Prediction of Driver Lane-Change Behavior -

Modeling, Feature Selection and Evaluation

vorgelegt von
M. Eng.
Xiaohan Li

von der Fakultät V - Verkehrs- und Maschinensysteme
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
-Dr.-Ing.-

genehmigte Dissertation

Promotionsausschuss:
Vorsitzender: Prof. Dr. phil. Manfred Thüring
Gutachter: Prof. Dr.-Ing. Matthias Rötting
Gutachter: Prof. Dr. phil. Klaus Bengler
Tag der wissenschaftlichen Aussprache: 21.10.2019

Berlin 2019

Zusammenfassung

Die Vorhersage von Spurwechselverhalten (SW) im motorisierten Straßenverkehr ist ein wichtiger Schritt zur Vermeidung von Spurwechselunfällen. Ziel dieser Dissertation ist deshalb die Entwicklung einer umfassenden Methode zur Vorhersage von SW, einschließlich experimentellen Designs, Modellierung, Auswahl relevanter Fahr- und FahrerInnen-Variablen und deren Evaluation. Die Methode konzentriert sich hierbei auf die Autobahn als Fahrumgebung. Es wurden drei aufeinander aufbauende Studien durchgeführt, ein Fahrsimulatorexperiment, eine Analyse bestehender Realfahrtdaten, sowie eine Realfahrtstudie. Hierbei kamen Methoden des Maschinellen Lernens (ML), der Informatik, der Statistik, sowie Methoden der Human Factors Forschung zum Einsatz.

In der ersten Studie der Dissertation wurde eine komplexe Methode zur Vorhersage von SW entwickelt, die bestehende Methoden zu SW Vorhersage weiterentwickelt. In einem ersten Schritt wurden Verkehrsumfeldfaktoren sowie der Fahrstil von AutofahrerInnen genutzt um einen *Fahrstil* Trainingsdatensatz zu erstellen. Zusätzlich wurde ein weiterer Trainingsdatensatz ohne Einbezug von Verkehrsumfeldfaktoren und Fahrstil erstellt, der *Unkategorisiert* Datensatz. Beide Datensätze wurden genutzt um ein ML Modell zu trainieren, welches SW und Spurhalte (SH) Daten klassifizieren kann. Zusätzlich wird eine neuartige blickbasierte Klassifikationsmethode (BKM) entwickelt, um SW- und SH-Daten zu beschriften, und die BKM mit der bestehenden Zeitfensterklassifikation (ZFK) verglichen. Die Ergebnisse zeigen, dass ML-Modelle, die durch Fahrstil-Datensätze trainiert werden, höhere Klassifizierungswerte erzielen können als ML Modelle die mit dem unkategorisierten Datensatz trainiert werden. Darüber hinaus ist die Klassifikationsleistung der Modelle durch die Verwendung der BKM-Methode vielversprechender als durch die ZFK-Methode. Um den Einschränkungen der ersten Studie, speziell der Datenerhebung im Fahrsimulator und der Variablenauswahl mit unzureichender empirischer Datenlage, entgegenzuwirken, wurde eine zweite Studie durchgeführt, die einen bestehenden Datensatz mit Realfahrtdaten analysiert.

Das Ziel der zweiten Studie ist die Entwicklung einer systematischen Methode zur Variablenauswahl, um Lücken in der bisherigen Forschung im Zusammenhang mit der Vorhersage des SW-Verhaltens zu schließen. Aus bestehenden Realfahrtdaten werden mehrere für das SW-Verhalten relevante Variablen extrahiert, z.B. dynamische Variablen der Fahrzeugbewegung, des FahrerInnenverhaltens, komplexere Variablen, die mehrere Variablen kombinieren, sowie Zeitfenstervariablen, die Änderungen im

Zeitverlauf integrieren. Darüber hinaus werden Variablen aus dem Frequenzbereich extrahiert. Im Gegensatz zu bestehenden Methoden zur Variablenauswahl verwendet diese Studie statistische Methoden, die ein tieferes Verständnis des Beitrags des einzelnen Merkmals zum SW-Verhalten der FahrerInnen ermöglichen. Dieser Ansatz ist außerdem allgemeiner als bestehende Ansätze, die eine Variablenauswahl nur für einen bestimmten Algorithmus erlauben.

In einem dritten Experiment werden die Methoden aus den ersten beiden Studien kombiniert, um SW-Verhalten in einem Realfahrtexperiment vorherzusagen. Obwohl diese Studie ähnliche Methoden wie die ersten beiden Studien verwendet, bietet sie eine einzigartige Gelegenheit, die Machbarkeit des vorgeschlagenen SW-Vorhersageansatzes unter realen Straßenbedingungen zu testen. Neben dem *Fahrstil* Datensatz und dem *Unkategorisiert* Datensatz wird in dieser Studie zusätzlich ein *Personalisiert* Datensatz genutzt, der für jede Versuchsperson individuell angelegt wird. Es wird ein Vergleich der Vorhersage von SW mittels der verschiedenen Trainingsdatensätze, ML-Modellen und Klassifikationsmethoden, sowie der Frage, ob Eye-tracking für die SW-Vorhersage einbezogen werden soll, durchgeführt. Die Ergebnisse deuten darauf hin, dass ML-Modelle am besten mit der Kombination aus der Verwendung personalisierter Trainingsdatensätze und der BKM-Methode mit der Verwendung von Eye-Tracking-Informationen funktionieren. Im abschließenden simulierten Test der Vorhersage in der Realfahrt konnte das entwickelte Modell das SW-Verhalten der FahrerInnen im Mittel 3,3 s (Genauigkeit 93,5 %) vor einem tatsächlichen SW-Manöver für einen linksseitigen SW, und 2,6 s (Genauigkeit 72,4 %) für einen rechtsseitigen SW vorhersagen.

Die in dieser Dissertation entwickelte Methode zur Vorhersage von SW-Verhalten kann in der Anwendung dazu beitragen, die Anzahl der Spurwechselunfälle zu verringern und somit die Zahl der daraus resultierenden Verletzungen und Todesfälle reduzieren.

Abstract

Prediction of driver lane-change (LC) behavior is very important to avoid LC related traffic accidents. The aim of this dissertation is to propose a comprehensive framework, including experimental design, modeling, feature selection as well as evaluation, which can be implemented for prediction of driver LC behavior. The framework is designed to concentrate on highway roads. To this end, three studies were conducted step by step including a driving simulator based experiment, a big data analysis, and a real-road experiment. Methods used in this dissertation involve machine-learning (ML), informatics, statistics as well as the human factor field.

In the first study of the dissertation, a complete framework of prediction of driver LC behavior was developed with several methodological innovations in comparison to prior research. Firstly, driving contextual traffic and driving style were considered for the preparation of the training datasets, termed as *driving style datasets*. Datasets without any additional consideration were termed as *non-categorized datasets*. These datasets were used to train the ML models to classify LC and lane-keep (LK) data samples. Secondly, a newly gaze-based labeling (GBL) method was further proposed to label LC and LK data samples compared with the time-window labeling (TWL) method which was commonly used by the related works. The results show that ML models trained by the driving style datasets can achieve higher classification scores than the non-categorized datasets. In addition, by using the GBL method, the classification performances of the models are more promising than by using the TWL method. To counter the limitations of the first study, i.e. data collection from the driving simulator and the feature selection based on the insufficient empirical knowledge, a second study was conducted based on a large set of naturalistic driving data.

The aim of the second study was to propose a systematic feature selection method to fill in gaps of prior research related to prediction of driver LC behavior. In order to enrich the feature sets, a comprehensive set of features related to driver LC behavior were extracted, e.g. dynamic features of vehicle movement, behavioral features of the driver, more complex features that combine multiple variables, as well as time-window features which integrate changes over time. In addition, features from the frequency domain were extracted by varying time-windows. In contrast to the established methods for feature selection, this study uses statistical methods that allow a deeper understanding of the contribution of the individual feature to driver LC behavior and is more generalized in

comparison to prior research where the feature selection methods tend to work only for one specific algorithm.

Finally, combining the methods developed in the first two studies, a real-road experiment was conducted to evaluate the complete framework for LC prediction proposed in this dissertation. While this study uses similar methods to the first two studies, it provides a unique opportunity to assess the feasibility of the proposed LC prediction approach under real-road conditions. Besides *driving style datasets* and *non-categorized datasets*, in this study, *personalized datasets*, i.e. each participant has his/her individual dataset, were added for comparison. Comparison was made between training datasets, ML models, and labeling methods as well as the comparison between fusing and without fusing eye-tracking signals and only using eye-tracking signals for prediction. The result suggests that ML models perform best with the combination of using the personalized training datasets and the GBL method by fusing eye-tracking signals. In the final simulated real-time prediction test, the model could predict driver LC behavior around 3.3 s (precision 93.5%) ahead of an actual LC maneuver for left LC case and 2.6 s (precision 72.4%) for right LC case.

In conclusion, the framework for driver LC prediction developed in this dissertation, if implemented, could help to decrease the number of LC related crashes and reduce the resulting number of injuries and fatalities.

DEDICATION

To my parents
for letting me pursue my dream
for so long
so far away from home

Acknowledgements

First of all, I would like to express sincere gratitude to my supervisor Prof. Dr.-Ing. Matthias Rötting, for his continuous support on my Ph.D study, for encouraging me whenever I was stuck, and for guiding me to grow as a real research scientist. I also would like to thank Prof. Dr. phil Klaus Bengler for his willingness to supervise my thesis and giving me valuable suggestions.

My sincere thanks also go to all of my colleagues at Chair of Human-Machine-Systems for the kindly help in the last four years. In particular, I would like to thank Mario Lasch and Stefan Damke for their technical support to my experiments, patiently. And also for our secretary Mrs. Elisabeth Langer who always kindly reminds me to have a break when I was working for too long. A special thanks goes to my colleague and friend Felix Siebert for his enlightening suggestion at the beginning of my research and for the German translation.

Beside, I would like to thank Joyson Safety Systems GmbH for their cooperation by providing the experimental vehicle for my third study.

Last but not least, I would like to thank my family for their endless love.

Table of Contents

List of Figures

List of Tables

Abbreviations

ADAS advanced driver assistance systems 1

ANN Artificial neural networks 11

AoI Area of interest 9

AUC Area under curve 27

BIC Bayesian information criterion 49

BN Bayesian networks 11

CAN Controller Area Network 8

CV Cross-validation 28

DLC Discretionary lane-change 6

DT Decision tree 11

EM Expectation-maximization 26

FFT Fast Fourier transform 71

FPR false positive rate 27

GBL Gaze-based labeling method 32

GMM Gaussian mixture model 18

HMM Hidden Markov model 11

KNN k-nearest neighbor 11

LC lane-change 1

LK lane-keep 31

LLC Left lane-change 32

ML Machine learning 11

MLC Mandatory lane-change 6

NB Naive Bayes 11

RLC Right lane-change 64

RNN Recurrent neural networks 8

Abbreviations

ROC Receiver operating characteristic 27

SPMD Safety pilot model deployment 66

SVM Support vector machines 11

TLC Time to line crossing 70

TPR True positive rate 27

TTC Time to collision 9

TW Time-window 71

TWL Time-window labeling method 31

1 Introduction

1.1 Motivation of the dissertation

In 2017 alone, a total of 3,180 people were killed in road traffic accidents in Germany (FederalStatisticalOffice, 2017). The number of highway fatalities in the USA is 37,461 in 2016, which was an increase of 5.6% from 2015 (NHTSA, 2016). New data shows that more than 90% of traffic accidents are related to human error (Singh, 2015). Under such backdrop, various advanced driver assistance systems (ADASs), e.g. adaptive cruise control systems, lane departure warning system etc., have been developed to assist the driver in order to increase driving safety. However, research still needs to be done one step further in order to improve the functionality and reliability of ADASs. Understanding and modeling driver behavior provides a solution for developing intelligent automotive application (Plöchl and Edelmann, 2007). If ADAS can predict driver behaviors several seconds in advance, it can prevent a potential accident from an improper driver behavior (Mokhiamar and Abe, 2002).

There are various driver behaviors that are necessary to be focused on, e.g. predicting driver lane-change (LC) behavior and driver turning behavior can avoid side crashes, braking or accelerating behavior for rear-end collision and detecting driver drowsiness behavior for severe accident etc. Researching on all of them takes plenty of time. Instead, this dissertation is focusing on one specific driver behavior comprehensively rather than doing research that covers all the aspects but only superficially.

Among the above mentioned driver behaviors, LC behavior is one of the most important one which may lead to a severe traffic accident but it is difficult to predict. Actually, LC crashes account for about 10% of all crashes (Barr and Najm, 2001) and 1.5% of all motor vehicle fatalities in the USA (NHTSA, 2017). Statistical studies also show that LC accidents because of human error are 89% (Luoma, Sivak, and Flannagan,

1995). Thus, understanding and predicting driver LC behavior is beneficial for the development of ADAS and thus we can reduce traffic accidents.

To this end, in this dissertation we are focusing on the prediction of driver LC behavior on highway road. The aim is to propose a comprehensive framework and methodology which can be implemented for prediction of driver LC behavior. The fields of works included in this dissertation are regarding to machine-learning, informatics, statistics as well as the human factor field.

1.2 Structure of the dissertation

The main work of this dissertation is structured in 7 chapters. In chapter 1, by placing emphasis on the importance of traffic safety, the aim and the background of the dissertation are introduced. Chapter 2 makes an introduction about the key components of prediction of driver LC behavior, which includes the general concept of driver LC behavior, LC types, driver decision-making process, predictors of driver LC behavior, as well as models used for prediction in the related works. All these aspects lay the foundation of the work presented in this dissertation. Chapter 3 details the mathematical theories as well as the evaluation methods used throughout this dissertation. Instead of just explaining mathematical formulas, the aim is to explain the theories in an easy-to-understand way even for those who have little theoretical knowledge.

From chapter 4 on, the core studies involved in this dissertation are detailed. In chapter 4 we propose a framework of prediction of driver LC behavior. Works include the design of a driving simulator-based experiment, feature extraction, modeling machine learning models, training datasets preparation, model selection and evaluation. Based on the limitations summarized from the prior research, this chapter is seeking to make improvements. Several improvements have been made i.e. considering driving contextual traffic and driving styles in preparing for training datasets, and proposing a gaze-based labeling method (GBL) to obtain high quality class labels.

Chapter 5 is related to a big data analysis on feature selection. The aim is to overcome the limitation of the driving simulator based study by proposing a systematic feature selection work in the perspective of statistics. The proposed feature selection method can be used in general rather than for a specific algorithm.

By summarizing the method proposed in the last two studies and considering the limitations, a real-road experiment based study is detailed in chapter 6. Works which have been done include the experimental design in the real traffic, real-road data processing, data labeling, feature selection as well as the evaluation of the ML models.

In chapter 7, a more general discussion about the results concluded in all the three studies is given. At the same time, the original contributions of this dissertation are also declared. In addition, an outlook is given regarding to the meaning of the practical

implementation using the methods proposed in this dissertation as well as its possible challenges.

Finally, the Appendix lists all the important materials regarding to the necessary documents used in the experiments, data samples as well as the full scale of the results in the form of tables and figures.

2

Review - Prediction of driver lane-change behavior

2.1 Overview

To prevent LC accidents, ADASs have been developed to predict forthcoming driver LC behaviors while driving. For example, LC assistance system could assess the risk levels of maneuvering a LC under the current driving situation. If the driver intends to make LC with high risk, an alarm will be delivered to avoid the potential accidents. This function is depicted in Figure 2.1. Comprehensively understanding of driver LC behavior is very important but nontrivial. Fortunately, carefully partitioning the LC procedure into small segments could make this goal achievable. From the perception-action level, a complete LC task can be roughly divided into three stages: *forming intent*, *preparing actions*, and *executing actions*. The driver first *forms* the LC intent according to his/her traveling plan and the current driving situation, and with such intent, *prepares* for taking LC actions by longitudinal adjustment (e.g., waiting, accelerating, or decelerating), and then *executes* a series of LC actions such as lateral controls as long as the driving situation is acceptable (Windridge, Shaukat, and Hollnagel, 2013). To ensure safe driving, predicting driver LC behavior as early as possible can leave enough time to prevent improper LC behaviors. In this chapter we will give an overview of the work regarding to the prediction of driver LC behavior from basic concept to methodology which will lay the foundation of the entire work.

2.2 Driver lane-change behavior

Lane-change is defined as the movement of a vehicle from one vehicle lane to another lane with continuing travel in the same direction in the new lane (J2944, 2013).

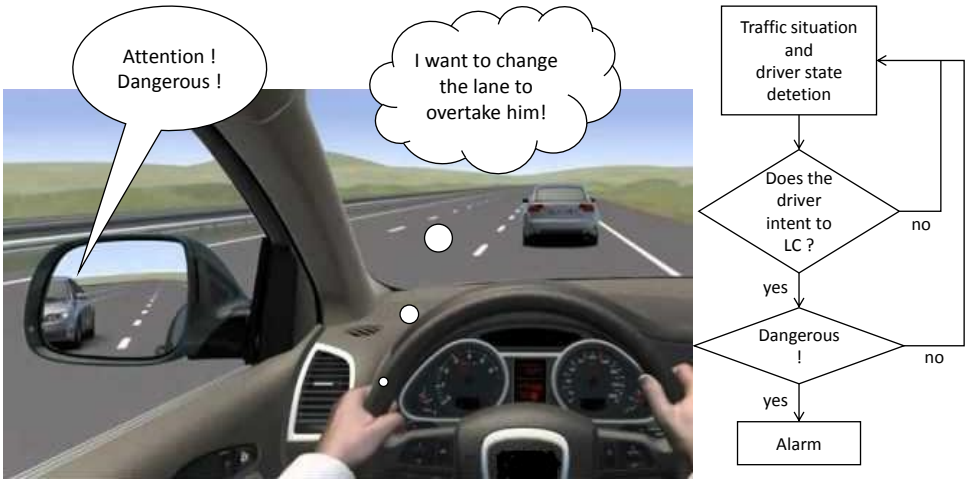


Figure 2.1: The use case of prediction of driver LC Behavior. Picture extracted from Audi (2013).

Actually, driver LC behavior can be varying from different types with varying purposes. Understanding driver LC behavior in depth is crucial for modeling driver LC behavior as well as the predictive models.

2.2.1 Types of lane-change behavior

Mandatory lane-change (MLC)

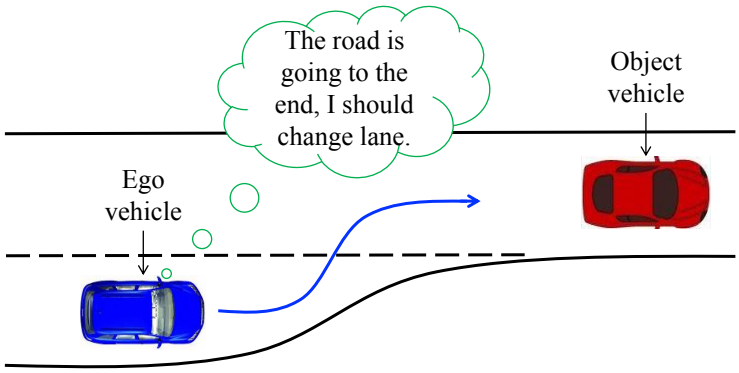
MLC occurs when a driver must leave a lane, such as when the lane in which they are driving ends (due to a lane drop or when merging from an on-ramp), to bypass a blockage downstream, or to avoid entering and using a restricted lane. MLC can also occur at the juncture of two or more traveled ways blending together in the same direction (J2944, 2013). This type of LC can be depicted in Figure 2.2a.

Discretionary lane-change (DLC)

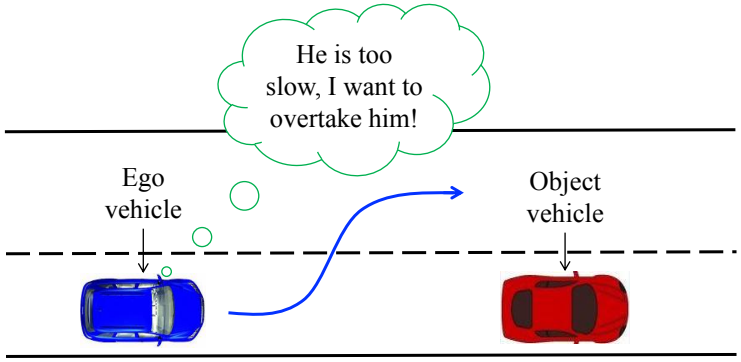
DLC occurs when a driver changes to a lane perceived to offer better traffic conditions, such as to achieve desired speed, avoid following trucks, avoid merging traffic etc. (Mathew, 2014). This type of LC case is shown in Figure 2.2b.

2.2.2 Lane-change decision-making process

One of the major problems to understand certain driver behavior is to phase the decision-making process of the behavior. LC decision-making process is very complex since the decision of a driver to make LC depends on a number of objectives, and at times LC decision can be changed. For example, imagine a driver is driving in the rightmost lane and wishes to turn right within 100 meters but still have to make LC to the left to pass a car parked in front. Or, imagine a driver wishes to accelerate shortly to execute LC to overtake the front car, but suddenly he/she finds that a fast moving car



(a) Mandatory lane-change.



(b) Discretionary lane-change.

Figure 2.2: An example of the two types of LC.

approaching from behind on the destination lane, then he/she may abort the LC until it is safe to maneuver the LC. From strategic level in the Gipps lane-change decision structure (Gipps, 1986), the LC decision-making process is the result of the answers of a number of questions:

- *Is it possible to change lanes?*
- *Is it necessary to change lanes?*
- *Is it desirable to change lanes?*

This process of LC comprises of two decisions: whether the driving conditions are satisfactory, and if not, whether any other lane is better than the current lane. The term driving conditions satisfactory implies that the driver is satisfied with the driving conditions of the current lane as he is able to maintain the desired speed. Important factors affecting the decision whether the driving conditions are satisfactory include the speed of the driver compared to the desired speed; the presence of heavy vehicles in front and behind the subject vehicle, if an adjacent on ramp merges with the current lane, whether the subject is tailgated etc. If the driving conditions are not satisfactory, the driver compares the driving conditions of the current lane with the adjacent lanes. Important factors affecting this decision include the difference between the speed of traffic in target lanes and the desired speed of the driver, the density of traffic in target

lanes, the relative speed with respect to the front vehicle in the target lane, the presence of heavy vehicles in target lanes ahead of the subject etc. (Mathew, 2014).

2.3 Predictor

Imagine we want to predict if it rains in the following few hours. Based on our prior knowledge, normally before it rains it would be cloudy and humid. And if it is going to rain, we can also observe some animal activities like ants building high ant mounts, low-flying birds, bees and butterflies returning home, cows gathering together and laying down etc. (Denham, 2012). These observations that can give clues to predict the rain are termed as *predictor*.

In order to predict driver LC behavior, we also seek the way to find the predictors. Some key predictors regarding to driver LC behavior can be listed as follows:

Steering

The driver maneuvers the steering wheel to execute a LC, so steering is a direct *predictor* of driver LC behavior. Steering wheel angel is a measurement of how much the driver steers. By detecting steering angle from Controller Area Network (CAN) bus, driver LC behavior can be modeled using a dynamic Markov model (Pentland and Liu, 1999). Similar work which uses steering wheel angle to model and predict LC behavior can be found in Kumar et al. (2013), Mandalia and Salvucci (2005), Salvucci (2004), and McCall et al. (2007).

Throttle

Research found when the driver wants to overtake a slow moving car, 5 s before executing LC the driver decelerates gradually, in order to avoid colliding with the slower vehicle. However, soon after lane-change onset, the driver accelerates to the overtaking speed and maintains that speed through the rest of the LC (Salvucci and Liu, 2002). This result indicates that we can find some clues to predict driver LC behavior by monitoring the opening angle of the throttle in certain scenario like overtaking. For instance, a recurrent neural network (RNN) approach was implemented by using throttle and brake data, as well as steering wheel angle to predict driver short-term LC intention (Xing and Xiao, 2018). Therefore, throttle data collected from CAN bus can be used for prediction.

Turn signal

It is quite common that the driver uses the turn signal to indicate his/her LC intention. It was found that the driver turns on the signals approximately 1.5 s before the start of a LC behavior (Salvucci and Liu, 2002), thus a computational driver model was used to detect driver LC behavior (Salvucci, Mandalia, et al., 2007). Other methods involve

the usage of the turn signals for LC prediction can be found in Xu et al. (2012) and Winner and Lueder (2005). However, one drawback of using the turn signal as model input for prediction is due to its instability. Research found that the turn signals were used only 44% of the time, with signals used more often for left lane changes (48%) than for right lane changes (35%) (Lee, Olsen, Wierwille, et al., 2004). Thus, turn signal is recommenced to be used together with other signals.

Time to collision (TTC)

TTC is the time required for two vehicles to collide if they continue at their present speeds on the same path. It is usually used to evaluate collision risk (Kusano and Gabler, 2011). If the driver follows a vehicle with a small TTC, he/she may execute a LC to overtake the slow leading vehicle. Thus the TTC can be regarded as a valuable feature to predict LC maneuver (Kasper et al., 2012). Using TTC as a feature, a dynamic probabilistic drivability map was modeled to give the driver recommended acceleration and timing to execute LC (Sivaraman and Trivedi, 2014). More related works that use TTC as a predictor for prediction of driver LC behavior can be listed in Liebner et al. (2013) and Peng et al. (2015).

Eye movement

Eye movement was found to be an indicator of information gathering and therefore can be used to derive information about the next planned objective of the driver (Lethaus and Rataj, 2007). A statistical analysis shows that the driver spends most of his/her gaze time before a LC looking at the current lane. As onset approaches, more gaze time is directed to the destination lane and the mirrors, since the driver is checking to the side and rear of the vehicle to ensure safe passage (Salvucci and Liu, 2002). The period of 3 - 4 s prior a LC is considered as critical phase of visual search to determine the feasibility of the maneuver (Beggiato et al., 2018). Based on Tijerina et al. (2005), during left LC process, the chance of looking at the left mirror is 65% – 85% and the duration on average is 1.1 s. This result indicates that driver LC behavior can be anticipated by observing mirror-glancing duration of the driver. Lethaus, Baumann, et al. (2013) used a mirror-glancing ratio during the last past seconds as the input of his model to predict driver LC behavior. In order to analyze gaze behavior in depth during LC behavior, the area of interests (AoI) of driver gaze performance is divided into several areas as it is shown in Figure 2.3. According to Lee, Olsen, Wierwille, et al. (2004) it was found that the most likely glance locations were forward (probability of 1.0), rear view mirror (0.52), and left mirror (0.52). The highest link probability value (0.34) was between the forward and rear view mirror locations. The most likely glance locations for right LC were forward (1.0), rear view mirror (0.55), and right mirror (0.21). The highest link probability value (0.60) was between the forward view and rear view mirror. The link

value probabilities between forward and right mirror and between forward and right blind spot were also relatively high at 0.12.

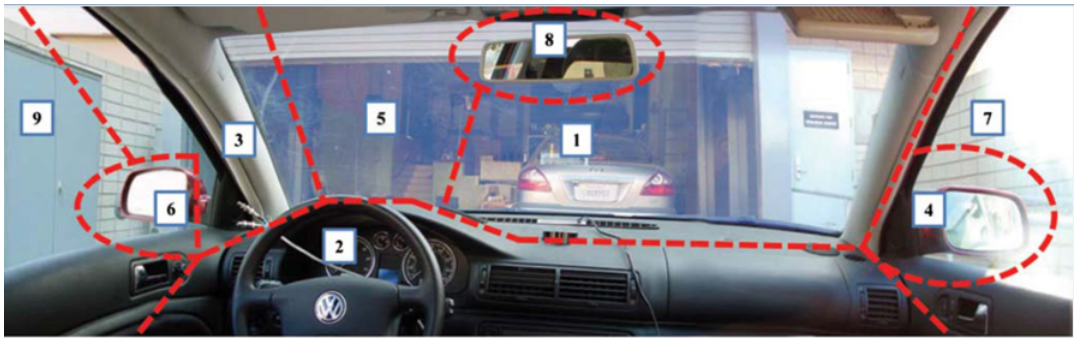


Figure 2.3: The AoIs of the driver while driving. Picture extracted from Doshi and Trivedi (2009).

These works indicate that driver gaze behavior is closely related to LC behavior, however, the challenge of using eye movement signals for real-time application is how to ensure tracking accuracy since gaze tracking is sensitive to lighting condition change (Zhu, Fujimura, and Ji, 2002).

Head pose

While driving, driver gaze behavior is closely related to his/her head pose, so head pose can be also regarded as a predictor of driver LC behavior. Unlike eye movement which can be monitored by various eye-tracking devices, the method of estimating head pose is usually camera-based image processing (Martin et al., 2012), which poses a challenge to the accuracy of head dynamic estimation technique. Once robust monocular in-vehicle head pose estimation systems have been developed (Murphy-Chutorian, Doshi, and Trivedi, 2007; Zhu and Fujimura, 2004), head pose thus can be used to model driver LC behavior. Doshi and Trivedi (2009) compared the method of using driver gaze position and head pose to predict driver LC behavior, and the result found that head pose can achieve earlier LC prediction. However, a limitation of the head tracking system is that it frequently loses track of the head direction at fast movements. It occurs when the driver is looking over his shoulder to check for bicycles (Liebner et al., 2013). In addition, head pose estimation is done by computer vision technology; the computational demand is high especially for vehicle on-board real-time application.

2.4 Prediction model

Now we go back to the example of weather prediction. In the ancient time, people forecasted the weather based on their prior knowledge. The prior knowledge mentioned in the last section, i.e. by observing animal activities, is actually based on statistics. Because those activities frequently happen, people can learn such lessons from the past

and then could forecast the weather. It is the same way for predictive model. Predictive model uses statistics to predict future events. Thanks to the development of informative technology, machine learning (ML) becomes a powerful tool for prediction modeling. Among various ML models, supervised learning models are the most popular tools, e.g. support vector machines (SVM), Naive Bayes (NB), decision tree (DT), k-nearest neighbor (KNN), artificial neural networks (ANN), Bayesian networks (BN), hidden Markov model (HMM) etc. A number of studies regarding to prediction of driver LC behavior have been carried out using supervised learning models as follows:

SVM

Kumar et al. (2013) proposed a solution to LC prediction based on the combination of a multi-class SVM classifier and Bayesian filtering using driver steering wheel angle and vehicle lane position data. The algorithm can predict on average 1.3 seconds before a LC occurs. The limitation is that only two drivers took part in the experiment, thus tests with more participants should be done to evaluate the method. Salvucci (2004) and Mandalia and Salvucci (2005) used a novel *mind-tracking* technique incorporated with SVM method to detect LC.

Bayesian model

A sparse Bayesian learning methodology was proposed by McCall et al. (2007) to infer driver LC intention by estimating the head pose of the driver and tracking the lane mark. It was found that by fusing driver state information it can predict driver LC behavior 3.0 s before an actual LC maneuver. However, the method used in the paper suffered from harsh lighting conditions, heavy traffic, occlusion of the lane markings and extremely poor road conditions. Kasper et al. (2012) introduced an object-oriented Bayesian networks to detect driver maneuvers on highway roads. Besides driver LC behavior, this model can also detect 26 other driver behaviors. But due to the parametrization of the network is computationally demand, a learning algorithm should be employed.

HMM

Pentland and Liu (1999) proposed a HMM using vehicle velocity, driver steering angle, brake pedal position as well accelerating pedal position signals to predict driver LC behavior based on a driving simulator experiment. By giving the driver text commands, the model can recognize LC maneuver 2 seconds after the onset of a LC command with an accuracy of 93.3%. The limitation of this work is that the experiment was conducted by giving text commands to the participants rather than letting them make LC based on their own preference. An autoregressive input-output HMM method was proposed by Jain, Koppula, Raghavan, et al. (2015). This model can estimate the head pose of the driver by tracking his/her face orientation to predict LC behavior.

Neural networks

Peng et al. (2015) developed a multi-parameter predictive model with a neural network model. Vehicle motion state, handing characteristics, driving conditions as well as head movement data were all involved in the model. A variation of neural network, termed as recurrent neural networks (RNN), was proposed by Jain, Koppula, Soh, et al. (2016) to predict driver behaviors. Using this model, maneuvers can be anticipated 3.5 seconds before they occur in real-time with a precision of 90.5%.

2.5 Application

The application of predicting driver LC behavior in ADAS has been developed and integrated into a human-machine interface to reduce driver workloads and enhance traffic safety with either active or passive feedback.

Active feedback

An active feedback is reported if a LC behavior is feasible, then the well-designed driver assistance system will cooperatively assist the driver changing lanes with recommended acceleration and speed (Sivaraman and Trivedi, 2014; Butakov and Ioannou, 2015).

Passive feedback

Passive feedback will display LC safety states or deliver alarms to the driver when changing lanes is infeasible (Schubert, Schulze, and Wanielik, 2010; Jain, Koppula, Raghavan, et al., 2015).

2.6 Summary

This chapter gives an overview of the key components of the prediction of driver LC behavior, including the general concept of driver LC behavior, LC types, driver decision-making processes, predictors of driver LC behavior, as well as models used for prediction in the related works. These contents are very important items which are the backbone of this field of research. All the work proposed in this dissertation is based on prior research, but not limited by them.

3

Mathematical background

This chapter details the mathematical theories as well as the evaluation methods used throughout this dissertation. Instead of just explaining mathematical formulas, we tend to use lively examples in order to make the theory to be understood easily even for those who have little theoretical knowledge. The detail of how to use these theories in practice will be mainly explained in the following sections.

3.1 Machine learning model

In the last chapter, we have reviewed the related works of how machine learning (ML) models have been implemented to solve the problem regarding to prediction of driver LC behavior. This section gives an insight into the math behind the ML models which would be mainly used in the following chapters.

3.1.1 Classification of ML models

When it comes to machine learning models, there are two main types of tasks: supervised, and unsupervised. The main difference between the two types is that supervised learning is done using a ground truth (class labels), or in other words, the prior knowledge of what the output values for our samples should be. Therefore, the goal of supervised learning is to learn a function that, given a set of data and desired outputs, best approximates the relationship between input and output observables in the data. Unsupervised learning, on the other hand, does not have labeled outputs, so its goal is to infer the natural structure within a set of data points[1]. Especially in our case, since we use *predictors* for prediction, the ML models used in this dissertation belong to supervised learning.

[1]More details can be found in this link: https://towardsdatascience.com/supervised-vs-unsupervised-learning-14f68e32ea8d, visited on 31.07.2019

The two main branches of supervised learning models are generic models and discriminant models. In the case of classification problems, generic models tend to model how the training data are generated and then try to find the properties based on the prior assumption. On the contrary, discriminant models do not care about how the training data are generated and simply try to find the best classification boundary to classify data samples. Both generic models (Bayesian network, naive Bayes etc.) and discriminant models (SVM, decision tree etc.) are applied in this dissertation.

3.1.2 Bayesian network

Bayesian networks are graphical structures for representing the probabilistic relationships among a large number of variables and doing probabilistic inference with those variables (Neapolitan et al., 2004). In order to have a better understanding of how BN predicts events, we use again the example of rain prediction. In chapter 2 we have learned that in order to predict the rain, we need to find some *predictors*. Now we take two predictors i.e. *Sprinkler* and *Humidity* for instance. Figure 3.1 depicts a famous BN structure for the rain prediction adapted from Russell and Norvig (1995). Based on our experience we know that before it rains, the humidity in the air is high. But when the sprinkler waters, it could also increase humidity. Also, the rain has a direct effect on the use of the sprinkler (considering that when it is going to rain, the sprinkler is usually not turned on). Then this situation can be modeled with a BN. All the three variables are binary values; the *joint probability* of this BN can be given as based on the chain rule (Schum, 2001):

$$P(H, S, R) = P(H|S, R) \cdot P(S|R) \cdot P(R), \tag{3.1}$$

where H, S and R are short for Humidity high, Sprinkler use and Rain, respectively.

Assume it is in a dry period which makes us believe that it is less likely to rain. So it is reasonable to set the probability of rain as 0.2 and fair weather as 0.8. In Bayesian inference theory it is called *prior*, which describes one's beliefs before some evidence is taken into account. The probability tables in Figure 3.1 regarding to sprinkler and humidity are called *conditional probability*. If we observe high humidity in air, then we can compute the probability of raining given high humidity condition, that is $P(R = \mathrm{T}|H = \mathrm{T})$, which is called *posterior*:

$$P(R = \mathrm{T}|H = \mathrm{T}) = \frac{P(R = \mathrm{T}, H = \mathrm{T})}{P(H = \mathrm{T})}, \tag{3.2}$$

where T is for true.

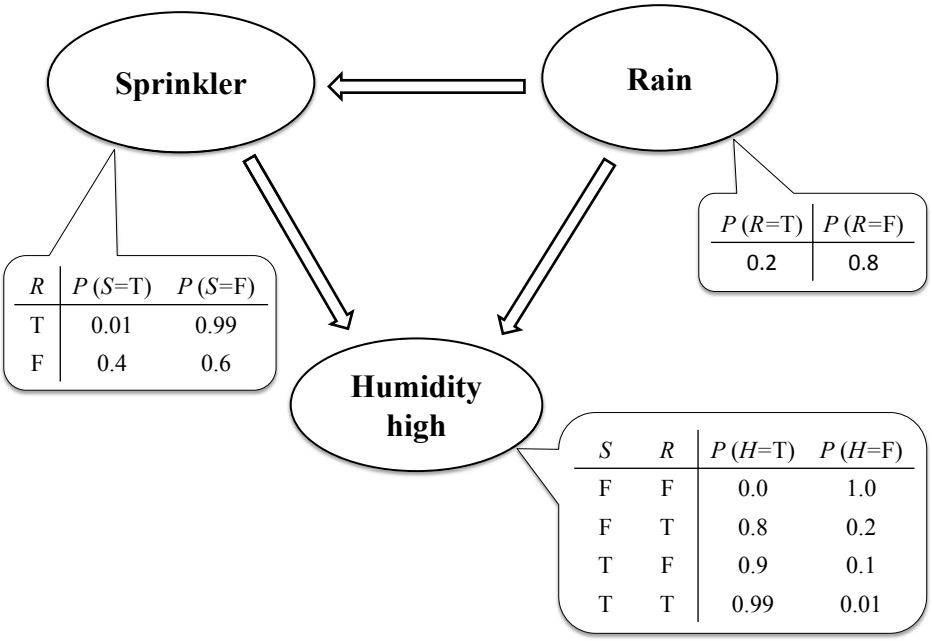


Figure 3.1: The Bayesian network of rain prediction with probability tables (R = Rain, S = Sprinkler, H = Humidity high, T = True and F = False).

Using a variation of total probability law[2], equation (3.2) can be calculated as:

$$\frac{P(R = \mathrm{T}, H = \mathrm{T})}{P(H = \mathrm{T})} = \frac{\sum_S P(H = \mathrm{T}, R = \mathrm{T}, S)}{\sum_{S,R} P(H = \mathrm{T}, S, R)}. \tag{3.3}$$

Combined with BN joint probability equation (3.1), $P(H = \mathrm{T}, R = \mathrm{T}, S = \mathrm{T})$ can be written as:

$$\begin{aligned} P(H = \mathrm{T}, R = \mathrm{T}, S = \mathrm{T}) &= P(H = \mathrm{T}|S = \mathrm{T}, R = \mathrm{T}) \cdot P(S = \mathrm{T}|R = \mathrm{T}) \cdot P(R = \mathrm{T}) \\ &= 0.99 \times 0.01 \times 0.2 = 0.00198, \end{aligned} \tag{3.4}$$

and $P(H = \mathrm{T}, R = \mathrm{T}, S = \mathrm{F})$ can be written as:

$$\begin{aligned} P(H = \mathrm{T}, R = \mathrm{T}, S = \mathrm{F}) &= P(H = \mathrm{T}|S = \mathrm{F}, R = \mathrm{T}) \cdot P(S = \mathrm{F}|R = \mathrm{T}) \cdot P(R = \mathrm{T}) \\ &= 0.8 \times 0.99 \times 0.2 = 0.1584. \end{aligned} \tag{3.5}$$

[2]For any given event A, the probability of A can be written as : $P(A) = \sum_B P(A, B)$, where $\sum_B$ means all the possibilities of event B.

So, $\sum_S P(H = \mathrm{T}, R = \mathrm{T}, S)$ can be written as:

$$\begin{aligned}\sum_S P(H = \mathrm{T}, R = \mathrm{T}, S) &= P(H = \mathrm{T}, R = \mathrm{T}, S = \mathrm{T}) + P(H = \mathrm{T}, R = \mathrm{T}, S = \mathrm{F}) \\ &= 0.00198 + 0.1584 = 0.16038.\end{aligned} \tag{3.6}$$

In the same way, we can also calculate $\sum_{S,R} P(H = \mathrm{T}, S, R)$:

$$\sum_{S,R} P(H = \mathrm{T}, S, R) = 0.44838. \tag{3.7}$$

Combining equation (3.2), (3.3), (3.6) and (3.7) we can get the probability of raining given high humidity:

$$\begin{aligned}P(R = \mathrm{T}|H = \mathrm{T}) &= \frac{\sum_S P(H = \mathrm{T}, R = \mathrm{T}, S)}{\sum_{S,R} P(H = \mathrm{T}, S, R)} \\ &= \frac{0.16038}{0.44838} \\ &\approx 35.77\ \%.\end{aligned} \tag{3.8}$$

From this example we know that by only observing high humidity in air, the chance of raining is 35.77 %, which means it is less likely to rain. One main reason is that the prior probability of rain is low ($P(R = \mathrm{T}) = 0.2$). Actually in BN, the posterior probability, to a great extent, is governed by the prior. If in the example we preset the prior of raining as 0.8 rather than 0.2, the outcome of posterior probability would be a much higher value. Thus, the prior knowledge is important for Bayesian probability inference.

The example of weather prediction only gives a discrete case of BN (all the variables are discrete with binary states), however, BN can also be implemented with continuous variables. Each variable in a BN is called a *node*. In Figure 3.1, *Sprinkler*, *Rain* and *Humidity high* are the three nodes of the BN. Based on the causality of the three nodes, the node *Rain* is termed as the parent node of *Sprinkler* and *Humidity high*, and *Sprinkler* is the parent node of *Humidity high*. For any given BN, the joint probability of the BN can be written as:

$$P(Z_1, Z_2, ..., Z_n) = \prod_{i=1}^{n} (Z_i | Pa(Z_i)), \tag{3.9}$$

where $Pa(Z_i)$ is the parent node of variable Z_i. This is a general form of equation (3.1). With the joint probability and Bayes theorem we can infer posterior probability. More BN structures and inference methods can be found in Neapolitan et al. (2004) and Murphy and Russell (2002).

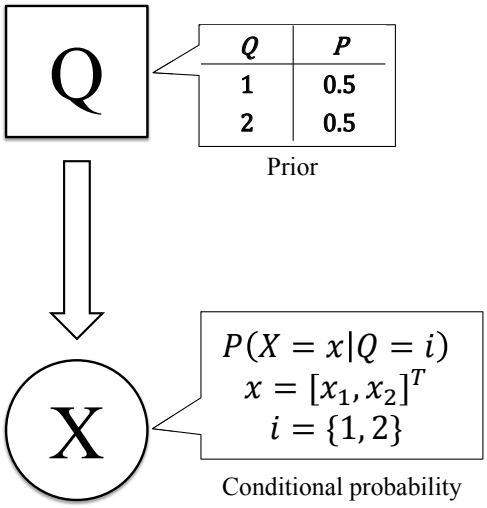


Figure 3.2: The Bayesian network for data classification problem.

3.1.3 Gaussian mixture model

Let us take an example of data classification problem. Figure 3.2 presents a BN with one discrete node Q and one continuous node X. In a BN with the mixture of both discrete and continuous nodes, discrete nodes are usually represented as squares and continuous nodes as circles. So, node Q is a discrete variable with binary states and X is a continuous variable that obeys certain distribution. We set the prior of Q as 0.5 for both two random variable values. This means that the chance of $Q = 1$ and $Q = 2$ is equal. We do not know what distribution X obeys, but we can observe some values of random variable X given $Q = 1$ and $Q = 2$, see Figure 3.3a. The red points are generated when $Q = 1$ and the blue are from $Q = 2$. The problem is that given some new data how can we know in which class they are from? In Figure 3.3b, we observe new data points, are they red or blue?

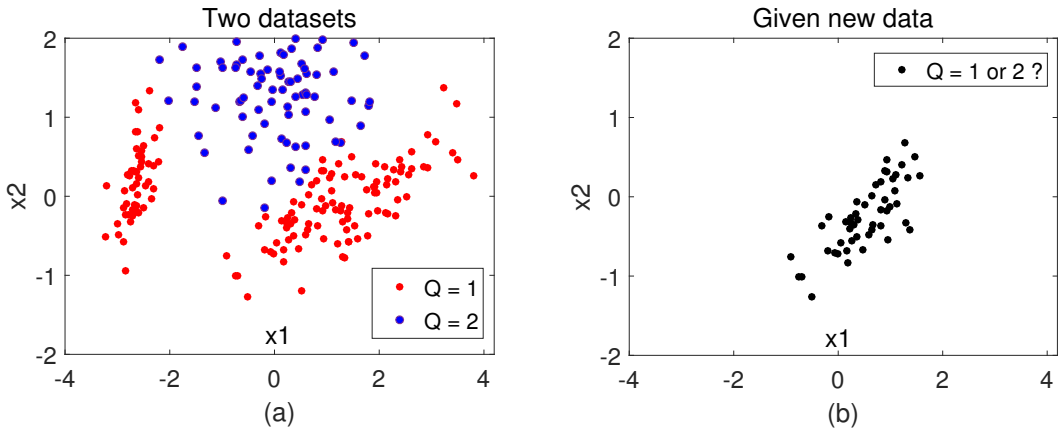


Figure 3.3: An example of data classification problem.

Without loss of generality, we assume the black points are generated by $Q = 1$. So we can know how much this assumption can be trusted by calculating the posterior probability $P(Q = 1|X)$ using Bayes theorem[3]:

$$P(Q = 1|X) = \frac{P(X|Q = 1) \cdot P(Q = 1)}{P(X)}. \tag{3.10}$$

Because the *marginal distribution* $P(x)$ is very difficult to get, $P(Q = 1|X)$ can normalized in the form of:

$$P(Q = 1|X) = c \cdot P(X|Q = 1) \cdot P(Q = 1), \tag{3.11}$$

where c is a constant value to make sure the posterior probability sum to one.

Then the problem of classification becomes how to fit the conditional probability density $p(x|Q = 1)$. Figure 3.4a shows the observed x values given $Q = 1$. Since Gaussian distribution is the most important and most widely used distribution in statistics, we assume $p(x|Q = 1)$ is subject to Gaussian distribution:

$$p(x|Q = 1) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma^2}), \tag{3.12}$$

where $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ are the mean and covariance of the observed data in Figure 3.4a.

The Gaussian fitting result can be seen in Figure 3.4b. However, from the probability contour of the fitted distribution we may disbelieve our assumption that the conditional distribution is subject to Gaussian distribution. Actually in Figure 3.4c we can see that the Gaussian contour which represents data from $Q = 1$ largely overlaps the blue data which generated from $Q = 2$. This result makes us reject the Gaussian assumption. But it does not mean that we cannot use Gaussian method to fit $p(x|Q = 1)$. Gaussian mixture model (GMM) is an alternative solution.

[3]For any given event A and B, the probability of A given B can be written as : $P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$, where $P(B) \neq 0$.

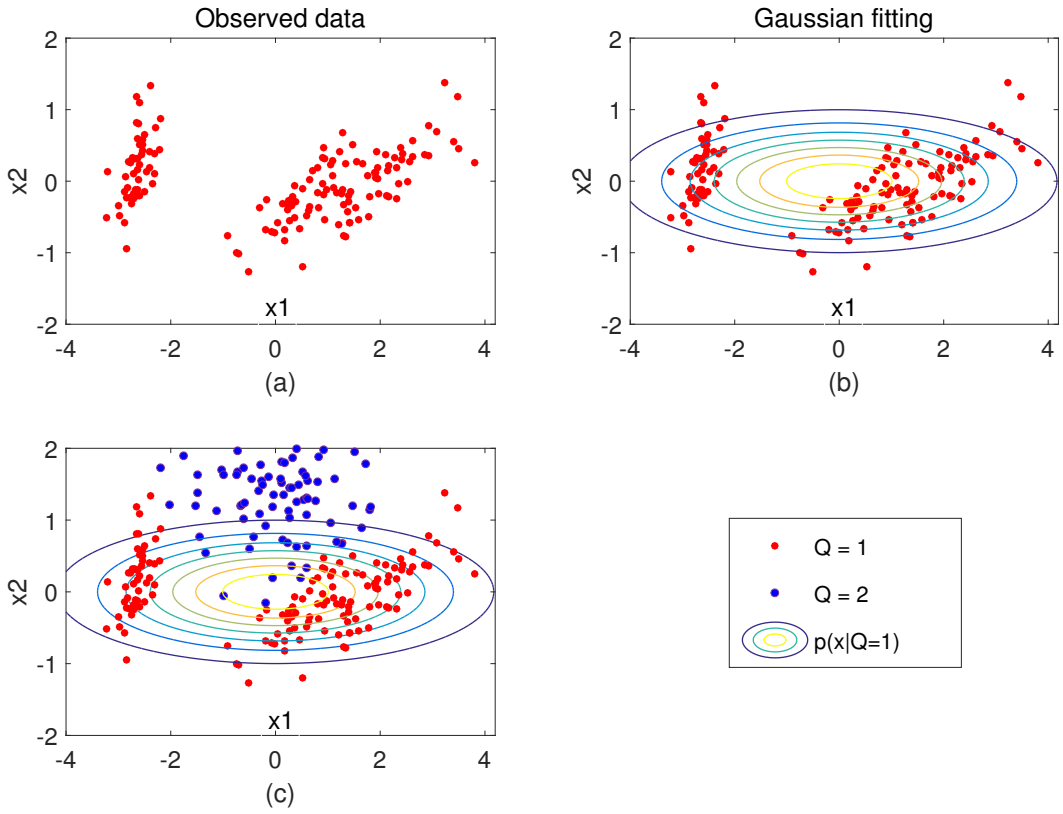


Figure 3.4: Fitting $p(x|Q=1)$ with Gaussian distribution.

GMM is a parametric probability density function represented as a weighted sum of Gaussian component densities (Reynolds, 2015). If random variable $p(x|Q=1)$ is subject to certain GMM, it can be written as:

$$p(x|Q=1) = \sum_{i=1}^{m} w_i \mathcal{N}(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}) \tag{3.13}$$

where m is the number of mixture components, w_i is the weight of i^{th} Gaussian distribution with $0 \leq w_i \leq 1, \sum_{i=1}^{m} w_i = 1$, and $\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}$ are the mean and covariance of i^{th} Gaussian distribution, respectively.

Figure 3.5 shows the fitting results by GMM with 2 and 3 mixture components, respectively. We can see that with 2 mixture components (Figure 3.5a), the probability contour is much better than it is in Figure 3.4b. And with 3 mixture components, the fitting result looks even better in Figure 3.5b.

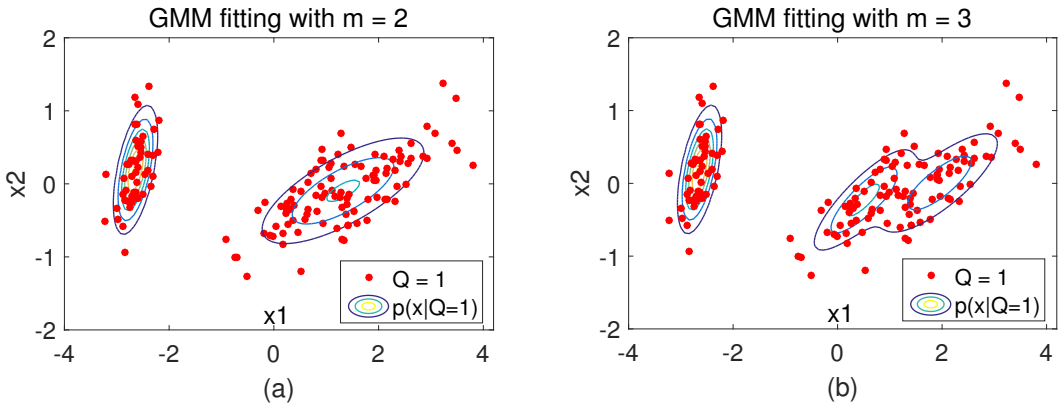


Figure 3.5: Fitting $p(x|Q = 1)$ with GMM by different numbers of mixture components.

From the probability contour in Figure 3.6, it can be found that using GMM with 3 mixture components to fit the conditional probability density $p(x|Q = 1)$, data from $Q = 1$ and $Q = 2$ can be classified in an efficient way. This is how GMM works to fit probability distribution.

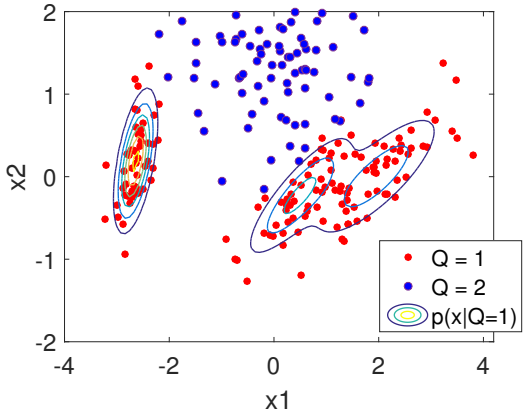


Figure 3.6: The classification result of using GMM with 3 mixture components.

3.1.4 Other ML models

Although BN and GMM is the main ML models detailed in this dissertation, however other ML models e.g. support vector machine (SVM), naive Bayes (NB), decision tree (DT) and k-nearest neighbors (KNN), are also important models in the field of machine learning.

Support vector machine

The support vector machine (SVM), in dealing with classification problems, is a supervised learning method that generates hyperplane functions from a set of labeled training data. With the decision boundary functions, new data can thus be classified into different classes.

Suppose we have data come from two classes. An SVM classifies data by finding the best hyperplane that separates all data points of one class from another. The best hyperplane for an SVM is with the largest margin between the two classes. The margin is the maximal width of the slab parallel to the separating hyperplane that has no interior data points (Wang, 2005). The support vectors are the data points that are closest to the hyperplane, which are on the boundary of the slab. Figure 3.7 illustrates these definitions.

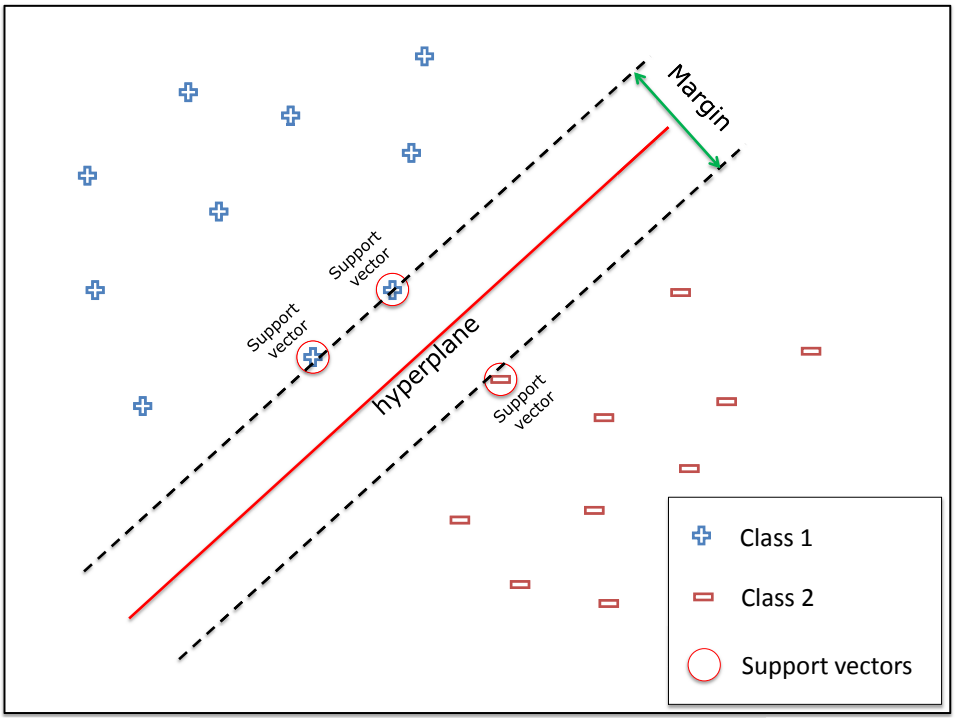


Figure 3.7: An example of how SVM works to classify data from two classes.

The above example is a linear classification case, which means that the best hyperplane can be described by a linear function. For the case whose hyperplane cannot be presented by a nonlinear function (in the space whose dimension is the same as the input data) is the non-linear classification problem. For non-linear classification problems, kernel function $K(x, x^{'})$ is used to transform the data from the current space to a new space where the hyperplane can be described as a linear function. This process can be illustrated in Figure 3.8. Common kernel functions are linear kernel, polynomial kernel and Gaussian kernel. The chosen of certain kernel function in a SVM is depending on the training data, and the guidance can be found in Scholkopf and Smola (2001) and Ben-Hur and Weston (2010).

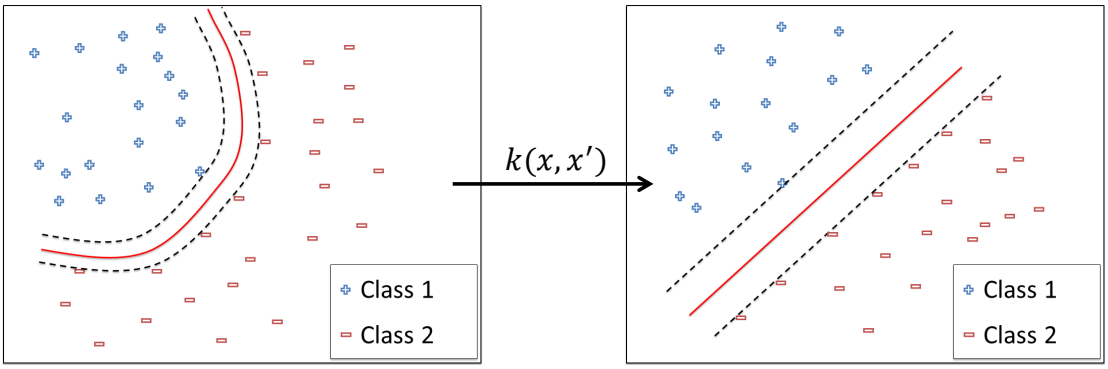


Figure 3.8: Using kernel function to transform data from one space to another, where the hyperplane used for classification can be described by a linear function.

The advantages of support vector machine is that it performs well and memory effective even with high dimensional features, however, when the number of feature dimension is much greater than the number of training samples, over-fitting could happen (Smola and Schölkopf, 2004).

Naive Bayes

The naive Bayes (NB) classifier is a supervised learning algorithm based on Bayes theorem with the simple assumption of conditional independence between each feature set given different classes. Assume a set of training data with feature set $x = [x_1, ..., x_t..., x_n]^{\mathrm{T}}$ (x_i means the i-th feature) are labeled by class C, so based on Bayes theorem the predicted probability given features can be written as:

$$P(C|x_1, x_2, ..., x_n) = \frac{P(C) \cdot P(x_1, x_2, ..., x_n|C)}{P(x_1, x_2, ..., x_n)}, \tag{3.14}$$

and with the conditional independent assumption:

$$P(x_1, x_2, ..., x_n|C) = \prod_{i=1}^{n} P(x_i|C), \tag{3.15}$$

thus equation (3.14) is simplified to:

$$P(C|x_1, x_2, ..., x_n) = \frac{P(C) \cdot \prod_{i=1}^{n} P(x_i|C)}{P(x_1, x_2, ..., x_n)}. \tag{3.16}$$

Since $P(x_1, x_2, ..., x_n)$ is a constant, equation (3.16) is then written as:

$$P(C|x_1, x_2, ..., x_n) \propto P(C) \cdot \prod_{i=1}^{n} P(x_i|C), \tag{3.17}$$

where $P(x_i|C)$ is the conditional distribution needs to be fitted. The common distributions used to fit the conditional distribution are Gaussian distribution, kernel distribution as well as multivariate multinomial distribution etc., which are detailed in Manning, Raghavan, and Schütze (2010).

Naive Bayes classifiers can be extremely fast compared to more sophisticated methods. The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality (Zhang, 2004). However, the downside is also caused by the conditional independent assumption between features since some features are dependent and thus lead to poor fitting problem.

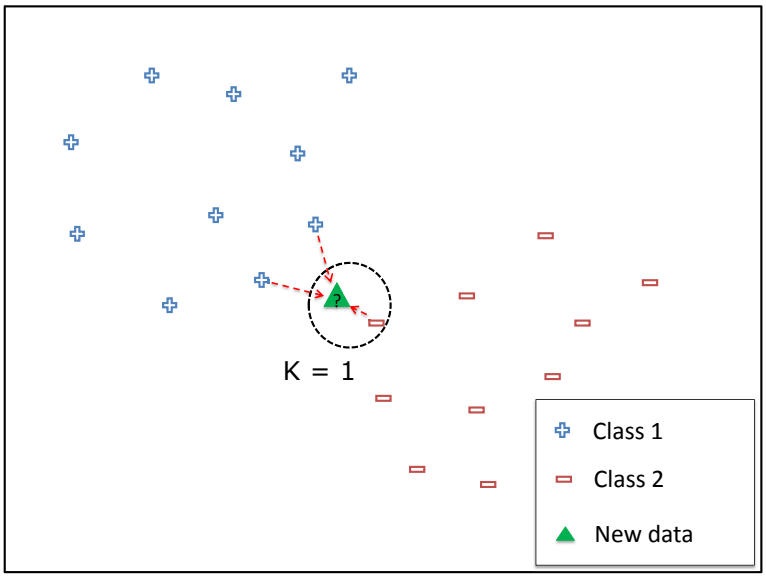
K-nearest neighbor

The nearest neighbor method for classification is a type of instance-based learning, which means that it does not attempt to construct a general internal model, like the mentioned SVM and NB, but simply stores instances of the training data. Given new data, the decision of classifying is computed from a simple majority vote of the nearest neighbors of each new data points: a query point is assigned the data class which has the most representatives within the nearest neighbors of the point (Pedregosa et al., 2011).

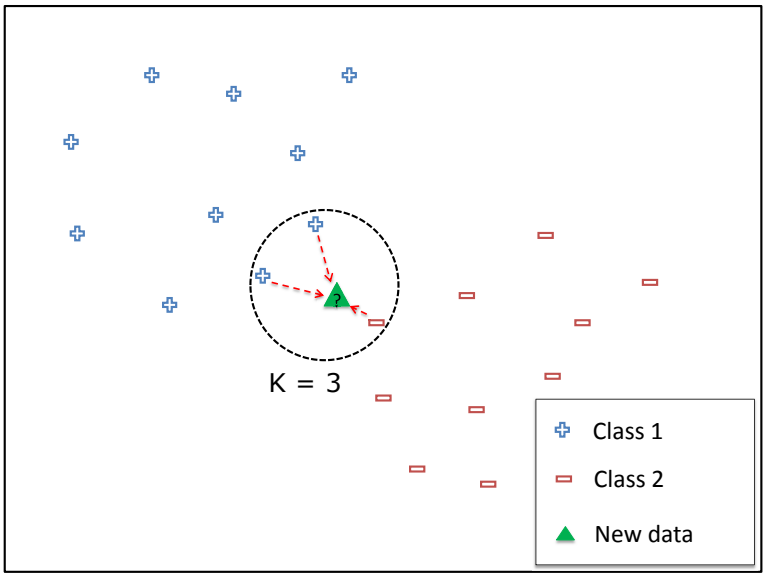
In KNN, K is the number of nearest neighbors. The number of neighbors is the core deciding factor. K is generally an odd number if the number of classes is 2. When K=1, then the algorithm is known as the nearest neighbor algorithm. Let us set an example. Given a new point (green triangle in Figure 3.9), for which a label needs to be predicted, KNN works in the following procedures:

1. Calculate distance: there are various metrics to determine the distance between the new point and its neighbors, e.g. Euclidean distance, city block metric, Mahalanobis distance, Minkowski metric, Chebychev distance, Cosine distance, Jaccard distance, Correlation distance etc.. One can choose them depend on certain requirement.
2. Find closest neighbors: select the closest neighbors depends on K. If K=1, then only one neighbor is selected (Figure 3.9a) and for K=3, three neighbors are selected (Figure 3.9b).
3. Vote for labels: classify points by the majority votes of its k neighbors. Each object votes for their class and the class with the most votes is taken as the prediction. In Figure 3.9b three neighbors are selected, with two points voting for class 1 and one for class 2, so the new point is predicted as class 1.

From the above example we can see that K does matter for new point prediction, however, there are no optimal number of neighbors that suits all kind of data sets. Each dataset has its own property. Generally, K can also be chosen by generating the model on different values of K and check their performance. The detailed guidance of choosing K can be found in (Friedman, 1997).



(a) KNN finding closest neighbors for the case K=1.



(b) KNN finding closest neighbors for the case K=3.

Figure 3.9: An example of how KNN works for classification of new data.

Decision tree

Decision Tree (DT) is a non-parametric supervised learning method, which creates a model that predicts the new data for which class it comes from by learning simple decision rules inferred from the features. Here is an example to show how DT works for making decision of *Whether to play tennis?*[4]. The decision of *Whether to play tennis?* is influenced by several conditions, e.g. outlook of the weather, temperature, humidity, wind etc. The decision tree of *Whether to play tennis?* can be drawn in Figure 3.10.

[4]This example is from the presentation in a machine learning course made by Prof. Dr. Martin Riedmiller of Albert-Ludwigs-Universität Freiburg, full source link: http://ml.informatik.uni-freiburg.de/former/_media/teaching/ss10/03_decisiontrees.pdf, visited on 31.07.2019

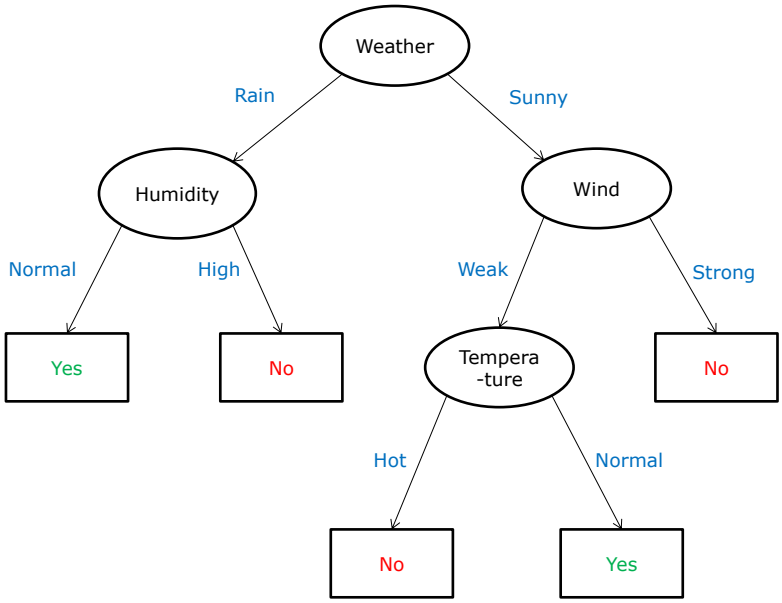


Figure 3.10: The decision tree of *Whether to play tennis?*.

A decision tree is drawn upside down with its root at the top, which is *Weather* in this case. The bold text in ellipse represents a condition node, and the blue texts are the status of the condition. Based on the condition node, the tree splits into branches. The end of the branch that cannot split anymore is the leaf, in this case, whether to play tennis, represented as red and green text respectively. In the real case of classification, a decision tree is trained by more features and has more branches with more complex structure. In practice, it is tricky to set when the branch to split and when stop splitting. Because if the dataset has a large number of features, it results in large number of splits, which in turn gives a huge tree. Such trees are complex and can lead to over-fitting problem. For more guidance of configuring DTs see (Safavian and Landgrebe, 1991).

The decision tree model for classification is easy to understand and only requires a small amount of training dataset. In addition, little data preparation is needed to train a decision tree, i.e. data normalization and dummy variables are not necessary because decision tree can use the original feature to split branches. However, the decision tree can be also sensitive to the data, because small variations in the data might result in a completely different tree being trained.

3.2 Parameter learning

In this thesis, the expectation-maximization (EM) algorithm is used to learn the parameters of a GMM (Bishop, 2006). The EM is an iterative algorithm that initiates from a random $\boldsymbol{\theta}$ and then proceeds to iteratively calculate the log-likelihood of $\boldsymbol{\theta}$ and update till the likelihood becomes convergence. Each iteration consists an E-step and M-step (Bishop, 2006):

- Initialize $\boldsymbol{\theta_i} = [w_i, \boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}]$ and evaluate the initial value of the log-likelihood:

$$l(\boldsymbol{\theta}) = log(p(x|\boldsymbol{\theta})) = \sum_{j=1}^{N} log \left\{ \sum_{i=1}^{m} w_i \mathcal{N}(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}) \right\} \tag{3.18}$$

  where $x = [x_1, ..., x_N]$.

- E-step: here a latent variable γ_i is introduced to represent the posterior probabilities of $P(\boldsymbol{\theta}|x)$, and based on Bayes rule, it could be written as

$$\gamma_i(x) = \frac{w_i \mathcal{N}(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i})}{\sum_{j=1}^{m} w_j \mathcal{N}(\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j})} \tag{3.19}$$

- M-step: Re-estimate the parameters using current posterior probabilities, so the new $\boldsymbol{\theta}$ can be written as

$$\begin{aligned} \boldsymbol{\mu_i^{New}} &= \frac{\sum_{j=1}^{N} \gamma_i(x_j) x_j}{\sum_{j=1}^{N} \gamma_i(x_j)} \\ \boldsymbol{\Sigma_i^{New}} &= \frac{\sum_{j=1}^{N} \gamma_i(x_j)(x_j - \boldsymbol{\mu_i})(x_j - \boldsymbol{\mu_i})^{\mathrm{T}}}{\sum_{j=1}^{N} \gamma_i(x_j)} \\ w_i^{New} &= \frac{1}{N} \sum_{j=1}^{N} \gamma_i(x_j) \end{aligned} \tag{3.20}$$

  After estimation of new $\boldsymbol{\theta^{New}}$ update the log-likelihood:

$$l(\boldsymbol{\theta^{New}}) = \sum_{j=1}^{N} log \left\{ \sum_{i=1}^{m} w_i^{New} \mathcal{N}(\boldsymbol{\mu_i^{New}}, \boldsymbol{\Sigma_i^{New}}) \right\} \tag{3.21}$$

The iteration equation (3.19)-(3.21) are repeated until $l(\boldsymbol{\theta^{New}}) - l(\boldsymbol{\theta}) < \varepsilon$, where ε is a very small positive value.

3.3 Evaluation method

3.3.1 Receiver operating characteristic

One of the most popular methods of evaluating classification or prediction performance are receiver operating characteristic (ROC) curves. Regarding to LC behavior research using ROC curve method can be found in McCall et al. (2007), Liebner et al. (2013), Peng et al. (2015), Doshi and Trivedi (2009), and Lethaus, Baumann, et al. (2013).

The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. TPR and FPR are computed by

$$\begin{aligned} TPR &= \frac{TP}{TP + FN} \\ FPR &= \frac{FP}{TN + FP} \end{aligned} \tag{3.22}$$

where TP, TN, FP and FN are true positives, true negatives, false positives, and false negatives, respectively. Thus TPR and FPR are the function of threshold T, termed as

$$\begin{aligned} TPR &= \int_{T}^{\infty} f_1(x)dx \\ FPR &= \int_{T}^{\infty} f_2(x)dx \end{aligned} \tag{3.23}$$

3.3.2 Area under curve

The metric of using ROC curve to compare model performance over different methods is to calculate the value of area under curve (AUC) of each ROC curve. AUC is given by Fawcett (2006):

$$AUC = \int_{\infty}^{-\infty} TPR(T)FPR'(T)dT, \tag{3.24}$$

where T is the different threshold setting. Basically, AUC is ranging from 0 to 1 and a larger AUC value indicates better performance.

3.3.3 Cross-validation

Training a prediction model and testing it on the same data is a methodological mistake. This would lead to over-fitting, which means the model just repeats the labels of the samples that it has just trained with and could get a perfect score with such trained data, however, would fail to predict anything useful on non-trained data. To avoid over-fitting, it is common practice when performing a supervised machine learning experiment to hold out part of the available data and train the model with the rest. In other words,

training data and test data should be separated. This method is called cross-validation (CV).

One popular CV method is k-fold cross-validation. The original sample is randomly partitioned into k equal sized subsets. Of the k subsets, a single subset is retained as the validation data for testing the model, and the remaining $k-1$ subsets are used as training data. The cross-validation process is then repeated k times, with each of the k subset used exactly once as the validation data. The k results can then be averaged to produce a single estimation (McLachlan, Do, and Ambroise, 2005).

3.4 Summary

In this chapter, the core machine learning mode, i.e. Bayesian network and Gaussian mixture model, as well as the dominated parameter learning algorithm, i.e. the expectation–maximization algorithm are detailed. At the same time, support vector machine, Naive Bayes, K-nearest neighbor as well as decision tree are also explained. In order to explain the math behind each model in an easy-to-understand way, this dissertation uses some lively examples to uncover the mystery of the theories. In addition, evaluation methods that are used for evaluating classification result are also introduced here. All these theories are the backbone of this dissertation.

4

Experiment 1 - Prediction of driver lane-change behavior based on a driving simulator experiment

A substantial portion of this chapter is based on the above paper.

4.1 Introduction

In chapter 2, the related works regarding to prediction of driver LC behavior have been introduced, however, to design a comprehensive framework is no easy task. There are still some aspects that should be taken into account or can be improved.

Contextual traffic

In chapter 2, we mentioned two types of lane-change, i.e. mandatory lane-change (MLC) and discretionary lane-change (DLC). Since the purposes of MLC and DLC are different, the decision-making process is different as well. Research found that most drivers who were involved in LC crashes did not attempt an avoidance maneuver (Olsen, Lee, and Wierwille, 2005). This suggests that the driver did not see or was unaware of the presence of another vehicle or crash hazard (Tijerina, 1999). In this situation, the driver fails to understand the danger of the current driving context.

Actually, contextual traffic is closely related to driver behavior. For instance, it impacts driver gaze behavior. Lee, Olsen, Wierwille, et al. (2004) found that driver

glance duration can increase by 0.25 s on average (a 20% increase) in driving situation where an overtaking vehicle appeared, in comparison to the situation with no traffic involved. More specifically, single glance duration was ranging from 1.1 s to 1.8 s (Mean = 1.25 s) when no overtaking occurred in the adjacent lane, and from 1.0 s to 2.3 s (Mean = 1.5 s) when overtaking occurred in the adjacent lane. Overall, the road traffic caused a large (50% to 85%) increase in both total and visual input times. Without the road traffic, visual search times were 3.7 s for left LC and 3.4 s for right LC. If there were more road traffic, visual search times were 6.1 s for left LC and 4.5 s for right LC. Considering that driver gaze behavior is an important predictor of driver LC behavior, the effect of *contextual traffic* should be considered. Beggiato et al. (2018) found that by monitoring driver gaze behavior the number of glances in a sequence was primarily associated with the road traffic density on the target lane. There are more vehicles on the target lane, the more and longer the driver glancing mirror behavior can be observed. Therefore, to predict driver LC behavior, the current contextual traffic should be taken into account.

Driving style

As long as the driver remains a part of the control loop, driving and safety behaviors are more than just the mechanical operation of a vehicle (Hennessy, 2011), but also affected by the driving style of the driver. The concept of driving style can be termed as either a dynamic behavior of a driver on the road (Murphey, Milton, and Kiliaris, 2009) or an intrinsic driving habit (Saad, 2004; Sagberg et al., 2015).

The former concept tends to consider driving style as a transient behavior, which means that a driver can be aggressive at one time period but normal at other situation. Murphey, Milton, and Kiliaris (2009) classified the dynamic driving style as calm driving, aggressive driving and normal driving. Velocity, acceleration and jerk were used as the measurements. The result suggested that drivers with aggressive driving have more fuel consumption.

The latter concept of driving style, however, is more popular since different driving habits may affect the design, effectiveness, and feedback mechanisms of future ADAS. For example, Doshi and Trivedi (2010) found that aggressive drivers are more consistent in behaviors and more predictable than non-aggressive drivers. In a paper by Johnson and Trivedi (2011), it was found that aggressive and non-aggressive drivers tend to behave in different ways in the similar situation. Non-aggressive drivers are quantifiable and significantly more compliant to feedback from ADAS. It indicated that the populations of non-aggressive drivers need to be further split in order to detect more significant behavioral trends. By monitoring driver gaze behavior, the results shows that when the driver is conducting a LC to overtake cars, conservative drivers prefer a higher time-headway (Mean = 1.76 s) than either the neutral drivers (Mean = 1.23 s) or the aggressive drivers (Mean = 1.15 s). So, it is reasonable to say that drivers with different

driving styles would have different preference when initiating a LC in an overtaking scenario (Fairclough, May, and Carter, 1997).

In conclusion, the driving style should be considered in prediction of driver LC behavior.

Data labeling method

The related works regarding to the prediction of driver behavior mainly use supervised machine learning models (Pentland and Liu, 1999; Kasper et al., 2012; Liebner et al., 2013; Jain, Koppula, Raghavan, et al., 2015; Peng et al., 2015). The fact is that to train good supervised learning models are strongly depended on high-quality labeled data. To some degree, model performance is closely related to the quality of the labeled data. Specifically for prediction of LC, how to label LC and lane-keep (LK) data samples from the raw dataset is a big issue because different training datasets could train different parameters.

Most of the related works applied a time-window (TWL) to label LC datasets, termed as the TWL method. A time-window with a fixed length of duration is used to label time series data. This method is commonly used by labeling two adjacent events in time series. By using the TWL method, data samples, which are within certain time-window before the moment that one specific part of the vehicle just hits the lane boundary, are labeled as LC datasets (Mandalia and Salvucci, 2005; Doshi and Trivedi, 2009; Lethaus, Baumann, et al., 2013; Doshi and Trivedi, 2008; Morris, Doshi, and Trivedi, 2011).

The limitation of the TWL method is that the fixed TW that is used to distinguish LC and LK events is the same for all the drivers, regardless of different driving situations. However, in practice the driver could start to prepare for a LC either early or late depends on traffic situation and his/her personal preference. In other words the suitable TW depends on the contextual traffic as well as driving style. For instance, what may happen in a complex situation is that the driver attempts to make LC and afterwards he/she aborts his/her intention. In other words, driver LC behavior is very driver and situation specific. Rehder et al. (2016) made a statistical analysis which counted how early the driver tends to start a LC behavior until the front wheel of the car just hits the lane boundary. The duration of this process suffers a high variance of 2.42 s, and this large difference does impair the prediction performance based on the results.

Therefore, a data labeling method that considers LC behavior case by case could improve the quality of the labeled datasets.

Motivation of this study

In this chapter we aim to design a framework which can be used for prediction of driver LC behavior. The experiment was conducted in a seat-box based driving simulator. For the purpose of making the simulated driving environment easier but without loss of

generality, the driving scenario was modeled on a two-lane highway and for the case of left lane-change (LLC). Inspired by the related works, several aspects that are discussed above could be improved through this study:

1. Considering contextual traffic, a cell-grid method is implemented to model the current driving situation.
2. Considering driving style, all the participants are classified into three groups, i.e. high aggressive, medium aggressive and low aggressive driving style by using a behavioral-psychological questionnaire.
3. In order to obtain high-quality labeled datasets for model training, a gaze-based labeling (GBL) method is implemented. With the GBL method, LC datasets can be labeled based on the moment that the driver really tends to make LC rather by assuming a fixed TW.
4. In the process of preparing for training datasets, the labeled datasets are manually organized by different driving scenarios as well as by different driving styles.

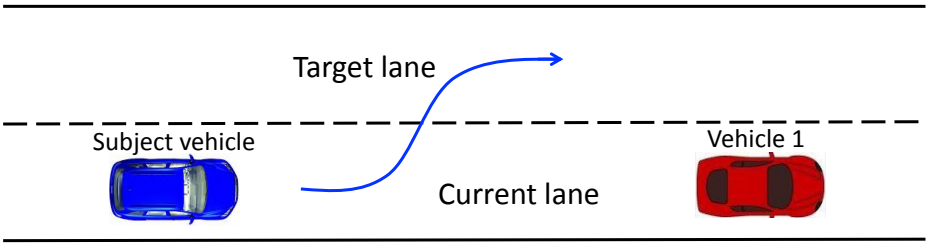
4.2 Driving scenario

4.2.1 Modeling contextual traffic

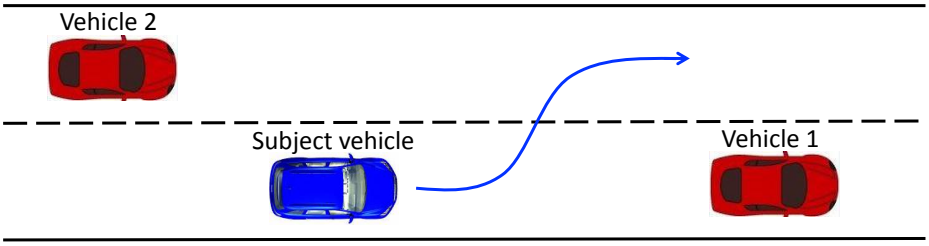
In order to model contextual traffic, the road traffic should be specified. Two kinds of vehicles, i.e. subject vehicle and nearest surrounding vehicles, are considered in the specified scenarios (Figure 4.1). The subject vehicle (blue) is the host vehicle drove by the participant. Vehicle 1, Vehicle 2, and Vehicle 3 are the nearest surrounding vehicles (red) to the subject vehicle on the current lane and/or on the target lane. The target lane is to which the subject vehicle wants to change. In order to term the scenarios easily, we define the LC scenarios as follows in Figure 4.1:

- *Scenario lead only* (Figure 4.1a): There is no vehicle on the target lane within a specific range. The only surrounding vehicle is Vehicle 1 in front of the subject vehicle.
- *Scenario lead + adjacent behind* (Figure 4.1b): The subject vehicle intends to make LLC to overtake the slow leading vehicle, meanwhile, a fast moving vehicle (Vehicle 2) is approaching from left behind on the target lane.
- *Scenario lead +2 adjacent* (Figure 4.1c): The subject vehicle intends to make LLC to overtake the slow leading vehicle, by then a left front vehicle (Vehicle 3) and a left behind vehicle (Vehicle 2) are on the target lane.

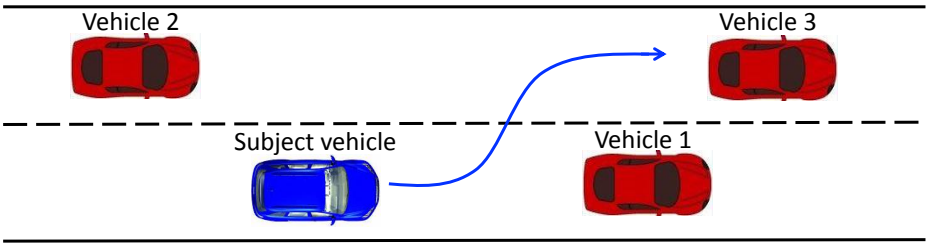
There are some existing works regarding to driving contextual traffic modeling. A potential field diagram composed of bubbles with different dynamic sizes has been proposed to describe the dynamic relationship between the subject vehicle and



(a) A slow leading vehicle in front of the subject vehicle on the current lane.



(b) A vehicle on the target lane with a slow leading vehicle in front of the current lane.



(c) Two vehicles on the target lane with a slow leading vehicle in front of the current lane.

Figure 4.1: Illustration of the defined LLC scenarios, (a) Scenario lead only (b) Scenario lead + adjacent behind and (c) Scenario lead +2 adjacent.

surrounding vehicles (Woo et al., 2016). However, it is not a direct modeling method which can be used for driving contextual traffic analysis. Leonhardt and Wanielik (2017) developed a probabilistic situation assessment model to evaluate the safety state of the subject vehicle with surrounding vehicles, however, it aims at recognizing driver behavior rather than modeling driving scenarios.

To easily describe the relationship between the subject vehicle and the surroundings, one of the most popular approaches is to segment the surrounding traffic into grid cells (Do et al., 2017; Nilsson, Silvlin, et al., 2016; Kasper et al., 2012) and then model the dynamic relationship of these grid cells. The occupancy status of each cell is represented by a binary value, i.e. occupied or empty. Since the experiment in this study is only focused on the two-lane highway, a five-cell contextual grid is enough to describe the contextual traffic which is demonstrated in Figure 4.2:

- f: the cell in front of the subject vehicle on the current lane.
- b: the cell behind the subject vehicle on the current lane.
- l: the cell on the left of the subject vehicle on the target lane.
- fl: the cell in front of the subject vehicle on the target lane.
- bl: the cell behind the subject vehicle on the target lane.

where $f, b, l, fl, bl \in \{0, 1\}$ are the status of each cell. The status in the corresponding cell is 1 if the cell is occupied by vehicles otherwise it is 0 for unoccupied. The length of cell l is the length of the subject vehicle plus a short safety distance. Here, the length of cell l is set as 5 m. According to Ayres et al. (2001), the preferred car-following time-headway ranges from 1 s to 2 s on highways. Based on the traffic density study in Fairclough, May, and Carter (1997), the average time-headway for conservative drivers to initiate overtaking maneuvers is 1.76 s. Time-headway is often determined by traffic counting devices that determine the inter-arrival times of vehicles. In our case, the time-gap (net time-headway) (Ha, Aron, and Cohen, 2012) is used to measure the inter-vehicle distance since it is usually measured via radars or LiDAR mounted on the subject vehicle and presumably reflected by the rear of the leading vehicle (J2944, 2013). The length of the front and rear cell grids is set to a time-gap of 2 s, see Figure 4.2.

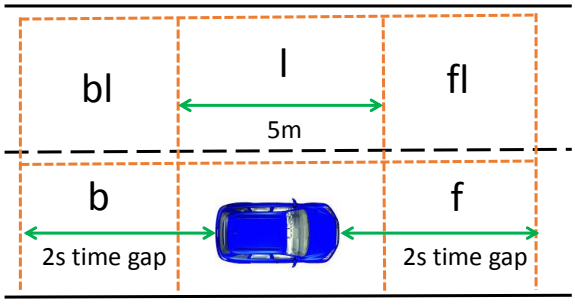


Figure 4.2: Illustration of the occupancy grid on a two-lane highway.

4.2.2 Feature extraction

Selecting suitable features is beneficial to improve model performance. Both TTC (Hayward, 1972) and time-gap are the preferable features to analyze or model LC behavior (Zhao, Lam, et al., 2017; Wakasugi, 2005). TTC is the time required for two vehicles to collide if they continue at their present speed and on the same path. It is usually used to measure collision risk (Kusano and Gabler, 2011). Time-gap is often used to assess safe car-following distance (Tordeux, Lassarre, and Roussignol, 2010). We select time-gap and TTC as prediction features which are defined according to J2944 (2013). The parameters of the definition can be seen in Figure 4.3:

- TTC with respect to the vehicle in cell f:

$$ttc_f = \frac{d_1}{v_0 - v_1}, \tag{4.1}$$

  where d_1 is the distance between the front edge of the subject vehicle and the rear edge of Vehicle 1, v_0 and v_1 are the speed of the subject vehicle and Vehicle 1, respectively.

- Time-gap with respect to the vehicle in cell f:

$$tgap_f = \frac{d_1}{v_0}, \tag{4.2}$$

- TTC with respect to the vehicle in cell bl:

$$ttc_{bl} = \frac{d_2}{v_2 - v_0}, \tag{4.3}$$

  where d_2 is the distance between the front edge of vehicle 2 and the rear edge of the subject vehicle. v_2 is the speed of vehicle 2.

- TTC with respect to the vehicle in cell fl:

$$ttc_{fl} = \frac{d_3}{v_0 - v_3}, \tag{4.4}$$

  where d_3 is the distance between the front edge of the subject vehicle and the rear edge of Vehicle 3. v_0 and v_3 are the speed of the subject vehicle and Vehicle 3, respectively.

When the speed of the subject vehicle is close to the speed of Vehicle 1, then equation (4.1) approaches to infinity. To avoid this case, we use the inverse of ttc_f, noted as ttc_f^{-1}, as the feature. Similarly, we compute the inverse of ttc_{fl}, ttc_{bl}, $tgap_f$ and then obtain ttc_{fl}^{-1}, ttc_{bl}^{-1}, $tgap_f^{-1}$, respectively.

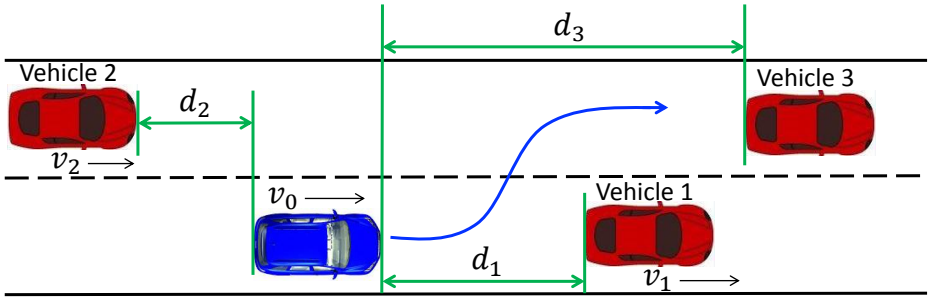


Figure 4.3: Illustration of the parameters regarding to the features.

4.3 Experiment

4.3.1 Experimental setup

Driving simulator

The experiments were conducted in a seat-box based driving simulator. The steering wheel control system was a Fanatec ClubSport Wheel Base V2.5 with torque feedback. The software we used to create the driving scenario is OpenDS Pro 3.5[5]. There are several basic driving scenarios such as urban roads, highways as well as countryside. In order to design the driving task, the basic driving scenarios were extended to a two-lane motorway scenario used for our driving experiment.

For the surrounding vehicles in the scenarios, we set the speed on four levels, i.e. 100 km/h, 120 km/h, 140 km/h, and 150 km/h. The length of each car is 4 m. Vehicles on the left lane were moving faster than vehicles on the right lane. Figure 4.4 depicts that one participant is doing the driving simulator experiments.

Eye-tracker

The eye-tracker used to monitor driver gaze behavior is a SMI ETG (SMI, 2015), see Figure 4.5. SMI ETG can record video from the first-person view. We process the eye-tracking data through BeGaze 3.6, which is an official software from SMI. In order to map driver gaze behavior, we define 5 areas of interests (AoI) in Figure 4.6, i.e. *Rear mirror*, *Left mirror*, *Right mirror*, *Speedometer* and besides *Wind screen* which is all the reset area of the screen. Thus, AoIs of the participant during the experiment can be extracted frame by frame by BeGaze 3.6.

In order to synchronize data collection from the driving simulator and the eye-tracker, the sampling rate is set to 30 Hz for both devices.

[5]https://opends.dfki.de/ (visited on 04.2016)

Figure 4.4: A participant is doing experiment on the simulator.

Figure 4.5: The eye-tracking used in the experiment. Picture extracted from the link: https://www.smivision.com/eye-tracking/products/mobile-eye-tracking/ (visited on 31.07.2019).

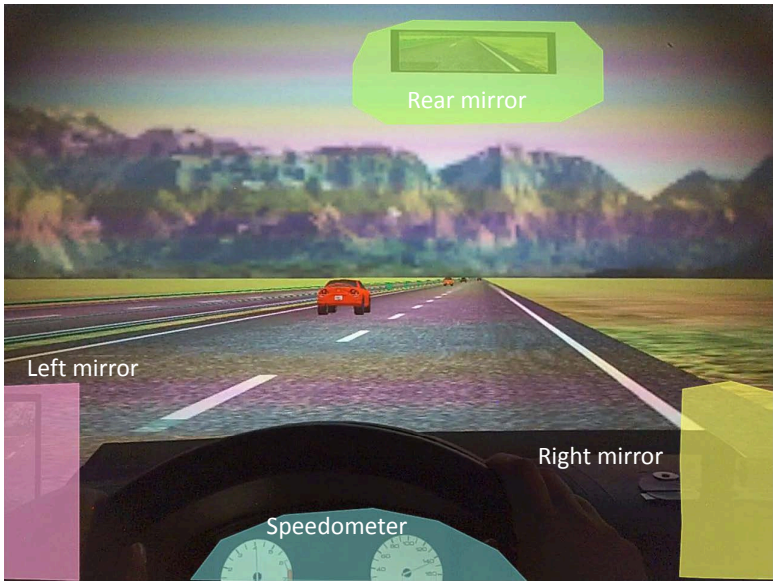


Figure 4.6: The AoIs of the driver in a frame using BeGaze 3.6.

Driving task

Before the experiment, the participants were asked to fill in a demographic questionnaire. The details can be found in Appendix A.1.1. After finishing the demographic questionnaire, the participants were given an oral instruction, including the calibration of the eye-tracker and a short description of the driving task etc. The detailed instruction in text form can be found in Appendix A.1.3 and A.1.4.

After calibration, participants were guided to drive in a warming-up scenario to be familiar with the simulator. The driving tasks were designed based on the three scenarios mentioned in Figure 4.1. Participants were instructed to drive on the right lane throughout the experiment and to only use the left lane for the purpose of overtaking vehicles. Figure 4.8 gives an example of car-following case where the participant is driving on the right lane behind a vehicle. And at the same time we can see from the left view mirror that a vehicle on the left lane is approaching from behind, so the participants have to decide if it is safe to overtake the slow vehicle.

No speed limit signs were set on the highway scenario since it is quite common in Germany. Participants were allowed to drive at their own speed preference to either follow or overtake vehicles. All participants took the identical driving task which lasts about 40 minutes.

4.3.2 Participants

In total, 32 participants (18 females and 14 males) with ages from 21 to 51 years (Mean = 30.4 years) participated in the experiment. All the participants had normal vision and had held their driving licenses for a minimum of 1.5 years and maximum of 27 years (Mean = 10.7 years). Among all participants, 28 of them got paid for the experiment

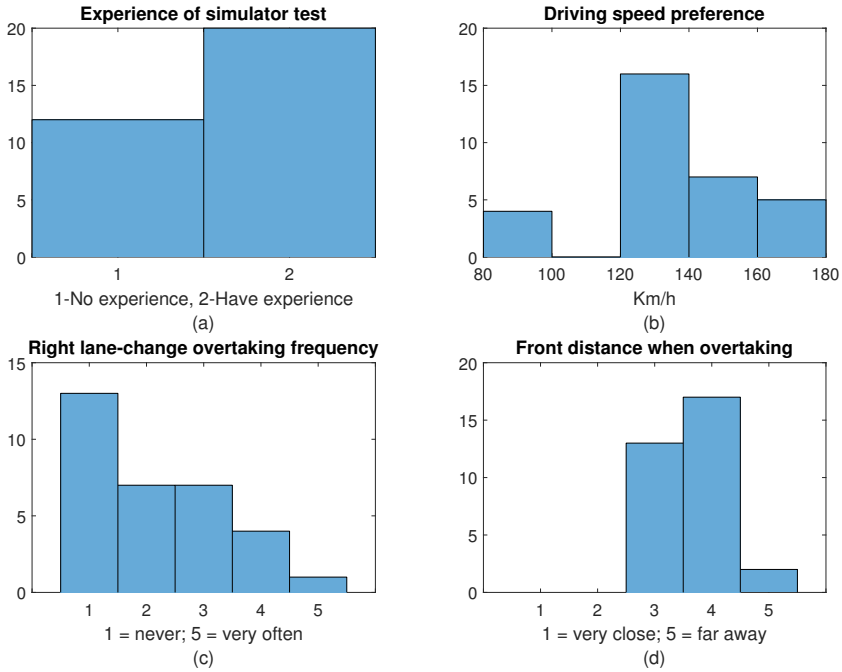


Figure 4.7: Illustration of the driving habits of the participants in histogram.

Figure 4.8: A screen shot of driving simulator and driving scenarios.

and the others were either volunteers or students who need credits. Figure 4.7 gives some driving background information regarding to the participants. We can see that almost 2/3 of them have experience of doing driving simulator experiment and most of them prefer to drive over 120 km/h on highway. This speed preference is in accordance with our speed setting. In addition, most of them rarely overtake cars by executing right lane-change, since it is not allowed based on German traffic rules. More detailed statistics of the participants about their background can be seen in Appendix Figure A.1.

4.3.3 Driving style classification

In order to classify driving style, all the participants were asked to fill in a behavioral-psychological questionnaire (in Appendix A.1.2) used for evaluating the aggressiveness of driving habit proposed by (Glaser and Waschulewski, 2005). A similar method which uses questionnaire to evaluate driving style can be found in French et al. (1993). Vöhringer-Kuhnt and Trexler-Walde (2005) listed the questions which can reflect driving aggressiveness. The correlation tests indicates that the driver who achieves a higher score by given specific questions represents a higher aggressiveness with Cronbach's $\alpha = 0.765$. Based on the their scores, participants were categorized into three groups: 8 drivers with scores between 0 and 5 (Mean = 2.5) were categorized as the low aggressive group, 15 drivers with scores greater than 5 and smaller than 10 (Mean = 7) were categorized as the medium aggressive group, and 9 drivers with scores greater than 10 (M = 14.7) were categorized as the high aggressive group, respectively. The whole statistics about the aggressiveness scores are illustrated in Figure 4.9.

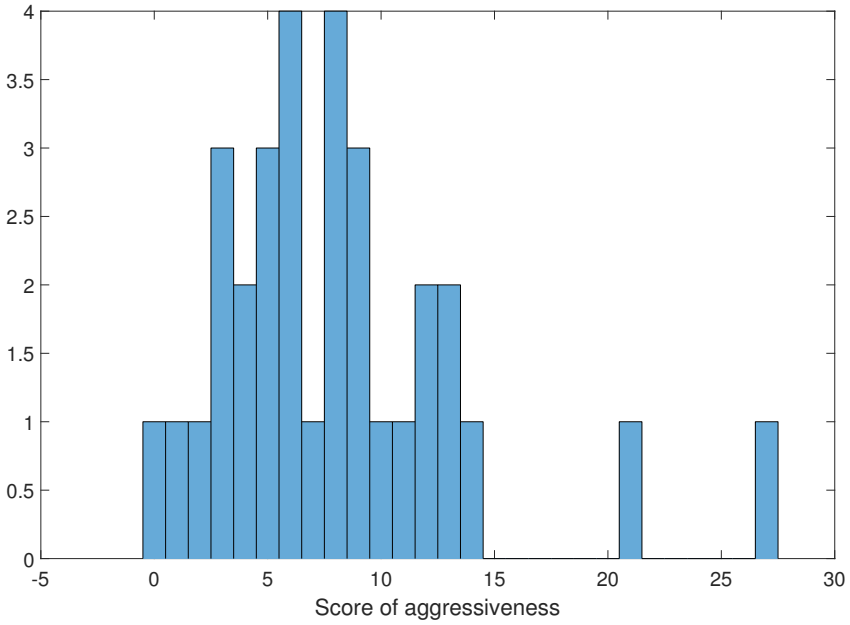


Figure 4.9: Illustration of the aggressiveness scores of the participants in histogram.

4.4 Lane-change data labeling

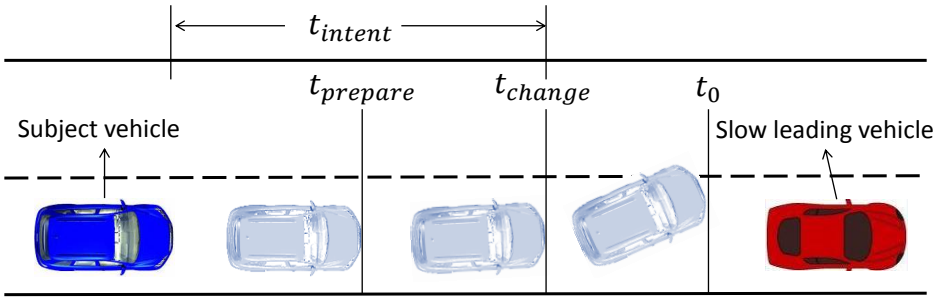


Figure 4.10: The key moments during a lane change course.

To make the data labeling process easy to understand, we define three key moments to describe driver LC behavior as it is depicted in Figure 4.10. Firstly, t_0 is the moment that the front left wheel of the vehicle just runs cross the central dotted line, which can be mapped out by lateral movement of the subject vehicle. Secondly, t_{change} is the moment that the driver really starts to execute an LC maneuver, which can be marked by finding the steering wheel angle threshold (Li, Wang, and Rötting, 2016). However, it is difficult to find the moment t_{intent}. Previous studies found that driver gaze behavior gives us clues to predict upcoming behaviors (Salvucci and Liu, 2002; Fitch et al., 2009). Result suggested that when LLC occurs, the driver takes 65% – 85% chance to glance at the left view mirror (Tijerina et al., 2005). Therefore, the glancing mirror behavior of the driver indicates a LC intention. Inspired by this result, a gaze-based labeling (GBL) method is proposed. This GBL method labels LC datasets based on driver gaze behavior. Since $t_{prepare}$ is different by LC cases and by different drivers, the GBL method could take advantage of driver gaze behavior to capture this moment exactly, compared with the TWL method which uses a fixed time-window to assume t_{intent} (Lethaus, Baumann, et al., 2013; Doshi and Trivedi, 2009).

4.4.1 Gaze-based labeling method

Labeling criterion

The LC decision-making procedure is described as follows: when the driver wants to make LC, he/she will firstly glance at the mirrors to check whether it is safe or not to finish a successful LC maneuver (Lee, Olsen, Wierwille, et al., 2004). In our driving simulator experiment, it was found that 84.37% (27/32) of the participants performed like what it is described above.

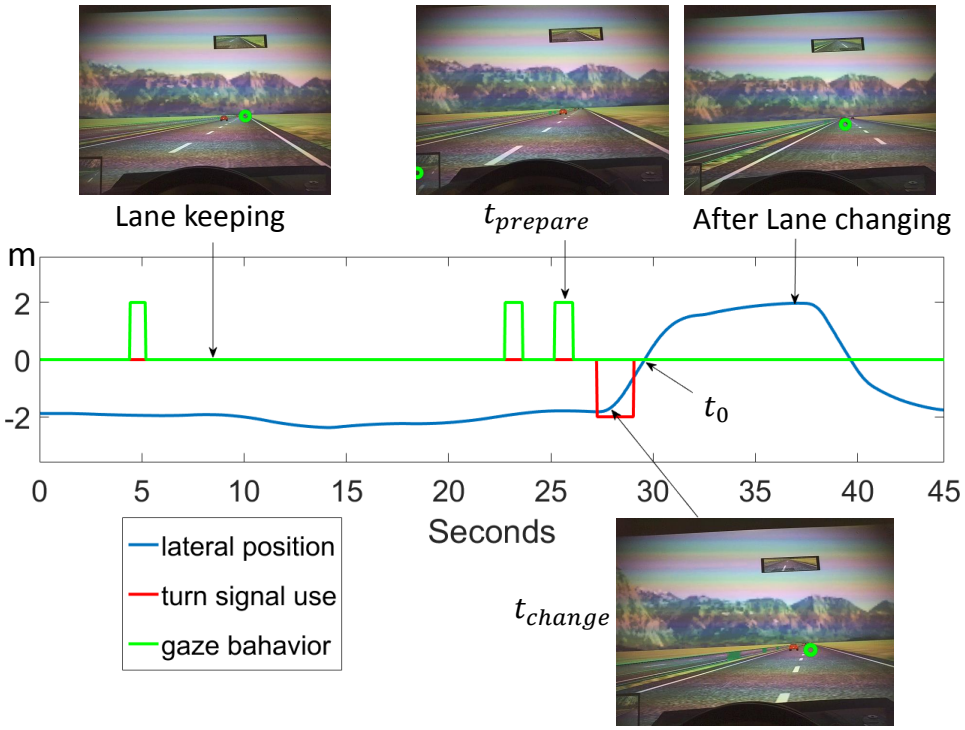
Figure 4.11a sets an example of how driver gaze behavior performs during a LC course. The blue line represents the lateral position of the subject vehicle. The green line, which is a binary signal, represents driver gaze behavior, i.e. glancing at the left-view mirror. If the driver glances at the left view mirror, the signal turns to 2 without unit[6] and if there is no glancing behavior it keeps 0.

The red line represents the status of the left-turn signal also with a binary signal with 0 for switching off and -2 for switching on. The inserted pictures are extracted from software BeGaze 3.6, which were recorded during the experiment. The green points in the pictures are fixations of the participant. By observing fixations, we can clearly know where the driver is looking at during the entire drive.

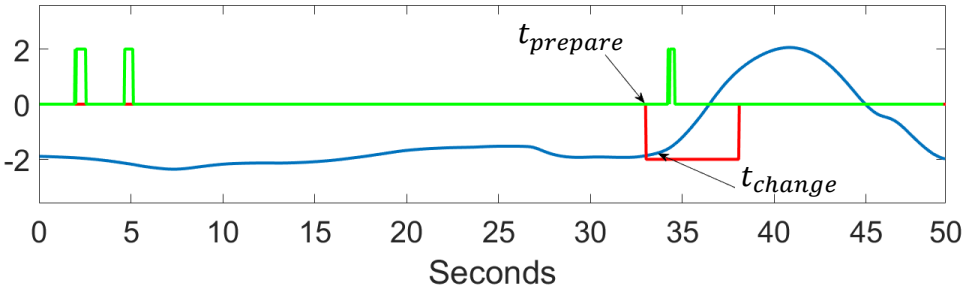
Thus for each LLC case, based on driver gaze behavior, $t_{prepare}$ can be defined as the last fixation on left view mirror (termed as mirror-glancing behavior) before the driver switching on the left-turn indicator. This case is illustrated by the green rectangular wave before the red one, in Figure 4.11a. The green rectangular waves far away from t_0 are not regarded as $t_{prepare}$ since they are just normal mirror-glancing behavior with no following LC maneuver executed. However, in case that a LC which no mirror-glancing behavior is detected before a left-turn signal, we set the moment of switching left-turn indicator on as $t_{prepare}$. This special case can be illustrated in Figure 4.11b, where we can see that the driver switches on the turn indicator but before it there is no mirror-glancing behavior detected.

With the GBL method, time series data between $t_{prepare}$ and t_{change} are labeled as LC datasets. For the problem of classification, because of the unequal class distribution, unbalanced data sets have commonly agreed that the performance of the classifiers tends to be biased towards the majority class (Ganganwar, 2012). In order to obtain balanced training datasets, the same amount of samples should be labeled as LK datasets as it is depicted in Figure 4.11c, where LK data samples are labeled right before $t_{prepare}$ and the number of the samples is equal to LC samples.

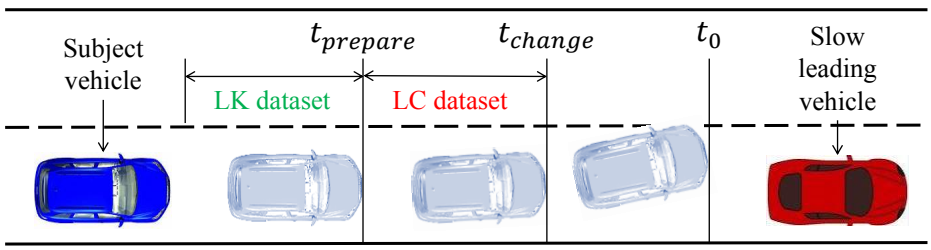
[6]The reason of setting this value is just used for visualization, to match the value of the vehicle position represented in the blue line.



(a) Lane-change course recorded by BeGaze 3.6. This case demonstrates the driver glances at the left-view mirror before switching on the left signal.



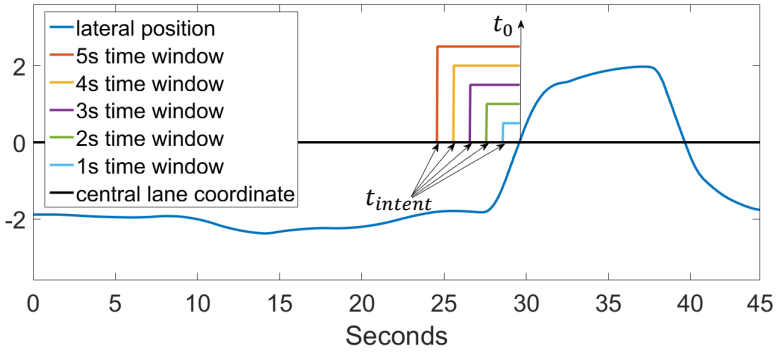
(b) Lane-change course recorded by BeGaze 3.6. This case demonstrates the driver glances at the left-view mirror after switching on the left signal.



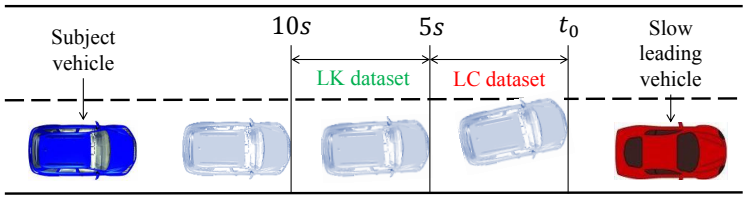
(c) GBL method labeling LC and LK datasets. The number of labeled LC samples and LK samples is equal.

Figure 4.11: Labeling LC and LK data samples to attain balanced datasets.

4.4.2 Time-window labeling method



(a) Using a time-window to emperically define t_{intent} moment.



(b) Labeling LC and LK datasets using TWL with time-window = 5 s.

Figure 4.12: Illustration of the TWL method.

In comparison to the GBL method, the process of TWL method is illustrated in Figure 4.12a. Instead of defining $t_{prepare}$, the TWL method directly uses a time-window with an ad hoc duration of time period before t_0 to label t_{intent}. And then it selects a better window size after evaluation of the classification performance. Take the example of using 5 s as the time-window, LC and LK datasets are labeled as is shown in Figure 4.12b. In order to keep balance labeling, time series data between t_0 and prior 5 s are labeled as LC class, while data between 10 s and 5 s are labeled as LK class.

4.5 Model implementation

In this section, we introduce how to use a BN to model driver LC behavior and how to learn the parameters of the BN. At the same time, the configuration of the SVM and the naive Bayes model are also detailed. In addition, the model evaluation methods are explained.

4.5.1 Bayesian network

Bayesian network (BN) is a directed acyclic graph network and has shown its effectiveness in many fields of prediction, e.g. parsing video events and agents' intent prediction (Pei, Jia, and Zhu, 2011), enemy's tactical intention prediction (Johansson and Falkman, 2006),

user's click intent for web search ranking (Chapelle and Zhang, 2009) and prediction of highway traffic maneuvers (Weidl et al., 2018). Since driver behavior could suffer from various uncertainties of driving contextual traffic and driver status, we should design an algorithm that is capable of dealing with such uncertainties. To this end, a BN is introduced to offer an explicit, graphical, and interpretable representation of uncertain knowledge (Bielza and Larrañaga, 2014). In addition, BN is compatible with many kinds of probability inference, including tabular probability, Gaussian distribution, soft-max function, root function, or Gaussian mixture distribution (Murphy, 1998a). Inspired by its extensive applications and effectiveness, we can use a BN to predict driver LC behavior that is able to consider driving uncertainties[7].

Lane-change Bayesian network

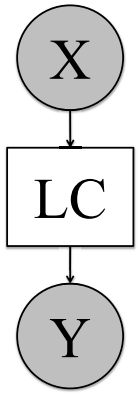


Figure 4.13: Illustration of the lane-change Bayesian network, where node X and Y represent the dynamic driving situation.

The directed acyclic graph of the lane-change Bayesian network (LCBN) is shown in Figure 4.13. The square represents a discrete variable (LC is a binary state) and the cycles represent the continuous variables X and Y that contain driving uncertainties. The meaning of variables in LCBN is detailed as follows:

- The relationship between the subject vehicle and the vehicle in cell f is

$$x = [x_1, ..., x_t..., x_n]^{\mathrm{T}}, \tag{4.5}$$

  where $x_t = [ttc_{f_t}^{-1}, tgap_{f_t}^{-1}]^{\mathrm{T}}$ is the data point at time t and n is the number of data points.

[7] A Bayes Net Toolbox is used to create customized Bayesian networks. This Toolbox is based on Matlab developed by Murphy (Murphy, 1998a), which can be found in the link http://www.cs.utah.edu/ tch/notes/matlab/bnt/docs/usage.html (visited on 31.07.2019)

- The relationship between the subject vehicle and the vehicles in cell fl and cell bl is

$$y = [y_1, ..., y_t ..., y_n]^{\mathrm{T}}, \tag{4.6}$$

  where $y_t = [ttc_{fl_t}^{-1}, ttc_{bl_t}^{-1}]^{\mathrm{T}}$ is the data point at time t and n is the number of data points.

- Node LC is a binary variable of representing whether the driver intends to make LLC:

$$LC = k,\ k \in \{0, 1\}, \tag{4.7}$$

  with $k = 0$ for lane-keeping and $k = 1$ for lane-changing.

In LCBN framework, we set X node at an upper layer and Y node at the bottom layer based on two facts:

1. Whether the driver wants to make LC highly depends on the traffic situation on the current lane in front of the subject vehicle.
2. When the driver makes the final LC decision, the driver would observe the traffic situation on the target lane to ensure safety.

The LCBN model aims to infer the probability of $LC = 1$ given the observations X and Y (gray circles), which can be computed by

$$P(LC = 1|X, Y). \tag{4.8}$$

Based on Bayes theorem, the posterior probability of making LC given the current observations of driving situation is

$$P(LC = 1|X, Y) = \frac{P(Y|LC = 1, X) \cdot P(LC = 1|X)}{\sum_{k=0}^{1} P(Y|LC = k, X) \cdot P(LC = k|X)}. \tag{4.9}$$

In a BN, it is assumed that each variable is independent of its non-descendants in the graph given the state of its parents (Friedman, Geiger, and Goldszmidt, 1997). For our LCBN, LC is the parent of Y, and X is non-descendant of Y. Therefore, variables X and Y are conditional independent to each other given LC, i.e.

$$P(Y|LC = k, X) = P(Y|LC = k). \tag{4.10}$$

And then equation (4.9) can be written as

$$P(LC = 1|X, Y) = \frac{P(Y|LC = 1) \cdot P(LC = 1|X)}{\sum_{k=0}^{1} P(Y|LC = k) \cdot P(LC = k|X)}. \tag{4.11}$$

In order to compute the posterior probability $P(LC = k|X, Y)$ in equation (4.11), we need to estimate the conditional probability distributions $P(LC = 1|X)$ and $P(Y|LC = k)$. Since LC is a binary variable and X is a continuous variable, we use a logistic function to estimate $P(LC = 1|X)$, given by

$$P(LC = 1|X, \beta) = \frac{1}{1 + e^{-\beta^{\mathrm{T}} X}}, \tag{4.12}$$

where β is the parameter with the same dimension of X. If we set X node as a Gaussian node, it is tractable to inference $P(Y|LC = k), k \in \{0, 1\}$ (Murphy, 1998b; Murphy, 1999; Murphy et al., 2001; Murphy, 2012).

In case $P(Y|LC = k)$ does not follow a standard Gaussian distribution, it would lead to poor performance. Just like what is discussed in section 3.1.3, where the model shows poor fitting result if we fit a data samples in Gaussian distribution however it does not follow Gaussian distribution.

Fortunately, previous research found that GMM has a good compatibility with BN in fitting the probability distribution of observations (Dielmann and Renals, 2004; Sun, Zhang, and Yu, 2006). This gives us clues to fit $P(Y|LC = k)$ with GMM.

LCBN with GMM

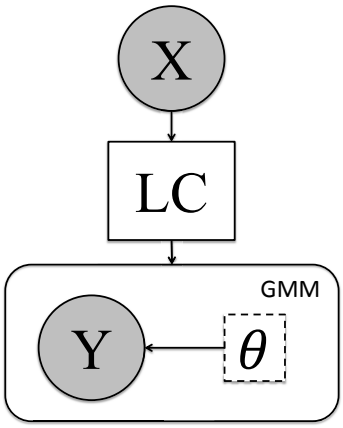


Figure 4.14: Illustration of LCBN incorporated with GMM.

GMM is a parametric probability density function represented as a weighted sum of Gaussian component densities (Reynolds, 2015; Bishop, 2006), given by

$$p(x|w_i, \boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}) = \sum_{i=1}^{M} w_i \mathcal{N}(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}) \tag{4.13}$$

where M is the number of mixture components, w_i is the weight of i^{th} Gaussian distribution with $0 \leq w_i \leq 1, \sum_{i=1}^{M} w_i = 1$, and $\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}$ are the mean and covariance of i^{th} Gaussian distribution, respectively. In order to integrate GMM into LCBN, termed as LCBN-GMM, we extend the LCBN framework by adding a new node $\theta = \{w_i, \boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}\}_{i=1}^{M}$. θ is a hidden node depicted as a dotted square, as is shown in Figure 4.14. Therefore, the posterior probability can also be written as

$$P(LC = 1|X, Y, \theta) = \frac{P(X|LC = 1, \theta) \cdot P(LC = 1|X, \theta)}{\sum_{k=0}^{1} P(Y|LC = k, \theta) \cdot P(LC = k|X, \theta)}, \tag{4.14}$$

where

$$P(LC = 1|X, \theta) = \frac{P(\theta|LC = 1) \cdot P(LC = 1|X)}{\sum_{k=0}^{1} P(\theta|LC = k) \cdot P(LC = k|X)} \tag{4.15}$$

and

$$p(Y|LC = 1, \theta) = \sum_{i=1}^{M} w_i \mathcal{N}(\mu_i, \Sigma_i). \tag{4.16}$$

Thus, the parameters of LCBN-GMM are $\boldsymbol{\xi} = [\beta, \theta]$.

Parameter estimation

In LCBN-GMM, given a dataset of observations $\boldsymbol{O} = [o_1, ..., o_t..., o_N]^{\mathrm{T}}$, then their likelihood is computed by

$$p(\boldsymbol{O}|\boldsymbol{\xi}, \mathrm{LCBN - GMM}) = \prod_{i=1}^{N} p(o_i|\boldsymbol{\xi}, \mathrm{LCBN - GMM}). \tag{4.17}$$

And the log-likelihood can be thus written as

$$\mathcal{L}(\boldsymbol{\xi}) = \sum_{i=1}^{N} log(p(o_i|\boldsymbol{\xi}, \mathrm{LCBN - GMM})). \tag{4.18}$$

Due to the nonlinearity with respect to the parameters, using maximum-likelihood for estimation is not possible (Dempster, Laird, and Rubin, 1977). Here, the expectation-maximization (EM) algorithm is used to estimate hyper-parameters in

LCBN-GMM (Ghahramani, 1998; Bishop, 2006). The iteration procedure of the EM is illustrated in section 3.2.

Before training LCBN-GMM, we need to determine the parameter M in equation (4.13) based on the Bayesian information criterion (BIC) (Caner, 2009):

$$BIC = -2 * \mathcal{L}(\boldsymbol{\xi}) + p * log(q) \tag{4.19}$$

where $\mathcal{L}(\boldsymbol{\xi})$ is the estimated log-likelihood in equation (4.18), $q = 2$ is the amount of observations in LCBN-GMM with two observations, and $p = 4$ is the amount of estimated number of parameters in GMM. Generally, a lower BIC value represents better fitting result (Steele and Raftery, 2010). Figure 4.15 shows the tendency of BIC values with respect to the number of GMM components presented by drivers with different driving styles. It indicates that BIC is mainly convergent at $M = 9$. Thus, we set $M = 9$ in for LCBN-GMM.

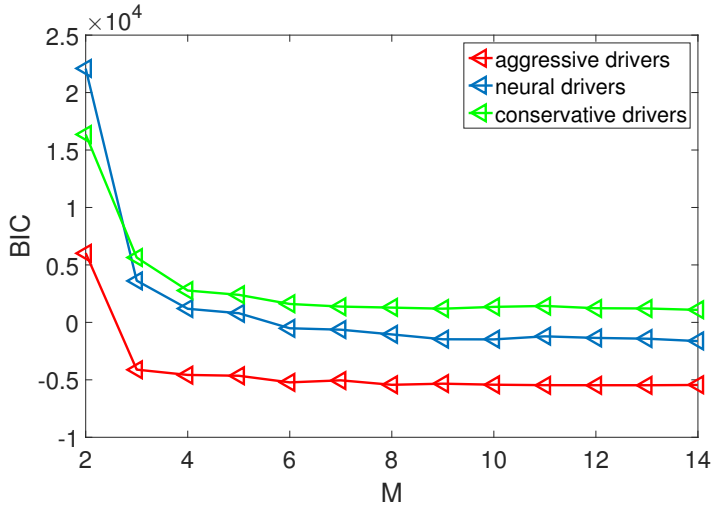


Figure 4.15: BIC values of three driving styles with respect to parameter M.

4.5.2 Other machine learning models

Besides BN, SVM and naive Byes (NB) are also popular machine learning models for classification. For instance, SVM is used for driver fatigue prediction (Senaratne et al., 2007; Camlica, Hilal, and Kulić, 2016), stress event prediction (Rigas, Goletsis, and Fotiadis, 2011) and naive Bayes classifier for driver LC intent prediction (Lethaus, Baumann, et al., 2013). Thus, these two models are also implemented in this chapter for further evaluation.

Unlike BN, which is very complex to model the network and learn the parameters, the implementation of SVM and NB is simpler. The non-linear kernel function used in SVM to generate decision boundary is a Gaussian kernel recommended by Ben-Hur and Weston (2010) and Wang et al. (2017). And we also map the prior distribution of NB in Gaussian distribution for simplicity. Both of them can be implemented using the

Statistics and Machine Learning Toolbox[8] by Matlab. All the models are trained by exactly the same training datasets.

4.5.3 Model training and evaluation method

All the data were firstly grouped into three categories according to driving styles (high aggressive, medium aggressive, and low aggressive) and again separated them by three pre-defined driving scenarios illustrated in Figure 4.1. Thus, training datasets are organized by driving styles and driving scenarios. In addition, the set of all the labeled data samples that are simply grouped together without doing any separation is termed as *non-categorized group*. Totally, the number of labeled data samples by the GBL method for each LC scenario and driving style are listed in Table 4.1.

Table 4.1: The number of labeled data samples using the GBL method.

Aggressiveness	**Type**	**Scenario lead only**	**Scenario lead + adjacent behind**	**Scenario lead + 2 adjacent**
High	LC	1280	2620	2502
	LK	1280	2620	2502
Medium	LC	2271	4789	4431
	LK	2271	4789	4431
Low	LC	1245	2214	2145
	LK	1245	2214	2145
Non-categorized	LC	4796	9623	9078
	LK	4796	9623	9078

In order to guarantee that the training data and testing data are disjoint, a cross-validation (CV) method is used to evaluate the models. Each labeled dataset is randomly and evenly divided into ten folds. Nine folds are used for training and the rest one fold is used for testing. Totally, such procedure is conducted ten times.

Receiver operating characteristic (ROC) curve (McCall et al., 2007; Morris, Doshi, and Trivedi, 2011) is used to access the performance of the models. It is already explained in section 3.3.1. We plot ROC curves for each model using the function provided by Matlab[9], by which given true class labels of the testing samples and the predicted confidence, true negatives, false positives, and false negatives as well as AUC values can be calculated. Classification performance can be evaluated based on AUC values (Huang and Ling, 2005; Vickers and Elkin, 2006).

[8]https://www.mathworks.com/products/statistics.html (visited on 31.07.2019)
[9]https://www.mathworks.com/help/stats/perfcurve.html (visited on 31.07.2019)

4.6 Result and analysis

Statistics of the labels by GBL method

Firstly we make a statistical analysis to see if $t_{prepare}$ is varying by driving styles as well as by driving scenarios. The statistics of $t_{prepare}$ (duration between $t_{prepare}$ and t_0) for the three driving styles in the different scenarios depicted in box plot is shown in Figure 4.16. The mean and standard deviation (SD) are listed in Table 4.2. The following results can be found:

- From the perspective of driving style: on average, $t_{prepare}$ of high aggressive driver is smaller than less aggressive driver (medium and low). This result means that high aggressive driver takes shorter time to prepare for a LC.
- Comparison of different LC scenarios: it indicates that the mean of $t_{prepare}$ in *Scenario lead only* is smaller than in other two scenarios. In addition, the mean of $t_{prepare}$ increases as the scenario becomes more complex (complexity: *Scenario lead only* < *Scenario lead + adjacent behind* < *Scenario lead +2 adjacent*). The results are also consistent with what is concluded in Beggiato et al. (2018) that the more vehicles on the target lane, the more and longer mirror glances.

Thus, it can be concluded that driving style and contextual traffic do impact on labeling results, which is coincide with the reference by Rehder et al. (2016).

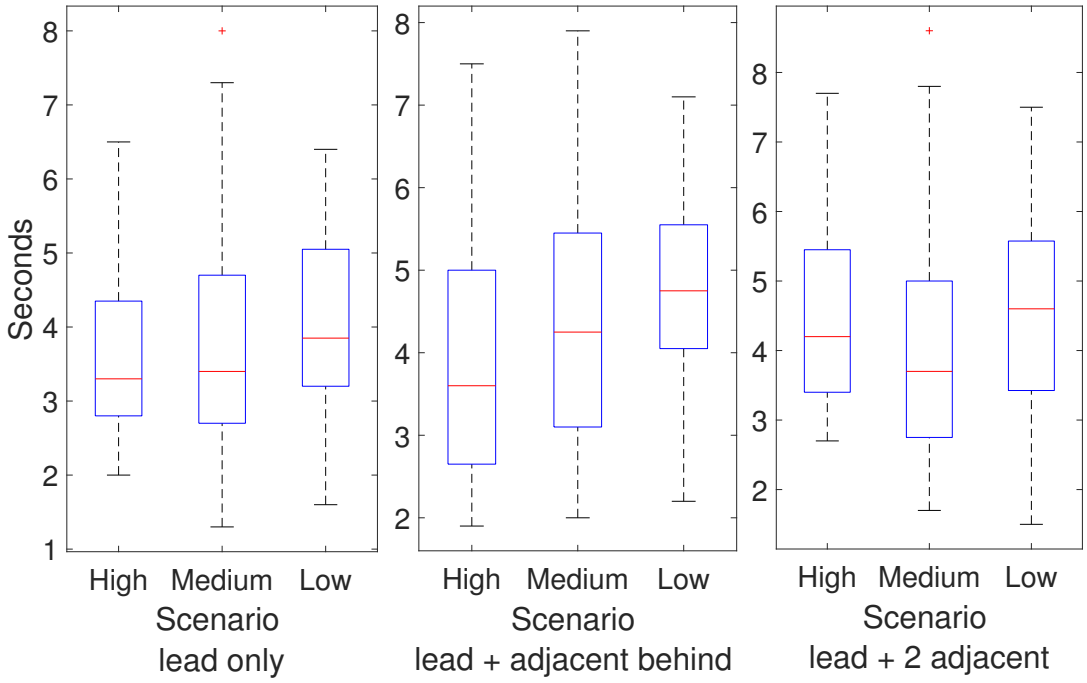


Figure 4.16: The box plots of the labeled moment $t_{prepare}$ before t_0 for different scenarios and different levels of aggressive driving styles.

Table 4.2: The mean and SD (in second) of the labeled $t_{prepare}$ before moment t_0 for different scenarios and driving styles.

Aggressiveness	Scenario lead only	Scenario lead + adjacent behind	Scenario lead + 2 adjacent
High	3.53 (SD = 1.06)	3.88 (SD = 1.35)	4.06 (SD = 1.60)
Medium	3.80 (SD = 1.51)	4.31 (SD = 1.53)	4.57 (SD = 1.44)
Low	3.98 (SD = 1.45)	4.89 (SD = 1.22)	4.45 (SD = 1.30)
Mean	3.76 (SD = 1.38)	4.31 (SD = 1.46)	4.40 (SD = 1.46)

4.6.1 Comparison between different models

Table 4.3: The AUC values performed by different models with GBL.

Scenario	Aggressiveness	LCBN-GMM	NB	SVM
lead only	High	**0.92** (↑)	0.90 (↓)	0.88 (↓)
	Medium	0.94 (↑)	**0.95** (↑)	0.91 (↑)
	Low	**0.93** (↑)	0.90 (↓)	0.91 (↑)
	Mean	**0.93** (↑)	0.92 (↑)	0.90
	Non-categorized	0.89	0.91	0.90
lead + adjacent behind	High	0.91 (↑)	0.92 (↑)	**0.93** (↑)
	Medium	**0.90** (↑)	0.88 (↑)	0.90 (↑)
	Low	**0.93** (↑)	0.83 (↑)	0.84 (↓)
	Mean	**0.91** (↑)	0.87 (↑)	0.89 (↑)
	Non-categorized	0.86	0.85	0.87
lead + 2 adjacent	High	**0.89** (↑)	0.83 (↑)	0.82 (↑)
	Medium	**0.86** (↑)	0.63 (↓)	0.67 (↓)
	Low	**0.88** (↑)	0.72 (↑)	0.82 (↑)
	Mean	**0.87** (↑)	0.73 (↑)	0.77 (↑)
	Non-categorized	0.74	0.71	0.70

In this section, model comparison is made between LCBN-GMM, SVM and NB using the GBL method. AUC values of the classification (LC and LK samples) performed by different models and using different training datasets are listed in Table 4.3. The corresponding ROC curves can be found in Appendix Figure A.2 – Figure A.4. Considering each LC scenario alone, the highest AUC value achieved by certain model (horizontal comparison) is marked as bold. We can see that on average (see the mean), LCBN-GMM performs best. And it almost achieves the best performance with

each driving style training dataset and in each LC scenario. The only exception are in *Scenario lead only* and *Scenario lead + adjacent behind*, where Naive Bayes with medium aggressive driving style datasets and SVM with high aggressive driving style datasets perform slightly better than LCBN-GMM.

In conclusion, considering the average AUC scores, LCBN-GMM outperforms SVM and NB.

4.6.2 Comparison between training datasets

In this section, comparison is made between different training datasets, i.e. the high aggressive, medium aggressive, low aggressive as well as the non-categorized dataset. We compare AUC values between the categorized datasets and non-categorized dataset with an up arrow '↑' for performance improvement and a down arrow '↓' for performance deterioration (vertical comparison).

From Table 4.3 it can be seen that using the non-categorized datasets, the three models achieve similar classification results in each scenario, e.g. from 0.89 to 0.91 in *Scenario lead only*, from 0.85 to 0.87 in *Scenario lead + adjacent behind*, and from 0.70 to 0.74 in *Scenario lead + 2 adjacent*.

The average performance can be improved if the model is trained by the categorized datasets in comparison to the non-categorized datasets. The only exception is SVM in *Scenario lead only*, where the average performance is the same as using the non-categorized dataset. The classification results in different scenarios will be discussed separately as follows.

- In *Scenario lead only*: using the categorized datasets the performance can be slightly improved from 0.89 to 0.93 for LCBN-GMM and from 0.91 to 0.92 for Naive Bayes. SVM is the exception.
- In *Scenario lead + adjacent behind*: using the categorized datasets rather than the non-categorized datasets, model performance can be improved for all methods. More specifically, model performances of LCBN-GMM, Naive Bayes and SVM are improved from 0.86 to 0.91, from 0.85 to 0.87, and from 0.87 to 0.89, respectively. But for the low aggressive driving style dataset, the performance of SVM slightly decreases compared with using the non-categorized dataset.
- In *Scenario lead +2 adjacent*: we find that compared with using the non-categorized datasets, model performance can be improved by using the categorized datasets for all methods. For instance, AUC value increases from 0.74 to 0.87 for LCBN-GMM, from 0.71 to 0.73 for Naive Bayes, and from 0.70 to 0.77 for SVM. It also indicates that for Naive Bayes and SVM, the performance of using the medium aggressive driving style datasets does not have significant improvement.

It can be concluded that model performance can be improved by using the categorized training datasets compared with using the non-categorized datasets.

4.6.3 Comparison between the labeling methods

Table 4.4: The mean and SD of the AUC values performed by LCBN-GMM with different labeling methods.

Label type	Scenario lead only	Scenario lead + adjacent behind	Scenario lead + 2 adjacent
GBL	0.93 (SD = 0.008)	0.91 (SD = 0.012)	0.87 (SD = 0.012)
TWL 5 s	0.85 (SD = 0.049)	0.82 (SD = 0.057)	0.72 (SD = 0.054)
TWL 4 s	0.90 (SD = 0.059)	0.85 (SD = 0.054)	0.80 (SD = 0.043)
TWL 3 s	0.92 (SD = 0.070)	0.87 (SD = 0.054)	0.78 (SD = 0.063)
TWL 2 s	0.94 (SD = 0.057)	0.91 (SD = 0.051)	0.82 (SD = 0.066)
TWL 1 s	0.95 (SD = 0.040)	0.96 (SD = 0.004)	0.87 (SD = 0.052)

In this section, comparison is made between the GBL and the TWL method performed by LCBN-GMM. Table 4.4 summarizes the mean and standard deviation (SD) of AUC values. The full AUC values of TWL can be seen in Table A.1 in Appendix.

It indicates that only by choosing time-windows that are less than 2 s in *Scenario lead only* and time-window of 1 s in *Scenario lead + adjacent behind*, the TWL method could performs slightly better than the GBL method. In *Scenario lead +2 adjacent*, the GBL method represents a smaller SD of AUC values compared with TWL method. For instance in *Scenario lead only*, SD of AUC values for the GBL method is 0.008, which is much smaller than that of the TWL method. From this perspective, the GBL method achieves more stable performance than the TWL method, except for *Scenario lead + adjacent behind* when TWL chooses 1 s as the time-window.

In practice, the TWL method works like this: firstly choosing different time-windows and selecting the best time-window size after evaluation like what we listed in Table 4.4. However, this process is inefficient. The selected time-window size is very scenario specific, which means it does not cover general situation. In comparison, the GBL method takes advantages of driver gaze behavior, it can find $t_{prepare}$ in each LC events with unified labeling criteria, and thus it can make sure that the datasets are labeled in high quality. In conclusion, considering efficiency the GBL method outperforms the TWL method.

4.6.4 Real-time lane-change behavior prediction

In order to further evaluate the GBL and the TWL method, a real-time LC behavior prediction test is made by feeding time series driving data to the well-trained model. Since LCBN-GMM represents good result in model comparison, we use this model to perform the test. Prediction precision and predicted time in advance of LC behavior are

as metrics to evaluate these two labeling method. For the TWL method, time-window size is chosen as 2 s time-window in *Scenario lead only*, and 1 s time-window in *Scenario lead + adjacent behind* and *Scenario lead +2 adjacent*. The reason of choosing such time-windows is that in Table 4.4 these time-windows are with highest AUC values in each driving scenario.

Data used for the real-time test are collected from the experiment described in section 4.3, including totally 304 LC cases. During test, if the output LC probability of LCBN-GMM is greater than a predefined threshold, then a report of LC intent will be given, otherwise not. The rule of determining whether it correctly predicts driver LC behavior is defined as: if the model predicts a LC at certain moment and in the following 8 seconds the driver dose execute a LC maneuver, then this prediction is regarded as correct, otherwise incorrect. Here a threshold of 8 s is selected since the largest time interval between $t_{prepare}$ and t_0 we labeled is close to 8 s. The statistics of $t_{prepare}$ can be seen in Figure 4.16.

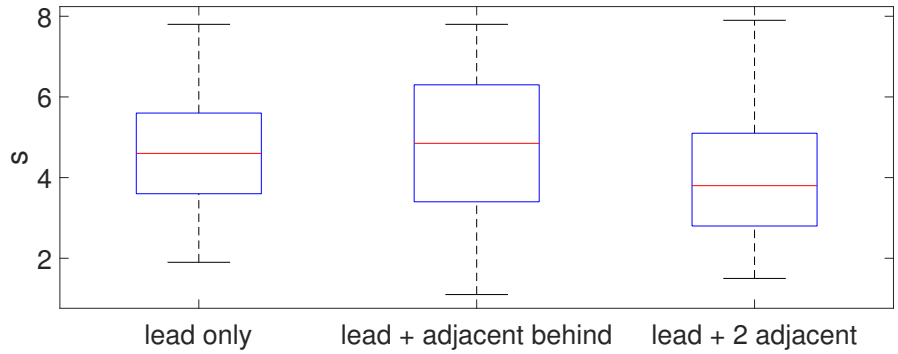
Table 4.5: The real-time prediction result performed by LCBN-GMM using GBL and TWL method.

Type	**Label method**	**Scenario lead only**	**Scenario lead + adjacent behind**	**Scenario lead + 2 adjacent**	**Mean**
Precision	GBL	56 (46) 82.1%	124 (89) 71.7%	124 (102) 82.2%	78.2%
	TWL	56 (50) 89.2%	124 (89) 71.7%	124 (45) 36.2%	60.7%
Prediction time (s)	GBL	4.5 (SD = 1.5)	4.8 (SD = 1.7)	4.1 (SD = 1.4)	4.5
	TWL	4.7 (SD = 1.9)	4.5 (SD = 1.7)	4.0 (SD = 2.4)	4.4

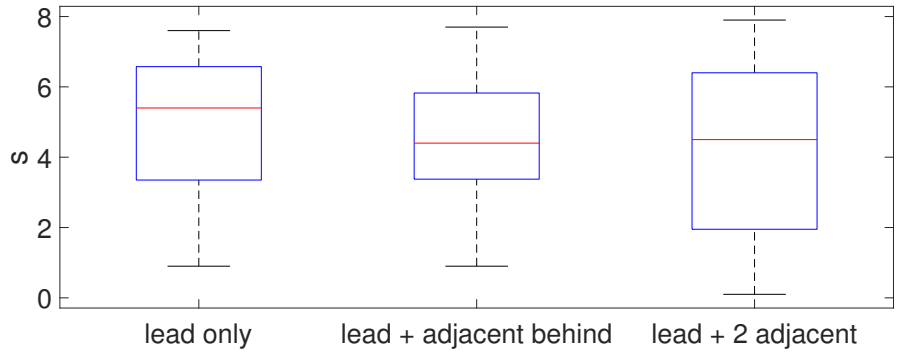
The final prediction results are listed in Table 4.5, and the box plot can be seen in Figure 4.17. Table 4.5 suggests that only in *Scenario lead only*, the TWL slightly outperforms the GBL method with earlier prediction time and higher precision. In *Scenario lead + adjacent behind* and *Scenario lead +2 adjacent*, however, the GBL method performs much better than the TWL with both earlier prediction time and higher precision. Especially in *Scenario lead +2 adjacent*, the TWL method gets poor precision of only 36.2%. However, the GBL method can achieve 82.2%. In addition, considering the average performance, the GBL method is still with higher accuracy, earlier prediction time as well as smaller SD.

In conclusion, the GBL method outperforms the TWL method with higher precision and earlier predicted time.

Figure 4.18 gives an example of how LCBN-GMM predicts driver LC behavior in real-time case. The blue line represents the lateral position of the subject vehicle. We can see that the driver changes to the left lane at around 30 s and steers back to the



(a) The GBL method



(b) The TWL method

Figure 4.17: The box plot of correctly predicted LC before t_0.

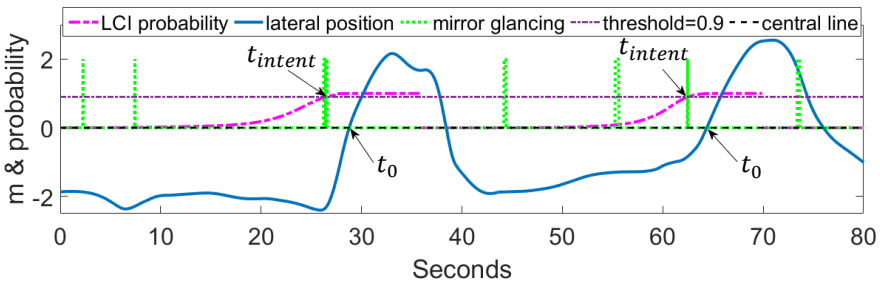


Figure 4.18: An example of the real-time prediction performance of LCBN-GMM.

right lane at around 40 s. The pink line is the posterior probability of LC behavior by LCBN-GMM. The probability threshold of giving a LC report is set as 0.9. That is to say if the probability is greater than 0.9, then LCBN-GMM would report a LC prediction report. The green dotted line represents driver mirror glancing behavior, which is a binary signal with value of 2 for observing a glancing and of 0 for no glancing behavior. Because the signal of driver mirror glancing behavior is only used for data labeling rather than an input of LCBN-GMM, the normal glancing signals do not impact on LC prediction probability. It can be seen that even there are two mirror glancing behaviors observed between 0 and 10 s, LCBN-GMM does not report LC. The driver executes the first LC maneuver at around 30 s and LCBN-GMM can predict it several seconds ahead of the actual LC maneuver. LCBN-GMM is modeled only for left LC case, however, for right LC case the output probability of LCBN-GMM is stay in 0.

4.7 Summary

This chapter proposes a framework of prediction of driver LC behavior including works like the design of driving simulator-based experiment, feature extraction, modeling machine learning models, training datasets preparation, model selection and evaluation. Inspired by the prior research, several improvements have been made e.g. considering driving contextual traffic and driving styles in preparing training datasets, proposing a gaze-based labeling method (GBL) to obtain high quality training datasets.

In conclusion, comparison of different ML models, result indicates that LCBN-GMM performs better than SVM and NB. Comparing different training datasets, the performance of the models trained by categorized datasets outperform by non-categorized training datasets. This applies to all the models in each driving scenario. In addition, comparison of the GBL and the TWL method, the GBL outperforms the TWL with both higher precision and earlier predicted time. Finally, using the GBL method LCBN-GMM achieves 78% prediction precision and could predict driver LC behavior nearly 4.5 seconds ahead of an actual LC maneuver.

The limitation of the study is that the experiment was conducted in a driving simulator, which means the experiment condition is tool idealized compared with the real-road traffic and thus the participants tend to be absent of real risk while driving (Schmitt et al., 2018). In addition, during the experiment the participants were given oral instruction before doing driving task. This may impact on the driving habits of the participants. For example, before the experiment, participants were suggested to use the turn signal before taking a left LC, however, the case of no turn signal usage during LC course might happen in real situation. Furthermore, the method used for classification of driving styles is limited by the questionnaire. The driving aggressiveness score tends to describe the driver's driving style globally, i.e. from personality trait level, however, two drivers who score similarly may show different temporary driving

styles which can be represented by different speed choice as well as accelerating or braking behavior etc.. This limitation also happens to all the methods that attempt to classify the driver's global driving style (Sagberg et al., 2015). Thus, seeking to consider individual driving style in preparing for datasets would be one solution. Additionally, there is no systematic feature selection work before model training. That is to say, research should be done to test whether the selected features are really suitable or not. Most of the feature selection work by previous research is based on the empirical knowledge. Therefore, a comprehensive study of feature selection on driver LC behavior is needed.

5

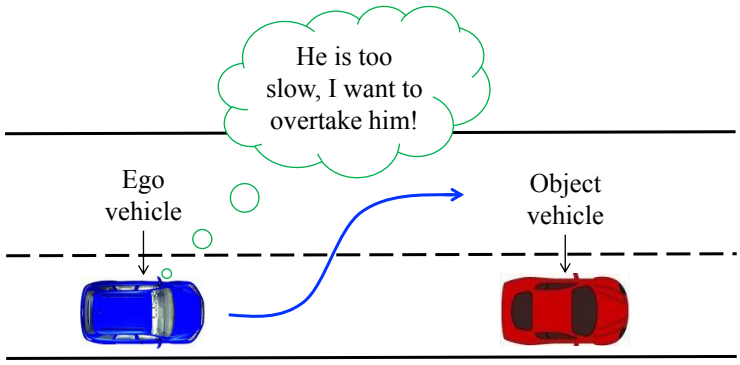
Big data analysis - Evaluation of feature selection for driver lane-change behavior

A substantial portion of this chapter is based on the above paper.

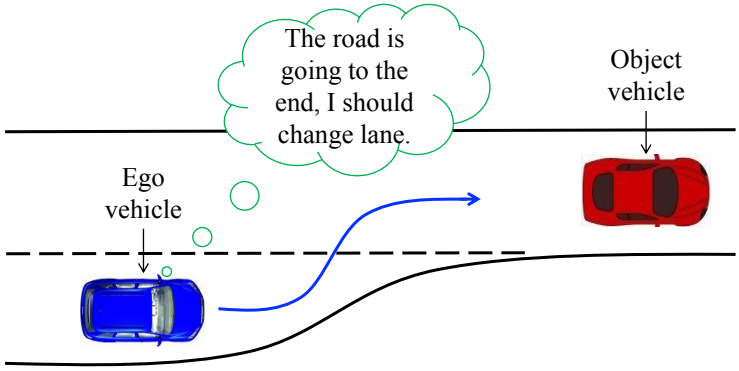
5.1 Introduction

At the end of chapter 4, it proposes the underlying problem of extracting features based on the empirical knowledge. In addition, another aspect that can be improved in the whole framework of driver LC prediction is using naturalistic driving data instead of using data collected from the driving simulator. In this chapter, we tend to make improvement in these two aspects.

Actually, the importance of feature selection has been emphasized on for a long time in the field of machine learning (ML). For instance, Menze et al. (2009) proposed a method of feature selection for classification of spectral data; Kira and Rendell (1992) put forward a practical approach of feature selection. Vafaie and Imam (1994) summarized two feature selection method i.e. genetic algorithms and greedy-like search. However, specifically for driver LC behavior, there are few related works regarding to this topic. Blindly using features without selection may lead to excessive computation time, however, insufficient feature selection may cause poor classification results. Selecting



(a) LC for the purpose of overtaking a slow leading vehicle.



(b) LC due to a lane drop.

Figure 5.1: An example of LC with two different purposes.

high contributive features to classify lane-change (LC) and lane-keep (LK) data samples is necessary for further LC prediction work.

Many kinds of features have been used for driver LC behavior classification. For instance, longitudinal features like time to collision (Sivaraman and Trivedi, 2014; Liebner et al., 2013; Peng et al., 2015), longitudinal acceleration, and lateral features e.g. steering angle (Xu et al., 2012), yaw rate (Sivaraman and Trivedi, 2014; Doshi, Morris, and Trivedi, 2011), and lateral acceleration (Kasper et al., 2012; Boubezoul, Koita, and Daucher, 2009). These features are assumed to be strong enough for LC behavior classification either based on intuition or the empirical knowledge, however, this assumption is still hanging on and yet comprehensively studied.

In general, driver LC behavior can be either discretionary or mandatory. A mandatory lane-change will occur when a driver must leave a lane due to a lane drop or bypass a blockage etc. A discretionary lane change occurs when a driver prefers a more efficient adjacent lane (J2944, 2013), e.g. when passing a slow-moving leading vehicle to maintain the current speed (Lee, Olsen, Wierwille, et al., 2004). This means that in discretionary and mandatory LC cases, the importance of the features is different as well. In other words, certain feature is useful to discretionary LC does not mean it is also useful to mandatory LC. Take the feature TTC for instance. Thought it is assumed to be very important for modeling LC decision-making process (Lee, Olsen, and Wierwille, 2003;

Nilsson and Sjöberg, 2013), the importance of TTC in the two different scenarios is different. As we can see in Figure 5.1, TTC can be an important feature for overtaking purpose (e.g. Figure 5.1a), however, it maybe not that important to mandatory LC (e.g. Figure 5.1b). Leonhardt and Wanielik (2017) evaluated the effects of various features in different LC driving scenarios. The result indicated that even for the same feature, the weight of overtaking a slow vehicle and merging is different. Thus, feature selection process should also take LC scenario into account.

To this end, in this chapter we investigate the contribution of each extracted feature from the perspective of statistics based on naturalistic driving data. The aim is to comprehensively figure out the importance of different types of features regarding to driver LC behavior and select the most contributive features that can be used for model training. The main improvements in this study compared with the prior research can be summarized as follows:

1. Present a feature selection method from the perspective of statistics to investigate the statistical significance of each feature that is related to driver LC behavior.
2. Not only time-domain features but also frequency-domain features are considered to fill in gaps in the previous works. This can largely enrich the candidate feature sets.
3. In feature extraction procedure, different driving scenarios are taken into account to comprehensively evaluate the extracted features.

5.2 Related work of feature selection

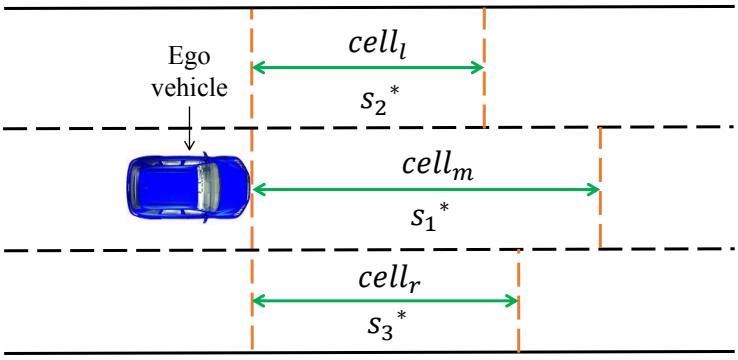
The goal of feature selection is to reduce the dimension of the feature sets by removing unimportant features. In general, feature selection method can be grouped into filter method and wrapper method. Filter method analyzes the intrinsic properties of data, ranking and selecting features without involving learning algorithms. In comparison, wrapper method involves learning algorithms. It would give scores to a given subset of features based on certain algorithm (Guyon et al., 2008; Geng et al., 2007). For wrapper method, the ranking of features can be varying from learning algorithm to algorithm. This study is aim at finding the intrinsic properties of the features and selecting the most contributive features among all features rather than ranking and selecting features for a specific ML algorithm. Therefore, the feature selection method in this chapter belongs to wrapper method.

For LC behavior prediction, the data collected from sensors are in the form of time series, so the properties of the features in time-domain are the most frequently extracted (Xu et al., 2012; Kasper et al., 2012; Liebner et al., 2013; Sivaraman and Trivedi, 2014; Peng et al., 2015; Doshi, Morris, and Trivedi, 2011; Boubezoul, Koita, and Daucher, 2009). On the other hand, frequency-domain features have already been

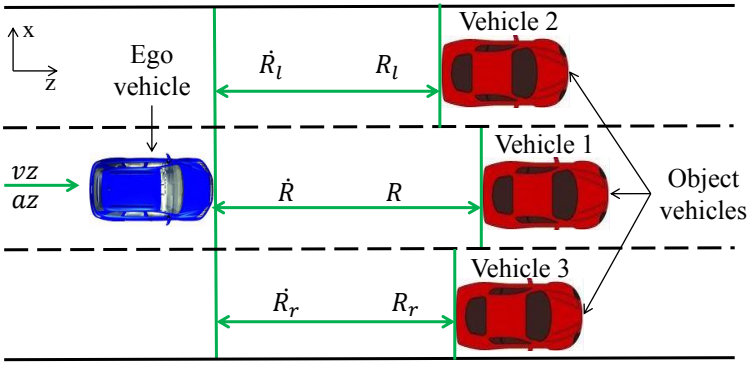
used to recognize driver state. For instance, the power spectrum features via wavelet transform are selected for Belief networks (Hajinoroozi et al., 2015; Chen et al., 2015). In other areas regarding to recognition such as speech recognition (Thomas, Ganapathy, and Hermansky, 2008) and anomaly detection (Zhang et al., 2008), frequency-domain features play important roles.

In this study we consider the properties of the features in both time-domain and frequency-domain.

5.3 Lane-change scenario modeling



(a) Illustration of LC occupancy schedule grid.



(b) The relationship between the ego vehicle and object vehicles.

Figure 5.2: Illustration of the parameters used for calculating the length of the cell-grid.

To easily describe the relationship between the ego vehicle and its surroundings vehicles, we adopt the method presented in section 4.2.1, i.e cell-grid based modeling method. Based on Do et al. (2017), totally 9 cells and 32 situations (2^5) are considered in the driving contextual traffic. But the paper does not give the specific boundary of the cells. Kasper et al. (2012) modeled the cell by considering the speed-dependent information when a cell will be occupied or will become free. But they assumed that the vehicle can move unobstructed towards certain cell, which cannot be satisfied in the situation where the ego vehicle is being overtaken by other vehicles. In this chapter

we aim to detail the cell-grid method by considering the dynamic relationship between the ego vehicle and the surrounding vehicles. A 3-cell grid is employed to model the contextual traffic for both left and right LC case with totally 8 scenarios.

The experimental vehicle, from which the data used in this study were collected, has no back view sensors installed, so only the front traffic can be detected. The traffic situation on the back of the ego vehicle is not considered. Despite such limitation, the method of modeling contextual traffic in this chapter can be extended to more cell-grids which can cover the traffic on back of the ego vehicle as well. But it is not within the scope of this study. Figure 5.2a depicts the modeled cell-grid.

We adopt the theory presented by Karim et al. (2013) to define the middle cell (cell_m) and the theory by Kesting and Treiber (2013) to define the left (cell_l)/right cell (cell_r). The dynamic length of each cell is s_1^*, s_2^*, s_3^*, respectively, as it is shown in Figure 5.2a. The length of cell_m is defined by a Mean Safe Time Gap (MSTG) based on Karim et al. (2013) as:

$$\text{MSTG} = BT_{EV} - BT_{OV} + RT \tag{5.1}$$

where BT_{EV} and BT_{OV} are the brake time of the ego vehicle and object vehicle 1, respectively. RT is the driver's perception-reaction time. And for certain vehicle, the BT is calculated by an empirical equation

$$BT = 0.02321 \cdot vz - 0.08785, \tag{5.2}$$

where vz is the vehicle speed and thus

$$BT_{EV} - BT_{OV} = 0.02321 \cdot \dot{R} \tag{5.3}$$

where, $\dot{R}$ is the range rate between the ego vehicle and object vehicle 1. So, the dynamic length of s_1^* can be written as

$$s_1^* = v \cdot \text{MSTG}, \tag{5.4}$$

where v is the longitudinal speed of the ego vehicle.

We define cell_l and cell_r based on the Intelligent Driver Model (IDM) (Kesting and Treiber, 2013). In the study, the safe distance is derived from the leading vehicle, driving at a desired speed, or preferring accelerations to be within a comfortable range. Additionally, kinematical aspects are taken into account, such as the quadratic relation between braking distance and speed. Firstly, on the left and right lane, a term *desired distance* on the left (s_l^*) and right (s_r^*) lane are defined respectively as:

$$s_l^* = s_0 + \max(0, v \cdot T + \frac{\dot{R}_l \cdot R_l}{2 \cdot \sqrt{a^* \cdot b^*}}) \tag{5.5}$$

$$s_r^* = s_0 + \max(0, v \cdot T + \frac{\dot{R}_r \cdot R_r}{2 \cdot \sqrt{a^* \cdot b^*}}), \tag{5.6}$$

where s_0 is the minimum (bumper-to-bumper) gap, T is the safe time gap, a^* and b^* are acceleration and comfortable deceleration. R_l, $\dot{R}_l$ and R_r, $\dot{R}_r$ are the range and range rate the ego vehicle with object vehicle 2 and vehicle 3 in Figure 5.2b, respectively. The dynamic term $\dot{R}_l \cdot R_l/(2 \cdot \sqrt{a^* \cdot b^*})$ and $\dot{R}_r \cdot R_r/(2 \cdot \sqrt{a^* \cdot b^*})$ imply the intelligent braking strategy for left lane-change(LLC) and right lane-change (RLC) cases.

Secondly, based on the *desired distance* on the left (s_l^*) and right (s_r^*) lane , the dynamic safety distance, namely the length of s_2^* and s_3^* can be written as:

$$s_2^* = \frac{s_l^*}{\sqrt{(\frac{s_l^*}{R_l})^2 - \frac{\Delta a + a_{bias}}{az}}} \tag{5.7}$$

$$s_3^* = \frac{s_r^*}{\sqrt{(\frac{s_r^*}{R_r})^2 - \frac{\Delta a - a_{bias}}{az}}}, \tag{5.8}$$

where az is the longitudinal acceleration of the ego vehicle. Δa is the LC threshold. a_{bias} represents the asymmetric property of LLC and RLC.

Table 5.1: The reference values regarding to the parameters for the cell grid

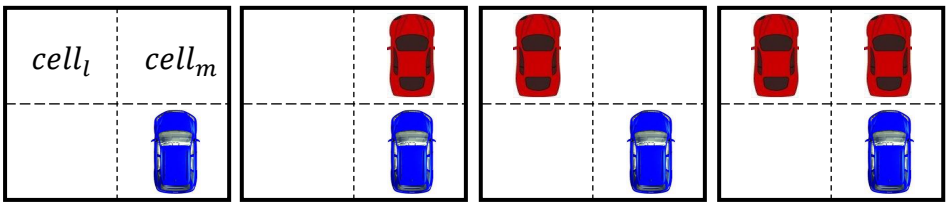
Parameter	**Value**
RT	1.9 s
T	1.0 s
s_0	2 m
a^*	1.0 m/s^2
b^*	1.5 m/s^2
Δa	0.1 m/s^2
a_{bias}	0.3 m/s^2

All the values of the parameters in equation (5.1) and equation (5.5) – equation (5.8) are listed in Table 5.1, and the occupancy states of cells can be given by

$$\begin{cases} \text{cell}_\text{m} = 0 & \text{if } R \geq s_1^* \\ \text{cell}_\text{m} = 1 & \text{if } R < s_1^* \end{cases} \tag{5.9}$$

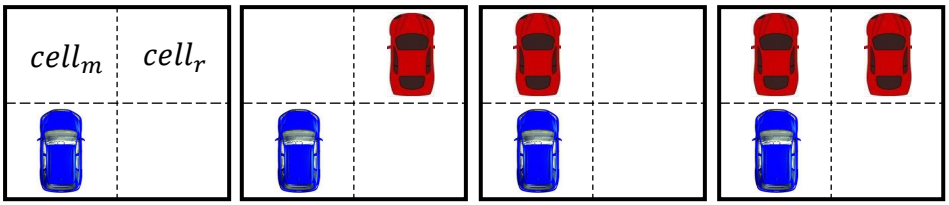
$$\begin{cases} \text{cell}_\text{l} = 0 & \text{if } R_l \geq s_2^* \\ \text{cell}_\text{l} = 1 & \text{if } R_l < s_2^* \end{cases} \tag{5.10}$$

$$\begin{cases} \text{cell}_\text{r} = 0 & \text{if } R_r \geq s_3^* \\ \text{cell}_\text{r} = 1 & \text{if } R_r < s_3^* \end{cases} \tag{5.11}$$



LLC Scenario 0_0 LLC Scenario 0_1 LLC Scenario 1_0 LLC Scenario 1_1

(a) Left lane-change scenarios.



RLC Scenario 0_0 RLC Scenario 0_1 RLC Scenario 1_0 RLC Scenario 1_1

(b) Right lane-change scenarios.

Figure 5.3: Illustration of the modeled LC scenarios using the cell-grid method.

Depend on the occupancy states of cell-grids, totally 8 scenarios (4 scenarios for each LLC and RLC) can be generated, as it is depicted in Figure 5.3:

- *LLC Scenario 0_0*: When the ego vehicle makes LLC, there are no object vehicles on both cell_m and cell_l.
- *LLC Scenario 0_1*: When the ego vehicle makes LLC, there is no object vehicle on cell_l but cell_m is occupied.
- *LLC Scenario 1_0*: When the ego vehicle makes LLC, there is no object vehicle on cell_m but cell_l is occupied.
- *LLC Scenario 1_1*: When the ego vehicle makes LLC, both cell_m and cell_l are occupied.
- *RLC Scenario 0_0*: When the ego vehicle makes RLC, there are no object vehicles on both cell_m and cell_r.
- *RLC Scenario 0_1*: When the ego vehicle makes LLC, there is no object vehicle on cell_m but cell_r is occupied.
- *RLC Scenario 1_0*: When the ego vehicle makes LLC, there is no object vehicle on cell_r but cell_m is occupied.

- *RLC Scenario 1_1*: When the ego vehicle makes LLC, both cell_m and cell_r are occupied.

The name of the LC scenarios termed *Scenario 0_1* and *Scenario 1_0* is in accordance with the binary states of the occupancy cells illustrated in Figure 5.3.

5.4 Data processing and feature extraction

5.4.1 Naturalistic driving data

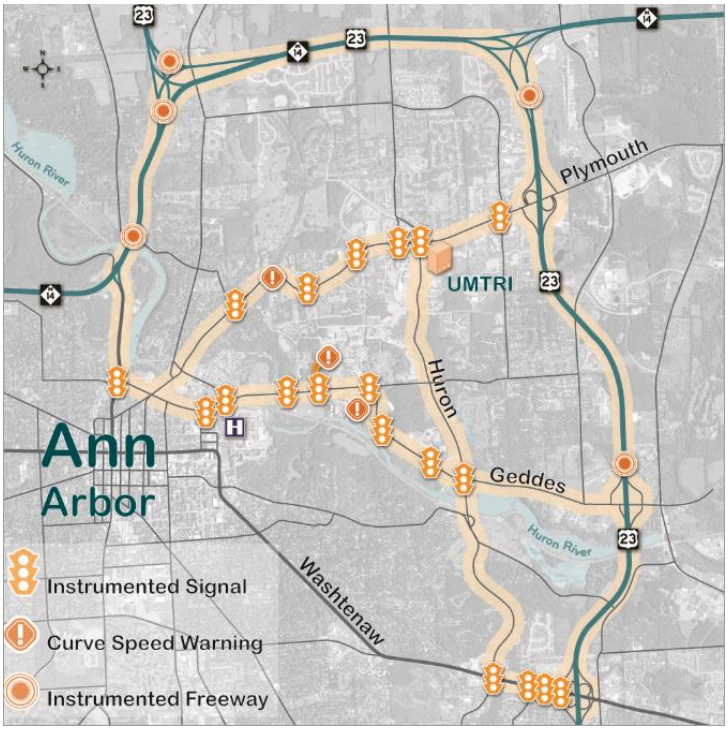


Figure 5.4: The experimental route of the SPMD project. Picture extracted from Bezzina and Sayer (2014).

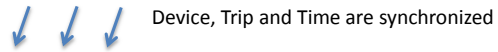
The naturalistic driving data that are used in this study is based on the project of Safety Pilot Model Deployment (SPMD), which is a comprehensive data collection project in real-road condition. This third party datasets is extremely useful for researchers who cannot conduct their own real-road experiment. It is completely open sourced which is popular in many field of research.

The real-road project includes multi-modal traffic, hosting approximately 3,000 vehicles equipped with vehicle-to-vehicle (V2V) communication devices. The datasets we used were collected from 20 vehicles, driving in the real-road including 75 miles of roadway. The route is shown in Figure 5.4. Roads that marked as yellow are the route SPMD vehicle drove. The drivers voluntarily joined in SPMD project. They drove the SPMD vehicle completely based on their own driving styles with no restriction on their driving behaviors. Each SPMD vehicle was equipped with data acquisition

systems (DAS) e.g. CAN and GPS as well as vision system like Mobileye. All the signals coming from different DAS were time-synchronized with sampling rate at 10 Hz. Datasets are available on-line on the website of U.S. transportation department[10], where *DataFrontTargets*, *DataLane* and *DataWsu* were downloaded for our study. The description of the three datasets can be found in Henclewood, Abramovich, and Yelchuru (2014) as follows:

- *DataFrontTargets*: Log of the data collected by the Mobileye sensor which is a part of the DAS; largely includes data about the (vehicle) object that is in front of the ego vehicle.
- *DataLane*: Logs quality of the lane markings next to the ego vehicle as well as the distances between each side of the vehicle and each lane line.
- *DataWsu*: Log of GPS and CAN Bus data obtained via the onboard Wsu.

The description of the tree types of datasets regarding to the features of this study is illustrated in Table 5.2, 5.3 and 5.4, respectively. The full details of the datasets can be found in Henclewood, Abramovich, and Yelchuru (2014).



Device	Trip	Time	LaneDistanceLeft	LaneDistanceRight	TargetId	ObstacleId	CIPV	Range	Rangerate	Transversal	Status	YawRateWsu	SpeedWsu	AxWsu
10116	671	420	-2.65625	1.0234375	1	14	1	50.458332	-1.5624962	-0.99305558	3	-0.88790071	106.9948	0.22062109
10116	671	430	-2.5976563	1.0546875	1	14	1	50.333332	-1.5416679	-0.7638889	3	-0.88790071	106.9948	0.22062109
10116	671	440	-2.5585938	1.0703125	1	14	1	50.180557	-1.7222214	-0.52777779	3	-0.88790071	107.0742	0.22062109
10116	671	450	-2.5078125	1.09375	1	14	1	49.986111	-1.7013893	-0.25694445	3	-0.69093382	107.1797	0.23589151
10116	671	460	-2.4726563	1.1054688	1	14	1	49.770832	-1.5208397	0.0069444...	3	-0.69093382	107.2647	0.23589151
10116	671	470	-2.4453125	1.109375	1	14	1	49.6875	-0.99305...	0.27083334	3	-0.59378427	107.3167	0.2247024
10116	671	480	-2.4257813	1.1054688	1	14	1	49.569443	-0.43054...	0.5625	3	-0.59378427	107.3975	0.2247024
10116	671	490	-2.40625	1.1015625	1	14	1	49.590279	-0.43749...	0.84027779	3	-0.62835371	107.5053	0.2340969
10116	671	500	-2.375	1.109375	1	14	1	49.604168	-0.58333...	1.1180556	3	-0.62835371	107.5896	0.2340969
10116	671	510	-2.3554688	1.109375	1	14	1	49.451389	-0.99306...	1.3888888	3	-0.46605161	107.5831	0.1968137
10116	671	520	-2.3320313	1.109375	1	14	1	49.347221	-1.0625038	1.6666666	3	-0.3368642	107.7995	0.2216838
10116	671	530	-2.3007813	1.1210938	1	14	1	49.222221	-0.64583...	1.9652778	3	-0.3368642	107.7995	0.2216838
10116	671	540	-2.2773438	1.1210938	1	14	1	49.1875	-0.41666...	2.2291667	3	-0.3368642	107.8793	0.2216838
10116	671	550	-2.3125	1.125	1	14	1	49.208332	-0.30555...	2.5	3	-0.42342499	107.8346	0.17296299
10116	671	560	-2.3125	1.125	1	14	1	49.145832	-0.45138...	2.75	3	-0.42342499	107.8969	0.17296299
10116	671	570	-2.3203125	1.125	1	14	1	49.090279	-0.6388855	3.0069444	3	-0.41731	108.1103	0.22848
10116	671	580	-2.3242188	1.1210938	1	14	1	49.020832	-0.37499...	3.2638888	3	-0.41731	108.1926	0.22848
10116	671	590	-2.3242188	1.1210938	1	14	1	48.951389	0.020832...	3.5347223	3	-0.36453179	108.2361	0.2227118
10116	671	600	-2.3242188	1.1210938	1	14	1	49.027779	0.57638931	3.8402777	3	-0.36453179	108.3163	0.2227118
10116	671	610	-2.3320313	1.1132813	1	14	0	49.097221	1.1805534	4.1597223	3	-0.50014162	108.3268	0.191559
10116	671	620	-2.328125	1.109375	1	14	0	49.236111	1.4513855	4.5138888	3	-0.50014162	108.3957	0.191559
10116	671	630	-2.3320313	1.109375	1	14	0	49.4375	1.6944427	4.8958335	3	-0.27709419	108.5802	0.2351979
10116	671	640	-2.3359375	1.1054688	1	14	0	49.583332	1.7222252	5.2986112	3	-0.27709419	108.6649	0.2351979
10116	671	650	-2.3359375	1.09375	1	14	0	49.770832	1.6527824	5.7986112	3	-0.16452	108.6209	0.1883624
10116	671	660	-2.3359375	1.0976563	1	14	0	49.930557	1.7361183	6.2847223	3	-0.16452	108.6887	0.1883624
10116	671	670	-2.3359375	1.1015625	1	14	0	50.090279	1.8750038	6.7986112	3	0.068834111	108.7467	0.1749355
10116	671	680	-2.3320313	1.1054688	1	14	0	50.291668	1.6597214	7.3958335	3	0.068834111	108.8097	0.1749355
10116	671	690	-2.3359375	1.125	1	14	0	50.527779	1.4027786	7.9930553	3	0.45582309	108.8873	0.1786069

Figure 5.5: An example of the jointed datasets of *DataFrontTargets*, *DataLane* and *DataWsu* using MySQL.

Due to the large quantity of the dataset (nearly 20 Gigabit), MySQL 6.3 is used to query the datasets. In addition, although all the sensors installed in the SPMD vehicles were synchronized, for research use all the three separated datasets, i.e. *DataFrontTargets*, *DataLane* and *DataWsu*, should be jointed in one datasets. Figure 5.5 illustrates the jointed datasets by using MySQL, where we can see that the whole datasets are synchronized by device, trip and time.

[10]https://data.transportation.gov/Automobiles/Safety-Pilot-Model-Deployment-Data/a7qq-9vfe (visited on 31.07.2019)

Table 5.2: The description of datasets *DataFrontTargets*

Data element	Units	Description
Device	none	A unique numeric ID assigned to each DAS. This ID also doubles as a vehicle's ID.
Time	$\frac{1}{10}$s	Time in centiseconds since DAS started, which (generally) starts when the ignition is in the on position.
Trip	none	Count of ignition cycles. Each ignition cycle commences when the ignition is in the on position and ends when it is in the off position.
TargetId	none	Numeric ID assigned by the Mobileye sensor to distinguish between the different objects being tracked; the closest obstacle is given a TargetId value of 1.
ObstacleId	none	ID of new obstacle, as assigned by the Mobileye sensor, and its value will be the last used free ID.
Range	m	Longitudinal position of an object, typically the closest object, relative to a reference point on the ego vehicle, according to the Mobileye sensor.
RangeRate	m/s	Longitudinal velocity of an object, typically the closest object, relative to the ego vehicle, according to the Mobileye sensor.
Transversal	m	The lateral position of the obstacle, as determined by the Mobileye sensor.
Status	none	Classification of the motion (kinematic state) of an identified obstacle/target as stopped, moving, etc.
CIPV	none	Field communicating whether an obstacle is the closest in a vehicle's path.

Table 5.3: The description of datasets *DataLane*

Data element	Units	Description
Device	none	A unique numeric ID assigned to each DAS. This ID also doubles as a vehicle's ID.
Time	$\frac{1}{10}$s	Time in centiseconds since DAS started, which (generally) starts when the ignition is in the on position.
Trip	none	Count of ignition cycles. Each ignition cycle commences when the ignition is in the on position and ends when it is in the off position.
LaneDistanceLeft	m	Distance between the left side of the vehicle and the left boundary of the travel lane.
LaneDistanceRight	m	Distance between the right side of the vehicle and the right boundary of the travel lane.

Table 5.4: The description of datasets *DataWsu*

Data element	Units	Description
Device	none	A unique numeric ID assigned to each DAS. This ID also doubles as a vehicle's ID.
Time	$\frac{1}{10}$s	Time in centiseconds since DAS started, which (generally) starts when the ignition is in the on position.
Trip	none	Count of ignition cycles. Each ignition cycle commences when the ignition is in the on position and ends when it is in the off position.
AxWsu	m/s^2	Longitudinal acceleration from vehicle CAN Bus.
YawRateWsu	deg/s^2	Yaw rate from vehicle CAN Bus.
SpeedWsu	km/h	Speed from vehicle CAN Bus.

The method of querying LC events from huge time series data is used by Zhao, Guo, and Jia (2017). Totally, 1375 lane-change cases (761 LLC and 614 RLC) are extracted for analysis. The statistics of the LC cases with respect to the corresponding LC scenarios in Figure 5.3 can be seen in Table 5.5. We can see that in LLC scenarios, most of the cases took place in *LLC Scenario 0_0* (365 cases) and *LLC Scenario 0_1* (354 cases). And in RLC scenarios, the dominating cases are *RLC Scenario 0_0* (371 cases) and *RLC Scenario 1_0* (214 cases). This result implies that when the driver wants to execute left/right LC maneuver, he/she tends to wait until the destination lane being empty (cell_l/ cell_r is unoccupied).

Table 5.5: The otal amount of LC cases

LC type	Scenario	Amount
LLC	0_0	365
	0_1	354
	1_0	15
	1_1	27
RLC	0_0	371
	0_1	10
	1_0	214
	1_1	16

5.4.2 Feature extraction

Vehicle dynamic feature

Vehicle dynamic feature refers to features that can describe the dynamic motion of the ego vehicle. Vehicle yaw rate and lateral acceleration are widely regarded as strong

features of vehicle lateral behavior. Together with longitudinal acceleration, the above signals are necessary for recognizing, predicting as well as modeling vehicle lateral behaviors (Leonhardt and Wanielik, 2017; Higgs and Abbas, 2015; Li, Li, et al., 2015; Luo et al., 2016). In this study we also choose the following features that can be directly collected from the on-board sensors:

- $yawRate_t$: yaw rate of the ego vehicle at time t.
- az_t: longitudinal acceleration of the ego vehicle at time t.
- ax_t: lateral acceleration of the ego vehicle at time t.

Combined feature

Combined feature is the feature combines different types of features. The common combined features are as follows.

Time to collision (TTC) is the time required for two vehicles to collide if they continue at their present speeds on the same path. It is usually used to evaluate the collision risk (Kusano and Gabler, 2011). If the driver follows a vehicle with a small TTC, he/she maybe execute a LC to overtake the slow leading vehicle. Thus the TTC can be regarded as a valuable feature to recognize driver LC behavior (Kasper et al., 2012). Time-to-lane crossing (TLC) represents the time available for a driver until the moment at which any part of the vehicle reaches one of the lane boundaries (Godthelp, Milgram, and Blaauw, 1984). It is an indicator to estimate if the ego vehicle is going to cross the lane. Based on J2944 (2013), TTC and TLC are given by

- TTC with the object vehicle in front on the current lane ($\mathrm{TTC_t}$) at time t:

$$\mathrm{TTC_t} = -\frac{R}{\dot{R}} \tag{5.12}$$

  where R and $\dot{R}$ (in Figure 5.2b) are the range and the range rate between the front edge of the ego vehicle and rear edge of the closest object vehicle in the same traveling path as the ego vehicle, respectively. Here, what needs to be mentioned is that TTC is only calculated for the LC case when $\mathrm{Cell_m} = 1$, because $\mathrm{Cell_m} = 0$ means there is no vehicle in the cell.

- TLC at time t ($\mathrm{TLC_t}$):

$$\mathrm{TLC_t} = \frac{dx}{vx}, \tag{5.13}$$

  where dx is lateral distance between the front wheel and the lane boundary of the ego vehicle. vx is the lateral speed.

In case that $\dot{R}$ and vx are equal to zero, equation (5.12) and (5.13) become infinity, we use the inverse of $\mathrm{TTC_t^{-1}}$ and $\mathrm{TLC_t^{-1}}$ instead.

Time-window feature

Data collected from the on-board sensors are in time series, so using a time-window (TW) to extract features is effective to capture the information during the past few seconds and this information can be used to recognize upcoming events (Thissen et al., 2003; Salfner and Malek, 2007). In the case of predicting driver LC behavior, different lengths of TW between 1 second to 5 seconds are selected for feature extraction (Mandalia and Salvucci, 2005). In order to capture the properties of time series data, statistical variables (mean, standard deviation, maximum, minimum and median) are calculated within each TW (Li, Li, et al., 2015) as is described in Table 5.6, i.e. feature number 6–80. The number of the top right corner of the feature is the length of TW. Thus, '5' in $mean\_yaw_t^5$ means choosing 5 seconds as TW and '4' in $mean\_yaw_t^4$ represents 4 seconds TW, see feature # 6 and # 7 as examples. Figure 5.6a demonstrates how time-window features are extracted.

Frequency-domain feature

Frequency-domain features have already been widely used in anomaly detection area as well as detecting driver mental states (Chen et al., 2015; Chandola, Banerjee, and Kumar, 2009). Fast Fourier transform (FFT) is a popular method to transform time-domain signals into frequency-domain (Heckbert, 1995). After FFT, the maximum value of FFT coefficients within TW is a good indicator to represent the property of frequency signals (Mörchen, 2003). The description of the frequency-domain features are listed in Table 5.6, i.e. feature number 81–95. Figure 5.6b depicts how frequency domain features are extracted.

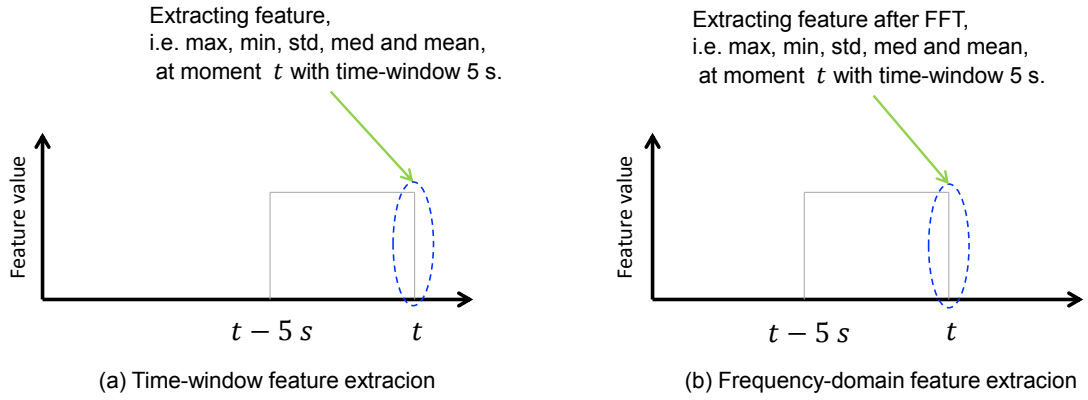


Figure 5.6: Illustration of the time-window feature and the frequency domain feature.

Table 5.6: Description of the extracted features.

#	Feature name	Feature description
1	$yawRate_t$	yaw rate of ego vehicle at time t
2	az_t	az of ego vehicle at time t
3	ax_t	ax of ego vehicle at time t
4	$\mathrm{TTC}_\mathrm{t}^{-1}$	$\mathrm{TTC}_\mathrm{t}^{-1}$ at time t
5	$\mathrm{TLC}_\mathrm{t}^{-1}$	$\mathrm{TLC}_\mathrm{t}^{-1}$ at time t
6	$mean\_yaw_t^5$	mean of $yawRate$ in TW 5 s
7	$mean\_yaw_t^4$	mean of $yawRate$ in TW 4 s
8-10	⋮	$mean\_yaw_t$ in TW 3 s, 2 s, 1 s
11	$std\_yaw_t^5$	std of $yawRate$ in TW 5 s
12-15	⋮	std_yaw_t in TW 4 s, 3 s, 2 s, 1 s
16	$max\_yaw_t^5$	maximum of $yawRate$ in TW 5 s
17-20	⋮	max_yaw_t in TW 4 s, 3 s, 2 s, 1 s
21	$min\_yaw_t^5$	minimum of $yawRate$ in TW 5 s
22-25	⋮	min_yaw_t in TW 4 s, 3 s, 2 s, 1 s
26	$med\_yaw_t^5$	median of $yawRate$ in TW 5 s
27-30	⋮	med_yaw_t in TW 4 s, 3 s, 2 s, 1 s
31	$mean\_az_t^5$	mean of the az in TW 5 s
32-35	⋮	$mean\_az_t$ in TW 4 s, 3 s, 2 s, 1 s
36	$std\_az_t^5$	standard deviation of az in TW 5 s
37-40	⋮	std_az_t in TW 4 s, 3 s, 2 s, 1 s
41	$max\_az_t^5$	maximum of az in TW 5 s
42-45	⋮	max_az_t in TW 4 s, 3 s, 2 s, 1 s
46	$min\_az_t^5$	minimum of az in TW 5 s
47-50	⋮	min_az_t in TW 4 s, 3 s, 2 s, 1 s
51	$med\_az_t^5$	median of az in TW 5 s
52-55	⋮	med_az_t in TW 4 s, 3 s, 2 s, 1 s
56	$mean\_ax_t^5$	mean of the ax in TW 5 s
57-60	⋮	$mean\_ax_t$ in TW 4 s, 3 s, 2 s, 1 s
61	$std\_ax_t^5$	standard deviation of ax in TW 5 s
62-65	⋮	std_ax_t in TW 4 s, 3 s, 2 s, 1 s
66	$max\_ax_t^5$	maximum of ax in TW 5 s
67-70	⋮	$mean\_ax_t$ in TW 4 s, 3 s, 2 s, 1 s
71	$min\_ax_t^5$	minimum of ax in TW 5 s
72-75	⋮	min_ax_t in TW 4 s, 3 s, 2 s, 1 s
76	$med\_ax_t^5$	median of ax in TW 5 s
77-80	⋮	med_ax_t in TW 4 s, 3 s, 2 s, 1 s
81	$max\_F\_yaw_t^5$	max $yawRate$ FFT coefficients in TW 5 s
82-85	⋮	$max\_F\_yaw_t$ in TW 4 s, 3 s, 2 s, 1 s
86	$max\_F\_az_t^5$	max az FFT coefficients in TW 5 s
87-90	⋮	$max\_F\_az_t$ in TW 4 s, 3 s, 2 s, 1 s
91	$max\_F\_ax_t^5$	max ax FFT coefficients in TW 5 s
92-95	⋮	$max\_F\_ax_t$ in TW 4 s, 3 s, 2 s, 1 s

5.4.3 Data labeling

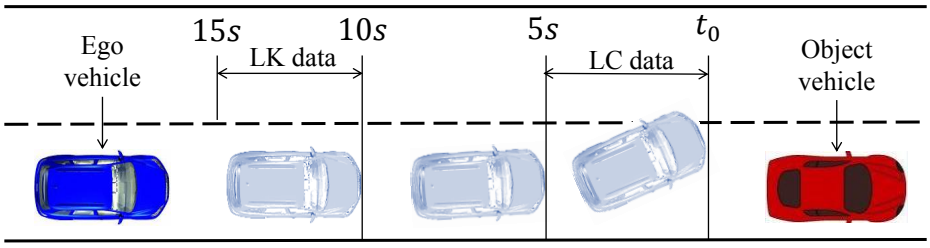


Figure 5.7: Labeling for LC and LK datasets.

In order to evaluate the extracted features, LC datasets and LK datasets should be labeled. To label LC events, one of the most important things is to find t_0. Take LLC for example. As it is shown in Figure 5.7, the ego vehicle (blue) intends to overtake the slow vehicle (red) by left lane change. The moment that the left wheel of the ego car just crosses the central dotted line is marked as the initial LC time t_0. Since there is no eye-tracker equipped in SPMD vehicles, the GBL method proposed in section 4.4.1 cannot be implemented. Instead, we can only use the TWL method to label datasets. Based on the study in Salvucci and Liu (2002), it suggests that the driver tends to start a LC maneuver approximately 5 s before an accrual LC. Thus in this study, time series data between t_0 and 5 seconds before are labeled as LC data samples. To ensure LK datasets are separation of LC datasets, LK data samples are labeled between 10 seconds and 15 seconds prior to t_0. The same rule is applied to RLC.

5.5 Evaluation method

5.5.1 Feature evaluation

From the perspective of statistics, p-value is commonly used to test whether there is statistical significance between two groups. If there is statistical significance between LC datasets and LK datasets, it is thus to say that the extracted features are probably good indicators for classification of LC and LK data samples. However, only using p-value to evaluate significance is not enough (Sullivan and Feinn, 2012). The effect size, such as Cohen's d (Cohen, 1988), is also an important evaluation metric (Cohen, 1990):

$$d = \frac{|M_1 - M_2|}{\sqrt{\frac{S_1^2 + S_2^2}{2}}} \tag{5.14}$$

where
d = Cohen's index,
M_1 = mean of the first group data,
M_2 = mean of the second group data,

S_1 = standard deviation of the first group data,
S_2 = standard deviation of the second group data.

In order to define significance level, Cohen defines the effect class as follow (J.Cohen, 1992):

- $d < 0.5$: small effect
- $0.5 \leq d < 0.8$: medium effect
- $d \geq 0.8$: large effect

For each LC behavior, we label LC and LK datasets and calculate both Cohen's d and p-value for each feature. Then for all LC cases, we average the Cohen's d and p-values to get the mean for each feature in each scenario.

5.5.2 Models used for feature evaluation

In order to test if using the selected feature has advantages over using all the features for ML models, SVM, naive Bayes (NB), Decision Tree (DT) and k-nearest neighbors (KNN) are chosen to evaluate the classification performance. We implement the above ML models through the Statistics and Machine Learning Toolbox[11] provided by Matlab. Here, the SVM model is set with a Gaussian kernel function, and the NB with Kernel smoothing density estimation method, the DT with the default setting and the KNN using an empirical prior with $k = 1$. The datasets used for training those models are the same.

[11] https://www.mathworks.com/products/statistics.html (visited on 31.07.2019)

5.6 Result and analysis

5.6.1 Analysis on effect size and p-value

All the evaluation results (Cohen's d and p-value) for each feature in both LLC and RLC scenarios can be found in Appendix Table B.1 and B.2, respectively. In statistical analysis, p-value that is smaller than 0.05 can be regarded as having statistical significance and Cohen's d that is greater than 0.8 represents a large effect level (J.Cohen, 1992). By following this two criterion, we mark each feature with Cohen's d greater than 0.8 and p-value smaller than 0.05 as red in Table B.1. So, those marked features are regarded as strong features statistically and thus can be used for classification of LC and LK data samples. Overall, based on the features marked as red, we find the following results:

- Although some features ($p < 0.05$) have shown statistical significance, marked as blue, they have only medium or small effect size level (Cohen's $d < 0.8$). This result is also coincide with what it is refereed in Sullivan and Feinn (2012) that only using p-value to evaluate statistical significance is not enough.
- Vehicle dynamic features e.g. $yawRate_t$ (#1), az_t (#2) and ax_t (#3), and combined feature $\mathrm{TLC}_\mathrm{t}^{-1}$ (#5) are not strong features for LLC case with no items marked as red. For RLC case, only az_t and $\mathrm{TLC}_\mathrm{t}^{-1}$ in *RLC Scenario 0_1* can be regarded as strong features. This implies that from the statistical view the common empirical knowledge of using these features is not that much convincing. Additional feature selection work should be done before using them.
- We mentioned that $\mathrm{TTC}_\mathrm{t}^{-1}$ is only calculated when the front cell of the ego vehicle is occupied by an object vehicle ($\mathrm{cell}_\mathrm{m} = 1$). $\mathrm{TTC}_\mathrm{t}^{-1}$ is marked as a strong feature in LLC case, which demonstrates that the potential of the rear-end collision does impact on driver LC decision. A hypothesis has been made by many research, i.e. *if the driver follows a leading vehicle which is too slow, he/she would probably maneuver a LC to overtake the slow leading vehicle*. And this analysis from the naturalistic driving data proves that this hypothesis is reasonable.
- Feature #56 – #60, which refer to *mean_ax*, represent no statistical significance at all and thus are regarded as unimportant features.
- To analyze the time-window (TW) features (#6 – #95), we take an example of the marked strong features in *LLC Scenario 0_0* and *LLC Scenario 0_1*. To make it clear, we segment the table horizontally with 5 features in a group, e.g. feature #6 – #10 are related to the same feature *mean_yaw* but with different TW from 5 s – 1 s etc. The detailed illustration in both LLC and RLC scenarios can be seen in Table B.1 and B.2 in Appendix, where features that have the largest Cohen's d and the smallest p-value are marked with '▲' and '▼', respectively. From these peak and valley values we find that features with the greatest Cohen's d are also likely

to have the smallest p-values, except for feature #31 and #32 in *LLC Scenario 0_0*. We select the final features for each scenario based on the marked peak and valley features, and for the special case like feature #31 and #32, features with greater Cohen's d (e.g. #31) are selected.

5.6.2 Final selected features for each LC scenario

Based on the marked features from the results, the final selected features in both LLC and RLC scenarios are listed in Table 5.7 and Table 5.8. It can be found that different LC scenarios have different features sets. The number of selected features from all 95 features for each LC scenario ranges from 8 to 16 differently. There is no feature set which is eligible for all the LC scenarios. This is why we select the features based on different LC scenarios. In *LLC Scenario 0_0* and *LLC Scenario 1_0*, *RLC Scenario 0_0*, and *RLC Scenario 1_0*, there are no vehicle dynamic features and combined features (#2, #4, #5) selected which means that these features are overestimated.

Although vehicle dynamic features related to vehicle lateral movement ($yawRate_t$ (#1), az_t (#2), and ax_t (#3)) are not contributive as expected, their corresponding time-window features indicate larger effect sizes. This implies that the property of the vehicle dynamic features within certain TW may contain more important information. In addition, frequency-domain features are also promising features, with nearly at least one feature being selected as strong feature in each scenario. The exception is in *LLC Scenario 0_1*, where no frequency-domain features are selected. We will use the final selected features to train ML models and further evaluate their classification performance.

Table 5.7: The final selected strong features of each LLC scenario.

#	Feature	LLC Scenario							
		0_0		0_1		1_0		1_1	
		d	p	d	p	d	p	d	p
4	$\mathrm{TTC}_{\mathrm{t}}^{-1}$	–	–	0.92	0.04	–	–	1.18	0.01
6	$mean\_yaw_t^5$	1.02	0.03	–	–	–	–	1.54	0.04
7	$mean\_yaw_t^4$	–	–	0.98	0.04	1.14	< 0.01	–	–
11	$std\_yaw_t^5$	–	–	1.05	0.03	–	–	–	–
12	$std\_yaw_t^4$	–	–	–	–	–	–	1.03	0.01
13	$std\_yaw_t^3$	0.99	0.04	–	–	–	–	–	–
16	$max\_yaw_t^5$	1.00	0.04	–	–	–	–	–	–
17	$max\_yaw_t^4$	–	–	0.97	0.03	–	–	1.36	0.03
21	$min\_yaw_t^5$	–	–	0.95	0.03	–	–	–	–
22	$min\_yaw_t^4$	1.03	0.03	–	–	–	–	1.20	0.04
23	$min\_yaw_t^3$	–	–	–	–	1.04	0.03	–	–
26	$med\_yaw_t^5$	0.92	0.04	–	–	–	–	–	–
27	$med\_yaw_t^4$	–	–	–	–	0.95	< 0.01	–	–
31	$mean\_az_t^5$	0.92	0.04	–	–	0.87	0.01	–	–
32	$mean\_az_t^4$	–	–	0.95	0.04	–	–	–	–
36	$std\_az_t^5$	0.91	0.04	1.05	0.04	–	–	–	–
38	$std\_az_t^3$	–	–	–	–	1.23	< 0.01	–	–
41	$max\_az_t^5$	0.98	0.04	–	–	–	–	–	–
42	$max\_az_t^4$	–	–	1.01	0.03	–	–	–	–
43	$max\_az_t^3$	–	–	–	–	–	–	0.85	0.04
46	$min\_az_t^5$	0.93	0.03	0.90	0.04	–	–	–	–
50	$min\_az_t^1$	–	–	–	–	–	–	1.04	0.03
51	$med\_az_t^5$	0.88	0.04	0.94	0.04	0.97	0.01	–	–
54	$med\_az_t^2$	–	–	–	–	–	–	0.91	0.03
61	$std\_ax_t^5$	–	–	–	–	1.21	< 0.01	–	–
62	$std\_ax_t^4$	–	–	–	–	–	–	1.02	0.03
68	$max\_ax_t^3$	–	–	–	–	1.06	0.04	–	–
71	$min\_ax_t^5$	0.94	0.04	–	–	–	–	–	–
72	$min\_ax_t^4$	–	–	–	–	0.99	< 0.01	–	–
82	$max\_F\_yaw_t^4$	–	–	–	–	0.87	0.01	–	–
83	$max\_F\_yaw_t^3$	–	–	–	–	–	–	1.14	0.04
86	$max\_F\_az_t^5$	0.98	0.04	–	–	–	–	0.83	0.02
93	$max\_F\_ax_t^3$	–	–	–	–	1.18	< 0.01	–	–
Selected amount		12		10		11		11	

Table 5.8: The final selected strong features of each RLC scenario.

#	Feature	RLC Scenario							
		0_0		0_1		1_0		1_1	
		d	p	d	p	d	p	d	p
2	az_t	–	–	0.96	0.04	–	–	–	–
4	$\mathrm{TTC}_\mathrm{t}^{-1}$	–	–	–	–	–	–	1.18	< 0.01
5	$\mathrm{TLC}_\mathrm{t}^{-1}$	–	–	0.82	0.02	–	–	–	–
6	$mean\_yaw_t^5$	0.92	0.04	1.29	0.04	0.96	0.04	1.54	< 0.01
11	$std\_yaw_t^5$	1.01	0.03	–	–	1.02	0.03	0.98	0.03
13	$std\_yaw_t^3$	–	–	1.60	< 0.01	–	–	–	–
16	$max\_yaw_t^5$	0.97	0.03	–	–	0.95	0.04	–	–
18	$max\_yaw_t^3$	–	–	–	–	–	–	1.40	< 0.01
21	$min\_yaw_t^5$	0.98	0.04	1.10	0.02	1.06	0.02	1.38	< 0.01
26	$med\_yaw_t^5$	–	–	–	–	1.04	0.03	–	–
28	$med\_yaw_t^3$	–	–	–	–	–	–	1.23	0.01
31	$mean\_az_t^5$	–	–	1.16	< 0.01	–	–	–	–
32	$mean\_az_t^4$	–	–	–	–	0.98	0.04	–	–
36	$std\_az_t^5$	0.91	0.04	0.85	0.02	0.94	0.04	–	–
38	$std\_az_t^3$	–	–	–	–	–	–	0.92	0.03
41	$max\_az_t^5$	–	–	–	–	0.98	0.04	–	–
42	$max\_az_t^4$	0.95	0.04	–	–	–	–	–	–
46	$min\_az_t^5$	0.84	0.04	1.28	0.02	0.94	0.03	–	–
51	$med\_az_t^5$	–	–	1.01	< 0.01	–	–	0.94	< 0.01
52	$med\_az_t^4$	–	–	–	–	1.00	0.04	–	–
61	$std\_ax_t^5$	–	–	1.16	0.02	–	–	–	–
62	$std\_ax_t^4$	–	–	–	–	–	–	1.02	0.01
66	$max\_ax_t^5$	–	–	1.13	0.01	–	–	–	–
71	$min\_ax_t^5$	–	–	1.10	0.02	–	–	–	–
72	$min\_ax_t^4$	–	–	–	–	–	–	1.12	0.03
76	$med\_ax_t^5$	–	–	1.06	0.04	–	–	–	–
81	$max\_F\_yaw_t^5$	–	–	0.95	0.01	–	–	–	–
82	$max\_F\_yaw_t^4$	0.88	0.04	–	–	–	–	1.20	0.03
86	$max\_F\_az_t^5$	–	–	0.88	0.02	0.95	0.04	–	–
87	$max\_F\_az_t^4$	–	–	–	–	–	–	0.86	0.04
91	$max\_F\_ax_t^5$	–	–	–	–	–	–	1.16	0.02
94	$max\_F\_ax_t^2$	–	–	1.11	< 0.01	–	–	–	–
Selected amount		8		16		11		13	

5.6.3 Evaluation of different machine learning models using the selected features

In order to test if using the selected features could really improve the model performance, we compare the classification results of the ML models in section 5.5.2 both with the selected features (termed as 'Selected') and all the features (termed as 'All Features'). To guarantee that the training data and testing data are disjoint, a cross-validation (CV) method is used. The datasets are evenly divided into ten folds. Nine folds are used to train the models and the remaining is used to test the models. ROC curves and AUC values are as the metrics to evaluate the model performance as it is used in chapter 4. All the ROC curves regarding to the classification performance are illustrated in Appendix, where Figure B.1–B.4 are for LLC case and Figure B.5–B.8 for RLC case. The corresponding AUC values are listed in Table 5.9 and 5.10.

In Table 5.9 and 5.10, comparison is made between using all the features and the selected features for each ML model in both LLC scenarios and RLC scenarios. Since the aim is to check if the model performance can be improved by using the selected features instead of using all the features, we denote '↓' as performance deterioration. The improvement in percentage can be also seen from the table. Summarily, we found the following result:

- Using the selected features, the performance of KNN can be improved in all the scenarios. DT has the similar results. Using selected features, the performance of DT can be improved significantly from 0.82 to 0.98 (an increase of 19.5%), and only it only shows tiny deterioration (1.0%) in *LLC Scenario 1_1* and *RLC Scenario 1_1*.
- For SVM, using the selected features can largely improve the classification performance (performance increase between 4.2% and 13.6%) compared with using all the features, and only show declination in LLC Scenario 0_1.
- NB represents different pictures. Using the selected features can hardly have any improvements compared with using all the features. For example there is no improvement in all LLC scenarios. It also happens to *RLC Scenario 0_1* and *RLC Scenario 1_1*. The poor performance of NB may be caused by the conditional independent assumption between features. This is the biggest downside of NB.

Table 5.9: The AUC values of the classification results by different models using the selected features and all features in LLC scenarios.

Feature type	LLC Scenario 0_0				LLC Scenario 0_1			
	SVM	NB	DT	KNN	SVM	NB	DT	KNN
All features	0.90	0.85	0.82	0.96	0.92	0.85	0.94	0.99
Selected	0.99	0.81 ↓	0.98	0.98	0.84 ↓	0.78 ↓	0.97	0.99
Improvement (%)	10.0	-4.7	19.5	2.0	-8.6	-8.2	3.1	0
Feature type	**LLC Scenario 1_0**				**LLC Scenario 1_1**			
	SVM	NB	DT	KNN	SVM	NB	DT	KNN
All features	0.88	0.94	0.99	0.98	0.93	0.97	0.99	0.98
Selected	1	0.93 ↓	0.99	1	0.99	0.96 ↓	0.98 ↓	0.99
Improvement (%)	13.6	-1.0	0	2.0	6.4	-1.0	-1.0	1.0

Table 5.10: The AUC values of the classification results by different models using the selected features and all features in RLC scenarios.

Feature type	RLC Scenario 0_0				RLC Scenario 0_1			
	SVM	NB	DT	KNN	SVM	NB	DT	KNN
All features	0.94	0.87	0.91	0.97	0.95	0.94	0.99	0.96
Selected	0.98	0.80 ↓	0.98	0.99	0.99	0.95	0.98 ↓	1
Improvement (%)	4.2	-8.0	7.6	1.0	4.2	1.0	-1.0	4.1
Feature type	**RLC Scenario 1_0**				**RLC Scenario 1_1**			
	SVM	NB	DT	KNN	SVM	NB	DT	KNN
All features	0.93	0.83	0.92	0.97	0.92	0.93	0.97	0.96
Selected	0.97	0.75 ↓	0.98	0.99	0.99	0.98	0.98	0.98
Improvement (%)	4.3	-9.6	6.5	2.0	7.6	5.3	1.0	2.0

5.7 Summary

In this chapter, we aim to propose a feature selection methodology in the view of statistics. Unlike other feature selection technologies which select features for specific algorithm, the method proposed in this study is more general and can be used for all the algorithms. In order to make the statistical analysis convincing, a big data analysis based on naturalistic datasets is implemented.

To enrich the feature sets, vehicle dynamic features directly collected from the on-board sensors, combined features e.g. TTC and TLC, as well as time-window features were extracted. In addition, features are not restricted to time-domain features, frequency-domain features are also considered. Totally 95 features were extracted. In order to comprehensively evaluate the features based on driving scenarios, a 3-cell grid was used to model the contextual traffic. Both effect size (Cohen's d) and p-value were calculated as the metrics of feature selection. Results show that frequency-domain features, which are rarely used in driver behavior related research, are also promising features, with nearly at least one feature being selected as strong feature in each scenario. From the final selected feature sets we find that for different LC scenarios, the final selected features are different. Thus, feature selection should be based on driving scenarios. In addition, features refer to vehicle lateral movement (az_t and $\mathrm{TLC}_\mathrm{t}^{-1}$) which are commonly used for recognition of driver LC behavior, do not represent statistical significance (only except for $\mathrm{TLC}_\mathrm{t}^{-1}$ in *RLC Scenario 0_1*). This counter-empirical result makes it more worthwhile doing feature selection work rather than just selecting features based on the empirical knowledge.

Finally, we compared the classification performance of using the final selected features to that of using all the features in each LC scenario. The result shows that except for the relatively poor performance of naive Bayes, the performance of SVM and Decision Tree, as well as KNN, can be improved from different levels in most of the LC scenarios. Summarily, the high performance achieved by the ML models using all the features (95 features) is at the expense of computation time. Considering the fact that using the selected features (nearly only 10 features), ML models can still achieve the same performance or even have improvement, it can be concluded that it is more efficient and effective to using the selected features.

Summarily, the methodology presented in this study can be used for selecting any features regarding to driver LC behavior, however, there are still some limitations that can be improved in the further study:

- the raw naturalistic datasets are provided by the third party from U.S, where the driving scenarios and driving rules are slightly different from Germany. In different driving scenarios and based on different traffic rules, the final selected features as well as feature evaluation result may lead to different results.

- the extracted feature sets in this study (in Table 5.6) are limited by vehicle dynamic features (like yaw rate, acceleration etc.). There are no features related to driver behavior. For example, driver eye movements, driver maneuvering features e.g. steering wheel angle and brake pedal data etc. Since there are no such features available in the naturalistic datasets we used.

In order to overcome the limitations mentioned above and to further evaluate the framework proposed in chapter 4, an experiment conducted on German road is necessary.

6

Experiment 2 - Evaluation based on a real-road experiment

6.1 Introduction

In order to further evaluate the methods proposed in the last two chapters and to overcome the limitations, a real-road based study is made in this chapter. The content includes experimental design, data processing, data labeling, feature selection as well as evaluation. The real-road experiment was carried out in the period between 15.01.2019 to 19.03.2019 with 12 subjects participated[12].

6.2 Experimental design

This part details the whole design of the real-road experiment which includes the materials and equipment, and the synchronization method for data acquisition as well as participants recruiting and driving task description.

6.2.1 Equipment

Testing vehicle

The testing vehicle is BMW 520 Touring with diesel engine and 8-gear automatic transmission. There are some ADAS functionalities in the testing vehicle, such as head-up display, adaptive cruise control system, lane departure warning system and parking assistance system. However, except for head-up display, during the entire experiment participants are not allowed to use the rest functionalities.

[12]This experiment is supported by Joyson Safety Systems GmbH. They provided the testing vehicle and the necessary sensors. In addition, the cost of recruiting participants is also covered by the company.

On-board CAN bus systems

The signals from sensors were collected through CAN bus systems and were recorded by an on-board PC, which was located below the truck floor of the vehicle. Totally there were four CAN bus systems used for data collection, i.e. S-CAN, K-CAN, PT1 and PT2-CAN[13]. The location of the four CAN systems and the on-board PC is shown in Figure 6.1.

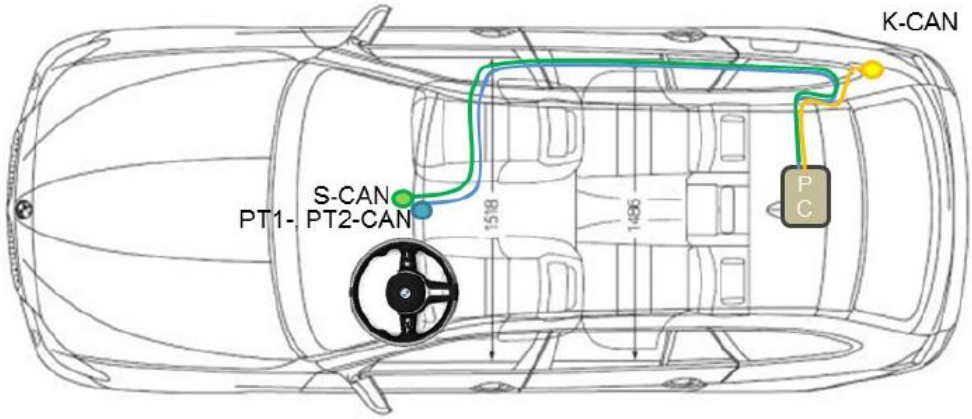


Figure 6.1: CAN bus setup in the testing vehicle. Picture extracted from Zühlsdorff (2018), where the same testing vehicle is used as in this study.

Figure 6.2: The first person view from the participant who wears the SMI glasses is driving on the highway, where the blue point is the fixation monitored by eye-tracker.

Eye-tracker setup

The eye-tracker used in this experiment is the same as it is used in chapter 4, see Figure 4.5. The difference is that in chapter 4, the experiment was conducted in

[13]S-CAN, K-CAN, PT1 and PT2-CAN are different CAN channels.

a driving simulator but in this chapter the experiment was a real-road experiment. Figure 6.2 illustrates the first person view from one participant on highway. The entire driving scenario during the test in the view of the participant can be recorded by the eye-tracking glasses, which can be used for further data processing. The sampling rate of the eye-tracker is at 30 Hz.

Data synchronization

Since the eye-tracking data are not collected through CAN bus, a real-time synchronization between signals from CAN bus and the eye-tracker is crucial for further data processing.

The real-time synchronization work is done as follows: during the experiment, messages are sent from PC used for recording CAN signals (Figure 6.1) to the eye-tracker at a frequency of 0.1 Hz. In Figure 6.3a, the blue points are the messages received by the eye-tracker from the on-board PC. Each message contains a Unix timestamp in seconds based on the current time from PC and the eye-tracker timestamp. An example of the messages for synchronization can be seen in Figure 6.3b, where the term *Timestamp* is the eye-tracker time and *Value* is the Unix timestamp[14] from the on-board PC. In this way, data collected from the on-board PC and the eye-tracker is synchronized.

6.2.2 Participants

Based on the regulation of Joyson Safety Systems GmbH, only the employees of the company could drive the testing vehicle, thus the participants were selected among the personnel. Since the participants were asked to wear eye-tracking glasses during the entire drive, for calibration issue[15], only participants who have normal vision or wear contact lenses can meet the requirement. The questionnaire that is used to select participants can be found in Appendix C.1.1.

Totally 12 volunteers (6 male and 6 female drivers) participated the experiment, among which 9 participants are in the age group between 25 and 39, 2 participants in the age group between 40 - 54 and one is over 55. All the participants are native German, having had their driver licenses with minimum 9 years and maximum 45 years (Mean = 19.08 years). They are familiar with the traffic rules in Germany, and could understand the questionnaire and the instruction in German without having any trouble. The detailed statistics of the demographic questionnaire regarding to participants' gender, age, driving kilometers as well as driving frequency can be found in Appendix Figure C.1.

[14]The unix time stamp is a way to track time as a running total of seconds. This count starts at the Unix Epoch on January 1st, 1970 at UTC. This is very useful to computer systems for tracking and sorting dated information in dynamic and distributed applications both online and client side. https://www.unixtimestamp.com/ (visited on 31.07.2019)

[15]If the driver wears glasses, the reflection of the glasses may lead to the failure of calibrating SMI eye-tracking glasses.

(a) A screen shot taken from the software used to process eye-tracking data, where the blue points are the messages from PC used to record CAN signals carrying Unix timestamps.

```
" Timestamp="114539" />  <TextData EyeFrameIndex="-1" Value="1552556171716
" Timestamp="124540" />  <TextData EyeFrameIndex="-1" Value="1552556181726
" Timestamp="134548" />  <TextData EyeFrameIndex="-1" Value="1552556191746
" Timestamp="144569" />  <TextData EyeFrameIndex="-1" Value="1552556201746
" Timestamp="154570" />  <TextData EyeFrameIndex="-1" Value="1552556211757
" Timestamp="164581" />  <TextData EyeFrameIndex="-1" Value="1552556221757
" Timestamp="174582" />  <TextData EyeFrameIndex="-1" Value="1552556231777
" Timestamp="184603" />  <TextData EyeFrameIndex="-1" Value="1552556241779
" Timestamp="194606" />  <TextData EyeFrameIndex="-1" Value="1552556251789
" Timestamp="204616" />  <TextData EyeFrameIndex="-1" Value="1552556261789
" Timestamp="214617" />  <TextData EyeFrameIndex="-1" Value="1552556271799
" Timestamp="224627" />  <TextData EyeFrameIndex="-1" Value="1552556281819
" Timestamp="234648" />  <TextData EyeFrameIndex="-1" Value="1552556291819
```

(b) Illustration of synchronized timestamps between CAN and the eye-tracker.

Figure 6.3: Data synchronization between CAN and the eye-tracker.

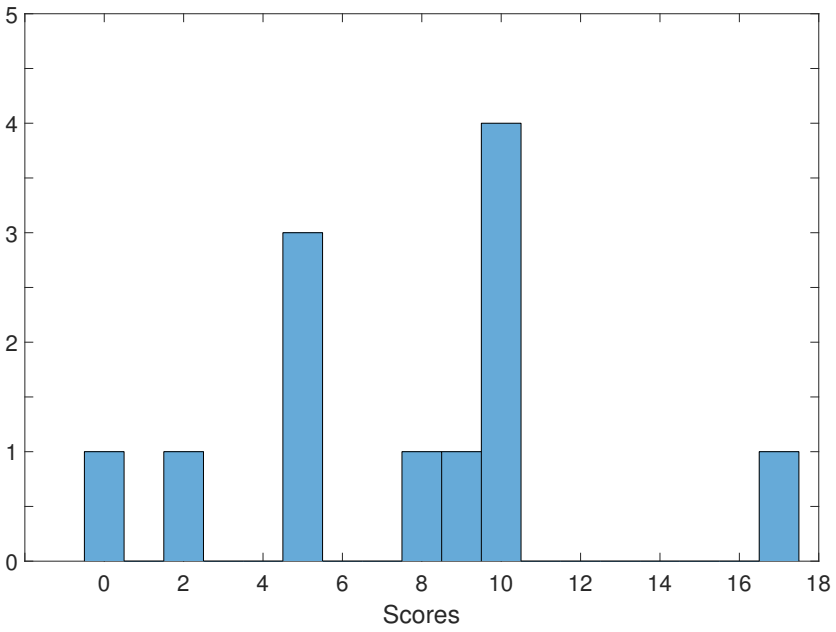


Figure 6.4: Illustration of the aggressiveness scores of the participants in histogram.

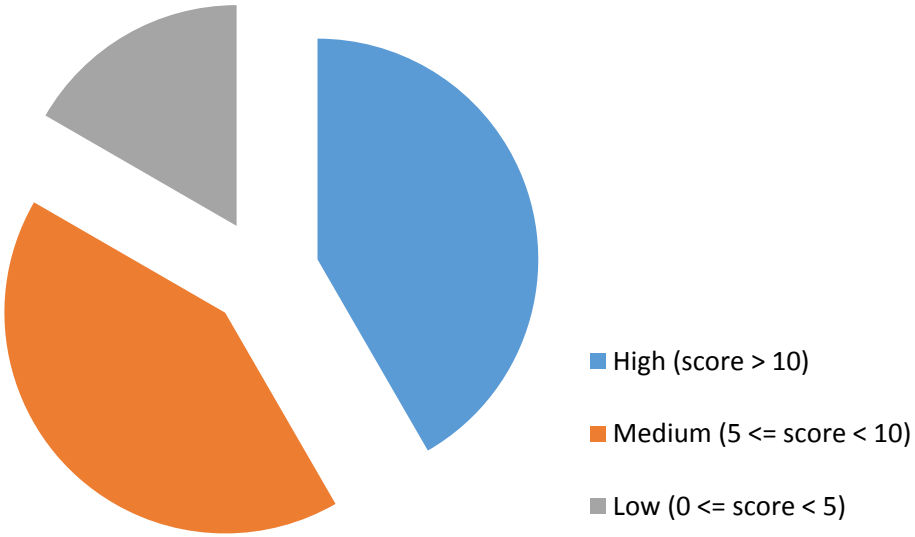


Figure 6.5: The pie chart of classification of driving styles based on scores, where high, medium and low represents the levels of aggressive driving styles.

In order to classify the driving styles of the participants, the same behavioral-psychological questionnaire as it is described in chapter 4 (in Appendix A.1.2) is used. Figure 6.4 illustrates the aggressiveness scores of all the participants. Based on the score of classification from 32 participants in chapter 4.3.3, the driving style classification results of the 12 participants in this experiment is as follows and the distribution of the pie chart can be found in Figure 6.5:

- Low aggressive driving style: scored between 0 and 5.
- Medium aggressive driving style: scored greater than 5 and smaller than 10.
- High aggressive driving style: scored over 10.

The above classification of driving style groups are used for organizing the training datasets which will be detailed in section 6.3.

6.2.3 Driving task

Driving instructions

After signing a declaration of consent to store personal data and completing a demographic questionnaire as well as the behavioral-psychological questionnaire, the participants were instructed by the assist[16] about how to use the testing vehicle before start the journey.

[16]There were two assists who are two master students sitting on back row of the testing vehicle. One assist was responsible for giving on-board instruction and checking the status of the on-board PC which is used for recording data. Another assist was responsible for manually labeling the driving behaviors of the participants which is detailed in 6.3.

The participants were informed that their eye movements will be recorded by the eye-tracker, and also the GPS data, driving dynamics movements. But the exact purpose of the experiment was not told to them in order not to impact on their driving habits. The functionalities of some assistance systems which are available in the vehicle were explained, e.g. the adaptive cruise control and head up display, but the participants can only use the head up display. The participants were also told the procedure of the experiment, e.g. how long it takes, when to break for a while, the possible on-board instruction, calibration of the eye-tracker as well as how to abort the experiment if they do not want to continue etc. The whole instruction document can be found in Appendix in C.1.4.

Calibration

A 3-points calibration method is used to calibrate the eye-tracking glasses. As we can see in Figure 6.6, the participant was asked to focus his/her fixation on a 3-by-3 matrix card. The three orange squares are used for calibration.

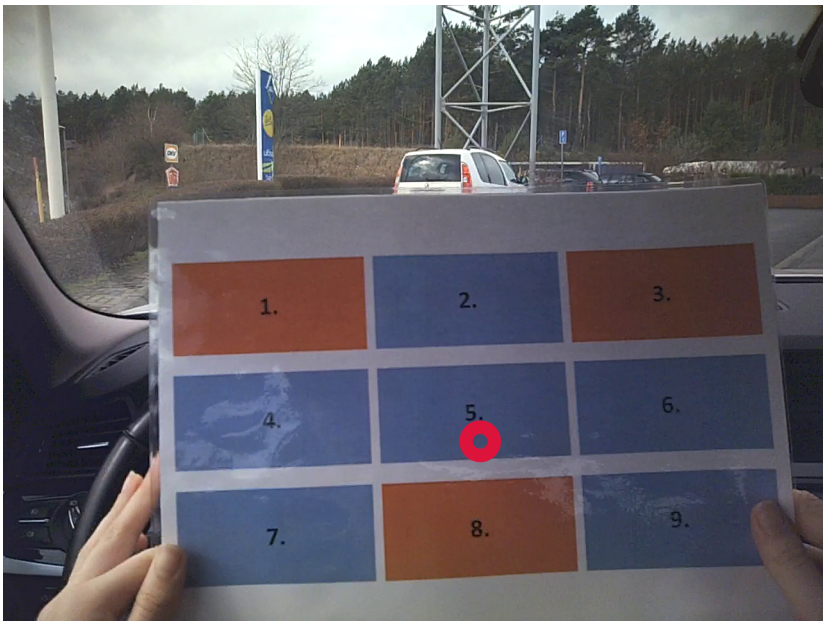


Figure 6.6: The 3-by-3 matrix card used for calibration of the eye-tracker using, where the red point is the fixation point of the participant.

Route of the experiment

After calibration, the participants can start their journey. They do not need to remember the way or use navigation; instead, all the driving guidance was given by the assist. The route of the entire experiment is illustrated in Figure 6.7, which follows clockwise. The start and end point is Hussitenstr. 34, where Joyson Safety System GmbH is located. There are three stages of driving and two breaks during the experiment. The first and the third are city scenarios, between them is on highway A10 where some part of the route has no speed limits. The net driving duration is around two hours and a half excluding the two breaks. Two breaks are before and after driving on highway so that the participants would not be fatigue because of long time driving. During the break the participants do not need to wear the eye-tracker anymore, but re-calibration is needed before they start again.

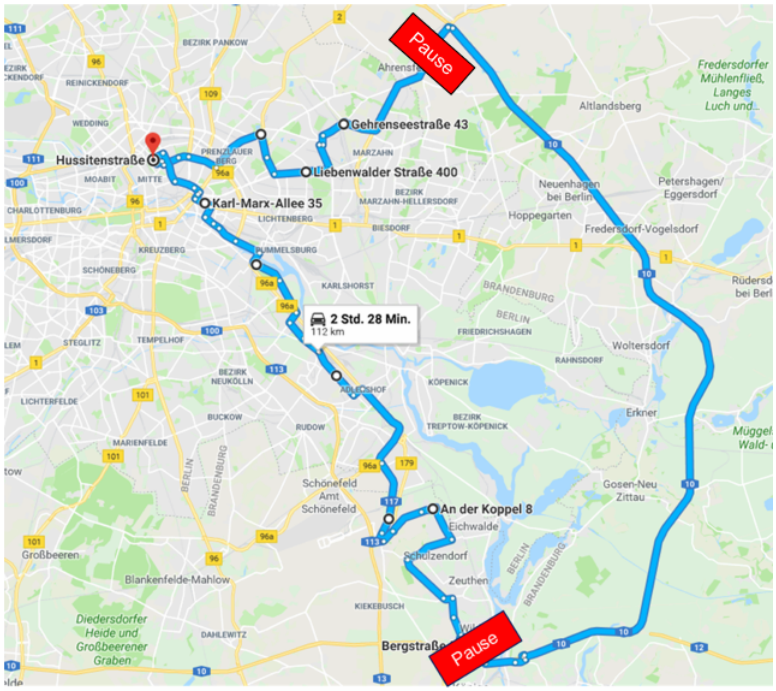


Figure 6.7: A screen shot of google map which captures the route (in clockwise) of the experiment in Berlin.

6.3 Data processing

After two months, the whole experiment was finished with totally 158 Gigabyte data collected. Data collected by SMI eye-tracker are the majority, taking up 152 Gigabyte. The videos of the entire experiment were also recorded as it is shown in Figure 6.2. Data from CAN bus take up the rest 6 Gigabyte. All the data were stored separately by each participant. After checking the integrity of the datasets, it is found that data collected from two participants failed to be synchronized and thus would not be used for further

study. In addition, only data collected on highway (the second stage in Figure 6.7) is processed since the focus of this study is mainly on highway roads[17].

6.3.1 Eye-tracking data processing

The software used for processing the eye-tracking data in this experiment is software BeGaze 3.7, which is the updated version of BeGaze 3.6 used in chapter 4 with no new feature upgraded.

Defining AoI for real-road scenario

According to Lee, Olsen, Wierwille, et al. (2004), the most likely glancing locations while driving are forward, rear view mirror, left mirror as well as blind spot region like left side window and right side window. In experiment 1 presented by chapter 4, the setting of AoIs is limited by the driving simulator. In the simulator scenario, there is no room for the participant to check the blind spot. See Figure 4.4 where the simulated driving scenario is projected on the wall with no blind spot. Limited by this experimental condition, only five AoIs are defined, i.e. *Rear mirror*, *Left mirror*, *Right mirror*, *Speedometer* and *Wind screen*. Considering blind spot checking is a very important driver behavior before executing a LC maneuver, conducting the experiment using the driving simulator suffers this limitation. However for the real-road experiment, there is no such limitation at all. Two additional AoIs are defined in this study, i.e. *Left window* and *Right Window*, used for monitoring driver blind spot checking behavior. Figure 6.8 is the illustration of the defined AoIs for the read-road driving scenario. The orange and blue points are the fixations of the participants.

Although in the real-road scenario it is possible to define more flexible AoIs than in the driving simulator scenario, it is also more challenging to label each defined AoI than in the driving simulator scenario. The reason is that in the driving simulator experiment, due to the limited viewing angle (the participants is focusing on the projected scenario on the wall), the participants do not need to check the blind spot so their head orientation are nearly fixed. With nearly fixed head orientation, the recorded frame by the eye-tracker is also nearly fixed[18]. This makes it easier to label AoIs for each frame. However, in the real-road scenario, the participants are more likely to rotate their heads in order

[17]The target of this study is on highway road, however, the experiment was conducted both on highway and in city road. The reason is that this real-road experiment, which is supported by Joyson Safety Systems GmbH, is not only for the purpose of this study but also for other research purposes by the company itself. This would not impact on this study since all the participants were not told the purpose of the experiment, but were told to drive like normal without giving them any additional instruction.

[18]Fixations, which are presented by the orange and blue points in Figure 6.8 are frame based. That is to say, the coordinate of the fixation is actually the pixel coordinate in reference image. Even for the same AoI, the fixation coordinate in different recorded images can be different. For example, the fixation coordinates in Figure 6.8a and in Figure 6.8d are likely the same, however, they represent different AoIs. The former refers to *Left window* but the latter refers to *Left mirror*. The same situation happens to Figure 6.8c and Figure 6.8f.

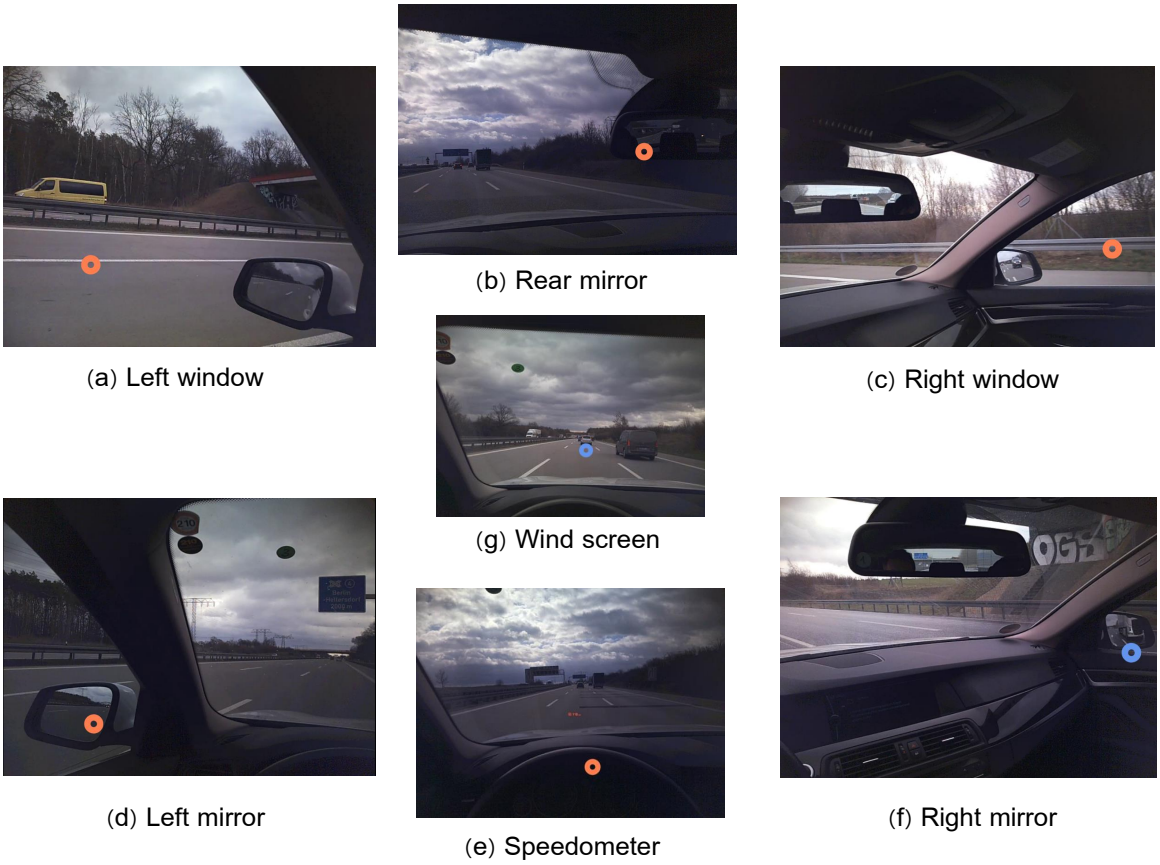


Figure 6.8: The definition of the 7-region AoIs in software BeGaze 3.7.

to check the driving situations. The result is that the recorded frame is also changing frequently, which makes it difficult to use the coordinate value of the fixation to label AoIs. Thus, labeling AoIs for the read-road data is more challenging than the driving simulator data. A Semantic gaze mapping method is used to label AoIs.

Semantic gaze mapping

Fixations and saccades are two important gaze events which indicate human's certain intention (Liu, Yttri, and Snyder, 2010). By monitoring fixation and saccade, Maltz and Shinar (1999) found that younger drivers and older drivers represent significant different visual information processing pattern. Some researches were focusing on identifying fixation and saccade (Salvucci and Goldberg, 2000; Nyström and Holmqvist, 2010). In this chapter we use a semantic gaze mapping way to label AoIs.

Semantic gaze mapping is to identify all the fixation and saccade events, and order them chunk by chunk in time series. The fixations and saccades are processed by SMI BeGaze 3.7. In Figure 6.9, the green chunks are fixations and the length of the chunk represents the duration of the fixation. Between two adjacent fixation chunks are

saccades. AoIs defined in Figure 6.8 are thus labeled by the semantic chunks. In order to avoid missing labels, we manually labeled all the AoIs, chunk by chunk[19].

After labeling all the AoIs, the entire gaze pattern including fixations and saccades as well as the AoIs can be represented in a reference frame, see Figure 6.10. This example uses the data collected from the real-road experiment by one of the participants. In Figure 6.10a, the points in circle are the fixations. The circle with a larger radius means a longer duration. Between two fixations is the scan path of corresponding saccade. We can clearly see that the fixations fall into our pre-defined AoIs. The distribution of the fixations as a heat map can be seen in Figure 6.10b. Regions that covered with larger areas with red color mean that there are more fixations falling in. So *Wind screen* and *Left mirror* are the most popular AoIs for this participant. The data format as well as the content of the output data after labeling AoIs can be found in Appendix in Figure C.2.

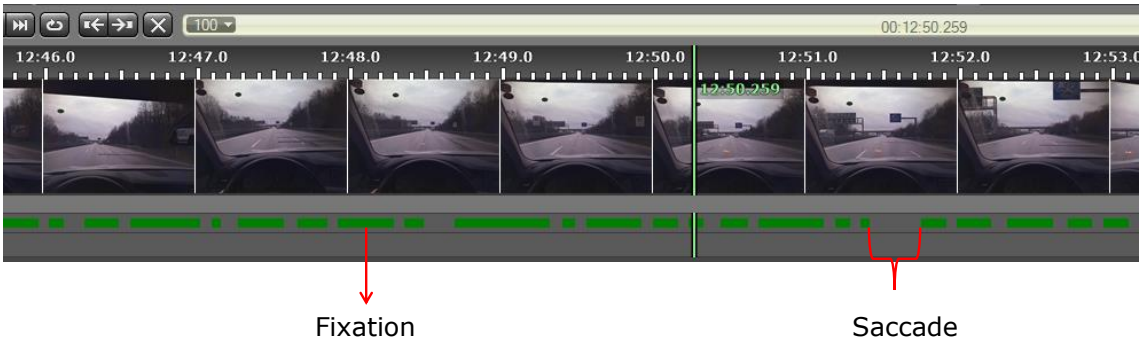


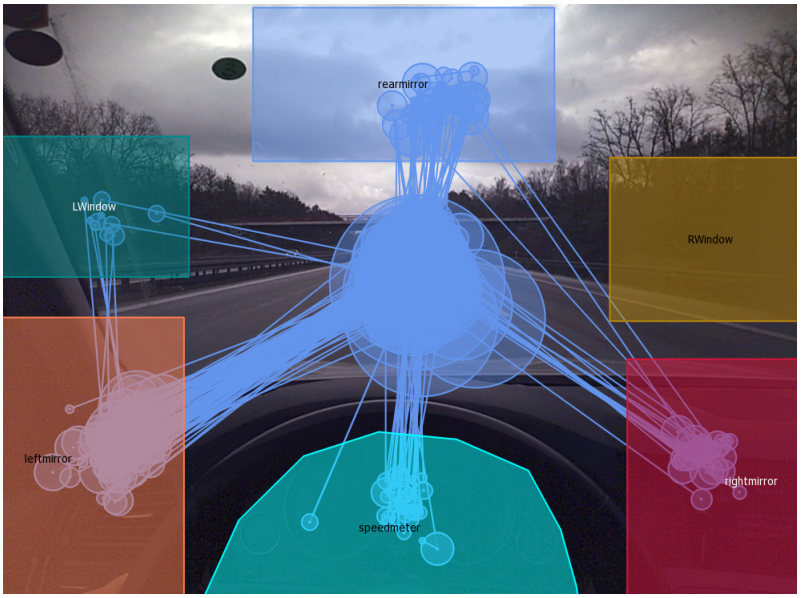
Figure 6.9: A screen shot represents the semantic gaze events in software BeGaze 3.7.

6.3.2 Parsing CAN data

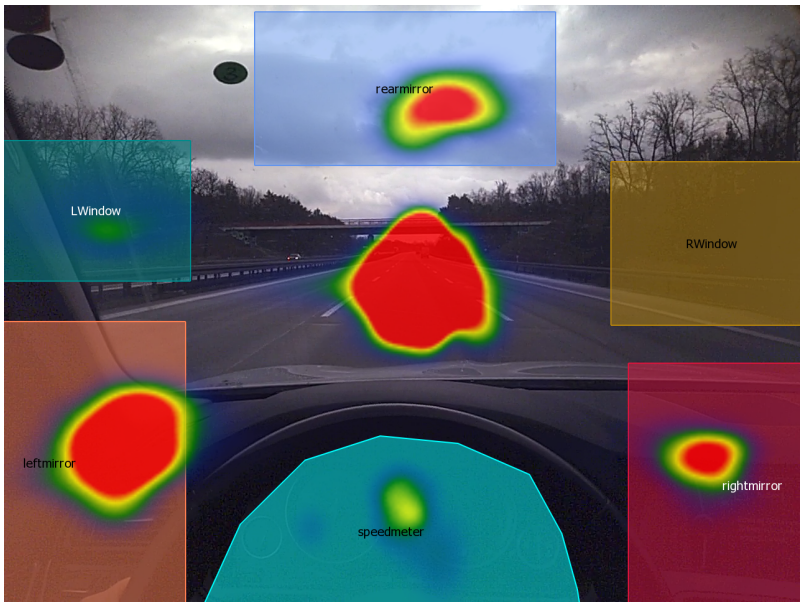
In section 6.2.1 we mentioned that all the signals from on-board sensors were collected through CAN bus and recorded by an on-board PC. To use the raw data for further study, parsing CAN data is needed[20]. The parsed CAN signals are listed in Table 6.1 which illustrates the name and origin of the signals.

[19]Manually labeling AoIs is time consuming, however, to get high quality AoI labels it is worth the effort. This work was done by the author of this dissertation and a master student. For each participant, it takes around 2 hours. If there are considerable more data need to be labeled, it is better to use computer vision techniques.

[20]Thanks to Joyson Safety Systems GmbH who provides all the necessary interpretation about the DBC file for the testing vehicle BMW 520, so that the CAN data can be parsed.



(a) Scan path of gaze movement as well as the fixation pattern with respect to the defined AoIs.



(b) Heat map of gaze fixation pattern with respect to defined AoIs.

Figure 6.10: A reference frame which illustrates the gaze map hitting on AoIs, where the term *LWindow* refers to *Left window* and *RWindow* refers to *Right window*.

Table 6.1: Illustration of the parsed CAN signals.

Name of signal	Origin of signal	Type of CAN
SF_CAN_DV_ACTN_STW	Steering wheel	S-CAN
SF_CAN_DV_ACTN_BRTORQ	Brake pedal	S-CAN
SF_CAN_DV_ACTN_ACPD	Throttle sensor	S-CAN
SF_CAN_V_VEH	Speed sensor	S-CAN
SF_CAN_VYAW_VEH	IMU	S-CAN
NAV_GPS1	GPS	S-CAN
BLINKEN	Turn signal	S-CAN
OBJDT_HDWOBS	ACC	S-CAN

Since the sampling rate between each CAN signal is unequal, a linear interpolation method is used to equalize the number of the samples among different signals. An example of linear interpolation is shown in Figure 6.11, where at t_2, t_4, t_6 and t_8 there are no data collected, thus a linear interpolation method can make up for the missing data. After re-sampling, all the data from CAN bus are at 25Hz. For data fusion purpose, eye-tracking data, which were collected at 30Hz, are down-sampled at 25Hz to match the CAN bus data. In this way, all the data are at 25Hz sampling rate.

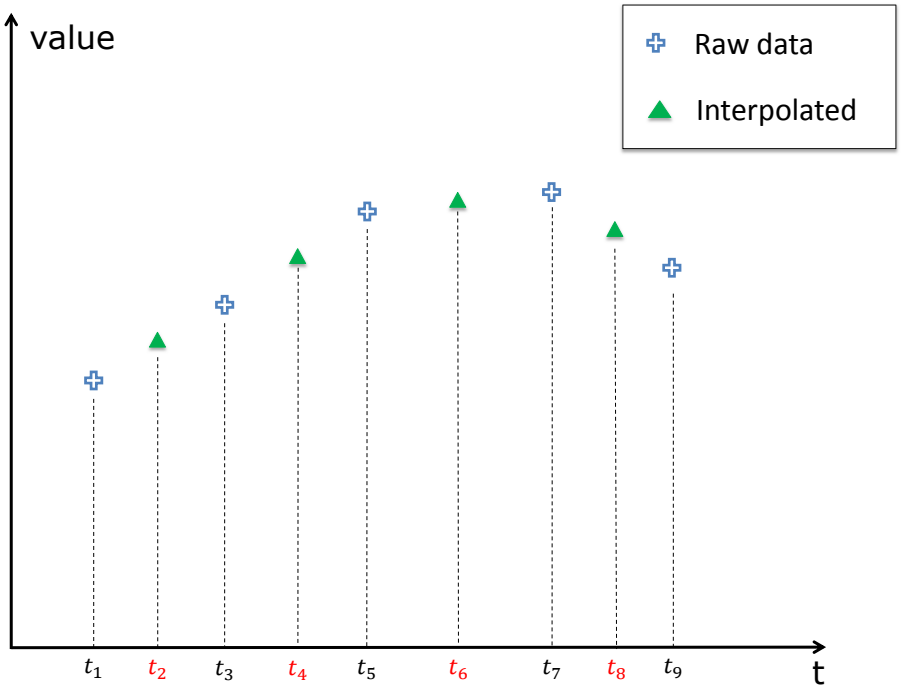


Figure 6.11: An example of how linear interpolation works.

6.3.3 Feature extraction

Although the feature extraction method as well as the extracted feature sets is already illustrated in chapter 4, however, due to the difference of the sensors, the datasets of the former study are different from the datasets in this chapter. In chapter 4, the testing vehicle was equipped with Mobileye system, which can detect targets in front of the ego vehicle on different lanes. In addition, the lane mark can also be detected by the vision system. However, the testing vehicle in this study is limited by no such sensors, only the forward targets driving on the same lane with the ego vehicle can be detected.

The advantage of the data collected in this study over the datasets from the third party in the last chapter is that there are more data types included, e.g. steering wheel angle, brake and gas pedal data etc. And we can also equip eye-tracker in the experiment. Thus the feature extraction work should be reconsidered.

Vehicle dynamic feature

Vehicle dynamic feature refers to the feature that can describe the dynamic motion of the ego vehicle. The extracted features are:

- $yawRate_t$: yaw rate of the ego vehicle at time t.
- a_t: absolute acceleration of the ego vehicle at time t.

Driver behavior feature

Features which reflect driver behavior are extracted as follows:

- $brpd_t$: brake pedal pressure of the ego vehicle at time t.
- $acpd_t$: throttle opening angle of the ego vehicle at time t.
- stw_t: steer wheel angle of the ego vehicle at time t.
- $stwRate_t$: steering wheel angle rate of the ego vehicle at time t.

Combined feature

In section 5.4.2, two combined features, i.e. time to collision (TTC) and time-to-lane (TLC) crossing are considered. But the testing vehicle in this experiment cannot detect the lane mark, TLC is not possibly calculated. Only TTC is considered as the combined feature. The inverse is given by[21]:

$$\mathrm{TTC}_\mathrm{t}^{-1} = \frac{v_{ego}}{\Delta d} \tag{6.1}$$

[21] As it is mentioned in section 5.4.2, if the velocity of the ego vehicle and the front vehicle is the same, TTC becomes infinite.

where v_{ego} and Δd is the velocity of the ego vehicle and the distance with the front vehicle at time t, respectively. The motion of the ego vehicle relative to the front vehicle is illustrated in Figure 6.12.

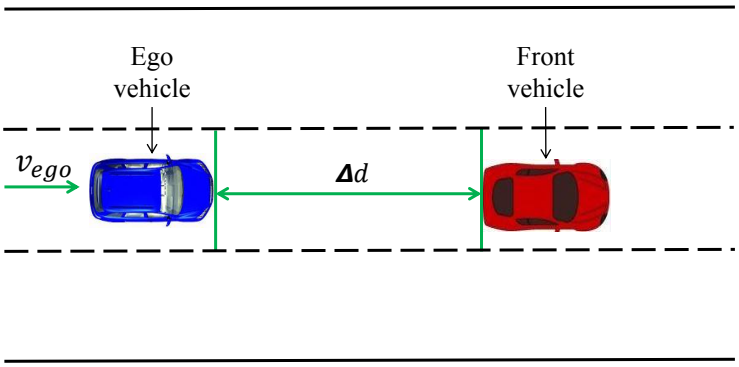


Figure 6.12: Illustration of the motion of the ego vehicle.

Time-window feature

The way of extracting time-window (TW) features is the same as it is described in section 5.4.2. TW between 1 second to 5 seconds are used. Statistical properties like mean, standard deviation, maximum, minimum and median of the features mentioned above are considered.

Frequency-domain features

Fast Fourier transform (FFT), which is already described in section 5.4.2, is used to transform the time-domain features into frequency-domain (Heckbert, 1995). After FFT, the maximum value of FFT coefficient within TW is chosen as the feature value (Mörchen, 2003). All the extracted features in this experiment can be found in Appendix Table C.1.

6.4 Method

6.4.1 Labeling lane-change dataset

It is known that the classification performance of the supervised learning models is highly depended on class labels. In section 4.4, two LC labeling methods are proposed, i.e. a gaze-based labeling (GBL) and a time-window labeling (TWL) method. For the real-road experiment, we also use these two labeling methods to label LC data samples.

On-board labeling lane-change event

In order to improve the efficiency of the off-line data labeling task, i.e. quickly querying LC events from huge collected raw data, an on-board LC event labeling method was used. In this method, an assist sitting on back of the testing vehicle was responsible for

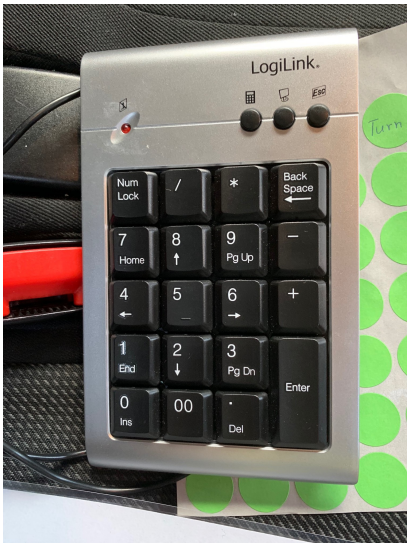


Figure 6.13: The keyboard which is used for manually labeling lane-change events on-board.

labeling LC events in real-time using a keyboard in Figure 6.13. When the participant prepared to make LC, he/she would perform certain pre-LC behaviors, e.g. mirror glancing, blind point checking, and then execute the actual maneuver. The task of the assist was to notice such behaviors and label the start time (the participant steers the steering wheel) by *pressing* and end time (LC finish) by *releasing* key 1[22]. This labeling process can be illustrated in Figure 6.14, where the assist presses key 1 from t_{start} to t_{end} to mark this lane-change event. This on-board labeling method is very useful and efficient to query each LC event from the big time series data. t_{end} functions as a very important criterion to extract LC and LK datasets, which will be detailed in the data labeling section.

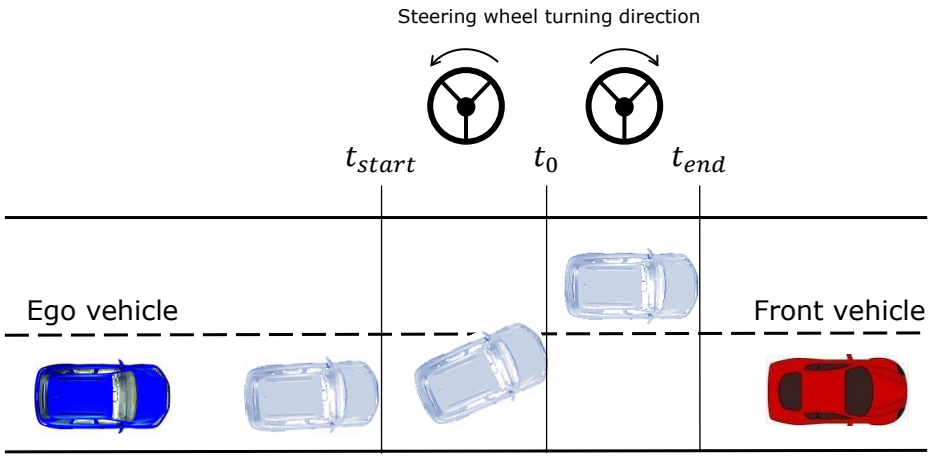


Figure 6.14: Illustration of the selected time of the on-board labeling task.

[22]Note that the keyboard was not only used for labeling driver LC behavior, but also for curve driving, turning at the intersection, for city scenarios etc. These behaviors are not the research scope of this study.

Table 6.2 lists the number of LC maneuvered by each participant. For the case that the participant attempted to make a LC but due to certain reasons, e.g. danger, he/she aborted the LC maneuver, it is counted as *Aborted LC*. Totally there are 232 LLC and 227 RLC cases observed. The LC cases that the driver did not use the turn signal 8 seconds before executing a LC are also marked. It indicates that this situation happened more in RLC case than LLC case. In addition, the classification of driving styles of each participant from section 6.2.2 is also listed in the right column.

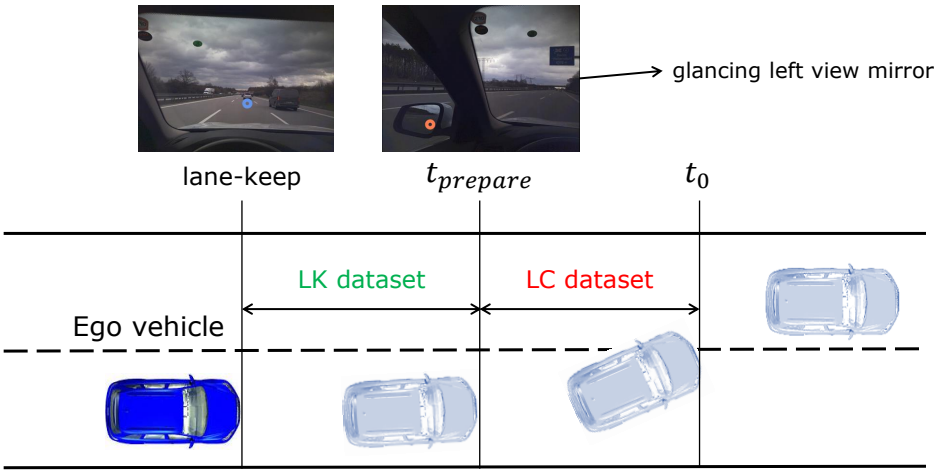
Table 6.2: The statistics of LC cases in the real-road experiment.

# Participant	LLC	RLC	Aborted LC	Driving style
1	19	19	0	High aggressive
2	18	18	3	Medium aggressive
3	23	23	3	Low aggressive
4	27	26	4	High aggressive
5	23	22	2	High aggressive
6	30	29	0	Low aggressive
7	16	16	1	Medium aggressive
8	30	30	1	High aggressive
9	28	26	4	High aggressive
10	18	18	1	High aggressive
Sum	232	227	19	-
No turn signal	12	21	-	-

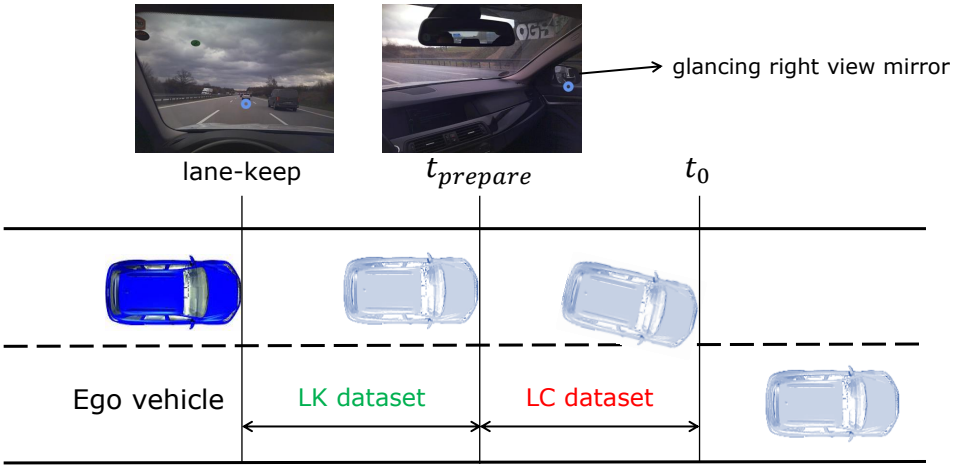
Gaze-based labeling method

The rule of the gaze-based labeling (GBL) method is the same as it is in section 4.4.1, where the moment $t_{prepare}$ is the critical moment. Take the LLC case for instance. As it is demonstrated in Figure 6.15a, $t_{prepare}$ is defined by the last mirror-glancing behavior by the participant before he/she uses the turn signal to indicate a LC. If the turn signal is observed before a mirror-glancing behavior, we choose the moment he/she indicates the turn signal as the $t_{prepare}$. It is the same rule for RLC case as it is depicted in Figure 6.15b. The LC case which will not be used for data labeling is when the driver does not use the turn signal 8 seconds before t_0. Totally there are 12 cases and 21 cases for LLC and RLC, respectively, which is illustrated in Table 6.2.

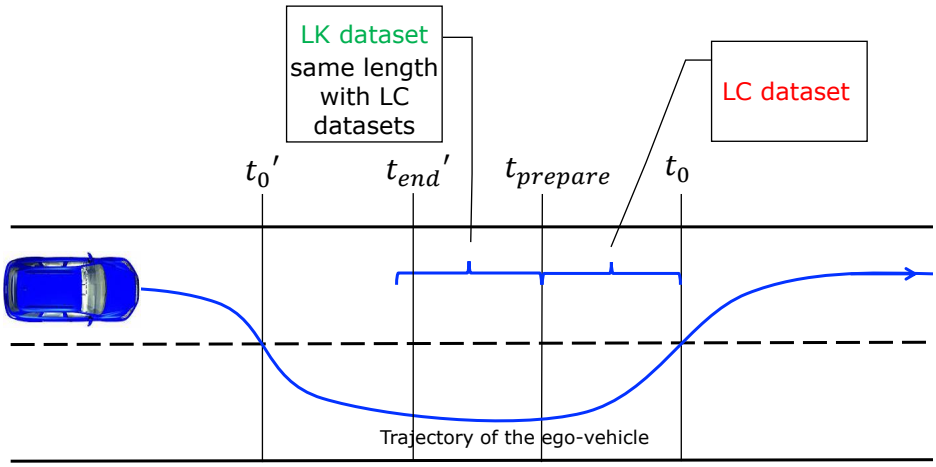
The statistics of $t_{prepare}$ in box plot is plotted in Figure 6.16, where the y axis is the duration between $t_{prepare}$ and t_0 in seconds. Then LC datasets can be labeled between $t_{prepare}$ and t_0. t_0 is defined as the moment that the wheel of the ego vehicle just crosses the dotted central line. In order to obtain a balanced dataset for training, lane-keep (LK) data samples are labeled as the equal number as the LC data samples.



(a) Gaze-based labeling method for LLC case. The number of labeled LC samples and LK samples is equal.



(b) Gaze-based labeling method for RLC case. The number of labeled LC samples and LK samples is equal.



(c) The case which is not suitable of using GBL method.

Figure 6.15: Demonstration of using the GBL method to label LC and LK datasets.

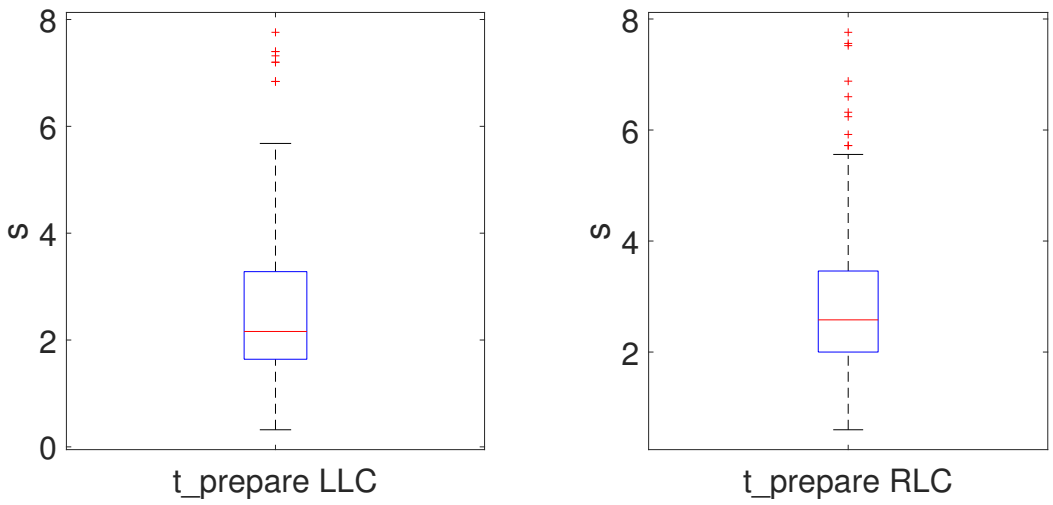


Figure 6.16: The box plot of $t_{prepare}$ ahead of t_0.

A special case we cannot use the GBL method is pictured in Figure 6.15c. That is: when $t_0 - t_{prepare} > t_{prepare} - t'_{end}$, where t'_{end} refers to the end of the last adjacent LC event. In other words, the difference between t'_{end} and $t_{prepare}$ should be greater than $t_{prepare}$ and t_0, otherwise it is not possible to extract equal amount of LK data samples as LC datasets.

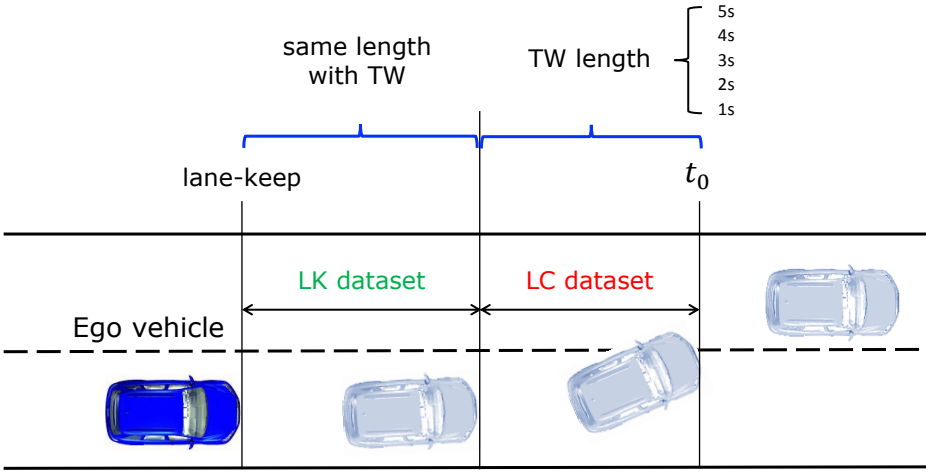
Time-window labeling method

The time-window labeling (TWL) method uses a fixed length time interval to label data samples. As we can see in Figure 6.17a, candidate TW, i.e. 5s, 4s, 3s, 2s, 1s are selected. Thus data before t_0 and within the selected TW are labeled as LC dataset. Right before LC dataset, the same amount of data are then labeled as LK samples for the need of obtaining balanced datasets. This rule applies to both LLC and RLC case.

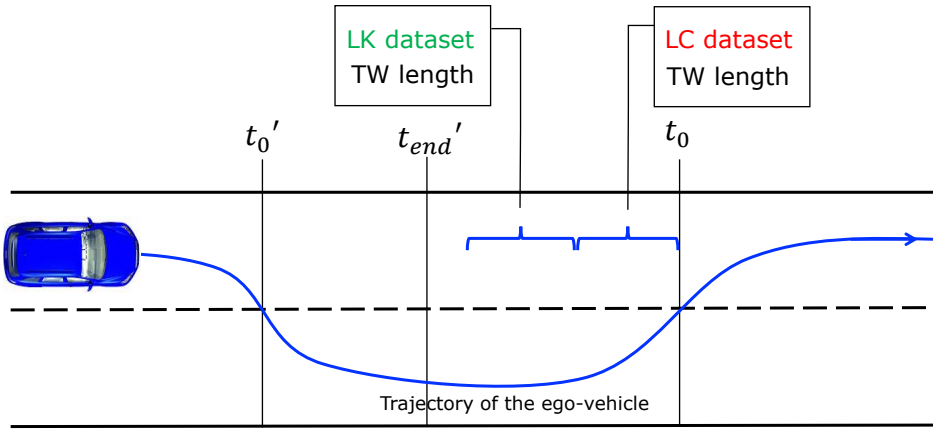
The criterion of choosing the largest TW for certain LC case follows what it is illustrated in Figure 6.17b:

$$\mathrm{TW} \leqslant \frac{t_0 - t'_{end}}{2} \tag{6.2}$$

where t_0 refers to the current LC event, and t'_{end} refers to the end of the last adjacent LC event. The reason of restricting the largest allowed TW is that once TW is too large, it is impossible to extract equal numbers of LK data samples as LC. If TW is greater than $t_0 - t'_{end}$, LC dataset cannot even be extracted. If choosing the smallest TW, i.e. TW = 1 s, it still cannot satisfy the above criterion, then this LC will be excluded for data extraction.



(a) An example of using TWL method to label LC and LK datasets. The number of labeled LC samples and LK samples is equal.



(b) The criterion of choosing the largest length of TW for certain LC case.

Figure 6.17: Demonstration of using the TWL method for both LLC and RLC case.

Data labeling result

Following the rules of the GBL and the TWL method, the numbers of the balanced training samples is illustrated in Table 6.3. What needs to be mentioned is, as we can see from the TWL method, the smaller the time-window is, the less the number of labeled samples. In addition, the number of the training samples by the GBL method is similar with the TWL method with 3 s time-window. The main difference of the GBL and the TWL is that the GBL method tends to label LC data case by case, however, the TWL method takes all the LC cases as the same by using a fixed time-window.

Table 6.3: The labeled training samples by the GBL and TWL method.

Labeling method	**Data type**	**Scenario LLC**	**Scenario RLC**
GBL	LC	12001	12971
	LK	12001	12971
TWL 5 s	LC	17750	17500
	LK	17750	17500
TWL 4 s	LC	15500	16400
	LK	15500	16400
TWL 3 s	LC	12600	13350
	LK	12600	13350
TWL 2 s	LC	9350	9350
	LK	9350	9350
TWL 1 s	LC	4950	4925
	LK	4950	4925

6.4.2 Feature selection

The same rule is applied to feature selection as it is mentioned in section 5.6.1, i.e. features whose p-value smaller than 0.05 at the same time Cohen's d larger than 0.8 are selected for model training. In statistical analysis, p-value < 0.05 represents statistical significance. Similarly, Cohen's $d > 0.8$ indicates high effect size level (J.Cohen, 1992). In addition, for the same feature with different time-windows, feature with a larger time-window is preferable because more information is included within larger time-window. For instance, if both $max\_yaw_t^5$ (5 s time-window) and $max\_yaw_t^3$ (3 s time-window) are qualified, $max\_yaw_t^5$ is selected.

The final selected features for both LLC and RLC case as well as by different labeling methods can be seen in Table 6.4 and Table 6.5 (features with ✗are the selected). The full scale of p-value and Cohen's d of all the extracted features are listed in Appendix Table C.2 and Table C.3. From the two tables we can see that the final selected features are different by labeling methods. Additionally, LLC and RLC scenario also show different feature selection results. This result is coinciding with the one we concluded in section 5.6.2 from the big data analysis, which emphasizes the importance of doing feature selection.

6.4.3 Training dataset

After feature selection, training datasets can be organized based on the selected features. Three kinds of training datasets are prepared for model training:

- Driving style dataset: training datasets are grouped by driving styles. In other words, data collected from the participants whose driving styles is in the same group, i.e. high aggressive, medium aggressive or low aggressive, are categorized together. The classified driving style of each participant is listed in Table 6.2.
- Personalized dataset: datasets are separated by each participant, which means that each participant has his/her individual training dataset. The aim of using the personalized datasets for training is to consider the individual driving style. As it is mentioned in chapter 4 that even two drivers are at similar aggressive level, they may show different temporary driving styles which can be represented by different speed choice as well as accelerating or braking behavior.
- Non-categorized dataset: all the datasets are directly grouped together without considering driving style and individuation. It is a super huge dataset.

Table 6.4: The feature selection results of different labeling methods for LLC scenario.

#	Feature		TWL				
		GBL	5 s	4 s	3 s	2 s	1 s
1	$yawRate_t$	–	–	–	✗	–	–
3	$\mathrm{TTC}_\mathrm{t}^{-1}$	✗	✗	✗	–	✗	✗
13	$std\_yaw_t^5$	✗	✗	✗	✗	✗	✗
18	$max\_yaw_t^5$	✗	✗	✗	✗	✗	✗
23	$min\_yaw_t^5$	–	✗	✗	✗	–	–
24	$min\_yaw_t^4$	–	–	–	–	✗	–
26	$min\_yaw_t^2$	✗	–	–	–	–	–
27	$min\_yaw_t^1$	–	–	–	–	–	✗
28	$med\_yaw_t^5$	–	✗	✗	✗	–	–
30	$med\_yaw_t^3$	–	–	–	–	✗	–
31	$med\_yaw_t^2$	–	–	–	–	–	✗
32	$med\_yaw_t^1$	✗	–	–	–	–	–
33	$mean\_a_t^5$	✗	✗	✗	✗	✗	–
34	$mean\_a_t^4$	–	–	–	–	–	✗
38	$std\_a_t^5$	✗	✗	✗	✗	✗	✗
43	$max\_a_t^5$	✗	✗	✗	✗	✗	–
48	$min\_a_t^5$	–	✗	✗	✗	–	–
52	$min\_a_t^1$	–	–	–	–	–	✗
53	$med\_a_t^5$	–	✗	✗	✗	–	–
54	$med\_a_t^4$	–	–	–	–	✗	–
58	$mean\_brpd_t^5$	–	✗	✗	✗	✗	–
60	$mean\_brpd_t^3$	✗	–	–	–	–	–
61	$mean\_brpd_t^2$	–	–	–	–	–	✗
113	$std\_stw_t^5$	✗	✗	✗	✗	✗	✗
118	$max\_stw_t^5$	✗	✗	✗	✗	✗	✗
123	$min\_stw_t^5$	–	✗	✗	✗	–	–
125	$min\_stw_t^3$	–	–	–	–	✗	–
126	$min\_stw_t^2$	✗	–	–	–	–	–
127	$min\_stw_t^1$	–	–	–	–	–	✗
128	$med\_stw_t^5$	–	✗	✗	✗	–	–
130	$med\_stw_t^3$	–	–	–	–	✗	–
131	$med\_stw_t^2$	–	–	–	–	–	✗
132	$med\_stw_t^1$	✗	–	–	–	–	–
133	$mean\_stwRate_t^5$	–	✗	✗	✗	–	–
134	$mean\_stwRate_t^4$	✗	–	–	–	✗	✗
163	$max\_F\_a_t^5$	✗	✗	✗	✗	✗	✗
168	$max\_F\_brpd_t^5$	✗	✗	✗	✗	–	✗
183	$max\_F\_stwRate_t^5$	–	✗	✗	✗	✗	✗

Table 6.5: The feature selection results of different labeling methods for RLC scenario.

#	Feature	GBL	TWL				
			5 s	4 s	3 s	2 s	1 s
1	$yawRate_t$	–	✗	✗	✗	–	–
3	$\mathrm{TTC}_\mathrm{t}^{-1}$	✗	–	–	–	✗	✗
13	$std\_yaw_t^5$	✗	✗	✗	✗	✗	✗
18	$max\_yaw_t^5$	✗	✗	✗	✗	✗	–
19	$max\_yaw_t^4$	–	–	–	–	–	✗
23	$min\_yaw_t^5$	–	✗	✗	✗	–	–
24	$min\_yaw_t^4$	–	–	–	–	✗	–
26	$min\_yaw_t^2$	✗	–	–	–	–	✗
28	$med\_yaw_t^5$	–	✗	✗	✗	–	–
29	$med\_yaw_t^4$	✗	–	–	–	✗	–
31	$med\_yaw_t^2$	–	–	–	–	–	✗
33	$mean\_a_t^5$	✗	✗	–	✗	✗	✗
34	$mean\_a_t^4$	–	–	✗	–	–	–
38	$std\_a_t^5$	✗	✗	✗	✗	✗	✗
43	$max\_a_t^5$	✗	✗	✗	✗	✗	–
45	$max\_a_t^3$	–	–	–	–	–	✗
48	$min\_a_t^5$	–	✗	✗	✗	–	–
50	$min\_a_t^3$	✗	–	–	–	–	–
53	$med\_a_t^5$	–	✗	✗	✗	–	–
55	$med\_a_t^3$	–	–	–	–	✗	–
56	$med\_a_t^2$	✗	–	–	–	–	–
58	$mean\_brpd_t^5$	✗	✗	✗	–	✗	–
59	$mean\_brpd_t^4$	–	–	–	–	–	✗
113	$std\_stw_t^5$	✗	✗	✗	✗	✗	✗
118	$max\_stw_t^5$	✗	✗	✗	✗	✗	✗
123	$min\_stw_t^5$	–	✗	✗	✗	–	–
124	$min\_stw_t^4$	✗	–	–	–	–	–
125	$min\_stw_t^3$	–	–	–	–	✗	–
127	$min\_stw_t^1$	–	–	–	–	–	✗
128	$med\_stw_t^5$	–	✗	✗	✗	–	–
130	$med\_stw_t^3$	✗	–	–	–	✗	–
132	$med\_stw_t^1$	–	–	–	–	–	✗
133	$mean\_stwRate_t^5$	–	✗	✗	✗	–	–
134	$mean\_stwRate_t^4$	✗	–	–	–	✗	–
136	$mean\_stwRate_t^2$	–	–	–	–	–	✗
163	$max\_F\_a_t^5$	✗	✗	✗	✗	✗	✗
168	$max\_F\_brpd_t^5$	✗	✗	✗	✗	✗	✗
169	$max\_F\_brpd_t^4$	–	–	–		–	–
183	$max\_F\_stwRate_t^5$	✗	✗	✗	✗	✗	✗

6.5 Evaluation result

In this section, evaluation is made from different aspects, i.e. model selection, training datasets comparison, labeling method comparison as well as the real-time prediction test. The method used for evaluation adopts the same rule as it is used in section 4.6 where ROC curves and AUC values are as the metrics. Similar methods can be found in McCall et al. (2007), Liebner et al. (2013), Peng et al. (2015), Doshi and Trivedi (2009), and Lethaus, Baumann, et al. (2013).

6.5.1 Model and dataset comparison

In this section, three machine learning models proposed in section chapter 4 are evaluated. They are lane-change Bayesian network with a Gaussian mixture model (LCBN-GMM), SVM and naive Bayes (NB) model. LCBN-GMM has the same structure as it is detailed in section 4.5.1. The prior distribution of NB is set as Gaussian. For SVM, the kernel function used to generate the decision boundary is modeled as a Gaussian kernel. The above three models are trained by the same training datasets mentioned in the last section and thus we can compare the classification performance. The performance of classifying LC and LK data samples by each model is listed in Table 6.6, where the results are after 10-fold cross-validation. The corresponding ROC curves can be found in Appendix Figure C.3.

Table 6.6: The AUC values of LCBN-GMM, SVM and NB trained by different datasets using GBL method .

Scenario	Training dataset	LCBN-GMM	SVM	NB
LLC	Driving style	0.9659	**0.9962**	0.8340
	Personalized	0.9877	**0.9970**	0.9247
	Non-categorized	0.8215	0.9900	0.7825
RLC	Driving style	0.9673	**0.9987**	0.8190
	Personalized	0.9892	**0.9981**	0.9203
	Non-categorized	0.8577	0.9964	0.7611

The results can be discussed as follows.

- Comparison of models: LCBN-GMM and SVM perform much better than NB in both LLC and RLC scenarios trained by each dataset. SVM performs slightly better than LCBN-GMM (AUC values in bold are the best among each model).
- Comparison of training datasets: using the personalized training dataset, models can achieve better performance than using the driving style dataset and much better than the non-categorized dataset. The only exception is for SVM in RLC

scenario where using the driving style dataset it performs slightly better than using the personalized dataset. The non-categorized datasets are the worst for all the models. In addition, we find that SVM is not influenced heavily by training datasets as LCBN-GMM and NB.

In conclusion, NB and the non-categorized datasets are not qualified since their performances are far worse than other combinations and thus would not be taken into account for further comparison.

6.5.2 Labeling method comparison

In model comparison and training dataset comparison, NB and the non-categorized are out. This section we use LCBN-GMM and SVM to compare the performance of using the driving style dataset and the personalized dataset by different labeling methods. The AUC values of LCBN-GMM and SVM are listed in Table 6.7 and Table 6.8, respectively. The corresponding ROC curves can be seen in Appendix Figure C.4, where the results are after 10-fold cross-validation.

Table 6.7: The AUC values of LCBN-GMM trained by different datasets using both GBL and TWL method.

Scenario	Labeling method	Driving style	Personalized
LLC	**GBL**	0.9659	**0.9877**
	TWL 5 s	0.9618	0.9867
	TWL 4 s	0.9578	0.9843
	TWL 3 s	0.9054	0.9767
	TWL 2 s	0.8856	0.9699
	TWL 1 s	0.8929	0.9706
RLC	**GBL**	0.9673	**0.9892**
	TWL 5 s	0.9260	0.9853
	TWL 4 s	0.9220	0.9873
	TWL 3 s	0.9050	0.9806
	TWL 2 s	0.9274	0.9823
	TWL 1 s	0.9044	0.9608

Table 6.8: The AUC values of SVM trained by different datasets using both GBL and TWL method.

Scenario	Labeling method	Driving style	Personalized
LLC	GBL	0.9962	0.9970
	TWL 5 s	0.9970	0.9972
	TWL 4 s	0.9964	0.9964
	TWL 3 s	0.9971	**0.9975**
	TWL 2 s	0.9919	0.9947
	TWL 1 s	0.9923	0.9941
RLC	**GBL**	**0.9987**	0.9981
	TWL 5 s	0.9982	0.9984
	TWL 4 s	0.9982	0.9975
	TWL 3 s	0.9966	0.9964
	TWL 2 s	0.9964	0.9948
	TWL 1 s	0.9896	0.9919

From the two tables we find the following results:

- Comparison of labeling method: except for SVM in LLC scenario where using the TWL method with 3 s time-window, the best AUC values are achieved by using the GBL method.
- Comparison of training datasets: except for SVM in RLC scenario, the best AUC values are all from the personalized datasets.

In conclusion, training the models using the personalized datasets labeled by the GBL method is the recommended combination. Thus for the final real-time LC prediction test, we test both LCBN-GMM and SVM using the personalized datasets with the GBL method.

6.5.3 Real-time performance evaluation

In order to evaluate the real-time LC prediction performance of LCBN-GMM and SVM, data from the entire drive of each participant are fed to the off-line trained model, i.e. by the personalized training datasets with the GBL method, to test the real-time LC prediction performance. One thing needed to be mentioned is that because of the sensor failure during experiment, part of the data collected from participant # 4 are missing and thus will not be used for testing.

Unlike the off-line training and testing task which only performs classification given the labeled LC and LK data samples, the real-time prediction task is much more

challenging. The reason is that in the real-time prediction test the whole data in time series are tested. It includes significant amount of untrained data samples and the majority of them are LK data samples. Thus, reducing false alarm and at the same time achieving high precision is a big challenge.

Prediction fusing eye-tracking signal

In the real-time prediction test, the eye-tracking signal is fused into the prediction algorithm, which is defined in terms of *glancing_ratio* by

$$\text{glancing\_ratio} = \frac{\text{mirror glancing duration}}{\text{TW}} \tag{6.3}$$

where the mirror-glancing duration is the total amount of time by the participant to glance at the left view mirror and the left window (for RLC case is the right view mirror and the right window). The time-window here is chosen as 5 s. If the participant does not check the left/right view mirror/window, this ratio should be zero. The non-zero histogram of glancing_ratio for both LLC and RLC is plotted in Figure 6.18.

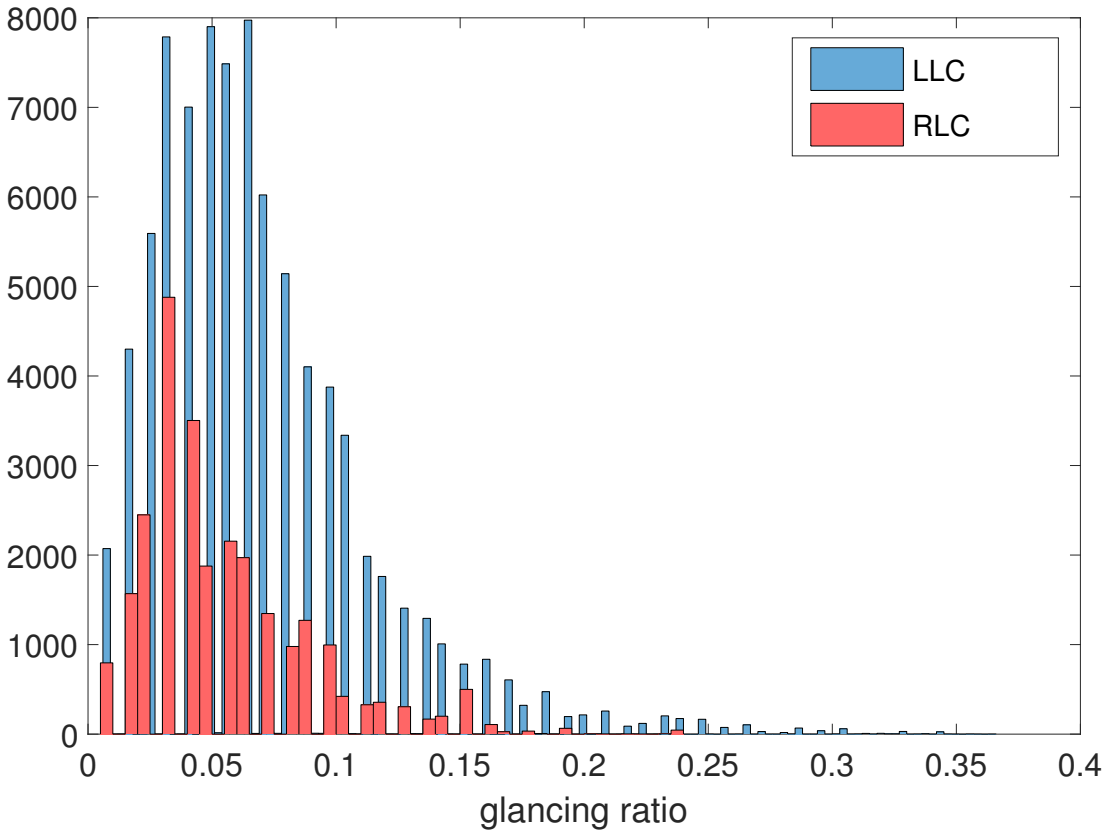


Figure 6.18: Non-zero histogram of mirror glancing ration for LLC and RLC scenario.

Algorithm

The algorithm of the real-time LC prediction is described as follow:

Algorithm 1 Algorithm of the real-time driver LC behavior prediction.

```
Input feature signals at time t.
if P(LLC) > P(RLC) then
    if P(LLC) ≥ ϵ & glancing_ratio ≥ ξ_left then
        LLC
    else Lane-keeping
    end if
else if P(LLC) < P(RLC) then
    if P(RLC) ≥ ϵ & glancing_ratio ≥ ξ_right then
        RLC
    else Lane-keeping
    end if
else
    Lane-keeping
end if
```

where, ϵ is the decision threshold which is set as 0.9 for LCBN-GMM and 0 for SVM[23]. ξ_{left} and ξ_{right} are set as 0.05 and 0.03, respectively.

Evaluation

Three metrics, i.e. precision, recall and predicted time ahead of t_0, are used to evaluate model performance. Precision and recall are defined as:

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} \\ Recall &= \frac{TP}{TP+FN} \end{aligned} \tag{6.4}$$

where TP, FP and FN are true positives, false positives and false negatives, respectively. For the case of LC behavior prediction, TP, FP and FN is defined as follows and can be illustrated in Figure 6.19:

- TP: if the algorithm reports LLC/RLC intent and within the next 8 seconds[24] the driver does execute a LLC/RLC maneuver, then this prediction is counted as a TP otherwise it is regarded as a FN.

[23]Since Bayesian network is a probability model, this threshold means that the model has 90% confidence for the prediction. For SVM, it simple classifies data by the decision boundary, where the positive class is greater than zero and negative class is smaller than zero

[24]The reason of choosing this threshold is that from Figure 6.16 we can see that the majority of $t_{prepare}$ are within the range of 0 - 6 s and the maximum is nearly 8 s. Thus the maximum $t_{prepare}$, i.e. 8 s, is selected as the threshold. For reference, in Leonhardt, Pech, and Wanielik (2018) the threshold is set as 10 seconds, which means a larger forgiveness.

- FP: if the driver executes LLC/RLC maneuver but in the previous 8 seconds the algorithm did not report the correct LLC/RLC or there is even no report at all, then this case is counted as a FP.

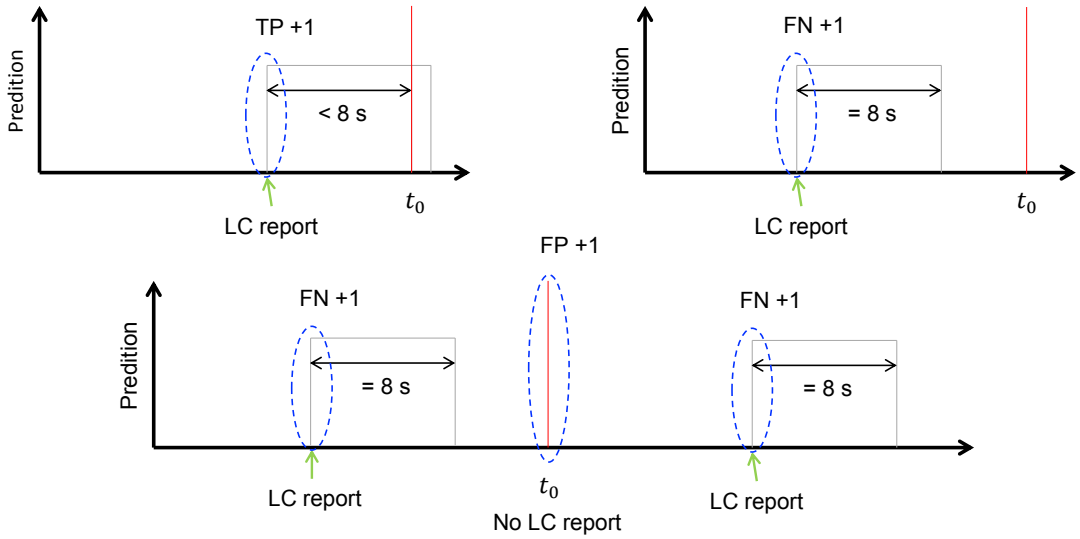


Figure 6.19: Illustration of TP, FN and FP of the real-time prediction.

Precision and recall are counted separately for LLC and RLC case. For comparison, we also perform the real-time prediction test for the case of without fusing eye-tracking signal as well as only using eye-tracking signal. The threshold setting is the same as it is in Algorithm 1. The final results are illustrated in Table 6.9, where the value in bond represents the best performance given metrics.

Table 6.9: The real-time LC prediction result performed by LCBN-GMM and SVM.

	Metric	**Fusion of eye-tracking**		**Only**		**Only**
		LCBN-GMM	SVM	LCBN-GMM	SVM	**eye-tracking**
LLC	Precision	89.42%	93.51%	93.03%	**98.21%**	92.23%
	Recall	**56.34%**	54.09%	26.71%	31.78%	42.59%
	Prediction time	3.1 s	3.3 s	4.7 s	**5.1 s**	3.6 s
RLC	Precision	71.21%	72.39%	94.40%	**97.61%**	62.78%
	Recall	48.70%	**82.90%**	26.95%	31.78%	63.34%
	Prediction time	2.6 s	2.6 s	4.7 s	**4.9 s**	2.8 s

From the table we find the following results:

- Comparison of LCBN-GMM and SVM fusing eye-tracking signal: SVM achieves slightly higher precision in both LLC and RLC scenario and much better recall value in RLC scenario. The recall of LCBN-GMM in LLC scenario is slight higher than SVM. In addition, SVM can predict LC behavior slight earlier than LCBN-GMM.
- Comparison of fusion and without fusion of eye-tracking signal: for LLC, the prediction of both LCBN-GMM and SVM fusing eye-tracking signal are slightly lower (around 5%) than without fusing eye-tracking signal. The reason is that by fusing eye-tracking signal, an additional condition is added to predict a LC report, which means the condition of reporting a true LC is stricter. This can increase the chance of missing-prediction. The missing-prediction case is more often in RLC case, which leads to a lower precision. The possible reason is that fusing eye-tracking signal is highly depended on if the driver performs mirror-glancing behavior before LC. And in RLC case, based on the statistics in Table 6.2, the chance of executing a LC without mirror-glancing behavior in RLC scenario is nearly 50% more than LLC scenario. This is why the precision of predicting RLC by fusing eye-tracking signal is much lower than LLC.

  However, the benefit of more strict condition is highly increasing the recall. We can see that the recall values of the fusing eye-tracking signal of both models are much higher than without fusing eye-tracking signal, which are nearly 30% for LCBN-GMM and 20% for SVM (in RLC case for SVM is more than 50%). a Higher recall value represent a lower false alarm level, which plays an essential role in the practical implementation of ADASs.
- Comparison of fusion eye-tracking and only using eye-tracking signal: prediction of LLC by only using eye-tracking signal can achieve relatively higher precision with earlier prediction time, however, the recall is more than 10% lower than fusion method. For RLC, only using eye-tracking signal suffers a low prediction. This is also in consistent with the result that glance pattern is promising LC predictors for certain types (Beggiato et al., 2018), implying that it does not apply to all the cases. Thus, it can be concluded that only using eye-tracking signal to predict driver LC behavior is not recommended.

In conclusion, balancing the precision and the recall, SVM fusing eye-tracking signal can achieve the best performance in LLC scenario. For RLC scenario, among all the methods, it is difficult to balance the precision and the recall, which indicates the prediction of RLC is more challenging than LLC. The precision of SVM fusing eye-tracking signal decreases significantly compared with its performance in LLC scenario. But it could achieve much higher recall values than other counterparts, which means the false alarm is very low.

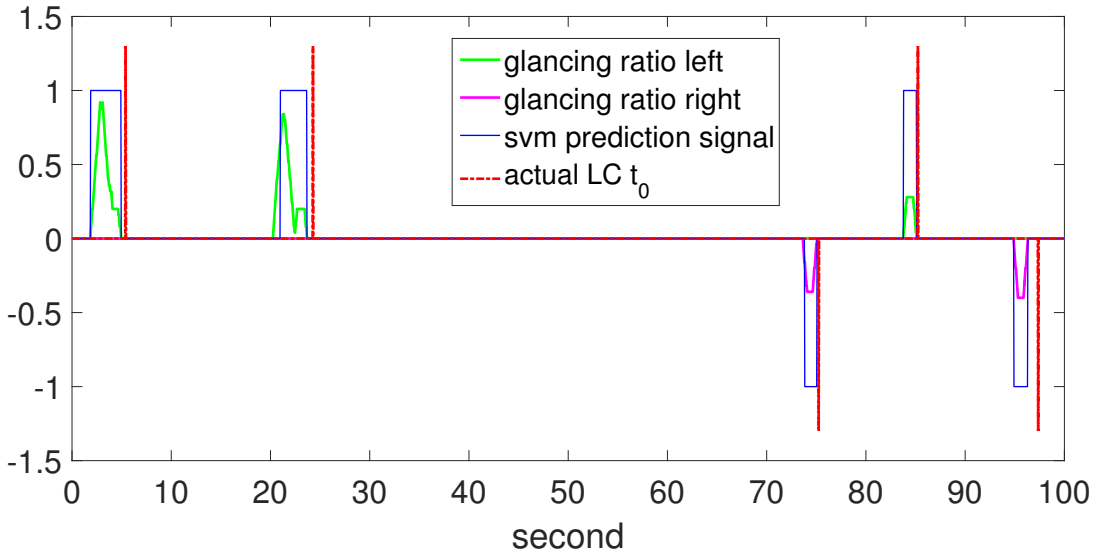


Figure 6.20: An example of the real-time prediction of LC by SVM.

Figure 6.20 gives an example of the real-time prediction of LC by SVM during 100 seconds. The red dotted line represents the moment t_0 in Figure 6.19. The positive values are for LLC and negative values for RLC. The blue line is the SVM prediction signal for prediction of LLC as 1 and RLC as -1. We can see that SVM can predict LC several seconds before t_0 for both LLC and RLC. The green and the pink line are the glancing_ratio in equation (6.3) for left and right side, respectively. It depicts the change of glancing_ratio before and after LC, and it shows that glancing_ratio has a dramatic increase before each LC.

6.6 Summary

The work presented in this chapter is the combination of chapter 4 and chapter 5, aiming to design a comprehensive framework of prediction of driver LC behavior. The content includes experimental design, data processing, feature selection, model selection and evaluation. Although the condition of the experiment conducted in this chapter is not exactly the same as it is in chapter 4 and chapter 5, the methods are mainly the same.

In order to evaluate the methods proposed in the last chapters, a real-road experiment was conducted on highway in Berlin. Totally 12 participants joined in the experiment with 3 hours' drive for each person (40 min on highway). During the experiment, the participants were asked to wear the SMI eye-tracking glasses to monitor their gaze behavior. A semantic gaze mapping method was used to capture the AoIs of the participant during the driving task. Synchronization work was done between CAN Bus and the eye-tracker so that data collected from different sensors are synchronized. Furthermore for the off-line data processing, all the data were re-sampled at 25Hz by using a interpolation method. In preparing for training datasets, besides the driving

style dataset and the non-categorized training datasets, a new training dataset, i.e. the personalized dataset, was added for comparison.

The feature selection method used in this chapter is the same as it is presented in chapter 5, where p- value and Cohen's d were chosen as the metrics. The result shows that the items of the final selected features are different by LC scenarios as well as by labeling methods, which implies that feature selection is necessary before model training.

The ML models being tested are the same as in chapter 4, i.e. LCBN-GMM, SVM and naive Bayes. Data are labeled using both the GBL and the TWL method. The evaluation applies to the same metrics used throughout this dissertation, i.e. ROC curves and AUC values. Result shows that the personalized training dataset with the GBL method is the best combination for model training. Comparison of the three models, LCBN-GMM and SVM outperform NB, thus NB is out of the final real-time prediction test.

In the final test, we mimic the real-time prediction scenario by feeding data from the entire drive (participant by participant) in time series to LCBN-GMM and SVM to evaluate their real-time prediction performance. Comparison was also made between fusing and without fusing eye-tracking signal as well as only using eye-tracking signal for prediction. Result shows that in LLC scenarios, SVM fusing eye-tracking signal can achieve the best performance. It can predict driver LC behavior 3.3 s before actual LC with precision 93.51% and recall 54.09%. However, it represents different picture in RLC scenario. It is difficult to balance the precision and the recall, which in other words indicates that predicting driver LC behavior in RLC scenario is more difficult than in LLC scenario.

The limitation of this study is that because there is no detection sensors equipped in our testing vehicle, no contextual traffic is modeled. Thus, we cannot analyze the influence of different contextual traffic on model performance. Especially for the fact that LCBN-GMM performs better than SVM in chapter 4 where the contextual traffic is modeled, however, in this chapter SVM slightly outperforms LCBN-GMM without modeling the contextual traffic. In addition, since the case of the driver making LC without glancing mirror happens more often for RLC than LLC, which makes it more challenging to predict RLC. If the contextual traffic can be modeled, it could be beneficial for the prediction. The influence of the contextual traffic on driver LC behavior as well as model performance will be further discussed in next chapter.

7

Discussion and outlook

This chapter summarizes the final conclusions throughout this dissertation and gives an outlook on the research of prediction of driver lane-change behavior.

7.1 Overall conclusion

In this dissertation, a systematic framework of prediction of driver lane-change (LC) behavior is proposed, including experimental design, data processing, data labeling as well as model performance evaluation. Several issues reading to the framework are scattered in three studies, i.e. chapter 4 - chapter 6. The pipeline of the research method is in the form of *question and answer* and can be summarized as: *What did the related works do?*, *What can be improved?*, *What is our limitation and how to overcome it?*. The conclusions from different aspects are discussed in the following sections.

7.1.1 Modeling

Modeling is the first issue to be considered for implementation. The main modeling work that is included in this dissertation is regarding to modeling of driving contextual traffic and machine learning (ML) models.

Contextual traffic

It has been concluded in many research that contextual traffic is very import in driver behavior related studies (Oliver and Pentland, 2000; Wahle et al., 2000; McGehee et al., 2002; Wahle et al., 2002). In this dissertation, the concept of the contextual traffic is related to the relationship between the subject vehicle and its surrounding vehicles. In the case of driver LC behavior, the contextual traffic could impact on driver decision-making process. In more complex contextual scenario, it takes longer time for

the driver to prepare for a LC maneuver. The contextual traffic also has influence on driver gaze behavior which is a very important indicator for prediction of driver LC behavior.

In order to dynamically modeling the contextual traffic, a cell-grid mapping method is used inspired by Do et al. (2017), Nilsson, Silvlin, et al. (2016), and Kasper et al. (2012). So far, this modeling method is only implemented on highway roads since the road situations are less complex than in city. In addition, this cell-grid modeling method is also depended on how many lanes it has on highway road and the sensor level of the experimental vehicle. For example, the number of cells is different for two-lane highway and three-lane highway. And if there are only front detection sensors, e.g. camera or LiDAR, then only the front contextual traffic can be modeled.

In chapter 4, where the experiment was conducted in a driving simulator, it is assumed that all the necessary sensors are well equipped and thus we could model both the front and the back contextual traffic. In chapter 5, where the ego-vehicle had only front view camera installed, thus only the front contextual traffic was modeled. However in chapter 6, limited by no detection sensors, the real-road experiment was conducted without contextual traffic modeling. The absence of modeling contextual traffic is also one reason that leads to the relative poor prediction performance of right LC (RLC) in comparison to left LC (LLC). It is also indicated in Beggiato et al. (2018) that by monitoring vehicle environment it can allow for better prediction performance, e.g. the number of driving lanes available, the possibility of changing the lane (i.e. traffic density on the target lane) and the presence of a slower leading vehicle etc.

Machine learning models

Supervised learning models are mainly focused on in this dissertation. The two main branches of supervised learning models are generic model and discriminant model. For the purpose of covering both of the two branches, an easier generic model of Naive Bayes (NB), and a more complex generic model of Bayesian networks (BN) as well as discriminant model of support vector machine (SVM) are implemented for model comparison. BN is modeled specially for LC case and is incorporated with a Gaussian mixture model in terms of LCBN-GMM.

By comparing the classification performance of these three models in both chapter 4 (a driving simulator based study) and chapter 6 (a real-road experiment based study), it can be concluded that LCBN-GMM and SVM outperforms NB. The poor performance of NB may be caused by the conditional independent assumption between features. This is the biggest downside of NB. However, the comparison of LCBN-GMM and SVM in chapter 4 and chapter 6 indicates different results. In the driving simulator based experiment, LCBN-GMM performs better than SVM, however, in the real-road based experiment SVM performs slight better than LCBN-GMM. This controversial result may be coursed by the different experimental condition. Because the real-road experiment

was conducted without contextual traffic modeling, this makes LCBN-GMM not perform at its best as it could. In addition, because SVM is a discriminant model, which does not care how the data are generated but simply tries to find the best classification boundary to classify data samples, it is less sensitive to the quality of the dataset compared with generic models.

Although it is still an open question which really performs better, it can be concluded that both LCBN-GMM and SVM could achieve high performance using the proposed method in this dissertation. In the real-road experiment, the best AUC values of LCBN-GMM is nearly 0.99 for both LLC and RLC scenario, and SVM could achieve slight over 0.99 in both LLC and RLC scenario. Based on the study by Fan, Upadhye, and Worster (2006), AUC > 0.97 can be regarded as a very good classification performance.

7.1.2 Feature selection

Feature selection is crucial for ML models. It can not only speed up the learning process but also can improve model performance (Kira and Rendell, 1992). A statistical method is proposed in this dissertation to select the most contributive features based on the metrics of p-value and Cohen's d. Instead of ranking features to a specific algorithm (Geng et al., 2007), this feature selection method tends to avoid heuristic search and thus the selected features can be used by all the ML models.

From the big data analysis in chapter 5, it can be concluded that the importance of the features is varying by different contextual scenarios. In other words, to predict driver LC behavior in different scenarios, the selected strong features are probably different as well. Frequency-domain features, which are rarely used in driver behavior related research, are also promising features, with nearly at least one feature being selected as the strong feature in each scenario. In addition, in some contextual scenarios, the commonly assumed and being used features regarding to vehicle lateral movement, e.g. lateral acceleration and time-to-lane cross (TLC), do not show statistical significance and thus would not be regarded as strong features. This result indicates that feature selection should be done in a systematic way rather than only based on the empirical knowledge.

After feature selection, 95 features are reduced to around 10 features for each scenario. Finally, ML modes are tested with the use of the selected features and all the features without selection. The result suggests that using the selected features, the performance of ML models shows increase in different levels (maximum 10% increase) compared with using all the features. The exception is NB model, whose performance can hardly be improved. Considering the fact that using the selected features (nearly only 10 features), ML models (except for NB) can still achieve the same performance or even have improvements, it can be concluded that it is more efficient and effective to use the selected features than use all the features without feature selection.

7.1.3 Evaluation

The main metrics used for evaluating classification performance are ROC curve and AUC value, which are widely used in many related works like in McCall et al. (2007), Liebner et al. (2013), Peng et al. (2015), Doshi and Trivedi (2009), and Lethaus, Baumann, et al. (2013). We also use precision and recall as the metrics to evaluate the real-time prediction performance. Evaluations are made in the following aspects.

Training dataset and Labeling method

In the last section we have discussed the importance of feature selection. Actually, the quality of the training datasets can also impact on the performance of the ML models.

Although feature selection is considered in many research to improve model performance, few research studied the influence of training dataset on prediction performance. As we mentioned the influence of driving style on driver LC behavior, it mainly affects the quality of the labeled training datasets. To improve the quality of the training datasets, datasets are separated into different categories to test their performance, i.e. driving style datasets (datasets are grouped based on driving styles), personalized datasets (data collected from each participant is a single group) and non-categorized datasets (a huge dataset without any classification).

Another issue can have influence on the quality of the training datasets is the labeling method. The most popular way to label LC and LK datasets by the related research is the time-window labeling (TWL) method, with the assumption that data within certain time-window (TW) ahead of the LC maneuver can be labeled as LC samples (Mandalia and Salvucci, 2005; Doshi and Trivedi, 2009; Lethaus, Baumann, et al., 2013; Doshi and Trivedi, 2008; Morris, Doshi, and Trivedi, 2011). The limitation of this labeling method is without considering the differences between drivers and LC cases. Because driver LC behavior is very driver and LC case specific. Considering these differences, a gaze-based labeling (GBL) method is proposed with the use of eye-tracker. The GBL method considers the mirror glancing behavior of the driver before an actual LC. All the training datasets are labeled by both the GBL and the TWL method for evaluation.

Result shows that using the driving style training dataset is much better than the non-categorized training dataset. The personalized training dataset performs slightly better than the driving style dataset. This result implies that the effect of driving style on model performance does exist. Thus, preparing for the training datasets by individual driver is preferable. What it needs to be mentioned here is that the personalized training datasets may suffer from being short of training samples. This is the reason that in chapter 4 there are no personalized training datasets being considered. For inadequate training samples, organizing dataset by driving styles is recommended since it outperforms the non-categorized dataset. In addition to the comparison between labeling method, the GBL achieves the best performance for both LCBN-GMM and

SVM almost in all scenarios, except for SVM in LLC scenario where the TWL performs slightly better than the GBL with 3 s time-window.

In conclusion, the best combination used for model training is the personalized training datasets with the GBL method.

Real-time prediction performance

All the work presented above, i.e. modeling, feature selection, model selection as well as training dataset and labeling method selection, is to pave the way to evaluate the real-time prediction performance. Both LCBN-GMM and SVM trained by the personalized datasets using the GBL method are tested. In the real-time test, we feed all the data (on highway) in time series to the two trained models and to see whether they can correctly predict LC maneuvers or not. Comparison is made between fusing and without fusing eye-tracking signal in the prediction algorithm as well as only using eye-tracking signal without ML models for prediction. Precision and recall are the two metrics for evaluation. Precision represents the level of how many LC behaviors can be correctly predicted, whereas recall reflects the false alarm level.

Comparison between LCBN-GMM and SVM both fusing eye-tracking signal, SVM performs slightly better than LCBN-GMM for LLC case and much better for RLC case by achieving significant higher recall vales. Thus, SVM fusing eye-tracking signal is chosen for further comparison.

For LLC case by SVM fusing eye-tracking information, although the precision decreases slightly in comparison to without fusing eye-tracking signal, the recall can be improved significantly. The reason of fusing eye-tracking makes the precision slightly decreases is that an additional threshold makes it stricter to report a LC. But it can also reduce false alarm significantly. Given that fusing eye-tracking signal the models can still achieve high precision (93.5%) and with significant recall increase, it can be concluded that fusing eye-tracking signal can improve the prediction performance. And by comparing fusing eye-tracking signal and only using eye-tracking signal for prediction, we found that SVM fusing eye-tracking signal outperforms only using eye-tracking signal in both precision and recall.

For RLC case, however, it shows different picture. Only using eye-tracking signal for prediction could lead to poor performance. And it is difficult to balance precision and recall by either fusing eye-tracking signal or not. For SVM without fusing eye-tracking signal, it can achieve high precision but suffer from low recall, which means high false alarm. Fusion of eye-tracking signal, the precision of SVM is going down significantly despite the recall also increases significantly. This result indicates that predicting driver right LC behavior is more difficult than left LC behavior. The possible reason is that we can observe less evidence in RLC case than LLC case. Research found that most lane changes are to the left with a mean duration of over 11 seconds than to the right which has a mean of 6.6 seconds (Lee, Olsen, Wierwille, et al., 2004). LLC cases are mainly

for the purpose of overtaking a slow leading vehicle. Especially based on traffic rules in Germany, it is not allowed to execute a RLC for overtaking. In an overtaking process, the driver has to do more preparation to execute a LLC and thus gives more clues that can be used for prediction. For example, turn signal used in LLC case is 13% more frequently than RLC case, and the chance of glancing at the left view mirror for LLC is also higher than glancing at the right view mirror for RLC (Lee, Olsen, Wierwille, et al., 2004). It was hypothesized that the driver executing RLC has often just passed a slow leading vehicle and therefore has a greater degree of situational awareness than the driver anticipating a LLC. Thus, the driver may feel that it is unnecessary to do the same safety check for RLC (Lee, Olsen, Wierwille, et al., 2004).

In conclusion, SVM fusing eye-tracking signal can achieve good prediction performance for LLC case, however for RCL case, more driving uncertainty, e.g. specified contextual traffic, possibility of changing the lanes, should be known in advance to predict LC more precisely and reliably.

7.1.4 Contribution

Although the topic of prediction of driver LC behavior has been studied in many related works for a couple of years, some methods proposed in this dissertation are original and can be extended to the other similar fields of research. The original contributions of this dissertation are summarized as follows:

1. Although the method of using a cell-grid to model the contextual traffic and the way of using a behavioral-psychological questionnaire to classify diving style are from related research, the method of considering these two factors in preparing for the training datasets is unique.
2. A novel data labeling method termed as the GBL method is proposed. This labeling method takes advantage of driver gaze behavior and can make LC data labeling work correlated with driver LC behavior. In this way the quality of the training datasets can be improved.
3. Regarding to prediction of driver LC behavior, few research have done with a systematic feature selection work before training their ML models. The most common case is just using features based on the empirical knowledge or recommended in the prior research rather than performing a comprehensive feature selection work. This dissertation proposes a systematic feature selection method in the perspective of statistics. Wide ranges of features related to driver LC behavior are extracted, e.g. vehicle dynamic features, driver behavior features, combined features and time-window features, to enrich the feature sets. In addition, features are not limited in time-domain but also covering frequency-domain. This feature extraction and selection method is general for all the ML models and can be also extended to other field of research regarding to machine learning techniques.

7.2 Outlook

Modern ADASs need to adapt to driver intentions and situations to match the driver's actual need for assistance, that is to say the future ADASs themselves need the knowledge about the driver's intention (Leonhardt, Pech, and Wanielik, 2018). For the case of the LC assistance system, the framework presented in this dissertation is very promising in practical application. In the simulated real-time prediction test, by using the method proposed in this dissertation, driver LC behavior can be predicted on average 3.3 s ahead of an actual LC maneuver for LLC case and 2.6 s for RLC case. There are mainly two use cases for the prediction:

1. The prediction signal can be implemented in ADAS by either active feedback or passive feedback. For the active feedback, when the system predicts the driver wants to make LC, then ADAS could assist the driver with recommended acceleration and speed as well as the LC path etc. And for the passive feedback, ADAS can alarm the driver if the LC intention is unsafe in current driving situation.
2. The technology of the connected vehicles plays a key role in realizing cooperative intelligent transportation systems (Narla, 2013). The prediction signal can be used for inter-vehicle communication by sharing it with other surrounding vehicles and can thus cooperatively prevent the potential traffic accidents.

Although this dissertation proposes a comprehensive framework for prediction of driver LC behavior, it still has a long way to go for the real-road implementation. The following aspects should be particularly taken into account.

- Driving uncertainty: being maximal aware of the driving uncertainty is the goal, since without understanding of the driving uncertainties may lead to prediction failure as it is discussed in RLC case. In this dissertation, we can conclude that modeling contextual traffic is helpful for prediction of LC. However, more uncertainties, e.g. the possibility of changing the lanes, lane mark information (Leonhardt, Pech, and Wanielik, 2018), and even weather or illumination condition, are also necessary to be covered.
- Synchronization: the importance of this issue is usually underestimated. In this dissertation, an off-line synchronization method is proposed by re-sampling the data with interpolation. However, it is much more challenging to synchronize different sensors in real-time case (Alemdar and Ibnkahla, 2007). The Network Time Protocol (NTP) (Mock et al., 2000) or reference-broadcast synchronization (RBS) (Elson and Römer, 2003) method is possible solutions.
- Extending driving scenarios: the framework presented in this dissertation is designed for highway scenario. In city scenario, however, the traffic situation is more complex than on highway and thus it is more challenging. Adopting vehicle to

infrastructure (V2I) technology (Djahel, Jabeur, et al., 2015; Djahel, Salehie, et al., 2013; Barrachina et al., 2015) as well as high precision map (Matthaei, Bagschik, and Maurer, 2014; Seif and Hu, 2016; Schreier, Willert, and Adamy, 2013) could largely reduce the driving uncertainties in city scenario.

- Eye-tracking: it has been proved in many related research that using eye-tracking signal has positive effect on predicting driver behavior (Salvucci and Liu, 2002; Lethaus and Rataj, 2007; Tijerina et al., 2005; Kaplan et al., 2015). However, the use of eye-tracking in this dissertation is limited by the software provided by SMI to extract features from the eye-tracker. To further extend the usage of eye-tracking, computer vision technology is necessary to be implemented (Kim and Ramakrishna, 1999).
- Driving style: Although it is concluded in this dissertation that considering driving style in preparing for training datasets can improve the predictive performance of driver LC behavior, however, more researches are needed to define driving style as well as the role it plays in the development of ADASs. For example, based on French et al. (1993), Glaser and Waschulewski (2005), and Vöhringer-Kuhnt and Trexler-Walde (2005), by using questionnaire it is capable of defining the global driving style (Sagberg et al., 2015) of the driver but the driver' temporary driving styles are still needed to be captured by going deep into the driving data. The temporary driving styles can be represented by the temporary speed choice, accelerating and braking behavior, emotional status (stress or fatigue), sensation seeking and risk taking (Quimby et al., 1999), as well as the attitudes towards speed limits (Ahie, Charlton, and Starkey, 2015). As long as the driver remains a part of the control loop, driving and safety behaviors are more than just the mechanical operation of a vehicle (Hennessy, 2011), and thus driving style cannot be neglected.

A

Experiment 1

A.1 Documents

A.1.1 Demographic questionnaire - German version

Datum: Pbn.-Nr.:

Im Folgenden werden Ihnen einige Fragen zu Ihrer Person gestellt. Ihre Daten werden selbstverständlich anonym erhoben und ausgewertet.

1: Geschlecht

◯ weiblich

◯ männlich

2: Alter

[ ]

In dieses Feld dürfen nur Ziffern eingetragen werden

3: Was ist Ihr höchster Bildungsabschluss?
Bitte wähle eine der folgenden Antworten

◯ Hauptschulabschluss

◯ Mittlere Reife

◯ (Fach-) Abitur

◯ Bachelor / Master / Diplom / Magister

◯ Promotion / Habilitation

4: Besitzen Sie einen Führerschein?

◯ Ja

◯ Nein

5: Seit wie vielen Jahren besitzen Sie einen Führerschein?

[ ] **Jahre**

In dieses Feld dürfen nur Ziffern eingetragen werden

6: Wie viele Kilometer sind Sie schon insgesamt seit Erwerb Ihres Führerscheins gefahren?

[ ] **Kilometer**

In dieses Feld dürfen nur Ziffern eingetragen werden

7: Wie viele Kilometer fahren Sie im Durchschnitt pro Jahr?

[ ] **Kilometer**

In dieses Feld dürfen nur Ziffern eingetragen werden

8: Haben Sie die Erfahrung mit dem Fahren im Simulator?

◯ Ja

◯ Nein

9: Wie schnell fahren Sie auf der Autobahn bei normalen Verkehrsbedingungen?

[ ] **Km/h**

In dieses Feld dürfen nur Ziffern eingetragen werden

10: Wie oft überholen Sie auf der Autobahn Autos auf der rechten Spur? (1 = nicht sehr oft; 5 = sehr oft)

◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5

11: Wieviel Abstand zum vorausfahrenden Fahrzeug halten Sie beim Überholvorgang auf der Autobahn? (1 = wenig Abstand; 5 = viel Abstand)

◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5

12: Wie fahren Sie im Schnitt bei normalen Verkehrsbedingungen (keine Glätte, kein Stau etc.)?
(1 = defensiv; 5 = eher flott & zügig)

◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5

Für mögliche Rückfragen möchte ich Sie bitten, Ihre E-Mailadresse anzugeben:

(Ihre E-Mailadresse wird natürlich nicht weitergegeben)

Vielen Dank!

A.1.2 Behavioral-psychological questionnaire - German version

Question NO.2, NO.3, NO.8, NO.10, NO.12, NO.16, NO.23 are used for calculating the score of aggressiveness, the rest would not count. The participants are not aware of the use of this questionnaire.

Die folgende Liste enthält kleinere Fehler und Regelübertretungen, die Verkehrsteilnehmern von Zeit zu Zeit passieren. Bitte geben Sie im folgenden an, wie häufig Ihnen diese im letzten Jahr passiert sind. Da eine genaue Angabe oft schwierig ist, kreuzen Sie bitte das Kästchen an, das Ihrer Meinung nach am ehesten zutrifft.

	Nie (0)	Fast Nie (1)	Selten (2)	Gele-gentlich (3)	Häufig (4)	Sehr häufig (5)
1. Sie versuchen, im falschen Gang an der Ampel anzufahren.						
2. Sie ärgern sich über ein auf der Autobahn links fahrendes langsames Fahrzeug und überholen es rechts.						
3. Sie fahren dicht auf ein vorausfahrendes Fahrzeug auf, um dem Fahrer zu signalisieren, dass er schneller fahren oder Ihre Spur verlassen soll.						
4. Sie versuchen, jemanden zu überholen und bemerken nicht, dass er bereits nach links blinkt und abbiegen möchte.						
5. Sie haben vergessen, wo Sie das Auto im Parkhaus oder auf dem Parkplatz abgestellt haben.						
6. Sie betätigen aus Versehen einen Schalter (z. B. für den Blinker), obwohl Sie eigentlich einen anderen betätigen wollten (z. B. für die Scheibenwischer).						
7. Sie stellen fest, dass Sie eigentlich nicht genau wissen, wie die Strecke aussah, die sie gerade gefahren sind.						
8. Sie fahren noch über eine Ampel, obwohl Sie wissen dass Sie eigentlich anhalten müssten.						
9. Sie bemerken beim Abbiegen Fußgänger nicht, die die Straße überqueren.						
10. Sie ärgern sich über einen anderen Fahrer und jagen ihm hinterher, um ihm zu zeigen, was Sie von ihm halten.						
11. Sie erwischen am Kreisverkehr die falsche Ausfahrt.						
12. Sie halten sich nachts oder bei wenig Verkehr nicht an Geschwindigkeitsbegrenzungen.						

	Nie (0)	Fast Nie (1)	Selten (2)	Gele-gentlich (3)	Häufig (4)	Sehr häufig (5)
13. achten beim Einbiegen in eine Vorfahrtsstraße so sehr auf den dortigen Verkehr, dass Sie beinahe auf den Vordermann auf Ihrer Spur auffahren.						
15. Sue fahren, obwohl Sie wissen, dass Sie möglicherweise mehr Alkohol getrunken haben als erlaubt.						
16. Sie haben eine Abneigung gegen eine bestimmte Art von Autofahrern und Sie zeigen ihnen das, wo immer Sie können.						
17. Sie unterschätzen beim Überholen die Geschwindigkeit eines entgegenkommenden Fahrzeugs.						
18. Sie fahren beim Zurückstoßen gegen etwas, was Sie vorher nicht gesehen haben.						
19. Sie wollen nach A fahren und merken plötzlich, dass Sie sich auf dem Weg nach B befinden, z. B. weil Sie sonst immer nach B fahren.						
20. Sie ordnen sich vor einer Kreuzung in die falsche Spur ein.						
21. Sie übersehen ein "Vorfahrt gewähren"-Schild und stoßen beinahe mit einem bevorrechtigten Verkehrsteilnehmer zusammen.						
22. Sie versäumen beim Spurwechsel, vor dem Aussteigen, etc. in den Rückspiegel zu schauen.						
23. Sie lassen sich auf Wettrennen mit anderen Autofahrern ein.						
24. Sie bremsen auf rutschiger Fahrbahn zu scharf oder lenken nicht richtig, so dass sie ins Schleudern kommen.						

A.1.3 Overall instruction on the participants - German version

Liebe Teilnehmerin, lieber Teilnehmer,

Sie sind eingeladen, an einer Studie der technischen Universität Berlin teilzunehmen, die sich mit dem Autofahren in einer Simulation beschäftigt. Voraussetzungen für die Teilnahme sind, dass Sie den Führerschein der Klasse B haben und keine Brille tragen.

Wir sichern Ihnen zu, dass die in dieser Studie erhobenen Daten lediglich für Forschungszwecke anonymisiert verwendet und streng vertraulich behandelt werden. Rückschlüsse auf die Identität des Ausfüllenden werden nicht möglich sein.

Die Teilnahme erfolgt freiwillig. Sie haben jederzeit das Recht Ihr Einverständnis zur Teilnahme an der Studie, ohne Angaben von Gründen, zu widerrufen.

Die Studie wird ca. 40 Minuten dauern und Sie bekommen für die Teilnahme an dieser Studie am Ende eine VP-Stunde oder eine angemessene Aufwandschädigung.

Vielen Dank für Ihre Mitarbeit und Ihr Engagement!

A.1.4 Experiment instruction - German version

Sehr geehrte/r Proband/in,

vielen Dank für Ihre Teilnahme an diesem Experiment. Der Test wird im Fahrsimulator stattfinden und ungefähr 40 min dauern. Dabei wird mithilfe einer Eye-tracking Brille die Blickbewegung mit erfasst.

1. Bitte füllen Sie den Fragebogen (Persönliche Informationen zur Fahrerfahrung) aus!
2. Bitte setzen Sie sich nun in den Fahrsimulator und stellen Sie sich den Sitz so ein, dass Sie bequem die Pedale und das Lenkrad erreichen können! Der Hebel dafür befindet sich unter dem Sitz auf der rechten Seite.
3. Bitte setzen Sie die Eye-tracking Brille erst unter meiner Anleitung auf! Während der Kalibrierung, werde ich Sie bitten an spezifische Orte zu schauen.
4. Der vierte Teil beschäftigt sich mit den Fahraufgaben und bitte beachten Sie die folgenden Punkte die gesamte Zeit während Sie fahren!
    - Die Fahraufgaben sind unterteilt in mehrere kleinere Aufgaben. Sie werden auf einer zweispurigen Autobahn fahren. Die erste Fahrt ist zur Gewöhnung an den Fahrsimulator gedacht. Danach beginnt die eigentliche Testung.
    - Stellen Sie sich während der Fahrt vor, dass sie auf einer realen Autobahn sind und fahren Sie Ihren persönlichen Fahrstil, so wie Sie gewöhnlich fahren! Falls langsamer Verkehr Sie aufhalten sollte, können Sie die Spur wechseln. Beachten Sie dabei den Sicherheitsabstand nach vorn und zur Seite einzuhalten!
    - Bitte beachten Sie, dass während der Fahrt die Verkehrsregeln möglichst einzuhalten sind, z.B. Innenspiegel, Seitenspiegel anschauen und rechtzeitiges Blinken beim Spurwechsel usw.. Hier bitte beachten Sie, dass Sie den Blinker richtig runter rücken und nach Spurwechsel wieder zurückstellen müssen.
    - Nach dem Überholvorgang kehren Sie bitte wieder auf Ihre ursprüngliche Fahrspur zurück! Bitte beachten Sie weiterhin, dass die Sicherheit aller Verkehrsteilnehmer gewährleistet ist, auch zu dem überholten Fahrzeug!
    - Bitte bewegen Sie während der Fahrt nicht die Brille, da diese ja bereits kalibriert wurde. Ihren Kopf dürfen Sie in kleinen Winkeln drehen, jedoch nicht in größeren Winkeln (kein Schulterblick!)!
    - Fahren Sie los, sobald das Szenario startet, und stoppen Sie erst, wenn auf dem Bildschirm „Quit?“ steht!
    - Auf dem Autobahn sind keine Geschwindigkeitsbegrenzungen vorhanden!

Falls Fragen aufkommen, können Sie mich gerne jederzeit fragen!

Vielen Dank für Ihre Mitarbeit!

A.2 Figures

A.2.1 Statistics of the demographic questionnaire

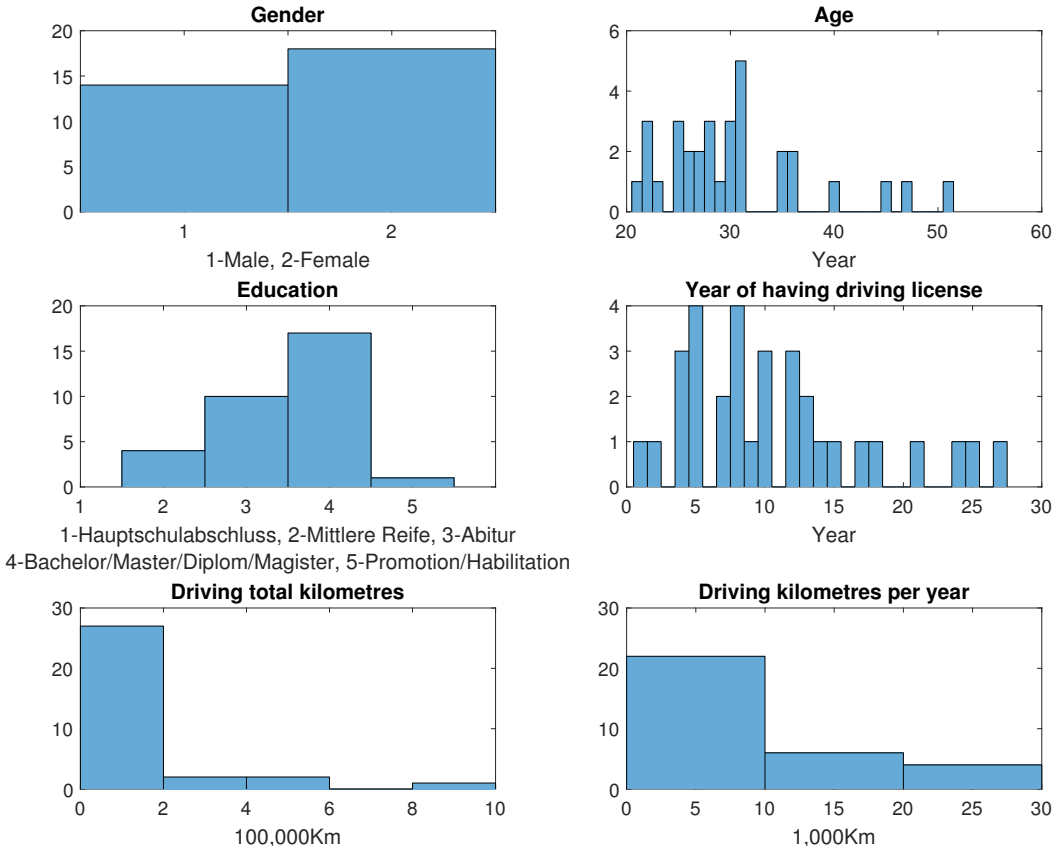


Figure A.1: Illustration of the background information of the participants in histogram.

A.2.2 Result of model comparison

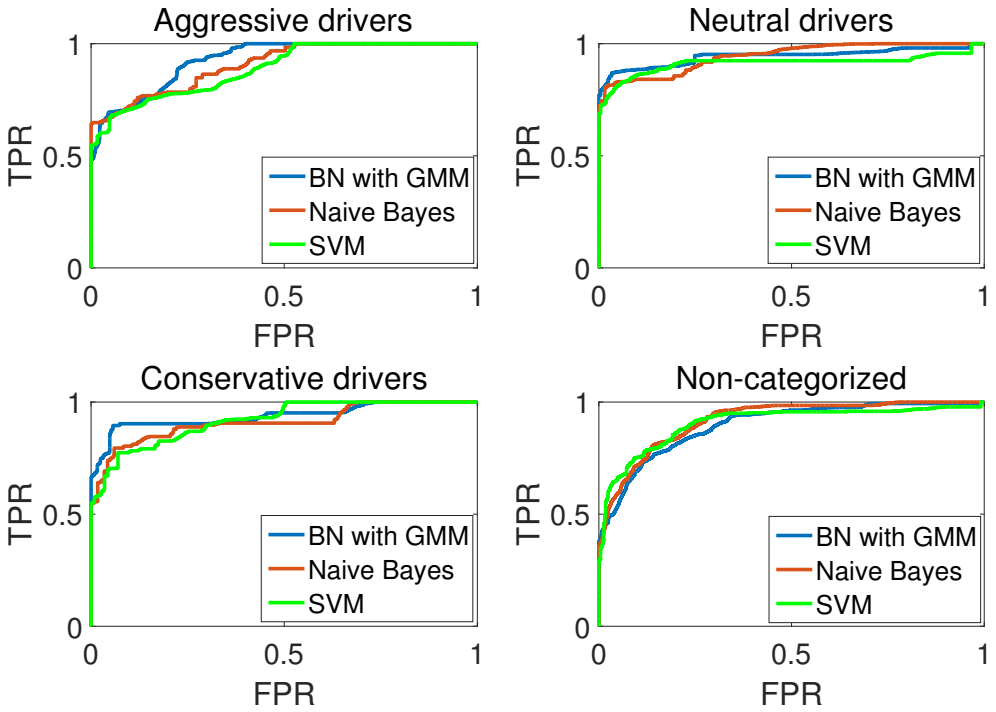


Figure A.2: The ROC curves of LCBN-GMM, Naive Bayes, and SVM in Scenario lead only.

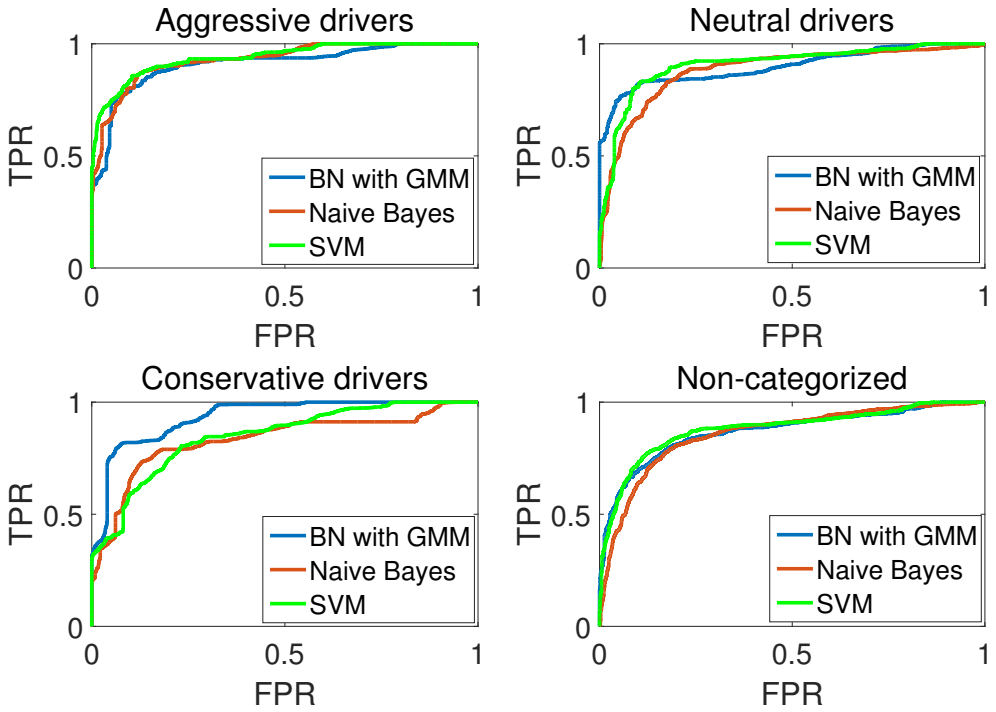


Figure A.3: The ROC curves of LCBN-GMM, Naive Bayes, and SVM in Scenario lead + adjacent behind.

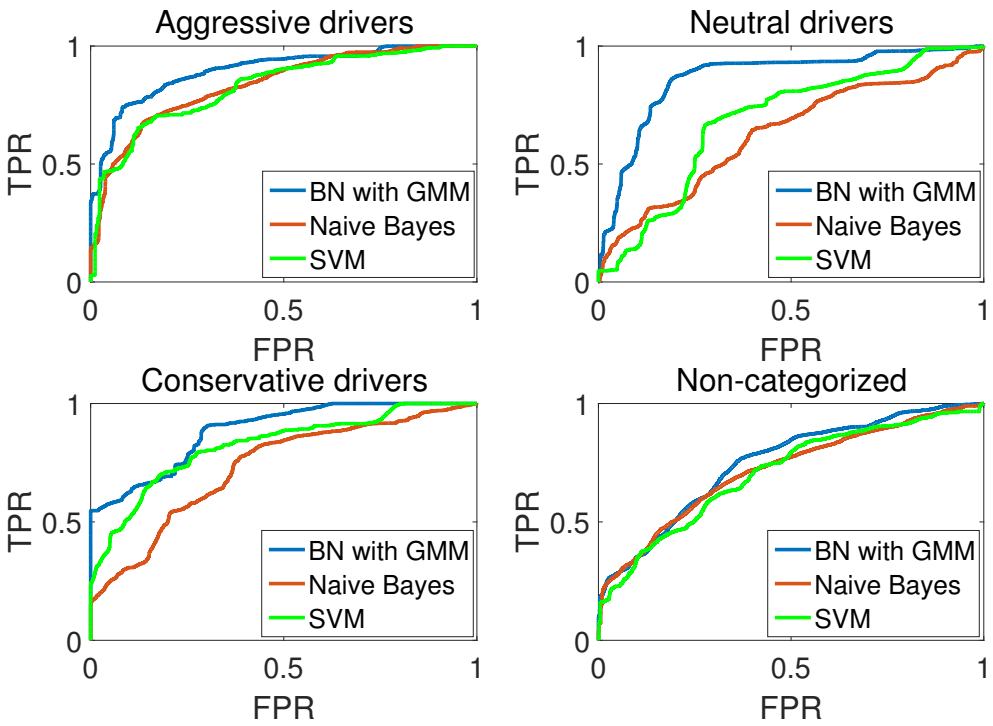


Figure A.4: The ROC curves of LCBN-GMM, Naive Bayes, and SVM in Scenario lead +2 adjacent.

A.2.3 Result of labeling method comparison

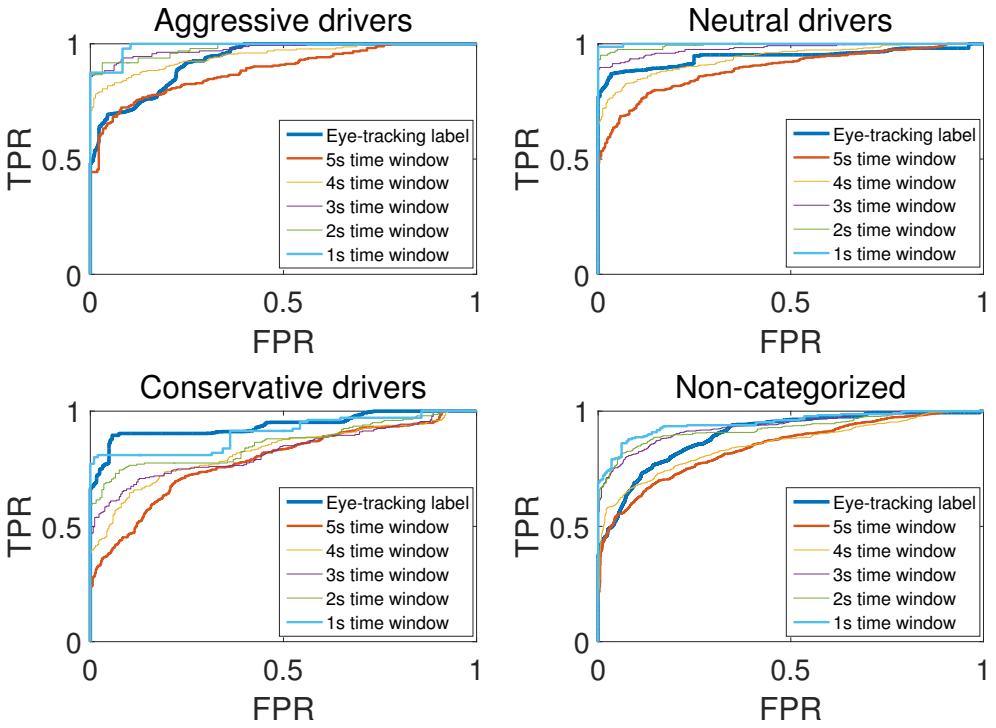


Figure A.5: The ROC curves of using different labeling strategies in Scenario lead only.

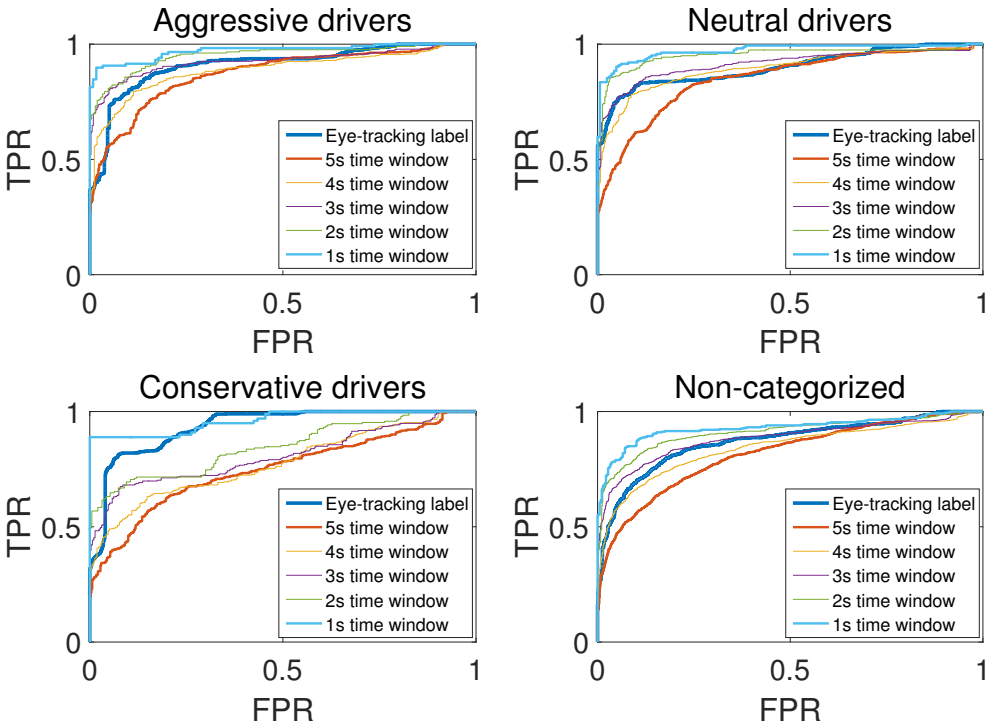


Figure A.6: The ROC curves of using different labeling strategies in Scenario lead + adjacent behind.

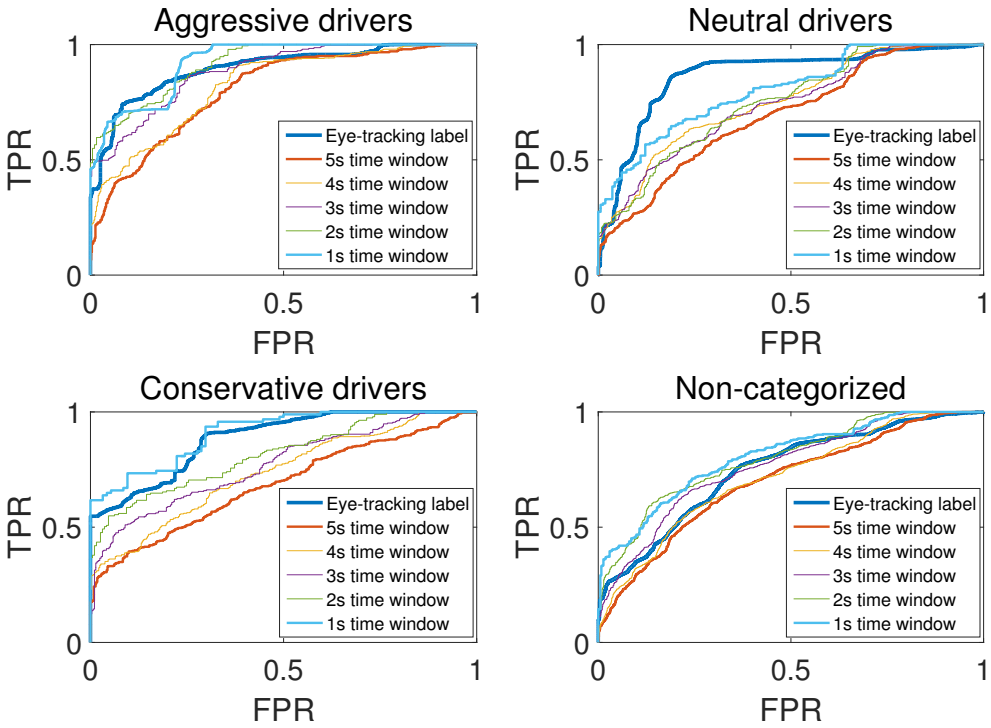


Figure A.7: The ROC curves of using different labeling strategies in Scenario lead +2 adjacent .

A.3 Tables

A.3.1 Full scale of AUC values by LCBN-GMM using TWL method

Table A.1: The AUC values of LCBN-GMM using TWL method in different time-window size.

Scenario	**Driving style**	**Length of TW**				
		5 s	4 s	3 s	2 s	1 s
lead only	Aggressive	0.88	0.94	0.97	0.97	0.98
	Neutral	0.89	0.95	0.97	0.99	0.99
	Conservative	0.78	0.82	0.82	0.86	0.90
	Mean	0.85	0.90	0.92	0.94	0.95
	Non-categorized	0.84	0.85	0.93	0.92	0.94
lead + adjacent behind	Aggressive	0.87	0.88	0.92	0.95	0.96
	Neutral	0.85	0.89	0.91	0.95	0.97
	Conservative	0.74	0.77	0.80	0.84	0.96
	Mean	0.82	0.85	0.87	0.91	0.96
	Non-categorized	0.81	0.84	0.88	0.91	0.92
lead + 2 adjacent	Aggressive	0.80	0.83	0.88	0.91	0.92
	Neutral	0.69	0.75	0.73	0.75	0.80
	Conservative	0.68	0.73	0.77	0.81	0.90
	Mean	0.72	0.80	0.78	0.82	0.87
	Non-categorized	0.69	0.77	0.70	0.79	0.80

B

Big data analysis

B.1 Tables

B.1.1 Result of feature selection for LLC scenarios

Table B.1: The full scale effect size of the features in LLC scenarios.

#	LLC Scenario							
	0_0		0_1		1_0		1_1	
	d	p	d	p	d	p	d	p
1	0.75	0.10	0.75	0.10	0.81	0.05	0.66	0.14
2	0.81	0.09	0.71	0.13	0.74	0.13	0.79	0.10
3	0.06	0.78	0.07	0.75	0.04	0.86	0.07	0.75
4	–	–	0.92 ▲	0.04 ▼	–	–	1.18 ▲	0.01 ▼
5	0.51	0.18	0.59	0.11	0.31	0.27	0.60	0.16
6	1.02 ▲	0.03 ▼	0.98	0.05	0.98	0.02	1.54 ▲	0.04 ▼
7	0.97	0.06	0.98 ▲	0.04 ▼	1.14 ▲	<0.01 ▲	1.32	0.07
8	0.91	0.06	0.89	0.06	1.00	0.01	0.83	0.07
9	0.86	0.06	0.88	0.06	0.89	0.06	0.92	0.09
10	0.82	0.07	0.79	0.09	0.86	0.10	1.04	0.10
11	0.98	0.04	1.05 ▲	0.03 ▼	0.91	0.05	0.98	0.08
12	0.97	0.04	1.04	0.05	1.21	0.12	1.03 ▲	0.01 ▼
13	0.99 ▲	0.04 ▼	0.97	0.06	1.08	0.06	0.79	0.02
14	0.91	0.06	0.90	0.08	0.77	0.03	1.07	0.09
15	0.80	0.09	0.73	0.12	0.63	0.17	0.84	0.11
16	1.00 ▲	0.04 ▼	0.93	0.04	0.62	0.08	1.34	0.02
17	0.94	0.05	0.97 ▲	0.03 ▼	0.94	0.08	1.36 ▲	0.03 ▼

Continued on next page

Table B.1 – Continued from previous page

#	LLC Scenario							
	0_0		0_1		1_0		1_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*
18	0.98	0.06	0.93	0.04	0.78	0.11	1.40	0.06
19	0.92	0.07	0.88	0.06	0.74	0.07	1.23	0.09
20	0.84	0.06	0.84	0.08	0.79	0.10	0.91	0.05
21	0.98	0.04	0.95 ▲	0.03 ▼	0.89	0.06	1.38	0.07
22	1.03 ▲	0.03 ▼	0.94	0.04	0.77	0.14	1.20 ▲	0.04 ▼
23	1.00	0.04	0.94	0.05	1.04 ▲	0.03 ▼	0.72	0.05
24	0.94	0.05	0.91	0.07	0.91	0.02	0.56	0.01
25	0.88	0.07	0.82	0.07	0.90	0.09	0.99	0.05
26	0.92 ▲	0.04 ▼	0.95	0.05	1.11	0.06	1.18	0.08
27	0.91	0.06	0.93	0.05	0.95 ▲	<0.01 ▼	1.15	0.10
28	0.87	0.06	0.83	0.06	0.80	0.03	1.23	0.07
29	0.80	0.09	0.82	0.06	0.87	0.06	0.98	0.06
30	0.79	0.08	0.77	0.09	0.82	0.09	1.01	0.09
31	0.92 ▲	0.04	0.94	0.04	0.87 ▲	0.01 ▼	0.70	0.02
32	0.91	0.03 ▼	0.95 ▲	0.04 ▼	0.67	0.03	0.59	0.02
33	0.87	0.04	0.86	0.06	0.59	0.11	0.85	0.07
34	0.86	0.07	0.85	0.06	0.66	0.13	0.80	0.07
35	0.83	0.07	0.81	0.10	0.75	0.06	0.71	0.10
36	0.91 ▲	0.04 ▼	1.05 ▲	0.04 ▼	1.06	0.01	1.00	0.05
37	0.93	0.05	0.98	0.05	1.07	0.01	1.05	0.09
38	0.99	0.05	0.93	0.07	1.23 ▲	<0.01 ▼	0.92	0.10
39	0.94	0.07	0.88	0.08	1.10	<0.01	0.74	0.06
40	0.75	0.10	0.75	0.11	0.87	0.01	0.54	0.08
41	0.98 ▲	0.04 ▼	0.99	0.04	0.76	0.08	0.73	0.03
42	0.96	0.04	1.01 ▲	0.03 ▼	0.78	0.06	0.53	<0.01
43	0.90	0.05	0.95	0.05	0.85	0.04	0.79 ▲	0.01 ▼
44	0.89	0.06	0.91	0.04	0.71	0.03	0.60	0.05
45	0.87	0.07	0.86	0.09	0.49	0.08	0.65	0.05
46	0.93 ▲	0.03 ▼	0.90 ▲	0.04 ▼	0.58	0.14	0.90	0.01
47	0.89	0.04	0.88	0.05	0.66	0.11	0.82	<0.01
48	0.85	0.06	0.87	0.06	0.74	0.17	1.00	<0.01
49	0.81	0.07	0.91	0.06	0.45	0.07	0.89	0.04
50	0.84	0.07	0.84	0.08	0.63	0.16	1.04 ▲	0.03 ▼
51	0.88 ▲	0.04 ▼	0.94 ▲	0.04 ▼	0.97 ▲	0.01 ▼	0.94	0.05
52	0.86	0.05	0.91	0.05	0.61	0.04	0.83	<0.01
53	0.81	0.05	0.88	0.06	0.61	0.15	0.82	<0.01
54	0.82	0.07	0.85	0.06	0.74	0.05	0.91 ▲	0.03 ▼

Continued on next page

Table B.1 – Continued from previous page

#	LLC Scenario							
	0_0		0_1		1_0		1_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*
55	0.84	0.07	0.81	0.10	0.79	0.05	0.83	0.05
56	0.37	0.34	0.44	0.28	0.31	0.38	0.30	0.39
57	0.36	0.34	0.44	0.27	0.33	0.37	0.39	0.31
58	0.35	0.33	0.44	0.25	0.30	0.40	0.34	0.25
59	0.31	0.35	0.43	0.23	0.28	0.48	0.38	0.19
60	0.24	0.40	0.34	0.27	0.23	0.48	0.39	0.25
61	0.91	0.05	0.96	0.05	1.21 ▲	<0.01 ▼	0.76	0.06
62	0.90	0.05	0.95	0.05	1.16	0.01	1.02 ▲	0.03 ▼
63	0.93	0.07	0.93	0.05	1.06	0.07	0.93	0.03
64	0.91	0.07	0.85	0.08	1.02	0.04	0.89	0.07
65	0.73	0.08	0.69	0.12	0.82	0.01	0.70	0.05
66	0.88	0.06	0.91	0.06	1.00	<0.01	0.79	0.06
67	0.87	0.07	0.90	0.06	0.97	0.07	0.77	0.02
68	0.91	0.07	0.89	0.07	1.06 ▲	0.04 ▼	0.70	0.04
69	0.87	0.07	0.82	0.10	0.98	0.05	0.63	0.02
70	0.70	0.08	0.67	0.12	0.70	0.03	0.59	0.06
71	0.94 ▲	0.04 ▼	0.94	0.06	0.98	<0.01	0.95	0.06
72	0.93	0.05	0.94	0.06	0.99 ▲	<0.01 ▼	1.12	0.05
73	0.94	0.06	0.91	0.07	1.03	0.11	0.98	0.06
74	0.91	0.07	0.82	0.08	0.98	0.07	0.98	0.07
75	0.71	0.10	0.65	0.11	0.81	0.05	0.80	0.05
76	0.83	0.06	0.84	0.07	0.71	0.07	0.64	0.04
77	0.82	0.07	0.81	0.08	0.76	0.03	0.65	0.08
78	0.79	0.09	0.80	0.11	0.62	0.04	0.59	0.07
79	0.67	0.11	0.64	0.13	0.59	0.18	0.52	0.09
80	0.48	0.20	0.44	0.21	0.47	0.16	0.42	0.13
81	0.91	0.05	0.96	0.05	0.81	0.02	0.75	0.06
82	0.94	0.05	0.83	0.07	0.87 ▲	0.01 ▼	1.20	0.08
83	0.90	0.07	0.85	0.09	0.70	0.07	1.14 ▲	0.04 ▼
84	0.84	0.09	0.78	0.11	0.59	0.12	0.90	0.02
85	0.69	0.14	0.62	0.15	0.83	0.10	0.73	0.13
86	0.98 ▲	0.04 ▼	0.97	0.06	0.61	0.07	0.83 ▲	0.02 ▼
87	0.86	0.06	0.89	0.06	0.79	0.08	0.86	0.07
88	0.89	0.07	0.87	0.07	1.03	0.05	0.66	0.06
89	0.82	0.09	0.79	0.10	0.87	0.07	0.83	0.10
90	0.66	0.14	0.59	0.17	0.74	0.10	0.72	0.13
91	0.92	0.05	0.84	0.05	0.97	<0.01	1.16	0.10
92	0.87	0.05	0.79	0.09	1.03	0.08	0.90	0.14

Continued on next page

Table B.1 – Continued from previous page

#	LLC Scenario							
	0_0		0_1		1_0		1_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*
93	0.81	0.09	0.81	0.10	1.18 ▲	<0.01 ▼	0.65	0.16
94	0.79	0.09	0.80	0.09	1.09	0.11	0.74	0.09
95	0.65	0.15	0.61	0.17	1.01	0.11	0.52	0.11

B.1.2 Result of feature selection for RLC scenarios

Table B.2: The full scale effect size of the features in RLC scenarios.

#	RLC Scenario							
	0_0		0_1		1_0		1_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*
1	0.66	0.11	0.60	0.12	0.76	0.09	0.66	0.19
2	0.69	0.12	0.96 ▲	0.04 ▼	0.81	0.09	0.79	0.08
3	0.06	0.78	0.07	0.72	0.07	0.76	0.07	0.74
4	–	–	–	–	0.84	0.06	1.18 ▲	<0.01 ▼
5	0.55	0.16	0.82 ▲	0.02 ▼	0.62	0.13	0.71	0.04
6	0.92 ▲	0.04 ▼	1.29 ▲	0.04 ▼	0.96 ▲	0.04 ▼	1.54 ▲	<0.01 ▼
7	0.91	0.04	0.01	0.11	0.94	0.06	1.32	0.02
8	0.92	0.05	0.72	0.03	0.85	0.07	0.83	0.01
9	0.89	0.07	1.11	<0.01	0.80	0.09	0.92	0.01
10	0.78	0.08	0.94	0.14	0.78	0.08	1.04	0.08
11	1.01 ▲	0.03 ▼	1.29	0.02	1.02 ▲	0.03 ▼	0.98 ▲	0.03 ▼
12	0.95	0.04	1.28	<0.01	0.99	0.03	1.03	0.11
13	0.96	0.05	1.60 ▲	<0.01 ▼	0.97	0.07	0.79	0.02
14	0.91	0.09	1.34	<0.01	0.88	0.09	1.07	0.10
15	0.79	0.10	1.02	0.03	0.74	0.12	0.84	0.09
16	0.97 ▲	0.03 ▼	1.28	<0.01	0.95 ▲	0.04 ▼	1.34	<0.01
17	1.02	0.05	1.13	<0.01	0.97	0.05	1.36	<0.01
18	0.99	0.05	1.03	<0.01	0.96	0.05	1.40 ▲	<0.01 ▼
19	0.93	0.06	1.38	<0.01	0.91	0.06	1.23	<0.01
20	0.88	0.06	0.93	<0.01	0.83	0.06	0.91	0.05
21	0.98 ▲	0.04 ▼	1.10 ▲	0.02 ▼	1.06 ▲	0.02 ▼	1.38 ▲	<0.01 ▼
22	0.93	0.04	0.98	<0.01	1.03	0.04	1.20	0.03
23	0.93	0.04	0.60	0.11	0.94	0.06	0.72	0.04
24	0.89	0.07	0.24	0.23	0.91	0.06	0.56	0.09
25	0.80	0.08	0.69	0.04	0.82	0.07	0.99	0.14
26	0.90	0.05	0.65	0.06	1.04 ▲	0.03 ▼	1.18	<0.01
27	0.92	0.05	0.83	0.06	1.00	0.04	1.15	0.03
28	0.89	0.05	1.05	0.11	0.85	0.07	1.23 ▲	0.01 ▼
29	0.77	0.08	0.90	0.16	0.77	0.08	1.01	0.07
31	0.87	0.06	1.16 ▲	<0.01 ▼	0.96	0.04	0.70	0.02
32	0.89	0.06	0.88	<0.01	0.98 ▲	0.04 ▼	0.59	0.05
33	0.89	0.07	0.69	0.02	0.95	0.03	0.85	0.06
34	0.87	0.06	1.04	0.07	0.95	0.06	0.80	0.06
35	0.83	0.06	1.07	0.09	0.83	0.07	0.71	<0.01
36	0.91 ▲	0.04 ▼	0.85 ▲	0.02 ▼	0.94 ▲	0.04 ▼	1.00	0.06

Continued on next page

Table B.2 – Continued from previous page

#	RLC Scenario							
	0_0		0_1		1_0		1_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*
37	0.96	0.05	0.66	0.03	0.92	0.05	1.05	0.06
38	0.97	0.05	0.79	0.05	0.90	0.07	0.92 ▲	0.03 ▼
39	0.90	0.06	0.81	0.21	0.84	0.11	0.74	0.05
40	0.71	0.12	0.89	0.11	0.69	0.13	0.54	0.22
41	0.92	0.04	1.00	0.10	0.98 ▲	0.04 ▼	0.73	0.07
42	0.95 ▲	0.04 ▼	1.12	0.16	0.87	0.04	0.53	0.09
43	0.93	0.05	0.90	0.09	0.90	0.04	0.79	0.11
44	0.93	0.04	0.63	0.06	0.84	0.06	0.60	0.13
45	0.91	0.05	0.90	0.02	0.83	0.07	0.65	0.01
46	0.84 ▲	0.04 ▼	1.28 ▲	0.02 ▼	0.94 ▲	0.03 ▼	0.90	0.06
47	0.85	0.05	1.30	0.05	0.92	0.03	0.82	0.08
48	0.83	0.05	1.01	0.09	0.95	0.05	1.00	0.04
49	0.87	0.07	0.90	0.02	0.95	0.05	0.89	0.05
50	0.80	0.09	0.89	0.05	0.90	0.08	1.04	0.01
51	0.91	0.05	1.01 ▲	<0.01 ▼	0.90	0.05	0.94 ▲	<0.01 ▼
52	0.88	0.05	0.63	0.09	1.00 ▲	0.04 ▼	0.83	0.06
53	0.85	0.07	0.54	0.02	0.93	0.04	0.82	0.10
54	0.84	0.07	0.92	0.09	0.93	0.08	0.91	0.04
55	0.82	0.06	1.07	0.06	0.83	0.07	0.83	0.01
56	0.41	0.31	0.58	0.12	0.48	0.26	0.30	0.46
57	0.42	0.30	0.54	0.10	0.48	0.26	0.39	0.24
58	0.39	0.31	0.71	0.09	0.47	0.25	0.34	0.23
59	0.36	0.30	0.58	0.13	0.43	0.26	0.38	0.17
60	0.28	0.35	0.38	0.27	0.32	0.30	0.39	0.12
61	0.96	0.05	1.16 ▲	0.02 ▼	0.95	0.06	0.76	0.03
62	0.93	0.07	1.11	0.10	0.91	0.07	1.02 ▲	0.01 ▼
63	0.97	0.08	0.96	0.09	0.86	0.07	0.93	0.01
64	0.90	0.05	0.80	0.04	0.79	0.09	0.89	0.07
65	0.72	0.09	0.79	0.03	0.70	0.09	0.70	0.17
66	0.92	0.07	1.13 ▲	0.01 ▼	0.97	0.05	0.79	0.01
67	0.91	0.06	0.86	0.15	0.93	0.07	0.77	0.07
68	0.90	0.08	0.96	0.08	0.88	0.07	0.70	0.18
69	0.84	0.09	0.86	0.05	0.79	0.10	0.63	0.17
70	0.65	0.11	0.67	0.17	0.65	0.13	0.59	0.24
71	0.94	0.06	1.10▲	0.02 ▼	0.85	0.08	0.95	0.13
72	0.92	0.05	1.09	0.01	0.92	0.08	1.12 ▲	0.03 ▼
73	0.95	0.07	0.88	0.11	0.89	0.08	0.98	0.01
74	0.86	0.07	0.76	0.05	0.82	0.09	0.98	0.02

Continued on next page

Table B.2 – Continued from previous page

#	RLC Scenario							
	0_0		0_1		1_0		1_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*
75	0.70	0.10	0.72	0.15	0.68	0.10	0.80	0.11
76	0.77	0.08	1.06 ▲	0.04 ▼	0.77	0.10	0.64	0.15
77	0.76	0.10	1.08	0.06	0.78	0.11	0.65	0.17
78	0.72	0.13	0.91	0.02	0.74	0.12	0.59	0.13
79	0.67	0.14	0.80	0.10	0.69	0.14	0.52	0.24
80	0.44	0.20	0.53	0.12	0.49	0.23	0.42	0.20
81	0.86	0.06	0.95 ▲	0.01 ▼	0.93	0.05	0.75	0.08
82	0.88 ▲	0.04 ▼	0.43	0.01	0.90	0.06	1.20 ▲	0.03 ▼
83	0.87	0.08	1.36	0.05	0.82	0.08	1.14	0.01
84	0.83	0.11	0.88	0.03	0.78	0.10	0.90	0.09
85	0.68	0.15	0.62	0.05	0.68	0.14	0.73	0.06
86	0.90	0.06	0.88 ▲	0.02 ▼	0.95 ▲	0.04 ▼	0.83	0.09
87	0.88	0.08	0.86	0.14	0.88	0.07	0.86 ▲	0.04 ▼
88	0.81	0.08	0.70	0.12	0.87	0.08	0.66	0.09
89	0.72	0.13	0.63	0.23	0.75	0.10	0.83	0.13
90	0.58	0.19	0.57	0.10	0.59	0.17	0.72	0.16
91	0.89	0.07	0.52	0.01	0.81	0.07	1.16 ▲	0.02 ▼
92	0.88	0.07	0.88	0.10	0.88	0.06	0.90	0.09
93	0.90	0.07	1.05	0.02	0.86	0.06	0.65	0.14
94	0.79	0.08	1.11 ▲	<0.01 ▼	0.82	0.09	0.74	0.11
95	0.59	0.15	1.09	<0.01	0.61	0.15	0.52	0.18

B.2 Figures

B.2.1 Result of model performance using selected features for LLC scenarios

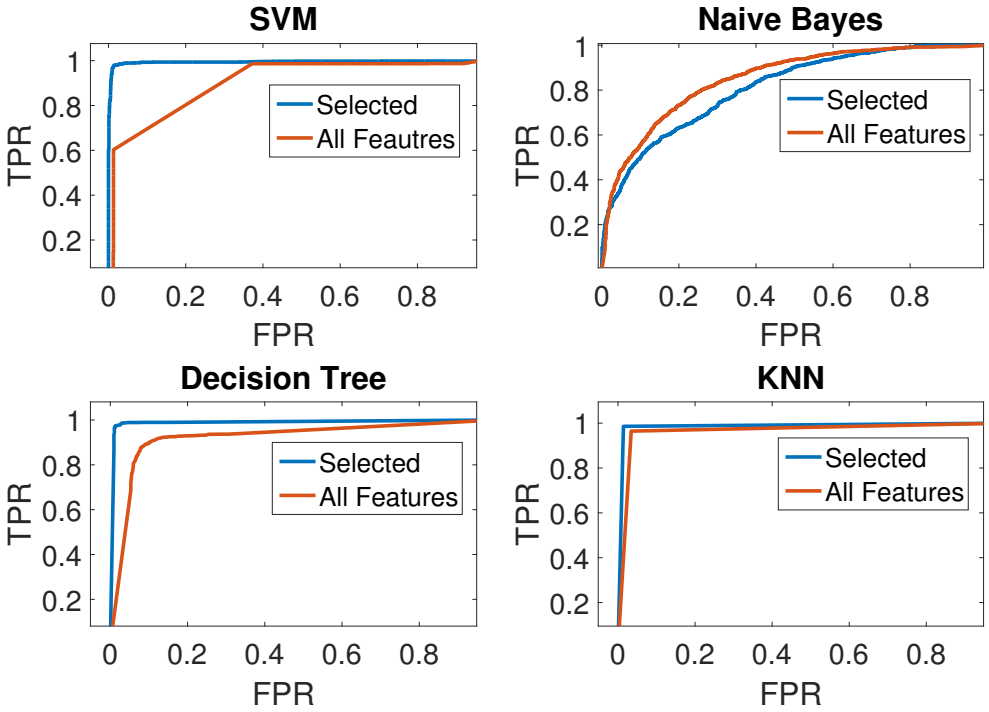


Figure B.1: The ROC curves of the classification performance for LLC Scenario 0_0.

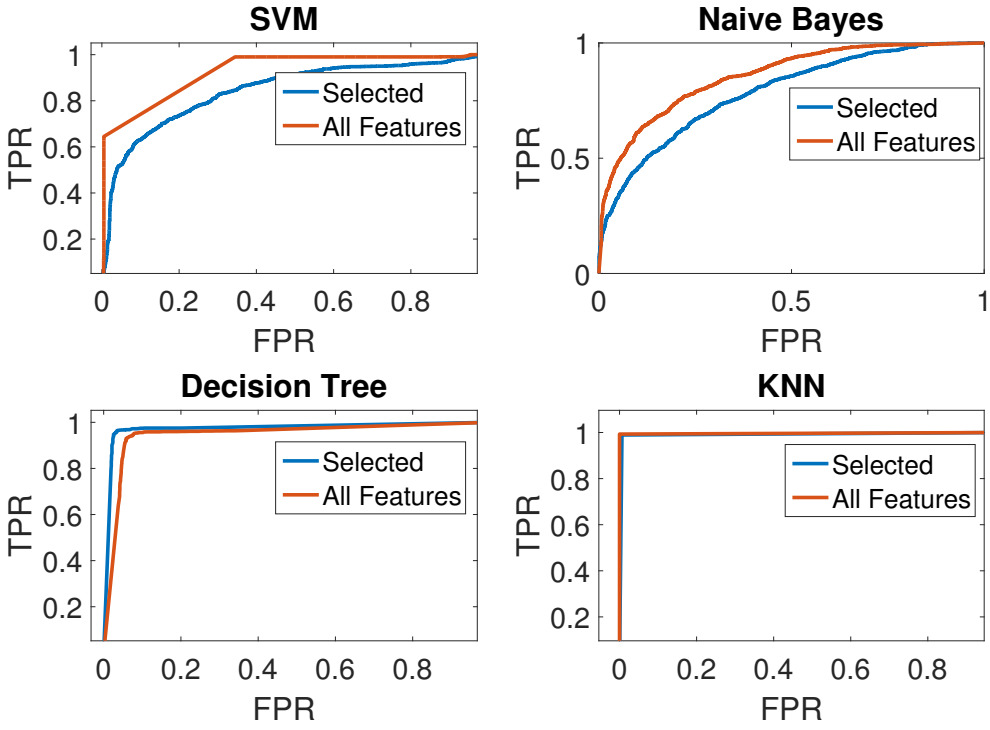


Figure B.2: The ROC curves of the classification performance for LLC Scenario 0_1.

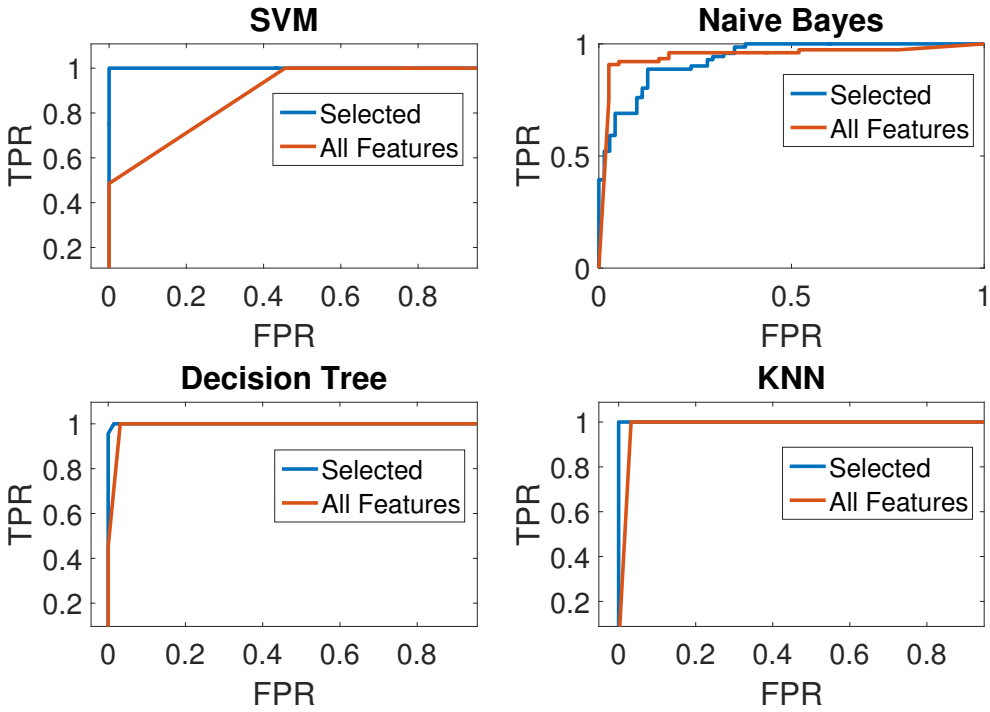


Figure B.3: The ROC curves of the classification performance for LLC Scenario 1_0.

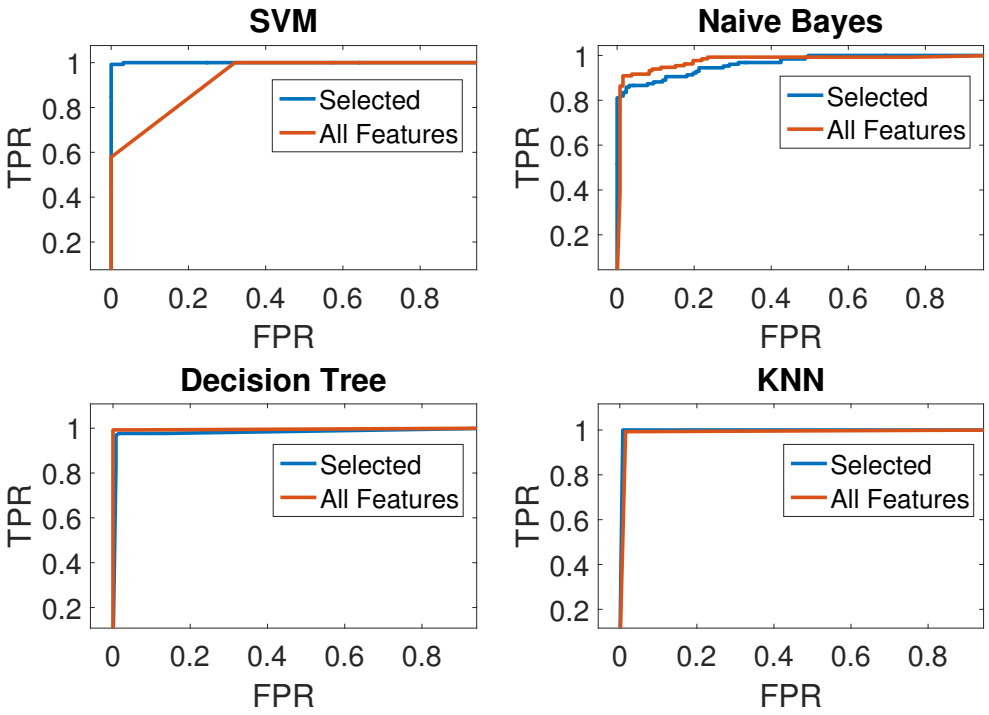


Figure B.4: The ROC curves of the classification performance for LLC Scenario 1_1.

B.2.2 Result of model performance using selected features for RLC scenarios

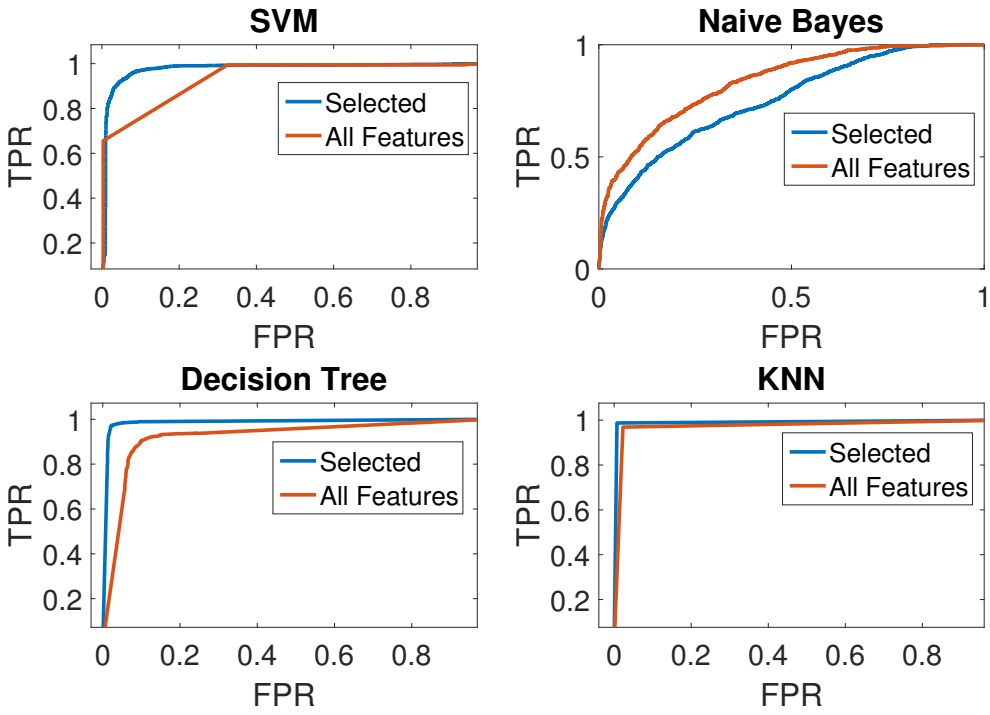


Figure B.5: The ROC curves of the classification performance for RLC Scenario 0_0.

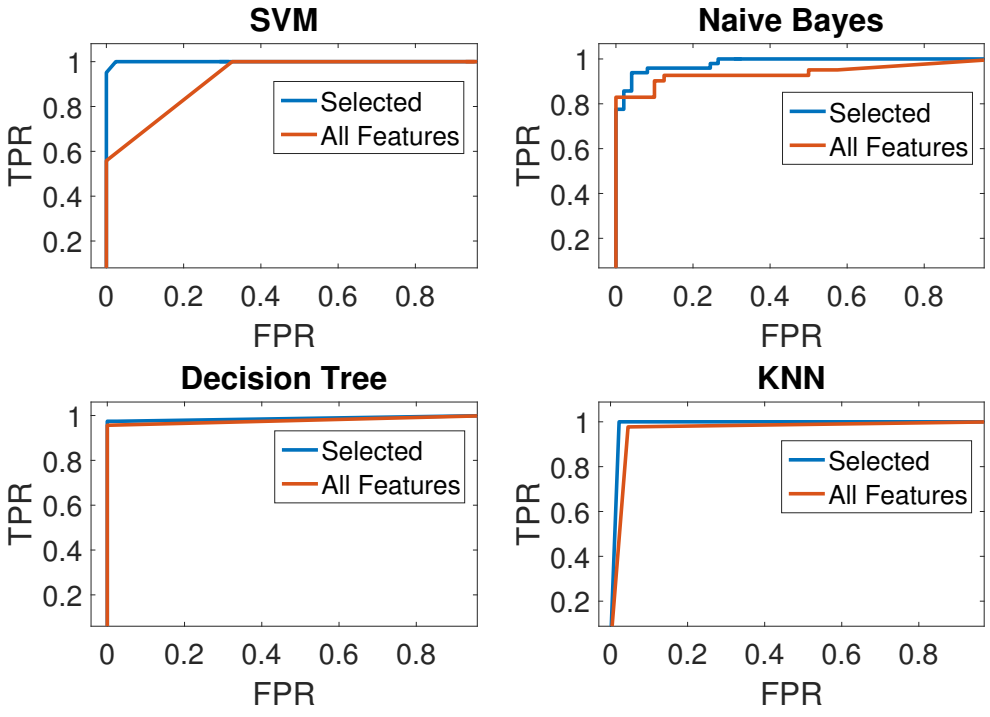


Figure B.6: The ROC curves of classification performance for RLC Scenario 0_1.

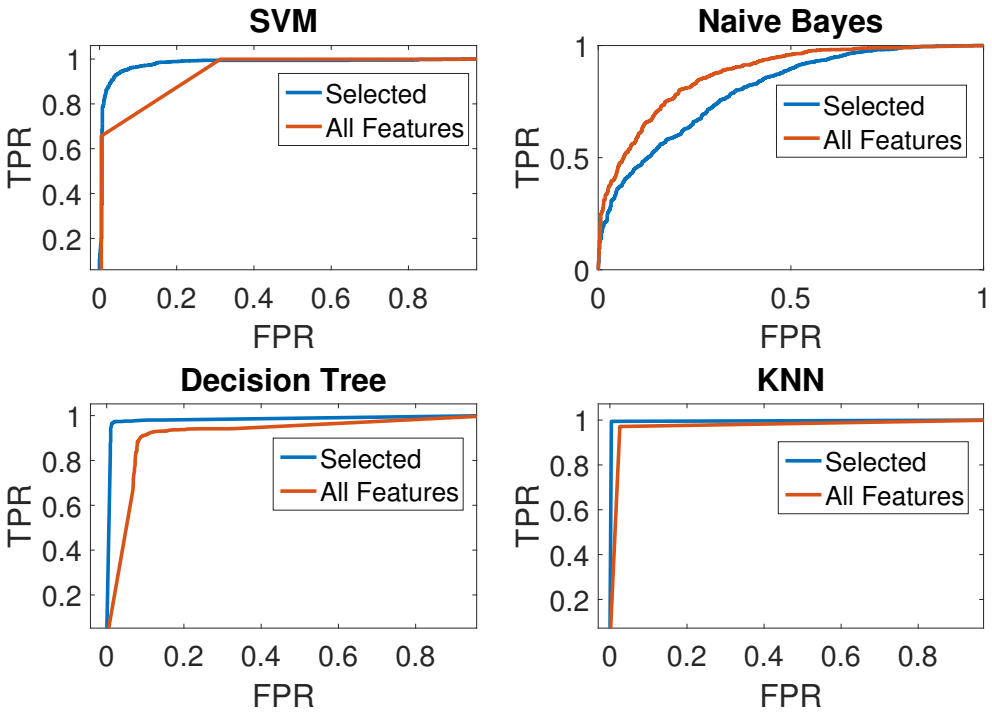


Figure B.7: The ROC curves of the classification performance for RLC Scenario 1_0.

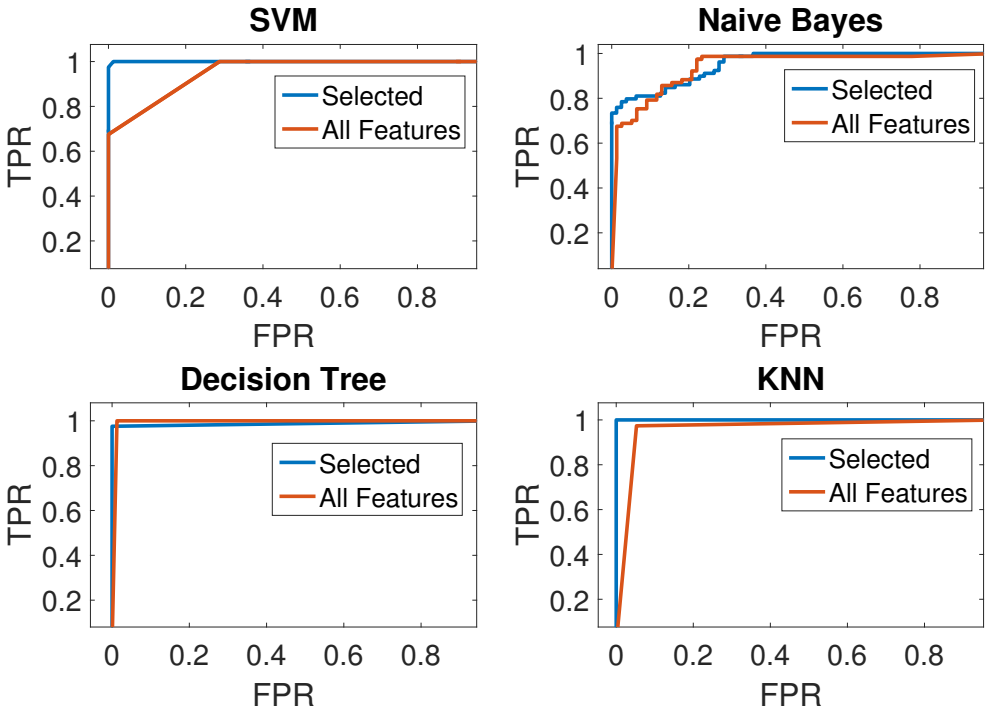


Figure B.8: The ROC curves of the classification performance for RLC Scenario 1_1.

C

Experiment 2

C.1 Documents

C.1.1 Driver selection questionnaire - German version

Question NO.4, NO.6, NO.7, NO.8, NO.9, NO.10 are used for selecting participants.

Screener: Driver Intention II **Name:_____**

1.) Wie alt sind Sie? _____

2.) Welche Führerscheinklassen besitzen Sie? _____

3.) Wie lange besitzen Sie Ihren Führerschein bereits? _____

4.) Haben Sie Probleme beim Hören?

Wenn ja → Ende ○ Ja, ○ Nein

6.) Haben Sie Schwierigkeiten Ihre Hände und/oder Finger zu bewegen?

Wenn ja → Ende ○ Ja, ○ Nein

7.) Haben Sie Probleme in der Motorik im rechten Fuß?

Wenn ja → Ende ○ Ja, ○ Nein

8.) Nehmen Sie regelmäßig Medikamente ein, die Ihre Fahrleistung beeinträchtigen können?

Wenn ja → Ende ○ Ja, ○ Nein

9.) Haben Sie eine Farbsehschwäche?

Wenn ja → Ende ○ Ja, ○ Nein

10.) Tragen Sie eine Brille? ○ Ja, ○ Nein

→ Wenn Brille: Besitzen Sie auch Kontaktlinsen, die Sie zum Versuch tragen können? Dies ist wichtig, da wir im Versuch Ihre Blickbewegung mit Eye-Tracking messen und eine Brille die Messung stören kann.

Wenn Fahrt nur mit Brille möglich → Ende

11.) Bitte erscheinen Sie zu dem Versuch ohne Augen-Make-up, oder stellen Sie sich darauf ein, dieses vor Ort zu entfernen. Sind Sie hiermit einverstanden?

Wenn nein → Ende ○ Ja, ○ Nein

12.) Bitte erscheinen Sie zum Versuch in einem gesundheitlich guten Zustand und sein Sie zum Zeitpunkt der Versuchsdurchführung fahrtüchtig. Sind Sie hiermit einverstanden?“

Wenn nein → Ende ○ Ja, ○ Nein

13.) Bitte bringen Sie zum Versuch ihren Führerschein und Personalausweis mit. Sind Sie damit einverstanden?

Wenn nein → Ende ○ Ja, ○ Nein

Uhrzeit und Datum des vereinbarten Termins: _____

Handynummer für Notfälle:_____

C.1.2 Recruiting participants - German version

PB-Nr.:_____

Datum_____

Uhrzeit_____

Besondere Ereignisse:

Uhrzeit	Was?	Strecke (Abschnitt 1,2 oder 3)

Verbale Ermahnung des Probanden war notwendig in Fahrt…nach…min…wegen:

Fahrt	Minute	Grund des verbalen Eingreifens

Fehlende Daten (z.B. fehlende Videoaufzeichnung), technische Schwierigkeiten:

_____

_____

_____

_____

Sonstiges:

_____
_____
_____
_____
_____

Sitzposition:

Titel	Maß
Winkel d. Rückenlehne	
Winkel d. Lenkrades	
Höhe d. Sitzes (cm)	
Abstand d. Sitzes (cm)	
Abstand d. Lenkrades (cm)	

C.1.3 Demographic questionnaire - The graphical user interface in German

Demographie

Age: [1] Bitte wählen Sie Ihre Altersgruppe aus.

Gender: [1] Bitte wählen Sie Ihr Geschlecht aus.

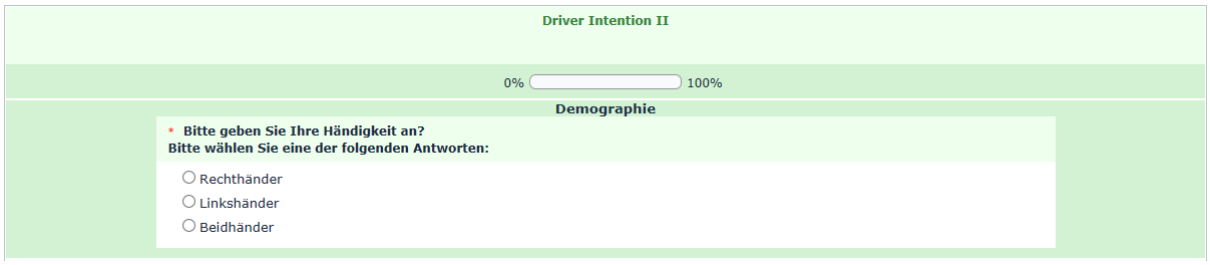
Hand: [1] Bitte geben Sie Ihre Händigkeit an?

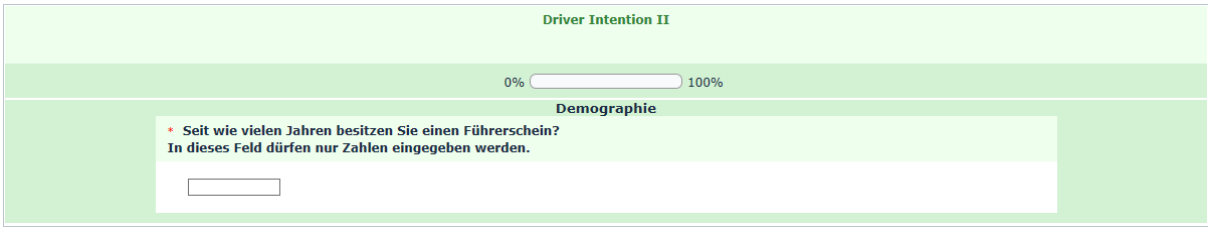
DrivingExp: [1] Seit wie vielen Jahren besitzen Sie einen Führerschein?

DrivingFreq: [1] Wie oft fahren Sie mit einem PKW?

AvgMileage: [1] Wie viele Kilometer fahren Sie durchschnittlich pro Jahr?

Gear: [1] Mit welche Art von Getriebe sind Sie es gewohnt zu fahren?





C.1.4 Experimental instruction - German version

Vorbereitung (Am Auto)

1. Rekrutierung

   ➔ Mit Probanden Screener durchgehen
   ➔ Probanden einladen (Kalendereintrag)

2. Auto starten:

   ➔ Tür öffnen
   ➔ Innerhalb von 1 Min. Auto starten
   ➔ Lenkrad für 2 Min. nicht anfassen
   ➔ Auf das blinkende grüne Licht in der Mittelkonsole achten

3. Laptop starten und Eyetracker anschließen:

   ➔ In Versuche → „Driver Intention Studie“ → Kodierung des Probanden: VP_Nr_Abschnitt (z.B. VP_1_1, VP_1_2, VP_1_3)

4. Kalibrierungsmatrix auf den Beifahrersitz legen.

5. Ereignisprotokoll (2x) und Einverständniserklärung (1x) ausdrucken

Begrüßung: (Im E-Labor)

Herzlich Willkommen zu unserer Echtfahrzeug-Studie!

Zusammenfassung der Studie

(Im E- Labor)

Bevor wir beginnen, fasse ich Ihnen zunächst einmal die wichtigsten Informationen der heutigen Studie zusammen. Bei weiteren Fragen können Sie sich selbstverständlich jederzeit an mich wenden.

Zweck der Studie ist die Untersuchung des Fahrverhaltens unter Berücksichtigung der Handposition, der Augenbewegung und verschiedener Fahrdaten in einer realen Fahrsituation.

Es handelt sich bei diesem Fahrzeug um ein Erprobungsfahrzeug mit Automatikgetriebe, dass mit einer speziellen Messtechnik ausgestattet ist. In dem Lenkrad befinden sich kapazitive Sensoren mit denen wir Ihre Handposition erfassen. Zusätzlich werden Ihre Blickbewegungen mit Hilfe eines Eyetrackers aufgenommen. Dieser muss vor Beginn der Fahrt kalibriert werden. Gegebenenfalls muss die Kalibrierung im Verlauf der Studie wiederholt werden. Außerdem werden verschiedene Fahrmanöver, wie z.B. Abbiegen, Überholen und Kurvenfahren, durch meinen Kollegen kodiert.

Haben Sie bis hierhin noch Fragen?

Diese Studie wird insgesamt etwa 3 Stunden dauern und beinhaltet zwei Pausen. Wir fahren darin eine festgelegte Strecke.

➔ Laminierte Strecke zeigen

Sie müssen sich den Weg nicht merken. Diese Abbildung dient lediglich der Orientierung. Während der Studie werde ich Ihnen stets rechtszeitig mitteilen, in welche Richtung Sie weiterfahren müssen (Siehe Kommentar). Sofern Sie keine weiteren Instruktionen erhalten, setzen Sie die Strecke in Fahrtrichtung fort.

Kommentar:

- ➔ *Die Änderung der Fahrtrichtung soll dem Probanden rechtszeitig mitgeteilt werden, dh. die Instruktion erfolgt mindesten 10 Sekunden bevor die Fahrtrichtung geändert wird. In die spätere Untersuchung gehen Daten von 4 Sekunden vor dem Fahrmanöver mit ein. Ziel ist es ein möglichst natürliches Fahrverhalten zu erreichen.*
- ➔ *Es soll bei Instruktionen vermieden werden Landmarken zu wählen, die unmittelbaren Einfluss auf das Blickverhalten haben (beispielsweise Straßennamen, die können, wenn sie nicht bekannt sind, erst aus der Nähe identifiziert werden). Besser ist es rechtzeitig große Gebäude/Ampeln, die bereits aus der Ferne zu sehen sind zu nennen. Alternativ soll auf Richtungsangaben (beispielsweise an der kommenden Straße rechts) zurückgegriffen werden.*
- ➔ *Falls es nicht möglich dem Probanden mindestens 10 Sekunden vor Fahrmanöver einen Fahrtrichtungswechsel mitzuteilen, werden Instruktion gebündelt mitgeteilt (beispielsweise an der kommenden Kreuzung rechts und danach direkt wieder links abbiegen).*
- ➔ *Sobald ein Fahrmanöver erfolgt ist, soll dem Probanden direkt angekündigt werden, wann er das nächste Mal die Fahrtrichtung gewechselt wird (beispielsweise folgen Sie dem Straßenverlauf für 3 km).*

Haben Sie dazu Fragen? Wenn nein, dann fortfahren.

Fragebogen und Einverständniserklärung

Bevor wir die Fahrt beginnen, würde ich Sie bitte die Einverständniserklärung genau zu lesen und zu unterschreiben.

Toilette

Sie haben jetzt nochmal die Möglichkeit auf die Toilette zu gehen, wenn Sie möchten. Später werden Sie in den Pausen ebenfalls die Möglichkeit haben die Toilette zu benutzen, ansonsten würde ich ungerne die Studie unterbrechen.

Instruktionen am Erprobungsfahrzeug

Sie können sich nun in das Fahrzeug setzen und den Fahrersitz sowie die Spiegel in eine geeignete Position einstellen. Falls Sie Hilfe benötigen, helfe ich Ihnen gerne weiter.

➔ Lüftung des Fahrzeugs auf maximal stellen

Dann zeige ich Ihnen zunächst einmal die Bedienung des Fahrzeugs.

Bedienung des Fahrzeugs:

Erläuterung wie:

1. das Fahrzeug gestartet und gestoppt wird

➔ „Das Fahrzeug ist bereits gestartet. Stoppen Sie das Fahrzeug nur, wenn wir Sie explizit drauf hinweisen."

➔ „Sie können bei Bedarf das Fahrzeug parken, indem Sie den P-Knopf drücken. Bitte schalten Sie das Fahrzeug dabei nicht aus."

2. die Handbremse und Gänge eingelegt werden

Assistenzsysteme:

1. Head-up-Display

Haben Sie zur Bedienung des Fahrzeugs noch Fragen? Wenn nein, dann fortfahren.

Sie können zu jedem Zeitpunkt den Versuch abbrechen. Möchten Sie den Versuch abbrechen, können Sie dies dem Versuchsleiter einfach mitteilen. Wir halten dann an der nächsten Gelegenheit an und bringen Sie zurück zu Joyson Safety Systems. Andererseits kann durch den roten Notabschaltungsknopf in der Mittelkonsole die Messtechnik auch direkt abschaltet werden. Achten Sie bitte darauf, dass der Knopf nicht versehentlich, sondern nur im Notfall gedrückt wird, da Ihre Daten sonst verloren gehen.

Selbstverständlich müssen Sie sich an die Straßenverkehrsordnung halten. Achten Sie besonders darauf, dass wenn Sie auf der Autobahn Fahrzeuge überholen, Sie sich im Anschluss wieder rechts einsortieren.

Haben Sie dazu noch Fragen?

Kalibrierung des Eyetrackers

Wir werden den Eyetracker mindesten dreimal im Verlauf der Studie kalibrieren. Gegebenenfalls muss der Eyetracker innerhalb eines Studienabschnittes außerplanmäßig nachkalibriert werden. Ist dies der Fall, werde ich Sie auffordern bei der nächsten Gelegenheit rechts anzuhalten, damit wir die Kalibrierung wiederholen können.

Dann beginnen wir jetzt mit der ersten Kalibrierung.

Erläuterung und Eyetracker aufsetzen:

1. Bitte setzen Sie sich den Eyetracker möglich bequem auf. Achten Sie darauf, dass dabei keine Druckstellen entstehen und er nicht verrutschen kann.

➔ Falls hier bereits Druckstellen auftreten, dann mit Schaumgummi abpolstern.

➔ Eyetracker Kabel mit ausreichend Spiel nach hinten führen und fixieren.

2. Probanden die Kalibrierungsmatrix zeigen

➔ Kalibrierungsmatrix etwa 50 cm vor den Körper halten

➔ Alle Punkte sollen von der Szenenkamera erfasst werden.

„Achten Sie bitte darauf, dass während Kalibrierung der Kopf nicht bewegt werden darf. Sie sollen nur mit den Augen die Punkte fixieren."

Kalibrierung und Schnell-Check:

3. Bitte halten Sie ihren Kopf ruhig und fixieren Sie die 1. nur mit den Augen

➔ Punkt anklicken

4. Bitte halten Sie ihren Kopf ruhig und fixieren Sie die 3. nur mit den Augen

➔ Punkt anklicken

5. Bitte halten Sie ihren Kopf ruhig und fixieren Sie die 8. nur mit den Augen

➔ Punkt anklicken

6. Kalibrierung anhand der gesamten Matrix überprüfen.
7. Schnell-Check:
    - Schauen Sie bitte in den linken Seitenspiegel
    - Schauen Sie bitte in den Rückspiegel
    - Schauen Sie bitte in den rechten Seitenspiegel

Dann beginnen wir jetzt mit der Studie. Bitte wundern Sie sich nicht, dass während des Versuchs wenig gesprochen wird, da ablenkende Gespräche einen Einfluss auf das Fahrverhalten haben. Wir wünschen Ihnen eine angenehme Zeit im Erprobungsfahrzeug und eine sichere Fahrt!

C.2 Figures

C.2.1 Statistics of the demographic questionnaire

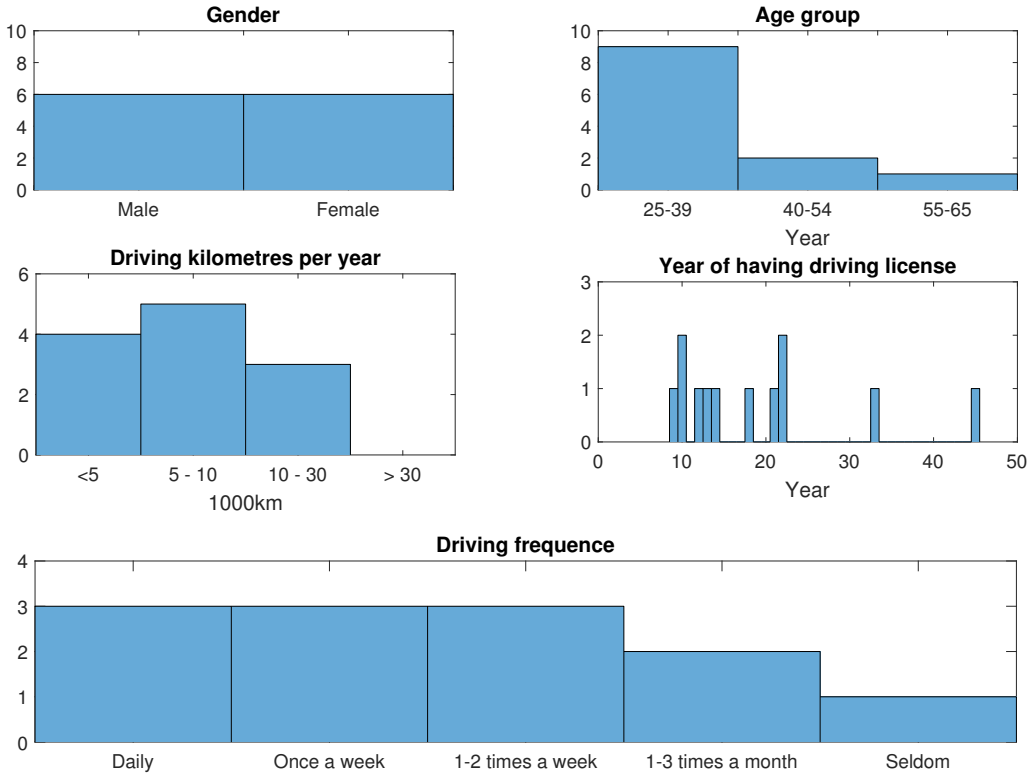


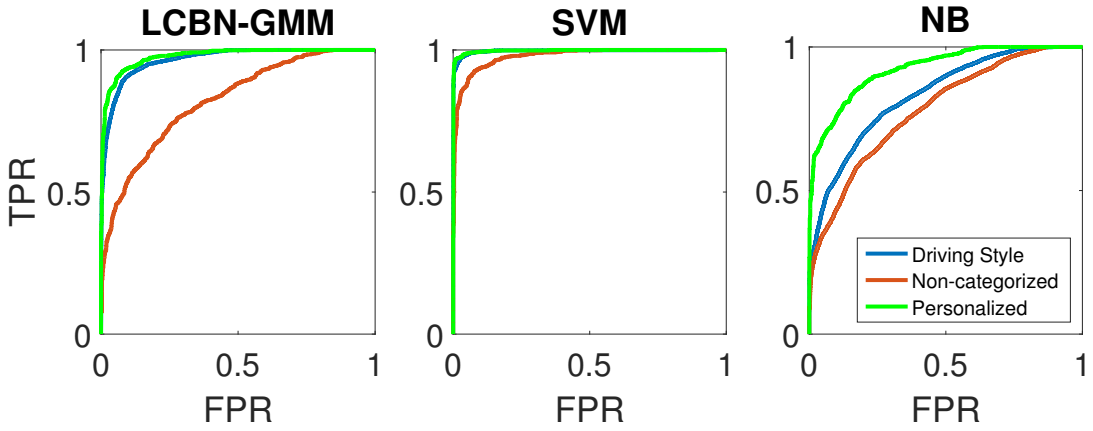
Figure C.1: The statistics of the demographical questionnaire in histogram.

C.2.2 Data sample

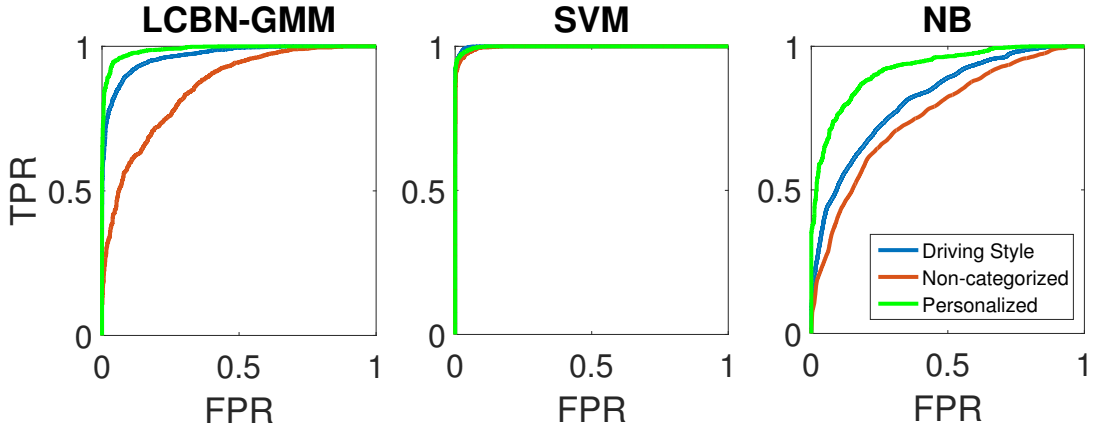
Time			AoI												Event
8346030	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.11	-0.20	0.97	-0.14	-0.13	0.98	Fixation
8399607	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.11	-0.20	0.97	-0.14	-0.13	0.98	Fixation
8412516	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.12	-0.20	0.97	-0.15	-0.13	0.98	Fixation
8445942	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.12	-0.20	0.97	-0.15	-0.13	0.98	Fixation
8481127	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.13	-0.21	0.97	-0.16	-0.13	0.98	Fixation
8513843	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.13	-0.21	0.97	-0.16	-0.14	0.98	Fixation
8545536	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.14	-0.22	0.97	-0.17	-0.14	0.97	Fixation
8578947	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.15	-0.22	0.96	-0.18	-0.14	0.97	Fixation
8623006	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.16	-0.22	0.96	-0.19	-0.14	0.97	Fixation
8657901	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.17	-0.22	0.96	-0.21	-0.14	0.97	Fixation
8679204	SMP	1	White Space	29.38	4.74	-23.08	-30.24	6.43	-22.84	-0.19	-0.21	0.96	-0.23	-0.13	0.97	Saccade
8715946	SMP	1	-	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Saccade
8745272	SMP	1	-	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Saccade
8778319	SMP	1	-	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Saccade
8814526	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.28	-0.35	0.89	0.29	-0.28	0.92	Fixation
8845126	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.25	-0.34	0.91	0.26	-0.26	0.93	Fixation
8878013	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.24	-0.33	0.91	0.24	-0.25	0.94	Fixation
8911250	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.23	-0.32	0.92	0.22	-0.24	0.94	Fixation
8944494	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.22	-0.32	0.92	0.21	-0.23	0.95	Fixation
8977742	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.21	-0.32	0.93	0.20	-0.23	0.95	Fixation
9011090	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.20	-0.31	0.93	0.19	-0.21	0.96	Fixation
9064116	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.18	-0.30	0.94	0.18	-0.20	0.96	Fixation
9100359	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.17	-0.30	0.94	0.16	-0.20	0.97	Fixation
9133784	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.15	-0.30	0.94	0.14	-0.20	0.97	Saccade
9146424	SMP	1	-	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Saccade
9178383	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.11	-0.31	0.94	0.10	-0.22	0.97	Fixation
9210660	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.08	-0.32	0.94	0.07	-0.22	0.97	Fixation
9243929	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.04	-0.33	0.94	0.04	-0.22	0.97	Fixation
9277111	SMP	1	leftmirror		29.38	4.74	-23.08	-30.24	6.43	-22.84	0.15	-0.33	0.93	0.17	-0.22	0.96	Saccade
9310358	SMP	1	-	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Saccade
9344995	SMP	1	-	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Saccade
9376866	SMP	1	Lwindow	29.38	4.74	-23.08	-30.24	6.43	-22.84	0.57	-0.28	0.77	0.62	-0.14	0.77	Fixation
9410113	SMP	1	Lwindow	29.38	4.74	-23.08	-30.24	6.43	-22.84	0.61	-0.27	0.74	0.66	-0.13	0.74	Fixation
9443459	SMP	1	Lwindow	29.38	4.74	-23.08	-30.24	6.43	-22.84	0.62	-0.26	0.74	0.64	-0.13	0.76	Fixation
9476611	SMP	1	Lwindow	29.38	4.74	-23.08	-30.24	6.43	-22.84	0.60	-0.26	0.76	0.61	-0.14	0.78	Fixation

Figure C.2: The output data example from software BeGaze 3.7, where the term *white screen* refers to *Wind screen* and *LWindow* refers to *Left mirror*.

C.2.3 Model and dataset comparison



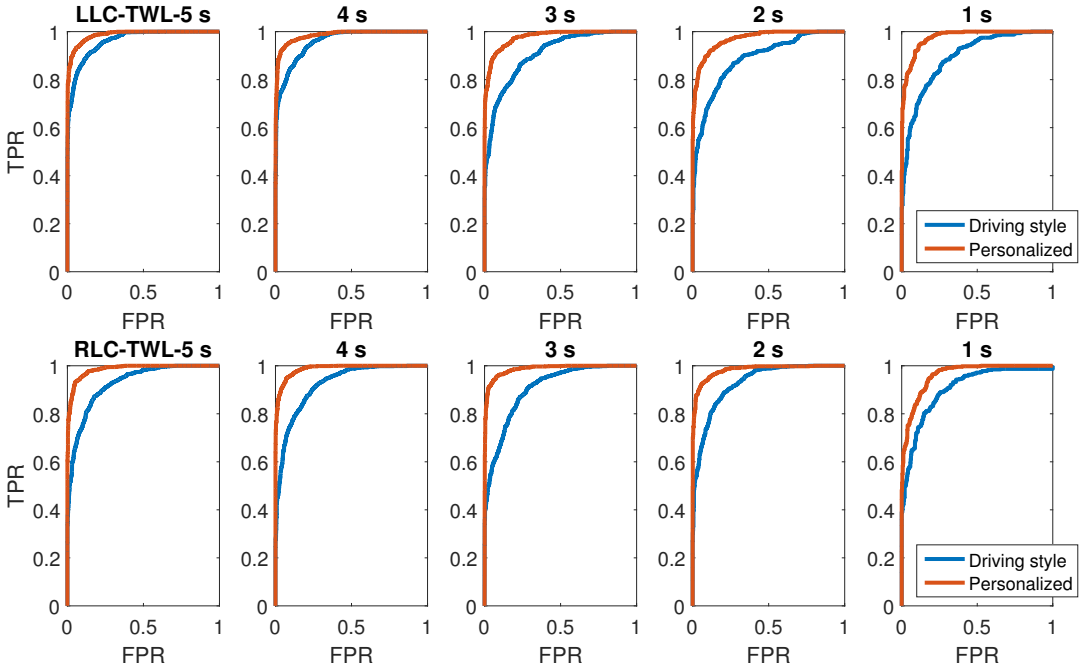
(a) The ROC curves of different models in LLC scenario.



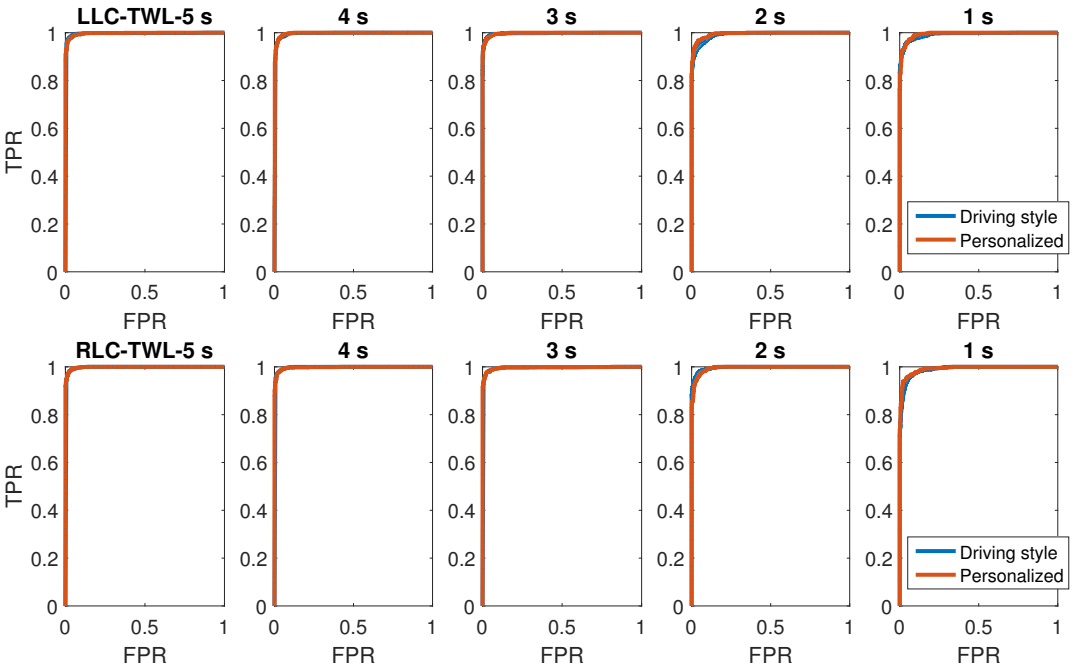
(b) The ROC curves of different models in RLC scenario.

Figure C.3: The classification performance of LCBN-GMM, SVM and NB trained by different datasets using GBL method.

C.2.4 Labeling method comparison



(a) The ROC curves of LCBN-GMM in both LLC and RLC scenarios.



(b) The ROC curves of SVM in both LLC and RLC scenarios.

Figure C.4: The classification performance of LCBN-GMM and SVM trained by two different datasets using TWL method.

C.3 Tables

C.3.1 Extracted features

Table C.1: Description of the extracted features from real road experiment.

#	Feature name	Feature description
1	$yawRate_t$	yaw rate of the ego vehicle at time t
2	a_t	acceleration of the ego vehicle at time t
3	$\mathrm{TTC}_\mathrm{t}^{-1}$	$\mathrm{TTC}_\mathrm{t}^{-1}$ at time t
4	$brpd_t$	brake pedal pressure of the ego vehicle at time t
5	$acpd_t$	throttle opening angle of the ego vehicle at time t
6	stw_t	steer wheel angle rate of the ego vehicle at time t
7	$stwRate_t$	steer wheel angle rate of the ego vehicle at time t
8	$mean\_yaw_t^5$	mean of *yawRate* in TW 5 s
9	$mean\_yaw_t^4$	mean of *yawRate* in TW 4 s
10-12	⋮	$mean\_yaw_t$ in TW 3 s, 2 s, 1 s
13	$std\_yaw_t^5$	standard deviation of *yawRate* in TW 5 s
14-17	⋮	std_yaw_t in TW 4 s, 3 s, 2 s, 1 s
18	$max\_yaw_t^5$	maximum of *yawRate* in TW 5 s
19-22	⋮	max_yaw_t in TW 4 s, 3 s, 2 s, 1 s
23	$min\_yaw_t^5$	minimum of *yawRate* in TW 5 s
24-27	⋮	min_yaw_t in TW 4 s, 3 s, 2 s, 1 s
28	$med\_yaw_t^5$	median of *yawRate* in TW 5 s
29-32	⋮	med_yaw_t in TW 4 s, 3 s, 2 s, 1 s
33	$mean\_a_t^5$	mean of the acceleration in TW 5 s
34-37	⋮	$mean\_a_t$ in TW 4 s, 3 s, 2 s, 1 s
38	$std\_a_t^5$	standard deviation of acceleration in TW 5 s
39-42	⋮	std_a_t in TW 4 s, 3 s, 2 s, 1 s
43	$max\_a_t^5$	maximum of acceleration in TW 5 s
44-47	⋮	max_a_t in TW 4 s, 3 s, 2 s, 1 s
48	$min\_a_t^5$	minimum of acceleration in TW 5 s
49-52	⋮	min_a_t in TW 4 s, 3 s, 2 s, 1 s
53	$med\_a_t^5$	median of acceleration in TW 5 s
54-57	⋮	med_a_t in TW 4 s, 3 s, 2 s, 1 s
58	$mean\_brpd_t^5$	mean of *brpd* in TW 5 s
59-62	⋮	$mean\_brpd_t$ in TW 4 s, 3 s, 2 s, 1 s
63	$std\_brpd_t^5$	standard deviation of *brpd* in TW 5 s

Continued on next page

Table C.1 – Continued from previous page

#	Feature name	Feature description
64-67	⋮	std_brpt_t in TW 4 s, 3 s, 2 s, 1 s
68	$max\_brpd_t^5$	maximum of *brpd* in TW 5 s
69-72	⋮	max_brpt_t in TW 4 s, 3 s, 2 s, 1 s
73	$min\_brpd_t^5$	minimum of *brpd* in TW 5 s
74-77	⋮	min_brpt_t in TW 4 s, 3 s, 2 s, 1 s
78	$med\_brpd_t^5$	median of *brpd* in TW 5 s
79-82	⋮	med_brpt_t in TW 4 s, 3 s, 2 s, 1 s
83	$mean\_acpd_t^5$	mean of *acpd* in TW 5 s
84-87	⋮	$mean\_brpt_t$ in TW 4 s, 3 s, 2 s, 1 s
88	$std\_acpd_t^5$	standard deviation of *acpd* in TW 5 s
89-92	⋮	std_acpd_t in TW 4 s, 3 s, 2 s, 1 s
93	$max\_acpd_t^5$	maximum of *acpd* in TW 5 s
94-97	⋮	max_acpd_t in TW 4 s, 3 s, 2 s, 1 s
98	$min\_acpd_t^5$	minimum of *acpd* in TW 5 s
99-102	⋮	min_acpd_t in TW 4 s, 3 s, 2 s, 1 s
103	$med\_acpd_t^5$	median of *acpd* in TW 5 s
104-107	⋮	med_acpd_t in TW 4 s, 3 s, 2 s, 1 s
108	$mean\_stw_t^5$	mean of *stw* in TW 5 s
109-112	⋮	$mean\_brpt_t$ in TW 4 s, 3 s, 2 s, 1 s
113	$std\_stw_t^5$	standard deviation of *stw* in TW 5 s
114-117	⋮	std_stw_t in TW 4 s, 3 s, 2 s, 1 s
118	$max\_stw_t^5$	maximum of *stw* in TW 5 s
119-122	⋮	max_stw_t in TW 4 s, 3 s, 2 s, 1 s
123	$min\_stw_t^5$	minimum of *stw* in TW 5 s
124-127	⋮	min_stw_t in TW 4 s, 3 s, 2 s, 1 s
128	$med\_stw_t^5$	median of *stw* in TW 5 s
129-132	⋮	med_stw_t in TW 4 s, 3 s, 2 s, 1 s
133	$mean\_stwRate_t^5$	mean of *stwRate* in TW 5 s
134-137	⋮	$mean\_stwRate_t$ in TW 4 s, 3 s, 2 s, 1 s
138	$std\_stwRate_t^5$	standard deviation of *stwRate* in TW 5 s
139-142	⋮	$std\_stwRate_t$ in TW 4 s, 3 s, 2 s, 1 s
143	$max\_stwRate_t^5$	maximum of *stwRate* in TW 5 s
144-147	⋮	$max\_stwRate_t$ in TW 4 s, 3 s, 2 s, 1 s
148	$min\_stwRate_t^5$	minimum of *stwRate* in TW 5 s
149-152	⋮	$min\_stwRate_t$ in TW 4 s, 3 s, 2 s, 1 s
153	$med\_stwRate_t^5$	median of *stwRate* in TW 5 s

Continued on next page

Table C.1 – Continued from previous page

#	Feature name	Feature description
154-157	⋮	$med\_stwRate_t$ in TW 4 s, 3 s, 2 s, 1 s
158	$max\_F\_yaw_t^5$	max *yawRate* FFT coefficients in TW 5 s
159-162	⋮	$max\_F\_yaw_t$ in TW 4 s, 3 s, 2 s, 1 s
163	$max\_F\_a_t^5$	max acceleration FFT coefficients in TW 5 s
164-167	⋮	$max\_F\_a_t$ in TW 4 s, 3 s, 2 s, 1 s
168	$max\_F\_brpd_t^5$	max *brpd* FFT coefficients in TW 5 s
169-172	⋮	$max\_F\_brpd_t$ in TW 4 s, 3 s, 2 s, 1 s
173	$max\_F\_acpd_t^5$	max *acpd* FFT coefficients in TW 5 s
174-177	⋮	$max\_F\_acpd_t$ in TW 4 s, 3 s, 2 s, 1 s
178	$max\_F\_stw_t^5$	max *stw* FFT coefficients in TW 5 s
179-182	⋮	$max\_F\_stw_t$ in TW 4 s, 3 s, 2 s, 1 s
183	$max\_F\_stwRate_t^5$	max *stwRate* FFT coefficients in TW 5 s
184-187	⋮	$max\_F\_stwRate_t$ in TW 4 s, 3 s, 2 s, 1 s

C.3.2 Result of feature selection

Table C.2 and Table C.3 list all the p-value and Cohen's d for LLC and RLC scenario. Feature number in these two tables can be matched exactly with Table C.1. Feature whose Cohen's d shows ' - ' is a invalid value.

Table C.2: The full scale effect size of the extracted feature for LLC case.

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	p	d	p	d	p	d	p	d	p	d	p
1	1.45	0.05	0.97	0.06	1.02	0.06	1.29	0.03	1.67	0.05	1.29	0.11
2	0.54	0.26	0.60	0.11	0.54	0.19	0.52	0.22	0.53	0.25	0.42	0.39
3	2.30	0.03	2.15	0.03	2.24	0.02	2.29	0.03	2.44	0.04	2.56	0.03
4	-	<0.01	-	<0.01	-	0.01	-	0.02	-	<0.01	-	<0.01
5	-	0.05	1.42	0.07	1.38	0.07	1.46	0.05	-	0.03	-	0.04
6	2.54	0.03	2.37	0.02	2.45	0.01	2.55	0.02	2.63	0.03	2.65	0.03
7	-	0.12	-	0.12	-	0.13	-	0.10	-	0.10	-	0.03
8	2.23	0.03	1.74	0.03	1.74	0.02	1.79	0.03	2.34	0.01	2.95	0.01
9	2.15	0.03	1.61	0.03	1.65	0.04	1.79	0.03	2.23	0.03	2.92	0.01
10	2.08	0.06	1.51	0.02	1.56	0.05	1.73	0.05	2.24	0.04	2.86	0.02
11	2.02	0.02	1.30	0.04	1.38	0.03	1.60	0.02	2.17	0.03	2.82	0.01
12	1.98	0.03	1.11	0.05	1.21	0.06	1.45	0.03	2.08	0.02	2.25	0.03
13	1.97	0.04	1.49	0.02	1.57	0.03	1.60	0.03	1.97	0.04	2.36	0.03
14	1.90	0.02	1.36	0.03	1.41	0.05	1.51	0.05	1.97	0.02	2.39	0.03
15	1.66	0.04	1.14	0.04	1.23	0.05	1.52	0.05	1.84	0.07	2.11	0.06
16	1.42	0.09	0.89	0.07	1.04	0.09	1.29	0.04	1.45	0.08	2.06	0.05
17	1.14	0.07	0.62	0.11	0.72	0.07	0.79	0.12	1.02	0.13	1.58	0.08
18	1.86	0.10	1.58	0.03	1.64	0.03	1.74	0.02	1.99	0.06	-	0.19
19	1.92	0.07	1.49	0.03	1.60	0.04	1.73	0.02	2.15	0.02	-	0.14
20	2	0.05	1.41	0.03	1.49	0.04	1.76	0.05	2.20	0.02	-	0.11
21	2.02	0.02	1.27	0.05	1.44	0.03	1.70	0.02	2.16	0.03	-	0.02
22	-	0.03	1.14	0.05	1.28	0.06	1.51	0.05	2.03	0.02	2.05	0.04
23	-	0.18	1.78	0.04	1.86	0.05	1.79	0.02	-	0.14	-	0.47
24	-	0.12	1.67	0.05	1.83	0.02	1.83	0.04	-	0.03	-	0.35
25	-	0.04	1.53	0.05	1.66	0.05	1.85	0.03	1.82	0.03	-	0.20
26	-	0.04	1.24	0.06	1.31	0.05	1.55	0.07	1.74	0.05	1.86	0.03
27	1.70	0.04	1.06	0.06	1.13	0.08	1.31	0.04	1.84	0.04	1.97	0.05
28	2.11	0.03	1.72	0.03	1.68	0.03	1.79	0.04	2.12	0.04	-	0.02
29	2.06	0.03	1.55	0.04	1.61	0.04	1.79	0.03	2.15	0.01	2.59	0.01
30	1.86	0.03	1.40	0.04	1.51	0.04	1.68	0.05	2.05	0.05	2.44	0.02
31	1.87	0.03	1.26	0.04	1.36	0.05	1.58	0.04	2.07	0.02	2.51	0.02
32	1.84	0.04	1.06	0.06	1.15	0.07	1.37	0.03	1.96	0.03	2.12	0.03
33	2.48	0.02	2.11	0.03	2.33	0.02	2.55	0.01	2.43	0.01	2.65	0.02
34	2.27	0.04	2.05	0.04	2.29	0.01	2.37	0.02	2.28	0.04	2.52	0.01
35	2.09	0.03	1.98	0.01	2.08	0.03	2.21	0.01	2.17	0.03	2.50	0.02
36	1.85	0.06	1.81	0.03	1.84	0.03	1.85	0.04	1.91	0.04	2.34	0.03
37	1.78	0.05	1.60	0.02	1.59	0.03	1.59	0.06	1.71	0.06	2.07	0.06

Continued on next page

Table C.2 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
38	1.94	0.04	1.77	0.03	1.80	0.02	1.94	0.04	1.98	0.04	2.12	0.05
39	1.95	0.03	1.61	0.03	1.72	0.03	1.97	0.04	1.91	0.05	2.19	0.04
40	1.76	0.04	1.42	0.06	1.66	0.03	1.85	0.05	1.86	0.03	2.11	0.04
41	1.61	0.06	1.23	0.08	1.45	0.05	1.61	0.03	1.71	0.05	1.86	0.05
42	1.30	0.07	0.97	0.08	1.09	0.06	1.15	0.06	1.26	0.08	1.59	0.08
43	-	0.05	1.69	0.02	1.81	0.02	1.87	0.03	-	0.05	-	0.06
44	-	0.03	1.67	0.04	1.78	0.03	1.81	0.03	-	0.03	-	0.07
45	-	0.05	1.59	0.02	1.60	0.02	1.71	0.04	-	0.03	-	0.04
46	-	0.04	1.37	0.04	1.38	0.07	1.49	0.08	1.69	0.09	-	0.03
47	1.40	0.06	1.17	0.05	1.15	0.07	1.17	0.07	1.42	0.07	1.72	0.04
48	-	0.04	1.70	0.03	1.70	0.03	1.73	0.04	-	0.09	-	0.10
49	-	0.02	1.64	0.05	1.69	0.03	1.86	0.04	1.73	0.01	-	0.09
50	-	0.04	1.46	0.03	1.56	0.05	1.86	0.04	1.91	0.01	-	0.06
51	-	0.05	1.34	0.05	1.42	0.02	1.59	0.04	1.78	0.06	-	0.03
52	-	0.07	1.11	0.07	1.14	0.05	1.26	0.06	1.40	0.06	1.63	0.08
53	-	0.03	2.12	0.01	2.25	0.03	2.39	0.01	2.21	0.03	-	0.04
54	-	0.04	2.05	0.04	2.17	0.02	2.27	0.01	2.15	0.03	-	0.04
55	1.91	0.05	1.92	0.04	1.98	0.02	2.10	0.03	1.98	0.04	-	0.03
56	1.72	0.05	1.72	0.03	1.75	0.02	1.77	0.05	1.81	0.04	2.16	0.03
57	1.63	0.04	1.47	0.04	1.47	0.05	1.42	0.06	1.54	0.08	1.93	0.07
58	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
59	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
60	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
61	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.02	-	<0.01
62	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
63	-	0.03	-	<0.01	-	<0.01	-	0.02	-	<0.01	-	<0.01
64	-	<0.01	-	<0.01	-	0.08	-	<0.01	-	0.02	-	<0.01
65	-	<0.01	-	<0.01	-	0.02	-	<0.01	-	<0.01	-	<0.01
66	-	<0.01	-	0.02	-	0.02	-	0.07	-	<0.01	-	0.03
67	-	<0.01	-	0.04	-	0.05	-	<0.01	-	0.02	-	<0.01
68	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
69	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.02	-	<0.01
70	-	0.01	-	0.01	-	0.01	-	0.01	-	0.01	-	<0.01
71	-	0.01	-	<0.01	-	<0.01	-	0.02	-	0.06	-	0.01
72	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
73	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.11
74	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.11
75	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.11
76	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
77	-	<0.01	-	<0.01	-	<0.01	-	0.02	-	<0.01	-	<0.01
78	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
79	-	0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.02
80	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
81	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.05	-	<0.01

Continued on next page

Table C.2 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
82	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
83	-	0.01	-	0.01	-	0.02	-	0.01	-	0.02	-	0.01
84	-	0.01	-	0.01	-	0.01	-	<0.01	-	0.02	-	0.01
85	-	0.02	-	0.01	-	0.03	-	0.01	-	0.03	-	<0.01
86	-	0.01	-	0.02	-	0.01	-	0.02	-	0.03	-	<0.01
87	-	0.01	-	0.02	-	0.02	-	0.03	-	0.01	-	0.01
88	-	0.02	-	0.03	-	0.02	-	0.02	-	0.02	-	0.01
89	-	0.02	-	0.03	-	0.02	-	0.02	-	0.01	-	0.02
90	-	0.03	-	0.03	-	0.02	-	0.01	-	0.04	-	0.02
91	-	0.05	-	0.03	-	0.04	-	0.01	-	0.02	-	0.04
92	-	0.05	-	0.03	-	0.04	-	0.04	-	0.03	-	0.03
93	-	0.04	-	0.01	-	0.02	-	0.02	-	0.03	-	0.12
94	-	0.03	-	0.01	-	0.01	-	0.01	-	0.02	-	0.09
95	-	0.02	-	0.01	-	0.02	-	0.01	-	0.01	-	0.04
96	-	0.01	-	0.01	-	0.02	-	0.01	-	0.01	-	0.02
97	-	0.01	-	0.03	-	0.02	-	0.01	-	0.03	-	0.01
98	-	0.06	-	<0.01	-	0.01	-	<0.01	-	0.04	-	0.13
99	-	0.03	-	0.01	-	<0.01	-	0.01	-	0.01	-	0.11
100	-	0.01	-	0.02	-	0.01	-	0.01	-	0.02	-	0.09
101	-	0.01	-	0.02	-	0.01	-	<0.01	-	0.01	-	0.05
102	-	0.02	-	0.01	-	0.02	-	0.02	-	0.02	-	0.02
103	-	0.01	-	<0.01	-	0.01	-	0.01	-	0.01	-	0.07
104	-	0.01	-	0.01	-	0.01	-	<0.01	-	0.01	-	0.03
105	-	0.02	-	0.01	-	0.03	-	0.01	-	0.01	-	0.02
106	-	0.02	-	0.03	-	0.01	-	0.02	-	0.02	-	0.02
107	-	0.02	-	0.02	-	0.02	-	0.03	-	0.02	-	0.03
108	2.87	0.02	2.80	<0.01	2.72	0.01	2.78	0.01	3.03	0.01	3.17	0.01
109	2.78	0.01	2.72	0.01	2.65	0.02	2.77	0.01	2.90	0.01	3.08	0.01
110	2.76	0.02	2.62	0.02	2.64	0.01	2.70	<0.01	2.90	0.01	3.13	0.01
111	2.73	0.01	2.52	0.03	2.56	0.02	2.64	0.02	2.90	0.02	3.12	0.01
112	2.62	0.04	2.41	0.02	2.51	<0.01	2.60	0.02	2.80	<0.01	2.88	0.02
113	2.65	0.03	2.54	0.02	2.50	0.02	2.51	0.02	2.73	0.02	2.95	0.01
114	2.63	0.01	2.43	0.03	2.41	0.03	2.43	0.03	2.71	0.02	2.85	0.01
115	2.55	0.03	2.35	0.02	2.32	0.02	2.42	0.04	2.66	0.02	2.88	0.01
116	2.54	0.02	2.21	0.03	2.24	0.04	2.33	0.03	2.55	0.02	2.76	0.02
117	2.28	0.04	2.12	0.05	2.16	0.05	2.21	0.04	2.37	0.03	2.56	0.03
118	-	0.01	2.68	0.01	2.70	<0.01	2.64	0.02	-	0.01	-	0.03
119	-	0.01	2.62	0.02	2.63	0.01	2.70	0.01	-	0.01	-	0.02
120	-	0.01	2.56	0.02	2.61	0.01	2.71	0.01	2.88	<0.01	-	0.01
121	2.80	<0.01	2.52	0.01	2.60	0.01	2.69	0.01	2.92	0.01	-	0.01
122	2.73	0.02	2.46	0.03	2.58	0.02	2.66	0.02	2.88	0.01	2.77	0.01
123	-	0.01	2.75	0.02	2.77	0.01	2.59	0.01	-	0.01	-	0.04
124	-	0.01	2.75	0.02	2.70	<0.01	2.56	0.01	-	0.01	-	0.04
125	-	0.01	2.64	0.01	2.56	0.01	2.65	0.02	2.62	0.01	-	0.03

Continued on next page

Table C.2 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
126	-	0.02	2.48	0.01	2.44	0.03	2.59	0.03	2.60	0.03	2.69	0.03
127	2.51	0.02	2.36	0.02	2.43	0.01	2.50	0.03	2.65	0.02	2.73	0.03
128	-	0.01	2.75	0.01	2.69	0.01	2.72	0.01	-	0.01	-	0.02
129	2.74	0.02	2.69	0.01	2.64	0.01	2.72	0.02	2.85	0.01	2.91	0.02
130	2.68	0.01	2.58	0.02	2.60	0.02	2.68	0.03	2.82	0.01	-	0.01
131	2.62	0.02	2.47	0.02	2.53	0.01	2.62	0.02	2.82	0.02	2.94	0.01
132	2.57	0.04	2.39	0.02	2.48	0.01	2.56	0.02	2.74	0.01	2.77	0.03
133	-	0.04	-	0.03	-	0.01	-	0.03	-	0.05	-	0.04
134	-	0.03	-	0.02	-	0.03	-	0.04	-	0.03	-	0.04
135	-	0.02	-	0.03	-	0.04	-	0.03	-	0.02	-	0.03
136	-	0.04	-	0.02	-	0.05	-	0.05	-	0.03	-	0.03
137	-	0.06	-	0.06	-	0.07	-	0.07	-	0.05	-	0.03
138	-	0.03	-	0.01	-	0.01	-	0.02	-	0.02	-	0.02
139	-	0.02	-	0.03	-	0.02	-	0.01	-	0.02	-	0.01
140	-	0.03	-	0.02	-	0.02	-	0.01	-	0.03	-	0.01
141	-	0.01	-	0.03	-	0.04	-	0.01	-	0.01	-	0.02
142	-	0.03	-	0.05	-	0.03	-	0.03	-	0.04	-	0.02
143	-	0.04	-	0.04	-	0.01	-	0.04	-	0.07	-	0.16
144	-	0.03	-	0.01	-	0.02	-	0.02	-	<0.01	-	0.10
145	-	0.02	-	0.02	-	0.02	-	0.02	-	0.02	-	0.10
146	-	0.02	-	0.03	-	0.03	-	0.02	-	0.02	-	0.02
147	-	0.03	-	0.06	-	0.03	-	0.05	-	0.03	-	0.02
148	-	0.07	-	0.02	-	<0.01	-	0.03	-	0.07	-	0.18
149	-	0.06	-	0.02	-	0.02	-	0.01	-	0.01	-	0.15
150	-	0.06	-	0.01	-	0.03	-	0.02	-	0.02	-	0.08
151	-	0.04	-	0.06	-	0.04	-	0.03	-	0.03	-	0.02
152	-	0.06	-	0.05	-	0.03	-	0.06	-	0.03	-	0.03
153	-	0.02	-	0.01	-	0.01	-	0.01	-	0.03	-	0.01
154	-	0.02	-	0.02	-	0.01	-	0.01	-	0.01	-	0.02
155	-	0.03	-	0.02	-	0.02	-	0.02	-	0.02	-	0.02
156	-	0.03	-	0.02	-	0.04	-	0.05	-	0.03	-	0.02
157	-	0.06	-	0.03	-	0.06	-	0.07	-	0.06	-	0.02
158	1.85	0.05	1.58	0.02	1.57	0.02	1.60	0.04	1.90	0.04	2.39	0.05
159	1.76	0.04	1.47	0.04	1.48	0.04	1.57	0.04	1.85	0.05	2.43	0.04
160	1.67	0.05	1.30	0.03	1.35	0.05	1.48	0.05	1.89	0.05	2.36	0.04
161	1.63	0.04	1.11	0.08	1.27	0.04	1.43	0.03	1.78	0.04	2.36	0.04
162	1.72	0.04	1.01	0.03	1.14	0.08	1.28	0.06	1.75	0.03	2.15	0.03
163	1.94	0.05	1.77	0.03	1.78	0.03	1.92	0.03	2.03	0.05	2.22	0.04
164	1.91	0.03	1.70	0.04	1.72	0.02	1.83	0.03	2	0.05	2.05	0.05
165	1.72	0.06	1.53	0.02	1.56	0.06	1.82	0.03	1.86	0.06	1.87	0.06
166	1.55	0.06	1.31	0.06	1.43	0.06	1.68	0.04	1.49	0.06	1.94	0.06
167	1.35	0.07	1.06	0.08	1.18	0.05	1.26	0.06	1.22	0.07	1.56	0.09
168	-	<0.01	-	0.31	-	0.25	-	0.18	-	0.33	-	0.19
169	-	<0.01	-	0.31	-	0.25	-	0.20	-	0.33	-	0.18

Continued on next page

Table C.2 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
170	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
171	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.02	-	<0.01
172	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
173	-	0.01	-	<0.01	-	0.01	-	0.01	-	0.02	-	0.01
174	-	0.02	-	0.02	-	0.02	-	0.01	-	0.01	-	0.01
175	-	0.02	-	0.01	-	0.03	-	0.01	-	0.03	-	<0.01
176	-	0.01	-	0.02	-	0.01	-	0.02	-	0.03	-	<0.01
177	-	0.01	-	0.02	-	0.01	-	0.03	-	0.01	-	0.01
178	2.81	0.02	2.79	0.01	2.72	0.01	2.78	0.01	2.97	0.01	3.11	0.01
179	2.76	0.01	2.72	0.01	2.67	0.02	2.75	0.01	2.88	0.01	3.02	0.01
180	2.76	0.01	2.65	0.01	2.65	0.02	2.67	0.01	2.88	0.02	3.04	0.01
181	2.73	0.02	2.53	0.03	2.55	0.02	2.61	0.01	2.89	0.02	3.05	0.01
182	2.61	0.03	2.41	0.02	2.49	0.01	2.59	0.02	2.79	<0.01	2.85	0.03
183	-	0.01	-	0.01	-	0.04	-	0.01	-	0.03	-	0.02
184	-	0.03	-	0.02	-	0.02	-	0.01	-	0.02	-	0.02
185	-	0.03	-	0.04	-	0.03	-	0.01	-	0.03	-	0.03
186	-	0.04	-	0.03	-	0.02	-	0.03	-	0.02	-	0.03
187	-	0.03	-	0.03	-	0.03	-	0.04	-	0.04	-	0.04

Table C.3: The full scale effect size of the extracted feature for RLC case.

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
1	1.22	0.05	1.01	0.04	1.09	0.04	1.29	0.02	1.19	0.08	1.55	0.06
2	0.40	0.29	0.57	0.15	0.49	0.17	0.42	0.26	0.38	0.35	0.32	0.47
3	2.52	0.02	2.44	0.01	2.46	0.01	2.51	0.02	2.51	0.02	2.52	0.03
4	-	<0.01	-	0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
5	-	0.06	1.58	0.02	1.51	0.03	1.40	0.06	-	0.05	-	0.04
6	2.36	0.02	2.40	0.03	2.39	0.02	2.53	0.01	2.30	0.05	2.80	0.01
7	-	0.12	-	0.17	-	0.15	-	0.11	-	0.06	-	0.10
8	2.09	0.02	1.47	0.03	1.68	0.03	2.01	0.02	2.58	0.01	2.75	0.02
9	2.14	0.02	1.44	0.03	1.64	0.03	1.97	0.02	2.69	0.01	2.67	0.02
10	2.24	0.03	1.37	0.03	1.61	0.04	2.03	0.01	2.60	0.01	2.64	0.02
11	2.17	0.02	1.33	0.02	1.61	0.01	1.95	0.05	2.41	0.01	2.37	0.03
12	1.82	0.03	1.21	0.03	1.42	0.01	1.68	0.02	1.96	0.04	1.94	0.07
13	1.96	0.02	1.56	0.03	1.66	0.03	1.90	0.03	2.24	0.02	2	0.06
14	1.92	0.02	1.44	0.04	1.58	0.04	1.89	0.03	2.03	0.05	2.10	0.05
15	1.79	0.03	1.29	0.03	1.46	0.05	1.66	0.03	1.88	0.05	2.02	0.04
16	1.37	0.07	1	0.06	1.13	0.06	1.26	0.07	1.49	0.06	1.81	0.07
17	0.98	0.10	0.65	0.08	0.70	0.10	0.80	0.09	1.05	0.09	1.20	0.11
18	-	0.10	1.62	0.05	1.76	0.04	1.83	0.01	-	0.13	-	0.37
19	-	0.10	1.49	0.02	1.62	0.03	1.79	0.02	1.56	0.02	-	0.26
20	-	0.05	1.29	0.05	1.50	0.05	1.68	0.04	1.73	0.04	-	0.16
21	1.58	0.05	1.12	0.04	1.27	0.04	1.53	0.03	1.76	0.03	1.74	0.05
22	1.63	0.03	1.05	0.04	1.24	0.05	1.53	0.01	1.75	0.05	1.70	0.07
23	-	0.03	1.62	0.02	1.80	0.04	2.04	0.01	2.18	0.06	-	0.25
24	2.11	0.03	1.59	0.05	1.76	0.03	2.05	0.01	2.31	0.01	-	0.19
25	2.17	0.02	1.51	0.02	1.77	0.01	2.05	0.02	2.32	0.02	-	0.16
26	2.10	0.03	1.47	0.02	1.69	0.04	1.96	0.02	2.23	0.02	1.72	0.02
27	1.74	0.03	1.30	0.02	1.45	0.03	1.65	0.03	1.86	0.04	1.82	0.06
28	1.92	0.02	1.43	0.04	1.54	0.06	1.84	0.02	2.32	0.03	2.47	0.02
29	2.05	0.03	1.33	0.04	1.51	0.03	1.82	0.03	2.39	0.02	2.33	0.02
30	2.04	0.02	1.26	0.03	1.47	0.03	1.88	0.02	2.35	0.01	2.30	0.02
31	2.05	0.02	1.21	0.03	1.46	0.03	1.86	0.02	2.30	0.03	2.21	0.05
32	1.70	0.03	1.13	0.03	1.33	0.02	1.56	0.03	1.79	0.03	1.85	0.07
33	2.50	0.02	2.27	0.02	2.36	0.03	2.53	0.02	2.68	0.01	2.51	0.01
34	2.39	0.02	2.15	0.01	2.22	0.01	2.46	0.01	2.43	0.02	2.47	0.03
35	2.18	0.03	1.99	0.04	2.10	0.04	2.27	0.01	2.27	0.03	2.34	0.01
36	1.94	0.02	1.90	0.02	1.98	0.02	1.94	0.05	1.96	0.04	2.19	0.04
37	1.60	0.03	1.67	0.04	1.69	0.03	1.61	0.03	1.66	0.05	1.78	0.07
38	1.95	0.04	1.79	0.02	1.86	0.04	1.98	0.03	2.16	0.03	2.03	0.05
39	1.76	0.05	1.63	0.04	1.78	0.04	1.99	0.03	1.87	0.05	1.98	0.06
40	1.70	0.06	1.55	0.04	1.67	0.05	1.79	0.03	1.72	0.05	2.02	0.04
41	1.45	0.07	1.37	0.05	1.42	0.04	1.39	0.08	1.74	0.03	1.74	0.07
42	1.15	0.10	1.05	0.07	0.99	0.08	1.07	0.07	1.30	0.07	1.59	0.10
43	-	0.04	1.80	0.04	1.84	0.04	1.84	0.03	-	0.04	-	0.03

Continued on next page

Table C.3 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
44	-	0.03	1.80	0.05	1.81	0.04	1.97	0.02	-	0.03	-	0.03
45	1.68	0.04	1.69	0.03	1.74	0.04	1.83	0.03	-	0.04	-	0.04
46	1.48	0.05	1.49	0.03	1.49	0.03	1.48	0.05	1.70	0.06	-	0.04
47	1.24	0.05	1.20	0.04	1.11	0.05	1.12	0.08	1.39	0.05	1.64	0.08
48	-	0.03	1.81	0.03	1.87	0.03	1.81	0.03	-	0.06	-	0.06
49	-	0.03	1.69	0.03	1.77	0.02	1.91	0.03	-	0.02	-	0.05
50	-	0.05	1.59	0.04	1.68	0.04	1.72	0.05	1.85	0.02	-	0.06
51	1.55	0.04	1.51	0.02	1.47	0.05	1.49	0.06	1.68	0.04	1.58	0.06
52	1.15	0.08	1.27	0.04	1.18	0.05	1.17	0.08	1.25	0.08	1.59	0.05
53	2.25	0.03	2.17	0.02	2.20	0.03	2.28	0.03	2.26	0.03	-	0.02
54	2.20	0.02	2.07	0.02	2.10	0.02	2.27	0.01	2.09	0.03	2.13	0.03
55	2	0.03	1.91	0.03	1.98	0.03	2.04	0.02	2	0.02	-	0.04
56	1.82	0.03	1.81	0.02	1.83	0.02	1.80	0.04	1.84	0.05	-	0.04
57	1.36	0.07	1.52	0.03	1.46	0.04	1.43	0.06	1.38	0.05	1.60	0.09
58	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
59	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.01
60	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
61	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.01	-	0.01
62	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
63	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.05	-	<0.01
64	-	<0.01	-	<0.01	-	<0.01	-	0.03	-	<0.01	-	<0.01
65	-	<0.01	-	<0.01	-	0.01	-	<0.01	-	<0.01	-	<0.01
66	-	<0.01	-	0.01	-	0.05	-	0.05	-	<0.01	-	<0.01
67	-	<0.01	-	0.05	-	0.02	-	0.01	-	0.03	-	<0.01
68	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
69	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
70	-	<0.01	-	0.01	-	<0.01	-	0.01	-	0.01	-	<0.01
71	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
72	-	<0.01	-	<0.01	-	<0.01	-	0.01	-	<0.01	-	<0.01
73	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.05	-	<0.01
74	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.01	-	0.01
75	-	<0.01	-	<0.01	-	0.01	-	<0.01	-	<0.01	-	<0.01
76	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
77	-	<0.01	-	0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
78	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.08
79	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
80	-	<0.01	-	<0.01	-	<0.01	-	0.02	-	0.01	-	<0.01
81	-	<0.01	-	<0.01	-	0.01	-	<0.01	-	<0.01	-	<0.01
82	-	0.02	-	<0.01	-	0.01	-	0.01	-	0.02	-	0.02
83	-	0.01	-	0.01	-	<0.01	-	0.01	-	0.02	-	0.01
84	-	0.01	-	0.01	-	0.02	-	0.01	-	<0.01	-	0.01
85	-	0.01	-	0.01	-	0.02	-	0.01	-	0.02	-	<0.01
86	-	0.01	-	0.01	-	0.01	-	0.02	-	<0.01	-	0.01
87	-	0.01	-	0.01	-	0.02	-	0.01	-	0.02	-	0.02

Continued on next page

Table C.3 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
88	-	0.01	-	0.02	-	0.03	-	0.02	-	0.02	-	0.02
89	-	0.01	-	0.03	-	0.03	-	0.02	-	<0.01	-	0.02
90	-	0.02	-	0.02	-	0.02	-	0.01	-	0.02	-	0.01
91	-	0.01	-	0.03	-	0.02	-	0.04	-	0.02	-	0.02
92	-	0.05	-	0.05	-	0.04	-	0.04	-	0.03	-	0.05
93	-	0.03	-	0.02	-	0.01	-	0.02	-	0.08	-	0.18
94	-	0.04	-	0.03	-	0.01	-	0.02	-	0.04	-	0.14
95	-	0.01	-	0.01	-	<0.01	-	0.01	-	0.01	-	0.09
96	-	0.02	-	0.01	-	0.01	-	0.02	-	0.02	-	0.06
97	-	0.01	-	0.03	-	0.02	-	0.01	-	0.01	-	0.02
98	-	0.03	-	<0.01	-	0.01	-	0.01	-	0.07	-	0.12
99	-	0.03	-	0.01	-	0.01	-	0.02	-	0.04	-	0.08
100	-	0.02	-	0.01	-	0.02	-	0.01	-	0.02	-	0.05
101	-	0.02	-	<0.01	-	0.02	-	0.03	-	0.02	-	0.04
102	-	0.01	-	0.01	-	0.02	-	0.02	-	0.02	-	0.01
103	-	<0.01	-	0.01	-	0.01	-	0.01	-	0.01	-	0.06
104	-	0.01	-	0.01	-	<0.01	-	<0.01	-	0.01	-	0.02
105	-	0.01	-	0.02	-	0.02	-	0.01	-	0.02	-	0.04
106	-	0.01	-	0.02	-	0.01	-	0.02	-	0.02	-	0.03
107	-	0.02	-	0.01	-	0.03	-	0.01	-	0.02	-	0.02
108	2.70	0.02	2.55	0.02	2.64	0.01	2.85	<0.01	3.09	<0.01	3.04	0.01
109	2.82	0.02	2.55	0.02	2.63	0.02	2.82	0.01	3.10	0.01	3.07	0.01
110	2.83	0.01	2.53	0.01	2.60	0.01	2.83	0.01	3.12	0.01	2.97	0.01
111	2.78	0.01	2.51	0.01	2.57	0.01	2.80	0.02	3.01	0.02	2.77	0.02
112	2.63	0.01	2.47	0.01	2.52	0.01	2.74	0.02	2.66	0.02	2.67	0.02
113	2.55	0.02	2.57	0.02	2.57	<0.01	2.71	0.01	2.81	0.02	2.76	0.02
114	2.48	0.02	2.51	0.02	2.51	0.01	2.67	0.01	2.67	0.03	2.71	0.03
115	2.49	0.03	2.42	0.01	2.45	0.01	2.56	0.04	2.61	0.03	2.65	0.03
116	2.33	0.04	2.35	0.02	2.30	0.04	2.40	0.03	2.40	0.05	2.64	0.01
117	2.17	0.04	2.22	0.03	2.14	0.04	2.16	0.06	2.17	0.08	2.36	0.05
118	-	0.03	2.63	0.01	2.50	<0.01	2.48	0.01	-	0.02	-	0.03
119	2.52	0.02	2.56	0.02	2.48	0.02	2.57	0.01	-	0.01	-	0.02
120	-	0.02	2.47	0.03	2.47	0.02	2.60	0.02	2.62	0.02	-	0.02
121	-	0.02	2.40	0.03	2.45	0.02	2.61	0.01	2.62	0.02	-	0.03
122	2.57	0.01	2.40	0.02	2.44	0.01	2.62	0.01	2.62	0.02	2.68	0.03
123	-	0.01	2.61	0.01	2.74	0.01	2.93	<0.01	-	0.01	-	0.03
124	-	0.02	2.60	0.01	2.75	<0.01	2.92	0.01	-	<0.01	-	0.02
125	2.72	0.01	2.59	0.01	2.71	<0.01	2.82	0.02	2.82	0.01	-	0.01
126	2.77	<0.01	2.58	<0.01	2.63	0.02	2.79	0.02	2.86	0.01	-	0.02
127	2.62	0.02	2.52	0.01	2.55	0.01	2.78	0.01	2.65	0.02	2.61	0.02
128	-	0.01	2.49	0.02	2.52	0.01	2.64	0.03	-	0.01	-	0.01
129	2.77	0.01	2.48	0.02	2.51	0.02	2.68	0.01	2.93	<0.01	-	<0.01
130	2.71	0.02	2.45	0.02	2.50	0.02	2.69	0.02	2.97	0.01	-	0.01
131	2.72	0.02	2.44	0.01	2.49	0.01	2.73	0.02	2.93	0.01	2.64	0.02

Continued on next page

Table C.3 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
132	2.58	0.02	2.42	0.01	2.46	0.01	2.68	0.01	2.61	0.03	2.58	0.01
133	-	0.02	-	0.01	-	0.01	-	0.02	-	0.02	-	0.03
134	-	0.04	-	0.02	-	0.02	-	0.03	-	0.04	-	0.04
135	-	0.03	-	0.02	-	0.03	-	0.03	-	0.06	-	0.03
136	-	0.06	-	0.03	-	0.03	-	0.06	-	0.06	-	0.02
137	-	0.07	-	0.09	-	0.08	-	0.09	-	0.04	-	0.03
138	-	0.02	-	<0.01	-	0.02	-	0.01	-	0.01	-	0.02
139	-	0.02	-	0.02	-	0.01	-	0.01	-	0.03	-	0.02
140	-	0.03	-	0.03	-	0.02	-	0.02	-	0.04	-	0.03
141	-	0.06	-	0.02	-	0.02	-	0.02	-	0.03	-	0.04
142	-	0.03	-	0.04	-	0.06	-	0.07	-	0.03	-	0.03
143	-	0.02	-	0.02	-	0.01	-	0.01	-	0.05	-	0.10
144	-	0.02	-	0.01	-	0.03	-	0.02	-	0.02	-	0.07
145	-	0.03	-	0.03	-	0.01	-	0.02	-	0.02	-	0.06
146	-	0.03	-	0.06	-	0.02	-	0.05	-	0.05	-	0.03
147	-	0.07	-	0.03	-	0.03	-	0.05	-	0.06	-	0.03
148	-	0.04	-	0.01	-	0.02	-	0.01	-	0.08	-	0.25
149	-	0.04	-	0.02	-	0.01	-	0.03	-	0.01	-	0.21
150	-	0.04	-	0.03	-	0.04	-	0.03	-	0.02	-	0.13
151	-	0.03	-	0.05	-	0.02	-	0.03	-	0.03	-	0.01
152	-	0.03	-	0.06	-	0.05	-	0.05	-	0.03	-	0.03
153	-	0.02	-	0.01	-	0.02	-	0.03	-	0.01	-	0.03
154	-	0.03	-	0.02	-	0.01	-	0.02	-	0.02	-	0.01
155	-	0.02	-	0.02	-	0.02	-	0.02	-	0.03	-	0.01
156	-	0.03	-	0.04	-	0.06	-	0.04	-	0.03	-	0.02
157	-	0.07	-	0.06	-	0.06	-	0.07	-	0.02	-	0.03
158	1.86	0.04	1.51	0.03	1.59	0.04	1.75	0.04	2.17	0.03	2.23	0.05
159	1.85	0.02	1.48	0.02	1.58	0.04	1.85	0.03	2.12	0.04	2.20	0.04
160	1.85	0.03	1.31	0.06	1.53	0.03	1.85	0.02	2.03	0.03	2.07	0.07
161	1.70	0.04	1.23	0.05	1.46	0.04	1.68	0.03	1.97	0.06	2.03	0.03
162	1.50	0.05	1.12	0.05	1.28	0.04	1.41	0.05	1.69	0.05	1.84	0.07
163	1.83	0.03	1.78	0.02	1.77	0.03	1.86	0.05	2.04	0.03	1.95	0.04
164	1.73	0.04	1.68	0.04	1.77	0.04	1.86	0.04	1.93	0.04	1.86	0.07
165	1.61	0.06	1.64	0.03	1.66	0.04	1.61	0.06	1.81	0.05	1.89	0.04
166	1.59	0.05	1.53	0.03	1.46	0.08	1.44	0.04	1.70	0.06	1.81	0.06
167	1.26	0.06	1.22	0.06	1.12	0.07	1.16	0.06	1.38	0.06	1.52	0.09
168	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
169	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.01
170	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
171	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	0.01	-	0.01
172	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01	-	<0.01
173	-	0.01	-	<0.01	-	<0.01	-	0.01	-	0.02	-	0.01
174	-	<0.01	-	0.02	-	0.01	-	0.02	-	<0.01	-	0.01
175	-	0.01	-	<0.01	-	0.02	-	0.01	-	0.01	-	<0.01

Continued on next page

Table C.3 – Continued from previous page

#	GBL		TWL_5		TWL_4		TWL_3		TWL_2		TWL_1	
	d	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*	*d*	*p*
176	-	0.01	-	0.01	-	0.01	-	0.02	-	<0.01	-	0.01
177	-	0.01	-	<0.01	-	0.02	-	0.01	-	0.02	-	0.02
178	2.70	0.02	2.55	0.02	2.64	0.01	2.85	<0.01	3.09	<0.01	3.04	0.01
179	2.82	0.01	2.55	0.02	2.63	0.02	2.82	0.01	3.10	0.01	3.07	0.01
180	2.83	<0.01	2.53	0.01	2.60	0.01	2.83	0.01	3.12	0.01	2.97	0.01
181	2.78	0.01	2.51	0.01	2.57	0.01	2.80	0.02	3.01	0.02	2.77	0.02
182	2.63	0.01	2.47	0.01	2.52	0.01	2.74	0.02	2.66	0.02	2.67	0.02
183	-	0.01	-	0.02	-	0.02	-	0.02	-	0.03	-	0.03
184	-	0.03	-	0.03	-	0.02	-	0.03	-	0.02	-	0.02
185	-	0.03	-	0.02	-	0.06	-	0.01	-	0.04	-	0.03
186	-	0.04	-	0.04	-	0.03	-	0.01	-	0.04	-	0.02
187	-	0.04	-	0.04	-	0.04	-	0.07	-	0.05	-	0.05

References

Ahie, Liv M, Charlton, Samuel G, and Starkey, Nicola J (2015). "The role of preference in speed choice". In: *Transportation Research Part F: Traffic Psychology and Behaviour* 30, pp. 66–73.

Alemdar, Ali and Ibnkahla, Mohamed (2007). "Wireless sensor networks: Applications and challenges". In: *2007 9th International Symposium on Signal Processing and Its Applications*. IEEE, pp. 1–6.

Audi (2013). *Q7-Side-Assist-Innen-V4*.

Ayres, TJ, Li, L, Schleuning, D, and Young, D (2001). "Preferred time-headway of highway drivers". In: *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*. IEEE, pp. 826–829.

Barr, Lawrence and Najm, Wassim (2001). "Crash problem characteristics for the intelligent vehicle initiative". In: *Transportation Research Board 80th Annual Meeting*.

Barrachina, Javier, Garrido, Piedad, Fogue, Manuel, Martinez, Francisco J, Cano, Juan-Carlos, Calafate, Carlos T, and Manzoni, Pietro (2015). "A V2I-based real-time traffic density estimation system in urban scenarios". In: *Wireless Personal Communications* 83.1, pp. 259–280.

Beggiato, Matthias, Pech, Timo, Leonhardt, Veit, Lindner, Philipp, Wanielik, Gerd, Bullinger-Hoffmann, Angelika, and Krems, Josef (2018). "Lane Change Prediction: From Driver Characteristics, Manoeuvre Types and Glance Behaviour to a Real-Time Prediction Algorithm". In: *UR: BAN Human Factors in Traffic*. Springer, pp. 205–221.

Ben-Hur, Asa and Weston, Jason (2010). "A user's guide to support vector machines". In: *Data mining techniques for the life sciences*. Springer, pp. 223–239.

Bezzina, D and Sayer, J (2014). "Safety pilot model deployment: Test conductor team report". In: *Report No. DOT HS* 812, p. 171.

Bielza, Concha and Larrañaga, Pedro (2014). "Discrete Bayesian network classifiers: a survey". In: *ACM Computing Surveys (CSUR)* 47.1, p. 5.

Bishop, Christopher M (2006). "Pattern recognition". In: *Machine Learning* 128, pp. 1–58.

Boubezoul, Abderrahmane, Koita, Abdourahmane, and Daucher, Dimitri (2009). "Vehicle trajectories classification using support vectors machines for failure trajectory prediction". In: *Advances in Computational Tools for Engineering Applications, 2009. ACTEA'09. International Conference on*. IEEE, pp. 486–491.

Butakov, Vadim A and Ioannou, Petros (2015). "Personalized driver/vehicle lane change models for ADAS". In: *IEEE Transactions on Vehicular Technology* 64.10, pp. 4422–4431.

Camlica, Zehra, Hilal, Allaa, and Kulić, Dana (2016). "Feature abstraction for driver behaviour detection with stacked sparse auto-encoders". In: *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, pp. 003299–003304.

Caner, Mehmet (2009). "Lasso-type GMM estimator". In: *Econometric Theory* 25.1, pp. 270–290.

Chandola, Varun, Banerjee, Arindam, and Kumar, Vipin (2009). "Anomaly detection: A survey". In: *ACM computing surveys (CSUR)* 41.3, p. 15.

Chapelle, Olivier and Zhang, Ya (2009). "A dynamic Bayesian network click model for web search ranking". In: *Proceedings of the 18th international conference on World wide web*. ACM, pp. 1–10.

Chen, Lanlan, Zhao, Yu, Zhang, Jian, and Zou, Jun-zhong (2015). "Automatic detection of alertness/drowsiness from physiological signals using wavelet-based nonlinear features and machine learning". In: *Expert Systems with Applications* 42.21, pp. 7344–7355.

Cohen (1990). "Things I have learned (so far)." In: *American psychologist* 45.12, p. 1304.

Cohen, Jacob (1988). "Statistical power analysis for the behavioral sciences. 1988, Hillsdale, NJ: L". In: *Lawrence Earlbaum Associates* 2.

Dempster, Arthur P, Laird, Nan M, and Rubin, Donald B (1977). "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38.

Denham, Alee (2012). *7 Ways to Predict Weather Without a Forecast*.

Dielmann, Alfred and Renals, Steve (2004). "Dynamic Bayesian networks for meeting structuring". In: *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*. Vol. 5. IEEE, pp. V–629.

Djahel, Soufiene, Jabeur, Nafaa, Barrett, Robert, and Murphy, John (2015). "Toward V2I communication technology-based solution for reducing road traffic congestion in smart cities". In: *2015 International Symposium on Networks, computers and communications (ISNCC)*. IEEE, pp. 1–6.

Djahel, Soufiene, Salehie, Mazeiar, Tal, Irina, and Jamshidi, Pooyan (2013). "Adaptive traffic management for secure and efficient emergency services in smart cities". In: *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. IEEE, pp. 340–343.

Do, Quoc Huy, Tehrani, Hossein, Mita, Seiichi, Egawa, Masumi, Muto, Kenji, and Yoneda, Keisuke (2017). "Human Drivers Based Active-Passive Model for Automated Lane Change". In: *IEEE Intelligent Transportation Systems Magazine* 9.1, pp. 42–56.

Doshi, Anup, Morris, Brendan, and Trivedi, Mohan (2011). "On-road prediction of driver's intent with multimodal sensory cues". In: *IEEE Pervasive Computing* 10.3, pp. 22–34.

Doshi, Anup and Trivedi, Mohan (2008). "A comparative exploration of eye gaze and head motion cues for lane change intent prediction". In: *Intelligent Vehicles Symposium, 2008 IEEE*. IEEE, pp. 49–54.

Doshi, Anup and Trivedi, Mohan M (2010). "Examining the impact of driving style on the predictability and responsiveness of the driver: Real-world and simulator analysis". In: *Intelligent Vehicles Symposium (IV), 2010 IEEE*. IEEE, pp. 232–237.

Doshi, Anup and Trivedi, Mohan Manubhai (2009). "On the roles of eye gaze and head dynamics in predicting driver's intent to change lanes". In: *IEEE Transactions on Intelligent Transportation Systems* 10.3, pp. 453–462.

Elson, Jeremy and Römer, Kay (2003). "Wireless sensor networks: A new regime for time synchronization". In: *ACM SIGCOMM Computer Communication Review* 33.1, pp. 149–154.

Fairclough, Stephen H, May, Andrew J, and Carter, C (1997). "The effect of time headway feedback on following behaviour". In: *Accident Analysis & Prevention* 29.3, pp. 387–397.

Fan, Jerome, Upadhye, Suneel, and Worster, Andrew (2006). "Understanding receiver operating characteristic (ROC) curves". In: *Canadian Journal of Emergency Medicine* 8.1, pp. 19–20.

Fawcett, Tom (2006). "An introduction to ROC analysis". In: *Pattern recognition letters* 27.8, pp. 861–874.

FederalStatisticalOffice (2017). *Traffic accidents*.

Fitch, GM, Lee, SE, Klauer, S, Hankey, J, Sudweeks, J, and Dingus, T (2009). "Analysis of lane-change crashes and near-crashes". In: *US Department of Transportation, National Highway Traffic Safety Administration*.

French, Davina J, West, Robert J, Elander, James, and WILDING, John Martin (1993). "Decision-making style, driving style, and self-reported involvement in road traffic accidents". In: *Ergonomics* 36.6, pp. 627–644.

Friedman, Jerome H (1997). "On bias, variance, 0/1—loss, and the curse-of-dimensionality". In: *Data mining and knowledge discovery* 1.1, pp. 55–77.

Friedman, Nir, Geiger, Dan, and Goldszmidt, Moises (1997). "Bayesian network classifiers". In: *Machine learning* 29.2-3, pp. 131–163.

Ganganwar, Vaishali (2012). "An overview of classification algorithms for imbalanced datasets". In: *International Journal of Emerging Technology and Advanced Engineering* 2.4, pp. 42–47.

Geng, Xiubo, Liu, Tie-Yan, Qin, Tao, and Li, Hang (2007). "Feature selection for ranking". In: *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 407–414.

Ghahramani, Zoubin (1998). "Learning dynamic Bayesian networks". In: *Adaptive processing of sequences and data structures*, pp. 168–197.

Gipps, Peter G (1986). "A model for the structure of lane-changing decisions". In: *Transportation Research Part B: Methodological* 20.5, pp. 403–414.

Glaser, R and Waschulewski, H (2005). "INVENT-Forschungsprojekt Fahrerassistenzsysteme (FAS) Teilprojekt Fahrerverhalten und Mensch-Maschine-Interaktion (FVM) Arbeitspaket 3200 Validierung und Weiterentwicklung des Bewertungsverfahrens I-TSA (Invent–Traffic Safety Assessment). Abschlussbericht". In: *Zugriff am* 23, p. 2015.

Godthelp, Hans, Milgram, Paul, and Blaauw, Gerard J (1984). "The development of a time-related measure to describe driving strategy". In: *Human factors* 26.3, pp. 257–268.

Guyon, Isabelle, Gunn, Steve, Nikravesh, Masoud, and Zadeh, Lofti A (2008). *Feature extraction: foundations and applications*. Vol. 207. Springer.

Ha, Duy-Hung, Aron, Maurice, and Cohen, Simon (2012). "Time headway variable and probabilistic modeling". In: *Transportation Research Part C: Emerging Technologies* 25, pp. 181–201.

Hajinoroozi, Mehdi, Jung, Tzyy-Ping, Lin, Chin-Teng, and Huang, Yufei (2015). "Feature extraction with deep belief networks for driver's cognitive states prediction from EEG data". In: *Signal and Information Processing (ChinaSIP), 2015 IEEE China Summit and International Conference on*. IEEE, pp. 812–815.

Hayward, John C (1972). "NEAR-MISS DETERMINATION THROUGH USE OF A SCALE OF DANGER". In: *Highway Research Record* 384.

Heckbert, Paul (1995). "Fourier Transforms and the Fast Fourier Transform (FFT) Algorithm". In: *Computer Graphics* 2, pp. 15–463.

Henclewood, D, Abramovich, M, and Yelchuru, B (2014). "Safety pilot model deployment-sample data environment data handbook, V. 1.2". In: *USDOT Research and Technology Innovation Administrations*.

Hennessy, Dwight (2011). "Social, personality, and affective constructs in driving". In: *Handbook of traffic psychology*. Elsevier, pp. 149–163.

Higgs, Bryan and Abbas, Montasir (2015). "Segmentation and clustering of car-following behavior: Recognition of driving patterns". In: *IEEE Transactions on Intelligent Transportation Systems* 16.1, pp. 81–90.

Huang, Jin and Ling, Charles X (2005). "Using AUC and accuracy in evaluating learning algorithms". In: *IEEE Transactions on knowledge and Data Engineering* 17.3, pp. 299–310.

J2944, SAE (2013). *Operational definitions of driving performance measures and statistics*. Tech. rep. SAE International.

Jain, Ashesh, Koppula, Hema S, Raghavan, Bharad, Soh, Shane, and Saxena, Ashutosh (2015). "Car that knows before you do: Anticipating maneuvers via learning temporal

driving models". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3182–3190.

Jain, Ashesh, Koppula, Hema S, Soh, Shane, Raghavan, Bharad, Singh, Avi, and Saxena, Ashutosh (2016). "Brain4cars: Car that knows before you do via sensory-fusion deep learning architecture". In: *arXiv preprint arXiv:1601.00740*.

J.Cohen (1992). "A power primer." In: *Psychological bulletin* 112.1, p. 155.

Johansson, Fredrik and Falkman, Goran (2006). "Implementation and integration of a Bayesian Network for prediction of tactical intention into a ground target simulator". In: *Information Fusion, 2006 9th International Conference on*. IEEE, pp. 1–7.

Johnson, Derick A and Trivedi, Mohan M (2011). "Driving style recognition using a smartphone as a sensor platform". In: *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, pp. 1609–1615.

Kaplan, Sinan, Guvensan, Mehmet Amac, Yavuz, Ali Gokhan, and Karalurt, Yasin (2015). "Driver behavior analysis for safe driving: A survey". In: *IEEE Transactions on Intelligent Transportation Systems* 16.6, pp. 3017–3032.

Karim, Mohamed Rehan, SAIFIZUL, Ahmad, YAMANAKA, Hideo, SHARIZLI, Airul, and RAMLI, Rahizar (2013). "Minimum Safe Time Gap (MSTG) as a new Safety Indicator incorporating Vehicle and Driver Factors". In: *Journal of the Eastern Asia Society for Transportation Studies* 10, pp. 2069–2079.

Kasper, Dietmar, Weidl, Galia, Dang, Thao, Breuel, Gabi, Tamke, Andreas, Wedel, Andreas, and Rosenstiel, Wolfgang (2012). "Object-oriented Bayesian networks for detection of lane change maneuvers". In: *IEEE Intelligent Transportation Systems Magazine* 4.3, pp. 19–31.

Kesting, Arne and Treiber, Martin (2013). "Traffic flow dynamics: data, models and simulation". In: *no. Book, Whole)(Springer Berlin Heidelberg, Berlin, Heidelberg, 2013)*.

Kim, Kyung-Nam and Ramakrishna, RS (1999). "Vision-based eye-gaze tracking for human computer interface". In: *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 99CH37028)*. Vol. 2. IEEE, pp. 324–329.

Kira, Kenji and Rendell, Larry A (1992). "A practical approach to feature selection". In: *Machine Learning Proceedings 1992*. Elsevier, pp. 249–256.

Kumar, Puneet, Perrollaz, Mathias, Lefevre, Stéphanie, and Laugier, Christian (2013). "Learning-based approach for online lane change intention prediction". In: *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, pp. 797–802.

Kusano, Kristofer D and Gabler, Hampton (2011). "Method for estimating time to collision at braking in real-world, lead vehicle stopped rear-end crashes for use in pre-crash system design". In: *SAE International Journal of Passenger Cars-Mechanical Systems* 4.2011-01-0576, pp. 435–443.

Lee, SE, Olsen, ECB, and Wierwille, WW (2003). *Naturalistic lanechange field data reduction, analysis, and archiving: A comprehensive examination of naturalistic lane-changes. Final Report*. Tech. rep. DTNH22-00-C-07007, Task Order 4. Submitted to the National Highway Traffic Safety Administration. Blacksburg, VA: Virginia Tech Transportation Institute.

Lee, Suzanne E, Olsen, Erik CB, Wierwille, Walter W, et al. (2004). *A comprehensive examination of naturalistic lane-changes*. Tech. rep. United States. National Highway Traffic Safety Administration.

Leonhardt, Veit, Pech, Timo, and Wanielik, Gerd (2018). "Fusion of driver behaviour analysis and situation assessment for probabilistic driving manoeuvre prediction". In: pp. 223–244.

Leonhardt, Veit and Wanielik, Gerd (2017). "Feature evaluation for lane change prediction based on driving situation and driver behavior". In: *Information Fusion (Fusion), 2017 20th International Conference on*. IEEE, pp. 1–7.

Lethaus, Firas, Baumann, Martin RK, Köster, Frank, and Lemmer, Karsten (2013). "A comparison of selected simple supervised learning algorithms to predict driver intent based on gaze data". In: *Neurocomputing* 121, pp. 108–130.

Lethaus, Firas and Rataj, Jürgen (2007). "Do eye movements reflect driving manoeuvres?" In: *IET Intelligent Transport Systems* 1.3, pp. 199–204.

Li, Guofa, Li, Shengbo Eben, Liao, Yuan, Wang, Wenjun, Cheng, Bo, and Chen, Fang (2015). "Lane change maneuver recognition via vehicle state and driver operation signals—Results from naturalistic driving data". In: *Intelligent Vehicles Symposium (IV), 2015 IEEE*. IEEE, pp. 865–870.

Li, X., Wang, WS., and Rötting, M. (2016). "Bayesian Network-Based Identification of Driver Lane-Changing Intents Using Eye Tracking and Vehicle-Based Data". In: *Advanced Vehicle Control: Proceedings of the 13th International Symposium on Advanced Vehicle Control (AVEC'16), September 13-16, 2016, Munich, Germany*. CRC Press, pp. 229–304.

Liebner, Martin, Ruhhammer, Christian, Klanner, Felix, and Stiller, Christoph (2013). "Generic driver intent inference based on parametric models". In: *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*. IEEE, pp. 268–275.

Liu, Yuqing, Yttri, Eric A, and Snyder, Lawrence H (2010). "Intention and attention: different functional roles for LIPd and LIPv". In: *Nature neuroscience* 13.4, p. 495.

Luo, Yugong, Xiang, Yong, Cao, Kun, and Li, Keqiang (2016). "A dynamic automated lane change maneuver based on vehicle-to-vehicle communication". In: *Transportation Research Part C: Emerging Technologies* 62, pp. 87–102.

Luoma, Juha, Sivak, Michael, and Flannagan, Michael J (1995). "Rapid communication Effects of driver-side mirror type on lane-change accidents". In: *Ergonomics* 38.10, pp. 1973–1978.

Maltz, Masha and Shinar, David (1999). "Eye movements of younger and older drivers". In: *Human factors* 41.1, pp. 15–25.

Mandalia, Hiren M and Salvucci, Mandalia Dario D (2005). "Using support vector machines for lane-change detection". In: *Proceedings of the human factors and ergonomics society annual meeting*. Vol. 49. 22. SAGE Publications Sage CA: Los Angeles, CA, pp. 1965–1969.

Manning, Christopher, Raghavan, Prabhakar, and Schütze, Hinrich (2010). "Introduction to information retrieval". In: *Natural Language Engineering* 16.1, pp. 100–103.

Martin, Sujitha, Tawari, Ashish, Murphy-Chutorian, Erik, Cheng, Shinko Y, and Trivedi, Mohan (2012). "On the design and evaluation of robust head pose for visual user interfaces: Algorithms, databases, and comparisons". In: *Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. ACM, pp. 149–154.

Mathew, Tom V (2014). "Lane Changing Models". In: *Transportation systems engineering anonymous*, pp. 15–1.

Matthaei, Richard, Bagschik, Gerrit, and Maurer, Markus (2014). "Map-relative localization in lane-level maps for ADAS and autonomous driving". In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, pp. 49–55.

McCall, Joel C, Wipf, David P, Trivedi, Mohan M, and Rao, Bhaskar D (2007). "Lane change intent analysis using robust operators and sparse bayesian learning". In: *IEEE Transactions on Intelligent Transportation Systems* 8.3, pp. 431–440.

McGehee, Daniel V, Brown, Timothy L, Lee, John D, and Wilson, Terry B (2002). "Effect of warning timing on collision avoidance behavior in a stationary lead vehicle scenario". In: *Transportation research record* 1803.1, pp. 1–6.

McLachlan, Geoffrey, Do, Kim-Anh, and Ambroise, Christophe (2005). *Analyzing microarray gene expression data*. Vol. 422. John Wiley & Sons.

Menze, Bjoern H, Kelm, B Michael, Masuch, Ralf, Himmelreich, Uwe, Bachert, Peter, Petrich, Wolfgang, and Hamprecht, Fred A (2009). "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data". In: *BMC bioinformatics* 10.1, p. 213.

Mock, Michael, Frings, Reiner, Nett, Edgar, and Trikaliotis, Spiro (2000). "Continuous clock synchronization in wireless real-time applications". In: *Proceedings 19th IEEE symposium on reliable distributed systems SRDS-2000*. IEEE, pp. 125–132.

Mokhiamar, O and Abe, Masato (2002). "Active wheel steering and yaw moment control combination to maximize stability as well as vehicle responsiveness during quick lane change for active vehicle handling safety". In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 216.2, pp. 115–124.

Mörchen, Fabian (2003). *Time series feature extraction for data mining using DWT and DFT*.

Morris, Brendan, Doshi, Anup, and Trivedi, Mohan (2011). "Lane change intent prediction for driver assistance: On-road design and evaluation". In: *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, pp. 895–901.

Murphey, Yi Lu, Milton, Robert, and Kiliaris, Leonidas (2009). "Driver's style classification using jerk analysis". In: *Computational Intelligence in Vehicles and Vehicular Systems, 2009. CIVVS'09. IEEE Workshop on*. IEEE, pp. 23–28.

Murphy (1999). "A variational approximation for Bayesian networks with discrete and continuous latent variables". In: *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., pp. 457–466.

Murphy, K (2012). "Machine learning: a probabilistic approach". In: *Massachusetts Institute of Technology*, pp. 1–21.

Murphy, Kevin (1998a). "A brief introduction to graphical models and Bayesian networks". In: *https://www.cs.ubc.ca/ murphyk/Bayes/bnintro.html*.

Murphy, Kevin et al. (2001). "The Bayes net toolbox for matlab". In: *Computing science and statistics* 33.2, pp. 1024–1034.

Murphy, Kevin Patrick (1998b). *Inference and learning in hybrid Bayesian networks*. University of California, Berkeley, Computer Science Division.

Murphy, Kevin Patrick and Russell, Stuart (2002). "Dynamic bayesian networks: representation, inference and learning". In:

Murphy-Chutorian, Erik, Doshi, Anup, and Trivedi, Mohan Manubhai (2007). "Head pose estimation for driver assistance systems: A robust algorithm and experimental evaluation". In: *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*. IEEE, pp. 709–714.

Narla, Siva RK (2013). "The evolution of connected vehicle technology: From smart drivers to smart cars to... self-driving cars". In: *Ite Journal* 83.7, pp. 22–26.

Neapolitan, Richard E et al. (2004). *Learning bayesian networks*. Vol. 38. Pearson Prentice Hall Upper Saddle River, NJ.

NHTSA (2016). *Traffic accidents 2016*.

— (2017). *TRAFFIC SAFETY FACTS 2015*. Tech. rep. United States. National Highway Traffic Safety Administration.

Nilsson, Julia, Silvlin, Jonatan, Brannstrom, Mattias, Coelingh, Erik, and Fredriksson, Jonas (2016). "If, when, and how to perform lane change maneuvers on highways". In: *IEEE Intelligent Transportation Systems Magazine* 8.4, pp. 68–78.

Nilsson, Julia and Sjöberg, Jonas (2013). "Strategic decision making for automated driving on two-lane, one way roads using model predictive control". In: *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, pp. 1253–1258.

Nyström, Marcus and Holmqvist, Kenneth (2010). "An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data". In: *Behavior research methods* 42.1, pp. 188–204.

Oliver, Nuria and Pentland, Alex P (2000). "Graphical models for driver behavior recognition in a smartcar". In: *Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No. 00TH8511)*. IEEE, pp. 7–12.

Olsen, Erik CB, Lee, Suzanne E, and Wierwille, Walter W (2005). "Eye glance behavior during lane changes and straight-ahead driving". In: *Transportation research record* 1937.1, pp. 44–50.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Pei, Mingtao, Jia, Yunde, and Zhu, Song-Chun (2011). "Parsing video events with goal inference and intent prediction". In: *Computer vision (iccv), 2011 ieee international conference on*. IEEE, pp. 487–494.

Peng, Jinshuan, Guo, Yingshi, Fu, Rui, Yuan, Wei, and Wang, Chang (2015). "Multi-parameter prediction of drivers' lane-changing behaviour with neural network model". In: *Applied ergonomics* 50, pp. 207–217.

Pentland, Alex and Liu, Andrew (1999). "Modeling and prediction of human behavior". In: *Neural computation* 11.1, pp. 229–242.

Plöchl, Manfred and Edelmann, Johannes (2007). "Driver models in automobile dynamics application". In: *Vehicle System Dynamics* 45.7-8, pp. 699–741.

Quimby, A, Maycock, G, Palmer, C, and Buttress, S (1999). *The Factors the Influence a Driver's Choice of Speed: A Questionnaire Study*. Citeseer.

Rehder, T, Louis, L, Muenst, W, and Schramm, D (2016). "Influence of different ground truth hypotheses on the quality of Bayesian networks for maneuver detection and prediction of driving behavior". In: *Advanced Vehicle Control: Proceedings of the 13th International Symposium on Advanced Vehicle Control (AVEC'16), September 13-16, 2016, Munich, Germany*. CRC Press, p. 305.

Reynolds, Douglas (2015). "Gaussian mixture models". In: *Encyclopedia of biometrics*, pp. 827–832.

Rigas, George, Goletsis, Yorgos, and Fotiadis, Dimitrios I (2011). "Real-time driver's stress event detection". In: *IEEE Transactions on intelligent transportation systems* 13.1, pp. 221–234.

Russell, Stuart and Norvig, Peter (1995). "Artificial intelligence: A modern approach prentice-hall". In: *Englewood cliffs, NJ* 26.

Saad, Farida (2004). "Behavioural adaptations to new driver support systems: some critical issues". In: *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*. Vol. 1. IEEE, pp. 288–293.

Safavian, S Rasoul and Landgrebe, David (1991). "A survey of decision tree classifier methodology". In: *IEEE transactions on systems, man, and cybernetics* 21.3, pp. 660–674.

Sagberg, Fridulv, Selpi, Bianchi Piccinini, Giulio Francesco, and Engström, Johan (2015). "A review of research on driving styles and road safety". In: *Human factors* 57.7, pp. 1248–1275.

Salfner, Felix and Malek, Miroslaw (2007). "Using hidden semi-Markov models for effective online failure prediction". In: *Reliable Distributed Systems, 2007. SRDS 2007. 26th IEEE International Symposium on*. IEEE, pp. 161–174.

Salvucci, Dario D (2004). "Inferring driver intent: A case study in lane-change detection". In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 48. 19. SAGE Publications Sage CA: Los Angeles, CA, pp. 2228–2231.

Salvucci, Dario D and Goldberg, Joseph H (2000). "Identifying fixations and saccades in eye-tracking protocols". In: *Proceedings of the 2000 symposium on Eye tracking research & applications*. ACM, pp. 71–78.

Salvucci, Dario D and Liu, Andrew (2002). "The time course of a lane change: Driver control and eye-movement behavior". In: *Transportation research part F: traffic psychology and behaviour* 5.2, pp. 123–132.

Salvucci, Dario D, Mandalia, Hiren M, Kuge, Nobuyuki, and Yamamura, Tomohiro (2007). "Lane-change detection using a computational driver model". In: *Human factors* 49.3, pp. 532–542.

Schmitt, Felix, Korthauer, Andreas, Manstetten, Dietrich, and Bieg, Hans-Joachim (2018). "Predicting strategies of driving in presence of additional visually demanding tasks: inverse optimal control estimation of steering and glance behaviour models". In: *UR: BAN Human Factors in Traffic*. Springer, pp. 183–204.

Scholkopf, Bernhard and Smola, Alexander J (2001). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press.

Schreier, Matthias, Willert, Volker, and Adamy, Jürgen (2013). "From grid maps to parametric free space maps—A highly compact, generic environment representation for ADAS". In: *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 938–944.

Schubert, Robin, Schulze, Karsten, and Wanielik, Gerd (2010). "Situation assessment for automatic lane-change maneuvers". In: *IEEE Transactions on Intelligent Transportation Systems* 11.3, pp. 607–616.

Schum, David A (2001). *The evidential foundations of probabilistic reasoning*. Northwestern University Press.

Seif, Heiko G and Hu, Xiaolong (2016). "Autonomous driving in the iCity—HD maps as a key challenge of the automotive industry". In: *Engineering* 2.2, pp. 159–162.

Senaratne, Rajinda, Hardy, David, Vanderaa, Bill, and Halgamuge, Saman (2007). "Driver fatigue detection by fusing multiple cues". In: *International Symposium on Neural Networks*. Springer, pp. 801–809.

Singh, Santokh (2015). *Critical reasons for crashes investigated in the national motor vehicle crash causation survey*. Tech. rep.

Sivaraman, Sayanan and Trivedi, Mohan Manubhai (2014). "Dynamic probabilistic drivability maps for lane change and merge driver assistance". In: *IEEE Transactions on Intelligent Transportation Systems* 15.5, pp. 2063–2073.

SMI (2015). "SMI Eye Tracking Glasses 2 Wireless". In: *Mobile eye tracking made easy, robust, efficient and versatile. URL: http://www. eyetracking-glasses. com/fileadmin/user_upload/documents/smi _ etg2w _ flyer _ naturalgaze. pdf (visited on 12/22/2015).*

Smola, Alex J and Schölkopf, Bernhard (2004). "A tutorial on support vector regression". In: *Statistics and computing* 14.3, pp. 199–222.

Steele, Russell J and Raftery, Adrian E (2010). "Performance of Bayesian model selection criteria for Gaussian mixture models". In: *Frontiers of Statistical Decision Making and Bayesian Analysis* 2, pp. 113–130.

Sullivan, Gail M and Feinn, Richard (2012). "Using effect size—or why the P value is not enough". In: *Journal of graduate medical education* 4.3, pp. 279–282.

Sun, Shiliang, Zhang, Changshui, and Yu, Guoqiang (2006). "A Bayesian network approach to traffic flow forecasting". In: *IEEE Transactions on intelligent transportation systems* 7.1, pp. 124–132.

Thissen, U, Van Brakel, R, De Weijer, AP, Melssen, WJ, and Buydens, LMC (2003). "Using support vector machines for time series prediction". In: *Chemometrics and intelligent laboratory systems* 69.1-2, pp. 35–49.

Thomas, Samuel, Ganapathy, Sriram, and Hermansky, Hynek (2008). "Recognition of reverberant speech using frequency domain linear prediction". In: *IEEE Signal Processing Letters* 15, pp. 681–684.

Tijerina, Louis (1999). "Operational and behavioral issues in the comprehensive evaluation of lane change crash avoidance systems". In: *Transportation Human Factors* 1.2, pp. 159–175.

Tijerina, Louis, Garrott, W, Stoltzfus, Duane, and Parmer, Edwin (2005). "Eye glance behavior of van and passenger car drivers during lane change decision phase". In: *Transportation Research Record: Journal of the Transportation Research Board* 1937, pp. 37–43.

Tordeux, Antoine, Lassarre, Sylvain, and Roussignol, Michel (2010). "An adaptive time gap car-following model". In: *Transportation research part B: methodological* 44.8, pp. 1115–1131.

Vafaie, Haleh and Imam, Ibrahim F (1994). "Feature selection methods: genetic algorithms vs. greedy-like search". In: *Proceedings of the International Conference on Fuzzy and Intelligent Control Systems*. Vol. 51, p. 28.

Vickers, Andrew J and Elkin, Elena B (2006). "Decision curve analysis: a novel method for evaluating prediction models". In: *Medical Decision Making* 26.6, pp. 565–574.

Vöhringer-Kuhnt, Thomas and Trexler-Walde, Lisa (2005). "Evaluation einer Kurzversion des Driver Behaviour Questionnaire". In: p. 345.

Wahle, Joachim, Bazzan, Ana Lúcia C, Klügl, Franziska, and Schreckenberg, Michael (2000). "Decision dynamics in a traffic scenario". In: *Physica A: Statistical Mechanics and its Applications* 287.3-4, pp. 669–681.

— (2002). "The impact of real-time information in a two-route scenario using agent-based simulation". In: *Transportation Research Part C: Emerging Technologies* 10.5-6, pp. 399–417.

Wakasugi, Takashi (2005). "A study on warning timing for lane change decision aid systems based on driver's lane change maneuver". In: *Proc. 19th International Technical Conference on the Enhanced Safety of Vehicles, Paper*. 05-0290.

Wang, Lipo (2005). *Support vector machines: theory and applications*. Vol. 177. Springer Science Business Media.

Wang, Wenshuo, Xi, Junqiang, Chong, Alexandre, and Li, Lin (2017). "Driving style classification using a semisupervised support vector machine". In: *IEEE Transactions on Human-Machine Systems* 47.5, pp. 650–660.

Weidl, Galia, Madsen, Anders L, Wang, Stevens, Kasper, Dietmar, and Karlsen, Martin (2018). "Early and accurate recognition of highway traffic maneuvers considering real world application: a novel framework using Bayesian networks". In: *IEEE Intelligent Transportation Systems Magazine* 10.3, pp. 146–158.

Windridge, David, Shaukat, Affan, and Hollnagel, Erik (2013). "Characterizing driver intention via hierarchical perception–action modeling". In: *IEEE Transactions on Human-Machine Systems* 43.1, pp. 17–31.

Winner, Hermann and Lueder, Jens (2005). *Method for recognizing a change in lane of a vehicle*. US Patent 6,889,161.

Woo, Hanwool, Ji, Yonghoon, Kono, Hitoshi, Tamura, Yusuke, Kuroda, Yasuhide, Sugano, Takashi, Yamamoto, Yasunori, Yamashita, Atsushi, and Asama, Hajime (2016). "Dynamic potential-model-based feature for lane change prediction". In: *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*. IEEE, pp. 000838–000843.

Xing, Zhou and Xiao, Fei (2018). "Predictions of short-term driving intention using recurrent neural network on sequential data". In: *arXiv preprint arXiv:1804.00532*.

Xu, Guoqing, Liu, Li, Ou, Yongsheng, and Song, Zhangjun (2012). "Dynamic modeling of driver control strategy of lane-change behavior and trajectory planning for collision prediction". In: *IEEE Transactions on Intelligent Transportation Systems* 13.3, pp. 1138–1155.

Zhang, Bin, Georgoulas, Georgios, Orchard, Marcos, Saxena, Abhinav, Brown, Douglas, Vachtsevanos, George, and Liang, Steven (2008). "Rolling element bearing feature extraction and anomaly detection based on vibration monitoring". In: *Control and Automation, 2008 16th Mediterranean Conference on*. IEEE, pp. 1792–1797.

Zhang, Harry (2004). "The optimality of naive Bayes". In: *AA* 1.2, p. 3.

Zhao, Ding, Guo, Yaohui, and Jia, Yunhan Jack (2017). "Trafficnet: An open naturalistic driving scenario library". In: *Intelligent Transportation Systems (ITSC), 2017 IEEE 20th International Conference on*. IEEE, pp. 1–8.

Zhao, Ding, Lam, Henry, Peng, Huei, Bao, Shan, LeBlanc, David J, Nobukawa, Kazutoshi, and Pan, Christopher S (2017). "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques". In: *IEEE transactions on intelligent transportation systems* 18.3, pp. 595–607.

Zhu, Youding and Fujimura, Kikuo (2004). "Head pose estimation for driver monitoring". In: *Intelligent Vehicles Symposium, 2004 IEEE*. IEEE, pp. 501–506.

Zhu, Zhiwei, Fujimura, Kikuo, and Ji, Qiang (2002). "Real-time eye detection and tracking under various light conditions". In: *Proceedings of the 2002 symposium on Eye tracking research & applications*. ACM, pp. 139–144.

Zühlsdorff, Anne-Marie (2018). *Vorhersage von Fahrmanövern mithilfe künstlicher neuronaler Netze*. Abschlussarbeiten - Universitätsbibliothek TU Berlin.