 0.914, while a 4-minute model maintained a strong accuracy of 0.874. Finally, merging trips for each participant further improved drowsiness detection by capturing long-term drowsiness progression. These results validated the effectiveness of HRV-based drowsiness prediction, demonstrating that a real-time, practical system could be implemented with minimal data collection while maintaining high accuracy.

8.1 Future Work

For future work on Driver Profile Classification, a more comprehensive profiling approach may be necessary to improve accuracy, especially for drivers with inconsistent behavior. The unsupervised learning phase could benefit from exploring alternative clustering algorithms and variations to refine trip categorization further. Additionally, applying other feature selection techniques may help achieve an even greater dimensionality reduction while ensuring consis-

tency among selected feature subsets. This could enhance the interpretability of results and improve classification performance.

For Drowsiness Classification, future improvements could include incorporating additional physiological and behavioral metrics to label the dataset, as the current approach relies solely on the Karolinska Sleepiness Scale (KSS), which may be subjective across individuals. A more refined and complex deep learning model could also be developed to improve generalization across different drivers, making the system more adaptable and robust.

Finally, merging Driver Profile Classification and Drowsiness Classification represents a promising direction for future research. By combining various behavioral and physiological metrics identified as correlated with safety and drowsiness, a comprehensive Road Safety Classifier could be developed. Such a model could provide a more holistic assessment of driving risks by integrating both aggressive driving behaviors and drowsiness indicators, which are critical factors contributing to road accidents.

Bibliography

Altman, K. (2018). M. the curse(s) of dimensionality. *Nat Methods*, 15(13):399–400.

Argentina Leite, Maria Eduarda Silva, A. P. R. (2020). Classification of hrv using long short-term memory networks. *FEP*.

Balakrishnan, K., Muthusamy, S., Sankaranarayanan, S., Srinivasan, H., Ganesan, S., and Radhakrishnan, V. (2019). Destress: Deep learning for unsupervised identification of mental stress in firefighters from hrv data. *arXiv preprint arXiv:1911.13213*. Accessed April 12, 2025.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Caliński, T. and Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. pages 785–794, New York, NY, USA.

Cho, J. (2022). Changes in autonomic nervous system activity during sleep deprivation and its correlation with cognitive performance and stress. Master's thesis, New Jersey Institute of Technology. Accessed April 12, 2025.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Davies, D. and Bouldin, D. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227.

Deepali Virmani, Shweta Taneja, G. M. (2015). Normalization based k means clustering algorithm. *ARXIV*.

Ferreira, A. and Figueiredo, M. (2012). Efficient feature selection filters for high-dimensional data. *Pattern Recognition Letters*, 33(13):1794–1804.

Firstbeat Technologies Oy (2025). Heart rate variability (hrv).

Fisher, R. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188.

Fred, A. and Jain, A. (2002). Data clustering using evidence accumulation. volume 4, pages 276–280 vol.4.

Hartigan, J. and Wong, M. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108.

Izquierdo, J. L., Romera, R., Garcia, L., and Gómez, A. (2010). Electrocardiogram analysis using a combination of statistical, geometric, and nonlinear hrv features. *Journal of Clinical Monitoring and Computing*, 24(6):407–414. Accessed April 12, 2025.

Jurecki, R. S. and Stańczyk, T. L. (2021). A methodology for evaluating driving styles in various road conditions. *Energies*, 14(12).

Kaid, K., Takahashi, M., Akerstedt, T., Nakata, A., Otsuka, Y., Haratani, T., and Fukasawa, K. (2016). Validation of the karolinska sleepiness scale against performance and eeg variables. *Clinical Neurophysiology: Official Journal of the International Federation of Clinical Neurophysiology*, 117(7):1574–1581.

Khushaba, R. N., Kodagoda, S., Lal, S., and Dissanayake, G. (2011). Driver drowsiness classification using fuzzy wavelet-packet-based feature-extraction algorithm. *IEEE Transactions on Biomedical Engineering*, 58(1):121–131. Accessed April 12, 2025.

Li, X., Zhou, H., Liu, Z., Liu, X., Lin, H., Zheng, W., and Liu, Z. (2019). Developing robust and high accurate ecg beat classification by combining gaussian mixtures and wavelet features. *Journal of Healthcare Engineering*, 2019:1–10. Accessed April 12, 2025.

Liu, T., Zhou, R., and Zhang, X. (2024). Heart rate variability as a predictor of fatigue in young drivers with short sleep duration. *Transportation Research Part F: Traffic Psychology and Behaviour*, 95:101–110. Accessed April 12, 2025.

Loureiro, L. M. P. (2023). Driver profile classification. Master's thesis, Instituto Superior de Engenharia de Lisboa, Lisbon, Portugal.

Ma, Y., Li, W., Tang, K., Zhang, Z., and Chen, S. (2021). Driving style recognition and comparisons among driving tasks based on driver behavior in the online car-hailing industry. *Accident Analysis Prevention*, 154:106096.

Matsuzaki, I., Nishimura, A., Morita, N., Satoh, S., Kobayashi, T., and Murakami, M. (2006). Autonomic nervous activity changes due to shift-work: An evaluation by spectral components of heart rate variability. *Journal of Occupational Health*, 38(2):80–81.

Oliveira, L., Cardoso, J. S., Lourenço, A., and Ahlström, C. (2018). Driver drowsiness detection: a comparison between intrusive and non-intrusive signal acquisition methods. *2018 7th European Workshop on Visual Information Processing (EUVIP)*, pages 1–6.

Quinlan, J. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

Reynolds, D. (2009). Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663).

Riahi Samani, A. and Mishra, S. (2022). Assessing driving styles in commercial motor vehicle drivers after take-over conditions in highly automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):19161–19172.

Rousseeuw, P. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20(1):53–65.

Roy, S., Thomas, J., and Anchan, A. (2024). Development of modular components for a smarteye-like assistive technology system.

Sander, J., Ester, M., Kriegel, H., and Xu, X. (1998). Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data mining and knowledge discovery*, 2(2):169–194.

Strang, G. and Borre, K. (1997). *Linear algebra, geodesy, and GPS*. Siam.

Tement, S., Musil, B., Plohl, N., Horvat, M., Stojmenova, K., and Sodnik, J. (2022). *Assessment and Profiling of Driving Style and Skills*, pages 151–176. Springer International Publishing, Cham.

Thomas, J. A. and Cover, T. (1991). Elements of information theory. *John Wiley & Sons, Inc., New York. Toni, T., Welch, D., Strelkowa, N., Ipsen, A., and Stumpf, MPH (2009),"Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems," Journal of the Royal Society Interface*, 6:187–202.

Widodo, P. and Arifin, Z. (2019). Drowsiness detection based on heart rate variability using ad8232 and microcontroller unit. volume 1153, page 012047. IOP Publishing. Accessed April 12, 2025.