



Lukas Wörle, M.Sc.

# Objective Criteria for the Driving Style of Race Car Drivers

## DOCTORAL THESIS

to achieve the university degree of

Doctor technicae

submitted to

**Graz University of Technology**

Assessors:

Assoc.Prof. Dipl.-Ing. Dr.techn. Arno Eichberger

Institute of Automotive Engineering  
Graz University of Technology

Prof. Dr.-Ing. Günther Prokop

Institute of Automobile Engineering  
Technische Universität Dresden

Graz, April 2020

# Acknowledgement

The present thesis was authored during my activity as doctoral student in the motorsport development department at BMW AG, in cooperation with the Institute of Automotive Engineering at Graz University of Technology. Here I want to thank the persons who have supported me on this endeavour.

My most sincere thanks go to Assoc.Prof. Dr.techn. Arno Eichberger, head of the research area Vehicle Dynamics at TU Graz. Not only for his excellent scientific guidance and technical advice, but also for the open discussions and his suggestions to the writing of this thesis. His continuous support and encouraging words helped me to finally finish this work.

I would also like to thank Prof. Dr.-Ing. Günther Prokop, Dean of the Faculty of Transport and Traffic Sciences and Professor for Automobile Engineering at TU Dresden, for his kind acceptance of being the second assessor of this dissertation.

The presented research could only be realised because BMW AG provided the necessary infrastructure, data, and budget. Special thanks go to Dr.-Ing. Michael Graf in the roles of scientific supervision from the company and head of the vehicle performance and simulation department. He supported the project right from the beginning and provided assistance the whole time. Likewise, I wish to express my thanks to my mentor Dr.-Ing. Florian Preuss, who always had an open ear for any concerns despite his various commitments at BMW AG.

Furthermore, I would like to acknowledge the great relationship with my colleagues and fellow doctoral students in Munich, as well as the welcoming atmosphere at the institute in Graz. I enjoyed the on- and off-topic discussions and activities in both places very much.

I want to thank my girlfriend Lisa for her great help with proof-reading and motivating me to finish this thesis. Finally, special thanks go to my family for their enduring support and encouragement, which enabled me to choose this path in the first place.

Last but not least, my gratitude goes to all of my friends, who each contributed in their own way to help me keep an eye on the goal and finish this work.

Thank you for your support!

Lukas Wörle  
Munich, April 2020

# AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present doctoral thesis.

---

Date

---

Signature

# Abstract

The aim of this dissertation is to identify suitable objective metrics to describe the *driving style* of racing drivers. The motivation for this research results from the problem of the individual adaptation of the dynamic driving behaviour of the race car to the *driving style* of the respective driver. Since only the combination of vehicle and driver is able to win races, this is an important aspect in racing. In turn, this means that focusing on an optimal combination of both is more effective than concentrating exclusively on each component individually. Currently, the individual tuning of race cars for the driver is achieved primarily through empirical knowledge about the respective driver. Since it is time-consuming to gather the required experience on each combination of driver and vehicle, an objective approach can accelerate this process.

The presented methodology uses pattern recognition to find characteristic points during cornering in motorsports. These are used to calculate the defined metrics from recorded time series data. The selection of the metrics is based on the *Relief* algorithm as well as *Recursive Feature Elimination* in combination with a *Random Forest* classifier, which is also used to prove the significance of the developed method.

The result of the present work is a set of objective metrics calculated from data typically available from race cars with data acquisition systems or driving simulators. As proof of the concept, 23 different metrics, selected from the 104 evaluated metrics, are used as features to identify the driver of a particular dataset by machine learning. The average accuracy is 77%.

# Kurzfassung

Das Ziel der Dissertation ist die Identifikation geeigneter objektiver Metriken zur Beschreibung des *Fahrstils* von Rennfahrern. Die Motivation für diese Untersuchung ergibt sich aus der Problematik der individuellen Anpassung des dynamischen Fahrverhaltens des Rennfahrzeugs auf den *Fahrstil* des jeweiligen Fahrers. Da nur die Kombination aus Fahrzeug und Fahrer in der Lage ist Rennen zu gewinnen, ist dies ein wichtiger Aspekt im Rennsport. Das bedeutet wiederum, dass der Fokus auf eine optimale Kombination von beiden zielführender ist, als sich ausschließlich auf jede Komponente einzeln zu konzentrieren. Derzeit wird die individuelle Abstimmung der Rennfahrzeuge auf den Fahrer vor allem durch empirisches Wissen über den jeweiligen Fahrer erreicht. Da es zeitaufwendig ist die erforderliche Erfahrung über jede Kombination aus Fahrer und Fahrzeug zu sammeln kann ein objektiver Ansatz den Prozess beschleunigen.

Die vorgestellte Methodik nutzt Mustererkennung, um charakteristische Punkte bei Kurvenfahrten im Motorsport zu finden. Diese werden für die Berechnung der definierten Metriken aus den aufgezeichneten Zeitreihendaten verwendet. Die Auswahl der Merkmale beruht auf dem *Relief* Algorithmus sowie der *Recursive Feature Elimination* in Verbindung mit einem *Random Forest* Klassifizierer, welcher auch für einen Nachweis der Signifikanz der Methode genutzt wird.

Das Ergebnis der vorliegenden Arbeit ist ein Satz objektiver Metriken, die aus Daten berechnet werden, welche üblicherweise von Rennfahrzeugen mit Datenerfassungssystem oder Fahrsimulatoren zur Verfügung stehen. Als Nachweis der Funktionalität des Konzeptes werden 23 verschiedene Metriken, ausgewählt aus den insgesamt 104 evaluierten Metriken, als Merkmale verwendet, um den Fahrer eines bestimmten Datensatzes mittels maschinellem Lernen zu erkennen. Die durchschnittliche Genauigkeit beträgt dabei 77%.

# Contents

|   |             |
|---|-------------|
| <b>Acknowledgement</b>                                  | <b>ii</b>   |
| <b>AFFIDAVIT</b>  | <b>iii</b>  |
| <b>Abstract</b>   | <b>iv</b>   |
| <b>Kurzfassung</b>                                      | <b>v</b>    |
| <b>Table of Contents</b>                                | <b>vii</b>  |
| <b>Abbreviations</b>                                    | <b>viii</b> |
| <b>Symbols</b>  | <b>x</b>    |
| <b>1 Introduction</b>                                   | <b>1</b>    |
| 1.1 Motivation . . . . .                                | 1           |
| 1.2 Limitations . . . . .                               | 4           |
| 1.3 Hypotheses and Outline . . . . .                    | 5           |
| <b>2 State of the Art</b>                               | <b>6</b>    |
| 2.1 The Driving Task . . . . .                          | 6           |
| 2.2 Driver Modelling . . . . .                          | 8           |
| 2.3 Driving Style . . . . .                             | 10          |
| <b>3 Methodology</b>                                    | <b>14</b>   |
| 3.1 Stabilisation Level . . . . .                       | 15          |
| 3.1.1 Detection of Characteristic Points . . . . .      | 15          |
| 3.1.2 Objective Criteria . . . . .                      | 23          |
| 3.2 Guidance Level . . . . .                            | 42          |
| 3.2.1 Curvature of the Vehicle Trajectory . . . . .     | 42          |
| 3.2.2 Reference Trajectory . . . . .                    | 44          |
| 3.2.3 Objective Criteria . . . . .                      | 47          |
| 3.3 Metric Evaluation . . . . .                         | 54          |
| <b>4 Evaluation</b>                                     | <b>61</b>   |
| 4.1 Simulation Environment and Available Data . . . . . | 61          |
| 4.1.1 Simulation Environment . . . . .                  | 62          |
| 4.1.2 Database . . . . .                                | 63          |

|          |   |            |
|----------|---|------------|
| 4.1.3    | Used Software . . . . .                   | 69         |
| 4.2      | Selection of Objective Criteria . . . . . | 69         |
| 4.2.1    | Stabilisation Level . . . . .             | 69         |
| 4.2.2    | Guidance Level . . . . .                  | 87         |
| 4.3      | Results . . . . .                         | 93         |
| 4.4      | Discussion . . . . .                      | 104        |
| <b>5</b> | <b>Summary and Outlook</b>                | <b>106</b> |
|          | <b>List of Figures</b>                    | <b>109</b> |
|          | <b>List of Tables</b>                     | <b>111</b> |
|          | <b>Bibliography</b>                       | <b>112</b> |

# Abbreviations

|      |                                    |
|------|------------------------------------|
| ADAS | Advanced Driver Assistance Systems |
| API  | Application Programming Interface  |
| CAD  | Computer Aided Design              |
| CFD  | Computational Fluid Dynamics       |
| CoG  | Center of Gravity                  |
| DAQ  | Data Acquisition                   |
| DiLS | Driver-in-the-Loop Simulator       |
| DoF  | Degree of Freedom                  |
| DP   | deliberate practice                |
| DRS  | Drag Reduction System              |
| ECU  | Electronic Control Unit            |
| FEM  | Finite Element Method              |
| FFT  | Fast Fourier Transform             |
| FIR  | finite impulse response            |
| FoV  | Field-of-View                      |
| GPS  | Global Positioning System          |
| IDE  | Integrated Development Environment |
| KnC  | Kinematics and Compliance          |
| KPI  | Key Performance Indicator          |
| NaN  | Not-a-Number                       |
| OEM  | Original Equipment Manufacturer    |
| OOB  | Out-of-Bag                         |



|      |                               |
|------|-------------------------------|
| RFE  | Recursive Feature Elimination |
| RMS  | Root Mean Square              |
| RMSE | Root Mean Square Error        |
| SVM  | Support Vector Machines       |

# Symbols

|                      |   |
|----------------------|---|
| $NU_i$               | Normalisation unit for relief weight of feature $i$           |
| $N_G$                | Gear number   |
| $N_{\text{oft}}$     | Sample of off-throttle event during cornering                 |
| $N_{\text{onb}}$     | Sample of on-brake event during cornering                     |
| $W_i$                | Relief weight of feature $i$                                  |
| <b>D</b>             | Matrix containing distance between detected cornering events  |
| <b>R</b>             | Correlation matrix  |
| <b>T</b>             | Rotation matrix for conversion between coordinate systems     |
| <b>o</b>             | Positional offset for conversion between coordinate systems   |
| $\mathbf{x}_k$       | Observation $k$   |
| $\kappa$             | Curvature of a circle, or instantaneous curvature for a curve |
| $\kappa_P$           | Path curvature  |
| $\rho$               | Pearson correlation coefficient                               |
| $a_S$                | Steering wheel angle  |
| $a'_S$               | Gradient of steering wheel angle                              |
| $a''_S$              | Curvature of steering wheel angle                             |
| $d_C$                | Distance of driving line to track centre line                 |
| $d_E$                | Distance of driving line to inner track edge                  |
| $f_T$                | Frequency of throttle signal                                  |
| $p_{B,\text{min},s}$ | Minimum brake pressure in static conditions                   |
| $p_{B,p99}$          | 99th quantile of brake pressure signal                        |
| $p_B$                | Brake pressure  |
| $p'_B$               | Gradient of brake pressure                                    |
| $p''_B$              | Curvature of brake pressure                                   |
| $r$                  | Radius of a circle, or instantaneous radius for a curve       |
| $r_{T,\text{max},s}$ | Maximum throttle ratio in static conditions                   |
| $r_{T,\text{min},s}$ | Minimum throttle ratio in static conditions                   |
| $r_{T,\text{not}}$   | Throttle ratio at no-throttle event                           |

---

|                    |   |
|--------------------|---|
| $r_{T,\text{oft}}$ | Throttle ratio at off-throttle event                    |
| $r_{T,p5}$         | Fifth quantile of throttle ratio signal                 |
| $r_{T,p95}$        | 95th quantile of throttle ratio signal                  |
| $r_T$              | Throttle ratio  |
| $r'_T$             | Gradient of throttle ratio                              |
| $r''_T$            | Curvature of throttle ratio                             |
| $s$                | Distance  |
| $s_L$              | Lap distance  |
| $s_{\text{fut}}$   | Distance of full-throttle event during cornering        |
| $s_{\text{ofb}}$   | Distance of off-brake event during cornering            |
| $s_{\text{oft}}$   | Distance of off-throttle event during cornering         |
| $s_{\text{ont}}$   | Distance of on-throttle event during cornering          |
| $t$                | Time  |
| $t_{L,\text{ref}}$ | Reference lap time                                      |
| $t_L$              | Lap time  |
| $t_{S,\text{max}}$ | Time with maximum steering wheel angle during cornering |
| $t_{\text{not}}$   | Time of no-throttle event during cornering              |
| $t_{\text{oft}}$   | Time of off-throttle event during cornering             |
| $t_{\text{tui}}$   | Time of turn-in event during cornering                  |
| $v_C$              | Velocity  |
| $x_c$              | X-coordinate of the vehicle                             |
| $y_c$              | Y-coordinate of the vehicle                             |
| $y_k$              | Class for observation k                                 |
| $z_c$              | Z-coordinate of the vehicle                             |
| $x_i$              | Feature i   |
| $x_j$              | Feature j   |
| $x_{k,i}$          | Feature i of observation k                              |
| $x_{k,j}$          | Feature j of observation k                              |

# 1 Introduction

The driver is a vital part of any motor racing activity, since he/she is a key component contributing to success when the car is out on track, together with the pit crew and the engineering team which have important roles nevertheless. Individual drivers show different approaches to achieve the ultimate goal of winning races while they all depend on the machinery at their disposal. This implies a close relationship of *driving style* and *driveability* in motorsport. Race cars can be tuned by many different adjustments, which add up to a complex system with a non-trivial amount of possible setups. The imposed engineering problem is finding the optimum setup for the car to achieve maximal performance. However, the car must remain controllable by the driver. As drivers differ in the way they drive a car they also prefer different setups for their cars. Subsequently, a specific vehicle behaviour may seem driveable to one driver but not necessarily to the other.

This section introduces the topic of subjective *driving style* in more detail, outlines the motivation as well as conditions that apply, and presents the hypotheses that will be investigated by this work as well as the structure of the document.

## 1.1 Motivation

The main goal in professional motorsport is to drive around a given race track within minimal time per lap, concluding in the desire for minimal race time or maximum race distance within given time. The motivation for this is similar to other sports, where athletes compete against each other. However, since motor racing is largely dependent on the vehicles, there is also a substantial interest of the Original Equipment Manufacturers (OEMs), not only for marketing purposes, but also as a proving ground for high performance technology. With the used vehicles becoming more and more complex, the possible changes in mechanical and software setup of the car increase as well and create systems with magnitudes of variables that are not trivial to understand. This involves three major components, the driver, the vehicle and the race track, as illustrated in Fig. 1.1. In this system the track includes all environmental conditions such as track grip, wind speed and direction, temperature and more. The vehicle is directly interacting with the track via the tyres as it supports all forces that occur while the car moves around the course. This also includes aerodynamic forces because air is part of this system. Furthermore, the driver uses the car's control inputs, thus is interacting with the vehicle as he/she guides it around the racetrack. The driver also interacts with the track, for

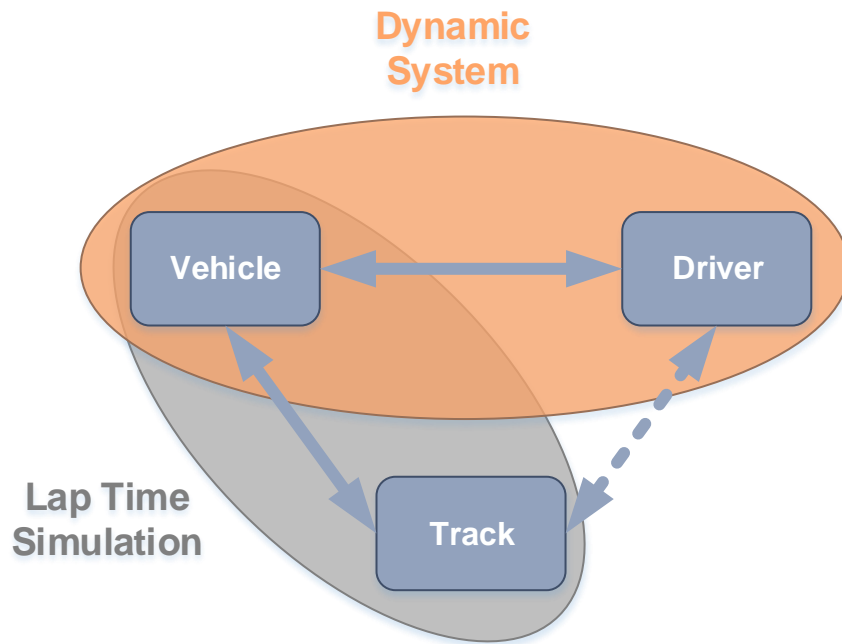


Figure 1.1: Dynamic system overview

example by using visual reference points on the lap, however this factor is not within the scope of this work, as illustrated by the dashed arrow in Fig. 1.1.

The three dynamic components of the systems, as just introduced, can be modified to affect the achieved lap time, which is to be minimised. While the race track can be influenced, for example, by changing its surface, it is not a parameter manipulated by competitors, which allows the assumption that it is equal for all participants. This contains a simplification as it is possible to gain an advantage by making use of changing track conditions, exemplary through careful timing of a qualifying lap within the permitted time slot. The vehicle generally offers various options to affect lap time performance by changing or adjusting mechanical parts and software on Electronic Control Units (ECUs). A good overview of the components that make up a race car and their adjustment possibilities is provided by Trzesniowski [56][57][58]. These influence the behaviour of the car directly. Additionally, the driver is a dynamic system itself and can change the way he/she drives the race car. In this model of the overall dynamic system the drivers dynamic behaviour can be interpreted as his/her *driving style*, which is an individual attribute of each driver. This leads to different *driving styles*, or in other words, ways of how to interact with a vehicle, and therefore demands considering this influence when

setting up a race car. With the overall goal of minimal lap time in mind, the setup process's main goal is finding the most performing race car for a given combination of track and driver. This leads to sometimes very different results depending on the individual driver. The dynamic behaviour of the car can also be understood as its *driveability*, meaning the vehicle is controllable and can be kept on the desired trajectory by the driver. Due to individual *driving styles* and also varying abilities to control the car, a measure for *driveability* is subjective to the respective driver doing the assessment.

The overall trend to use computer simulations in the automotive industry, which is mainly fuelled by cost saving, is also present in motorsport. It is a consequence of restrictions for physical component and vehicle tests as well as on data acquisition systems. A second factor is the number of possible setups, which can not be assessed efficiently without virtual development. Lap time simulation is an established way to objectively judge the combination of track and vehicle and allows finding a theoretical optimum. Völkl describes an optimal control based approach in his work on lap time simulation [61]. The simulated optimum is theoretical, because it does not incorporate the driver in a way that is sufficient to replicate the complete system of track, vehicle and driver. The challenge to find the right dynamic behaviour of the car for an individual driver is currently solved by an empiric approach that incorporates test driving, where the driver and associated engineers can assess different setups. The engineers must understand each driver and his/her needs to allow finding a good solution, with a key factor being the ability to adjust in response to changes of conditions while keeping the specific characteristics needed by the driver. This requires experience, which in turn demands time and is contrary to the aforementioned target of cost saving. The latter is usually employed through limitation of test days, leading to the need for an objective approach that allows making use of virtual development to adjust a race car for individual drivers. The work of Goy aims to provide a measure for the amount of vehicle performance that the driver is actually able to use, which targets to help the explained process of finding a balance between *driveability* and *performance* [18].

A methodical and objective approach to set up a race car for an individual driver requires knowledge about all involved systems' dynamic behaviours in order to assess their combination. An important step is finding objective criteria that describe each system. Consequently, objective criteria are needed to describe the driver, or *driving style*, and the vehicle, or *driveability*, to study their interaction in detail. In addition to the purpose of setting up the car for race weekends, this approach can also be used in initial car design. Through simulations, the car behaviour can already be evaluated in early design stages, enabling to consider the drivers' needs. Accordingly, a target corridor for dynamic properties can be set to positively influence car development. The scope of this work is to provide a set of objective metrics that allow differentiation of individual *driving styles*, that is to distinguish differences between drivers as well as show similarities.

## 1.2 Limitations

In this study the track, or environment respectively, is treated as a static component that does not affect the overall system of vehicle and driver. This assumption is made because the track is not meant to be changed by competitors in motorsport. Although conditions can change, which will affect car performance and setup choices, this simplification is chosen to be adequate as environmental changes within short time ranges are negligible in most cases. This shortcoming can be compensated by running multiple simulations with different track conditions to account for changes on a larger time scale.

The presented method requires a vehicle driven to its limit in order to obtain the desired results. The method is making use of the specific sequence of control inputs to the car that is shown in racing by professional drivers. This pattern as well as deviations from it are discussed in detail in chapter 3. The analysis relies solely on the main pattern to reduce the complexity of the problem. Some alternative patterns are detected in the process, but it must be stated that the approach is not suited to classify road car users. This topic is subject of current and past research and will be outlined in chapter 2.

Since motorsport involves many different categories and usable vehicles, it needs to be pointed out that this study is based entirely on car racing. More specifically only on-road racing activities are analysed. This limitation does not necessarily prohibit any use of the proposed method for other categories. It could be well suited for application with on-road motorcycle racing for example, however this would require a separate evaluation of the method for a specific area.

As previously discussed, the main goal in motorsport is the reduction of race time. As this is represented by the addition of all single lap times, lap time is used synonymously throughout this thesis. While the presented approach ultimately serves this purpose, it is not intended to assess driver or vehicle performance. The main idea is to accumulate knowledge about the interaction of driver and vehicle while focusing purely on the objective description. Subsequently, the derived metrics are also suitable for evaluation purposes that can be based on the presented approach. This is however outside the scope of this work.

## 1.3 Hypotheses and Outline

The discussed goal of describing the *driving style* for individual race car drivers relies on hypotheses that are presented in this section. Furthermore an outline of this thesis is provided.

**Hypothesis 1:** Race car drivers show individual differences in *driving style*.

This hypothesis is based on empirical observations by many experts working in the field of motor racing. However, it remains to be proven that drivers show characteristic properties in a consistent way.

**Hypothesis 2:** The *driving style* of race car drivers can be described by objective criteria.

Building on the first hypothesis it is further assumed that it is possible to describe a drivers dynamic behaviour by objective metrics. This approach should ultimately lead to significant differentiation among sampled drivers through their respective set of metrics.

The presented research is organised into five chapters, following a typical approach to scientific reporting. The current chapter summarises the motivation of the study, its limitations, and the proposed hypotheses. Subsequently, chapter two presents a commonly used model of the driving task and aggregates approaches used for similar problems, although mostly related to road car users. Following the definition of the problem and available strategies, chapter three elaborates the new method tailored to the specific purpose of describing the *driving style* of race car drivers. This includes an initial pre-processing step of available data, the definition and calculation of various metrics, and a presentation of methods to reduce the derived set of measures to contain only significant metrics. The process is then outlined in chapter four in the form of presentation and evaluation of the results in three steps. The final results after the selection process are illustrated in detail for selected examples. Finally, the last chapter provides a summary of this work as well as an outlook to further research possibilities and applications of the method.



## 2 State of the Art

The following chapter summarises existing work on *driving style*, outlines the used methods and explains the respective focus of application, which essentially implies limits of the solution as well. A commonly used model of the driving task is presented together with selected current uses. Aspects of driver modelling are elaborated building on the model of the driving task. These incorporate also driver behaviour imitation. Furthermore, the field of *driving style* and driver skill recognition and classification is discussed with the main outcome that a new approach is needed for the specific problem addressed by this work, which is the objective description of the *driving style* of race car drivers.

### 2.1 The Driving Task

An often cited model for the driving task is mentioned by Donges and McRuer et al. Early works of both authors focus on modelling driver steering behaviour on two levels by a combination of open-loop and closed-loop control. This approach is due to the duality of the human steering behaviour and results in the definition of two levels. First the guidance level, where the driver perceives the current and estimated future driving path. This results in anticipatory, open-loop reactions. The second level is the stabilisation level, where deviations from the intended course are compensated. [9][11] Additionally, three control modes are proposed, which are compensatory control, pursuit control and precognitive control. While compensatory control relies purely on closed-loop reactions to lateral and heading errors, pursuit control permits the driver to anticipate the desired path. Precognitive control conversely relies on patterns often executed and leads to a combination of open-loop execution of a well known driving manoeuvre with closed-loop correction of any deviations to the effective driving situation. [36]

Following this early model, Donges enhanced the approach with a third level, resulting in the navigation-, guidance- and stabilisation layers. The latter two levels are actively conscious while driving. His work identifies potential to enhance road safety regarding communication with other road users and their visibility on the guidance level, as well as allowing the driver to use more of the available vehicle potential in critical situations on the stabilisation level. [10] Additionally, Michon links facets of risk to the three levels of the driving task which are referred to as the strategic, tactical and operational level in his work. The respective facets of risk are risk acceptance, risk taking and coping with threat. Michon states that each kind of dealing with a risk can affect each level of the driving task, however main interactions exist between risk acceptance and the strategic level, risk taking and the tactical level as well as coping with risk and the operational level. [38]

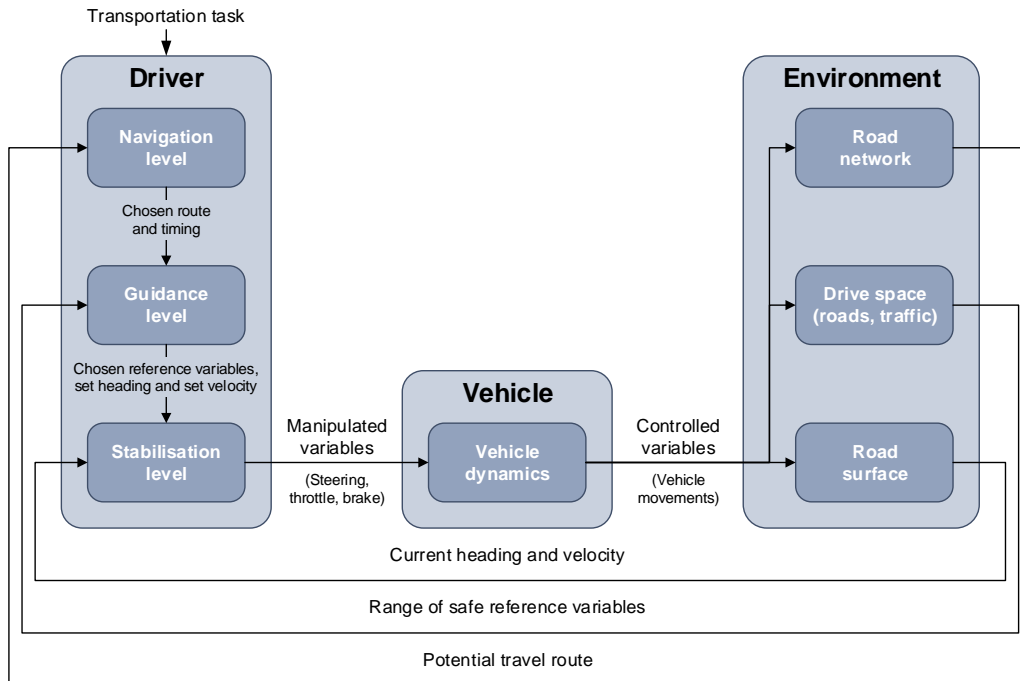


Figure 2.1: Three-level model of the driving task [10]

Figure 2.1 illustrates the model as proposed by Donges. The driving task is split in the aforementioned three levels. The navigation task, which is on the highest level, consists of selecting an appropriate route based on the available roads and estimating the temporal sequence of the trip. On the guidance level, the driver is using knowledge of the intended route and information about the environmental conditions around the vehicle to derive a nominal track as well as a nominal velocity. Literature uses the terms trajectory-, motion- and path-planning as equivalent synonyms. Usually objective conditions such as lane markings, signage and other road users do not explicitly define nominal track and velocity, but specify a range of vehicle movement from which the driver subjectively chooses the control targets. Finally, the actual lateral and longitudinal movement of the vehicle is controlled to follow the derived nominal values on the stabilisation level. [10]

Apart from the mentioned publications that are closely related to the automotive sector, Rasmussen proposed that human actions can be divided into categories, which show strong ties to the discussed model of the driving task. While skill-based behaviour, as the lowest level of control, yields actions directly from the sensory system such as fast limb movements, rule-based behaviour relies on empirically learned rules and leads to conscious actions that can be explained. Skill-based behaviour in turn happens unconsciously. Particularly in unknown situations where no rules have been learned, knowledge-based behaviour is used. On this level, a plan to achieve a certain goal is derived and

weighed against other options, based on individual knowledge. [48] This description matches well with the definition of open-loop and closed-loop control as seen for the different levels of the driving task.

Although the original publications on this model are from the late 1970s and early 1980s, the derived model of the driving task still serves as a baseline for current research and engineering work. A recent study of Lappi targeted deliberate practice (DP) procedures in professional motorsport. It states that true expertise in any domain is not the result of just exceptional cognitive ability or superior sensorimotor skills. It is rather the result of extraordinary amounts of specific practice to obtain said abilities. DP means designing special tasks that give immediate and unambiguous feedback, which are then repeated with the goal of improving a specific skill. A review of the available literature reveals DP procedures that are used in the motorsport context. These are then classified into groups. Following the presented definition of the driving tasks, practice routines are grouped into control, guidance and navigation categories. While the details of the respective practice procedure are of less importance in the scope of this thesis, their organisation shows that the three-level model of the driving task is used in recent studies, as well as its interdisciplinary relevance. An example for a DP routine on the control level is to target improvement of the sense for the available tyre grip. Training visual skills, as in guiding the focus of the eyes specifically, on the other hand falls into the guidance category. [28]

## 2.2 Driver Modelling

Another achievement of the previously discussed model of the driving task is the architecture it provides for driver modelling, which is becoming more and more important with regards to driver assistance systems and autonomous vehicles [68]. This subsection summarises the different approaches to this task and provides more detailed explanations for some exemplary works.

Plöchl and Edelman gathered articles on driver modelling for different purposes. Depending on the objective, driver models can be specifically designed for different aspects of the driving task. [43] A focus on the vehicle is generally used if a virtual test driver is needed for tasks such as vehicle component design and validation or analysis of a vehicles dynamic behaviour [2][23]. Contrary there are models which specifically target driver behaviour and aim to improve the understanding of this topic [32][51]. Furthermore, the investigation of the interaction of driver and vehicle is analysed, exemplary in the field of Advanced Driver Assistance Systems (ADAS) [40][16]. Finally, studies with more than one car focus on topics such as car following or traffic flow and employ driver models suited for this purpose [8][31]. Overall, this shows that there is no single driver model capable of solving the different tasks associated with vehicle simulation.

The concept of Prokop's work includes the description of a human driver as a dynamic controller with the following abilities: The use and coordination of sensory perception, the ability to learn a dynamic systems behaviour as well as the capacity to optimise this

behaviour. This leads to using a model predictive control approach, which consists of a prediction model and an inner control loop to compensate deviations. The prediction model can be varied regarding its complexity, resembling driver experience according to the author. Essentially, this means that a model containing yaw dynamics will result in a driver model that is more capable of handling situations with non-linear vehicle behaviour, contrary to a point mass model that will rely mostly on the compensatory control in such situations. Additionally, the behaviour of the modelled driver can be adjusted by the cost function used for optimisation of the inputs to the vehicle (longitudinal, steering and gear), which consists of multiple objectives. The weights applied to each of the objectives dictate the outcome of the optimisation. Exemplary objectives are time-optimal, meaning to cover a given distance in the shortest possible time, and acceleration-optimal, which implies minimising accelerations that are felt by the passengers as a means to improve comfort. Constraints are used to force the driver to stay on the road, limit possible accelerations due to tyre potential, limit the engine speed, and impose the ability to stop the vehicle in half sight distance. The driver model is verified against human drivers with different objectives defined as the most important and shows good correlation with the participants' control inputs. Finally, a double lane-change manoeuvre is simulated at different speeds to assess the driver models capability of controlling the vehicle. While it is able to handle the manoeuvre at lower speeds, it shows difficulties to keep the car on track at higher speeds. This is explained by the used linear prediction model, which is exemplary for a driver that has no experience with highly dynamic driving. [45]

Wegscheider aims to provide a method to evaluate ADAS. The hypothesis is that generally a support system that behaves similar to the current driver is expected to get a good subjective rating, given it does what the driver would do. Based on this assumption a reference driver behaviour is extracted from experimental data for four driver types which are rated by aggressiveness. The aforementioned reference behaviour is based on lateral and longitudinal accelerations of the vehicle as well as cornering radius and lateral deviation from the driving lane. A driver model is then used to generate the reference behaviour for simulations using ADAS. The author is using the previously discussed model of Prokop [45] for this purpose, with two additional cost function objectives. The experimentally acquired reference behaviour is implemented through a map of lateral and longitudinal accelerations, which are depending on the current car speed. This is essentially a modification of the existing envelope of the vehicle's capabilities under the assumption that the driver does not use all of the available potential. Second to that, the lateral and longitudinal jerk is evaluated against a target value. The hypothesis is subsequently evaluated in an experiment. Trajectories that were generated with the driver model are driven by a real vehicle that is capable of following given path and acceleration target values using a method developed by Waldmann [62]. This allows to drive test persons around a test track autonomously, using their own driving behaviour and evaluate it regarding comfort. The original hypothesis is proven by the experiment. [64]

Prettenthaler investigates existing driver models and identifies the need for a path planning method specific to motor racing. The motivation for this results in the dependence

of the fastest trajectory to the given car setup, which means a generic racing line will not capture these differences. The main optimisation criteria in this case is minimising time, or as used in this implementation, maximising the travelled distance for a given time. A mathematical model of the vehicle is derived from the simulation model that is to be used with the generated trajectory, however in a form that is usable for optimisation algorithms. Additionally, an objective function containing the optimisation goals is defined. It consists of the main target of maximising the travelled distance, as well as other conditions such as the available vehicle performance, track limits, and maximum gradients for driver inputs. The optimisation is then carried out for multiple segments due to limitations of the polynomial description of the track. Since the optimal path through a single corner is dependent on the track characteristic ahead of it, these optimised segments are overlapped. This induces some redundancy but provides reasonable starting points for each following segment. Finally, the author compares the results for two different vehicles and illustrates the subsequently different optimal paths around the same race track. [44]

Togelius investigates computational intelligence in the domain of video games, where car racing games are exemplarily discussed. Besides ways of controlling simulated vehicles, means to imitate *driving styles* are presented. Modelling such human behaviour can be achieved by either direct or indirect methods. Direct modelling incorporates training a driver model on specific car states and the resulting control inputs of the driver. Contrary the indirect method means tuning an existing driver model to match the desired behaviour, which is found to produce a more stable solution. [55]

The research of Wang and Liaw aims to imitate *driving styles* with non-person players in a computer game based on the indirect modelling principle. This is achieved by tuning a driver controller that incorporates fuzzy logic to determine acceleration and steering targets. The parameters of this controller are then optimised with a multi-objective approach, that targets lap time and *driving style* similarity to a target record. The latter is defined by a vector of driver control actions as well as curvature of the track ahead of the car in 10 m intervals and a range sensor. The defined similarity function compares the instantaneous vector with a range of samples around the current position from previously recorded data. Overall, the approach seems to be capable to quantify *driving style* similarities between a driver and a record for the respective track. [63]

## 2.3 Driving Style

*Driving style* or driver behaviour recognition is a broadly followed research topic in the road car domain. Some of the main applications are driver modelling, advanced driver assistance systems and safety assessment as well as economic rating and assistance with driving in a sustainable way. Some concepts are outlined in this section, although none deals with the problem of individual nuances of *driving style* in a high-performance motorsports environment.

A good overview of recent research is provided by Martinez et al., which summarises available literature concerning driver recognition. The general motivation is summarised as the progression from vehicle software tailored for a standard driver to adaptive systems. Additionally, *driving style* is differentiated from driver behaviour, where the former includes all external influences while the latter is solely focused on driver decisions. Furthermore, both terms should not be mistaken with driver skill, the ability of a driver to control the vehicle to his/her desire. The reported methods cover influences from driving events up to human factors such as character traits and are implemented through rules, through models or through machine learning. [33] Meiring et al. provide another review of current literature with a focus on machine learning and artificial intelligence [37]. The work of Martinez et al. provides a definition of *driving style* as follows:

Hereby driver *driving style* is understood as the way the driver operates the vehicle controls in the context of the driving scene and external conditions such as time in the day, day of the week, weather and mood, between other factors. This definition agrees with previous descriptions and contemplates detecting more than one style for the same driver. That is to say that the same driver could exhibit disparate styles under different conditions ... [33]

An often seen approach is to assess drivers into general categories with different methods. A classification into the categories comfortable, normal and sporty is used by Dörr et al., who judge *driving style* by commonly available signals in road cars, such as inertial accelerations and vehicle speed. Furthermore, information from the navigation system is used to employ different subsystems for varying road types, for example urban streets, rural roads or motorways. Each subsystem uses event detection with following fuzzy logic to output an overall *driving style* coefficient. [12] Lei et al. propose the quite similar economical, moderate and sporty types and establish a connection between those classes and the dynamic driver demand, which is calculated from vehicle speed and acceleration. Additionally, a different method for online calculation and prediction is presented. [29] Constantinescu et al. use statistical data obtained from time-series, such as the mean velocity or relative amount of time spent at velocities exceeding 60km/h, together with data mining techniques. The results of cluster analysis as well as principal component analysis lead to a classification of drivers into five levels of aggressiveness. [7] Similarly, Qi et al. propose three levels of aggressiveness, although the classification is using latent Dirichlet allocation, which is originally intended for text mining. Their study contains recorded data and additional correlation analysis with questionnaires completed by participants of the study. [46]

Driving skill is commonly rated into novice and expert categories. You et al. employ a two-point visual driver model to derive information about the driving skill regarding steering inputs from the model parameters. Kalman filtering is used for state and parameter estimation. The study contains a field test for data gathering and shows separability of skilled and novice drivers using the employed model. Furthermore, wavelet transformation is used to rate steering smoothness. [71] Zhang et al. use a different approach to classify steering behaviour into the same skill levels. Data is sampled from two driving

manoeuvres and discriminant features are obtained by the coefficients of discrete fourier transformation. These are then classified using artificial neural networks, support vector machines, and decision trees. [72]

A different approach is detailed by Taubman et al. for *driving style* assessment from a more psychological point of view. The multidimensional *driving style* inventory is introduced, which contains 44 items that are organised into eight categories derived from factor analysis. Exemplary categories are risky or high-velocity *driving styles*, as opposed to patient or careful *driving styles*. The data was gathered by questionnaires where each participant noted his/her agreement with the 44 items. Additionally, the results were correlated with personality traits obtained from further questionnaires completed by each participant, employing existing testing methods for the respective personality trait, such as global self-esteem or desire for control. [53]

Chen et al. approached the subject of *driving style* recognition from a big data point of view with the aim to provide measures for fleet management and individual insurance policies based on detected *driving style*. Commonly available vehicle data is logged and stored in a cloud computing solution, with further analysis to select discriminant features by feature selection methods. The selected signal, car velocity, is then analysed by statistical means to derive the skewness of its distribution for individual trips from the original time-series data, which forms the basis for a rule-based classification into four *driving styles*: novice, cautious, aggressive and expert. [6] While this approach uses methods of unsupervised learning, contrary to the supervised learning approach presented in this work, the general idea of gathering a large data set for further processing with feature selection methods forms a part of the presented methodology.

Deviating from the discussed aims for *driving style* recognition, Schöggel et al. present a method to adopt road car characteristics based on individual *driving style*. The vehicle is rated based on categories, exemplary sportiness and comfort. These ratings are derived from a driving mode detection using Fuzzy Logic and following evaluation of data by a neural network which was trained with subjective ratings. The *driving style* is then assessed from seven commonly available signals on road cars, such as clutch pedal position or engine torque. Consequently, characteristic objective measures are obtained from these signals, exemplary the engine speed before de-clutching. These criteria are then used to calculate the factor levels of eight defined driver criteria with sporty or economical being two of them. The combination of both sets of metrics allows analysis of mutual influence and ultimately the online adoption of drive train characteristics to a specific driver. [52] The general idea of adjusting the vehicle's dynamic behaviour to the driver is very similar to the problem stated in chapter 1, although the main goal is to provide the customer with a vehicle tuned to his/her specific preference, as opposed to maximising the performance of the combination of driver and vehicle at the limits of driving dynamics. The approach is interesting nevertheless, because the derived metrics are focused on detailed examination rather than statistical evaluation, which is a concept that is used in this work as well. Unfortunately, the method is not published to a level of detail that allows examination of the approach. Above all the objective metrics

necessary to distinguish race car drivers achieving similar high levels of performance from each other requires significantly different metrics, which are defined, assessed and reduced to a usable amount in the following chapters.

Trzesniowski proposes Key Performance Indicators (KPIs) in the context of data analysis and driver rating within the motorsport environment. Suggested metrics include, for example, derivatives of the driver's control inputs to the car which are then averaged over a lap. Further calculations propose limiting such signals to specific scenarios, exemplary when the driver is on full throttle, before averaging or relating control inputs to the achieved accelerations of the vehicle. [59] This approach relies on absolute values which incorporate all three dynamic systems outlined in the motivation of this thesis. For this reason the obtained measures are only relevant to compare different drivers if all other boundary conditions remain unchanged, which essentially means the same car as well as identical track conditions are mandatory. The proposed method of data analysis allows general performance indications, however the presented data does compare an inexperienced with an expert driver where larger differences can be expected, contrary to comparing drivers on similar performance levels.

In summary, a detailed objective description of race car driver's *driving styles* requires a new approach, which is elaborated and evaluated in the following. The main goal is to find objective metrics showing significant differences between drivers who are all competing on a similar high level and achieve comparable performance, yet with different car setups tailored to their individual *driving style*.



## 3 Methodology

The following chapter provides a detailed description of the methods used to accomplish the goal of characterising *driving style* for individual drivers. This includes the necessary data, the approach to processing this data and the calculation of the derived metrics in detail. Additionally, it presents methods to evaluate the newly defined metrics and their significance for race car *driving style* recognition.

To evaluate the *driving style* of race car drivers, the three-level model of the driving task, as explained in section 2.1, is used in compliance with the fact that race car driving is a driving task as well. While the purpose of the stabilisation task is the same for driving on public roads and race car drivers, there are fewer similarities on the guidance level. The target to derive a corridor for nominal vehicle movement is comparable, although the objective is shifted towards finding the fastest possible path around a race track. Overtaking manoeuvres are also related to this level, yet they are not within the scope of this thesis. The navigation level is of least importance for this study, as the course is given and no routing decisions have to be taken, except for pit-stops which are not considered in this thesis. While an overtaking manoeuvre falls in the guidance level, the overall overtaking strategy, for instance, by studying opponents mistakes over multiple laps to find the right time and location for an attempt to pass, shares similarities with the tasks on the navigational level. Due to relevance for the *driving style* of race car drivers only the stabilisation and guidance levels are considered for analysis. As suggested by the model of the driving task, the evaluation of *driving style* is split into the control inputs of the driver to the car and the trajectory of the vehicle.

The methodology used to define new metrics and evaluate their meaningfulness is similar for the control inputs and the trajectory. While this is originally a problem of finding objective metrics to describe dynamic behaviour, the process is inspired by a classification problem that can be solved with machine learning. This typically involves feature extraction and feature selection. The term feature is commonly used in the machine learning domain, and within the scope of this work the defined objective metrics to describe the driver behaviour are features of the dataset used for the classification task. Additionally, a set of features describing the same instance, or sample, is commonly referred to as an observation. The approach is split into the feature or metric extraction using domain knowledge about vehicle dynamics of race cars as well as the empiric knowledge of differences in *driving style* among race car drivers, and the feature selection part relying on data mining and machine learning methods. The aim is to gather a large set of metrics without further assumptions about significance or redundancies. Consequently, an assessment of the criteria is necessary. This is accomplished by defining a classification problem of identify-

ing the driver from the given set of metrics. As the driver is known for each observation, this can be solved using supervised learning. The feature selection methods, which are described in more detail in section 3.3, result in a set of features or objective metrics, that are most useful to classify the driver. In other words these metrics capture the individual differences among the drivers, which is their *driving style*. An important notice is that classifying the driver is used to evaluate the defined metrics for the *driving style* of race car drivers. The absolute classification accuracy is therefore an indication but not linked to the main goal of this thesis, which is providing numeric metrics instead of categories.

## 3.1 Stabilisation Level

The stabilisation level of the driving task focuses on correcting lateral and heading errors to the nominal track, as well as maintaining the nominal speed profile. This is achieved by using the control inputs of the car which are mainly throttle pedal, brake pedal and steering wheel in the context of this work. Secondary controls include the gear shifter, Drag Reduction System (DRS) activation, if available, and the clutch pedal or lever for example. The three main control inputs and their respective representation in logged vehicle data are selected to be used for the following approach. The methodology elaborated in this subsection is partially published by Wörle et al. in [69].

The mentioned data is usually recorded over the entire outing<sup>1</sup> of the car and split into laps. Consequently, the data contained in a single lap needs to be split into relevant parts. As illustrated in Fig. 3.1, an ideal lap allows the classification of the vehicle state into few categories. With the assumption of an ideal lap, meaning the car is always driven at the maximum possible accelerations, it will either be limited by power available from the drive train, hence "power limited", or by the forces the tyres can transmit, thus "grip limited". A power limited state can occur on straights and corners that are driven with full throttle. Grip limited states however imply that the driver is not on full throttle, but either limited by lateral or longitudinal acceleration. This can happen either in a typical corner where the driver lifts the throttle, applies the brakes and then steers into the corner or in corners where the driver only lifts the throttle before turning the steering wheel. Due to the majority of corners falling in the former category and all available primary control inputs to the car being used, this pattern is decided to be taken into account for *driving style* characterisation, as highlighted in Fig 3.1.

### 3.1.1 Detection of Characteristic Points

Due to the focus on cornering states of the car for further analysis, the available data, which is usually split into single laps, needs further segmentation into single cornering events. As defined, these need to have a braking phase to be considered. Within this

---

<sup>1</sup>The term outing means the car is leaving the pits and can be used to enumerate those runs of the car over a day

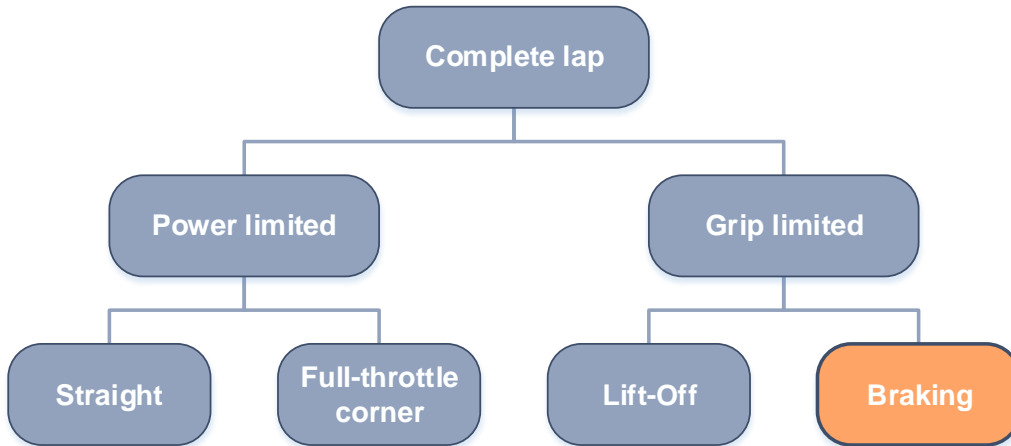


Figure 3.1: Classification of vehicle states

scope the throttle signal  $r_T$  and brake signal  $p_B$  follow a very distinct pattern for each corner, except for any irregularities which cause a lap to be not representative. These patterns are exploited to detect cornering and segment data for further analysis on a single corner basis.

Exemplary signals for one corner over lap time  $t_L$  are plotted in Fig. 3.2. A clear pattern can be seen in  $r_T$  and  $p_B$ , as the driver arrives to the corner with full throttle and no brake pressure applied, then lifts the throttle and increases brake pressure, which is followed by a fade out of the brake pressure and transition to full throttle again. This leaves two static reference values for  $r_T$  and one for  $p_B$  respectively, that allow detection of cornering events from the signals. However, the steering wheel angle  $a_S$  does not allow such a clear distinction because the signal in this example is positive before the braking phase starts and then turns negative, meaning the driver is steering to the left until he turns the steering wheel right for the following corner. Additionally, the driver turns the steering wheel about  $-20^\circ$  until the braking phase is finished, followed by a decrease to reach the desired value. Although this is just one exemplary corner, it illustrates why  $a_S$  is not suitable for robust cornering detection. Therefore, the signal is analysed with respect to the events found in  $r_T$  and  $p_B$ .

Static signal values for  $r_T$  are usually 0% and 100%, while  $p_B$  should remain static at 0 bar respectively. However, data logging is prone to measurement errors, for example due to misconfiguration or electrical interference. For this reason, the static signal values are calculated for each lap to allow robust analysis of data from various sources. In the first step the extreme values of the signal are calculated as the fifth and 95th percentile of the signal range,  $r_{T,p5}$  and  $r_{T,p95}$  exemplary for the throttle signal, to exclude outliers. The static intervals are then found as the areas where the signal remains within five

| Minimum               |        | Maximum               |          |
|-----------------------|--------|-----------------------|----------|
| Minimum signal value: | 0.00 % | Maximum signal value: | 104.95 % |
| Fifth quantile:       | 3.03 % | 95th quantile:        | 100.06 % |
| Static minimum value: | 3.11 % | Static maximum value: | 100.00 % |

Table 3.1: Exemplary values for static throttle signal calculation

percent of the extreme signal values. To remove spikes and gradients, these intervals are cut by 0.5 s on each end, which also eliminates static intervals shorter than one second. The resulting intervals are indicated in Fig. 3.2 by dashed and dotted lines. The lines marking the intervals are scaled to 110 % or 30 *bar* respectively for better illustration. Finally, the static signal values are calculated as the median of all values within either the maximum or minimum intervals, resulting in  $r_{T,\min,s}$  and  $r_{T,\max,s}$  for the throttle signal. The approach to consider all values within a range around the extreme values is chosen for robustness, since the signal does not stay ideally static. Therefore, a range that can be considered as static has to be defined. Depending on data quality, this range can be adjusted. Errors due to this range are compensated by the mentioned cutting of intervals and the usage of the median value. Exemplary numbers for the calculation are listed in table 3.1, clearly showing the advantage of the method as the maximum signal value is an outlier.

With defined static signal levels, the desired patterns can now be detected by comparing the data with these static signal values. The throttle signal is chosen to be described by four specific points, the brake signal features three characteristic points respectively, and the steering signal features one such point. These are introduced and further described in table 3.2. Additionally, plots of the signals with marked characteristic points are illustrated in Fig. 3.3.

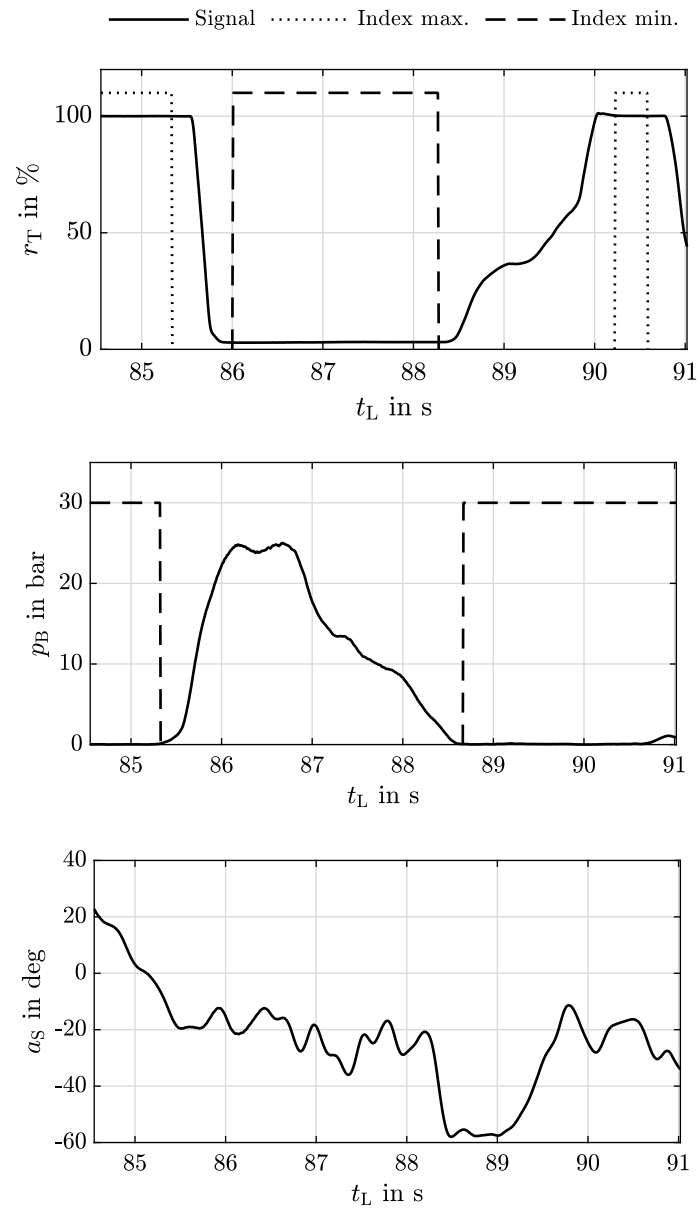


Figure 3.2: Throttle, brake pressure and steer wheel angle signals with detected extreme value intervals

---

|                      |  |
|----------------------|--|
| Throttle signal      |  |
| <i>Off-throttle</i>  | Where the driver starts to lift the throttle pedal, defined by a deviation from the maximum static value.                |
| <i>No-throttle</i>   | When the driver has released the throttle pedal completely, defined by a value within range of the minimum static value. |
| <i>On-throttle</i>   | When the driver starts to apply the throttle, defined by a deviation from the minimum static value.                      |
| <i>Full-throttle</i> | When the driver reaches full throttle again, defined by a value within range of the maximum static value.                |

---

|                  |   |
|------------------|---|
| Brake signal     |   |
| <i>On-brake</i>  | When the driver starts to press the brake pedal, defined by a deviation from the minimum static value.            |
| <i>Max-brake</i> | When the brake pressure reaches a maximum, defined by the first maximum in the signal.                            |
| <i>Off-brake</i> | When the driver releases the brake pedal completely, defined by a value within range of the minimum static value. |

---

|                 |  |
|-----------------|--|
| Steering signal |  |
| <i>Turn-in</i>  | An artificial point where the driver turns the steering wheel for the upcoming corner, defined by reaching a percentage of the integrated steering angle used throughout the corner. |

---

Table 3.2: Description of characteristic points for detecting cornering of a race car

The first characteristic point occurring in the throttle signal is the *off-throttle* point, however the *no-throttle* point is detected first and serves as a reference for detection of the remaining points of the pattern. This approach enhances robustness as only occasions where the driver fully lifts the throttle are detected and further analysed. A partial lift, as could be seen for a corner where no braking is needed or after going on full throttle to correct a driving mistake, is ignored and not considered a cornering event. A percentage of  $r_{T,\max,s}$  is considered for detection of all points. It is chosen to be one percent, however this value can be varied depending on the data to analyse in accordance with the previously explained method to calculate the static signal values. As mentioned, the *no-throttle* point is found first by comparing  $r_T$  to  $r_{T,\min,s}$ . The sample where  $r_T$  is first within an interval of one percent around  $r_{T,\min,s}$  is considered the *no-throttle* point, as shown in (3.1). From this reference the *off-throttle* point is found as the last sample within the defined range of  $r_{T,\max,s}$  in compliance to (3.2). As illustrated in the

throttle signal plot in Fig. 3.3 there might be throttle blips<sup>2</sup> in the signal, therefore the *full-throttle* point detection is next in the sequence. It is defined as the first data sample within the defined range around  $r_{T,max,s}$ , with the condition shown in (3.2), similar to the *off-throttle* point. From this point the *on-throttle* point is found as the last point within the interval around  $r_{T,min,s}$ , analogue to the *no-throttle* point and as per (3.1).

$$r_T \leq r_{T,min,s} + 0.01 * r_{T,max,s} \quad (3.1)$$

$$r_T \geq r_{T,max,s} - 0.01 * r_{T,max,s} \quad (3.2)$$

Similar to the process of finding characteristic points for  $r_T$ , the defined points for  $p_B$  are found by comparing the signal to  $p_{B,min,s}$ . Since there is no maximum static value, the 99th percentile  $p_{B,p99}$  is used as maximum brake pressure for the respective lap. This value is needed to define the one percent interval around  $p_{B,min,s}$ . It cannot be calculated for each corner individually, because this would require knowledge of the specific points for the calculation of which the value is intended originally. An alternative approach by using the extreme value in between detected throttle points would be possible but was neglected to keep the pattern recognition independent for each signal. With the now defined boundary values, the *on-brake* point can be found as the first sample exceeding the one percent interval around  $p_{B,min,s}$ , as indicated in (3.3). The *max-brake* point is then found as the first peak following this point, indicating the first maximum in  $p_B$ . For many datasets this coincides with the absolute maximum during the corner. However, as Fig. 3.3 shows the driver might increase the brake pressure after shortly releasing the pedal. This is already an example for driver specific behaviour, as discussed in the following section 3.1.2. Finally, the *off-brake* point is found from the *max-brake* point as the first sample to be within the defined interval around  $p_{B,min,s}$ , similar to the condition for the *on-brake point*, again with reference to (3.3).

$$p_B \leq p_{B,min,s} + 0.01 * p_{B,p99} \quad (3.3)$$

As previously explained and illustrated in Fig. 3.2,  $a_S$  does not show patterns of similar distinction compared to  $r_T$  and  $p_B$ . Therefore, a different approach is used to characterise the *turn-in* point. The *turn-in* point can be idealised as the instant where the driver turns the steering wheel for the upcoming corner, however it is not necessarily remaining at a static value before due to constant use for stabilisation of the car. The *off-throttle* point is chosen as a reference point for *turn-in* point detection for this reason. Furthermore, the integral of  $a_S$  is used to ignore short steering intervals for the detection. Such short steering events can be seen, for example, when the driver has to correct a car that is unstable under braking, where considerable amplitudes in  $a_S$  can be noticed. The reference value for the *turn-in* point is defined as the integral of  $a_S$  from the time at the *off-throttle* point  $t_{off}$  to the time  $t_{S,max}$  where the maximum steering angle within the

<sup>2</sup>Automatic opening of the throttle body by the ECU to control and match the engine revolutions during down shifting.

detected throttle event occurs. The *turn-in* point is then found as the sample where the integral of  $a_S$  reaches five percent of the reference value, as shown in (3.4). This method defines an artificial point to characterise the steering behaviour which can be used in relative comparison between datasets, however its absolute value has to be treated with care. This definition is used for the approach presented in this work but does not claim to be the right description for the *turn-in* point as this would be a debatable assertion.

$$\int_{t_{\text{oft}}}^{t_{\text{tui}}} a_S dt \geq 0.05 * \int_{t_{\text{oft}}}^{t_{S,\text{max}}} a_S dt \quad (3.4)$$

Following the identification of cornering patterns for both  $r_T$  and  $p_B$ , a further plausibility check is necessary with a possible fault being a mismatch in the total number of throttle and braking events. This is solved by an algorithm that puts together each pair of throttle and brake events which share the minimum distance among all combinations and dismisses any superfluous events. Firstly, the matrix  $\mathbf{D}$  containing the distance in samples between any two detected events is calculated, as presented in (3.5), by subtracting each brake event index  $N_{\text{onb}}$  from each throttle event index  $N_{\text{oft}}$ . An exemplary result is displayed in (3.6). Because seven throttle events and eight braking events were detected for this lap,  $\mathbf{D}$  is a 8x7 matrix. The brake events are represented by rows and throttle events are gathered by the columns of  $\mathbf{D}$  respectively. In this case it is evident that brake event six needs to be removed to come to the desired solution of matching all close by throttle and brake events. This requires the minima of each row and each column to be on the diagonal of the matrix which needs to have the same size in both dimensions. In other words this means that there is a similar number of throttle and brake events which share the same order.

$$\mathbf{D}_{k,l} = N_{\text{oft},k} - N_{\text{onb},l} \quad (3.5)$$

$$\mathbf{D} = \begin{bmatrix} 9 & -918 & -3280 & -5054 & -6527 & -7145 & -8119 \\ 933 & 6 & -2356 & -4130 & -5603 & -6221 & -7195 \\ 3298 & 2371 & 9 & -1765 & -3228 & -3856 & -4830 \\ 5069 & 4142 & 1780 & 6 & -1467 & -2085 & -3059 \\ 6557 & 5630 & 3268 & 1494 & 21 & -597 & -1571 \\ 6846 & 5919 & 3557 & 1783 & 310 & -\mathbf{308} & -1282 \\ 7157 & 6230 & 3868 & 2094 & 621 & \mathbf{3} & -971 \\ 8139 & 7212 & 4850 & 3076 & 1603 & 985 & 11 \end{bmatrix} \quad (3.6)$$

In detail the example for  $\mathbf{D}$  shown in (3.6) illustrates that the first detected throttle event is closest to the first detected braking event. The first column indicates the distances for all braking events to the first throttle event. Similarly, the first row indicates the distances for all throttle events to the first braking event. The minimum absolute value for both the first row and the first column is nine and has the same index in both. This



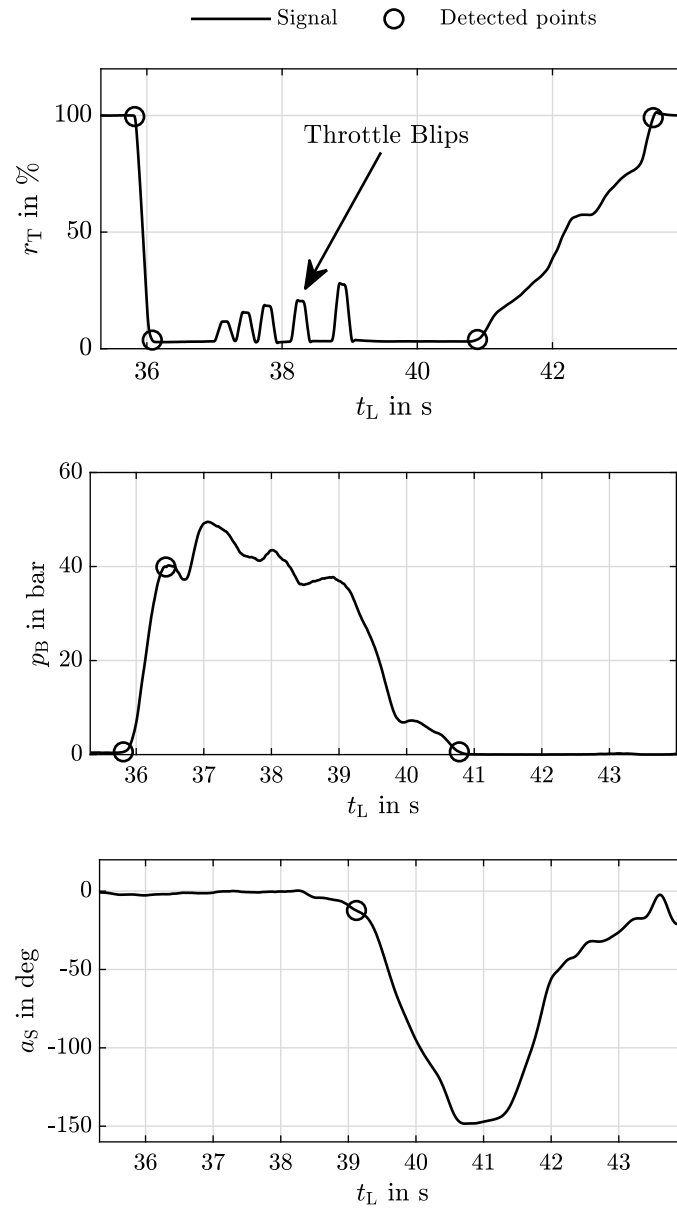


Figure 3.3: Throttle, brake pressure and steering wheel angle signals with detected characteristic points

means the first braking event is located closest to the first throttle event and the other way round. The same pattern can be seen for rows and columns two to five, which is the desired end result. As already outlined, throttle event six, noted in column six, has a minimum absolute distance of three samples to braking event seven. Braking event six, noted in row six, has its minimal absolute distance of 308 samples relative to throttle event six as well. This leaves two possible braking events for one throttle event. According to  $3 < 308$ , braking event seven is chosen to match with throttle event six because it is located closer to it, while braking event six is removed. Following this practical example, the algorithm for the plausibility check compares the minimum absolute values for each row and column. If they coincide no further action is necessary. If there is a difference, the combination with the lower absolute distance is kept while the remaining events are removed. The resulting matrix  $\mathbf{D}$  in the desired form, with the minima on the diagonal, is shown in (3.7).

$$\mathbf{D} = \begin{bmatrix} \mathbf{9} & -918 & -3280 & -5054 & -6527 & -7145 & -8119 \\ 933 & \mathbf{6} & -2356 & -4130 & -5603 & -6221 & -7195 \\ 3298 & 2371 & \mathbf{9} & -1765 & -3228 & -3856 & -4830 \\ 5069 & 4142 & 1780 & \mathbf{6} & -1467 & -2085 & -3059 \\ 6557 & 5630 & 3268 & 1494 & \mathbf{21} & -597 & -1571 \\ 7157 & 6230 & 3868 & 2094 & 621 & \mathbf{3} & -971 \\ 8139 & 7212 & 4850 & 3076 & 1603 & 985 & \mathbf{11} \end{bmatrix} \quad (3.7)$$

### 3.1.2 Objective Criteria

After the detection of characteristic points for each corner, objective metrics can be calculated based on these points. The main goal is to derive scalar values that describe the content of a time-series signal, more accurately any driver specific features concerning the inputs of the driver to the car on the stabilisation level. As this topic is barely examined and little information is available, the approach is to generate a wide range of metrics which are potentially insignificant or redundant. Thereupon the defined metrics are to be evaluated further in the following feature selection process to select a range of significant objective metrics, which describe the *driving style* on the stabilisation level, hence the control inputs of the driver.

The simplest set of metrics consists of gradients as well as distances and times between detected corner points. These can be calculated directly from the respective signal value at the characteristic points. A list of those metrics is provided in table 3.3 with an explanatory plot for each metric. In these graphs the lap time is denoted as  $t_L$ , whereas  $s_L$  is the current distance along the lap. The *off-throttle* gradient  $drThrOff$ , for example, is calculated as the quotient of the throttle difference between the *off-throttle* and *no-throttle* point and the time difference between those points, as indicated in (3.8). Similar gradients can be calculated for the *off-throttle*, *on-throttle*, *on-brake* and *off-brake* phases as a linear approximation of the driver's behaviour when applying those controls.

Furthermore, distances and time intervals between corner points are considered. The *rolling* phase, for example, is the time where the driver does neither engage throttle or brake, hence the car is rolling. This phase is defined by the *off-brake* and *on-throttle* points. The absolute distance covered between those, referred to as absolute *rolling* distance  $sRollAbs$ , can be calculated by subtracting the respective lap distance values  $s$  as shown in (3.9). The distance covered in the *rolling* phase relative to the length of the corner  $sRollRel$  is then obtained through dividing  $sRollAbs$  by the distance between *off-throttle* and *full-throttle* point in accordance to (3.10). Additionally, the time  $t$  passed during this phase is calculated in the same form as absolute and relative metric. Another example is the *trail braking* phase, which is characterised in a similar way to the *rolling* phase. *Trail braking* is a common term in motor racing for steering into a corner and braking at the same time. Consequently, the points of interest for these metrics are the *turn-in* point and the *off-brake* point to detect their overlap. Three additional metrics have been defined to describe the corner of the observation instead of the driver's inputs, as the latter will probably be different for different corner types. To add some information about the corner, the maximum and minimum velocity of the car within the detected corner points, as well as their difference are added to the feature set. While the minimum velocity gives an indication of the cornering speed at approximately maximum lateral load, the combination of both, hence the difference, is a measure for the severity of the braking phase before the corner.

$$drThrOff = \frac{r_{T,oft} - r_{T,not}}{t_{oft} - t_{not}} \quad (3.8)$$

$$sRollAbs = s_{ont} - s_{ofb} \quad (3.9)$$

$$sRollRel = \frac{sRollAbs}{s_{fut} - s_{oft}} \quad (3.10)$$

| Nr. | Metric Name | Illustration | Description   |
|-----|-------------|--------------|---|
| 1   | $drThrOff$  |              | Gradient of $r_T$ respective to time between <i>off-throttle</i> and <i>no-throttle</i> point |

Table 3.3: List of gradient, time and distance based metrics on the stabilisation level

| Nr. | Metric Name  | Illustration | Description  |
|-----|--------------|--------------|--|
| 2   | $drThrOn$    |              | Gradient of $r_T$ respective to time between <i>on-throttle</i> and <i>full-throttle</i> point   |
| 3   | $dpBrkOn$    |              | Gradient of $p_B$ respective to time between <i>on-brake</i> and <i>max-brake</i> point  |
| 4   | $dpBrkOff$   |              | Gradient of $p_B$ respective to time between <i>max-brake</i> and <i>off-brake</i> point   |
| 5   | $tBrkDlyAbs$ |              | Absolute time passed between <i>off-throttle</i> and <i>on-brake</i> point   |
| 6   | $sThrOffAbs$ |              | Absolute distance covered between <i>no-throttle</i> and <i>on-throttle</i> point  |
| 7   | $sThrOffRel$ |              | Distance covered between <i>no-throttle</i> and <i>on-throttle</i> point relative to the distance between <i>off-throttle</i> and <i>full-throttle</i> point |

Table 3.3: List of gradient, time and distance based metrics on the stabilisation level

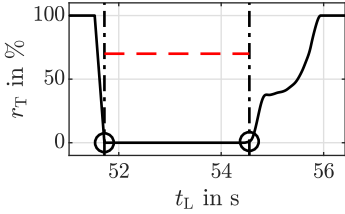
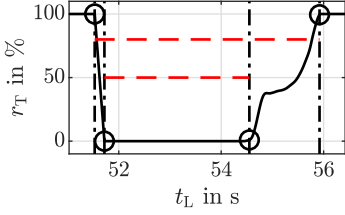
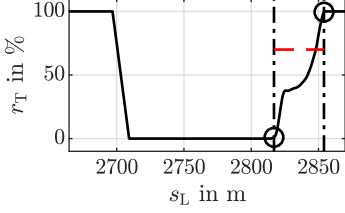
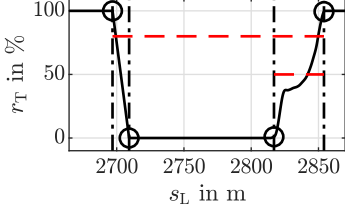
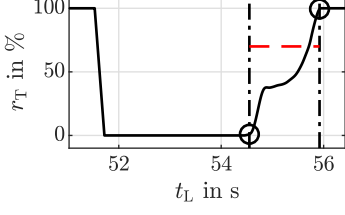
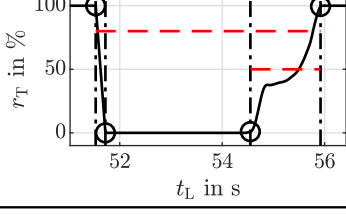
| Nr. | Metric Name  | Illustration  | Description  |
|-----|--------------|---|--|
| 8   | $tThrOffAbs$ |    | Absolute time passed between <i>no-throttle</i> and <i>on-throttle</i> point   |
| 9   | $tThrOffRel$ |    | Time between <i>no-throttle</i> and <i>on-throttle</i> point relative to the time interval between <i>off-throttle</i> and <i>full-throttle</i> point          |
| 10  | $sThrOnAbs$  |   | Absolute distance covered between <i>on-throttle</i> and <i>full-throttle</i> point  |
| 11  | $sThrOnRel$  |  | Distance covered between <i>on-throttle</i> and <i>full-throttle</i> point relative to the distance between <i>off-throttle</i> and <i>full-throttle</i> point |
| 12  | $tThrOnAbs$  |  | Absolute time passed between <i>on-throttle</i> and <i>full-throttle</i> point   |
| 13  | $tThrOnRel$  |  | Time between <i>on-throttle</i> and <i>full-throttle</i> point relative to the time interval between <i>off-throttle</i> and <i>full-throttle</i> point        |

Table 3.3: List of gradient, time and distance based metrics on the stabilisation level

| Nr. | Metric Name | Illustration | Description  |
|-----|-------------|--------------|--|
| 14  | $sBrkAbs$   |              | Absolute distance covered between <i>on-brake</i> and <i>off-brake</i> point   |
| 15  | $sBrkRel$   |              | Distance covered between <i>on-brake</i> and <i>off-brake</i> point relative to the distance between <i>off-throttle</i> and <i>full-throttle</i> point    |
| 16  | $tBrkAbs$   |              | Absolute time passed between <i>on-brake</i> and <i>off-brake</i> point  |
| 17  | $tBrkRel$   |              | Time passed between <i>on-brake</i> and <i>off-brake</i> point relative to the time interval between <i>off-throttle</i> and <i>full-throttle</i> point    |
| 18  | $sRollAbs$  |              | Absolute distance covered between <i>off-brake</i> and <i>on-throttle</i> point  |
| 19  | $sRollRel$  |              | Distance covered between <i>off-brake</i> and <i>on-throttle</i> point relative to the distance between <i>off-throttle</i> and <i>full-throttle</i> point |

Table 3.3: List of gradient, time and distance based metrics on the stabilisation level

| Nr. | Metric Name  | Illustration | Description   |
|-----|--------------|--------------|---|
| 20  | $tRollAbs$   |              | Absolute time passed between <i>off-brake</i> and <i>on-throttle</i> point  |
| 21  | $tRollRel$   |              | Time between <i>off-brake</i> and <i>on-throttle</i> point relative to the time interval between <i>off-throttle</i> and <i>full-throttle</i> point       |
| 22  | $sTurnInAbs$ |              | Absolute distance covered between <i>off-throttle</i> and <i>turn-in</i> point  |
| 23  | $sTurnInRel$ |              | Distance covered between <i>off-throttle</i> and <i>turn-in</i> point relative to the distance between <i>off-throttle</i> and <i>full-throttle</i> point |
| 24  | $tTurnInAbs$ |              | Absolute time passed between <i>off-throttle</i> and <i>turn-in</i> point   |
| 25  | $tTurnInRel$ |              | Time between <i>off-throttle</i> and <i>turn-in</i> point relative to the time interval between <i>off-throttle</i> and <i>full-throttle</i> point        |

Table 3.3: List of gradient, time and distance based metrics on the stabilisation level

| Nr. | Metric Name | Illustration | Description  |
|-----|-------------|--------------|--|
| 26  | $sTrailAbs$ |              | Absolute distance covered between <i>turn-in</i> point and <i>off-brake</i> point  |
| 27  | $sTrailRel$ |              | Distance covered between <i>turn-in</i> point and <i>off-brake</i> point relative to the distance between <i>off-throttle</i> and <i>full-throttle</i> point |
| 28  | $tTrailAbs$ |              | Absolute time passed between <i>turn-in</i> point and <i>off-brake</i> point   |
| 29  | $tTrailRel$ |              | Time covered between <i>turn-in</i> point and <i>off-brake</i> point relative to the time between <i>off-throttle</i> and <i>full-throttle</i> point         |
| 30  | $vCarMax$   |              | Maximum velocity within the corner interval  |
| 31  | $vCarMin$   |              | Minimum velocity within the corner interval  |

Table 3.3: List of gradient, time and distance based metrics on the stabilisation level



| Nr. | Metric Name | Illustration | Description  |
|-----|-------------|--------------|--|
| 32  | $vCarDiff$  |              | Difference between maximum and minimum velocity within the corner interval |

Table 3.3: List of gradient, time and distance based metrics on the stabilisation level

Because the introduced metrics only capture linear aspects of the driver’s control inputs, further measures are defined to describe the non-linearity of the signal. The *on-throttle* and *off-brake* phases have been characterised by a linear gradient so far, which is extended by computing the Root Mean Square Error (RMSE) of the actual signal to its linear approximation for a measure of the linearity of the driver’s inputs. This is however only possible for  $r_T$  and  $p_B$ , because  $a_S$  does not have similar distinct patterns, for example, when the driver applies the throttle from 0% to 100%. Exemplary graphs are shown in Fig. 3.4, where the topmost graph displays  $p_B$ , the detected corner points and the linear approximation for the *off-brake* phase. The two additional graphs in Fig. 3.4 illustrate the first and second derivative of  $p_B$ . The derivatives are calculated from the respective signal with a low-pass filter applied to smooth it before differentiation. Within this work a finite impulse response (FIR)-filter with a passband of 2 Hz has been used. Each zero crossing in the first derivative  $p'_B$  represents either a local maximum or minimum in  $p_B$ . The zero crossings in the second derivative  $p''_B$  correlate with extreme values of the first derivative, which in turn are inflection points of  $p_B$ . A set of metrics is defined using  $p_B$ ,  $p'_B$  and  $p''_B$  to describe the number of peaks in  $p_B$ , as well as the number of zero crossings, the variance and the RMSE to zero for both derivatives. Those are applied in a similar way to  $r_T$  and  $a_S$ . A threshold of two is used for the detection of peaks, which means the respective signal needs to drop by this value on each side of the peak to be considered. These metrics are again expected to be correlated, however it is not clear which ones contain the most information about the driver, thus the definition of potentially redundant features to be evaluated at a later stage. These additional metrics are listed in table 3.4.

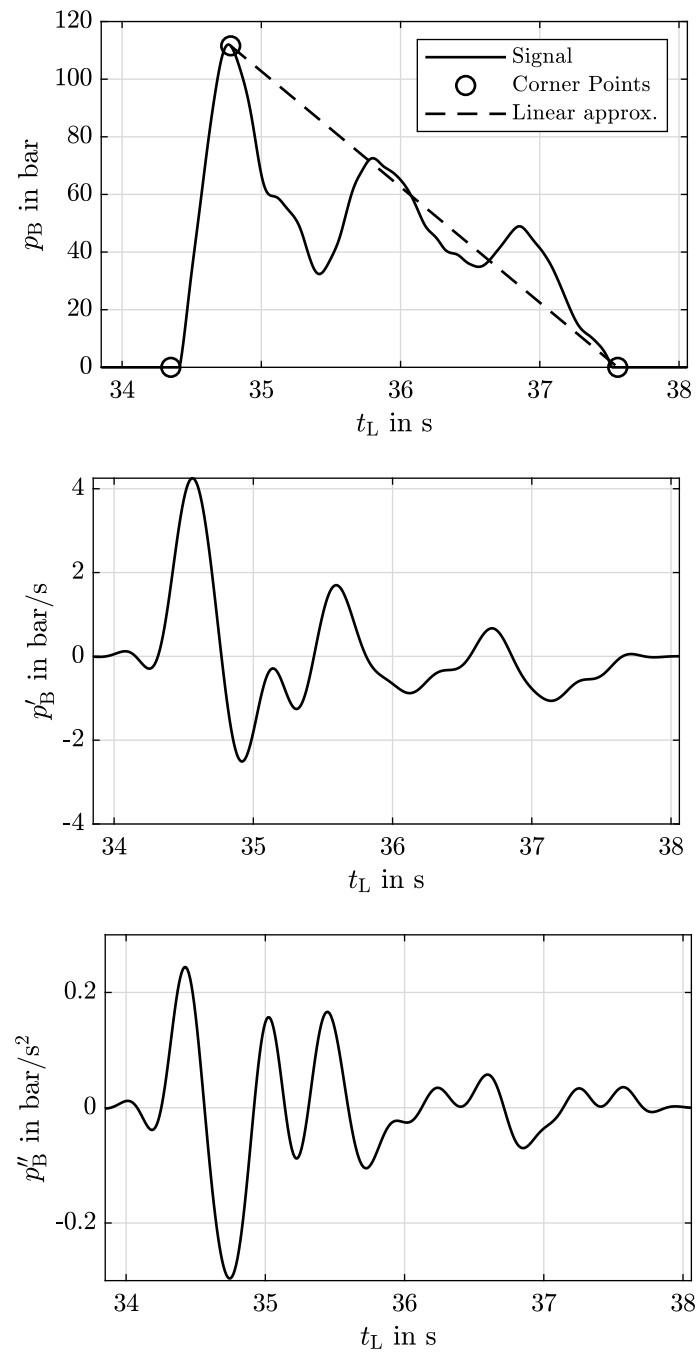


Figure 3.4: Exemplary brake signal with linear approximation for the *off-brake* phase, first and second derivative

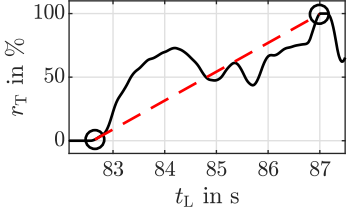
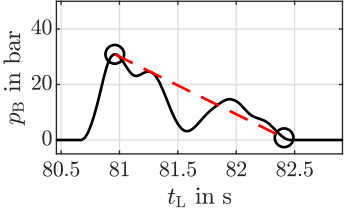
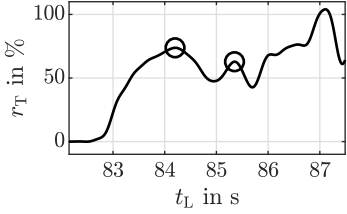
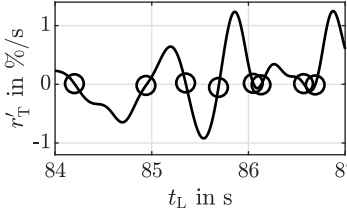
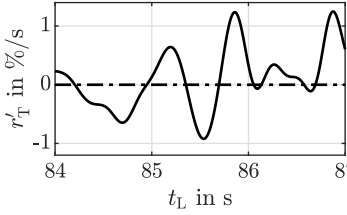
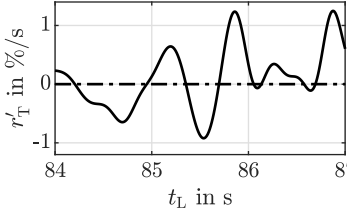
| Nr. | Metric Name   | Illustration  | Description  |
|-----|---------------|---|--|
| 33  | $rThrRmseLin$ |    | RMSE of $r_T$ respective to a linear approximation between <i>on-throttle</i> and <i>full-throttle</i> point |
| 34  | $pBrkRmseLin$ |    | RMSE of $p_B$ respective to a linear approximation between <i>off-brake</i> and <i>no-brake</i> point        |
| 35  | $NThrPeaks$   |   | Number of local maxima in $r_T$ during the <i>on-throttle</i> phase  |
| 36  | $NdThrZero$   |  | Number of zero crossings in $r'_T$ during the <i>on-throttle</i> phase                                       |
| 37  | $rdThrVar$    |  | Variance of $r'_T$ during the <i>on-throttle</i> phase   |
| 38  | $rdThrRmse$   |  | RMSE of $r'_T$ to zero during the <i>on-throttle</i> phase   |

Table 3.4: List of additional metrics on the stabilisation level

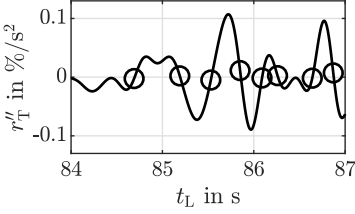
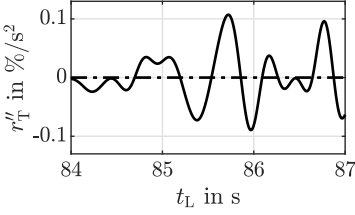
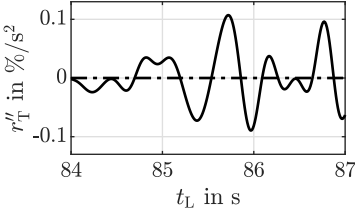
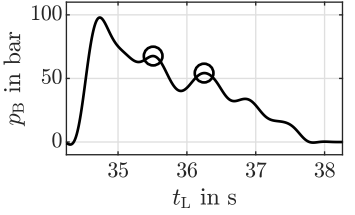
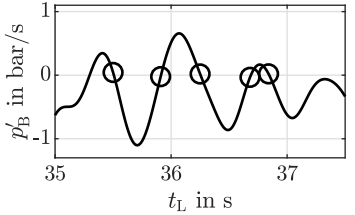
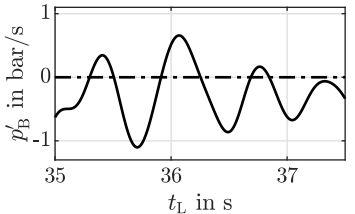
| Nr. | Metric Name       | Illustration  | Description   |
|-----|-------------------|---|---|
| 39  | <i>Nd2ThrZero</i> |    | Number of zero crossings in $r_T''$ during the <i>on-throttle</i> phase |
| 40  | <i>rd2ThrVar</i>  |    | Variance of $r_T''$ during the <i>on-throttle</i> phase                 |
| 41  | <i>rd2ThrRmse</i> |   | RMSE of $r_T''$ to zero during the <i>on-throttle</i> phase             |
| 42  | <i>NBrkPeaks</i>  |  | Number of local maxima in $p_B$ during the <i>off-brake</i> phase       |
| 43  | <i>NdBrkZero</i>  |  | Number of zero crossings in $p_B'$ during the <i>off-brake</i> phase    |
| 44  | <i>rdBrkVar</i>   |  | Variance of $p_B'$ during the <i>off-brake</i> phase                    |

Table 3.4: List of additional metrics on the stabilisation level

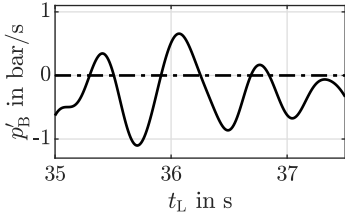
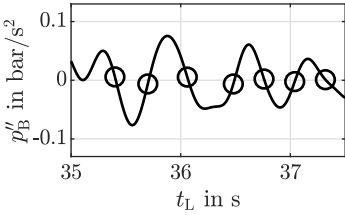
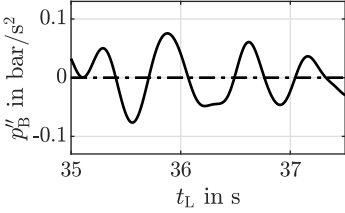
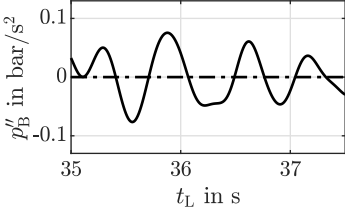
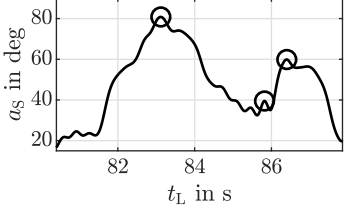
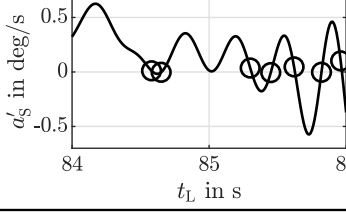
| Nr. | Metric Name  | Illustration  | Description   |
|-----|--------------|---|---|
| 45  | $rdBrkRmse$  |    | RMSE of $p'_B$ to zero during the <i>off-brake</i> phase              |
| 46  | $Nd2BrkZero$ |    | Number of zero crossings in $p''_B$ during the <i>off-brake</i> phase |
| 47  | $rd2BrkVar$  |   | Variance of $p''_B$ during the <i>off-brake</i> phase                 |
| 48  | $rd2BrkRmse$ |  | RMSE of $p''_B$ to zero during the <i>off-brake</i> phase             |
| 49  | $NStrPeaks$  |  | Number of local maxima in $a_S$ after the <i>turn-in</i> point        |
| 50  | $NdStrZero$  |  | Number of zero crossings in $a'_S$ after the <i>turn-in</i> point     |

Table 3.4: List of additional metrics on the stabilisation level

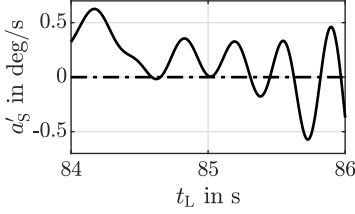
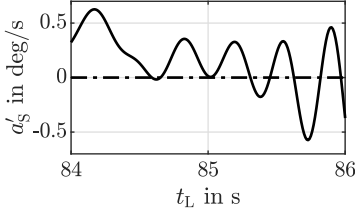
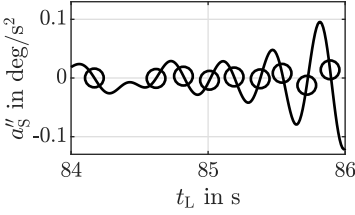
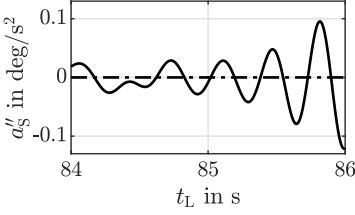
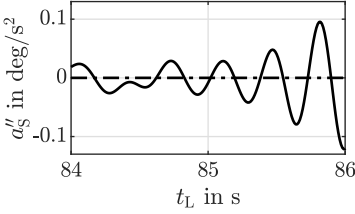
| Nr. | Metric Name       | Illustration  | Description  |
|-----|-------------------|---|--|
| 51  | <i>rdStrVar</i>   |    | Variance of $a'_S$ after the <i>turn-in</i> point                  |
| 52  | <i>rdStrRmse</i>  |    | RMSE of $a'_S$ to zero after the <i>turn-in</i> point              |
| 53  | <i>Nd2StrZero</i> |   | Number of zero crossings in $a''_S$ after the <i>turn-in</i> point |
| 54  | <i>rd2StrVar</i>  |  | Variance of $a''_S$ after the <i>turn-in</i> point                 |
| 55  | <i>rd2StrRmse</i> |  | RMSE of $a''_S$ to zero after the <i>turn-in</i> point             |

Table 3.4: List of additional metrics on the stabilisation level

Additional to the direct evaluation of time-series signals, the frequency spectrum of  $r_T$ ,  $p_B$  and  $a_S$  was assessed for any driver specific characteristics. Two approaches to describe the spectrum have been implemented to determine their significance in the following feature selection process. The signal needs to be converted from the time domain to the frequency domain in the first step. This is achieved by calculating the discrete Fourier transform using a Fast Fourier Transform (FFT) algorithm based on the `fftw` library published by Frigo and Johnson [15].

Figure 3.5 shows an exemplary  $r_T$  signal in the distance domain over  $s_L$  as well as in the frequency domain over  $f_T$ . The time-series data has been cut to the *on-throttle* phase before applying the FFT algorithm. One approach to characterise the shown spectrum is to find the first local maximum after the zero frequency component. This allows to get four metrics to describe this peak further. In addition to the magnitude and frequency of the maximum, a measure for the peak width and prominence can also be calculated. The peak prominence is indicated with the dotted line in the frequency spectrum graph in Fig. 3.5 and shows the height of the local maximum compared to the surrounding minima. The peak width is measured at half-prominence, as shown by the dashed line. The second approach to describe the frequency spectrum of the control input signals is to sample the magnitude at specific frequencies, thus generating a metric for each frequency. In accordance to the general method of generating metrics that are evaluated and selected in a second step, six metrics capturing the range from 0 Hz up to 2.5 Hz have been chosen based on the gradient of the frequency spectrum, alongside to the four metrics describing the first peak. All metrics derived from the aforementioned approaches using the  $r_T$ ,  $p_B$ , and  $a_S$  signals are listed in table 3.5.

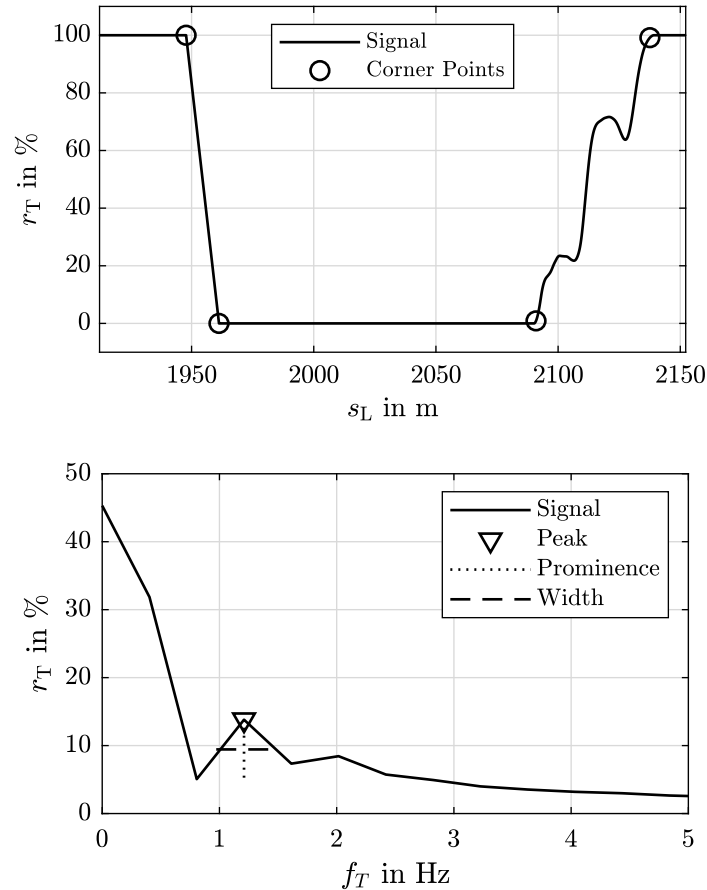


Figure 3.5: Exemplary throttle signal in time and frequency domain

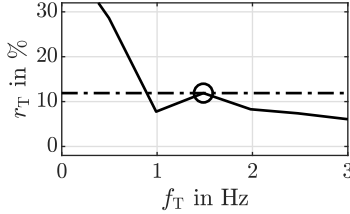
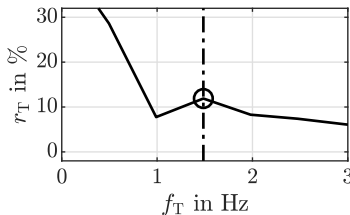
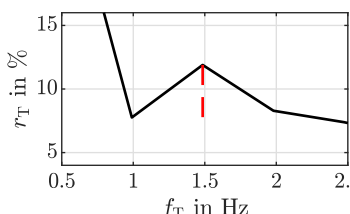
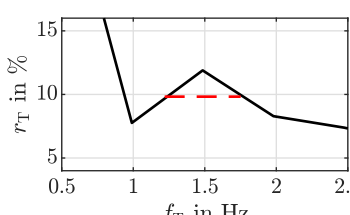
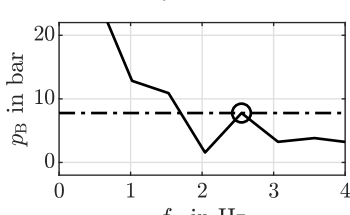
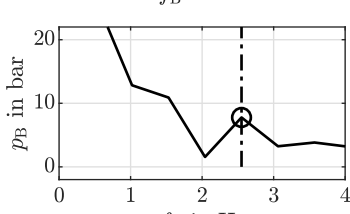
| Nr. | Metric Name        | Illustration  | Description  |
|-----|--------------------|---|--|
| 56  | $rThrFftPeak$      |    | Magnitude of the first peak in the frequency spectrum of $r_T$ during the <i>on-throttle</i> phase                                 |
| 57  | $fThrFftPeak$      |    | Frequency of the first peak in the frequency spectrum of $r_T$ during the <i>on-throttle</i> phase                                 |
| 58  | $rThrFftPeakProm$  |   | Magnitude of the first peak in the frequency spectrum of $r_T$ during the <i>on-throttle</i> phase relative to neighbouring minima |
| 59  | $rThrFftPeakWidth$ |  | Width of the first peak in the frequency spectrum of $r_T$ during the <i>on-throttle</i> phase at half-prominence                  |
| 60  | $pBrkFftPeak$      |  | Magnitude of the first peak in the frequency spectrum of $p_B$ during the <i>off-brake</i> phase                                   |
| 61  | $fBrkFftPeak$      |  | Frequency of the first peak in the frequency spectrum of $p_B$ during the <i>off-brake</i> phase                                   |

Table 3.5: List of frequency based objective metrics on the stabilisation level



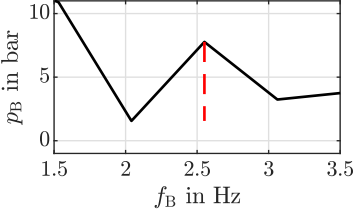
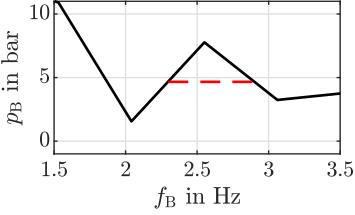
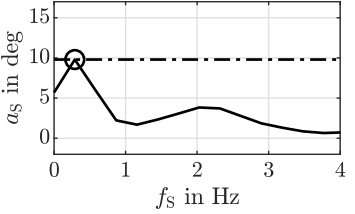
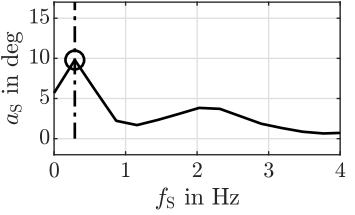
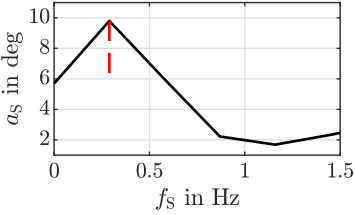
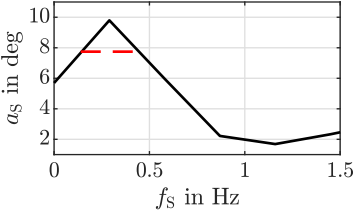
| Nr. | Metric Name        | Illustration  | Description  |
|-----|--------------------|---|--|
| 62  | $rBrkFftPeakProm$  |    | Magnitude of the first peak in the frequency spectrum of $p_B$ during the <i>off-brake</i> phase relative to neighbouring minima |
| 63  | $rBrkFftPeakWidth$ |    | Width of the first peak in the frequency spectrum of $p_B$ during the <i>off-brake</i> phase at half-prominence                  |
| 64  | $aStrFftPeak$      |   | Magnitude of the first peak in the frequency spectrum of $a_S$ after the <i>turn-in</i> point                                    |
| 65  | $fStrFftPeak$      |  | Frequency of the first peak in the frequency spectrum of $a_S$ after the <i>turn-in</i> point                                    |
| 66  | $rStrFftPeakProm$  |  | Magnitude of the first peak in the frequency spectrum of $a_S$ after the <i>turn-in</i> point relative to neighbouring minima    |
| 67  | $rStrFftPeakWidth$ |  | Width of the first peak in the frequency spectrum of $a_S$ after the <i>turn-in</i> point at half-prominence                     |

Table 3.5: List of frequency based objective metrics on the stabilisation level

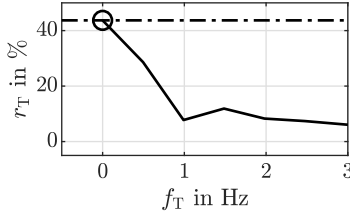
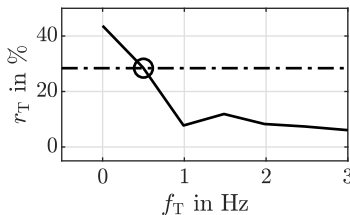
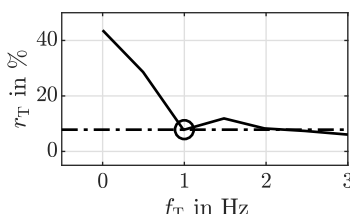
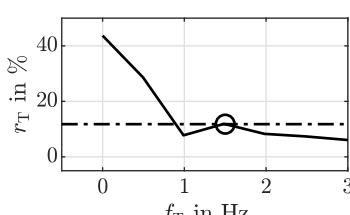
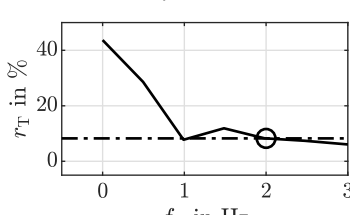
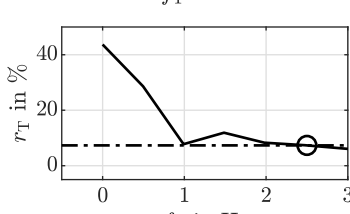
| Nr. | Metric Name | Illustration  | Description   |
|-----|-------------|---|---|
| 68  | $rThrFft00$ |    | Magnitude of $r_T$ frequency spectrum during the <i>on-throttle</i> phase at 0 Hz   |
| 69  | $rThrFft05$ |    | Magnitude of $r_T$ frequency spectrum during the <i>on-throttle</i> phase at 0.5 Hz |
| 70  | $rThrFft10$ |   | Magnitude of $r_T$ frequency spectrum during the <i>on-throttle</i> phase at 1.0 Hz |
| 71  | $rThrFft15$ |  | Magnitude of $r_T$ frequency spectrum during the <i>on-throttle</i> phase at 1.5 Hz |
| 72  | $rThrFft20$ |  | Magnitude of $r_T$ frequency spectrum during the <i>on-throttle</i> phase at 2.0 Hz |
| 73  | $rThrFft25$ |  | Magnitude of $r_T$ frequency spectrum during the <i>on-throttle</i> phase at 2.5 Hz |

Table 3.5: List of frequency based objective metrics on the stabilisation level

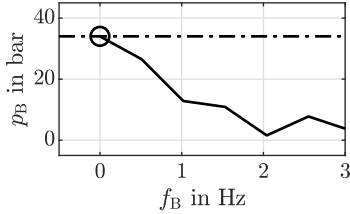
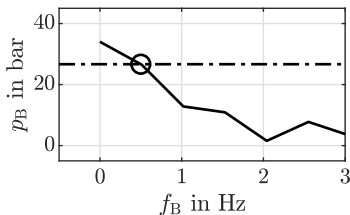
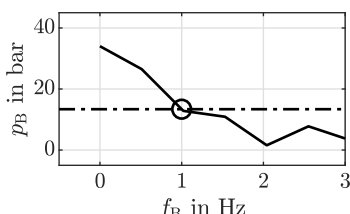
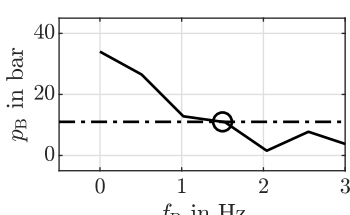
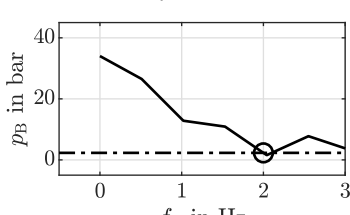
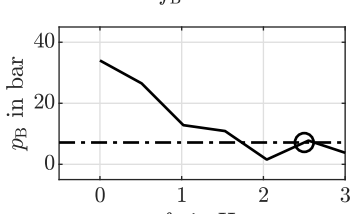
| Nr. | Metric Name | Illustration  | Description   |
|-----|-------------|---|---|
| 74  | $pBrkFft00$ |    | Magnitude of the $p_B$ frequency spectrum during the <i>off-brake</i> phase at 0 Hz   |
| 75  | $pBrkFft05$ |    | Magnitude of the $p_B$ frequency spectrum during the <i>off-brake</i> phase at 0.5 Hz |
| 76  | $pBrkFft10$ |   | Magnitude of the $p_B$ frequency spectrum during the <i>off-brake</i> phase at 1.0 Hz |
| 77  | $pBrkFft15$ |  | Magnitude of the $p_B$ frequency spectrum during the <i>off-brake</i> phase at 1.5 Hz |
| 78  | $pBrkFft20$ |  | Magnitude of the $p_B$ frequency spectrum during the <i>off-brake</i> phase at 2.0 Hz |
| 79  | $pBrkFft25$ |  | Magnitude of the $p_B$ frequency spectrum during the <i>off-brake</i> phase at 2.5 Hz |

Table 3.5: List of frequency based objective metrics on the stabilisation level

| Nr. | Metric Name | Illustration | Description  |
|-----|-------------|--------------|--|
| 80  | $aStrFft00$ |              | Magnitude of the $a_S$ frequency spectrum after the <i>turn-in</i> point at 0 Hz   |
| 81  | $aStrFft05$ |              | Magnitude of the $a_S$ frequency spectrum after the <i>turn-in</i> point at 0.5 Hz |
| 82  | $aStrFft10$ |              | Magnitude of the $a_S$ frequency spectrum after the <i>turn-in</i> point at 1.0 Hz |
| 83  | $aStrFft15$ |              | Magnitude of the $a_S$ frequency spectrum after the <i>turn-in</i> point at 1.5 Hz |
| 84  | $aStrFft20$ |              | Magnitude of the $a_S$ frequency spectrum after the <i>turn-in</i> point at 2.0 Hz |
| 85  | $aStrFft25$ |              | Magnitude of the $a_S$ frequency spectrum after the <i>turn-in</i> point at 2.5 Hz |

Table 3.5: List of frequency based objective metrics on the stabilisation level

## 3.2 Guidance Level

According to the three-level-model of the driving task, the previously discussed objective metrics describe the *driving style* on the stabilisation level where the driver controls the car to stay on a desired trajectory. This section covers objective metrics to describe the guidance level of the driving task where the driver comes up with the desired trajectory to navigate the track as quickly as possible. While a trajectory can be seen as a combination of velocity and direction of the vehicle movement, the latter plays an important role in the form of the driving line through a corner.

### 3.2.1 Curvature of the Vehicle Trajectory

As shortly mentioned in the introduction to this section, the driving line plays a major role in analysing the *driving style* on the guidance level. A good description of the driving line is possible using the coordinates of the vehicle in its surrounding space,  $x_c$  and  $y_c$ , giving the position of the car on a surface. Since the height of the car is depending on track position due to a given altitude profile and is not a degree of freedom under the driver's influence, the  $z_c$  coordinate is neglected. Despite the accurate information of the car's position these variables can be improved to describe the approach a driver is using to negotiate a turn. The instantaneous radius of the driving line  $r$ , or its inverse, the instantaneous curvature of the path  $\kappa$ ,  $\kappa = 1/r$ , allows the reduction of the information of  $x_c$  and  $y_c$  into one variable. Figure 3.6 shows two line segments  $\mathbf{p}_{12}$  and  $\mathbf{p}_{23}$ , which serve as examples for discrete driving line data. The calculation of both  $r$  and  $\kappa$  require approximation by a circle, while the discrete samples and consequently line segments are an approximation to the actual driving line itself. The circumscribing circle of the three points  $P_1$ ,  $P_2$  and  $P_3$  is used to approximate the two linear segments as it contains the discrete points. Contrary the inscribed circle would not contain the points, therefore the circumscribed circle approximates closer to the original continuous driving line. In this example the curvature is calculated at  $P_2$ . The derived circle's curvature  $\kappa$  is defined as shown in (3.11). In other words, the curvature is the change of inclination, or heading angle, for the vehicle reference, over the path length to do so. The approximation of  $s$  by  $\overline{P_1P_2}$  and  $\overline{P_2P_3}$  now allows calculation of  $\kappa$  as shown in (3.12), while  $\varphi$  can be calculated directly from the angle between  $\mathbf{p}_{12}$  and  $\mathbf{p}_{23}$  using the commonly known equation shown in (3.13). The relation of the angles  $\varphi$  between  $\mathbf{p}_{01}$  and  $\mathbf{p}_{03}$  and in this case  $\tilde{\varphi}$  between  $\mathbf{p}_{12}$  and  $\mathbf{p}_{23}$  can be proven using the angle  $\alpha$  and the isosceles triangle spanned by  $P_0$ ,  $P_2$  and  $P_3$ . As shown in (3.14),  $\frac{\varphi}{2}$  is defined by the sum of angles in a triangle, while  $\frac{\tilde{\varphi}}{2}$  is the angle between  $\mathbf{p}_{23}$  and the tangent of the circumscribing circle in  $P_2$ .

$$\kappa := \lim_{\Delta s \rightarrow 0} \frac{\Delta \varphi}{\Delta s} = \frac{d\varphi}{ds} \quad (3.11)$$

$$\kappa \sim \frac{\varphi}{\|\mathbf{p}_{12}\| + \|\mathbf{p}_{23}\|} \quad (3.12)$$

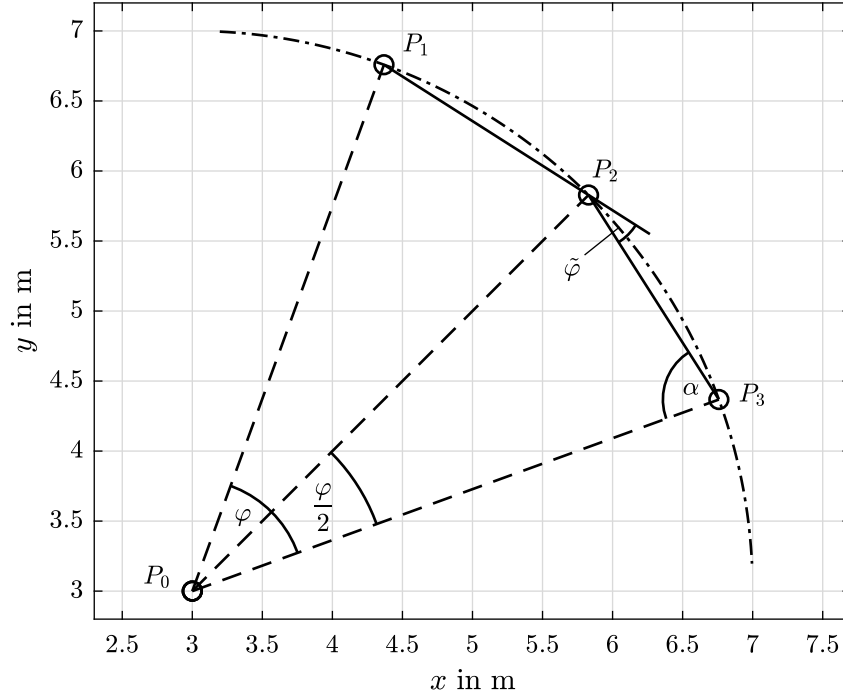


Figure 3.6: Exemplary circle for curvature calculation

$$\frac{\varphi}{2} = \arccos \frac{\mathbf{p}_{12} \cdot \mathbf{p}_{23}}{\|\mathbf{p}_{12}\| \|\mathbf{p}_{23}\|} \quad (3.13)$$

$$\begin{aligned} \frac{\varphi}{2} &= \pi - 2\alpha \\ \frac{\tilde{\varphi}}{2} &= \frac{\pi}{2} - \alpha \\ \tilde{\varphi} &= \frac{\varphi}{2} \end{aligned} \quad (3.14)$$

An important difference between  $r$  and  $\kappa$  is the behaviour for driving lines that approach straight lines. In this case  $\varphi$  will approximate to zero, resulting in the two line segments considered before and after a point becoming collinear. The radius will lead to computational problems in this extreme case of collinear segments, due to  $\lim_{\varphi \rightarrow 0} r = \infty$ . The curvature in contrary follows  $\lim_{\varphi \rightarrow 0} \kappa = 0$ , leading to a more stable solution in the context of running calculations automatically over various data sets and is therefore the preferred metric in this work.

### 3.2.2 Reference Trajectory

If data of the vehicle's Center of Gravity (CoG) coordinates are available, differences in driving line can easily be visualised as illustrated in Fig. 3.7. The driving lines of three drivers are shown, with a clear difference in their respective approach to the corner. The specific turn is number six in Hockenheim, the hairpin after the "Parabolica" which is a very long corner. This is a right-hand turn, thus the braking zone is in the lower part of the shown track segment. It is clearly visible that driver one, shown by the line with circle markers, turns in earlier compared to the other two drivers. Driver three, marked with triangles, on the other hand keeps the car on a straight line longer with a later *turn-in*. Regarding the corner exit, driver one is on a wider line which in turn means drivers two and three exit the corner on a more straight trajectory. While these differences are clearly visible, it is not trivial to describe the aforementioned characteristics without a second driving line for reference. This however would imply only relative measures between different driving lines, which is why an absolute reference is necessary to ensure comparability between data that has been gathered and analysed on different occasions. This leaves three possible approaches:

1. A reference racing line
2. The track centre line
3. The track outlines

A racing line, either artificially created or from an exemplary dataset, is always biased. This could result from a specific *driving style* if it is sourced from recorded car data, a specific car setup when defined by lap simulation, or by personal opinion if drawn by an engineer. From this point, centre and outlines seem to be better alternatives as they are based on the available track layout and can be measured accurately. Furthermore, these lines are the same for each driver and car, given a specific track layout.

A considerable fact about absolute coordinates is that they need to be in the same coordinate system to be comparable. Within the scope of this work this fact plays a role for the aforementioned absolute references, as they are not supplied in the same coordinate space where the vehicle data is recorded in. In this specific case, the centre line is not available in the vehicle coordinate system, but track outlines are stored in both the vehicle and the alternative coordinate space. However, the track outline in the vehicle space is of minor quality compared to the alternative source, for example because of sharp edges where multiple roads meet to allow different track layouts. An example is shown in Fig. 3.8. For the reason of sufficient data being available within the vehicle coordinate space to match it with any other sources, such as drawings<sup>3</sup>, an optimisation approach was chosen to derive the transformation rules between the coordinate systems. Generally such a transformation consists of a rotation of coordinates by a rotational matrix  ${}^1\mathbf{T}_2$  and a positional offset vector  $\mathbf{o}_{21}$  following (3.15). The optimisation problem

<sup>3</sup>Usually from Computer Aided Design (CAD) systems

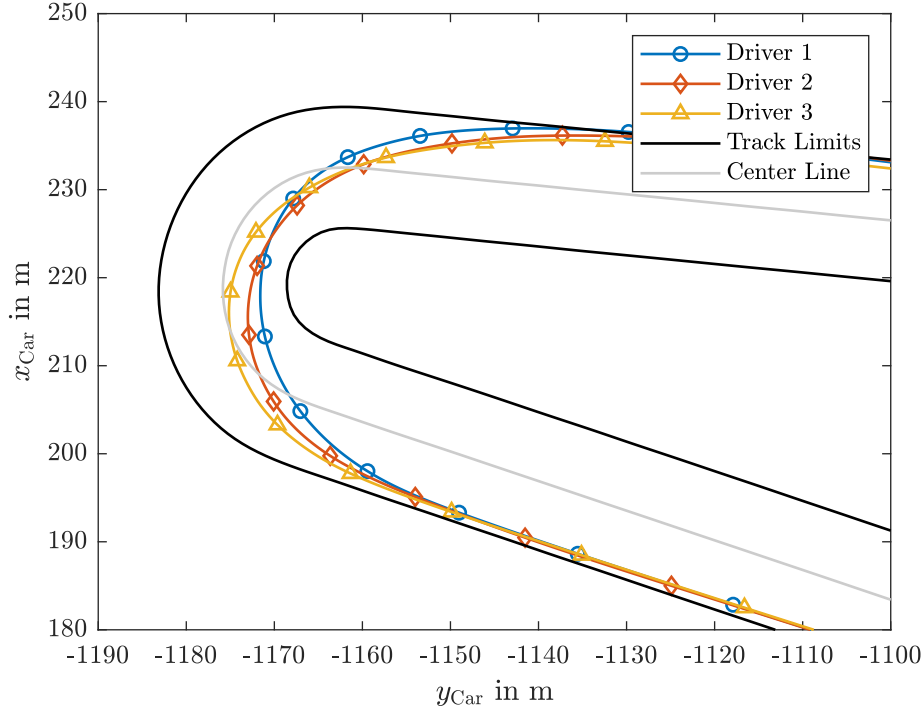


Figure 3.7: Exemplary driving lines and track definition

is formulated to minimise a cost function that measures the distance between one path and the transformed second path. The distance metric needed for this approach is discussed in more detail in the following paragraph. Its output is a vector of distances for each point of the transformed path to its nearest point on the reference path. The cost function then calculates the Root Mean Square (RMS) of this distance vector, which is essentially the RMSE of the transformed path to the reference path. The Inner-Points algorithm [5] is used in order to minimise the defined cost function. The result is a set of  $\mathbf{T}$  and  $\mathbf{o}$  to transform the alternative data source into the reference sources coordinate system.

$$\mathbf{x}_1 = {}^1\mathbf{T}_2\mathbf{x}_2 + \mathbf{o}_{21} \quad (3.15)$$

A more detailed explanation of the metric used to measure the distance between driving lines and track paths is provided in this paragraph. The function is iterated over each discrete point of the input path, which is then compared to the reference path. The shortest distance between those two paths is a perpendicular line from the analysed point to the closest section of the reference path. First, the euclidean distance of the current point to all points on the reference path is calculated to find the nearest point on the reference. Together with its preceding and succeeding point two lines are defined. The closest distance between the analysed point on the input path and the reference path



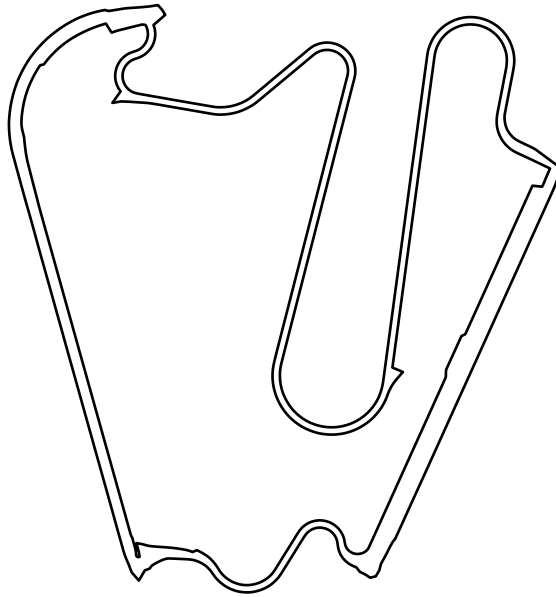


Figure 3.8: Track outlines for the Lausitzring in DTM configuration

must be an element of those two lines. Figure 3.9 shows an exemplary path illustrated by a solid line and a reference path shown by a dashed line. The selected point  $P_0$  and its closest point  $P_2$  on the reference path are marked by circles. The normals of both  $\mathbf{p}_{12}$  and  $\mathbf{p}_{23}$  through  $P_2$  are drawn with dotted lines. It is now evident that  $P_0$  can be located on either side of both normals or in between them which leaves three choices to calculate the distance between  $P_0$  and the reference path:

1. Orthogonal distance between  $P_0$  and  $\mathbf{p}_{12}$
2. Orthogonal distance between  $P_0$  and  $\mathbf{p}_{23}$
3. Distance between  $P_0$  and  $P_2$ ,  $\overline{P_0P_2}$

The third option is introduced to prevent measuring perpendicular from  $P_0$  to an extrapolation of either  $\mathbf{p}_{12}$  or  $\mathbf{p}_{23}$ , which could be of smaller magnitude compared to  $\overline{P_0P_2}$  and distort the measure. The decision between those options is made by using the angle  $\alpha_1$  between  $\mathbf{p}_{20}$  and  $\mathbf{p}_{21}$  or  $\alpha_2$  between  $\mathbf{p}_{20}$  and  $\mathbf{p}_{23}$  respectively. The conditions for each aforementioned option are in the same order:

1.  $\alpha_1 \leq \frac{\pi}{2} \cap \alpha_2 > \frac{\pi}{2}$
2.  $\alpha_1 > \frac{\pi}{2} \cap \alpha_2 \leq \frac{\pi}{2}$
3.  $\alpha_1 > \frac{\pi}{2} \cap \alpha_2 > \frac{\pi}{2}$

Finally, the distance of  $P_0$  to  $P_S$ , indicated by a triangle marker in Fig. 3.9, can be calculated by exploiting the orthogonality of  $\mathbf{p}_{0S}$  and  $\mathbf{p}_{23}$  using their trigonometric relation as shown in (3.16). In case of a direct measurement from  $P_0$  to  $P_2$  the distance is calculated as the norm of the vector between them.

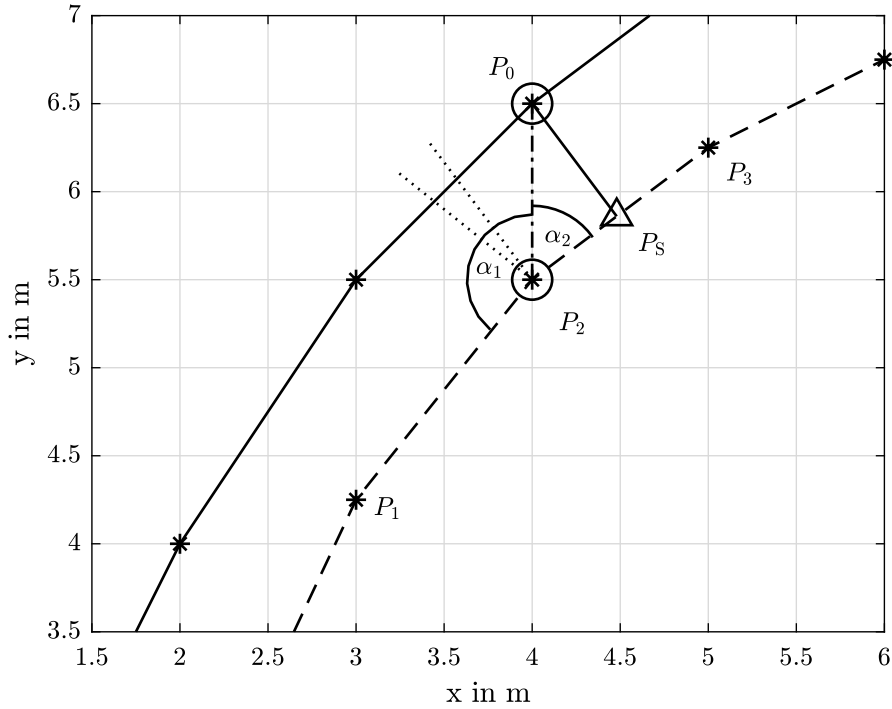


Figure 3.9: Exemplary paths and distance metric illustration

$$\overline{P_0 P_S} = \overline{P_0 P_2} * \sin \alpha_2 \quad (3.16)$$

In addition to the absolute distance value, the metric is also directional to allow distinguishing on which side of the reference path a point is located. It is defined that the sequence of points indicates the direction of a path and a point that is left of the reference path with respect to its direction has a positive distance measure. Consequently, a point on the right of the reference path has a negative distance measure. The sign of the distance metric is obtained by using the cross product of  $\mathbf{p}_{20}$  and  $\mathbf{p}_{21}$  or  $\mathbf{p}_{23}$  respectively. All coordinates are extended by a z-component equal to zero. The cross product of those vectors will then be directed along the z-axis and change sign depending on the position of the vectors to each other. The sign of the z-component of the cross product is used to define the sign of the distance measure.

### 3.2.3 Objective Criteria

The approach to define objective criteria for the *driving style* on the guidance level follows the previously outlined method for the stabilisation level, as discussed in subsection 3.1.2. The mainly used signals are the already introduced curvature  $\kappa_P$  and distances to the reference lines as a representation of the essential information contained in the coordinates of the vehicle  $x_c$  and  $y_c$ .

The general idea is to capture the different approaches to a corner, as exemplarily shown in Fig. 3.7. One approach is to analyse the point with the maximum curvature. Because the curvature is signed for left- and right-hand corners, this could either be a minimum or a maximum. Common to the previous definition of metrics the value itself, as well as the time and distance of the instance both in absolute means from the *off-throttle* point and relative to the corner are calculated. The magnitude of the extreme point is simplified to its absolute value allowing only positive measures. All metrics defined for *driving style* on the guidance level are noted in table 3.6.

As already introduced, also the track limits are considered as an absolute reference. The point where the vehicle path reaches the minimum distance to the inner track edge  $d_E$  is commonly referred to as the apex in motor racing. The derived metrics follow the same approach of absolute value and relative position in time and lap distance domain once again. The distance between the driving line and the inner track limit is calculated using the previously explained distance metric and makes use of the introduced sign to allow detecting a driver cutting the corner which will lead to a change of sign in the result. The magnitude of the distance at the apex would have different signs for left- and right-hand corners with this approach. Since a change of sign is wanted to indicate crossing the reference line independently of the corner, a different approach is used. Instead of analysing the absolute value, the sign of the distance measure throughout the corner is manipulated by the sign of its value at the *off-throttle* point. This leads to a positive measure that will turn negative if the reference line is crossed. The exact same approach is used for another set of metrics using the distance to the track centre line  $d_C$  instead. The main difference is that the distance to the centre line usually features two zero crossings, except for very few corners where drivers tend to stay on one side of the centre line.

Exemplary driving lines are shown together with the track limits and track centre line in Fig. 3.7, as discussed in section 3.2.2. The elaborated approach to a corner using the example in Fig. 3.7 is commonly referred to as "V-ing" a corner which is due to the shape of the car velocity and driving path. Longer straight line braking with a more sharp turn of the car and the consequently straighter trajectory on corner exit, as well with usually a late apex will result in a clearer edge visible in time-series data, which will in turn lead to the impression of the letter "V". This is contrary to a more rounded approach to a corner, essentially employing more *trail braking* that usually results in smoother time-series signals. Three metrics are defined to capture this aspect to examine which is suited best as an objective metric for the *driving style*. All three metrics measure an angle, the "V-angle", to describe the behaviour of the signals. The analysed signals are the curvature of the path, the distance to the inner track limits and the distance to the track centre line. The "V-angle" is then calculated as the angle between the vectors from the respective signal's minimum to the *turn-in* point as well as to the *full-throttle* point. Exemplary signals are shown in Fig. 3.10 to illustrate the metrics. The graphs show curvature, distance to inner track limit and distance to track centre line from top to bottom, with the *turn-in* point marked by a diamond, the minimum of the signal marked by a triangle and the *full-throttle* point marked by a circle.

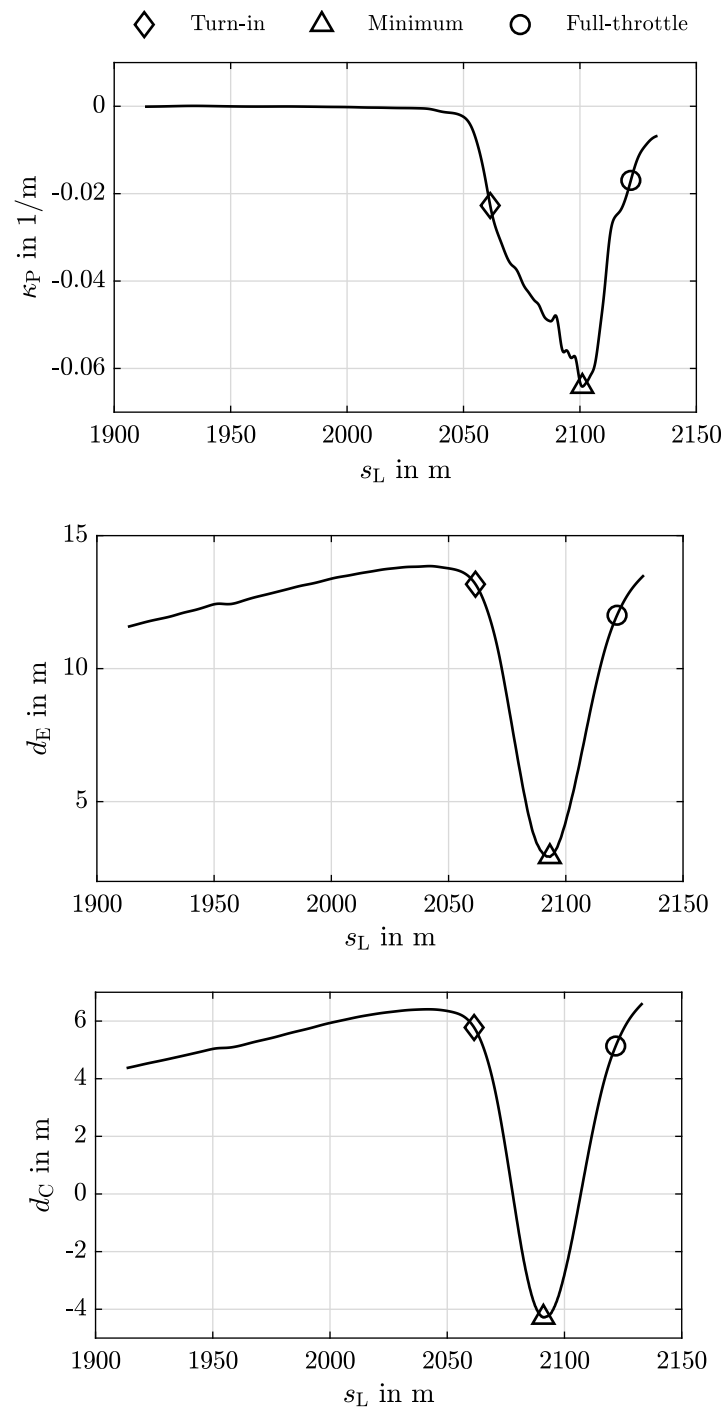


Figure 3.10: Exemplary signals for curvature, distance to inner track limit, and track centre line with points for "V-angle" calculation

| Nr. | Metric Name | Illustration | Description  |
|-----|-------------|--------------|--|
| 86  | $sCurvAbs$  |              | Absolute distance covered between <i>off-throttle</i> point and the maximum of $\kappa_P$  |
| 87  | $sCurvRel$  |              | Distance covered between <i>off-throttle</i> point and the maximum of $\kappa_P$ relative to the distance between the <i>off-throttle</i> and <i>full-throttle</i> point |
| 88  | $tCurvAbs$  |              | Absolute time passed between <i>off-throttle</i> point and the maximum of $\kappa_P$   |
| 89  | $tCurvRel$  |              | Time passed between <i>off-throttle</i> point and the maximum of $\kappa_P$ relative to the time between the <i>off-throttle</i> and <i>full-throttle</i> point          |
| 90  | $sApexAbs$  |              | Absolute distance covered between <i>off-throttle</i> point and the apex of the corner   |
| 91  | $sApexRel$  |              | Distance covered between <i>off-throttle</i> point and corner apex relative to the distance between the <i>off-throttle</i> and <i>full-throttle</i> point               |

Table 3.6: List of objective metrics for the guidance level

| Nr. | Metric Name   | Illustration | Description   |
|-----|---------------|--------------|---|
| 92  | $t_{ApexAbs}$ |              | Absolute time passed between <i>off-throttle</i> point and the apex of the corner   |
| 93  | $t_{ApexRel}$ |              | Time passed between the <i>off-throttle</i> point and the corner apex relative to the time between the <i>off-throttle</i> and <i>full-throttle</i> point   |
| 94  | $s_{CenAbs}$  |              | Absolute distance covered between <i>off-throttle</i> point and the minimum distance measured to the centre line of the corner  |
| 95  | $s_{CenRel}$  |              | Distance covered between <i>off-throttle</i> point and the minimum distance measured to the centre line of the corner relative to the distance between the <i>off-throttle</i> and <i>full-throttle</i> point |
| 96  | $t_{CenAbs}$  |              | Absolute time passed between <i>off-throttle</i> point and the minimum distance measured to the centre line of the corner   |

Table 3.6: List of objective metrics for the guidance level

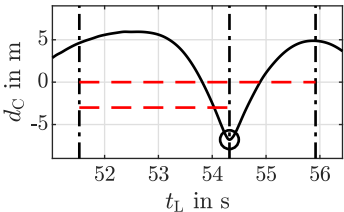
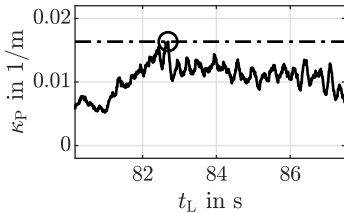
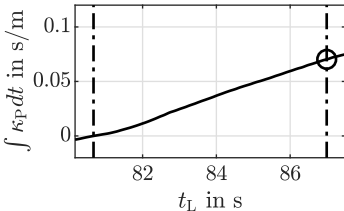
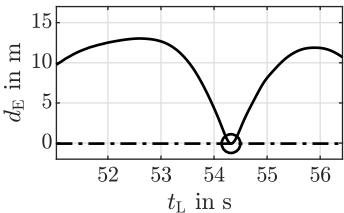
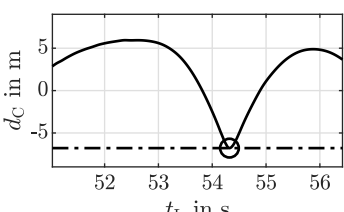
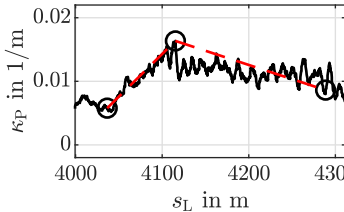
| Nr. | Metric Name | Illustration  | Description  |
|-----|-------------|---|--|
| 97  | $tCenRel$   |    | Time passed between the <i>off-throttle</i> point and the minimum distance measured to the centre line of the corner relative to the time between the <i>off-throttle</i> and <i>full-throttle</i> point |
| 98  | $cMax$      |    | Maximum of the absolute value of $\kappa_P$  |
| 99  | $cInt$      |   | Integral of $\kappa_P$ between <i>off-throttle</i> and <i>full-throttle</i> point  |
| 100 | $xApex$     |  | Minimum distance measured from the driving path to the inner track limits between the <i>off-throttle</i> and <i>full-throttle</i> point   |
| 101 | $xCen$      |  | Minimum distance measured from the driving path and the centre line of the track between the <i>off-throttle</i> and <i>full-throttle</i> point  |
| 102 | $aVCurv$    |  | Angle between the slopes of $\kappa_P$ from <i>turn-in</i> point to the maximum of $\kappa_P$ and back to the <i>full-throttle</i> point   |

Table 3.6: List of objective metrics for the guidance level

| Nr. | Metric Name | Illustration | Description  |
|-----|-------------|--------------|--|
| 103 | $aV_{Apex}$ |              | Angle between the slopes of the distance measure to the inner track limits from <i>turn-in</i> point to its minimum and back to the <i>full-throttle</i> point |
| 104 | $aV_{Cen}$  |              | Angle between the slopes of the distance measure to the track centre line from <i>turn-in</i> point to its minimum and back to the <i>full-throttle</i> point  |

Table 3.6: List of objective metrics for the guidance level



### 3.3 Metric Evaluation

The previous sections described in depth the calculation of objective criteria, or features as referred to in the machine learning domain. The general method applied throughout this work leads to a large set of available features to describe the driver. Those are likely of varying significance and are assumed to contain redundancies as well. Therefore, a classification problem is formulated, in which the driver should be identified from a given observation, meaning all available features for one detected corner. Feature selection methods are presented and applied to the dataset to reduce the amount of metrics to a number that can be interpreted yet still contains the wanted information about the *driving style*, represented as the information needed to identify the driver in this formulation. This classification problem additionally provides a proof of concept, as the identification of a driver from the available metrics implies that enough information is contained within those to differentiate individual drivers.

Guyon et al. present a broad overview of feature selection methods. These can be split into three general categories: filter methods, wrapper methods, and embedded methods. Filter methods are a preprocessing step that is independent of the chosen predictor for the classification problem. Feature ranking falls into this category, with the goal to compute a score for each feature that indicates its value to the predictor. For a dataset of  $m$  observations  $\mathbf{x}_k, y_k$  ( $k = 1..m$ ) consisting of  $n$  features  $x_{k,i}$  ( $i = 1..n$ ) and one output target  $y_k$ , a scoring function  $S(i)$  is computed from the values  $x_{k,i}$  and  $y_k$ ,  $k = 1..m$ . The method relies on the assumption, that a high score indicates a useful feature which allows sorting all features in decreasing order of  $S(i)$ . Feature ranking is computationally efficient, because for the given example dataset only  $n$  scores have to be computed and sorted. [20]

Contrary to filter methods, wrapper methods rely on the chosen predictor to find an optimal subset of features. The wrapper methodology evaluates the quality of predictions using different subsets of features and the chosen predictor. This generally needs three prerequisites, namely the choice of the predictor to solve the classification problem, the metric used to assess the prediction performance, and a method to navigate through the space of all possible feature subset combinations. Consequently, the exploration has to be defined carefully to avoid unnecessary computational effort or even running into a NP-hard<sup>4</sup> problem [1]. The main difference between filter and wrapper methods is that the latter incorporates the prediction method while filter methods are independent from it. [20]

An example for a filter method is using correlation criteria to rank features according to their correlation with the target, or to detect correlated features that might be redundant. Unfortunately, the calculation of correlations with the target is not possible for classification problems, since the target is non-ordinal. A common measure for correlation is the *Pearson* coefficient, which gives a measure of linear relationship between two

<sup>4</sup>A measure used for assessment of calculation complexity, NP means non-deterministic polynomial time.

variables. The *Pearson* product-moment coefficient  $\rho$  is defined as shown in (3.17) [17]. It is generally computed using the mean values  $\bar{x}_i, \bar{x}_j$  of  $x_i$  and  $x_j$ , in form of the sample correlation coefficient  $\rho$  according to (3.18). A correlation matrix  $\mathbf{R}$  can be calculated where  $\mathbf{R}_{i,j} = \rho(x_i, x_j)$ . The diagonal of this matrix is 1 as it represents the linear correlation of each feature with itself. The other entries show a measure for correlation of each feature with each other feature. This allows to inspect individual correlations as well as calculating a mean correlation value  $\bar{\rho}_i$  for all metrics. While correlations among features and generally highly correlated features can be identified with this method, it does not allow assumptions about which of two correlated features is better suited to solve the classification problem.

$$\rho_{x_i, x_j} = \frac{\text{cov}(x_i, x_j)}{\sqrt{\text{var}(x_i)\text{var}(x_j)}} \quad (3.17)$$

$$\rho_{x_i, x_j} = \frac{\sum_{k=1}^m (x_{k,i} - \bar{x}_i)(x_{k,j} - \bar{x}_j)}{\sqrt{\sum_{k=1}^m (x_{k,i} - \bar{x}_i)^2 \sum_{k=1}^m (x_{k,j} - \bar{x}_j)^2}} \quad (3.18)$$

Another filter method is the *Relief* algorithm first proposed by Kira et al. *Relief* basically uses the *Euclidean Distance* to weigh features, but overcomes the general limitations of distance metrics for high-dimensional spaces by making use of the target class, which is available for supervised learning approaches. The original algorithm is limited to two-class problems. It picks samples as triplets of an observation  $\mathbf{x}_k$ , its near-hit observation and its near-miss observation. Near-hit describes observations, which are close to  $\mathbf{x}_k$  in terms of the n-dimensional *Euclidean Distance* and in the same class as  $\mathbf{x}_k$ . Contrary near-miss means, that an observation is close to  $\mathbf{x}_k$  but with a different class compared to  $\mathbf{x}_k$ . A difference measure  $\text{diff}(x_i, y_i)$  is used to calculate feature weights, where  $x_i$  and  $y_i$  are the values of the same feature for two samples. For numerical features  $\text{diff}(x_i, y_i)$  is calculated according to (3.19). The difference of the features is normalised to the interval  $[0, 1]$  with the normalisation unit  $NU_i$ . Feature weights  $W_i$  are then updated iteratively for each sample according to (3.20), using the triplet of sample, near-hit and near-miss instance. Kira et al. tested the relevance of the feature weights assigned by *Relief* using known datasets, which have been manipulated to contain irrelevant features and noise. The feature weights assigned by *Relief* are in the interval  $[-1, 1]$ , where negative weights indicate irrelevant features and positive weights show relevance and allow ranking features as well. [24]

$$\text{diff}(x_i, y_i) = (x_i - y_i)/NU_i \quad (3.19)$$

$$W_i = W_i - \text{diff}(x_i, \text{near-miss}_i)^2 + \text{diff}(x_i, \text{near-hit}_i)^2 \quad (3.20)$$

The *Relief* algorithm has been evolved further by Kononenko [26], resulting in the variants *Relief-A* to *Relief-F*. The first evolution is *Relief-A*, which averages the contributions of more than one nearest-hit and nearest-miss per sample to increase the reliability of the probability approximation. *Relief-B* to *Relief-D* are variants seeking to solve the problem of missing values, where *Relief-D* performs significantly better compared to the other options. In this approach unknown values are handled by calculating the probability that two instances have different values instead of the difference measure. The conditional probabilities are approximated with relative frequencies from the training set. The inability of *Relief* to solve multi-class problems is addressed by *Relief-E* and *Relief-F*. Where *Relief-E* simply takes the nearest neighbour with a different class as a near-miss instance, *Relief-F* finds a near-miss for each other class and averages their contributions to the feature weights. *Relief-F* outperforms *Relief-E* and, incorporating the advantages of *Relief-A* and *Relief-D*, became the most commonly used *Relief* algorithm. *Relief* has been presented as being *non-myopic*, i.e. it estimates the quality of a given feature in the context of other features, and *non-parametric*, i.e. it makes no assumptions regarding the population distribution or sample size [60].

Wrapper methods, as briefly mentioned before, use a predictor together with a performance metric to assess feature subsets. Since it is computationally extensive to evaluate the whole feature subset space, two general approaches are used independently of the predictor. Forward selection starts with an empty feature set and increases the feature subset size on each iteration, trains the predictor and rates its performance. Controversially backward selection starts with the full feature set, decreasing the subset size on each iteration. Both approaches can also add or remove multiple features at once to reduce calculation time. While forward selection is generally computationally more efficient than backward selection, it tends to find weaker subsets as features are assessed individually rather than in context with other features. Backward selection on the other hand might eliminate a feature that is better suited to split data by itself in favour of other features that provide higher performance when combined [20]. An example of possible feature subsets for a dataset with four features is illustrated in Fig. 3.11. While a forward selecting algorithm would work through the combinations from left to right, the backward selection method works the other way round from right to left.

The metric used to evaluate prediction performance throughout this work is accuracy, which is defined as the probability of a correct classification for a random observation. Other metrics, for example precision and recall, place higher emphasis on false positive or false negative classification. Such metrics are useful for applications where the cost for misclassification is high. Coming back to the accuracy of a classifier, there are different ways to estimate this probability measure. Kohavi provides a study comparing different methods and gives a recommendation based on tests with various datasets. The *Holdout* method is described as a global split of the available data, which is often referred to as a training and a testing set. The purpose of the test set is to evaluate the prediction performance on observations that have not been used for training. Such a dataset with unseen data helps with detecting over fitting, as in a classifier that simply memorises the

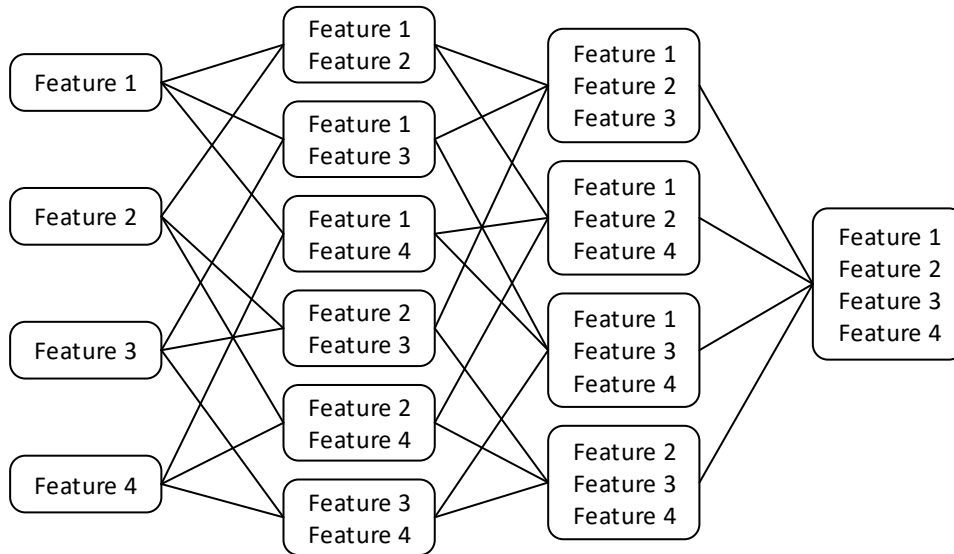


Figure 3.11: Exemplary feature subsets for four features

training set (a perfect learner). This approach significantly reduces the available training data, which is commonly around two-thirds of the available data. One approach to avoid this reduction of data is *k-Fold Cross-Validation*. This approach divides available data in  $k$  mutually exclusive folds. The classifier is trained  $k$  times with  $k - 1$  folds, using the remaining fold for testing. The accuracy is then averaged over the folds, which provides an additional measure of its variance. While complete *k-Fold Cross-Validation* would require to draw all  $\binom{m}{m/k}$  possible combinations of  $m/k$  samples out of  $m$  available observations, usually only one split into  $k$  folds is considered, approximating the complete *Cross-Validation*. A second approach contrary to the *k-Fold* method is *Bootstrap Sampling*. This means to draw  $m$  instances from a dataset of size  $m$  with replacement. Subsequently the sampled dataset contains the same number  $m$  of instances, however due to drawing with replacement some observations of the original dataset are never chosen. These form the test set. Similar to *k-Fold Cross-Validation*, this process is repeated to obtain multiple combinations of training and test data. Finally, *Stratifying* is a method to preserve the original class distribution within the test and training sets. Kohavi suggests that *Stratified 10-Fold Cross-Validation* is best for model selection regarding bias to specific classification problems and variance of the estimation [25].

The classifier used within the scope of this thesis is *Random Forests*, which is based on *Decision Trees*. These were introduced as Classification And Regression Trees (CART) by Breiman et al. Classification Trees are a machine learning method to predict the class of an observation from several features by splitting the data at nodes, which each use one of the available features to split data. The final nodes are called leafs and have

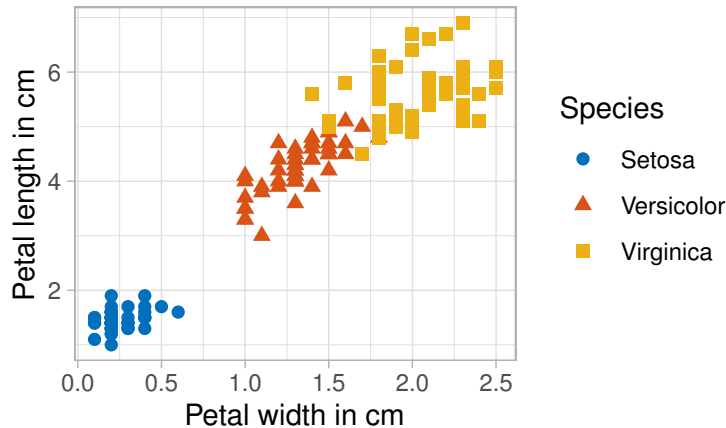
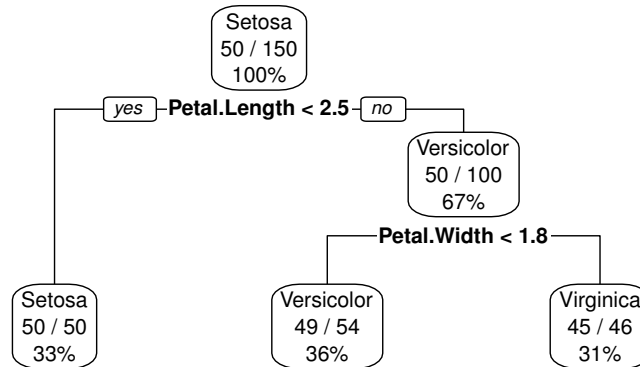


Figure 3.12: Data points of the iris dataset [14]

a class value assigned. For better illustration, Fisher’s iris dataset<sup>5</sup> [14] is used as an example. The dataset consists of 150 observations of iris plants and five features. The feature *species* contains labels for the three different plants and is chosen as the class. Features *petalength* and *petalwidth* allow a good separation of the available data, as demonstrated in Fig. 3.12, while *sepalength* and *sepalwidth* are of minor importance to the *Decision Tree*. Figure 3.13 shows an exemplary *Classification Tree* trained from the iris dataset. The top most node splits the data by the feature *petalength*, and all observations where *Petal.Length* < 2.5 is true are labelled Setosa according to the following leaf. The observations with *Petal.Length* >= 2.5 are split at a following node using the feature *petalwidth* on the condition *Petal.Width* < 1.8. The presented tree structure represents what is also visible to the viewer in Fig. 3.12, however this is only possible in such a simple example. *Decision Trees* allow using the same feature more than once, which caters for complex interdependencies between features. Furthermore, *Decision Trees* are robust against outliers. [3]

The *Random Forest* method used in this work is based on the discussed *Decision Trees*. It was proposed by Breiman and is an ensemble of *Decision Trees* that vote for the most popular class. The individual trees use independent, identically distributed random vectors together with the training set. Each vector contains information about the observations to use, and the features considered for each split. Breiman states that *Random Forests* are of similar or better accuracy compared to *Adaboost*, while being relatively robust against outliers and noise as well as faster than bagging or boosting algorithms. Furthermore, *Random Forests* give internal estimates of error, as well as feature importance and are easy to implement on parallel computing platforms. *Bagging* is used to enhance accuracy, but also to provide the Out-of-Bag (OOB) error as an estimate for the generalisation error, which is proven to be of similar value than a test set. The OOB error is calculated by testing on the observations left out in each *Bootstrap*

<sup>5</sup>Available from the UCI Machine Learning Repository [13]

Figure 3.13: *Classification Tree* for the iris dataset [14]

| Model               | Accuracy estimation | Error rate |
|---------------------|---------------------|------------|
| Classification tree | Test set            | 6.67 %     |
| Random forest       | OOB                 | 5.71 %     |
| Random forest       | Test set            | 4.44 %     |

Table 3.7: Accuracy estimations for *Classification Tree* and *Random Forest*

training set for the individual trees. Exploration of the forest is possible by permuting features to see the effect on the prediction accuracy. After each tree is constructed the values for a feature in the OOB observations are randomly permuted and the tree is tested with the OOB data. At the end of the run all class votes with the permuted feature are compared to the true class, which gives a misclassification rate for each feature. The output is a percentage increase in misclassification rate compared to the OOB rate with intact features. [4]

Using the iris dataset [14] again allows to illustrate a comparison of *Classification Trees* and *Random Forests*. A test set containing 30 % of the data is used to estimate the accuracy for both models. Table 3.7 reports the achieved accuracies. *Random Forest* is able to further improve the error rate of the *Decision Tree*. The feature importance for the *Random Forest*, in terms of decrease in accuracy using permutation, is shown in Fig. 3.14.

Another method used for feature selection in this work is Recursive Feature Elimination (RFE), a backward searching wrapper method. The general approach for each iteration, starting from the full feature set, is:

1. Train and rate the classifier
2. Compute the ranking criterion for all used features
3. Eliminate the feature with smallest ranking criterion

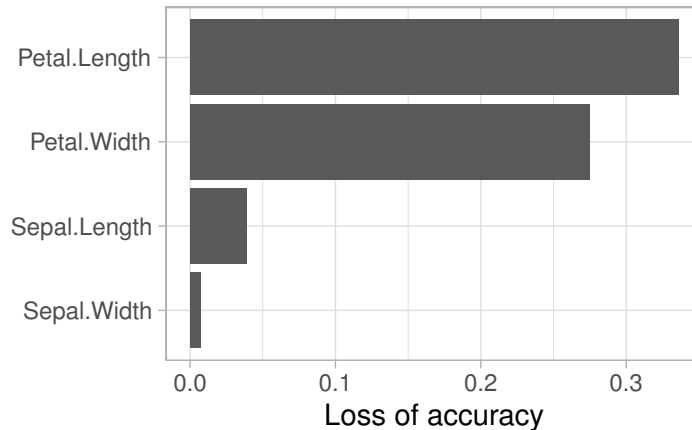


Figure 3.14: Feature importances for *Random Forest* trained on the iris dataset [14]

The main difference to a complete backwards search, as illustrated in Fig. 3.11, is that features evaluation is based on a ranking metric which is loss of accuracy in this work, for each feature subset. Subsequently, the least important feature(s) will be removed and not all possible combinations have to be evaluated. When only one feature is removed at each iteration, a corresponding feature ranking can be drawn from the classifier rating for decreasing subset size, where the rating results in the loss of accuracy by removing a specific feature. The top-ranked features are individually not necessarily the most relevant but provide the best performance in conjunction with other highly ranked features [21]. Because this method relies on a feature ranking that is of relevance to the chosen classifier, it is most commonly used with Support Vector Machines (SVM) and *Random Forests*. Both provide built in ranking metrics.

The following feature evaluation process is based on the methods presented in this section. Generally, the overall prediction accuracy is chosen as performance criteria for all classifications. It is used in combination with *k-Fold Cross Validation* following Kohavi's findings on classification accuracy estimation [25]. The feature selection process employs a combination of *Pearson* coefficients, the *Relief* algorithm, and *Random Forests's* built in loss of accuracy metric for specific subsets of features. This combination of algorithms is extended with RFE for rating all features at once, since it does only provide a ranking metric for features that were selected during the process. This means in turn that specific subsets of features cannot always be compared if some features are not selected by RFE and thus have no ranking value assigned. Finally, the *Random Forest* classifier is chosen for the aforementioned robustness [4], built in loss of accuracy metric, as well as good implementation for parallel computing, which saves calculation time.

## 4 Evaluation

The following chapter focuses on applying the defined methodology to different data sets and evaluating the results as well as the significance of the metrics. The available data including the respective sources and used software are discussed in the following section. Thereafter the feature selection methods, which were explained in detail in chapter 3.3, are applied and conclude in the presentation of results regarding the finally selected metrics for the given data sets.

### 4.1 Simulation Environment and Available Data

Two main data sources are used in this work. These are data logged on real race cars during race weekends, as well as logged data from the Driver-in-the-Loop Simulator (DiLS). According to the previous chapter a small selection of signals is needed to calculate the defined metrics. This corresponds with the aim of the method to allow application in as many racing environments as possible. The five needed signals are:

- $r_T$ , throttle pedal position
- $p_B$ , brake pressure
- $a_S$ , steering wheel angle
- $x_c, y_c$ , position of the car in a world coordinate system

The first three signals are usually available on any vehicle with a Data Acquisition (DAQ) system, since they represent the smallest set of sensor data to know how the driver is using the car. The latter two are more difficult to acquire with sufficient accuracy. One possibility is the Global Positioning System (GPS), a global satellite based navigation system, which is able to provide lateral and longitudinal coordinates on the earth's surface. These can be transformed to two-dimensional coordinates. However, GPS for common applications is limited in accuracy. Especially the environment reduces relative comparability of data obtained from different locations and in different atmospheric conditions. These errors are estimated to deteriorate the measurement by a magnitude of up to five meters [54]. As this value is greater than the expected differences in driving lines, the common GPS is not used for *driving style* analysis within the scope of this work. Alternative systems such as differential GPS are available and capable of providing the needed accuracy, but those systems are not suited to be fitted on race cars during normal use due to their cost, weight, and packaging requirements. Overall this leads to the choice



of the DiLS to gather data regarding the driving line because the simulation environment allows recording the vehicles coordinates. This simulation environment is explained in the following subsection and the available data from both sources is detailed afterwards.

#### 4.1.1 Simulation Environment

The used simulation environment consists of a transient vehicle model, which forms the DiLS together with a chassis mock-up and actuation hardware. The parametrisation is carried out by a database application that is storing the necessary information for all currently used vehicles and their respective setup options. The mentioned parts are all implemented into an overall simulation framework to control the interactions between different subsystems. As these are multiple complex topics by themselves, only a short description to outline the used simulation environment is supplied.

The used vehicle model is a transient 14+ Degree of Freedom (DoF) model with four input signals. The inputs to the model are  $r_T$ ,  $p_B$ ,  $a_S$  and finally the selected gear  $N_G$ . From these the DoF of the model are excited, which consist of three rotational as well as three translational DoF for the vehicle, four rotational DoF for the wheels and one additional DoF for each wheel. The latter can be interpreted as suspension movement and defines the remaining dependant translations and rotations of the wheels. For example, this leads to the wheel's rotation respective to the  $x$  and  $z$  axis, commonly referred to as camber and toe, and depends on the suspension kinematic and its current position. The engine is modelled to a sufficient degree by a combination of maps and can be de-clutched, which essentially adds one more DoF that however is of minor relevance as the clutch is usually engaged once the car is moving. Since the aerodynamic forces affect car performance and handling substantially, their representation in the model is crucial. Among other parameters, the aerodynamic coefficients are derived from the front and rear ride height, which are a measure for ground clearance under the front or rear axle, thus defining the frontal aerodynamic angle of attack as well as the ground clearance of the underbody. One of the most important parts of the vehicle model is the representation of the tyre. Within the scope of this work a tyre model based on Pacejka's Magic Formula is used [41][42]. This approach allows a detailed simulation of longitudinal, lateral, and combined tyre dynamics. The necessary parameters for such a detailed vehicle model are obtained through various rig measurements, together with CAD methods for some parts. These include Kinematics and Compliance (KnC) rig, tyre test bench, engine test bench, wind tunnel, and Finite Element Method (FEM) as well as Computational Fluid Dynamics (CFD) calculations. The usage of simulation within the scope of this work implies that the car behaves similar to reality, especially in context of the aspects that influence the driver's behaviour, which are essentially the handling characteristics. Even though the validation of the model and parameters is not discussed in this thesis, it is an imperative step before applying the derived method. Detailed descriptions of vehicle modelling and the used concepts can be found in [19][39] and more specifically on the suspension system in [34][22].

The discussed vehicle model is compiled for a suitable real-time target to be used for the DiLS. The demands specified for real-time execution, which are timeliness, simultaneity and determinacy, apply to driving in a simulator as there is a Driver-, Hardware-, or both in-the-Loop. While the vehicle model is run on one real-time system, the surrounding framework, including all hardware control tasks, runs on a separate machine. This is mainly to prevent extreme set-points for any actuators in the system should the vehicle model reach an unstable state and crash the machine it is hosted on. Regarding actuation, the DiLS consists of various systems to supply cues to the driver. Most importantly an electric motor applies torque to the steering column, which is essential feedback for the driver. Additional cues include some DoF of platform movement that excite the whole chassis, as well as tactile transducers to represent more detailed vibrations. Figure 4.1 shows a picture of the DiLS with a formula style chassis mounted. Some other characteristics can also be seen, such as the cylindrical screen featuring a  $210^\circ$  Field-of-View (FoV) and the rendered image that the driver is seeing. The projection achieves an update rate of 120 Hz to ensure a flicker free impression to the driver. The chassis mock-up is designed to mimic the real car as good as possible while keeping a low weight to maximise the motion platform's capabilities. Any ergonomic aspects are kept from the real car, including the original seat per driver, as well as other padding and other parts. The platform also allows changing the chassis to a GT-style mock-up, since those cars require a completely different ergonomic approach. Finally, the vehicle model is run at rates above 1 kHz due to the high stiffness of the modelled system, but supports logging all available states and calculated signals with lower rates and additionally provides real-time telemetry to the engineers in the control room. Overall, this allows to have all surrounding processes similar to those when running a real car on track to create sufficient immersion for the driver and usability of the DiLS for the engineers.

### 4.1.2 Database

As previously mentioned, data logged on various race cars is used to test the derived method and metrics. While the signals  $x_c$  and  $y_c$  are generated using the DiLS, the metrics for *driving style* analysis on the stabilisation level rely only on  $r_T$ ,  $p_B$  and  $a_S$ , which are available in almost any race car data log. A typical application, which is employed in this case as well, is a set of sensors that is wired to the ECU of the car. The ECU controls the engine and various other available systems, if available, and also takes care of logging the sensor inputs. As any DAQ system, this requires calibrating sensors and is prone to faults in either the system configuration, the sensors, or to errors induced over the wiring loom. Subsequently, this leads to the measures discussed in section 3.1.1 to ensure robust detection of the specified cornering patterns.

Regarding available data, three race seasons are analysed. This includes 28 race weekends, each consisting of different sessions. A session is a specified time interval in which the race cars are allowed to enter the track, for example "Free Practice 1", "Qualifying 3" or "Race 2". The amount of sessions per weekend depends on the racing series, and



Figure 4.1: Driver-in-the-Loop Simulator ©BMW AG

each session is following a scheme with regulations in place, such as a specified qualifying format to derive the starting positions for the race. Within a session every outing of the car is called a run and essentially means the interval between the car leaving the pit lane and the car entering the pit lane again after driving a certain amount of laps around the track. The first lap of a run is called Out-Lap, the last last is called In-Lap. The analysed dataset features seven sessions per race weekend. The amount of runs and laps per session is not defined and depends on the session itself, as well as on boundary conditions. For example, a race with one pit stop will have two runs but far more laps than a qualifying session that might have two runs as well, but only contain three laps each. Overall, the dataset consists of 27690 laps and comprises 10 different drivers. The additionally used DiLS data consists of four events which are not race weekends and consequently do not feature different sessions. Each event was run with a different driver which leads to the data consisting of four drivers and 747 laps in total.

Not all laps are suited for the proposed analysis, therefore a selection process is used to detect significant laps. It is assumed, that if a lap contains errors or is not driven at the limit which is an imperative condition for the method, this will result in an increase of lap time  $t_L$ . Consequently, laps are selected by a reference lap time  $t_{L,\text{ref}}$ , together with an additional margin around it. As  $t_{L,\text{ref}}$  needs to represent any changes of the environment that occur, as well as cater for drivers not being able to achieve the same performance, it represents the average best  $t_L$  per session. This means the best laps for each driver in a specific session are averaged to the reference time for this session. Subsequently, each session has its own  $t_{L,\text{ref}}$ . The boundaries to this reference were chosen empirically by inspecting the available data to select laps with  $t_L \geq 0.95 * t_{L,\text{ref}} \cap t_L \leq 1.05 * t_{L,\text{ref}}$ . The limit of 95% of  $t_{L,\text{ref}}$  assures to reject laps with faulty data logging resulting in a

| Source    | Total laps | Selected laps | Observations | Features |
|-----------|------------|---------------|--------------|----------|
| Car data  | 27690      | 17132         | 119563       | 109      |
| DiLS data | 747        | 602           | 4093         | 128      |

Table 4.1: Summary of available data

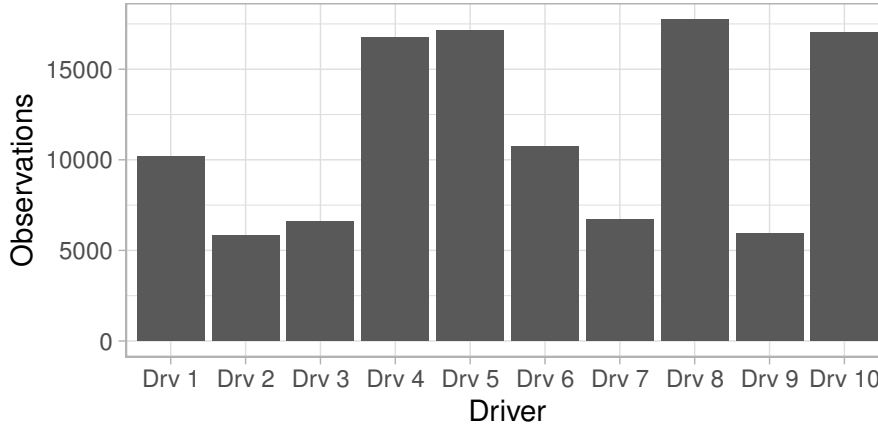


Figure 4.2: Distribution of car data

time significantly faster compared to the best laps. Additionally, a lap marker is used for selection. It is set by the engineers in charge of the car. Out- and In-Laps have their own marker and are excluded accordingly. Another marker is available for compromised laps, allowing automatic exclusion of those as well. Running the explained lap selection process and the method for feature extraction defined in this work on both datasets results in the total amount of available data, as summarised in table 4.1, where the number of observations is the number of detected corners from the specified amount of selected laps. The distribution of drivers over the observations is shown in Fig. 4.2 for car data and Fig. 4.3 for DiLS data.

Because the distribution is not equal among drivers, the influence of this distribution is further assessed. Additionally, the distribution among the used tracks is taken into account, as displayed in Fig. 4.4. Some tracks have more observations than others in general, and also specific drivers are not equally spread across the tracks as well. The influence of this on the feature importances, as introduced in section 3.3, is analysed by using the loss of accuracy metric for a *Random Forest* classifier trained on different datasets. For this purpose, two subsets are sampled from the complete dataset. One subset consists of equally distributed samples across the drivers for each track. The distribution of this set is shown in Fig. 4.5 and illustrates the different sample sizes per track. The second dataset is sampled in a way that it is equally distributed over all drivers and tracks, it is illustrated in Fig. 4.6. Both approaches reduce the available data significantly, as listed

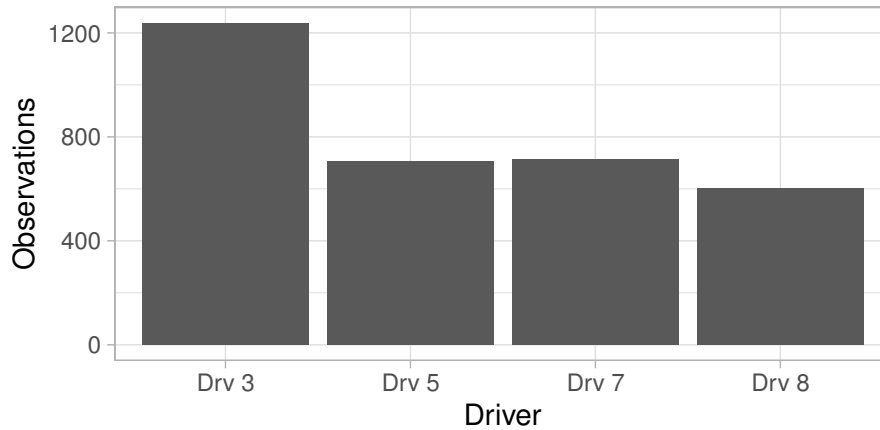


Figure 4.3: Distribution of DiLS data

| Dataset                | Observations |
|------------------------|--------------|
| Complete               | 114728       |
| Equal Drivers          | 44840        |
| Equal Tracks & Drivers | 27650        |

Table 4.2: Number of observations in reduced datasets

in table 4.2. The number of observations for the complete dataset is reduced because those containing Not-a-Number (NaN) values have been removed. Finally, a *Random Forest* classifier is trained for each of the sets with the intention to compare feature importances as per loss in accuracy. The resulting graph is displayed in Fig. 4.7 and only shows the 50 top ranked features for better visibility. The main shape of the importance metric over all features is clearly visible. Changes in importance of features between the datasets can be noticed but do not change the general distribution. Consequently, the complete dataset is used for further analysis to exploit all available observations.

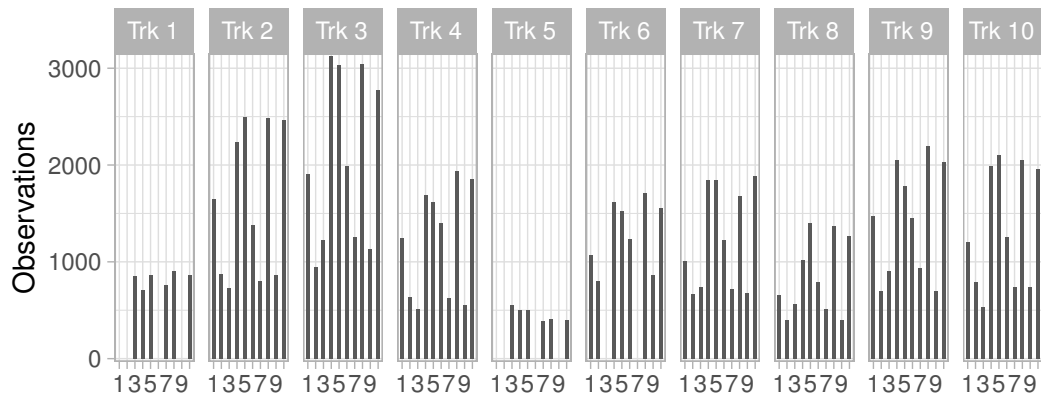


Figure 4.4: Distribution of car data per track

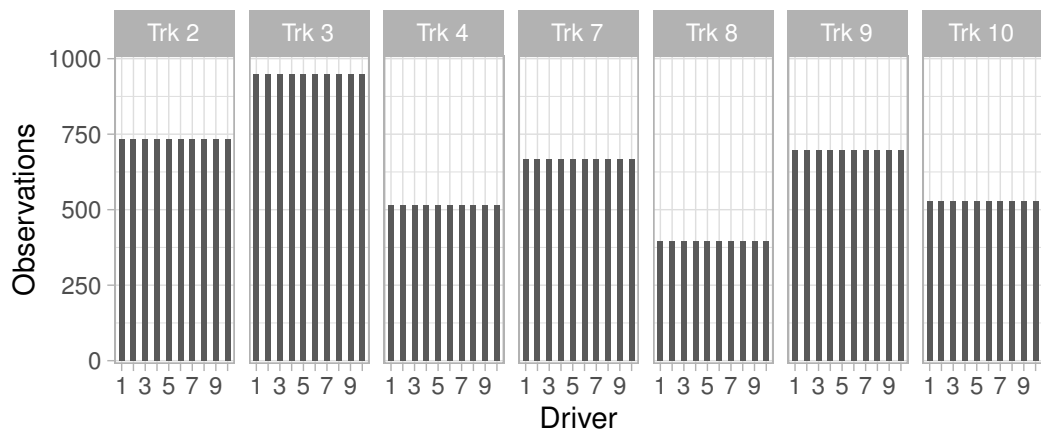


Figure 4.5: Distribution of car data sampled equally per track

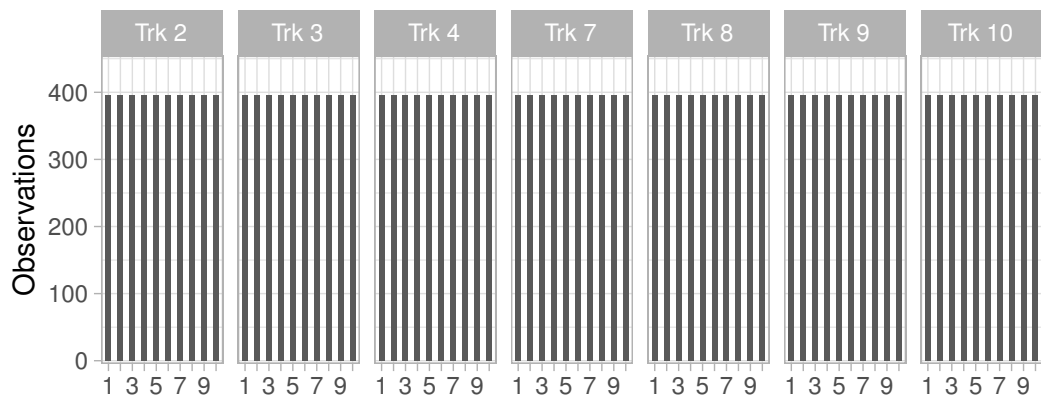


Figure 4.6: Distribution of car data sampled equally per track and driver

## 4.1 Simulation Environment and Available Data

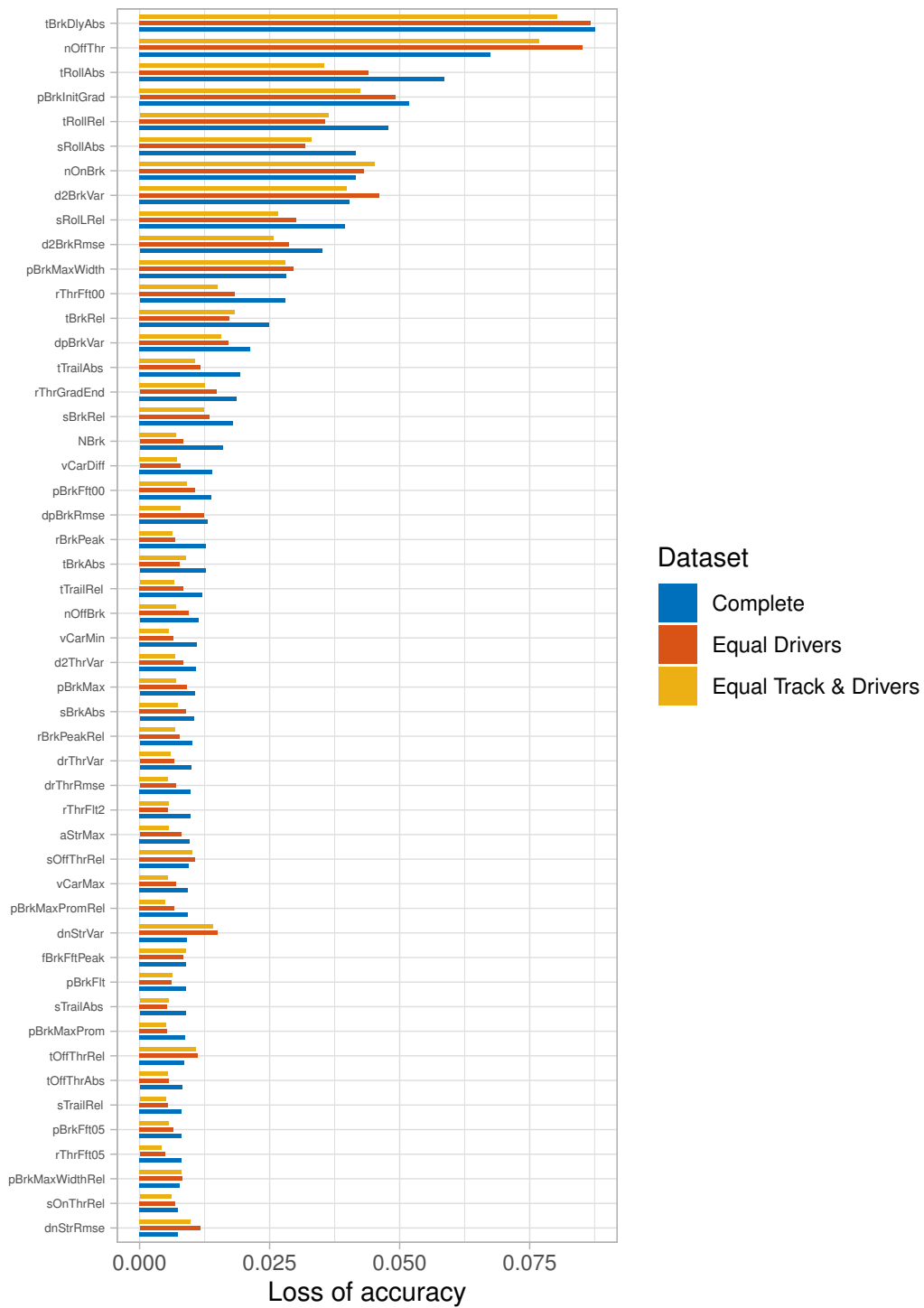


Figure 4.7: Loss in accuracy per feature for all three data sets

### 4.1.3 Used Software

This section gives a short overview of the software used to obtain the results discussed in the following sections. All measured data is stored in a format compatible with **Magneti Marelli WinTAX4**, a data analysis software commonly used in motorsports. It provides an Application Programming Interface (API) to access data structures, values, and properties through other applications. This functionality is used together with **The MathWorks® MATLAB** for pattern recognition and metric calculation. The resulting dataset is then further analysed with the open-source statistical computing language **R** [47] and the **R Studio** Integrated Development Environment (IDE) [50]. This involves the following packages, which are all accessible as open-source projects:

- **tidyverse** [66], specifically the included packages **dplyr** [67] for data restructuring and **ggplot2** [65] for plotting data
- **randomForest** [30], for the implementation of Breiman's *Random Forest* algorithm [4]
- **ranger** [70], for the efficient and parallel implementation of *Random Forest*
- **caret** [27], for splitting test and train data as well as Recursive Feature Elimination
- **CORElearn** [49], for the efficient implementation of the *Relief* algorithm

## 4.2 Selection of Objective Criteria

The following section describes the results of applying the feature selection methods introduced in section 3.3 to the generated metrics. In the first step, the amount of NaN values per feature is assessed and those with a significant amount compared to the number of observations are dismissed. All features that are expected to have strong correlations are evaluated with the aim to select the most significant ones. This step is then followed by an analysis of the remaining features regarding unintended correlations between features, as well as their scores from feature selection methods. While machine learning techniques, especially for classification problems, are widely used to pick a set of features from all proposed candidates, the overall aim of this work is not to provide the features suited best for classifying drivers. The scope of identifying a driver's *driving style* is closely related to a classification problem, but in some instances a consistent selection of features is favourable for further engineering work over the best performing ones.

### 4.2.1 Stabilisation Level

As mentioned, the percentage of NaN values per feature is examined first. This step eliminates the features resulting from the FFT analysis of the throttle signal with respect to the first peak in the frequency spectrum. The metric calculation algorithm sets the



| Metric           | Number of NaNs | NaN Percentage |
|------------------|----------------|----------------|
| rThrFftPeak      | 31857          | 26.74 %        |
| fThrFftPeak      | 31857          | 26.74 %        |
| nThrFftPeakWidth | 31857          | 26.74 %        |
| nThrFftPeakProm  | 31857          | 26.74 %        |
| pBrkFftPeak      | 2989           | 2.51 %         |
| fBrkFftPeak      | 0989           | 2.51 %         |
| nBrkFftPeakWidth | 2989           | 2.51 %         |
| nBrkFftPeakProm  | 2989           | 2.51 %         |
| pBrkFft00        | 36             | 0.03 %         |
| pBrkFft05        | 36             | 0.03 %         |

Table 4.3: Number of NaN observations for control metrics

values for peak magnitude, peak frequency, peak width and peak prominence to NaN if no peak is found at all, which is usually caused by a monotonic decreasing frequency spectrum. The same metrics respective to the brake signal do also contain significantly more NaN values compared to the remainder of the metrics, however, as listed in table 4.3, the relative amount is only 2.51 % of the dataset. The list in table 4.3 is in descending order and shows the subsequent metrics  $pBrkFft00$  and  $pBrkFft05$  as well for better comparison.

A complete correlation matrix of all metrics for the *driving style* on the stabilisation level is shown in Fig. 4.8. The *Pearson* correlation coefficient is illustrated for each combination of metrics by its value multiplied by 100 for easier reading, as well as a colour gradient from blue to red indicating strong negative or strong positive correlations. While the scale does not allow detailed analysis, blocks of correlated metrics can easily be identified. The metrics based on relative and absolute time or distance covered between corner points, such as the relative distance travelled in the *on-throttle* phase  $sThrOnRel$ , were defined with the expectation that they share the same information thus are strongly correlated. This can be confirmed as the most distinct blocks of correlated features in Fig. 4.8 are located at the diagonal and contain sets of four features each. A more detailed correlation matrix of those 24 features is shown in Fig. 4.9 in the same formatting to display the *Pearson* correlation coefficients as explained.

## 4.2 Selection of Objective Criteria

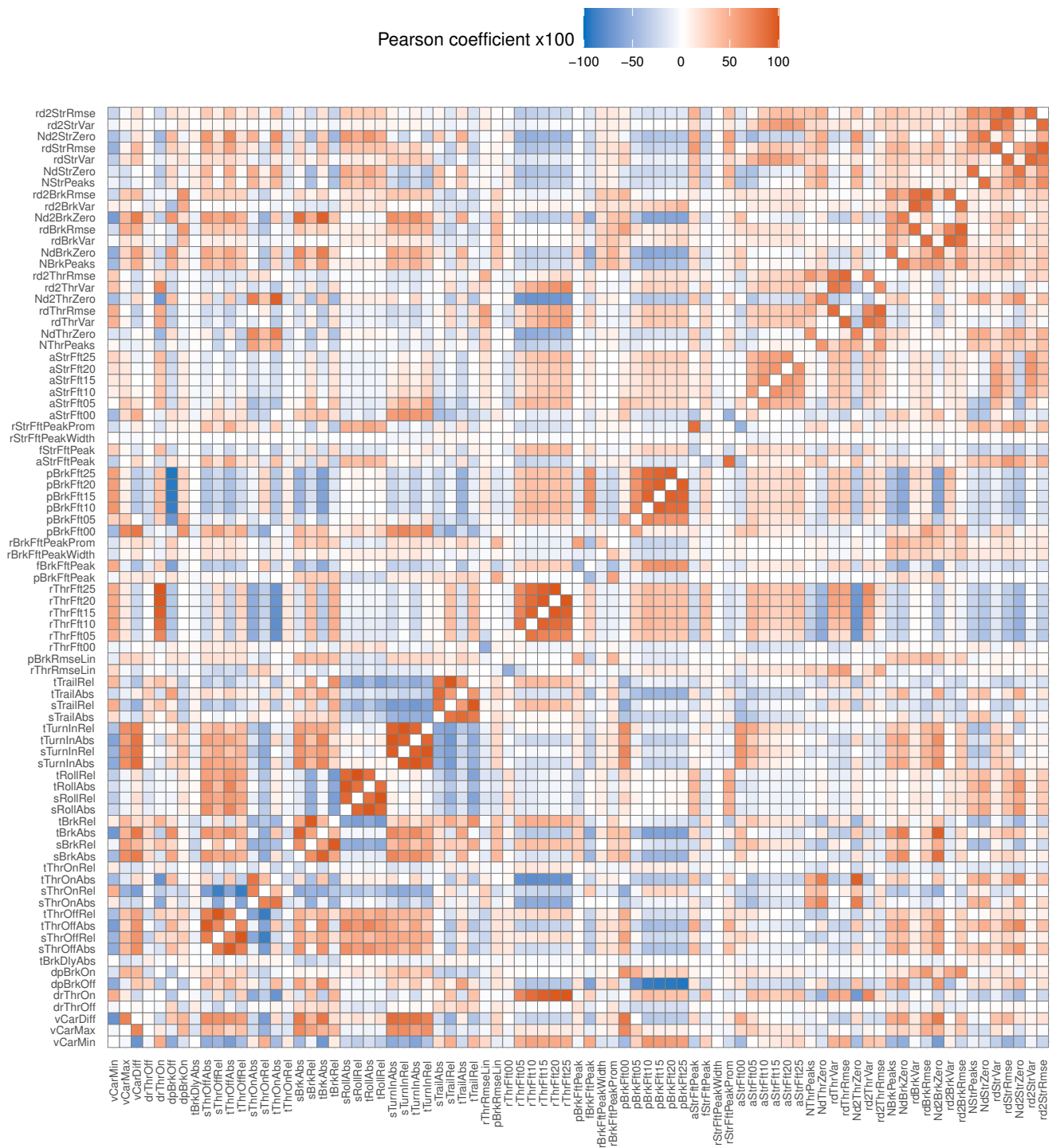


Figure 4.8: Correlation matrix for objective metrics describing the stabilisation level of the driving task

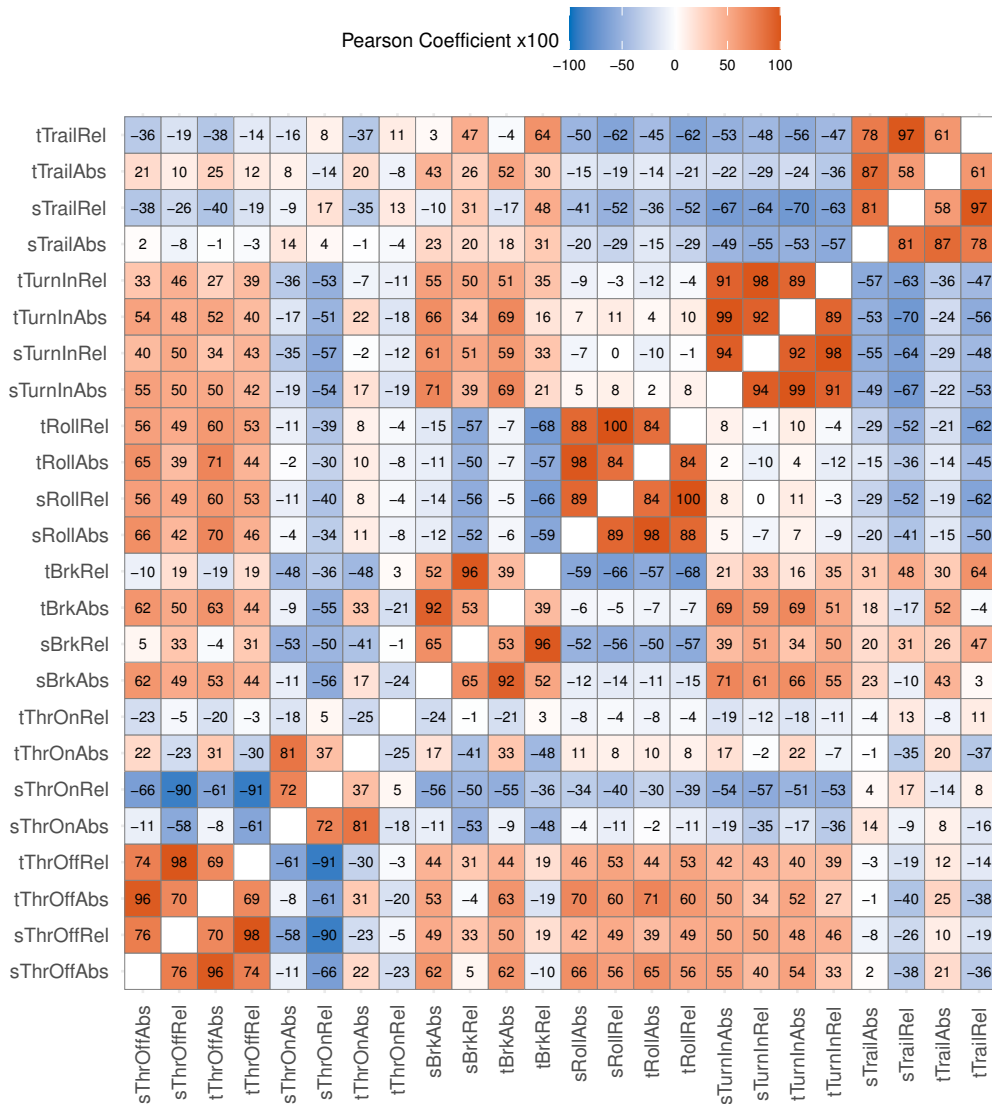


Figure 4.9: Correlation matrix for correlated time and distance based metrics

Those features shall be reduced to a set of six independent features using the feature selection techniques detailed in section 3.3. This is achieved mostly by using the *Relief* weights for each set of correlated features, as shown in Fig. 4.10. Additionally, the loss of accuracy derived from a *Random Forest* classifier trained on the whole feature set is analysed and displayed in Fig. 4.11. The mean *Pearson* correlation coefficient per set of metrics allows to judge how much each four metrics are correlated. These values are listed in table 4.4 and indicate that the metrics describing the *on-throttle* phase are significantly less correlated with an average *Pearson* coefficient of 0.40, while the metrics

describing the *rolling* and *turn-in* phase show the highest average correlation coefficients of 0.90 and 0.94. These observations are also visible in Fig. 4.9 from the shade of the colour fill.

The *Relief* weights shown in Fig. 4.10 indicate a significant differentiation of absolute metrics and those measured relative to the corner length, with the only exception of the braking metric shown in Fig. 4.10 (c). The comparison between time and distance based metrics is not as significant, however the distance based relative metric is favoured in all cases except for the aforementioned braking phase. Notably the *on-throttle* metrics shown in 4.10 (b), which are clearly less correlated among themselves, have the highest weight on the distance based relative metric but do not follow the pattern for the other three. A comparison to the ranking of the metrics in each set derived from the loss of accuracy for a *Random Forest* classifier, as shown in Fig. 4.11, does not provide the same clear picture. The braking metrics shown in Fig. 4.11 (c) indicate the same anomaly as seen in 4.10 (c) for the *Relief* weights, although in slightly different order. The *rolling* metrics shown in 4.11 (d) have a completely different pattern with both time based metrics ranked higher compared to the distance based metrics and the distance based relative metrics ranked last. Controversially the ranking of the *turn-in* metrics displayed in 4.11 (e) rates both distance based metrics higher. Both sets of metrics have the highest average correlation coefficients, which is an indication to the usefulness of the importance derived from a *Random Forest* classifier. Since the loss of accuracy for removing one of a set of highly correlated metrics is similar for each of them due to the fact that three redundant features are still available, this importance ranking is to be used with care. The clear picture of the *Relief* weights, which is partly supported by the importance assigned from the trained model, leads to the decision to select distance based relative metrics. This type is selected from all six feature sets for consistency of the final metric set.

| Metric              | Mean Pearson coefficient |
|---------------------|--------------------------|
| Off-throttle phase  | 0.81                     |
| On-throttle phase   | 0.40                     |
| Braking phase       | 0.66                     |
| Rolling phase       | 0.90                     |
| Turn-in point       | 0.94                     |
| Trail-braking phase | 0.77                     |

Table 4.4: Average correlation of distance and time based metrics

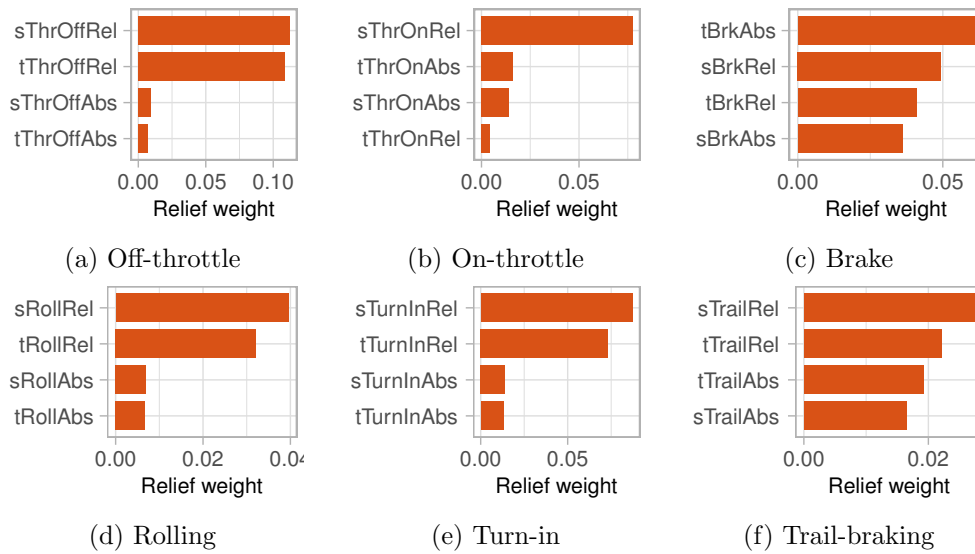


Figure 4.10: Relief weights for correlated time and distance based metrics

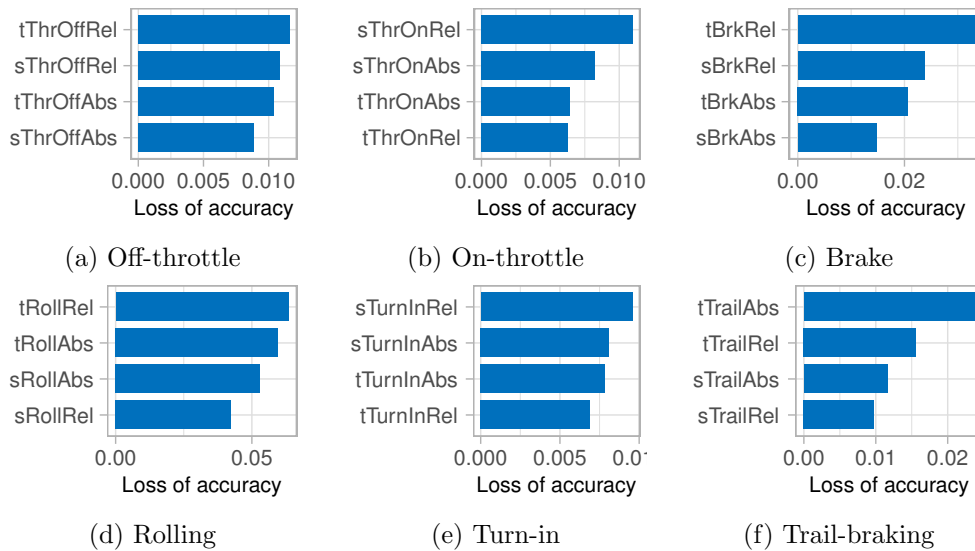


Figure 4.11: Loss of accuracy for correlated time and distance based metrics

A similar approach is used to select relevant metrics from the sets describing the RMSE, variance and zero crossings of  $r_T$ ,  $p_B$  and  $a_S$  concerning their first and second derivatives. The correlation matrix of the analysed features is shown in Fig. 4.13 in the same format as used before. The correlation coefficients show the highest absolute values around the diagonal, which allows separation into three blocks of features each related to one of the analysed driver control signals. The average absolute correlation coefficients are lower than the previously assessed time and distance based met-

rics, yet high enough to consider reducing those feature sets significantly. The metrics calculated from  $r_T$  have an average absolute correlation coefficient that is 0.1 lower compared to the other two feature sets, however there is no clear difference as seen before with time and distance based metrics. The respective values for derivative based metrics are listed in table 4.5. The *Relief* weights are used as a means of ranking the available features again, in accordance to the previously discussed outcomes. Figure 4.12 shows these weights in descending order for each of the currently discussed metrics separated into one sub-figure per signal. For all three sets the metrics using the RMSE are ranked the highest compared to the variance and zero crossing based metrics. The ranking is not as clear for both RMSE based metrics for  $p_B$ , however  $rd2BrkRmse$  has a significantly higher weight compared within its feature set, as well as compared to any other derivative based metric. Interestingly, the metrics analysing the peaks of the non-derived signals are rated similar to other metrics for  $r_T$  and  $p_B$ , while it is ranked highest for  $a_S$ . This leads to keeping the number of peaks, as well as the RMSE to zero of the first and second derivative for further evaluation.

| Metric | Mean Pearson coefficient |
|--------|--------------------------|
| $r_T$  | 0.44                     |
| $p_B$  | 0.53                     |
| $a_S$  | 0.55                     |

Table 4.5: Average correlation of derivative based metrics

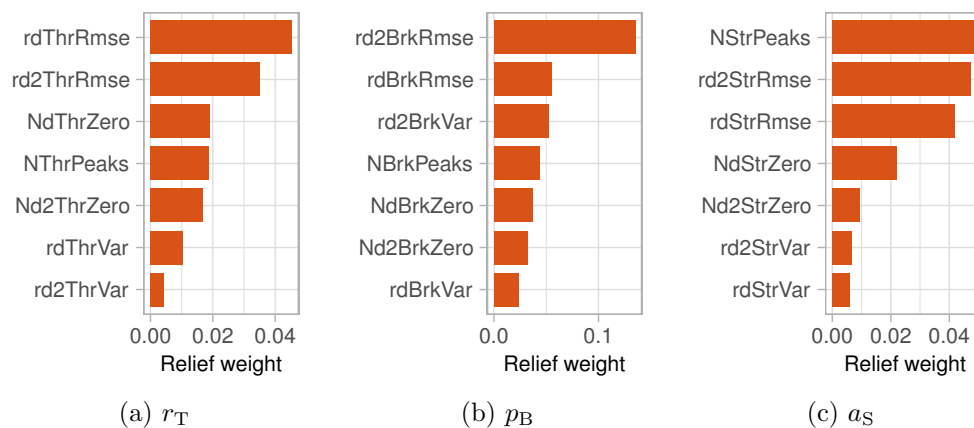


Figure 4.12: Relief weights for correlated derivative based metrics

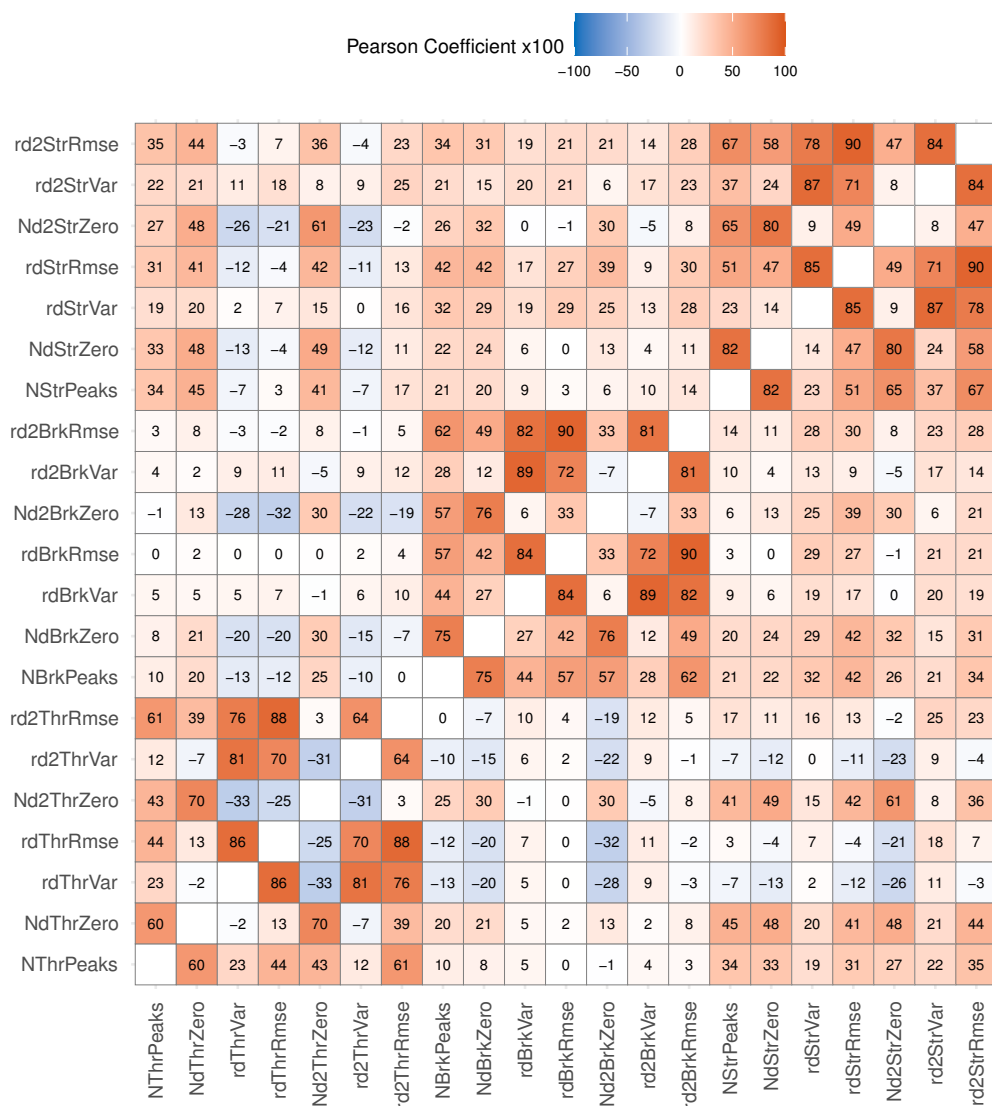


Figure 4.13: Correlation matrix for correlated derivative based metrics

In accordance to the discussed steps, all metrics based on FFT analysis are assessed and selected as well. The correlation matrix reduced to the relevant features is shown in Fig. 4.15. While the metrics derived from the signal magnitude at discrete frequencies are available for  $r_T$ ,  $p_B$ , and  $a_S$ , the analysis of a peak in the frequency spectrum is only looked at for the latter two, because the calculation of the  $r_T$  based metrics produced too many NaNs and those were dropped consequently as discussed. The average absolute correlation coefficients per set of metrics are listed in table 4.6. The feature sets related to the peak analysis are significantly less correlated among themselves compared to the features queried at specific frequencies, however  $a_S$  does not follow this pattern. Figure

4.15 suggests that the  $a_S$  metric for 0 Hz is not significantly correlated with the remaining features of this set, however the other five metrics show the same pattern as seen for  $r_T$  and  $p_B$ . For this reason, the metrics concerning  $a_S$  are reduced similar to the others while the peak analysis metrics are kept for further analysis. The *Relief* weights are illustrated in Fig. 4.14 in the same way as before. The ranking shows a clear tendency to assign the feature weights decreasing with increasing frequency value of the metrics. The values queried at 0 Hz and 0.5 Hz have a significant difference while the metrics ranging from 1.0 Hz down to 2.5 Hz are not clearly differentiable. Based on these results the FFT based metrics for discrete frequencies from 1.0 Hz to 2.5 Hz are eliminated from the overall feature set for all three control signals.

| Metric     | Mean Pearson coefficient |
|------------|--------------------------|
| $r_T$      | 0.58                     |
| $p_B$      | 0.60                     |
| $a_S$      | 0.38                     |
| $p_B$ Peak | 0.33                     |
| $a_S$ Peak | 0.27                     |

Table 4.6: Average correlation of FFT based metrics

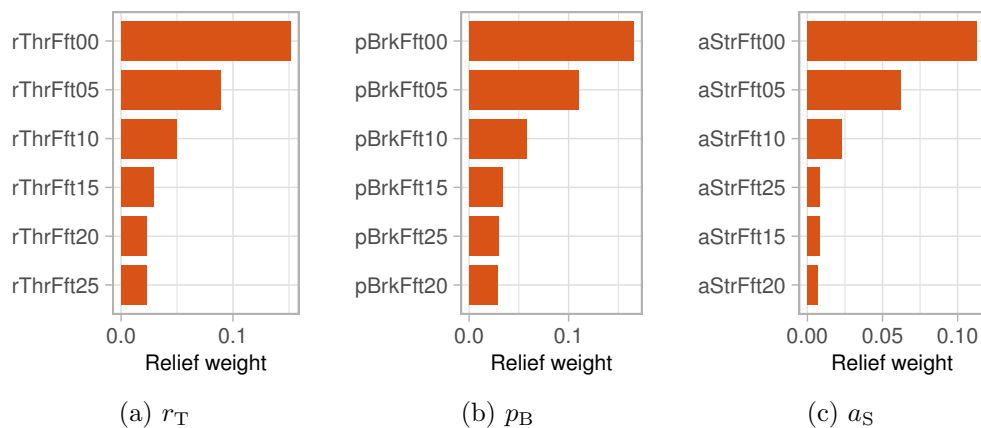


Figure 4.14: Relief weights for correlated FFT based metrics



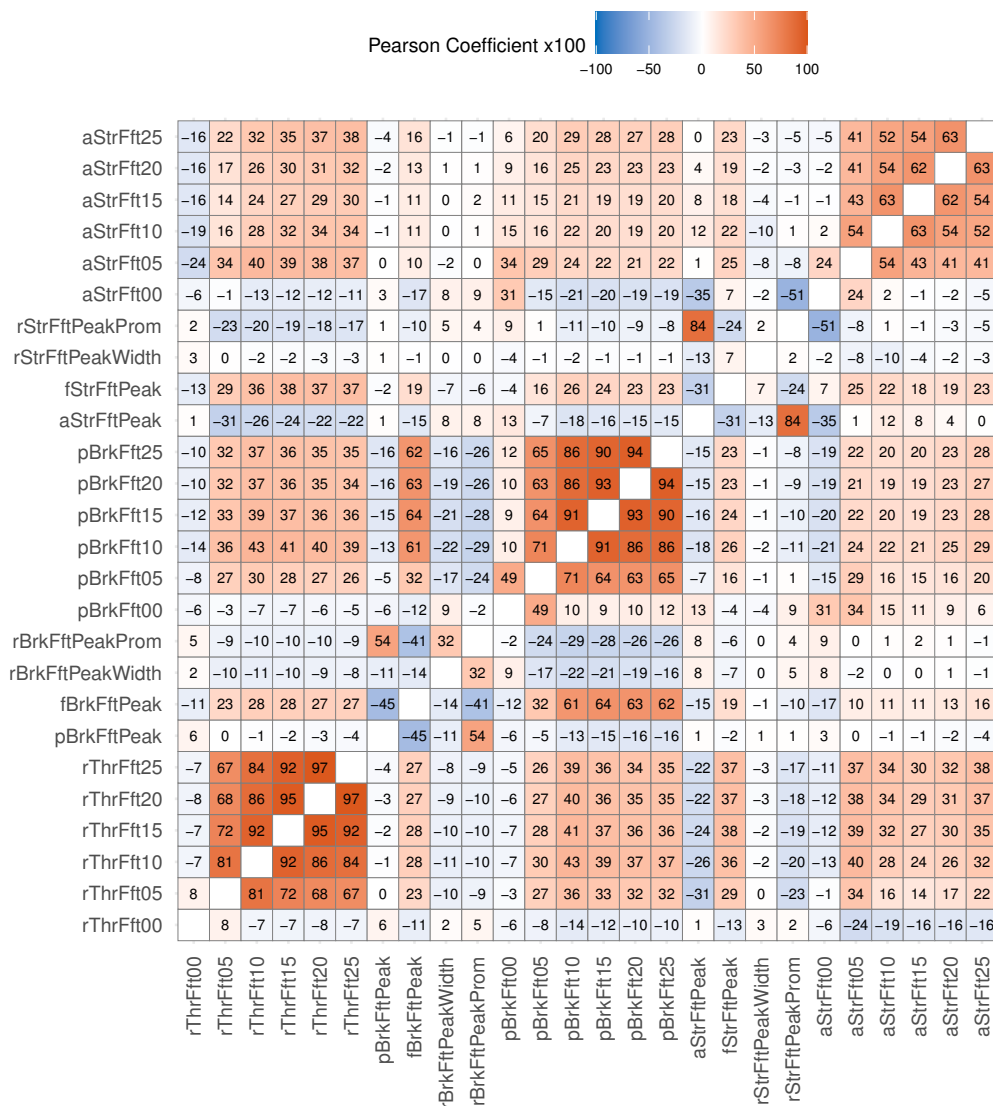


Figure 4.15: Correlation matrix for correlated FFT based metrics

The first step of reducing metrics that contain the same information in different variants leads to a reduction of metrics for the stabilisation level from 81 to 39 features. Consequently, further reduction is done on basis of ranking features. This includes reducing accidentally correlated features after the expected redundancies have been removed in the previous step. Additionally, uncorrelated features are ranked against each other as well to evaluate their significance. As outlined before, this is done with an engineering application in mind, as opposed to pure improvement of the driver classification. The correlation matrix for the remaining features is displayed in Fig. 4.16, together with the *Relief* weights in Fig. 4.17 and the loss of accuracy from the *Random Forest* model as well as from RFE in Fig. 4.18. The latter results are only shown for features selected by the RFE algorithm. All others can be identified by the missing RFE score.



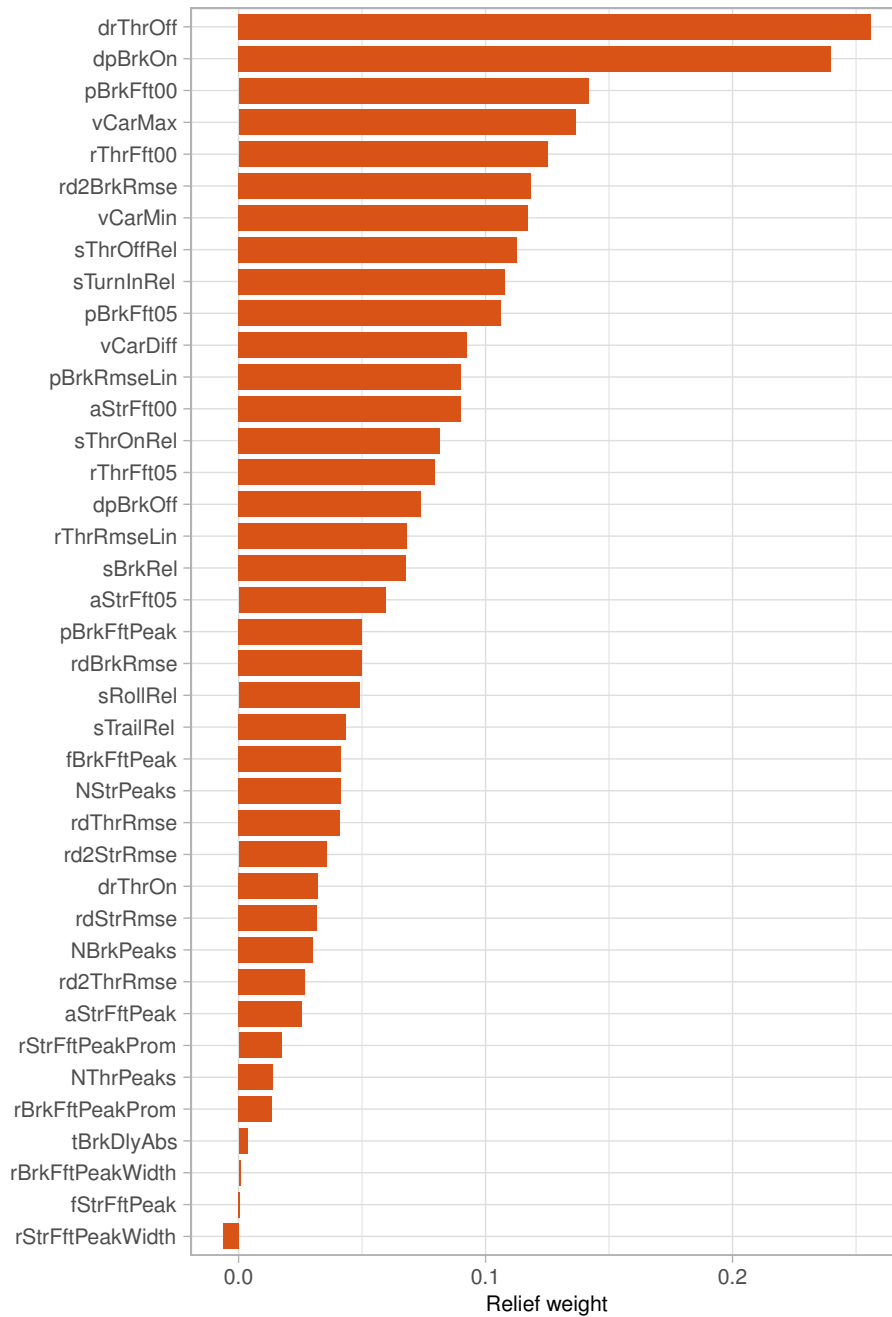


Figure 4.17: Relief weights for reduced feature set regarding stabilisation level

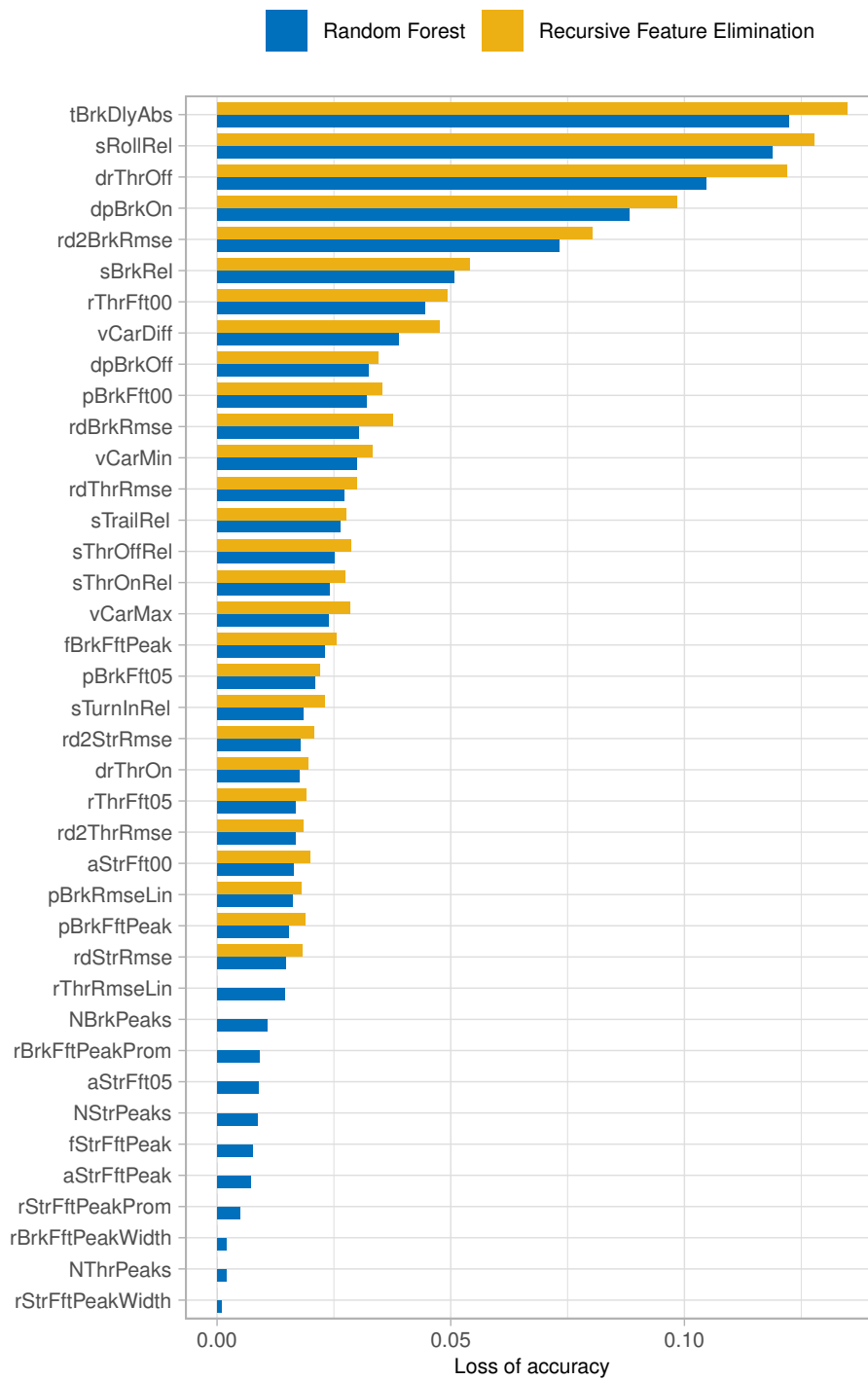


Figure 4.18: Loss of accuracy for reduced feature set regarding stabilisation level

Figure 4.16 immediately reveals that there are still strong correlations between the number of peaks and the RMSE to zero of the first and second derivative for all three analysed signals. These were chosen to be kept because the previously examined graphs did not imply a clear selection since they were focused only on the respective metrics. However, the now reduced feature set allows a better comparison in the overall context, which shows that *rd2BrkRmse* is clearly ranked higher regarding *Relief* weights as well as loss of accuracy than any other of the currently discussed features. While *rdThrRmse* is ranked slightly higher than the metric obtained from the first derivative, the peak count is favoured over the second derivative for  $a_S$ . Due to all peak counts not being selected by the RFE algorithm and ranked low by *Relief* as well, together with the significant difference in ranking for *rd2BrkRmse*, the metrics obtained from the second derivative are kept while the six remaining metrics are eliminated.

The previous results led to the decision for distance based metrics measured relative to the individual corner length. The six selected metrics are now evaluated further. A correlation matrix for those is shown in Fig. 4.19, while the *Relief* weights and loss of accuracy graphs can be seen in Figs. 4.20 and 4.21. A significant correlation is noticeable between *sThrOffRel* and *sThrOnRel*, which is easily explainable by the fact that these two phases almost make up the entire cornering phase, with only the very short transition from full throttle to idle being left out. The *Relief* rating prefers the metric concerning the *off-throttle* phase, whereas loss of accuracy shows only a slight tendency to support this with both metrics being rated almost equally low. As the remainder of the discussed metrics is ranked with strong disagreements between *Relief* weights and loss of accuracy all five except *sThrOnRel* are kept.

Regarding the metrics calculated from FFT results, there are some correlations among them, as well as a clear picture from the feature rankings. First, the metrics used to analyse the major peak in the FFT of  $r_T$  were removed due to robustness issues leading to many NaN values. The feature selection derived with RFE suggests removing the peak width and prominence features for  $p_B$  also, as well as all four similar metrics related to  $a_S$ . The *Relief* weights support this selection and due to the rather low ranking of the two remaining features *pBrkFftPeak* and *fBrkFftPeak* those are eliminated all together. A more detailed correlation matrix for the remaining FFT based metrics and their correlated features is shown in Fig. 4.24. This reveals significant correlations between the aforementioned metrics and the gradient based metrics for  $r_T$  and  $p_B$ . Reduced graphs for *Relief* weights and loss of accuracy in the usual format are provided in Figs. 4.22 and 4.23. Both methods agree in a higher ranking of *drThrOff* and *dpBrkOn* in comparison to the other metrics evaluated in this context. There are no particularly strong correlations for *drThrOff*, but the high importance of *dpBrkOn* implies removing the correlated FFT based metrics for  $p_B$ . While *dpBrkOff* is not preferred by the rankings, it is correlated to the aforementioned metrics. The remaining strong correlation between *drThrOn* and *rThrFft05* does not provide such a clear image and especially the related but not correlated *rThrFft00* shows high scores for both feature rankings. Due to the overall difference in favour for gradient based metrics, as well as the noticeable difference

in computational effort between FFT and gradient analysis, the gradient based features are decided to be kept. The remaining FFT based features are removed, which leads to the elimination of all FFT based features.

The remaining features  $rThrRmseLin$  and  $pBrkRmseLin$  do not show significant correlations to other metrics with *Pearson* correlation coefficients of up to 0.56. While they are both dropped by RFE, the *Relief* weights imply that they are not completely irrelevant. The metric  $tBrkDlyAbs$  shows very little correlation to all other features. It is ranked high by loss of accuracy but almost least regarding *Relief* weights. Due to this discrepancy but two scores in favour of the metric it is kept as well.

Finally, the three metrics describing the corner,  $vCarMin$ ,  $vCarMax$  and  $vCarDiff$  are evaluated. The overall correlation matrix shown in Fig. 4.16 reveals that  $vCarDiff$  is strongly correlated to the other two for obvious reasons, however the metrics for minimum and maximum car velocity throughout the cornering phase do not show correlations amongst each other. The overall rankings shown in Figs. 4.17 and 4.18 indicate that all three metrics provide use to the classification, but are inconsistent regarding the order of aforementioned three features. Based on the correlation coefficients it is decided to keep two of the three metrics which leads to eliminating  $vCarDiff$ , because it seems this metric can not provide the entire information contained in both remaining metrics.

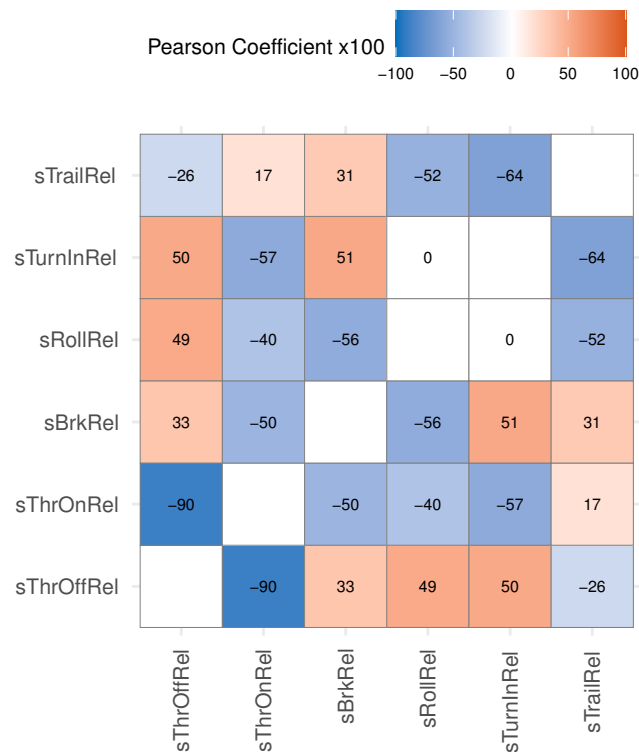


Figure 4.19: Correlation matrix for distance relative features of the reduced set

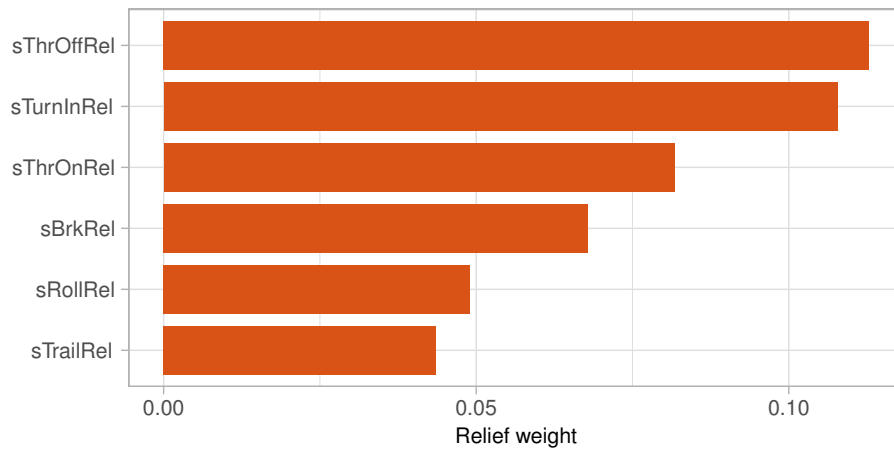


Figure 4.20: Relief weights for distance relative features of the reduced set

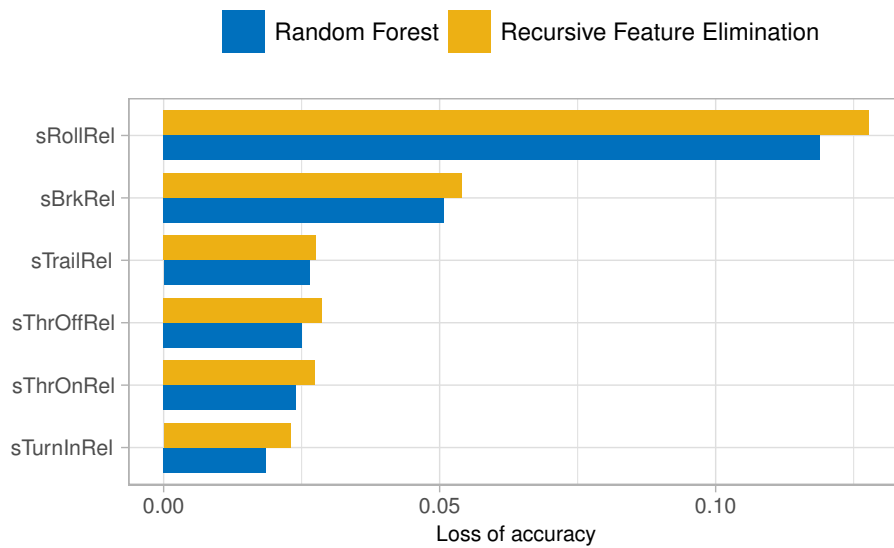


Figure 4.21: Loss of accuracy for distance relative features of the reduced set

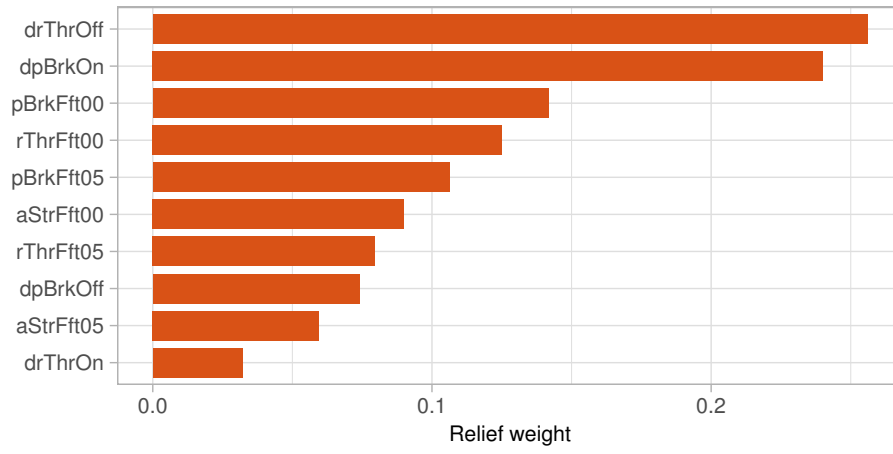


Figure 4.22: Relief weights for FFT related features of the reduced set

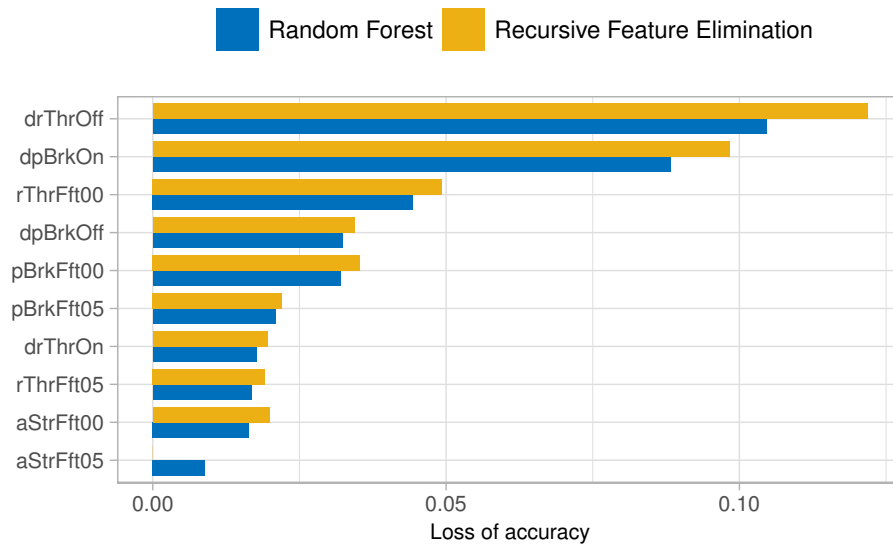


Figure 4.23: Loss of accuracy for FFT related features of the reduced set



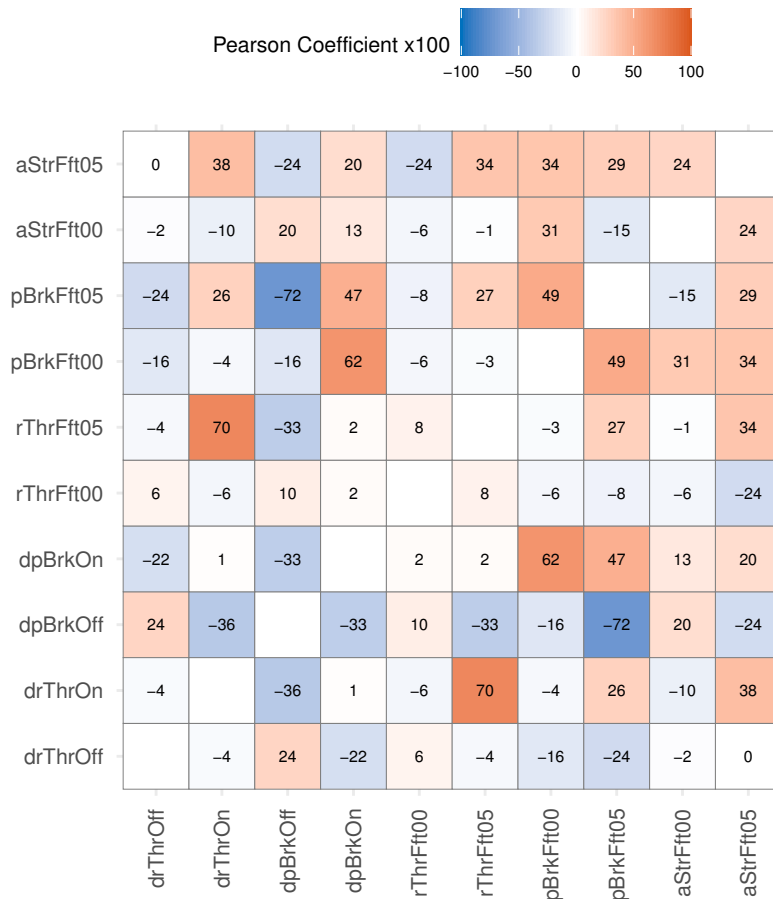


Figure 4.24: Correlation matrix for FFT related features of the reduced set

### 4.2.2 Guidance Level

The analysis of features to describe the *driving style* on the guidance level follows the same approach as described for the stabilisation level. The examination of the NaN values per metric shows that the highest percentage is 9.06 %, followed by 7.65 %, and a decrease to 1.05 % for the metric ranked third. These are listed in table 4.7. While no features are eliminated at this stage, it is noticeable that all three metrics with the most NaN values are related to a "V-angle" derived from a time-series signal. This indicates that the calculation based on trigonometric functions is suffering robustness issues for certain observations.

Time and distance based metrics in absolute and relative variants have been defined for the point of maximum path curvature, the point with minimal distance to the inner track limits and the point with minimal distance to the track centre line in a similar way to the discussed metrics on the stabilisation level. Accordingly, those are analysed for correlations first, which is displayed in Fig. 4.25. Additionally, the average absolute correlation coefficients among the features of each set are listed in table 4.8. As Fig. 4.25 shows, the distinction of the three main blocks of correlated features is not as clear from the color gradient as with for example Fig. 4.9, which shows the correlations among the previously discussed time and distance based metrics. While the individual correlation numbers per set imply that there are redundant metrics, the overall correlation value suggests that also the three defined categories of metrics share common information and can be possibly reduced.

| Metric | Number of NaNs | NaN Percentage |
|--------|----------------|----------------|
| aVCen  | 371            | 9.06 %         |
| aVApex | 313            | 7.65 %         |
| aVCurv | 43             | 1.05 %         |

Table 4.7: Number of NaN observations for control metrics on the guidance level

| Metric              | Mean Pearson coefficient |
|---------------------|--------------------------|
| All                 | 0.39                     |
| Curvature           | 0.58                     |
| Apex distance       | 0.68                     |
| Centerline distance | 0.61                     |

Table 4.8: Average correlation of distance and time based metrics

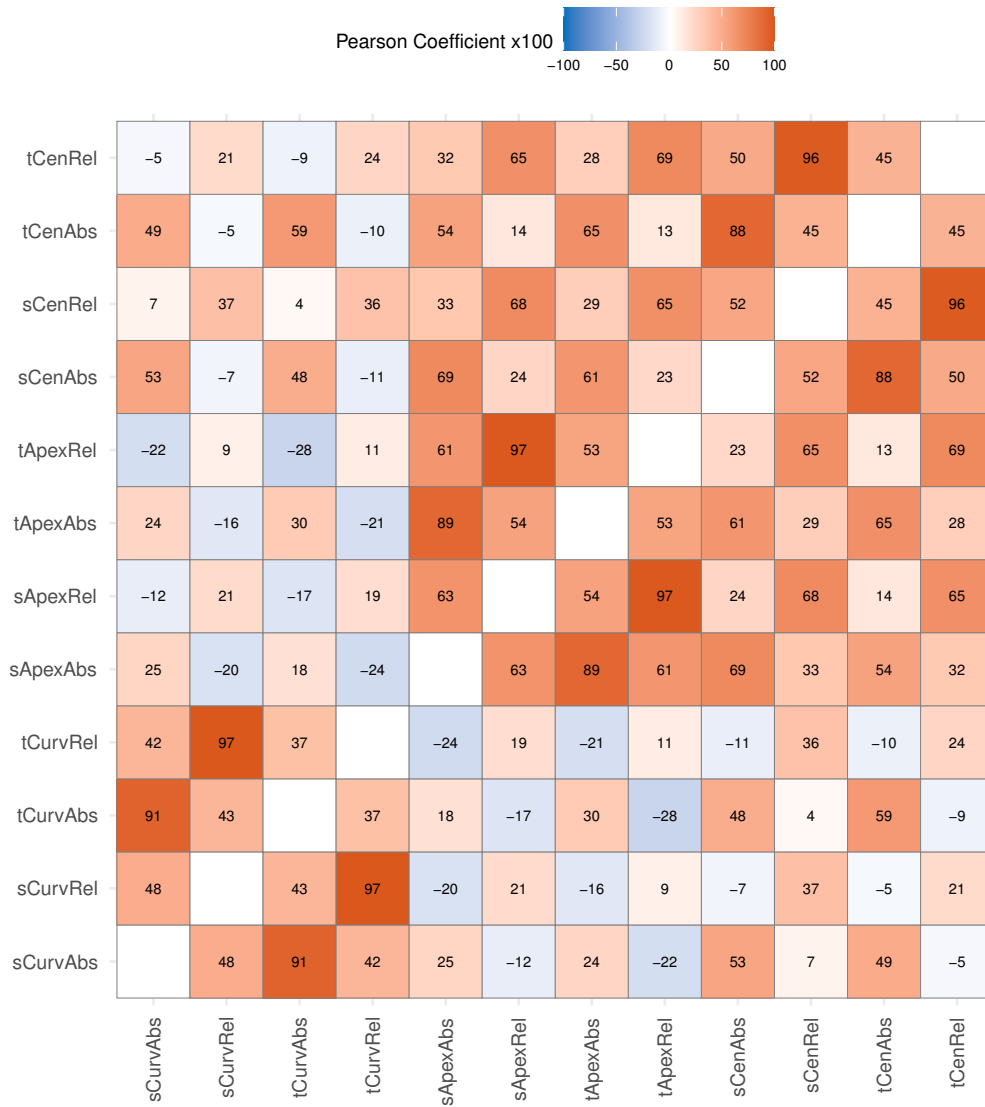


Figure 4.25: Correlation matrix for objective metrics describing the guidance level of the driving task

The analysis of *Relief* weights does not entirely follow the previous results. A sub-plot for each set of metrics showing the *Relief* weights of the respective features is displayed in Fig. 4.26. While for the metrics concerning the maximum path curvature the distance based relative metric is ranked highest and both relative metrics have significantly higher weights compared to the absolute variants, the other feature sets do not show such a clear picture, except from ranking the time based absolute metric last. For this reason also the ranking derived from a *Random Forest* classifier, which was trained on the trajectory metrics, is taken into account and shown in Fig. 4.27. The three sub-figures follow the same pattern of ranking the distance based absolute metric highest, followed by the time based absolute metric and both relative metrics ranked last and less differentiable. While this implies to use distance based metrics also for the guidance level, it also suggests using their absolute variants. Based on the scope of this work, a consistent approach is favoured over classification accuracy leading to the decision to keep distance based relative metrics for the guidance level, similar to the stabilisation level. This is supported by the *Relief* weights, which either favour this variant or rank it insignificantly lower compared to other available options.

Like for the stabilisation level, the elimination of correlated features reduced the metrics to describe the *driving style* on the guidance level from 19 to 10 metrics. The next step is again to further reduce the remaining features. Following the previous approach a correlation matrix, *Relief* weights, and loss of accuracy derived from the *Random Forest* model, as well as the RFE algorithm are shown in Figs. 4.28, 4.29 and 4.30. The correlation matrix reveals a strong correlation between the maximum curvature and the integral of the curvature throughout a corner and implies to eliminate either metrics respective to the centre line or related to the inner track limit. The *Relief* weights of the aforementioned metrics regarding path curvature suggest that they are both irrelevant. Contrary, the integral of curvature is ranked first according to loss of accuracy, whereas the maximum curvature is ranked third. Opposing to these results *cMax* is kept while *cInt* is removed from the feature set, following the idea to keep

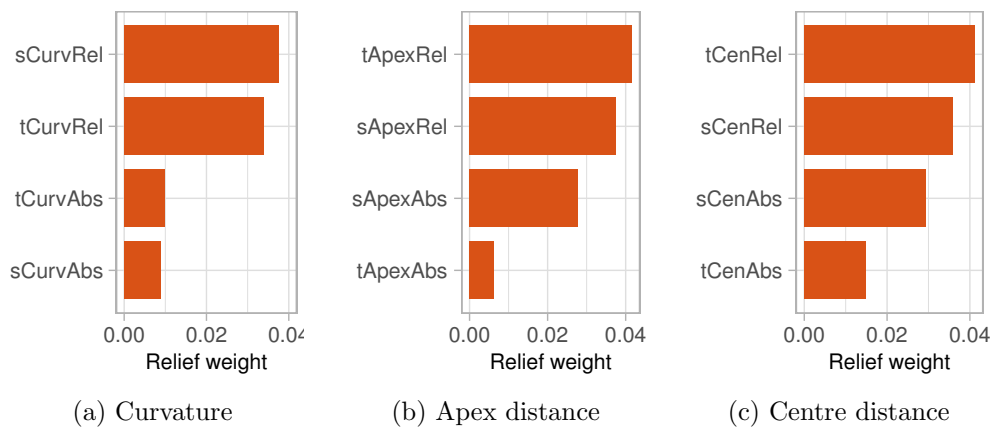


Figure 4.26: Relief weights for correlated distance and time based metrics

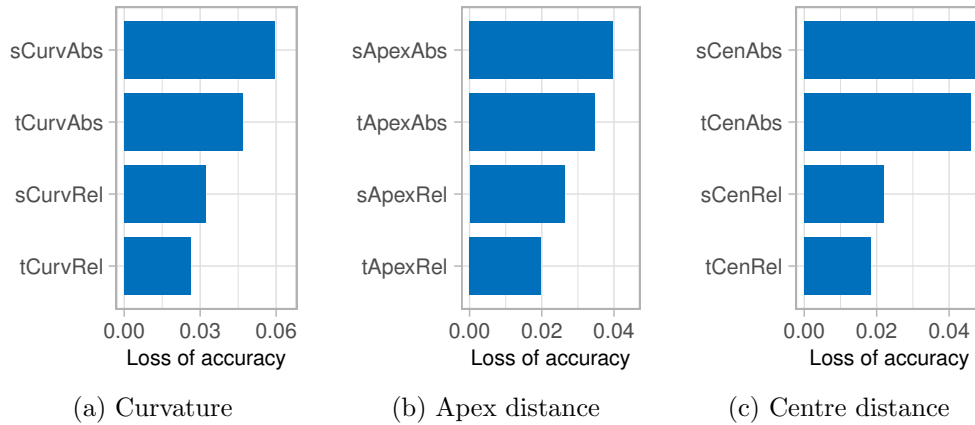


Figure 4.27: Loss of accuracy for correlated distance and time based metrics

consistent sets of metrics. This is because the related metrics for the relative distance of the maximum absolute curvature and the "V-angle" derived from it show only minor correlation to other metrics but seem important from the rankings except for the *Relief* weight for  $aVCurv$ . Therefore, both metrics are kept in the feature set together with the actual value of the maximum of the absolute path curvature. When comparing metrics based on the centre line with those related to the inner track limits, there is a disagreement between both rankings as well. Regarding *Relief* weights, both metrics describing the relative distance of the minimum clearance to the respective reference line are ranked similar and most important. The value for this clearance is ranked higher in the apex variant, while the "V-angle" is ranked higher for the centre line, although not as significant. Concerning loss of accuracy, the relative distance of each variants minimum clearance is ranked almost least with a slight preference for the apex variant. The actual clearance value is ranked equally low, while the centre line variant is ranked higher. Both "V-angles" imply medium importance to the classification accuracy, the Apex variant is again ranked higher. This overall picture leads to keeping all three metrics for the apex variant.

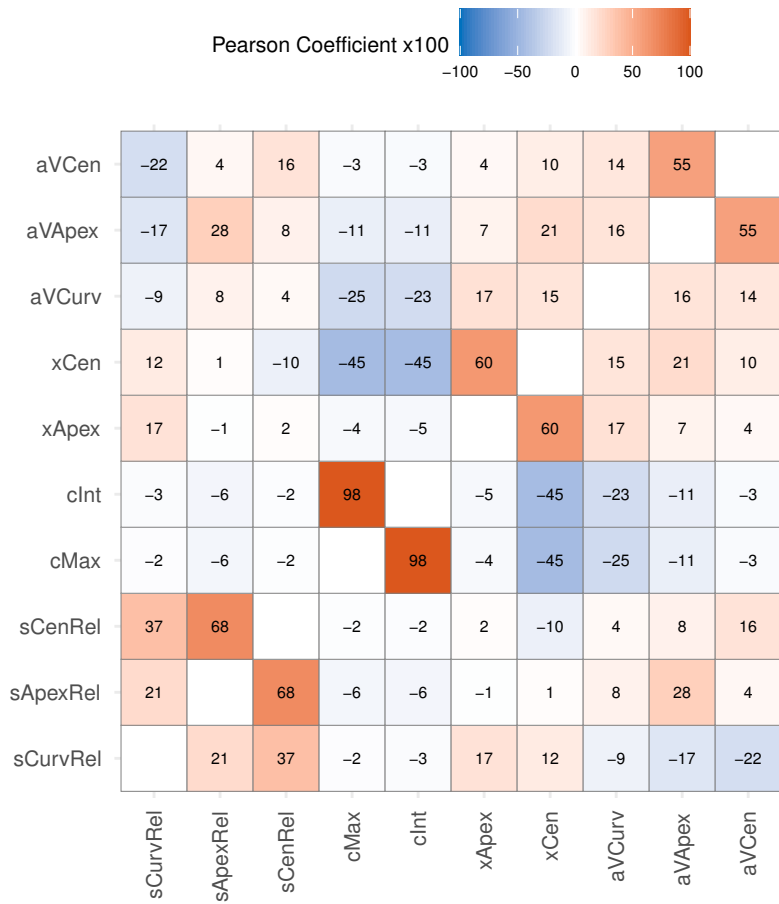


Figure 4.28: Correlation matrix for the reduced set of guidance level features

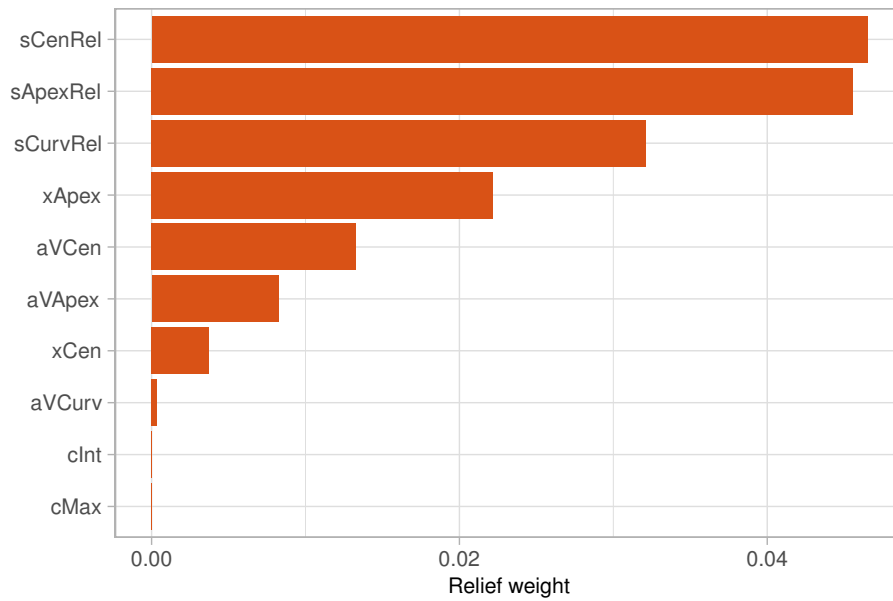


Figure 4.29: Relief weights for the reduced set of guidance level features

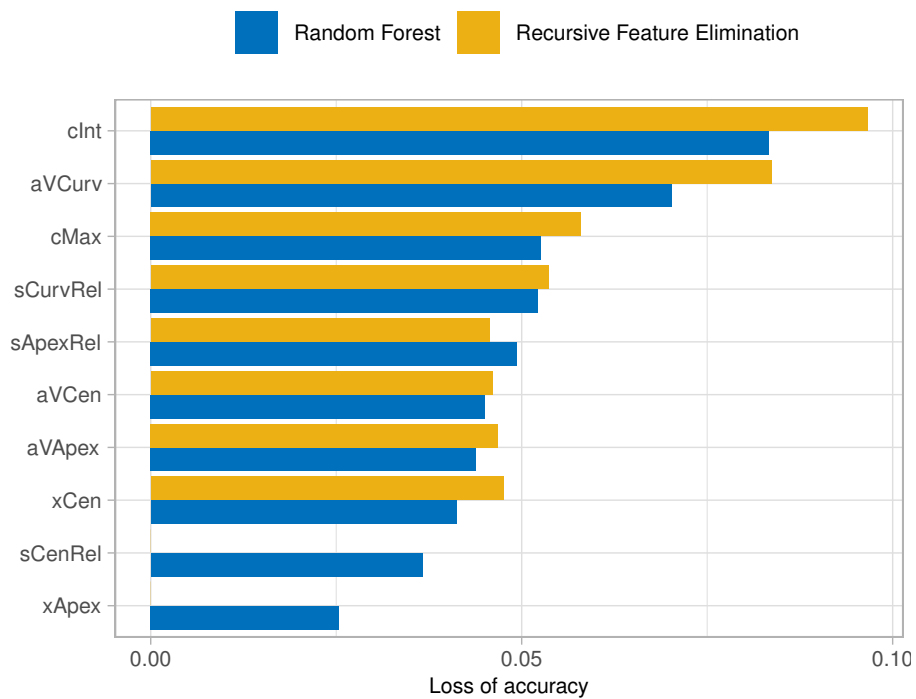


Figure 4.30: Loss of accuracy for the reduced set of guidance level features

### 4.3 Results

Following the detailed feature selection process, the final feature set is evaluated in this section. The selection process started with a total of 104 available metrics, which are used as features for the defined classification problem. These comprise 85 metrics focused on the control inputs of the driver, thus the stabilisation level of the driving task, and 19 calculated from the vehicle trajectory which is a description of the driving line and resembles the guidance level of the driving task. These were reduced to a total amount of 23 metrics, split into 17 and six metric for the respective levels of the driving task. The following section provides an overall comparison of classification accuracies. Detailed outcomes for selected metrics are presented afterwards. The finally selected metrics are listed below for a better overview.

#### Stabilisation level:

- *drThrOff*
- *drThrOn*
- *dpBrkOff*
- *dpBrkOn*
- *tBrkDlyAbs*
- *sThrOffRel*
- *sBrkRel*
- *sRollRel*
- *sTurnInRel*
- *sTrailRel*
- *vCarMax*
- *vCarMin*
- *rThrRmseLin*
- *pBrkRmseLin*
- *rd2ThrRmse*
- *rd2BrkRmse*
- *rd2StrRmse*

#### Guidance level:

- *sCurvRel*
- *sApexRel*
- *cMax*
- *xApex*
- *aVCurv*
- *aVApex*



| Data set              | OOB accuracy | Test set accuracy |
|-----------------------|--------------|-------------------|
| Control inputs (car)  | 75.5 %       | 76.5 %            |
| Control inputs (DiLS) | 75.1 %       | 76.4 %            |
| Trajectory (DiLS)     | 66.0 %       | 65.8 %            |
| All (DiLS)            | 76.7 %       | 76.9 %            |

Table 4.9: Classification accuracies for available data sets

As previously introduced, the data set comprises data logged during race weekends and laps from the DiLS, which allows comparing the vehicle path with sufficient precision. Unfortunately, this does not allow a comparison of control input specific metrics with vehicle trajectory based ones for the majority of the data set, although possible for the smaller amount of data obtained from the simulator. Each data set is split into 70 % training data and 30 % test data, which allows to evaluate the classification performance on unknown data. This approach would reveal an over fitting classifier that is essentially just remembering the training data. The classification accuracies for all three aforementioned data sets are listed in table 4.9. The values between training and test data are compliant for all data sets. The comparison indicates that the control inputs alone allow better classification than the trajectory with only a small bias by the significant difference in amount of observations used to train the classifier for car and DiLS data, respectively. The trajectory metrics alone show a significantly decreased accuracy. The lower performance when compared to control input metrics may result from the analysis of only one channel of information, namely the vehicle’s position, as opposed to three control input signals, which also results in a lower number of features to describe the guidance level. Finally, the combination of all selected features increases the classification accuracy to 77 % on the test set, which is the best overall value.

Table 4.10 lists the classification accuracies achieved during the feature selection process. Test set accuracies are not specifically listed but comply with differences between training and test sets shown in table 4.9. An interesting difference can be noticed for control input and trajectory related metrics. While the accuracy increases with the first reduction of features for the former, the accuracies of the latter decrease with eliminating features. The final step decreases the accuracy seen for a classifier trained with control input specific metrics by 2 %, which implies too many features have been eliminated to achieve the best classifier performance. This decrease is accepted in favour of a smaller amount of features.

The classification accuracy per driver can be read from the confusion matrix, which is displayed in Fig. 4.32. The confusion matrix shows the amount of observations predicted for each class over the respective true class. The values are percentages of the available data per true class. The main information is located on the diagonal, which is the amount of correct predictions per driver. Interestingly, the value is particularly high for

| Feature set                   | Control inputs (car) | Trajectory (DiLS) |
|-------------------------------|----------------------|-------------------|
| Full set OOB accuracy         | 76.1 %               | 69.7 %            |
| Intermediate set OOB accuracy | 77.2 %               | 66.6 %            |
| Final set OOB accuracy        | 75.5 %               | 66.0 %            |

Table 4.10: Classification accuracies for feature selection steps

drivers five and ten. This result is in line with empiric knowledge about both drivers, which have a very distinguishable *driving style*. Additionally, the remaining values show false classifications, whereas higher numbers indicate similar *driving styles*. Particularly drivers one and six are the only combination with a significant misclassification rate in both ways. For example, while 14 % of data from driver six is predicted to be driver eight, driver eight’s data is only classified as driver six by a share of 5 %.

As RFE was used throughout the feature selection process, there is also a curve of classification accuracy over number of features used for the final set of metrics. Figure 4.31 illustrates these for all previously mentioned subsets of available data. Interestingly, the control metrics for the DiLS data indicate a steeper initial rise in accuracy when compared to car data. The same can be seen for trajectory related metrics, although the gradient decreases earlier because the achieved accuracy is lower compared to the other subsets. The shapes of the curves also indicate that the best accuracy is achieved using all available features of the final set, with the exception of a classifier which is only trained on control metrics for DiLS data. In this case a higher prediction accuracy can be achieved by leaving out *pBrkRmseLin* and *rd2BrkRmse*. This indicates a subtle difference in the braking phase between DiLS and real car, which will be analysed further in this section.

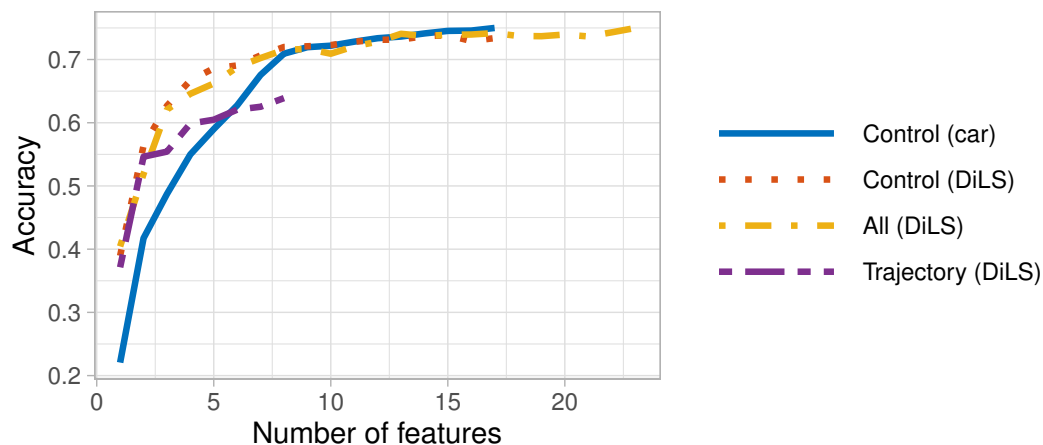
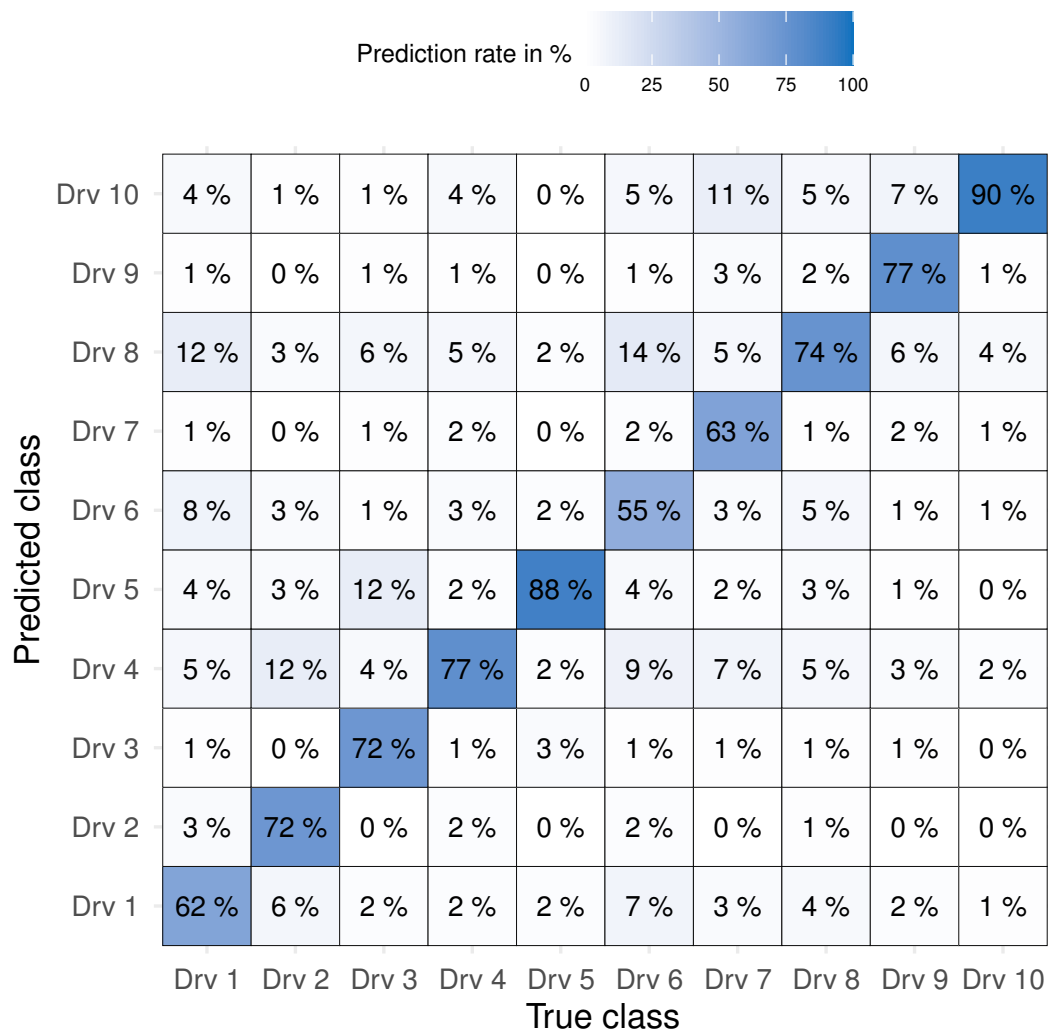
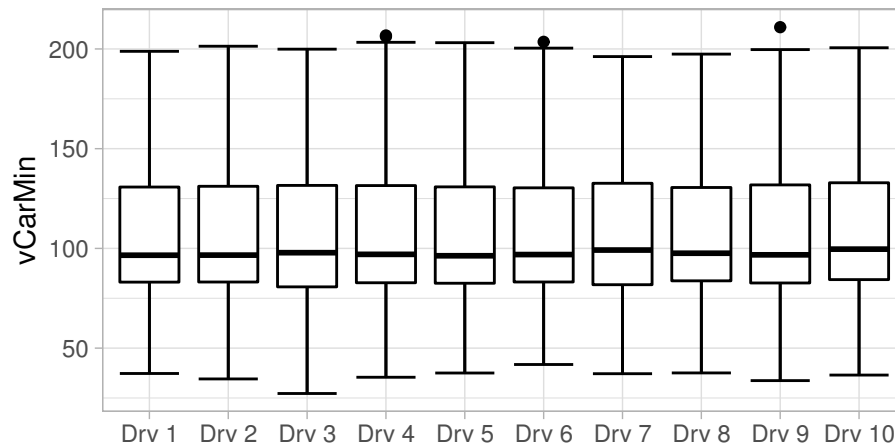
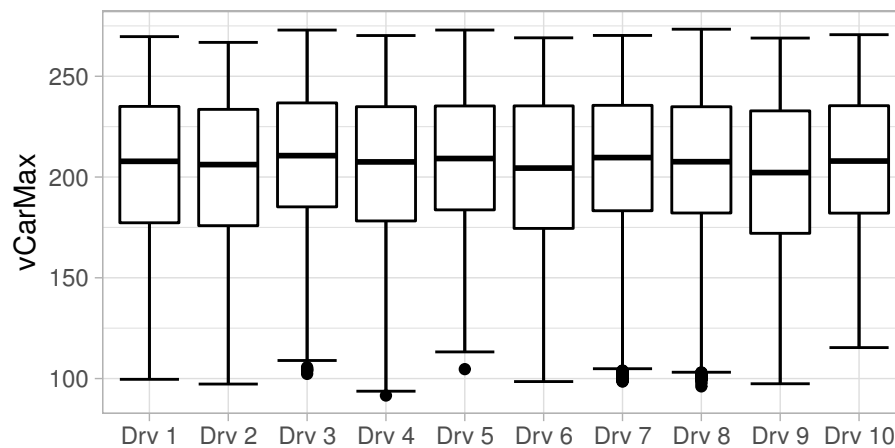


Figure 4.31: Classification accuracy over number of features used for final feature set

Figure 4.32: Confusion matrix for *Random Forest* trained with the final feature set

(a)  $vCarMin$ (b)  $vCarMax$ Figure 4.33: Box plots for  $v_C$  metrics

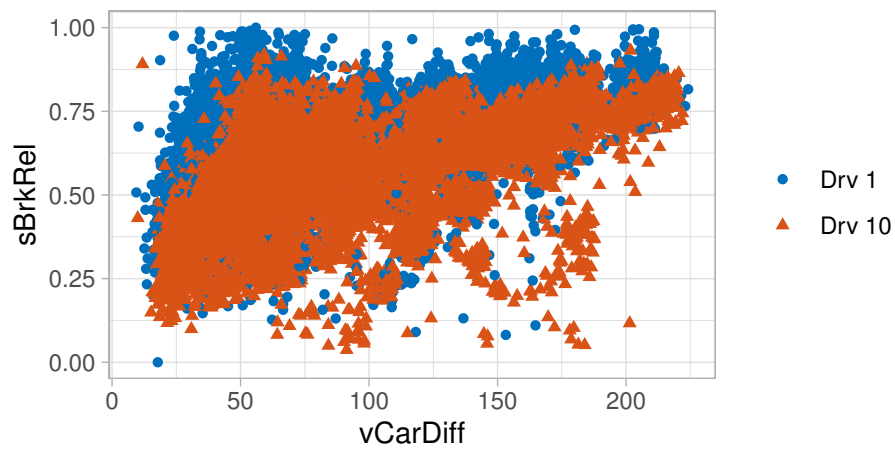
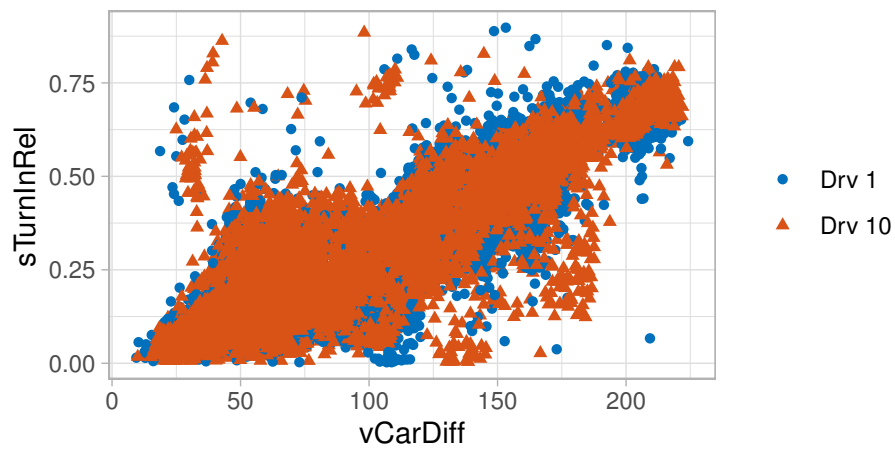
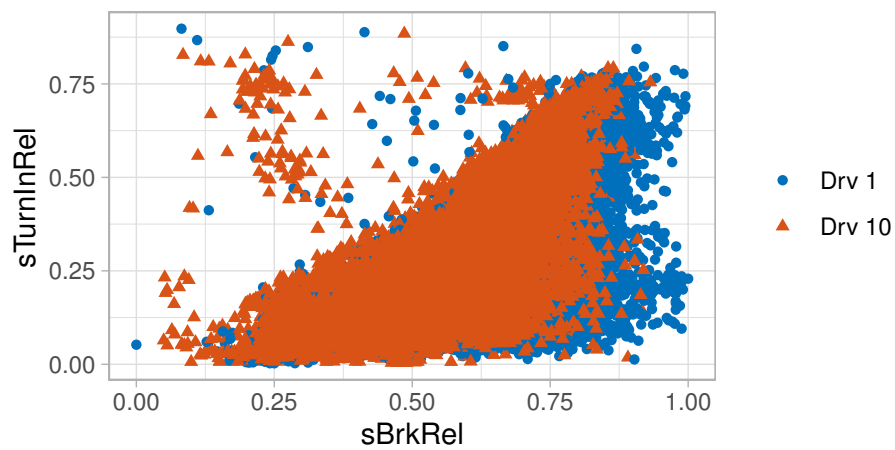
Some exemplary metrics are discussed in detail in the following. The results of the metric calculation are displayed in box plots [35]. The following box plots feature all drivers along the x axis with individual boxes per driver. The box itself is defined by the first and third quartile of the data, the bar inside the box marks the second quartile or median. Additionally, the range of data is shown with whiskers. Those extend up to 1.5 times the interquartile range. Data points out of this range are subsequently considered as outliers and marked by individual dots.

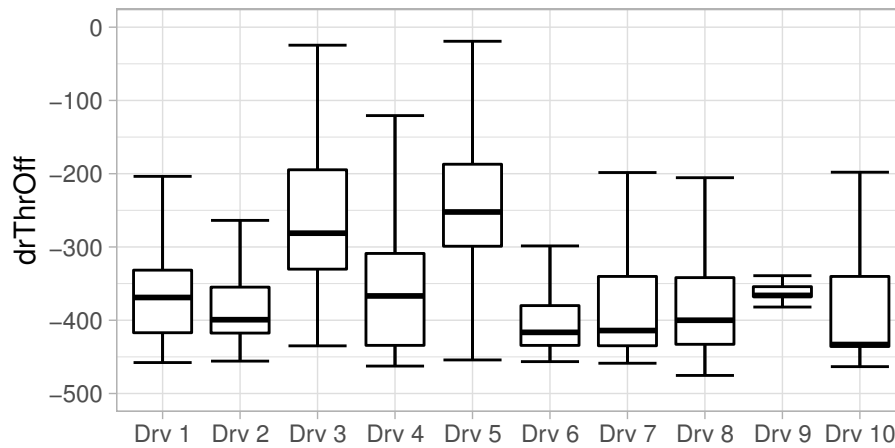
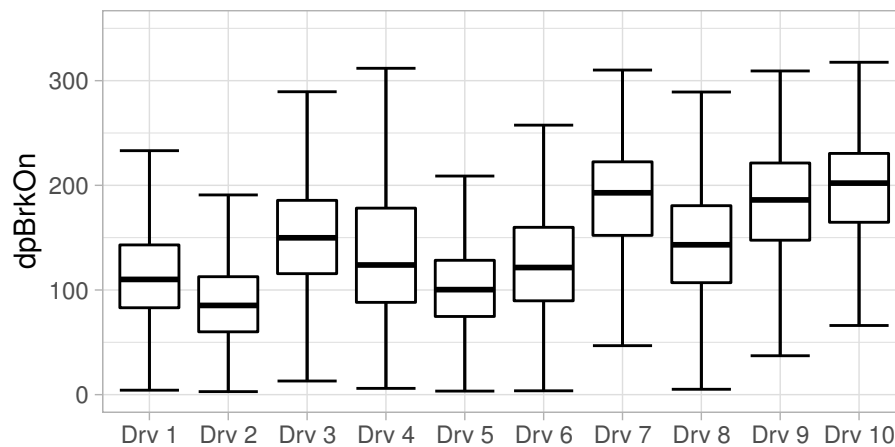
The first two metrics,  $vCarMin$  and  $vCarMax$ , as displayed in Fig. 4.33, are defined mainly to preserve information about the individual corners. According to the box plots no significant differentiation between drivers is possible using these two metrics, yet the used feature selection methods suggested. This indicates that the information becomes relevant to improve other metrics.

The aforementioned dependency between metrics and corner type is detailed by two exemplary metrics. Both  $sBrkRel$  and  $sTurnInRel$  are expected to be affected by the corner type. For easier visualisation  $vCarDiff$  is used although it was eliminated from the feature set due to redundancy with the minimum and maximum velocity. Figure 4.34 (a) shows the length spent under braking relative to the corner length over car velocity difference for two different drivers. As expected, there is a linear trend showing that the relative amount of braking rises with increasing speed difference. Additionally, a similar trend is shown in Fig. 4.34 (b) for the *turn-in* point relative to the corner length. Since the braking phase takes longer with higher velocity difference, also the *turn-in* point is expected to happen later. Figure 4.34 (c) illustrates the relation of both discussed metrics and reveals that later *turn-in* points go in line with a longer braking phase, however the reverse is not true. According to the data points, a long braking phase can coincide with a broad range of *turn-in* points, however a short braking phase also implies an early *turn-in* point. All three graphs show some outliers which are of minor importance given the large amount of observations.

Some of the defined metrics describe the driver's *driving style* and also have a direct connection to the driver's performance. Figures 4.35 and 4.36 show box plots of  $drThrOff$  and  $dpBrkOn$  respectively, with outliers removed for better visibility. Both the throttle lift-off and the application of brake pressure happen at the beginning of the corner, and share the characteristic that a steeper gradient usually results in better performance, because the desired value is reached quicker. Attention needs to be paid to the difference in sign, where  $drThrOff$  should be as low as possible and  $dpBrkOn$  the opposite. Driver ten, who showed outstanding classification accuracy, has the best median values respective to both metrics. Opposing to this, driver five with almost equally good classification results, is slowest to release the throttle apparent both from the median and the first quartile as well. This is in line with a shallow gradient on braking relative to other drivers. Overall, both metrics show significant differences among the drivers regarding median value and interquartile range, which can be interpreted as the precision of executing a repeated task.

The aforementioned difference in the braking phase is also visible in the delay between throttle lift-off and brake application, as captured by the metric  $tBrkDlyAbs$ . A box plot for observations obtained from real cars is shown in Fig. 4.37. Driver ten features the smallest interquartile range, which complies again with the higher classification accuracy. Overall, there is a significant spread of median values as well as interquartile ranges among the drivers. Noticeable as well is the fact that the majority of observations is negative for each driver, with only values above the third quartile reaching positive numbers. This implies that the drivers press the brake pedal before lifting off the throttle pedal. A possible explanation for this is compliance in the brake system of the real car. The throttle lift-off results in almost immediate negative acceleration caused by engine drag. Contrary, the brake pressure builds up gradually before the car is decelerated. Two exemplary reasons for this behaviour are deflection of brake lines, and the brake pads that need to cover some clearance before making contact with the

(a)  $sBrkRel$  over  $vCarDiff$ (b)  $sTurnInRel$  over  $vCarDiff$ (c)  $sTurnInRel$  over  $sBrkRel$ Figure 4.34: Dependencies to  $v_C$

Figure 4.35: Box plot for  $nOffThr$  metricFigure 4.36: Box plot for  $nOnBrk$  metric

brake disc, which both reduce the brake system's stiffness. Opposing this behaviour, the simulation model does not contain any brake compliance and also the used hardware is significantly less compliant. This can be seen immediately in a similar box plot for DiLS data in Fig. 4.38. This plot shows only positive values for all four drivers and captures the adjustment of brake pedal application, which is only  $0.05\text{ s}$  for driver eight and up to  $0.075\text{ s}$  for driver five. The characteristics seen for the real car in Fig. 4.37 however are also present in DiLS data. Drivers eight and seven have smaller interquartile ranges indicating better precision with brake pedal application. Driver five has a significantly larger range in both data sets. The individual differences are seen in both data sets, as well as the slight difference in *driving style* by all drivers on the DiLS.

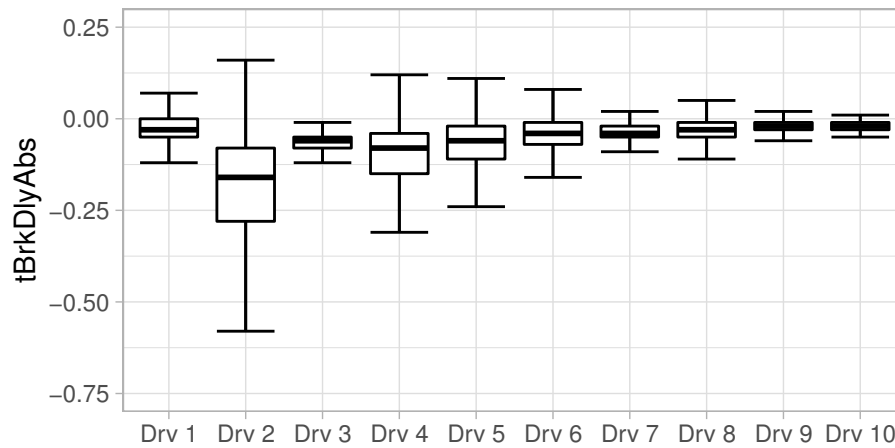


Figure 4.37: Box plot for tBrkDlyAbs metric with car data

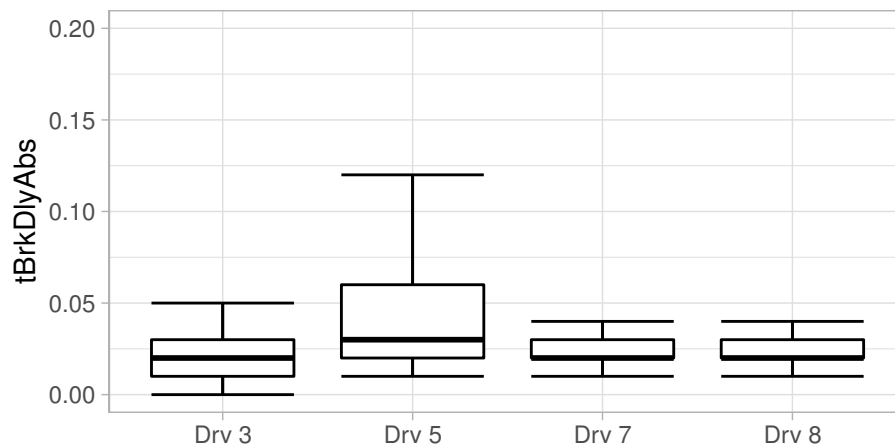
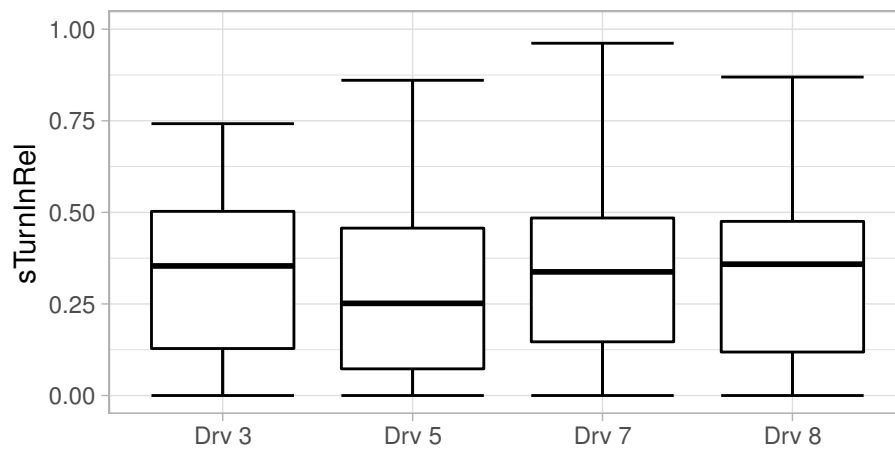
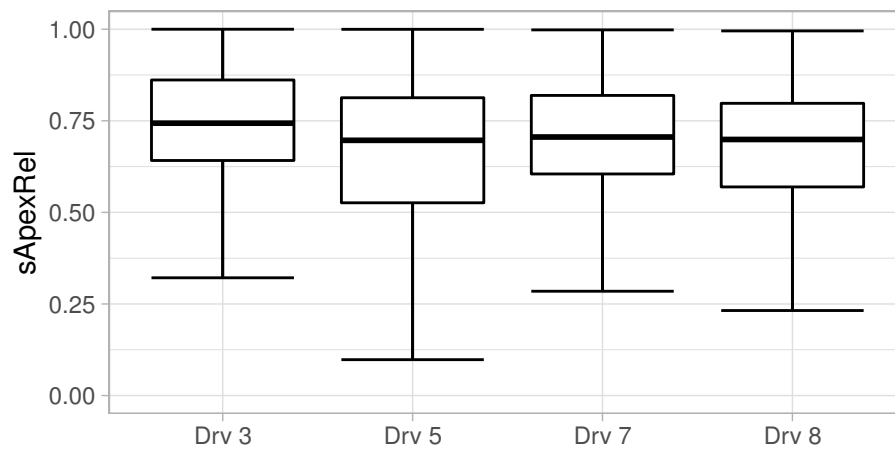
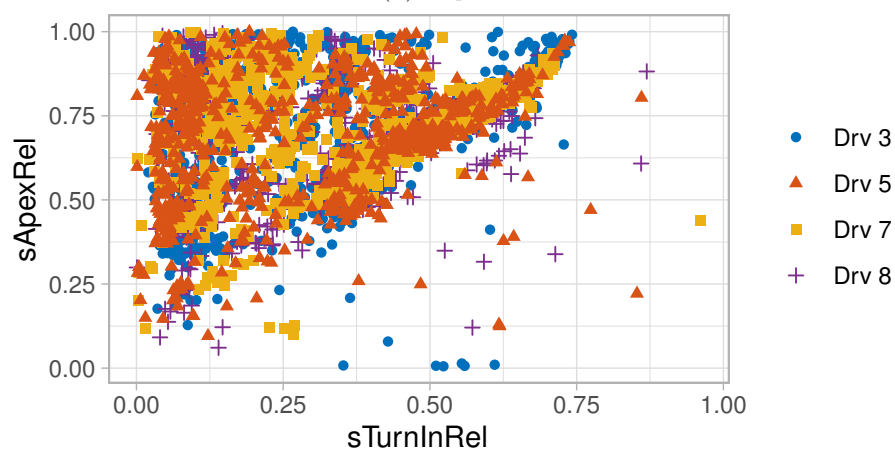


Figure 4.38: Box plot for tBrkDlyAbs metric with DiLS data

Another example of relation between metrics can be observed in Fig. 4.39, which shows box plots for  $sTurnInRel$  in subfigure (a) and  $sApexRel$  in subfigure (b). A scatter plot of both metrics is displayed in subfigure (c). Before analysing the results, a dependency of both metrics is assumed. This is due to the idea that a later *turn-in* will lead to a later apex as well. The box plots indicate a similar comparison of median values with both metrics for drivers three and five, however especially driver eight generally shows a later *turn-in* point in combination with a comparatively early apex. This does not comply with the assumption and is further illustrated by subfigure (c). The points form a triangular cloud, which means additional to the assumed proportionality also a late apex with early *turn-in* is employed by drivers. A stronger tendency for driver eight to use this *driving style* is apparent from the box plots. The opposite, meaning an early apex with late *turn-in* is not of importance, judged from the data and from an engineering point of view as well.



(a)  $sTurnInRel$ (b)  $sApexRel$ (c)  $sTurnInRel$  over  $sApexRel$ Figure 4.39: Box plots for  $sTurnInRel$  and  $sApexRel$  metrics

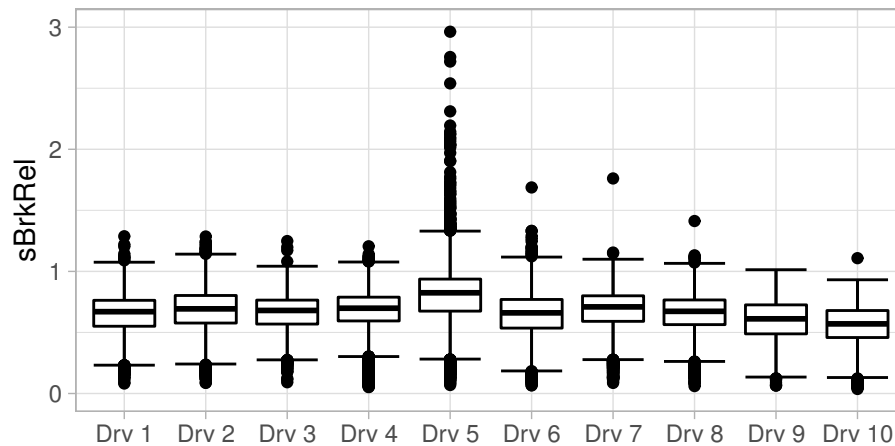


Figure 4.40: Box plot for *sBrkRel* metric from car data

The final example of the results for the data analysed within the scope of this work is the relative length of the braking phase *sBrkRel*. Figure 4.40 shows a box plot for this metric including outliers, since they are of importance in this graph. It is apparent that driver five has the biggest share of outliers, which range out of the desired scope of values between zero and one. A value above one requires a detected braking phase that is longer than the interval from throttle lift-off to coming back at full throttle. While generally a few data points with wrongly detected corner points and resulting outliers are expected, driver five seems to particularly not comply with the used strict pattern detection algorithm. Overall, this is also treated as a driver characteristic. This result needs careful observation though, as the metric will gain importance for the classifier used for feature selection, although this is caused mostly by a single driver. This can also be seen in the high rating for loss of accuracy in Fig. 4.21. The metric does however allow differentiation of the remaining drivers as well, which is supported by the *Relief* weights shown in Fig. 4.20. The ranking from *Relief* weights is not as high as for loss of accuracy, although it still implies significant feature importance.

## 4.4 Discussion

The main scope of this work is the *driving style* of race car drivers. With reference to chapter 1 this research originates from the problem of driver specific vehicle setups in motorsports. Since both the driver and the car show dynamic behaviour that is also mutually influenced, tuning the overall system proves difficult. The main idea to solve this problem is an objective description of each subsystem that may subsequently allow evaluation of the interaction of those with each other. This calls for an objective measure for *driving style*, which is specifically suited to the application for race car drivers.

**Hypothesis 1:** Race car drivers show individual differences in *driving style*.

The first hypothesis claims that race car drivers show distinct *driving styles*. This assumption is based on empirical observations by many experts in the field, but has not been scientifically proven yet. The review of the state of the art revealed many promising concepts about characterising *driving styles* and mostly classifying drivers into certain predefined categories. Due to the variety of driver behaviours found, it seems reasonable to assume that these differences also exist for racing drivers. The main limitation though is the general setting in which a driver finds himself. Normal usage of road cars in cities and on rural roads or highways allows a variety of approaches. This is also resembled in the found driver categories, such as cautious or aggressive. Contrary, the purpose of race car driving is solely focused on achieving minimal lap time, which imposes a certain approach to the driving task. This subsequently leads to the need of a more detailed description to elaborate if distinct *driving styles* can be found for race car drivers. An example for a metric used in the literature to classify driver behaviour is the average vehicle velocity. In the car racing domain the average speed is similar to the achieved lap time, since the track length is a boundary condition. This in turn means that the average car speed is a performance metric in motorsports. If it were suited to differentiate *driving style* in this context, that would impose all drivers capable of accomplishing the same lap time, which includes all drivers considered in this work, would share the same *driving style*. The examination of the first hypothesis is inherently linked to the second hypothesis because metrics are needed to analyse possible differences between drivers. While these metrics are discussed in context of the second hypothesis, the first hypothesis can already be confirmed from the shown results. The presented data clearly shows differences in *driving style* among the participating drivers.

**Hypothesis 2:** The *driving style* of race car drivers can be described by objective criteria.

Based on the assumption that *driving style* is differentiable among race car drivers, and the available literature on respective methods for road car users, this hypothesis reflects the main goal of this work. As previously mentioned, the goal differs from the presented existing solutions, which are capable of classifying drivers into certain groups and even distinguish racing drivers, though into beginner and expert categories. For this reason, the presented approach was evaluated. The analysis of the available data on a corner

by corner basis proved to make the required level of detail available and increase the amount of usable data points compared to a lap or outing basis. The verification of the hypothesis that the *driving style* of race car drivers can be described by objective metrics was accomplished through a driver detection algorithm that is using the aforementioned objective metrics as inputs to detect the driver for a certain corner. While the goal is not to detect the driver in the end, the ability to do so proves both that the defined criteria contain information which allows differentiation of race car drivers, as well as the implicit presence of different *driving styles*. The detection could be achieved with reasonable accuracy. It must be noted though, that the level of correct predictions can be significantly increased. An exemplary method to do so is to set the limitation of allowing only one driver class for a set of corners on the same lap. This approach is in accordance with the condition that the driver does not change over a lap, however is examined in future work and not further presented here.

The employed machine learning techniques also contributed to evaluate the significance of the aforementioned metrics. Multiple ranking mechanisms have been employed, which proved to be useful given the sometimes ambiguous results of the evaluation. The combination of different methods allowed to build an overview, as well as an estimate of how useful these indications are to evaluate each of the defined *driving style* characteristics. Finally, engineering know how helped to select metrics in a consistent way, which aids any future application of the method. An example for this process is the selection of distance based metrics measured relative to the respective corner. These were ranked highest for classification accuracy in most cases, however few would need to be replaced with their correlated variants to optimise the prediction accuracy. As elaborated, this was not the main scope of this work. Furthermore, this example emphasises that results must be considered with care when employing undeniably powerful machine learning methods.

The finally selected 23 metrics show promising results, not only because the driver can be detected from them but also for the distinct distribution of results for individual drivers. The ability to classify race car drivers based on the elaborated metrics proves that they show different *driving styles* and that these can be described objectively. This means that both hypotheses are proven by the present work. Since the evaluation was carried out with data from three race seasons, this also means that the approach works with changing conditions. With reference to Fig. 1.1, especially changing dynamic behaviour of the vehicle and different tracks are the key components besides the driver. This was also verified for the Driver-in-the-Loop Simulator (DiLS). Of course this application is highly dependent on the used vehicle model as well as simulator hard- and software. However, if the simulation comprises the required level of detail, the obtained results show that the drivers display similar differentiation as in the real car. This is illustrated by the brake delay metric, which is the delay between throttle lift-off and the application of the brake pedal. Despite the change of absolute values, which even changed sign, the relative differences among the drivers can be seen data from the car and from the DiLS alike. In summary, the presented results show that the developed method to objectively describe the *driving style* of race car drivers is robust and provides significant results.

## 5 Summary and Outlook

The following chapter summarises the present thesis regarding its motivation, the derived methodology, and the achieved results. Additionally, it aims to indicate some possibilities for further research and application possibilities building on the presented outcomes.

The scope of this work was entirely in the motorsport context, which is built around the main goal to achieve minimal lap time or maximal distance for a combination of driver, vehicle, and track. The aforementioned three components are closely linked and interact with each other during the process. In fact they form a dynamic system which needs to be understood if further optimisation is required. The presented work focuses on the interaction of the driver with the vehicle while the track was simplified to a constant boundary condition. The resulting engineering problem is consequently about setting up a race car specifically for a given circuit, and equally important, in a way that suits the driver's individual *driving style*. As the vehicle offers a non-trivial amount of parameters for adjustment the mainly used empiric process to understand a driver's needs and consequently the requirements concerning the car, as well as the corresponding setup characteristics, is time consuming. Therefore, the aim of this research was to provide a detailed description of a race driver's *driving style*, which can be understood as the driver's dynamic behaviour. This is the first step towards a comprehensive, objective understanding of the aforementioned dynamic system involved in motor racing comprising driver, car, and track. The initial hypotheses, which were proven in this work, state that individual *driving styles* exist between professional race car drivers, as well as that they can be described by objective metrics.

Following the introduction to the topic, a review of available literature concerning the driving task itself, driver modelling, and *driving style* was presented. The commonly used three-level model of the driving task was summarised, since especially the stabilisation and guidance level are used by the derived method. Driver modelling was described by exemplary works, including the principal concept of a virtual driver as well as methods to incorporate driver behaviour to a certain extent. This part is mainly about setting the general goal of a driver which would be time-optimal in the case of race car driving. Subtle details as expected for different *driving styles* in the scope of this work are however not yet considered in the literature. Additionally, selected works about *driving style* were presented mainly for road car users. These include different methods of evaluation, for example by analysing recorded vehicle data or psychological questionnaires. A tendency for classification of drivers into several categories was found. This can be used for various topics, for Advanced Driver Assistance Systems (ADAS) or for insurance policies, just to name two. The characterisation with general terms, such as cautious or aggressive, does

conversely not allow differentiation of race car drivers. This led to the development of a new method to provide an objective description of the *driving style* of race car drivers.

The proposed method relies on specific patterns of the driver's control inputs to the vehicle which are especially present in motorsports. If the car is driven on the limit, the usual approach to a corner shows distinct characteristics in the recorded signals. For example, the throttle is lifted from full throttle to idle and carefully applied again up to full throttle after the *braking* phase and *turn-in*. The detection of these patterns, or corner points, was implemented through rules defined by a margin around the static signal values. With the detected characteristics objective metrics can be calculated such as the *on-throttle* or *turn-in* point. The approach was split into the stabilisation and guidance level, according to the aforementioned three-level model of the driving task. Metrics to describe the *driving style* on the stabilisation level are calculated from the main control inputs which are throttle, brake, and steering. These include gradients of signals between the detected points, relative positions of points, and lengths of intervals between those points in context of the analysed corner. Furthermore, a description of the signals in the frequency spectrum as well as analysis of the derivatives of each signal is included. The characterisation of the *driving style* on the guidance level employs the vehicle's position as a representation of the driving line. It is described by metrics based on the path curvature, the distance to absolute reference lines along the corner, and "V-angles" as an objective implementation of a term commonly used by engineers to describe the approach to a corner. The aforementioned reference lines are either the track centre line or the inner track edge of the corner.

The definition of objective metrics resulted in 104 characteristics, 85 for the driver's control inputs and 19 for the description of the chosen driving line. These were evaluated regarding correlations and significance with the goal to reduce the set to an amount that allows practical usage but still contains the necessary information. Machine learning techniques and feature selection methods were used for this purpose. The proposed problem of driver detection served for this selection and also to generally proof the existence and differentiability of *driving style* in the motorsport context. Supervised learning was applied to detect the driver from the metrics that have been calculated for a specific corner. The used *Random Forest* classifier is combined with feature selection methods, such as Recursive Feature Elimination (RFE), and ranking criteria like the *Pearson* correlation coefficient and the *Relief* algorithm. The defined metrics were reduced to 49 in a first step, 39 for control inputs and 10 for the driving line. This was done on the basis of removing those with too many erroneous values, as well as reducing the intentionally redundant features. In a second step the remaining features were compared not only respective to correlated ones but also regarding their absolute significance. This led to a feature set with 23 metrics, 19 for the stabilisation and six for the guidance level.

The classification algorithm trained on these 23 selected features achieves a detection rate of 77% proving that substantial information about individual drivers is contained in the objective metrics. Detailed analysis of the results showed distinctive patterns for different drivers and allowed further investigations regarding their interactions. An

exemplary comparison of the same metric for real car and simulator data revealed that the same driver specific patterns can be seen even though the absolute values change with the different environment. The results confirmed the original hypotheses. Different *driving styles* can be seen among race car drivers competing on the same level and in the same category. These can be characterised by objective metrics.

While rating drivers with the derived approach is outside the scope of this thesis, some of the objective metrics are directly linked to driver performance. The gradient of the throttle release, as discussed in section 4.3, is an example for such a metric. Despite the mentioned link to driver performance, a more detailed analysis using the gathered data could potentially reveal more information about this topic. Further studies could lead to recommendations for specific driving styles depending on track, car type, and other conditions. An adoption of the approach to a corner by the driver possibly improves performance more than adjustments to the car.

Apart from describing the drivers the next step to achieve the overall goal is to describe the dynamic vehicle behaviour by a similar objective approach. There are various examples available in the literature for road cars, but an adoption to the different conditions in racing is not yet published. The main difference is that the state of the vehicle is mainly close to the driving limits of a car, as opposed to the evaluation of road cars which is typically focused on the linear region. Following such a description of the vehicle, the understanding of both systems allows investigation of the interactions between them. This could provide information about the driver individual subjective perception of *driveability* if objective methods are combined with the driver's feedback.

The presented research is an important step towards a systematic understanding of the complex interactions between driver, vehicle, and environment in motor racing. It will significantly contribute to the future goal of replacing empirical tuning of the vehicle setups in training sessions by virtual methods for time- and cost effective development in motorsports.

# List of Figures

|      |   |    |
|------|---|----|
| 1.1  | Dynamic system overview . . . . .   | 2  |
| 2.1  | Three-level model of the driving task [10] . . . . .  | 7  |
| 3.1  | Classification of vehicle states . . . . .  | 16 |
| 3.2  | Throttle, brake pressure and steer wheel angle signals with detected extreme value intervals . . . . .                                | 18 |
| 3.3  | Throttle, brake pressure and steering wheel angle signals with detected characteristic points . . . . .                               | 22 |
| 3.4  | Exemplary brake signal with linear approximation for the <i>off-brake</i> phase, first and second derivative . . . . .                | 31 |
| 3.5  | Exemplary throttle signal in time and frequency domain . . . . .  | 36 |
| 3.6  | Exemplary circle for curvature calculation . . . . .  | 43 |
| 3.7  | Exemplary driving lines and track definition . . . . .  | 45 |
| 3.8  | Track outlines for the Lausitzring in DTM configuration . . . . .   | 46 |
| 3.9  | Exemplary paths and distance metric illustration . . . . .  | 47 |
| 3.10 | Exemplary signals for curvature, distance to inner track limit, and track centre line with points for "V-angle" calculation . . . . . | 49 |
| 3.11 | Exemplary feature subsets for four features . . . . .   | 57 |
| 3.12 | Data points of the iris dataset [14] . . . . .  | 58 |
| 3.13 | <i>Classification Tree</i> for the iris dataset [14] . . . . .  | 59 |
| 3.14 | Feature importances for <i>Random Forest</i> trained on the iris dataset [14] . . . . .   | 60 |
| 4.1  | Driver-in-the-Loop Simulator ©BMW AG . . . . .  | 64 |
| 4.2  | Distribution of car data . . . . .  | 65 |
| 4.3  | Distribution of DiLS data . . . . .   | 66 |
| 4.4  | Distribution of car data per track . . . . .  | 67 |
| 4.5  | Distribution of car data sampled equally per track . . . . .  | 67 |
| 4.6  | Distribution of car data sampled equally per track and driver . . . . .   | 67 |
| 4.7  | Loss in accuracy per feature for all three data sets . . . . .  | 68 |
| 4.8  | Correlation matrix for objective metrics describing the stabilisation level of the driving task . . . . .                             | 71 |
| 4.9  | Correlation matrix for correlated time and distance based metrics . . . . .   | 72 |
| 4.10 | Relief weights for correlated time and distance based metrics . . . . .   | 74 |
| 4.11 | Loss of accuracy for correlated time and distance based metrics . . . . .   | 74 |
| 4.12 | Relief weights for correlated derivative based metrics . . . . .  | 75 |
| 4.13 | Correlation matrix for correlated derivative based metrics . . . . .  | 76 |



---

|      |  |     |
|------|--|-----|
| 4.14 | Relief weights for correlated FFT based metrics . . . . .  | 77  |
| 4.15 | Correlation matrix for correlated FFT based metrics . . . . .  | 78  |
| 4.16 | Correlation matrix for reduced feature set regarding stabilisation level . . . . .                   | 79  |
| 4.17 | Relief weights for reduced feature set regarding stabilisation level . . . . .                       | 80  |
| 4.18 | Loss of accuracy for reduced feature set regarding stabilisation level . . . . .                     | 81  |
| 4.19 | Correlation matrix for distance relative features of the reduced set . . . . .                       | 83  |
| 4.20 | Relief weights for distance relative features of the reduced set . . . . .                           | 84  |
| 4.21 | Loss of accuracy for distance relative features of the reduced set . . . . .                         | 84  |
| 4.22 | Relief weights for FFT related features of the reduced set . . . . .                                 | 85  |
| 4.23 | Loss of accuracy for FFT related features of the reduced set . . . . .                               | 85  |
| 4.24 | Correlation matrix for FFT related features of the reduced set . . . . .                             | 86  |
| 4.25 | Correlation matrix for objective metrics describing the guidance level of the driving task . . . . . | 88  |
| 4.26 | Relief weights for correlated distance and time based metrics . . . . .                              | 89  |
| 4.27 | Loss of accuracy for correlated distance and time based metrics . . . . .                            | 90  |
| 4.28 | Correlation matrix for the reduced set of guidance level features . . . . .                          | 91  |
| 4.29 | Relief weights for the reduced set of guidance level features . . . . .                              | 92  |
| 4.30 | Loss of accuracy for the reduced set of guidance level features . . . . .                            | 92  |
| 4.31 | Classification accuracy over number of features used for final feature set . . . . .                 | 95  |
| 4.32 | Confusion matrix for <i>Random Forest</i> trained with the final feature set . . . . .               | 96  |
| 4.33 | Box plots for $v_C$ metrics . . . . .  | 97  |
| 4.34 | Dependencies to $v_C$ . . . . .  | 99  |
| 4.35 | Box plot for nOffThr metric . . . . .  | 100 |
| 4.36 | Box plot for nOnBrk metric . . . . .   | 100 |
| 4.37 | Box plot for tBrkDlyAbs metric with car data . . . . .   | 101 |
| 4.38 | Box plot for tBrkDlyAbs metric with DiLS data . . . . .  | 101 |
| 4.39 | Box plots for <i>sTurnInRel</i> and <i>sApexRel</i> metrics . . . . .                                | 102 |
| 4.40 | Box plot for <i>sBrkRel</i> metric from car data . . . . .   | 103 |

# List of Tables

|      |  |    |
|------|--|----|
| 3.1  | Exemplary values for static throttle signal calculation . . . . .                    | 17 |
| 3.2  | Description of characteristic points for detecting cornering of a race car           | 19 |
| 3.3  | List of gradient, time and distance based metrics on the stabilisation level         | 30 |
| 3.4  | List of additional metrics on the stabilisation level . . . . .                      | 35 |
| 3.5  | List of frequency based objective metrics on the stabilisation level . . .           | 41 |
| 3.6  | List of objective metrics for the guidance level . . . . .                           | 53 |
| 3.7  | Accuracy estimations for <i>Classification Tree</i> and <i>Random Forest</i> . . . . | 59 |
|      |  |    |
| 4.1  | Summary of available data . . . . .  | 65 |
| 4.2  | Number of observations in reduced datasets . . . . .                                 | 66 |
| 4.3  | Number of NaN observations for control metrics . . . . .                             | 70 |
| 4.4  | Average correlation of distance and time based metrics . . . . .                     | 73 |
| 4.5  | Average correlation of derivative based metrics . . . . .                            | 75 |
| 4.6  | Average correlation of FFT based metrics . . . . .                                   | 77 |
| 4.7  | Number of NaN observations for control metrics on the guidance level .               | 87 |
| 4.8  | Average correlation of distance and time based metrics . . . . .                     | 87 |
| 4.9  | Classification accuracies for available data sets . . . . .                          | 94 |
| 4.10 | Classification accuracies for feature selection steps . . . . .                      | 95 |

## Bibliography

- [1] Edoardo Amaldi and Viggo Kann. “On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems”. In: *Theoretical Computer Science* 209.1 (1998), pp. 237–260. ISSN: 0304-3975.
- [2] A. Apel and M. Mitschke. “Adjusting vehicle characteristics by means of driver models”. In: *Int. J. of Vehicle Design* 18 (1997).
- [3] Leo Breiman. *Classification and regression trees*. New York, N.Y.: Chapman & Hall, 1984. ISBN: 9781351460491.
- [4] Leo Breiman. “Random Forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32. ISSN: 0885-6125.
- [5] Richard H. Byrd, Jean Charles Gilbert, and Jorge Nocedal. “A trust region method based on interior point techniques for nonlinear programming”. In: *Mathematical Programming* 89.1 (2000), pp. 149–185. ISSN: 1436-4646.
- [6] Mu-Song Chen et al. “Driving behaviors analysis based on feature selection and statistical approach: a preliminary study”. In: *The Journal of Supercomputing* 129.6 (2018), pp. 459–479. ISSN: 0920-8542.
- [7] Z. Constantinescu, C. Marinoiu, and Vladioiu M. “Driving Style Analysis Using Data Mining Techniques”. In: *International Journal of Computers, Communications & Control* 5.5 (2010), pp. 654–663. ISSN: 1841-9836.
- [8] Sanjoy Das et al. “A knowledge based model of traffic behavior in freeways”. In: *Proceedings of the 1999 ACM symposium on Applied computing - SAC '99*. Ed. by Barrett Bryant et al. New York, USA: ACM Press, 1999, pp. 14–18. ISBN: 1581130864.
- [9] Edmund Donges. “A Two-Level Model of Driver Steering Behavior”. In: *Human Factors* 20.6 (1978), pp. 691–707. ISSN: 0018-7208.
- [10] Edmund Donges. “Aspekte der Aktiven Sicherheit bei der Führung von Personenkraftwagen”. In: *Automobil Industrie* 27.2 (1982), pp. 183–190. ISSN: 0005-1306.
- [11] Edmund Donges. “Ein regelungstechnisches Zwei-Ebenen-Modell des menschlichen Lenkverhaltens im Kraftfahrzeug”. In: *Zeitschrift für Verkehrssicherheit* 24.3 (1978). ISSN: 0044-3654.
- [12] D. Dörr, D. Grabengiesser, and F. Gauterin. “Online driving style recognition using fuzzy logic”. In: *17th International IEEE Conference on Intelligent Transportation Systems*. 2014, pp. 1021–1026. ISBN: 978-1-4799-6078-1.

- 
- [13] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository*. 2017. URL: <http://archive.ics.uci.edu/ml>.
- [14] Ronald A. Fisher. “THE USE OF MULTIPLE MEASUREMENTS IN TAXONOMIC PROBLEMS”. In: *Annals of Eugenics* 7.2 (1936), pp. 179–188. ISSN: 2050-1420.
- [15] Matteo Frigo and Steven G. Johnson. “FFTW: An adaptive software architecture for the FFT”. In: *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing*. Vol. 3. 1998, pp. 1381–1384. ISBN: 0-7803-4428-6.
- [16] Y. Fujiwara et al. “Control design of driver support system using multiple driver models”. In: *Proceedings of the SICE 2004 Annual Conference*. Ed. by Jidō Keisoku and Gakkai Seigyo. Vol. 3. Tokyo, Japan and Piscataway, NJ: Society of Instrument and Control Engineers (SICE), 2004, pp. 2443–2448. ISBN: 4-907764-22-7.
- [17] Jean Dickinson Gibbons and Subhabrata Chakraborti. *Nonparametric Statistical Inference*. 4th ed. Vol. 168. Statistics, textbooks and monographs. New York: Marcel Dekker, 2003. ISBN: 0-8247-4052-1.
- [18] Florian Goy et al. “Development of objective criteria to assess the vehicle performance utilized by the driver in near-limit handling conditions of racecars”. In: *16. Internationales Stuttgarter Symposium*. Ed. by Michael Bargende, Hans-Christian Reuss, and Jochen Wiedemann. Proceedings. Wiesbaden: Springer Vieweg, 2016, pp. 1213–1231. ISBN: 978-3-658-13254-5.
- [19] Massimo Guiggiani. *The Science of Vehicle Dynamics: Handling, Braking, and Ride of Road and Race Cars*. 2nd ed. 2018. Cham: Springer International Publishing, 2018. ISBN: 978-3-319-73220-6.
- [20] Isabelle Guyon and André Elisseeff. “An Introduction to Variable and Feature Selection”. In: *Journal of Machine Learning Research* 3 (2003), pp. 1157–1182. ISSN: 1532-4435.
- [21] Isabelle Guyon et al. “Gene Selection for Cancer Classification using Support Vector Machines”. In: *Machine Learning* 46.1-3 (2002), pp. 389–422. ISSN: 0885-6125.
- [22] Bernd Heißing, Metin Ersoy, and Stefan Gies, eds. *Fahrwerkhandbuch: Grundlagen, Fahrdynamik, Komponenten, Systeme, Mechatronik, Perspektiven. Mit 80 Tabellen*. 4., überarb. u. erg. Aufl. ATZ/MTZ-Fachbuch. Wiesbaden: Springer Vieweg, 2013. ISBN: 978-3-658-01992-1.
- [23] D. J. G. James et al. “Adaptive driver model using a neural network”. In: *Artificial Life and Robotics* 7.4 (2004), pp. 170–176. ISSN: 1433-5298.
- [24] Kenji Kira and Larry A. Rendell. “A Practical Approach to Feature Selection”. In: *Machine Learning Proceedings 1992*. Ed. by Derek Sleeman and Peter Edwards. San Francisco: Morgan Kaufmann, 1992, pp. 249–256. ISBN: 978-1-55860-247-2.

- 
- [25] Ron Kohavi. “A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection”. In: *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*. Vol. 14. San Francisco: Morgan Kaufmann, 1995, pp. 1137–1143. ISBN: 1-55860-363-8.
- [26] Igor Kononenko. “Estimating Attributes: Analysis and Extensions of RELIEF”. In: *Machine Learning: ECML-94*. Ed. by Francesco Bergadano and Luc Raedt. Lecture Notes in Computer Science, Lecture Notes in Artificial Intelligence. Berlin, Heidelberg: Springer, 1994, pp. 171–182. ISBN: 978-3-540-48365-6.
- [27] Max Kuhn. *caret: Classification and Regression Training*. R package version 6.0-80. 2018. URL: <https://CRAN.R-project.org/package=caret>.
- [28] Otto Lappi. “The Racer’s Mind - How Core Perceptual-Cognitive Expertise Is Reflected in Deliberate Practice Procedures in Professional Motorsport”. In: *Frontiers in Psychology* 9 (2018).
- [29] Yulong Lei et al. “Research on driving style recognition method based on driver’s dynamic demand”. In: *Advances in Mechanical Engineering* 8.9 (2016).
- [30] Andy Liaw and Matthew Wiener. “Classification and Regression by randomForest”. In: *R News* 2.3 (2002), pp. 18–22. ISSN: 1609-3631.
- [31] V. N. Lukanin et al. “Traffic flows modelling and the evaluation of energy-ecological parameters. Part I and II”. In: *International Journal of Vehicle Design* 33.4 (2003). ISSN: 0143-3369.
- [32] C. MacAdam et al. “Using Neural Networks to Identify Driving Style and Headway Control Behaviour of Drivers”. In: *Vehicle System Dynamics* 29.sup1 (1998), pp. 143–160. ISSN: 0042-3114.
- [33] Clara Marina Martinez et al. “Driving Style Recognition for Intelligent Vehicle Control and Advanced Driver Assistance: A Survey”. In: *IEEE Transactions on Intelligent Transportation Systems* 19.3 (2018), pp. 666–676. ISSN: 1524-9050.
- [34] Wolfgang Matschinski. *Radführungen der Straßenfahrzeuge: Kinematik, Elasto-Kinematik und Konstruktion*. Zweite Auflage. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. ISBN: 978-3-662-09652-9.
- [35] Robert McGill, John W. Tukey, and Wayne A. Larsen. “Variations of Box Plots”. In: *The American Statistician* 32.1 (1978), pp. 12–16. ISSN: 0003-1305.
- [36] Duane T. McRuer et al. “New Results in Driver Steering Control Models”. In: *Human Factors* 19.4 (1977), pp. 381–397. ISSN: 0018-7208.
- [37] Gys Albertus Marthinus Meiring and Hermanus Carel Myburgh. “A Review of Intelligent Driving Style Analysis Systems and Related Artificial Intelligence Algorithms”. In: *Sensors* 15.12 (2015), pp. 30653–30682. ISSN: 1424-8220.
- [38] John A. Michon. *Dealing with Danger: Report of the European Commission MRC Workshop on physiological and psychological factors in performance under hazardous conditions: Technical Report Nr. VK 79-01*. Ed. by University of Groningen. Gieten, 1979.

- 
- [39] Manfred Mitschke and Henning Wallentowitz. *Dynamik der Kraftfahrzeuge*. 5., überarb. u. erg. Aufl. 2014. Wiesbaden: Springer Fachmedien Wiesbaden, 2014. ISBN: 978-3-658-05068-9.
- [40] H. Ohno. “Analysis and modeling of human driving behaviors using adaptive cruise control”. In: *Proceedings of the 26th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2000, pp. 2803–2808. ISBN: 0-7803-6456-2.
- [41] Hans B. Pacejka and Egbert Bakker. “THE MAGIC FORMULA TYRE MODEL”. In: *Vehicle System Dynamics* 21.S1 (1992), pp. 1–18. ISSN: 0042-3114.
- [42] Hans B. Pacejka and Igo Besselink. *Tire and vehicle dynamics*. 3rd ed. Oxford UK: Butterworth-Heinemann, 2012. ISBN: 978-0-080-97017-2.
- [43] Manfred Plöchl and Johannes Edelmann. “Driver models in automobile dynamics application”. In: *Vehicle System Dynamics* 45.7-8 (2007), pp. 699–741. ISSN: 0042-3114.
- [44] Christian Prettentahler. “Optimale Kursplanung für Rennsportsimulationen”. Dissertation. Graz: Technische Universität Graz, 2008.
- [45] Günther Prokop. “Modeling Human Vehicle Driving by Model Predictive Online Optimization”. In: *Vehicle System Dynamics* 35.1 (2001), pp. 19–53. ISSN: 0042-3114.
- [46] Geqi Qi et al. “What is the Appropriate Temporal Distance Range for Driving Style Analysis?” In: *IEEE Transactions on Intelligent Transportation Systems* 17.5 (2016), pp. 1393–1403. ISSN: 1524-9050.
- [47] R Core Team. *R: A Language and Environment for Statistical Computing*. Ed. by R Foundation for Statistical Computing. R version 3.5.1. Vienna, Austria, 2018. URL: <https://www.R-project.org/>.
- [48] Jens Rasmussen. “Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models”. In: *IEEE Transactions on Systems, Man, and Cybernetics* 13.3 (1983), pp. 257–266. ISSN: 0018-9472.
- [49] Marko Robnik-Sikonja and Petr Savicky. *CORElearn: Classification, Regression and Feature Evaluation*. R package version 1.53.1. 2018. URL: <https://CRAN.R-project.org/package=CORElearn>.
- [50] RStudio Team. *RStudio: Integrated Development Environment for R*. Ed. by Inc RStudio. RStudio version 1.1.456. Boston, MA, 2016. URL: <http://www.rstudio.com/>.
- [51] Dario D. Salvucci. “Modeling driver behavior in a cognitive architecture”. In: *Human Factors* 48.2 (2006), pp. 362–380. ISSN: 0018-7208.
- [52] Peter Schoeggl, Erich Ramschak, and Erik Bogner. “On-board Optimization of Driveability Character Depending on Driver Style by Using a New Closed Loop Approach”. In: *SAE 2001 World Congress: Detroit, Michigan, March 5-8, 2001*. SAE International, 2001.

- 
- [53] Orit Taubman-Ben-Ari, Mario Mikulincer, and Omri Gillath. “The multidimensional driving style inventory—scale construct and validation”. In: *Accident Analysis & Prevention* 36.3 (2004), pp. 323–332. ISSN: 0001-4575.
- [54] Li Nyen Thin et al. “GPS Systems Literature: Inaccuracy Factors And Effective Solutions”. In: *International journal of Computer Networks & Communications* 8.2 (2016), pp. 123–131. ISSN: 0975-2293.
- [55] Julian Togelius. “Optimization, Imitation and Innovation: Computational Intelligence and Games”. Dissertation. Essex: University of Essex, 2007.
- [56] Michael Trzesniowski. *Antrieb*. Handbuch Rennwagentechnik. Wiesbaden: Springer Fachmedien Wiesbaden, 2017. ISBN: 978-3-658-15535-3.
- [57] Michael Trzesniowski. *Fahrwerk*. Handbuch Rennwagentechnik. Wiesbaden: Springer Fachmedien Wiesbaden, 2017. ISBN: 978-3-658-15545-2.
- [58] Michael Trzesniowski. *Gesamtfahrzeug*. Handbuch Rennwagentechnik. Wiesbaden: Springer Fachmedien Wiesbaden, 2017. ISBN: 978-3-658-15537-7.
- [59] Michael Trzesniowski and Philipp Eder. *Datenanalyse, Abstimmung und Entwicklung*. Handbuch Rennwagentechnik. Wiesbaden: Springer Fachmedien Wiesbaden, 2017. ISBN: 978-3-658-15547-6.
- [60] Ryan J. Urbanowicz et al. “Relief-Based Feature Selection: Introduction and Review”. In: *Journal of Biomedical Informatics* 85 (2018), pp. 189–203. ISSN: 1532-0464.
- [61] Timo Völkl. *Erweiterte quasistatische Simulation zur Bestimmung des Einflusses transienten Fahrzeugverhaltens auf die Rundenzeit von Rennfahrzeugen*. Göttingen: Cuvillier, 2013. ISBN: 978-3-736-94503-6.
- [62] Peter Waldmann. *Entwicklung eines Fahrzeugführungssystems zum Erlernen der Ideallinie auf Rennstrecken*. Vol. 1. Schriftenreihe des Lehrstuhls Fahrzeugtechnik und -antriebe der BTU Cottbus. Aachen: Shaker, 2009. ISBN: 978-3-832-28229-5.
- [63] T. Wang and K. Liaw, eds. *Driving style imitation in simulated car racing using style evaluators and multi-objective evolution of a fuzzy logic controller: 2014 IEEE Conference on Norbert Wiener in the 21st Century (21CW)*. 2014.
- [64] Michael Wegscheider. “Modellbasierte Komfortbewertung von Fahrerassistenzsystemen”. Dissertation. Graz: Technische Universität Graz, 2009.
- [65] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016. ISBN: 978-3-319-24277-4.
- [66] Hadley Wickham. *tidyverse: Easily Install and Load the 'Tidyverse'*. R package version 1.2.1. 2017. URL: <https://CRAN.R-project.org/package=tidyverse>.
- [67] Hadley Wickham et al. *dplyr: A Grammar of Data Manipulation*. R package version 0.7.6. 2018. URL: <https://CRAN.R-project.org/package=dplyr>.

- 
- [68] Hermann Winner et al., eds. *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. 3rd ed. ATZ/MTZ-Fachbuch. Wiesbaden: Springer Vieweg, 2015. ISBN: 9783658057336.
- [69] Lukas Wörle, Michael Graf, and Arno Eichberger. “Objective Metrics for Control Inputs of Racecar Drivers”. In: *FISITA 2018 World Automotive Congress*. FISITA, 2018. ISBN: 978-0-9572076-5-3.
- [70] Marvin N. Wright and Andreas Ziegler. “ranger : A Fast Implementation of Random Forests for High Dimensional Data in C++ and R”. In: *Journal of Statistical Software* 77.1 (2017).
- [71] C. You, J. Lu, and P. Tsiotras. “Nonlinear Driver Parameter Estimation and Driver Steering Behavior Analysis for ADAS Using Field Test Data”. In: *IEEE Transactions on Human-Machine Systems* 47.5 (2017), pp. 686–699. ISSN: 2168-2291.
- [72] Y. Zhang, W. C. Lin, and Y. K. S. Chin. “A Pattern-Recognition Approach for Driving Skill Characterization”. In: *IEEE Transactions on Intelligent Transportation Systems* 11.4 (2010), pp. 905–916. ISSN: 1524-9050.