The Future of Car Displays

The Development of a Virtual Reality Toolkit

S.L.G. Koning

University of Twente

Master Thesis

August, 2020

Student number: S1863002

First Supervisor: dr. S.Borsci

Second Supervisor: prof.dr.ing. W.B. Verwey

Faculty of Behavioural, Management and Social Sciences

Abstract

In this paper, two studies will be presented. The primary goal of these studies was to support the effective use of the new virtual reality car simulator of the University of Twente. This work will facilitate future research with a driving simulator by developing two assets: I) a user manual was developed and tested to support researchers in conducting driving-related experiments with the simulator II) an experiment regarding the effect of digital mirror positions on user driving performance was formulated and piloted in the driving simulator. The present work details the performed usability test and the formulated virtual reality experiment. Results of the pilot were used to build a toolkit composed of the experimental design, methods, procedures and a package for the data analysis in R.

Keywords: Driving Simulator, Digital Mirrors, Virtual Reality, Usability, Research Toolkit.

Content

Abstract 2

1. Introduction 6

Section I: Test of the User Manual for the Driving Simulator 8

2.1 Background of the User Manual Analysis 8

2.2 Methods 9

2.2.1 Participants 9

2.2.2 Design 9

2.2.3 Material 10

2.2.4 Procedure 10

2.2.5 Data Analysis 11

2.3 Results 11

2.3.1 Usability Problem 1 11

2.3.2 Usability Problem 2 12

2.3.3 Usability Problem 3 13

2.3.4 Usability Problem 4 13

2.3.5 Usability Problem 5 14

2.3.6 Usability Problem 6 14

2.3.7 Usability Problem 7 15

2.3.8 Additional Findings 15

2.4 Discussion of User Manual Analysis 16

Section II: Pilot Study on Digital Mirror Positions 17

3.1 Background of the Pilot Study 17

3.2 Methods 18

3.2.1 Participants....18
3.2.2 Design....18
3.2.3. Materials....18
3.2.4 Procedure....21
3.2.5. Data Analysis....21
3.3 Results....22
3.3.1 Subjective Data....22
3.3.2 Performance Data....22
3.3.3 Eye-tracking data....28
3.4 Discussion of Pilot Study....29
4. Conclusion....30
References....31
Appendix A....33
User Manual....33
Appendix B....48
Informed Consent....48
Appendix C....51
Protocol....51
Appendix D....53
Problem Description Form....53
Appendix E....54
User Manual Adjusted....54
Appendix F....75
NASA TLX....75

Appendix G 76

User Experience Questionnaire 76

Appendix H 77

Confidence Questionnaire 77

Appendix I 78

Simulator Sickness Questionnaire 78

Appendix J 79

Demographic Questionnaire 79

Appendix K 80

Informed Consent 80

Appendix L 83

Analysis Package 83

1. Introduction

Each year, approximately 1.35 million people die because of involvement in a traffic accident (World Health Organisation, 2018). Around 94% of these accidents are caused by human error instead of technical issues (National Highway Traffic Safety Administration, 2017). Therefore, tools for enabling assessment on driving performance are crucial. During this project, a virtual reality (VR)-based driving simulator was developed running on Unity software (version 2019.2.19). Unity is a game engine used for the production of 2D and 3D video games. To realise the virtual driving simulation, various Unity assets were acquired to use as a base for the simulator software. These included files consisting of graphics, 3D models, game objects and animations. Combined, these assets contained all necessary elements to create an appropriate driving experience including characteristics of the city, car and traffic[1]. The hardware consisted of a VR headset, a steering wheel and pedals. The software was developed according to the agile approach. This software development methodology promotes an incremental delivery of software shares instead of an "all at once" approach. Therefore, the development process was subdivided into several short iterative cycles followed by a release of working software. This allowed for continuous, frequent evaluation and adaptation of the process in order to deliver a greater quality of software at the end of the project.

The current simulator is highly adaptable. Users have the possibility to change the system to fit a wide variety of experiments (e.g. change traffic situations, record several types of data etc.). Even though Unity software is characterized by its customizability, the staff of the Behaviour, Management and Social science (BMS) lab experienced that students faced difficulties while operating the software. Findings of a study performed by Mercan and Durdu (2017) revealed that especially novice users have difficulty using Unity. Therefore, it was acknowledged that researchers have an increased demand for support using this software. During this project, a foundation was set supporting students and researchers at the University of Twente in conducting various driving-related studies at the BMS lab. This paper is divided into two sections: section I presents the development process of a manual for the simulator and the results of a usability test that was performed to assess the quality of

[1] The assets used included the NWH Vehicle Physics 2, the Fantastic City Generator and the iTS - Intelligent Traffic System. These assets are available at the Unity asset store (https://assetstore.unity.com/).

interaction between the user and the system; section II presents the methods and results of a pilot experiment that was performed to test the created simulator. Besides testing the simulator, the pilot study was also conducted to examine the use of digital mirrors; a new method to display the rear-road scene to the driver (Large, Crundall, Burnett, Harvey, & Konstantopoulos, 2016). The results of the pilot were used to build a toolkit composed of the experimental design, methods, procedures and the potential results including a package for the data analysis in R. The toolkit assists researchers in exploring the use of driver assistance technologies.

Section I: Test of the User Manual for the Driving Simulator

2.1 Background of the User Manual Analysis

A driving simulator is a suitable tool for conducting experiments in human factor research. It has several advantages over real-world testing, including; experimental control, efficiency, safety and easy data collection (Godley, Triggs, & Fildes, 2002). However, as aforementioned, the difficulty of operating the hardware and software can be an obstacle for researchers in successfully conducting their experiments. Since a wide range of researchers should be able to operate the simulator, it is required that appropriate manuals are developed to facilitate the use of this system. Software documentation is essential to illustrate the operation of the system and assists in using it (Kipyegen & Korir, 2013). Therefore, it was acknowledged that a user manual of good quality must be created and tested to optimize the interaction between the user and the system.

Several studies examined ways of presenting instructions in user manuals. Karreman, Ummelen, and Steehouder (2005) studied the nature of the information presented in manuals. They introduced two types of information: procedural and declarative information. Procedural information is the most important type of information as it clarifies to the user what actions must be executed in order to achieve the desired outcome (action-centred). The declarative information type explains the internal working of the device. The benefits of including declarative information in user instruction are yet unknown. Procedural information often suffices in helping users operating the system.

Mayer and Moreno (2003) introduced three principles for displaying information in manuals. Adhering these principles could reduce cognitive load and enhance efficiency of learning. The first principle is the multimedia principle. This principle states that people learn quicker when words and pictures are combined in instructions. The second principle, the signalling principle, states that learning performance improves when cues are added that highlight the most important information. These cues include e.g. arrows or different colour coding of information. The third principle, the segmenting principle, states that instruction should be presented in learner paced segments instead of a continues unit.

Instructions in the user manual for the driving simulator are presented according to the aforementioned principles proposed by Mayer and Moreno (2003) and Karreman et al. (2005). For instance, the manual mainly consists of procedural information. In addition, written instructions are combined with pictures including arrows and different colour coding to indicate important information. Also, instructions are broken down into small, easy to perform steps. The user manual includes information on each element in the simulator. Large parts of it are dedicated to operating procedures of the Unity software and the assets. The first chapter includes information on the environment. It explains how the city is composed and can be adjusted e.g. lane composition. The second chapter includes information on car physics. An explanation is given on changing car components e.g. acceleration sensitivity and steering components. The third chapter contains information on adjusting traffic objects. In this part, an explanation is given on adjusting elements in the traffic surrounding the ego car e.g. the traffic density and speed of surrounding cars. The fourth chapter on logs includes information on data collection in Unity e.g. to record the steering deviation and acceleration levels. The last component consists of information on setting up an experimental run e.g. calibration of controllers and eye trackers. The initial version of the manual can be found in Appendix A. As aforementioned, the described difficulties in using the Unity software were the most important reasons for optimizing the manual. The goal of the present study was to address the main usability problems; issues in the system that make it inefficient, difficult or impossible for the users to achieve their goals (Lavery, Cockton, & Atkinson, 1997). In order to redesign the manual, an interaction test was designed and conducted aiming at achieving the desired level of usability.

2.2 Methods

2.2.1 Participants

For this study, 5 participants (3 males and 2 females) were recruited between the age of 19 and 57 (Mean=34.2; SD= 19.0). All participants were of Dutch nationality and were recruited based on availability and willingness to take part. Prior to the participation, they all signed the informed consent form. None of the participants had any prior experience with using the Unity software.

2.2.2 Design

During the test, a think-aloud protocol was used to collect data on the use of the manual. This involved that participants were asked to verbalise their thoughts as they were performing a set of tasks. The test was supplemented by a short interview about the use of the manual. The primary outcome measure included usability problems verbalised by the participants.

2.2.3 Material

The test was executed using a PC on which the Unity software was installed. Participants had to perform four tasks presented in one scenario while using the Unity software with the assets. Open Broadcaster Software (version 25.0.8) was used to capture the screen and speech of the participant.

2.2.4 Procedure

At the beginning of the test, participants were asked to read and sign the informed consent form (Appendix B). They were seated in front of the PC and were given the following scenario description;

> You want to conduct a study in the driving simulator. For this study, you will use the Unity game software. This software consists of a virtual world in which you can drive around in a created city. You will use the manual to make some adjustments in the scene to fit your experiment.

The scenario included multiple tasks that induced them to use the manual. Participants had to adjust aspects of the scene to the requirements of a fictional experiment. They had to select the right elements in the Unity interface to finish the following tasks;

1. Change the maximum speed of the car to an arbitrary new value.
2. Adjust the position of the car to an arbitrary new position.
3. Add traffic to the scene and make the density higher.
4. Let the Unity system record the speed of the car.

Subsequently, four questions about the use of the manual were asked. This was done to determine if participants required the manual to complete the tasks and to obtain useful recommendations on how the manual could be improved. These questions were;

1. What was your overall experience using the manual?
2. Were you able to perform the tasks properly while using the manual?
3. What recommendations would you give to improve the manual?
4. Was the manual helpful in accomplishing the tasks?

Lastly, the participants were asked demographic questions and were thanked for participating in the study. In Appendix C, the full protocol can be found including the questionnaire, the tasks presented in the scenario and the corresponding solutions.

2.2.5 Data Analysis

The video and audio recordings along with notes taken during the test were used to analyse user behaviour. Problems that occurred were administered in a problem description form (Appendix D). Additionally, a discovery matrix was composed presenting the frequency of the encountered usability problems.

2.3 Results

The discovery matrix (Table 1) presents if a certain participant encountered a usability problem (indicated by a 1) or not (indicated by a 0). The bottom row shows how many times the usability problems (UP) were found. On average, 69% of the participants identified each one of the seven main usability problems. In the next section, a detailed description is given for each usability problem.

Table 1

Discovery Matrix of the Usability Problems

	UP001	UP002	UP003	UP004	UP005	UP006	UP007
Participant01	1	1	0	1	1	1	1
Participant02	1	1	1	1	1	1	1
Participant03	1	0	0	1	0	1	0
Participant04	0	0	1	1	0	0	1
Participant05	1	1	1	1	1	0	0
Sum	4	3	3	5	3	3	3

Note: The table presents the frequency of the encountered usability problems (UP). It presents if a certain participant encountered a usability problem (indicated by a 1) or not (indicated by a 0).

2.3.1 Usability Problem 1

Usability problem 1 was mentioned by 80% of the participants. Participants had difficulty finding the right tabs in the Unity software (Figure 1). References to the “hierarchy tab” and the “inspector window” in the manual were not understood by the user.

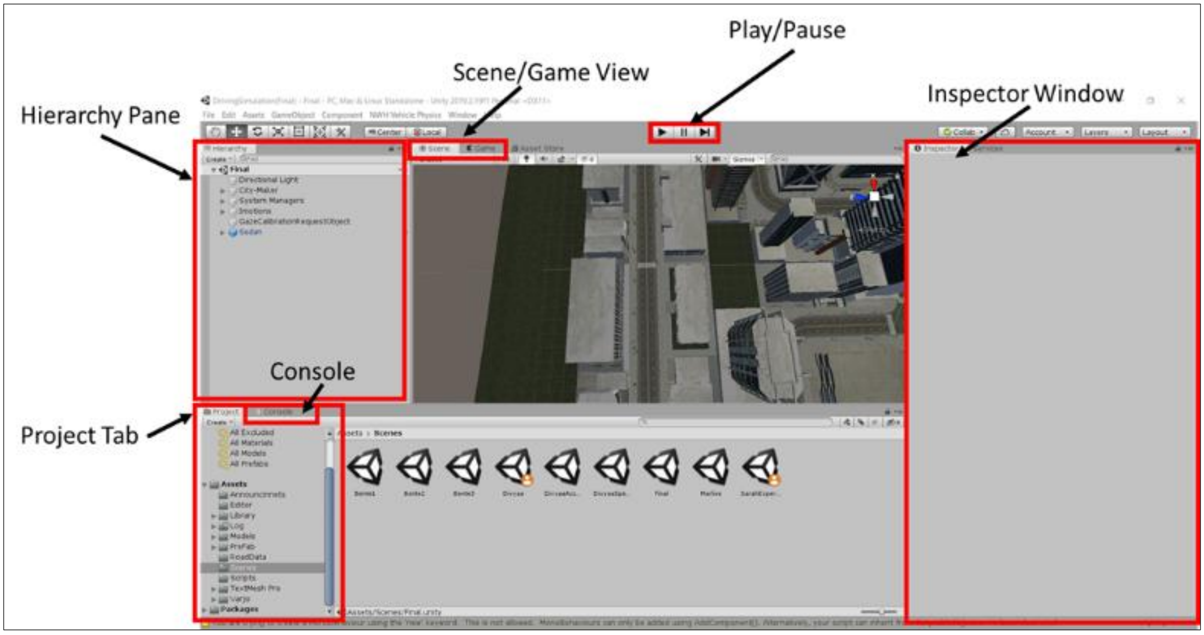
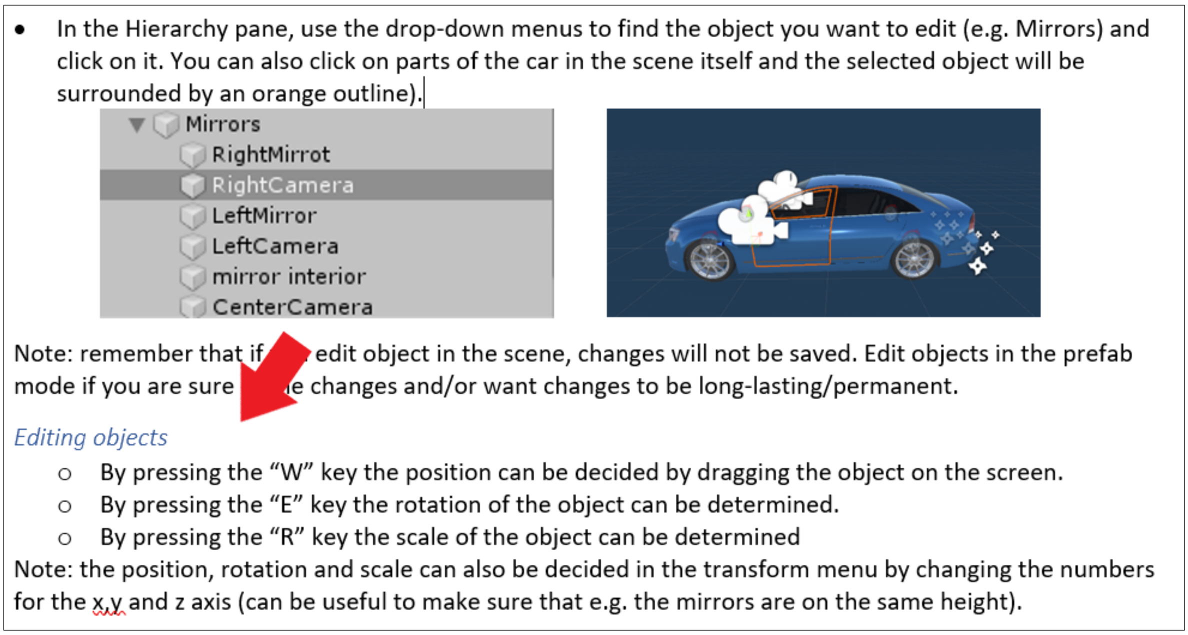


Figure 1. Unity interface layout with the five key functions highlighted in red: Hierarchy Pane, Project Tab, Console, Scene/Game View, Play/ Pause, Inspector Window.

The visual presentation of the functions is described in the first part of the manual. Nonetheless, the majority of the participants did not take proper notice of this explanation. Instead, they initially searched for keywords in the content table and thus missed the information concerning the functions. Participants who did manage to find the information in the manual gained a better understanding of the references made to the "hierarchy tab" or the "inspector window". Therefore, the reader should be instructed on first reading this section of the manual.

2.3.2 Usability Problem 2

Usability problem 2 was mentioned by 60% of the participants. Participants had difficulty repositioning the car. They were unaware of the fact that the object should be selected before repositioning by dragging the object or changing the values within "*transform*". Participants started reading from the header in Figure 2 and neglected the information above it. Subsequently, users tried pressing the W-key but experienced that nothing happened. A potential remedy to this problem could be to add information on "object selection" underneath the header.



- In the Hierarchy pane, use the drop-down menus to find the object you want to edit (e.g. Mirrors) and click on it. You can also click on parts of the car in the scene itself and the selected object will be surrounded by an orange outline).

Note: remember that if edit object in the scene, changes will not be saved. Edit objects in the prefab mode if you are sure e changes and/or want changes to be long-lasting/permanent.

Editing objects

- By pressing the "W" key the position can be decided by dragging the object on the screen.
- By pressing the "E" key the rotation of the object can be determined.
- By pressing the "R" key the scale of the object can be determined

Note: the position, rotation and scale can also be decided in the transform menu by changing the numbers for the x,y and z axis (can be useful to make sure that e.g. the mirrors are on the same height).

Figure 2. Manual instruction on editing objects in the Unity scene.

2.3.3 Usability Problem 3

Usability problem 3 was mentioned by 60% of the participants. When participants activated the output log, they were uncertain if the system would automatically save the changes they made. This happened because the Unity software does not confirm when data recording is activated. Users became confused and were uncertain if their experiment would be recorded. To improve this issue, information on saving changes should be added to the description.

2.3.4 Usability Problem 4

Usability problem 4 was mentioned by 100% of the participants. Users had difficulty identifying the right headers in the manual. For example, the description under the header "*traffic spawner*", including information on adding traffic to the scene and changing the traffic density, was not recognized by users. This gives way to the presumption that participants were unaware of the meaning of the word "*spawner*" and therefore ignored the information given by the header. Besides, users did not expect to find information about changing the position of the car under the header "*editing objects*". Both headers may consist of too much slang e.g. the word "*spawn*"; a word often used in the gaming context. This can be remedied by changing this "technical language" to more practical language, by changing "*traffic spawner*" to "*add traffic*" and "*editing objects*" to "*positioning, scaling and rotating objects*".

2.3.5 Usability Problem 5

Problem 5 was mentioned by 60% of the participants. The participants checked the wrong box to activate the *traffic spawner* or did not select it at all (Figure 3).

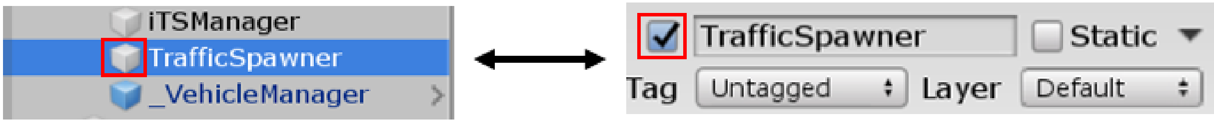


Figure 3. Boxes to activate the traffic spawner.

When the participant had to fill in the file name to prepare the output log, they confused the input bars shown in Figure 4. The input bar at the top defines the name of the output module object. The lower one sets the file name of the output file. Changing the wrong file name could lead to loss of information. Adding cues to the manual indicating necessary adjustments can remedy this.

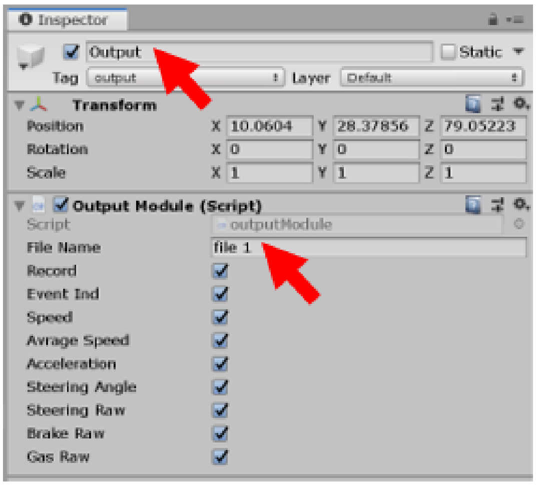


Figure 4. Input Bars for defining object name and file name.

2.3.6 Usability Problem 6

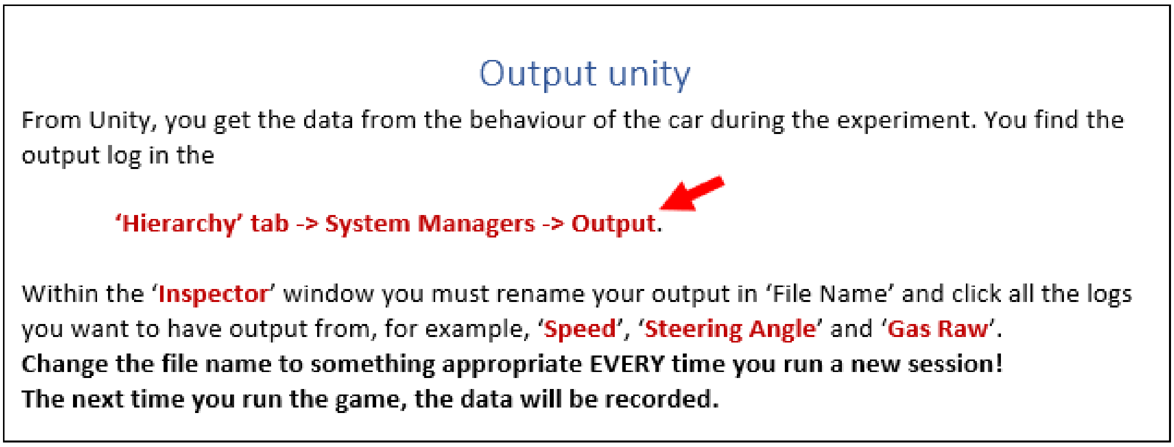
Problem 6 was mentioned by 60% of the participants. In the manual, a reference is made to other pages of the manual. In this section, an explanation is given on activating the output log. However, users often did not recognize this and searched for similar elements in the software. A design recommendation would be to repeat the message instead of referring to other parts of the manual. Redundancy would, in this case, be beneficial.

2.3.7 Usability Problem 7

Problem 7 was mentioned by 60% of the participants. Users became confused by the headers "*Setting up an experimental run*" and "*Setting up your experiment*". These headers are too alike, which makes it difficult for the user to distinguish between the two sources of information. As these chapters respectively describe how to prepare an experiment and how to run it, it can be improved by specifying these chapters as "*Preparing your experiment*" and *"Starting an experimental run"*.

2.3.8 Additional Findings

Some additional findings were gathered during the interviews. In general, users had a positive attitude towards the operation of the manual. The manual was in their opinion essential to understanding the software and finishing the tasks in a proper amount of time. Nonetheless, participants gave general recommendations on improving the use of the manual. To begin with, users acknowledged the usefulness of reading manual entirely before using the software. If participants were given time to read the entire manual, they expected to be less prone to miss out on crucial information. Additionally, highlighting the step-by-step instructions in the manual was recommended by participants (Figure 5). Tracing back information becomes more difficult when users are forced to shift their attention from the software to the manual. Highlighting information could help users to trace back information while moving back and forth from the manual to the software.


Output unity

From Unity, you get the data from the behaviour of the car during the experiment. You find the output log in the

'Hierarchy' tab -> System Managers -> Output.

Within the '**Inspector**' window you must rename your output in 'File Name' and click all the logs you want to have output from, for example, '**Speed**', '**Steering Angle**' and '**Gas Raw**'.
Change the file name to something appropriate EVERY time you run a new session!
The next time you run the game, the data will be recorded.

Figure 5. Example of highlighted step-by-step instruction in the manual.

2.4 Discussion of User Manual Analysis

The present study examined the quality of the user manual created for the VR-based driving simulator of the BMS Lab. During the usability test and interviews, participants verbalized issues regarding; the structure of the manual, use of too much slang, unclear headers and confusion of elements. Improvements were made based on the results of the usability tests and recommendations from the interviews. The new improved manual can be found in Appendix E. Nevertheless, two limitations should be mentioned regarding the test and the use of the manual. First, the test only covered the implementation of simple experiments. It is unknown whether the manual is of sufficient quality to support the researcher in conducting experiments that require major adjustment in the software. Besides, the tasks only focussed on the use of the software. Another test should be performed to examine the quality of the instructions on the simulator's hardware.

None of the participants had any prior experience with using the software. Nevertheless, it is proven that users can make independent adequate changes to the software when using the manual. By improving and redesigning the initial manual, it is assumed that students will be better able to operate the software on their own. Therefore, this manual can support students in conducting driving-related experiments at the University of Twente regardless of the user's prior experience with using the software.

Section II: Pilot Study on Digital Mirror Positions

3.1 Background of the Pilot Study

Digital mirrors can be beneficial to support the driver. They can be defined as displays that present the image that is captured by a rear-facing camera on the car (Large et al., 2016). It is important to consider and examine the use of digital mirrors because traditional mirrors have their limitations. These include image reversal, creation of a blind spot and limitations on physical placement and heavy weather limiting the sight of the mirrors (Large et al., 2016). Research has been performed on the implementation of digital mirrors. Flannagan, Sivak, Schumann, Kojima, and Traube (1997) stated that drivers using the passenger side mirrors, overestimated the distance more frequent compared to drivers using the driver-side mirrors. Hahnel and Hecht (2012) pointed out the correlation in distance between the observer's eye-point and the mirror, and judgment and estimation of distance. Digital mirrors offer the opportunity to place the offside, nearside and rear-view mirrors in more intuitive positions since they are camera-based. Lamble, Laakso, and Summala (1999) studied the effect of different mirror positions. They concluded that the most optimal position of the mirrors is as close as possible to the driver's normal line of sight. This is the visible path from the vehicle to the target area. According to this study, optimal locations for mirrors include the top of the dashboard or the sides of the steering wheel. Another study performed by Wittmann et al. (2006) stated that the amount of distance between the driver's line of sight and the display affects individuals' driving performance. Driving tasks performed with mirrors that are placed far away from the driver's line of sight turned out to be less supportive. According to this study, mirrors should therefore be placed just above the mid console in the line of sight of the driver. To summarise, digital mirrors create new opportunities to overcome the limitations of traditional mirrors. However, digital mirrors can affect normal visual scanning behaviour of the driver. They create new perspectives that are incongruent with the mental model of the driver (Large et al., 2016). New mirror positionings and digital displays can cause an increase in cognitive load. McLaughlin, Hankey, Green, and Kiefer (2003) studied the use of digital mirrors and video view during a parking task. They found an increase in total eyes-off-road-time during the task while participants were using the digital mirrors. Studies showed that long and frequent glances away from the road present a significant risk for safety (Simons-Morton, Guo, Klauer, Ehsani, & Pradhan, 2014). Eye glances away from the road for more than 2 seconds significantly increase the risk for a crash and should be avoided (National Highway Traffic Safety Administration, 2013).

During the pilot test, a comparison was made between traditional mirror positions and new intuitive digital mirror positions. The experiment was performed in the developed driving simulator. Subjective ratings, performance data and eye-tracking data were gathered to assess the usability of the different digital mirror setups. A toolkit was composed of a procedure, test material and an R data analysis package. Future research can benefit from the use of this toolkit to conduct a full virtual reality experiment. The present study delivered an initial contribution for future research on the subject and can therefore be considered a pilot project to a more extensive study.

3.2 Methods

3.2.1 Participants

Two participants were recruited in the pilot study. Participant one was a 44 year old male. He had 22 years of driving experience and was of Dutch nationality. Participant two was a 19 year old female. She had one year of driving experience and was of German nationality. The participants were both psychology students. They were recruited through the test subject pool of the University of Twente and were rewarded with study points for test subject hours. Prior to the participation, they both signed the informed consent form.

3.2.2 Design

The study included a within-subject design as all the participants performed the tasks using all the setups. The independent variables included the mirror setup type: the traditional mirror setup and the two new digital mirror setups (Setup A and Setup B). Also, the effect of adjustment was examined by assessing each sequential task that participants performed in each condition. Dependent variables included driving performance, workload, user experience, confidence level and gaze behaviour.

3.2.3. Materials

A VR-based driving simulator was used to perform the test. The experiment was conducted under three conditions. Each contained a driving scenario with a different mirror setup. The first condition involved a classic mirror setup. Mirrors were placed in traditional areas on the left and right side of the car. The rear-view mirror was placed at the top in the middle of the window (Figure 6). In the second condition (Setup A), off-side and near-side mirrors were positioned inside the car at the corners of the window (Figure 7). In the third condition (Setup B), the off-side mirror was positioned on the left side of the steering wheel. The rear-view mirror was placed at the bottom centre of the dashboard (Figure 8) which according to Wittmann et al. (2006) is the most effective and intuitive position for rear-view sight. The near-side mirror was placed at the right of the rear-view mirror. According to Flannagan et al. (1997), this decreases the distance between the mirror and the driver and thus reduces the risk of overestimating distance.

Figure 6. Classic setup with traditional mirror positions.

Figure 7. Setup A with near-side and off-side mirror inside the car.

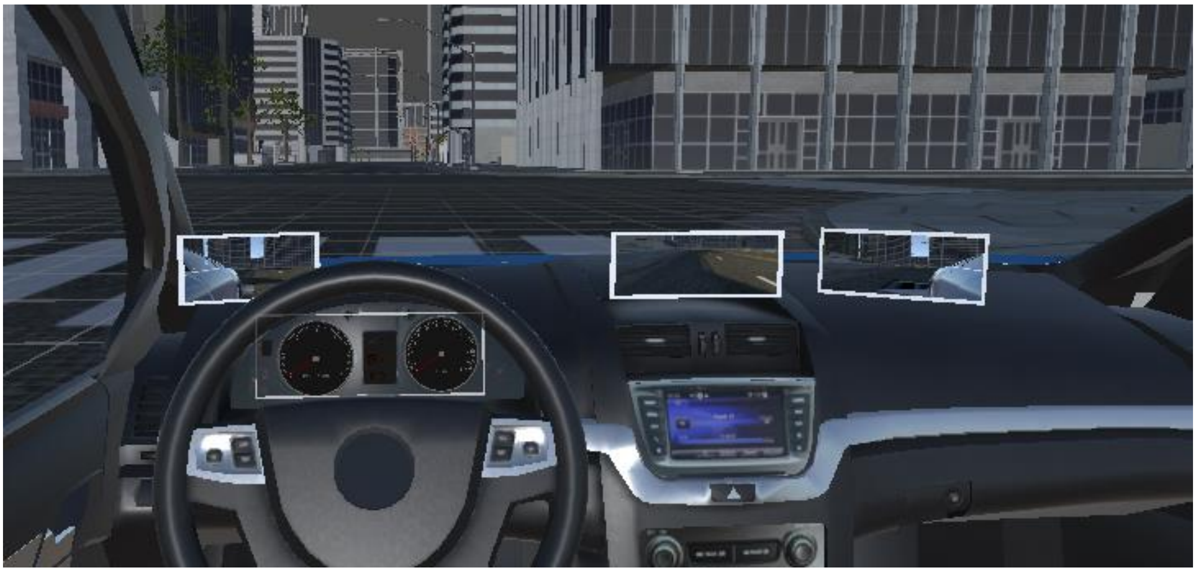


Figure 8. Setup B with mirrors on the dashboard.

During the experiment, eye-tracking equipment was used to capture gaze behaviour and fixations. To measure workload and user experience for each condition, the NASA TLX (Hart & Staveland, 1988) (Appendix F) and the dependability and perspicuity scale of the User Experience Questionnaire (UEQ) (Schrepp, Hinderks, & Thomaschewski, 2017) (Appendix G) were used. These scales were chosen because they were most relevant for testing the usability of the mirror setups (scales regarding e.g. attractiveness were not applicable). A single item scale was used to measure confidence level of the participants when they performed the tasks (Appendix H). To check if the participant suffered from simulator sickness, the SSQ (Sevinc & Berkman, 2020) was used to safeguard the wellbeing of the participant (Appendix I). Additionally, performance measures were captured. First, the time was measured that people required for the lane change tasks. Second, the headway distance was measured at the point where the lane change manoeuvre was completed. This was captured to examine if there was a difference in lane change behaviour for the different setups. These measures could indicate how rapid people absorb information from the mirrors of the different setups (Madigan, Louw, & Merat, 2018). Additionally, a questionnaire was used to collect demographic data (Appendix J).

3.2.4 Procedure

First, the participants were informed about the study goals and asked to read and sign the informed consent form (Appendix K). A VR-headset was placed and calibrated. During the experiment, the participants were immersed in a virtual driving environment. Before the start of the experiment, they were given the opportunity to familiarise themselves with the simulator. At the beginning of each test run, questions from the simulator sickness questionnaire were asked to safeguard the well-being of the participant. Subsequently, the participants had to perform different tasks with each setup. These tasks included a lane change and a parking task. Multiple lane change tasks were performed on a track with a dual carriageway. Afterwards, participants were instructed to drive to the parking lot and to back up between two markers. The procedure is schematically presented in Figure 9. The tasks were first performed with the classic setup. Afterwards, the tasks were performed with Setup A and Setup B. The order of the used conditions was balanced. Each sequential participant started alternately with a different condition. Between the test runs, participants were given a break of approximately 10 minutes. At the end of each test run the NASA TLX, the UEQ and the single item confidence scale were administered. Each test run lasted for approximately 10 minutes.

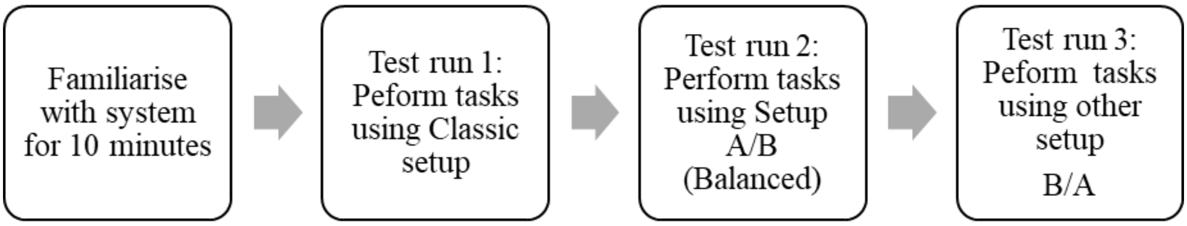


Figure 9. Schematic representation of the experiment's procedure.

3.2.5. Data Analysis

Data analysis was performed in RStudio (version 1.3). Subjective data, performance data and eye-tracking data were analysed separately. First, summary tables and graphs were created to visualise the data. A group difference test was conducted to compare the effect of the mirror setup type on workload, confidence and usability. Regarding performance data, a generalized linear model was used to assess how people adjusted to different designs. Subsequently, another group difference test was conducted to compare the effect of the mirror setup type on lane change time and headway distance. To prepare the eye-tracking data, areas of interest were defined in the IMotion software (version 8.1) and the data file was exported and loaded into R. Lastly, a group difference test was conducted to compare the effect of the mirror setup type on fixation duration and time spent looking in the mirrors.

3.3 Results

3.3.1 Subjective Data

The SSQ results showed no problems regarding cybersickness of the two participants. In Table 2, the averages and standard deviation for the subjective data are presented. A group difference test (One-way ANOVA and Kruskal Wallis test) was conducted to compare the effect of the mirror setup type on subjective ratings. There was no significant effect of the mirror setup type on workload score ($F(2, 3) = 0.131$, $p = .882$), lane change confidence ($F(2, 3) = 1.455$, $p = .362$), parking confidence ($F(2, 3) = 0.463$, $p = .668$), perspicuity ($F(2, 3) = 0.013$, $p = .987$) and dependability ($H(2) = 1.25$, $p = .535$).

Table 2

Averages and Standard Deviation of Subjective data per Setup.

	Workload		Confidence Lane changing		Confidence Parking		Perspicuity		Dependability	
Setup	*M*	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*
Classic	36.25	17.50	7.75	0.35	7.00	1.41	1.88	1.24	0.88	0.88
Setup A	34.60	19.20	8.75	0.35	4.50	3.53	2.00	0.35	1.00	0.71
Setup B	28.75	16.30	7.75	1.06	4.50	3.53	1.88	0.88	0.38	0.18

Note: The workload scale ranges from 1 to 100, the confidence scale ranges from 1 to 10 and the perspicuity and dependability scales range from -3 to +3.

3.3.2 Performance Data

In Table 3, the averages and standard deviation for the performance data are presented. Figure 10 presents the average time that was required for the participants to perform sequential lane changes. The duration widely differed from the first lane change to the last lane change performed. In Figure 11, the lane change time for each sequential lane change is displayed per participant.

Table 3

Averages and Standard Deviation for Lane change time (s) and Headway Distance (m) per Setup

	Lane change time (s)		Headway Distance (m)	
Setup	*M*	*SD*	*M*	*SD*
Classic	3.64	1.69	37.86	36.05
Setup A	3.63	2.71	39.87	35.01
Setup B	4.69	2.43	32.93	16.79

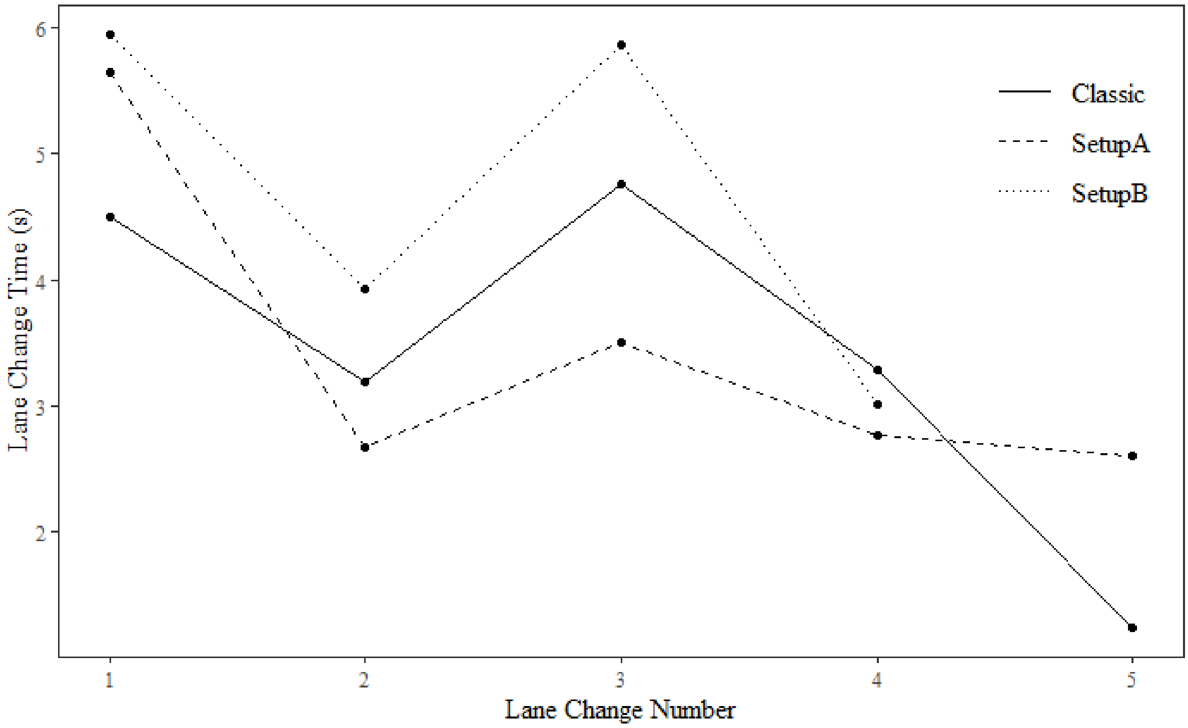


Figure 10. Average time needed to perform the task for each sequential lane change and each setup used.

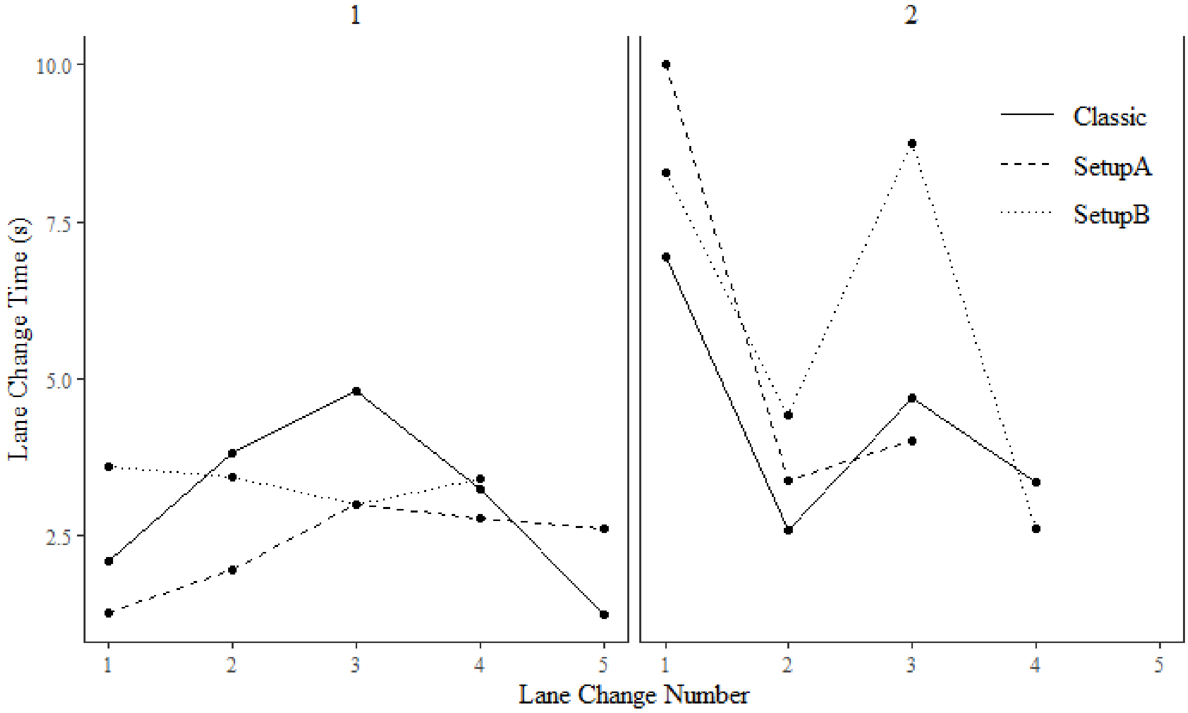


Figure 11. Time needed to perform the task for each sequential lane change and each setup used. The charts show the individual score of the two participants.

In Table 4, the results of the regression analysis are presented. The intercept represents the time that people needed to perform the lane change tasks with the classic setup while making the first lane change. Based on the regression analysis, it is expected that in the first session it will take people 0.19 seconds longer with Setup A and 0.92 seconds longer with Setup B. Subsequently, the time will decrease with 0.55 seconds per session for the classic setup. For Setup A, performance is expected to improve with 0.66 seconds (-0.55 – 0.11) per session. For Setup B, this will be 0.60 seconds (-0.55 – 0.05). However, the confidence intervals are broad-based which makes the results less certain. The predicted model is visualised in Figure 12.

Table 4

Results of Regression Analysis for Lane Change Time (s) with 95% Confidence Intervals (Lower and Upper)

Parameter	Location	Lower CI	Upper CI
Intercept	5.16	1.87	8.49
Group Setup A	0.19	-4.39	4.66
Group Setup B	0.92	-3.65	5.70
Lane change number	-0.55	-1.62	0.51
Group Setup A: Lane change number	-0.11	-1.62	1.46
Group Setup B: Lane change number	-0.05	-1.76	1.62

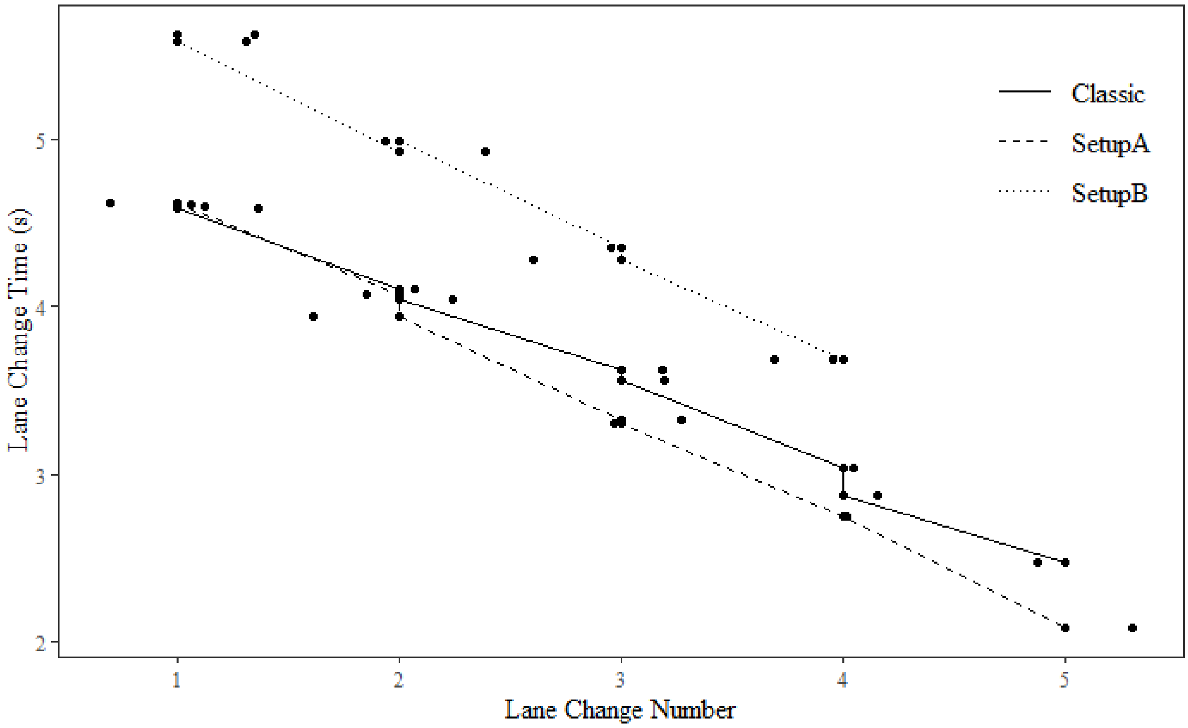


Figure 12. Prediction of time needed to perform each sequential lane change per setup.

In Figure 13, the average headway distance is presented for each setup. Compared to the lane change time, a less clear decay is present in this graph. In Figure 14, the headway distance for each lane change is displayed per participant.

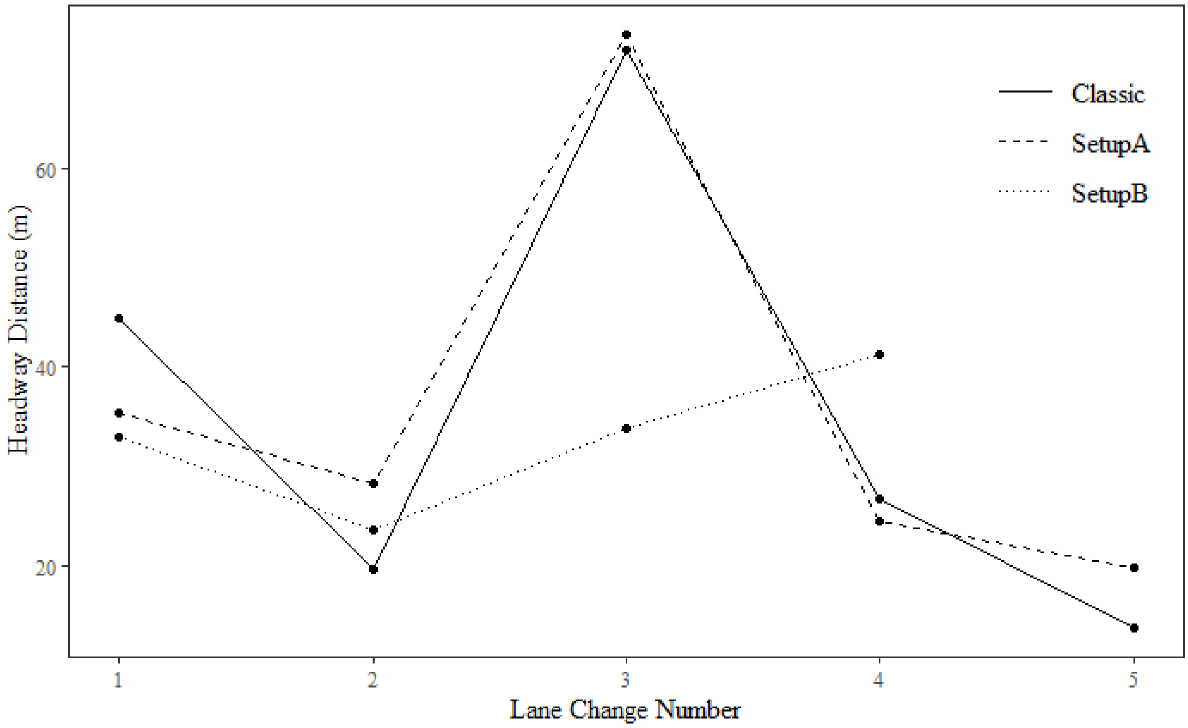


Figure 11. Average headway distance for each sequential lane change and each setup used.

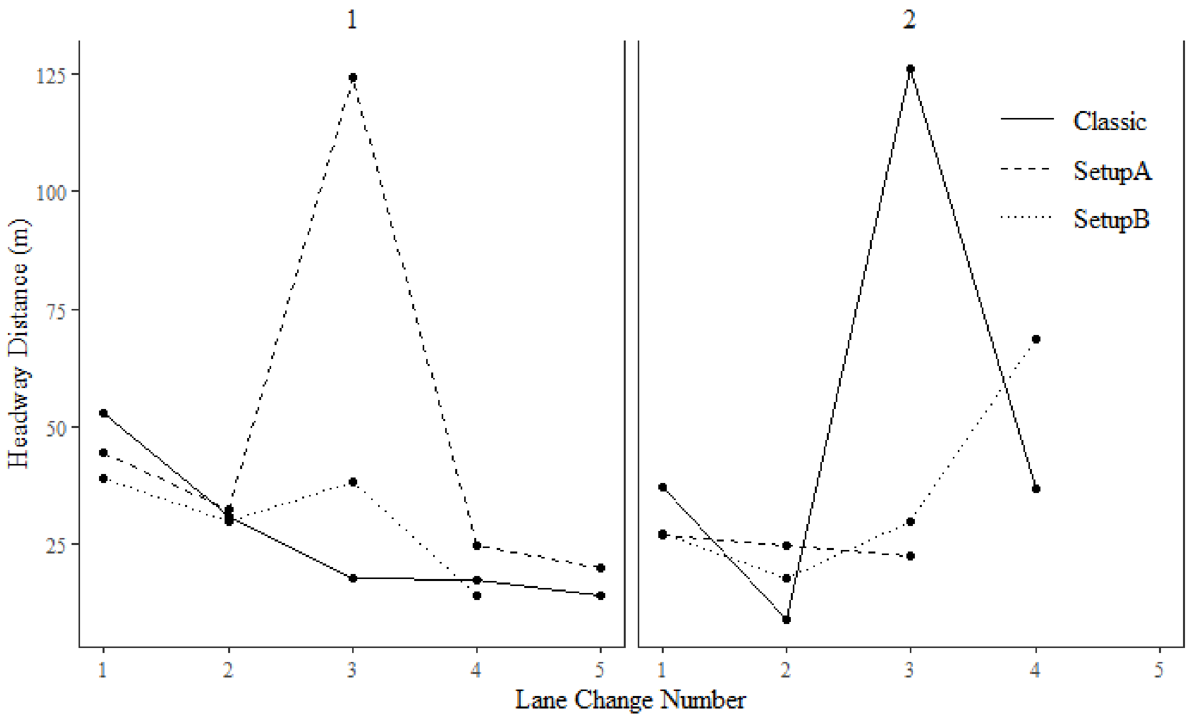


Figure 12. Headway distance for each sequential lane change and each setup used. The charts show the individual score of the two participants.

Table 5 presents the results of the regression analysis for the headway distance. The intercept represents the distance participant kept between the ego car and the following car with the classic setup when they made the first lane change. Based on the regression analysis, it is expected, that in the first session the headway distance is 2.71 meters less with Setup A and 19.20 meters less with Setup B. Subsequently, per session the distance is expected to decrease with 3.22 meters for the classic setup. For Setup A, the distance will decrease with 1.70 meters (-3.22 + 1.52) per session. For Setup B, it will increase with 2.81 per session (-3.22 + 6.03). However, the confidence intervals are broad-based which makes the results less certain. The predicted model is visualized in Figure 15.

Table 5
Results of Regression Analysis for Headway Distance (m) with 95% Confidence Intervals (Lower and Upper)

Parameter	Location	Lower CI	Upper CI
Intercept	46.43	-0.52	92.42
Group Setup A	-2.71	-69.49	61.13
Group Setup B	-19.20	-83.93	47.80
Lane change number	-3.22	-18.28	11.89
Group Setup A: Lane change number	1.52	-19.93	23.63
Group Setup B: Lane change number	6.03	-17.76	28.66

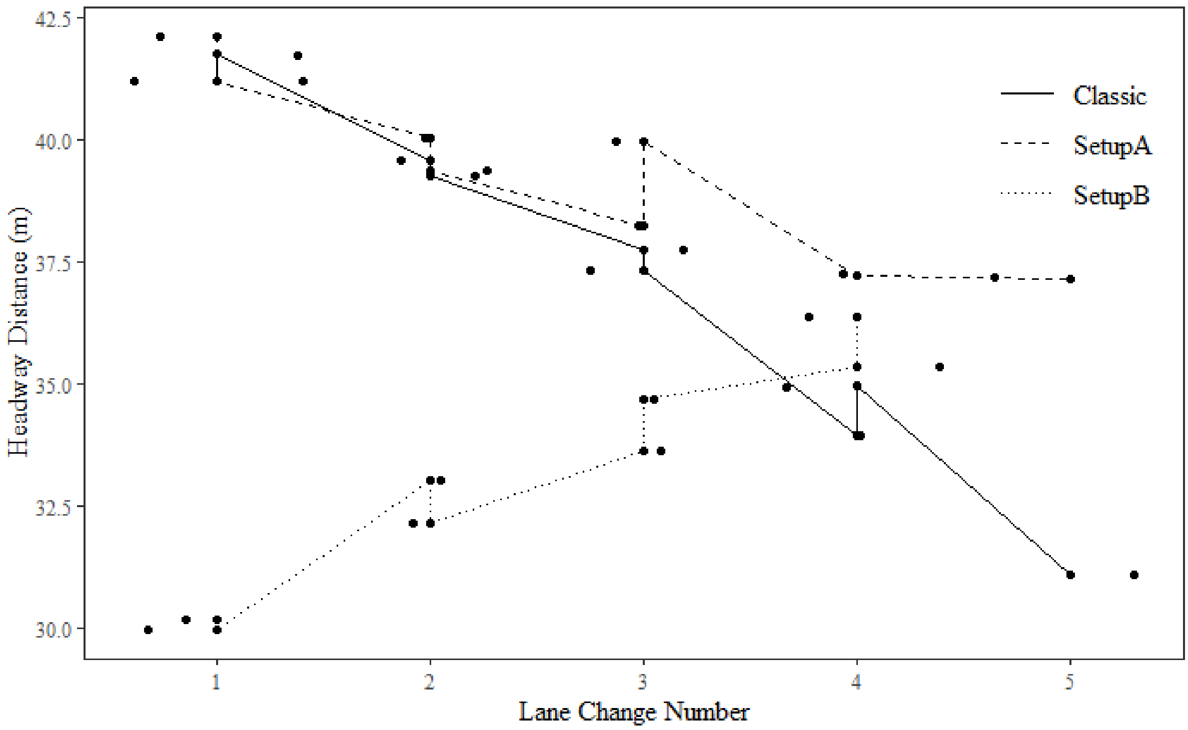


Figure 15. Prediction of headway distance for each sequential lane change per setup.

A Kruskal Wallis test was conducted to compare the effect of the mirror setup type on performance. The results indicated there was no significant effect of the mirror setup type on lane change time ($H(2) = 1.9917$, $p = .3694$) and headway distance ($H(2) = 0.224$, $p = .894$).

3.3.3 Eye-tracking data

The average fixation duration and time spent watching the mirrors were examined for each setup. A longer fixation time indicates a longer glance away from the road. Table 6, the averages and standard deviation for the eye-tracking data are presented.

Table 6

Averages and Standard Deviation for Fixation Duration (ms) and Percentage of Time Spent watching the Mirrors per Setup

	Average fixation (ms)		% Time spent	
Setup	*M*	*SD*	*M*	*SD*
Classic	146.00	48.05	2.08	1.22
Setup A	160.67	35.23	2.85	0.78
Setup B	193.50	53.41	3.12	1.22

A One-way ANOVA test was conducted to compare the effect of mirror setup type on gaze behaviour. The results indicated there was no significant effect of mirror setup type on time spent watching the mirrors ($F(2, 15) = 1.466$, $p = .262$) and average fixation duration ($F(2, 15)= 1.663$, $p = .223$).

The full R data analysis package can be found in Appendix L.

3.4 Discussion of Pilot Study

During the pilot study, the use of new digital mirror positions was investigated. Traditional mirror positions were compared with two experimental mirror setups (Figure 7 and 8). Additionally, a toolkit was created for conducting a full experiment on digital mirrors. The revised manual (Appendix E) and the R data analysis package (Appendix L) can be used to perform the virtual reality experiment. Regarding the pilot study, results indicated that there were no significant main effects of the mirror setup type on subjective ratings, performance and gaze behaviour for the three setups. This was most likely due to the low sample size. Nonetheless, other research outcomes of this study suggested that the new mirror positions can have a positive effect on workload, confidence and user experience and can improve the performance of the driver. The results showed that participants experienced the lowest workload while performing the tasks with use of Setup B. On the other hand, a higher score on confidence, usability and performance was observed with use of Setup A. Furthermore, it is expected that people needed to adjust to the new mirror positions as performance slightly improved after each session. These results advocate further research into use of Setup A. Still, the results showed a higher fixation duration for the new setups compared to the classic setup. Therefore, it would be useful to examine whether the average fixation duration will decrease after using the new setups over a longer period of time. The results of the pilot, in line with previous findings (Lamble et al., 1999; Large et al., 2016; Wittmann et al., 2006), seem to indicate that the use of new digital mirror positions can improve driving performance. This preliminary but relevant indication from the pilot should be further explored with a larger population. Future studies can focus on replicating the pilot study on a larger scale but can also extend it by adding more mirror setups to the experimental design. Besides, the use of other driving assisting tools can be tested e.g. by testing the use of perception cues (augmented reality) in digital mirrors to improve depth perception of the driver (Smith, Kane, Gabbard, Burnett, & Large, 2016).

Two limitations should be considered before conducting a full experiment. First, eye-tracking results are difficult to separate per task. It is to be advised to make one recording per task to obtain segmented results. Second, the performance data was manually recorded. This method of data collection may prove to be less accurate than an automatic data recording process. Furthermore, this method increases the risk of obtaining missing data. Therefore, further development of the system is needed to enable automatic recording of events.

4. Conclusion

By conducting the two studies, a base was created for researching driving performance at the University of Twente. First, by developing and testing the manual students have been empowered to independently conduct experiments with use of the simulator. As a result, less attention and intervention is required from the staff of the BMS lab which will save time and resources in the long run. Furthermore, an opportunity is created for safe and efficient execution of various driving-related experiments. Second, by performing the pilot study new insights are gained about the use of digital mirrors. The methods and procedures are released to allow for replication and extension of the experiment to further examine the benefits of digital mirrors. By performing research on the future of car displays a new and unexplored research field is entered. Initial findings serve as a basis for conducting automotive research in the field of human factors and engineering psychology. Expected is that future research will continue focussing on testing driver assisting technologies. Altogether, human factor research could deliver a significant contribution to the creation of a safer traffic environment and improve the wellbeing of drivers and passengers all over the world.

References

Flannagan, M. J., Sivak, M., Schumann, J., Kojima, S., & Traube, E. C. (1997). *Distance perception in driver-side and passenger-side convex rearview mirrors: Objects in mirror are more complicated than they appear* (Report No. UMTRI-97-32). Retrieved from https://deepblue.lib.umich.edu/bitstream/handle/2027.42/49363/UMTRI-97-32.pdf?sequence=1&isAllowed=y

Godley, S. T., Triggs, T. J., & Fildes, B. N. (2002). Driving simulator validation for speed research. *Accident analysis & prevention, 34*(5), 589-600. doi:10.1016/S0001-4575(01)00056-2

Hahnel, U. J., & Hecht, H. (2012). The impact of rear-view mirror distance and curvature on judgements relevant to road safety. *Ergonomics, 55*(1), 23-36. doi:10.1016/j.apergo.2013.04.008

Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Advances in psychology, 52*, 139-183. doi:10.1016/S0166-4115(08)62386-9

Karreman, J., Ummelen, N., & Steehouder, M. (2005). *Procedural and declarative information in user instructions: What we do and don't know about these information types*. Paper presented at the International Professional Communication Conference, Limerick, Ireland.

Kipyegen, N. J., & Korir, W. P. (2013). Importance of software documentation. *International Journal of Computer Science Issues (IJCSI), 10*(5), 223.

Lamble, D., Laakso, M., & Summala, H. (1999). Detection thresholds in car following situations and peripheral vision: Implications for positioning of visually demanding in-car displays. *Ergonomics, 42*(6), 807-815. doi:10.1080/001401399185306

Large, D. R., Crundall, E., Burnett, G., Harvey, C., & Konstantopoulos, P. (2016). Driving without wings: The effect of different digital mirror locations on the visual behaviour, performance and opinions of drivers. *Applied ergonomics, 55*, 138-148. doi:10.1016/j.apergo.2016.02.003

Lavery, D., Cockton, G., & Atkinson, M. P. (1997). Comparison of evaluation methods using structured usability problem reports. *Behaviour & Information Technology, 16*(4-5), 246-266. doi:10.1080/014492997119824

Madigan, R., Louw, T., & Merat, N. (2018). The effect of varying levels of vehicle automation on drivers' lane changing behaviour. *PloS one, 13*(2), 1-17. doi:10.1371/journal.pone.0192190

Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist, 38*(1), 43-52. doi:10.1207/S15326985EP3801_6

McLaughlin, S. B., Hankey, J. M., Green, C. A., & Kiefer, R. J. (2003). *Driver performance evaluation of two rear parking aids.* Paper presented at the 18th international technical conference on the enhanced safety of vehicles, Washington, DC.

Mercan, Ş., & Durdu, P. O. (2017). *Evaluating the Usability of Unity Game Engine from Developers' Perspective*. Paper presented at the 2017 IEEE 11th International Conference on Application of Information and Communication Technologies (AICT), Moscow, Russia.

National Highway Traffic Safety Administration. (2013). *Visual-manual NHTSA driver distraction guidelines for in-vehicle electronic devices* (NHTSA-2010-0053). Retrieved from https://www.govinfo.gov/content/pkg/FR-2013-04-26/pdf/2013-09883.pdf

National Highway Traffic Safety Administration. (2017). *USDOT Releases 2016 Fatal Traffic Crash Data*. Retrieved from https://www.nhtsa.gov/press-releases/usdot-releases-2016-fatal-traffic-crash-data

Schrepp, M., Hinderks, A., & Thomaschewski, J. (2017). Design and Evaluation of a Short Version of the User Experience Questionnaire (UEQ-S). *IJIMAI, 4*(6), 103-108. doi:10.9781/ijimai.2017.09.001

Sevinc, V., & Berkman, M. I. (2020). Psychometric evaluation of Simulator Sickness Questionnaire and its variants as a measure of cybersickness in consumer virtual environments. *Applied ergonomics, 82*, 102958. doi:10.1016/j.apergo.2019.102958

Simons-Morton, B. G., Guo, F., Klauer, S. G., Ehsani, J. P., & Pradhan, A. K. (2014). Keep your eyes on the road: Young driver crash risk increases according to duration of distraction. *Journal of Adolescent Health, 54*(5), 61-S67. doi:10.1016/j.jadohealth.2013.11.021

Smith, M., Kane, V., Gabbard, J. L., Burnett, G., & Large, D. R. (2016). Augmented mirrors: Depth judgments when augmenting video displays to replace automotive mirrors. *Human Factors and Ergonomics Society Annual Meeting, 60*(1), 1590-1594. doi:10.1177/1541931213601367

Wittmann, M., Kiss, M., Gugg, P., Steffen, A., Fink, M., Pöppel, E., & Kamiya, H. (2006). Effects of display position of a visual in-vehicle task on simulated driving. *Applied ergonomics, 37*(2), 187-199. doi:10.1016/j.apergo.2005.06.002

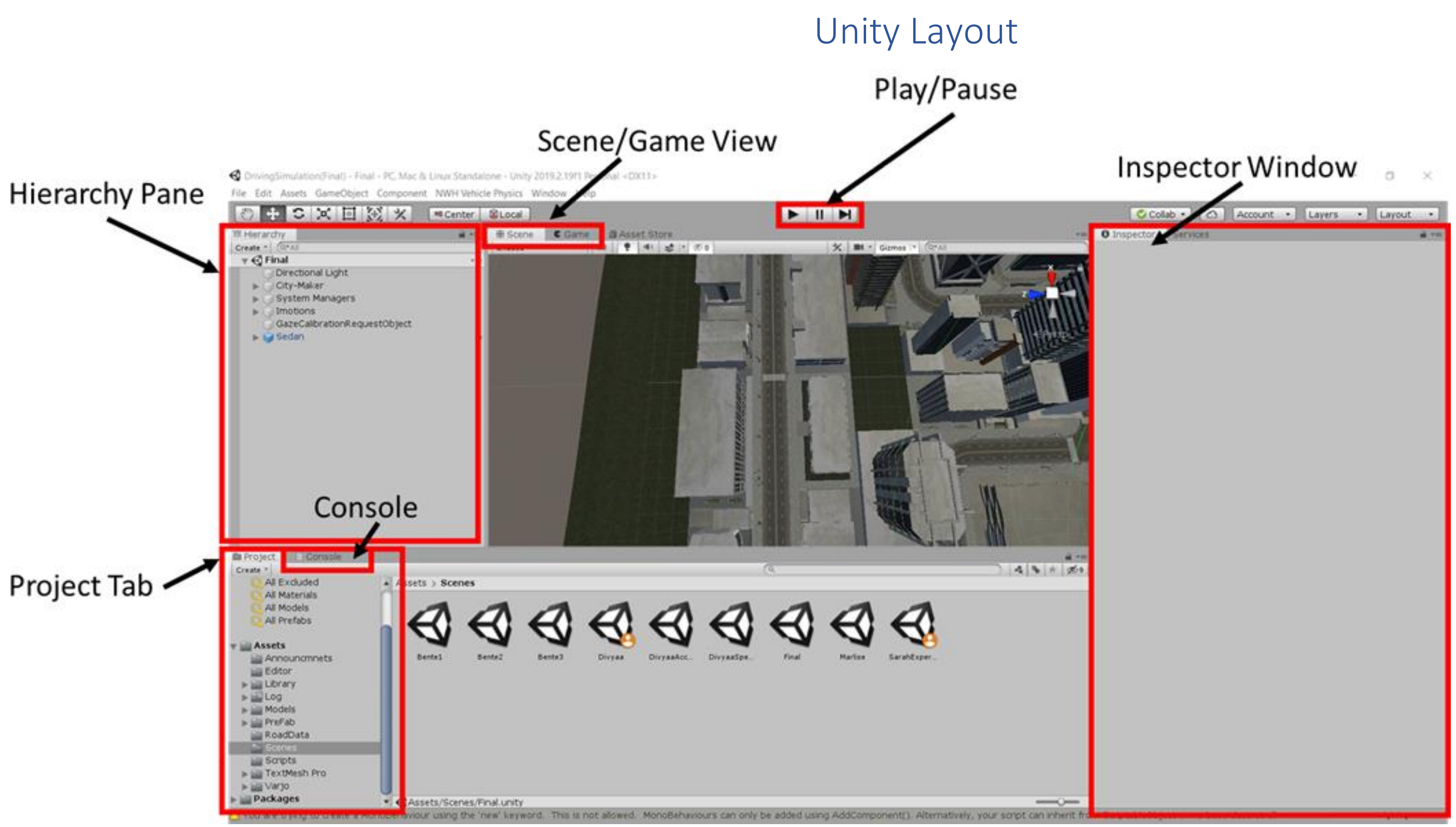
World Health Organisation. (2018). *Road traffic injuries*. Retrieved from https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

Appendix A

User Manual

Contents

Unity Layout 34
Setting up your experiment 36
Scene (or Environment) 36
Selecting the environment 36
Checkpoint Manager 36
Triggering events 36
Car 38
Inserting car into Scene 38
Making changes in the car 38
Controller settings 38
Car settings 38
Modifying the car (e.g. adding, removing, changing, etc.) 39
Traffic (optional) 40
Activate Traffic Light Objects 40
Traffic Spawner 40
Activate Traffic Light Behaviour 41
Setting up an experimental run 42
1. Calibrate: Logitech G920 Controller 42
2. Calibrate: Varjo Camera Location (special case only) 42
3. Output Log 43
4. Set-up: iMotions (optional: in case of eye-tracking) 43
Output unity 44
File 44
Variables 44
Events 45
Manually recorded 45
Automatically recorded 46
Exporting and analysing Imotions data 47
Exporting data 47
Analysing 47


Unity Layout
Play/Pause
Scene/Game View
Inspector Window
Hierarchy Pane
Console
Project Tab

Hierarchy Pane

The Scene name (e.g. Final) is at the top of the pane. Each cube (grey or blue) listed underneath refers to a game object present in the scene. If there is a grey arrow on the left of a game object, this is a dropdown which reveals that there are more (child) objects placed under the main (parent) object.

Inspector Window

When you click on a game object in the Hierarchy Pane, the Inspector Window shows all of the properties and scripts attached to the object.

Project Tab

All of the folders listed under the Project Tab refer to folders you can also find physically on your computer under the main folder named 'DrivingSimulation(final)'. Important folders include 'Scenes', 'Prefab' and 'Log'. When you click on a folder under Assets, the folder's contents are expanded in the window right next to it.

Console

When you switch to this tab after running the Scene, it prints warnings and errors as well as debug messages that can help you decipher what's wrong.

Scene and Game view

The Scene view allows you to see the world and objects in your Scene. When you run your scene, the Game view shows you what your Scene looks like to the participant/user.

Play/Pause

The Play button (right-facing triangle) runs the Scene. The Pause button (two vertical parallel lines) pauses the Scene after you run it. Setting up your experiment

Setting up your experiment

Scene (or Environment)

Selecting the environment

1) Open UnityHub and select the project ‘DrivingSimulation(Final)’.

2) Click: ‘Project’ -> ‘Assets’ -> ‘Scenes’

3) Identify the scene you wish to use (or create one in the same folder by duplicating the main scene and rename it as your own).

Checkpoint Manager

Introduction to the lane system

The roads within the scene are made up of lanes where every lane has its own number and location. To see this information, click: System Managers > iTSManager in the Hierarchy pane. Note: when you select the iTSManager game object, you can see that not the entire city is mapped by lane numbers. If you have a specific route in mind, make sure it is defined by only the lanes which are numbered. See the map of the lane numbers in Map Lanenumbers iTSManager. All lanes will be added in future developments.

With the Checkpoint Manager, a specific route consisting of checkpoints can be created by adding the corresponding sequence of lane numbers. It can be checked if the driver follows this route or not.

Create a route

1) Steps in Unity: ‘Hierarchy’ tab -> System Managers -> CheckPointManager -> ‘Inspector’ window.

2) Number of lanes: Check in the iTSManager, how many lanes your route consists of. This is the number you should fill in. When filling out a number, a box appears where you can fill in your lane number.

3) Lane number: Add a lane number you want to use in your route. All these lane numbers together create your route.

4) Click ‘initialize’

Generate checkpoints

- To generate the checkpoints in the scene, click ‘Generate’ at the bottom of the ‘Inspector’ window. Now you should be able to see several cubes along the roads you specified in your scene - these are the checkpoints.
- If you want to make changes to the lanes on your route -> Click ‘Clear Points’ before you make any changes to the lane order.

Note: if you want to use the checkpoints for something other than simply ensuring that the participant is following the route you want, refer to the section below on triggering events before you proceed to generating checkpoints.

Triggering events

During your experiment, you can set up different events to occur and record. These can be triggered manually by the researcher (i.e. using the keyboard or mouse) or automatically by the system if certain conditions are met. Examples of such events are described below but you are not limited to them.

Manually

It is possible to record events while the game is running, for example if you want to record at what time stamps a specific event was happening. For example, you can determine the time at which an individual is starting to and finishes making a lane change. This was implemented in a scene and is done by pressing the Z key while the game is running. The event will start. By pressing the M key, it will stop recording the event.

In addition to determining specific timestamps associated with events, the same scene also makes it possible to record the headway distance to the car that is in the back of the ego car:

- This can be recorded by pressing the B key.
- Make sure that you first press the Z key -> press the B key (several times) -> press the M key to finish the event.

See example → Click: Project > Assets > Scenes > MirrorExperiment

Automatically

It is also possible to trigger an event automatically when the car reaches a certain location in the scene. The checkpoint system discussed earlier can be used for this and there is an example scene in which this is implemented (see below).

1. After creation of your route and clicking 'initialize', points appear for every lane number. The number of check points appearing is dependent on the length of the lane.
2. Click 'Generate Points' to see the location of every points in the scene.
3. You can choose a point on the lane you want to add an event or announcement to:
    a. SP = start point
    b. MP = middle point
    c. EP = end point
4. Click the drop-down menu next to the chosen checkpoint, choose between:
    a. Announcement Point: When you click this, a bar appears in which you can add your announcement which you want to let appear on that point in the scene.
        i. Next, you can add the actual announcement in the 'Image' tab. This image can be imported in Unity via the navigation bar in the top: *Assets -> Import New Asset*
        ii. Save this image into your Project. Then drag the image from the location you saved it in Unity to the 'Image' tab in the Inspector window.
    b. Event Point: When you click this, you can link an event to this specific point in the scene.

See example → Click: Project > Assets > Scenes > Final

Car

Inserting car into Scene

- Click: Project Tab > Assets > PreFab
- Select the car of your choice (e.g. "Sedan") and drag it into the Hierarchy Pane
- Click on the game object in the Hierarchy Pane and make necessary changes using the Inspector Window

Making changes in the car

For every time you make changes in the car, this should NOT be done in the scene, but in the PREFAB of the car So, before you change anything in the car:

- Right mouse click on your Sedan in the 'Hierarchy' tab -> Open Prefab Asset
- Now, the prefab of your car opens without the surroundings of the scene.
- Make any changes you want

Controller settings

Transmission Type: A car with manual or automatic gear shift:

- Right mouse click on your Sedan in the Hierarchy tab -> Open Prefab Asset
- Inspector window -> Vehicle Controller -> Transmission -> Transmission type: You can choose between:
  - Manual: The driver shifts gear manually, with a gear stick and the clutch.
  - Automatic: The driver does not have to shift gear manually, this is done by the car.
  - Automatic sequential: With this, the driver still can shift the gear manually using pedals on the steering wheel or a shift lever. However, this is just a choice by the driver. In essence, the transmission is automatic.

Controller Type: You can choose if you want to control the animated car by the Logitech simulator (Steering Wheel control) or by your keyboard (Desktop control):

- Click in the 'Hierarchy' tab -> System Managers -> _VehicleManager
- In the 'Inspector' window you see different tabs
- Disable (deactivate the checkmark) Logitech Input Manager (Script)
- Enable (activate the checkmark) Desktop Input Manager (Script)

For advanced settings, such as engine power, brakes and steering, see the separate documentation file on Car controller settings.

Car settings

Set a speed limit: So, drivers do not drive too fast.

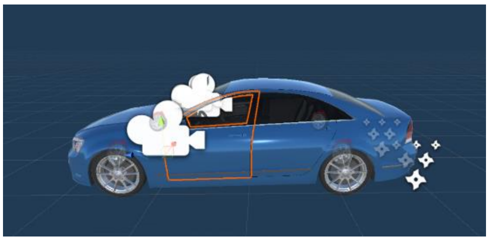
- Right mouse click on your Sedan in the Hierarchy tab -> Open Prefab Asset
- Inspector window -> Vehicle Controller -> General -> Speed Limiter

- The Speed Limiter is set in meters per second

Modifying the car (e.g. adding, removing, changing, etc.)

Selecting specific objects

- In order to find the prefab car, click: Project Tab > Assets > Prefab > Sedan
- Double click on the prefab car to open it up in edit mode.
- In the Hierarchy pane, use the drop-down menus to find the object you want to edit (e.g. Mirrors) and click on it. You can also click on parts of the car in the scene itself and the selected object will be surrounded by an orange outline).

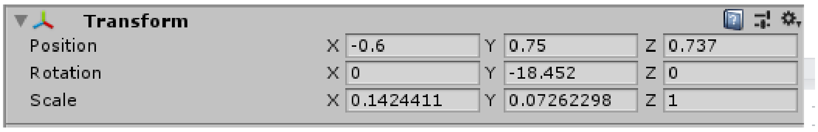


Note: remember that if you edit object in the scene, changes will not be saved. Edit objects in the prefab mode if you are sure of the changes and/or want changes to be long-lasting/permanent.

Editing objects

- By pressing the “W” key the position can be decided by dragging the object on the screen.
- By pressing the “E” key the rotation of the object can be determined.
- By pressing the “R” key the scale of the object can be determined

Note: the position, rotation and scale can also be decided in the transform menu by changing the numbers for the x,y and z axis (can be useful to make sure that e.g. the mirrors are on the same height).

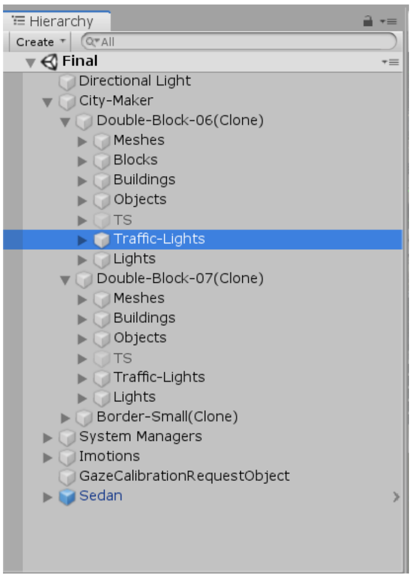


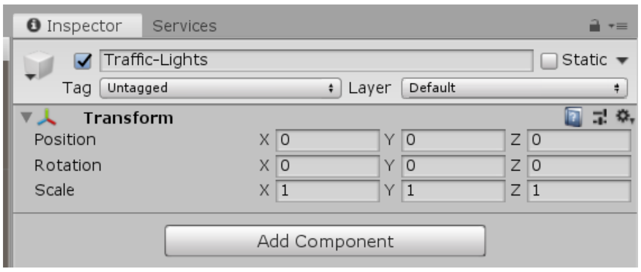
3D Objects can be added to the car by dragging and dropping them from folders within the Project tab into the Hierarchy.

Traffic (optional)

Activate Traffic Light Objects

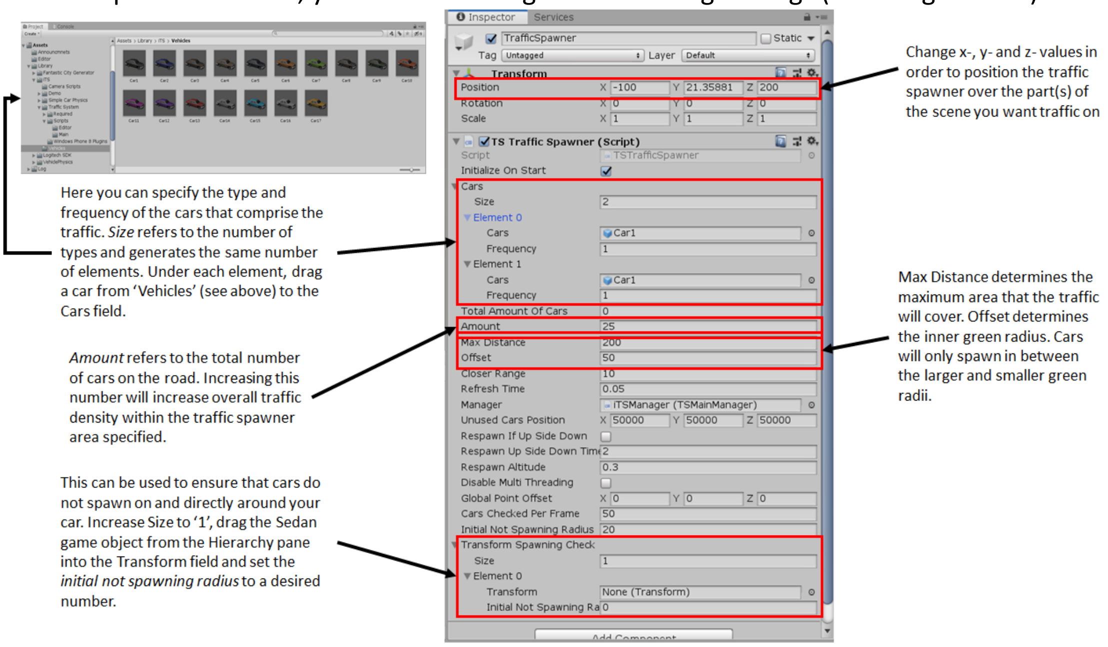
1) In the Hierarchy pane, click: City-Maker > Double-Block-06(Clone) > Traffic-Lights and click on the game object (as highlighted below)
2) In the Inspector Window, make sure that the checkbox beside "Traffic-Lights" is checked. If it is not, check it.
3) Repeat as follows for the other set of traffic lights present in Double-Block-07(Clone).
4) Check to see if all traffic lights are visible in the environment.





Traffic Spawner

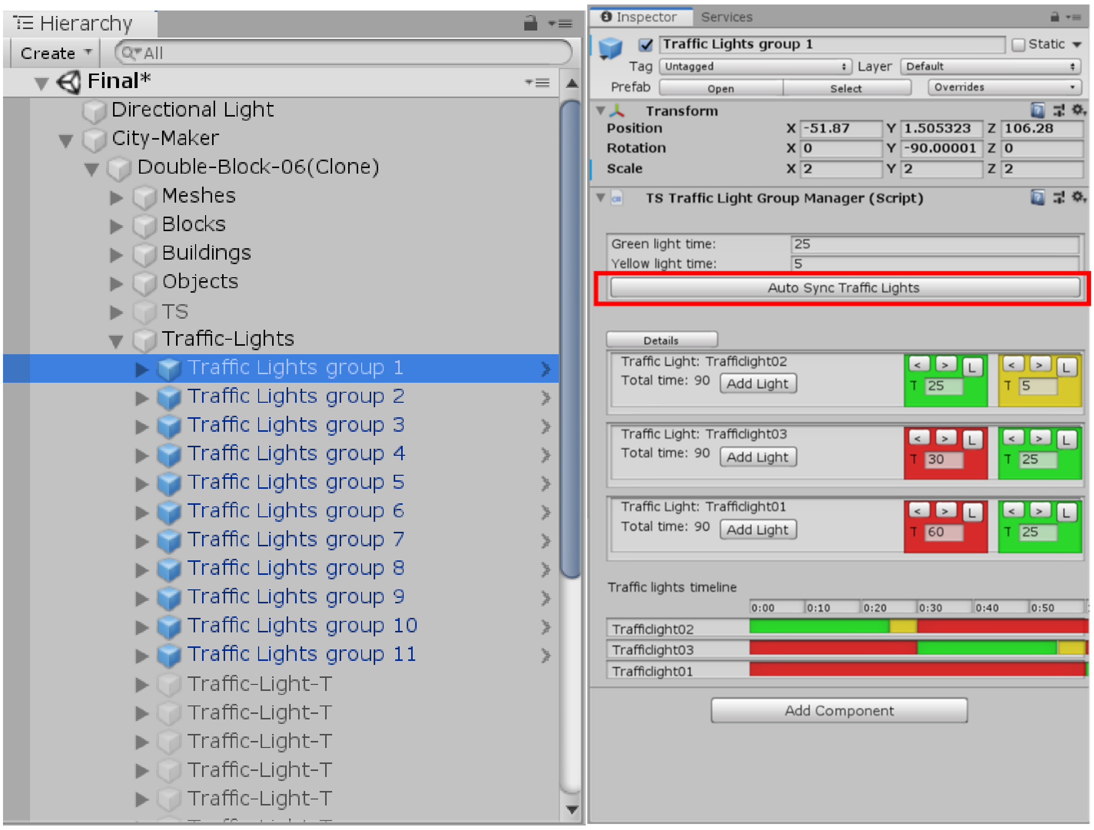
1) In the Hierarchy Pane, click: System Managers > Traffic Spawner
2) In the Inspector Window, make sure that the checkbox beside "TrafficSpawner is checked. If it is not, check it.
3) In the Inspector Window, you can also change the following settings (see image below).



Activate Traffic Light Behaviour

In order to ensure that the traffic lights operate consistently and in sync with each other, the following steps should be carried out:

1) In the Hierarchy pane, click: City-Maker > Double-Block-06(Clone) > Traffic-Lights > Traffic Lights group 1
2) In the Inspector Window, click on *Auto Sync Traffic Lights.*
3) Repeat steps 1 and 2 for every traffic lights group in Double-Block-06(Clone) and Double-Block-07(Clone).



Setting up an experimental run

These steps should be followed in the correct order as they are explained when starting an experimental run.

1. Calibrate: Logitech G920 Controller

These steps should be followed only at the start of your first experimental run of the day, or when you notice that the steering wheel is not in line with the animation.

- Find the Search bar located in the bottom-left corner of your screen beside the Windows Logo.
- In the Search bar, type ‘Set up USB game controllers’ and select this option.
- Click: ‘Properties’ -> ‘Settings’ -> ‘Reset to default’ -> ‘Calibrate’
- Follow the instructions presented to you.
- When finished, click ‘Apply’.

2. Calibrate: Varjo Camera Location (special case only)

Every time (a) the simulator or (b) the VR cameras physically move in the room, the view that is seen through the VR headset when Unity is run might change.

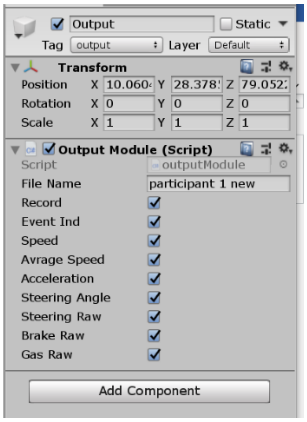
Check to see if the view is still in place by wearing the VR headset and running the scene.

If it is not in place:

- Determine in which way the camera needs to be moved in order for it to be correctly positioned.
- End Scene.
- Find and click on the GameObject ‘VarjoUser’ in the Hierarchy Tab. Place your cursor on the Scene view and press F to find the object on the Scene.
- Click on the ‘Move’ button (two arrows intersecting each other) along the top or press ‘W’ – three coloured arrows will appear around the object you have selected.
- Select the appropriate arrow (based on the axis along which you want to move the object) and change the position of the camera.
- Run Scene and check to see if the position is now correct. Repeat until satisfied.

3. Output Log

From Unity you get the data from the behaviour of the car during the experiment. You find the output log in the 'Hierarchy' tab -> System Managers -> Output. Within the 'Inspector' window you must rename your output in 'File Name' and click all the logs you want to have output from, for example 'Speed', 'Steering Angle' and 'Gas Raw'. **Change the file name to something appropriate EVERY time you run a new session!**



4. Set-up: iMotions (optional: in case of eye-tracking)

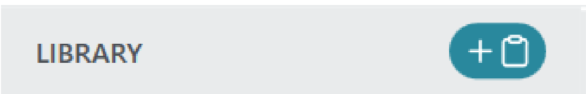
In case you use eye-tracking, follow these steps EVERY time you start running a new session!

For every time you add another prefab car in your scene, you need to do these steps:

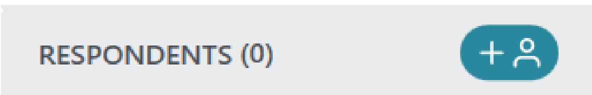
- Navigate to: Sedan -> VarjoUser -> VarjoCamera -> [Varjo PoV Camera]
- Imotions -> [Varjo Gaze Server] -> 'Inspector' window -> Point of View
- Drag the [Varjo PoV Camera} into the 'Point of View' in the Inpector tab

At the beginning of the study:

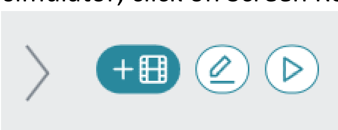
- Start iMotions software
- Create a study: In the left side of the screen click on the blue '+' -> Name your study -> Click next -> Click add



- Create a respondent: In the right side of the screen click on the blue '+' -> Name your respondent and add demographic information.



- Create a stimulus: Click on the blue '+' in the right upper corner. In case of working with the driving simulator, click on Screen Recording -> Click Add



For each participant:

- Run scene on Unity
- On the iMotions application, click on: 'preferences' -> 'reconnect sensors'.
- Calibrate the Varjo: If using eye-tracking, the Varjo headset needs to be calibrated every time the headset is removed. There are two ways to calibrate the eye-tracker:
    - Method 1: Through VarjoBase:
        - Click on Unity
        - Press the space bar. The calibration process will be activated by itself and the user can calibrate the eye-tracker. The left button (Application button) on the headset is 'Okay'.
    - Method 2: Through the headset:
        - Place the headset on your head.
        - Press the right button (System button) on the headset.
        - A navigation menu appears, click on Calibration -> Eye tracking -> Start
        - The calibration process starts.
- Start recording in iMotions
- Drag the Game Scene of Unity to the screen that will be recorded by the iMotions.
- Go back to Unity, and leave Unity opened during the entire recording, so the participant can keep the car in control.
- When the session is finished, stop the recording and the recording will be saved in your study.

Run: Unity Scene

- Press the Play button and click on the Game tab in order to make sure the participant can control the car.

Output unity

To record data, go to section *output log* in *Setup and experimental run* in the manual.

File

After running the game, the datafile can be found in the project folder under *'Assets' -> 'Log'*

Variables

The following data can be gathered.

- Time stamps in seconds
- Events (can be recorded manually, by pressing the z key, or automatically)
- Speed in Km/h
- Average speed in Km/h (Avg speed)
- Acceleration in m/s
- Steering Angle
- Raw steering values
- Raw gas values

- Raw brake values

After running the game, the data is saved in an excel file.

	A	B	C	D	E	F	G	H	I
1	TimeStam	Event	Speed	AvgSpeed	Acceleration	Steering Angle	SteeringRaw	GasRaw	BrakeRaw
2	0		0	0	0	0	0	0	0

Each value is linked to a time stamp which is displayed in the first column in the excel file. The time stamp in column 1 is connected to the data in the other columns. For example, the speed at 63,6 seconds in the game was approximately 45,70 km/h (856, column C).

A858 | 63,6

	A	B	C	D	E	F	G	H	I	J
856	63,51		45,70372	0,520665	2,520879	0,259191	351	18061	32767	
857	63,555		46,12461	0,530863	2,635702	0,160624	218	18061	32767	
858	63,6		46,56333	0,541107	2,74334	-0,02646	-36	18061	32767	
859	63,66		47,17379	0,55146	2,874247	-0,24913	-340	18061	32767	
860	63,705		47,64914	0,561964	2,962685	-0,4093	-566	18061	32767	

Events

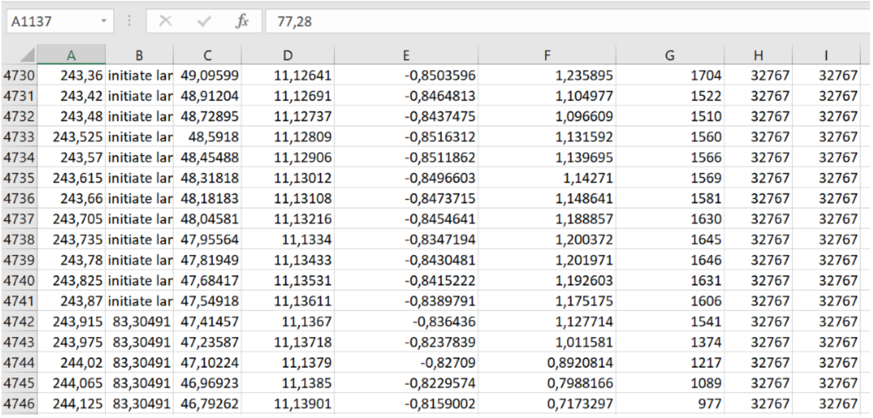
Manually recorded

By pressing the M key, it will stop printing the event in the excel file. This way you can see at which time stamps the event was happening. Here the event was triggered between the Z key was pressed at 77,28 second and finished when the M key was pressed at 77,895 second (row 1137 till 1156)

A1137 | 77,28

	A	B	C	D	E	F	G	H	I
1137	77,28	initiate lar	49,30546	3,540991	-0,9411494	2,942832	4062	32767	32767
1138	77,325	initiate lar	49,1592	3,548383	-0,9353002	2,900906	4001	32767	32767
1139	77,37	initiate lar	49,01362	3,555619	-0,9302775	2,853822	3933	32767	32767
1140	77,415	initiate lar	48,86868	3,562611	-0,9250641	2,7747	3821	32767	32767
1141	77,46	initiate lar	48,72451	3,569873	-0,9175619	2,744859	3777	32767	32767
1142	77,505	initiate lar	48,58096	3,577201	-0,9150823	2,741869	3770	32767	32767
1143	77,55	initiate lar	48,43773	3,584509	-0,9138743	2,7345	3757	32767	32767
1144	77,595	initiate lar	48,29481	3,591656	-0,9112676	2,700889	3708	32767	32767
1145	77,64	initiate lar	48,15241	3,598616	-0,9062449	2,640246	3622	32767	32767
1146	77,685	initiate lar	48,01073	3,605723	-0,8998235	2,5474	3492	32767	32767
1147	77,715	initiate lar	47,9181	3,613078	-0,8589427	2,442825	3347	32767	32767
1148	77,75999	initiate lar	47,77636	3,620196	-0,8818945	2,32926	3189	32767	32767
1149	77,805	initiate lar	47,63747	3,627049	-0,876236	2,29516	3140	32767	32767
1150	77,85	initiate lar	47,49902	3,633931	-0,8752823	2,296113	3139	32767	32767
1151	77,895	initiate lar	47,36074	3,640925	-0,875028	2,297795	3139	32767	32767
1152	77,93999		47,22268	3,647734	-0,8733749	2,299475	3139	32767	32767
1153	77,985		47,08492	3,654522	-0,8714041	2,301151	3139	32767	32767
1154	78,03		46,94748	3,661192	-0,8691788	2,297687	3127	32767	32767
1155	78,075		46,81039	3,667828	-0,8667628	2,281732	3108	32767	32767
1156	78,13499		46,62891	3,673779	-0,8302689	2,24644	3051	32767	32767

Make sure that you first press the Z key (row 4730), than press the B key (several times) (row 4742) and lastly press the M key to finish the event (row 4746). The distance from the back of the car and the vehicle behind here was 83,30m when the B key was pressed.



A1137 | 77,28

	A	B	C	D	E	F	G	H	I
4730	243,36	initiate lar	49,09599	11,12641	-0,8503596	1,235895	1704	32767	32767
4731	243,42	initiate lar	48,91204	11,12691	-0,8464813	1,104977	1522	32767	32767
4732	243,48	initiate lar	48,72895	11,12737	-0,8437475	1,096609	1510	32767	32767
4733	243,525	initiate lar	48,5918	11,12809	-0,8516312	1,131592	1560	32767	32767
4734	243,57	initiate lar	48,45488	11,12906	-0,8511862	1,139695	1566	32767	32767
4735	243,615	initiate lar	48,31818	11,13012	-0,8496603	1,14271	1569	32767	32767
4736	243,66	initiate lar	48,18183	11,13108	-0,8473715	1,148641	1581	32767	32767
4737	243,705	initiate lar	48,04581	11,13216	-0,8454641	1,188857	1630	32767	32767
4738	243,735	initiate lar	47,95564	11,1334	-0,8347194	1,200372	1645	32767	32767
4739	243,78	initiate lar	47,81949	11,13433	-0,8430481	1,201971	1646	32767	32767
4740	243,825	initiate lar	47,68417	11,13531	-0,8415222	1,192603	1631	32767	32767
4741	243,87	initiate lar	47,54918	11,13611	-0,8389791	1,175175	1606	32767	32767
4742	243,915	83,30491	47,41457	11,1367	-0,836436	1,127714	1541	32767	32767
4743	243,975	83,30491	47,23587	11,13718	-0,8237839	1,011581	1374	32767	32767
4744	244,02	83,30491	47,10224	11,1379	-0,82709	0,8920814	1217	32767	32767
4745	244,065	83,30491	46,96923	11,1385	-0,8229574	0,7988166	1089	32767	32767
4746	244,125	83,30491	46,79262	11,13901	-0,8159002	0,7173297	977	32767	32767

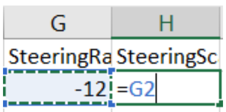
Automatically recorded

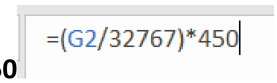
Events can also be recorded automatically with the use of the Checkpoint Manager in Unity. When the Checkpoint System is used, the Checkpoints will be printed in the datafile when these are hit in the scene.

Raw steering values

In column SteeringRaw, column G, the raw values of the steering wheel are printed. The range of these values are from –32767 to 32767. The maximum negative value is recorded when the physical steering wheel is totally turned to the left. The maximum positive value is recorded when the physical steering wheel is totally turned to the right.

To get the valuable steering angle out of these numbers, it is possible to rescale them in excel. The physical steering wheel has the possibility to turn 900° in total, so 450° to the left and 450° to the right. Therefore, the scale should be between –450 and 450. Steps in excel:

- Make a new column next to the SteeringRaw column and give this a preferred name.
- Type an '=' in the first cell of this new column
- Click the cell of the old column next to this 

- Type a new formula in the bar above the table: **=(*value*/32767)*450** 

- Push 'Enter'
- Click the right bottom corner of the cell with the new value and drag this down for the entire column. The other values will be calculated automatically.

Raw gas and brake values

In the column GasRaw and BrakeRaw, the raw values of the physical gas and brake throttles are printed. Just as the SteeringRaw, the range of these values are from –32767 to 32767. **Pay attention: -32767 means that the throttle is maximally pushed and 32767 means that the throttle is not pushed at all.**

Also, these values can be scaled to a valuable scale. In this case, the range from 0 to 100 would be valuable, in which 0 means that the throttle is maximally pushed and 100 means that the throttle is not pushed at all. Similar steps should be taken as in scaling the steering angle:

- Make a new column next to the GasRaw or BrakeRaw column and give this a preferred name.
- Type an '=' in the first cell of this new column
- Click the cell of the old column next to this

I	J
GasRaw	GasScaled
18577	=I2

- Type a new formula in the bar above the table: **=((*value*+32767)/(32767+32767))*100**

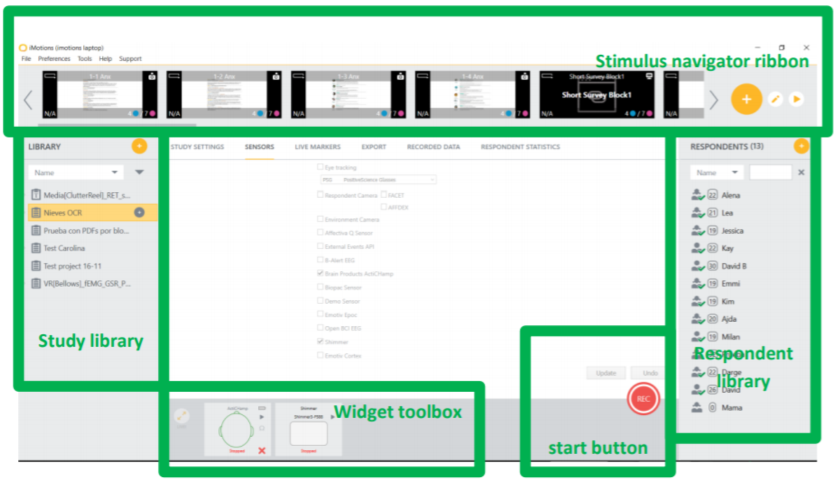
=((I2+32767)/(32767+32767))*100

- Push 'Enter'
- Click the right bottom corner of the cell with the new value and drag this down for the entire column. The other values will be calculated automatically.
-

Exporting and analysing Imotions data

Exporting data

Right click on the study you want to use in in the study library -> click on export sensor data -> select the element you want to export.



Analysing

For information on analysing data following link can be consulted:

help.imotions.com/hc/en-us/articles/213046585-Gaze-Mapping-

For the creation of AOI the following link can be consulted:

https://help.imotions.com/hc/en-us/articles/205144072-Automated-Areas-of-Interest-AOIs-

Appendix B

Informed Consent

Dear Participant,

You are invited to participate in the study called; "Future of car displays". This research is being done as part of my Master's thesis at the University of Twente. The aim of the research is to gain more insight into the usability of the manual that was created for the driving simulation at the university. This research has been approved by the ethics committee of the faculty; "Behavioural Management and social sciences".

The study will start with questions about personal data (place of residence and age etc.). Then, I will ask you to take behind the pc and perform several tasks. You will be given a scenario that will lead to a task that you will have to perform while using the unity software. You will have to use the manual to complete the tasks that are given.

Your participation in this study is entirely voluntary and you can stop at any time. You are free to refuse to answer questions.

Remember that we are investigating how the tool works. Poor performance will only indicate shortcomings in the system's usability.

We believe that there are no further risks associated with this investigation. However, we will do our best to ensure that your answers remain confidential and will not be used for other purposes.

Informed consent.

By signing this document you indicate that you are at least 16 years old; that you are well informed about the research and how the research data is collected, used and treated and what possible risks are involved by participating in this research.

If you had any questions, indicate when signing that you were able to ask these questions and that these questions were answered clearly. You indicate that you voluntarily agree with your participation in this study.

This research project is guided by Dr. Borsci. The purpose of this document is to establish the conditions of my participation in the project.

	Yes	No
1. I have received sufficient information about this research project. The purpose of my participation as a participant in this project has been clearly explained to me and I know what this means for me.		
2. My participation as a participant in this project is voluntary. There is no explicit or implicit compulsion to participate in this study.		
3. The experiment will take 30 minutes. I give the researcher permission to make screen and voice recordings and take written notes during the research. It is clear to me that if I object to certain issues, I can stop my participation at any time, without giving a reason.		
4. I have received the guarantee from the investigator that the investigator ensures that I am not identifiable in data, reports or articles brought forward by the investigation. My privacy is guaranteed as a participant in this study.		
5. I have been guaranteed that this research project has been reviewed and approved by the ethics committee of the BMS Ethics Committee. For objections with regard to the design and / or implementation of the research, I can contact the Secretary of the Ethics Committee of the faculty of Behavioural, Management and Social Sciences at the University of Twente via ethicscommittee-bms@utwente.nl.		
6. I have read and understood this form. All my questions have been answered to my satisfaction and I voluntarily agree to participate in this study.		

______________________ ______________________ ________

Name of participant Signature Date

______________________ ______________________ ________

Name Researcher Signature Date

Appendix C

Protocol

1. ***The participant is asked to read and sign the informed consent.***
2. ***Ask the participant demographic questions.***

Questionnaire participant no. ______

A. What is your age?

______ Year

B. What is your gender?

Male / Female

C. What is your country of birth?

____

D. Do you have experience with using Unity software?

YES / NO

3. ***Explain unity to the participant***

4. ***Let the participant take place behind the pc and give a scenario.***

You want to conduct a study in the driving simulator. For this study, you will use the unity game software. This software consists of a virtual world in which you can drive around in a created city. You will use the manual to make some adjustments in the scene to fit your experiment;

- *You want to change maximum speed of the Car (the sedan). Where can you change this? SOLUTION: GO TO SEDAN IN THE HIERARCHY TAB -> OPEN PREFAB ASSET -> INSPECTOR WINDOW -> VEHICLE CONTROLLER -> GENERAL -> SPEED LIMITER*
- *You want to adjust the position of the car on the map to an arbitrary new position. SOLUTION: CLICK ON SEDAN-> USE W, E AND R KEY TO CHANGE POSITION OF THE CAR ON THE MAP. OR*
  CLICK ON SEDAN -> CHANGE VALUES OF TRANSFORM .
- *You want to adjust the traffic situation. Try to add traffic to the scene and make the density higher.*
  SOLUTION: HIERARCHY-> TRAFFIC -> CHECK BOX TS TRAFFIC SPAWNER ->AJUST NUMBER IN AMOUNT
- *You want to know afterward what the speed of the car was at each moment. Try to let the unity system record this in an excel file.*
  SOLUTION: OUTPUT MODULE -> CHECK BOX OUTPUT MODULE -> CHECK BOXES OF SPEED

5. ***Start of recording***

6. ***Ask questions***

1) What was your overall experience using the manual?
2) Where you able to perform the tasks properly while using the manual?
3) What recommendations would you give to improve the use of the manual?
4) Was the manual helpful in accomplishing the tasks?

7. ***Thank the participant for participation.***

Appendix D

Problem Description Form

Problem #		Incidents
Cause (design issue)		
Breakdown (observed intercation)		
Outcomes (expected loss, impact of problem)		
Design change (how can the problem be solved)		

Appendix E

User Manual Adjusted

User manual

Driving simulator

Content

READ ME FIRST! 56

Hierarchy Pane 57
Inspector Window 57
Project Tab 57
Console 57
Scene and Game view 57
Play/Pause 57

Preparing your experiment 58

Scene (or Environment) 58
Selecting the environment 58
Introduction to the lane system 58
Create a route 58
Generate checkpoints 59

Triggering events 59
Manually 59
Automatically 59

Car 61
Inserting car into Scene 61
Making changes in the car 61
Controller settings 61
Car settings 62
Modifying the car (e.g. adding, removing, changing, etc.) 62
Selecting specific objects 62
Changing Position, Rotation and Scale of Objects 62

Traffic (optional) 64
Activate Traffic Light Objects 64
Activate Traffic Light Behaviour 65
Add Traffic 65

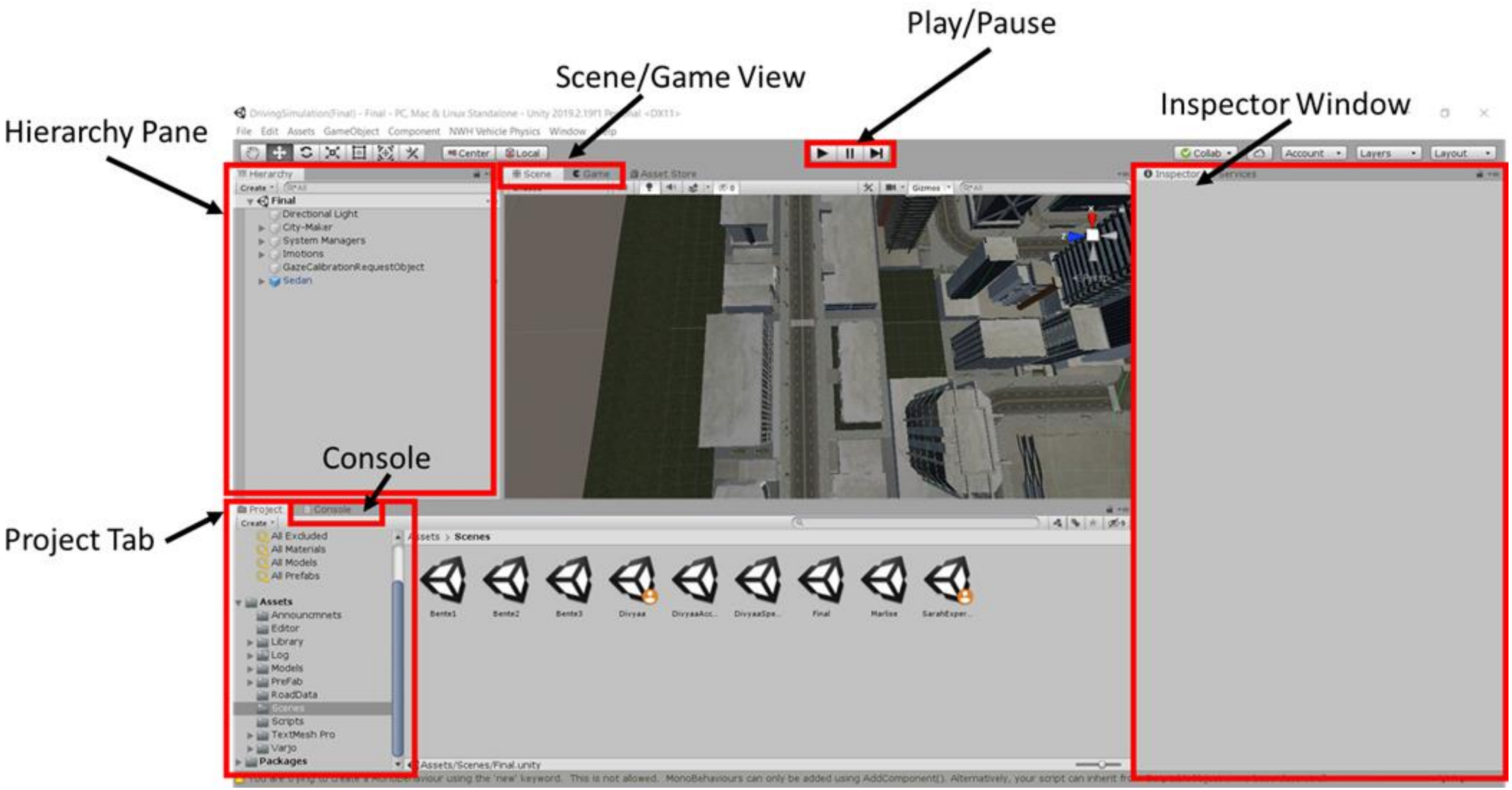
Starting an experimental run 67

Calibrate: Logitech G920 Controller 67
Calibrate: Varjo Camera Location (special case only) 67
Output Log 68
Set-up: iMotions (optional: in case of eye-tracking) 68
At the beginning of the study: 69
For each participant: 69
Run: Unity Scene 70

Output unity 71

File 71
Variables 71
Events 72
Manually recorded 72
Automatically recorded 73

READ ME FIRST!



Hierarchy Pane

The Scene name (e.g. Final) is at the top of the pane. Each cube (grey or blue) listed underneath refers to a game object present in the scene. If there is a grey arrow on the left of a game object, this is a dropdown which reveals that there are more (child) objects placed under the main (parent) object.

Inspector Window

When you click on a game object in the Hierarchy Pane, the Inspector Window shows all of the properties and scripts attached to the object.

Project Tab

All of the folders listed under the Project Tab refer to folders you can also find physically on your computer under the main folder named '**DrivingSimulation(final)**'. Important folders include 'Scenes', 'Prefab' and 'Log'. When you click on a folder under Assets, the folder's contents are expanded in the window right next to it.

Console

When you switch to this tab after running the Scene, it prints warnings and errors as well as debug messages that can help you decipher what's wrong.

Scene and Game view

The Scene view allows you to see the world and objects in your Scene. When you run your scene, the Game view shows you what your Scene looks like to the participant/user.

Play/Pause

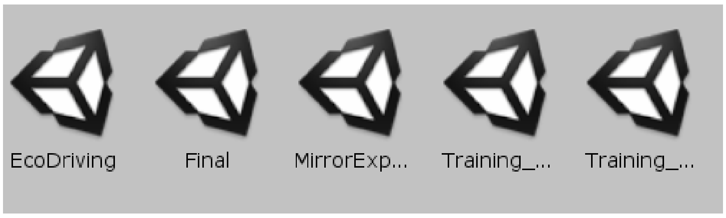
The Play button (right-facing triangle) runs the Scene. The Pause button (two vertical parallel lines) pauses the Scene after you run it. Setting up your experiment

Preparing your experiment

Scene (or Environment)

Selecting the environment

1) Open UnityHub and select the project **'DrivingSimulation(Final)'.**
2) Click:
'Project' -> 'Assets' -> 'Scenes'
3) Identify the scene you wish to use (or create one in the same folder by duplicating the main scene and rename it as your own).



Checkpoint Manager

Introduction to the lane system

The roads within the scene are made up of lanes where every lane has its own number and location. To see this information, click:
System Managers > iTSManager in the Hierarchy pane.

Note: when you select the iTSManager game object, you can see that not the entire city is mapped by lane numbers. If you have a specific route in mind, make sure it is defined by only the lanes which are numbered. See the map of the lane numbers in Map Lanenumbers iTSManager. All lanes will be added in future developments.

With the Checkpoint Manager, a specific route consisting of checkpoints can be created by adding the corresponding sequence of lane numbers. It can be checked if the driver follows this route or not.

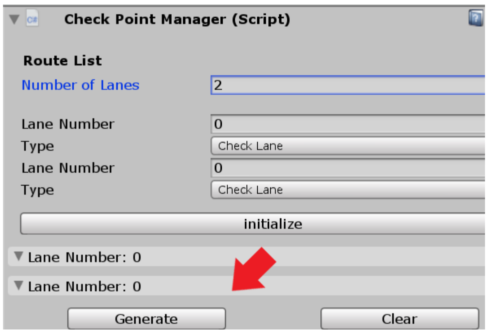
Create a route

1) Steps in Unity:
'Hierarchy' tab -> System Managers -> CheckPointManager -> 'Inspector' window.

2) Number of lanes: Check in the iTSManager, how many lanes your route consists of. This is the number you should fill in. When filling out a number, a box appears where you can fill in your lane number.
3) Lane number: Add a lane number you want to use in your route. All these lane numbers together create your route.

4) Click 'initialize'

Generate checkpoints



- To generate the checkpoints in the scene, click '**Generate**' at the bottom of the 'Inspector' window. Now you should be able to see several cubes along the roads you specified in your scene - these are the checkpoints.
- If you want to make changes to the lanes on your route -> Click '**Clear Points'**
  before you make any changes to the lane order.

Note: if you want to use the checkpoints for something other than simply ensuring that the participant is following the route you want, refer to the section below on triggering events before you proceed to generate checkpoints.

Triggering events

During your experiment, you can set up different events to occur and record. These can be triggered manually by the researcher (i.e. using the keyboard or mouse) or automatically by the system if certain conditions are met. Examples of such events are described below but you are not limited to them.

Manually

It is possible to record events while the game is running, for example, if you want to record at what timestamps a specific event was happening. For example, you can determine the time at which an individual is starting to and finishes making a lane change. This was implemented in a scene and is done by pressing the Z key while the game is running. The event will start. By pressing the M key, it will stop recording the event.

In addition to determining specific timestamps associated with events, the same scene also makes it possible to record the headway distance to the car that is in the back of the ego car:

- This can be recorded by pressing the B key.
- Make sure that you **first press the Z key -> press the B key (several times) -> press the M** key to finish the event.

See example → Click: **Project > Assets > Scenes > MirrorExperiment**

The next time you run the game the data will be saved in an excel file (check *''Output Unity'' in manual)*

Automatically

It is also possible to trigger an event automatically when the car reaches a certain location in the scene. The checkpoint system discussed earlier can be used for this and there is an example scene in which this is implemented (see below).

1) After the creation of your route and
Clicking **'initialize'**, points appear for every lane number. The number of checkpoints appearing is dependent on the length of the lane.
2) Click **'Generate Points'**
to see the location of every point in the scene.
3) You can choose a point on the lane you want to add an event or announcement to:
   a. SP = start point
   b. MP = middle point
   c. EP = endpoint
4) Click the drop-down menu
next to the chosen checkpoint, choose between:
   a. Announcement Point: When you click this, a bar appears in which you can add your announcement which you want to let appear on that point in the scene.

i. Next, you can add the actual announcement in the 'Image' tab. This image can be imported in Unity via the navigation bar in the top:

Assets -> Import New Asset

ii. Save this image into your Project. Then drag the image from the location you saved it in Unity to the 'Image' tab in the Inspector window.

b. Event Point: When you click this, you can link an event to this specific point in the scene.

See example → Click: **Project > Assets > Scenes > Final**

The next time you run the game the data will be saved in an excel file (check ''*Output Unity*'' in manual)

Car

Inserting car into Scene

- Click:

  Project Tab > Assets > PreFab

- Select the car of your choice (e.g. "Sedan") and drag it into the Hierarchy Pane

- Click on the game object (e.g. the car "Sedan")in the Hierarchy Pane and make necessary changes using the Inspector Window

Making changes in the car

For every time you make changes in the car, this should NOT be done in the scene, but in the PREFAB of the car So, before you change anything in the car:

- Right mouse click on your Sedan in the '**Hierarchy' tab -> Open Prefab Asset**
- Now, the prefab of your car opens without the surroundings of the scene.
- Make any changes you want

Controller settings

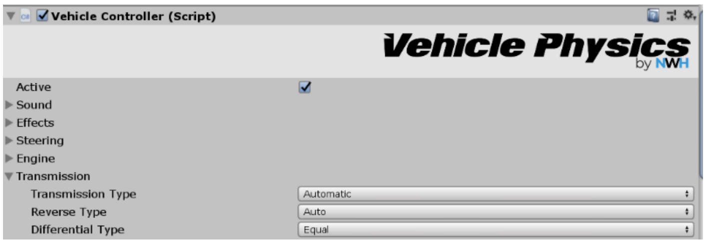
Transmission Type: A car with manual or automatic gear shift:

- Right mouse click on your Sedan in the

  Hierarchy tab -> Open Prefab Asset

- **Inspector window -> Vehicle Controller -> Transmission -> Transmission type:**

- You can choose between:



  - Manual: The driver shifts gear manually, with a gear stick and the clutch.
  - Automatic: The driver does not have to shift gear manually, this is done by the car.
  - Automatic sequential: With this, the driver still can shift the gear manually using pedals on the steering wheel or a shift lever. However, this is just a choice by the driver. In essence, the transmission is automatic.
  - The next time you run the game the changes will be incorporated.

Controller Type: You can choose if you want to control the animated car by the Logitech simulator (Steering Wheel Control) or by your keyboard (Desktop control):

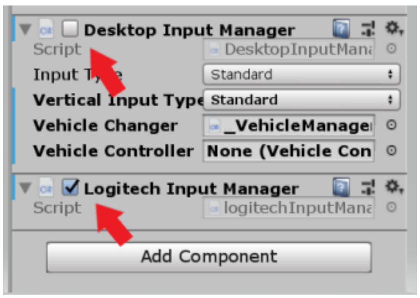
- Click in the:

  'Hierarchy' tab -> System Managers -> _VehicleManager

- In the '**Inspector'** window you see different tabs

- Disable (deactivate the checkmark) Logitech Input Manager (Script)

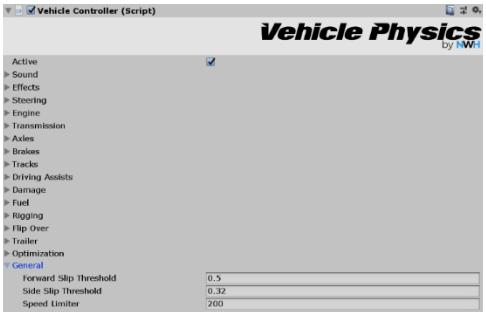
- Enable (activate the checkmark) Desktop Input Manager (Script)



For advanced settings, such as engine power, brakes and steering, see the separate documentation file on Car controller settings.

Car settings

Set a speed limit: So, drivers do not drive too fast.



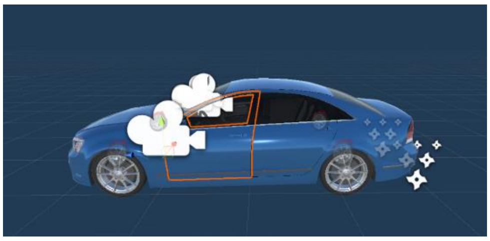
- Right mouse click on your Sedan in the

  Hierarchy tab -> Open Prefab Asset
- **Inspector window -> Vehicle Controller -> General -> Speed Limiter**
- The Speed Limiter is set in meters per second
- Next time you run the game the speed limiter will be activated.

Modifying the car (e.g. adding, removing, changing, etc.)

Selecting specific objects

- In order to find the prefab car, click: **Project Tab > Assets > Prefab > Sedan**
- Double click on the prefab car to open it up in edit mode.
- In the Hierarchy pane, use the drop-down menus to find the object you want to edit (e.g. Mirrors) and click on it. You can also click on parts of the car in the scene itself and the selected object will be surrounded by an orange outline).



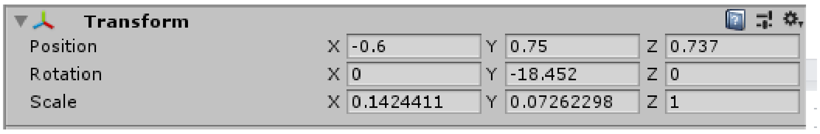
Note: remember that if you edit an object in the scene, changes will not be saved. Edit objects in the prefab mode if you are sure of the changes and/or want changes to be long-lasting/permanent.

Changing Position, Rotation and Scale of Objects

First, select the object that you want to edit in the '**Hierarchy**' tab

- By pressing the "W" key or the  button (left corner)the position can be decided by dragging the object on the screen.
- By pressing the "E" key or the  button (left corner) the rotation of the object can be determined.
- By pressing the "R" key or the  button (left corner) the scale of the object can be determined

Note: the position, rotation and scale can also be decided in the transform menu in the '**Inspector**' window by changing the numbers for the x,y and z-axis (can be useful to make sure that e.g. the objects are on the same height).



3D Objects can be added to the car by dragging and dropping them from folders within the '**Project tab**' into the '**Hierarchy**' tab.

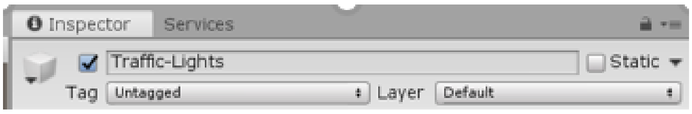
Traffic (optional)

Activate Traffic Light Objects

1) In the Hierarchy pane, click: **City-Maker > Double-Block-06(Clone) > Traffic-Lights**

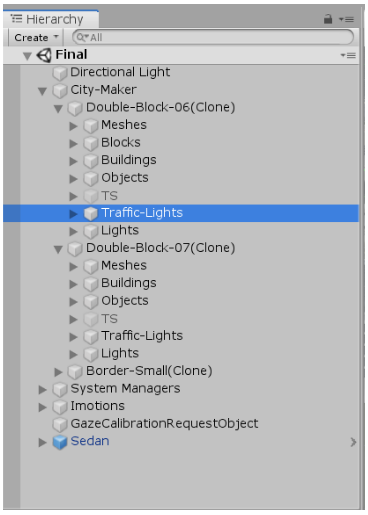
   and click on the game object (as highlighted below)

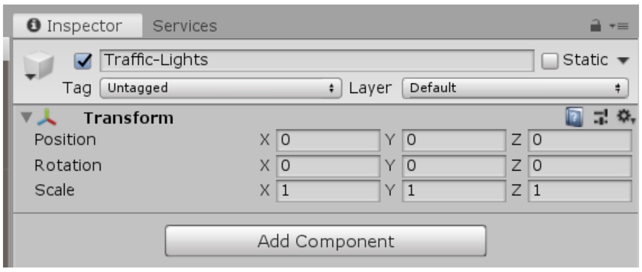
2) In the Inspector Window, make sure that the checkbox beside "**Traffic-Lights**" is checked. If it is not, check it.



3)Repeat as follows for the other set of traffic lights present in Double-Block-07(Clone).

4) Check to see if all traffic lights are visible in the environment.

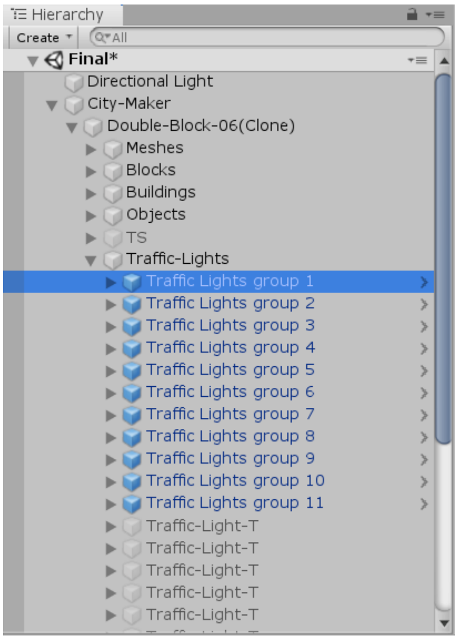


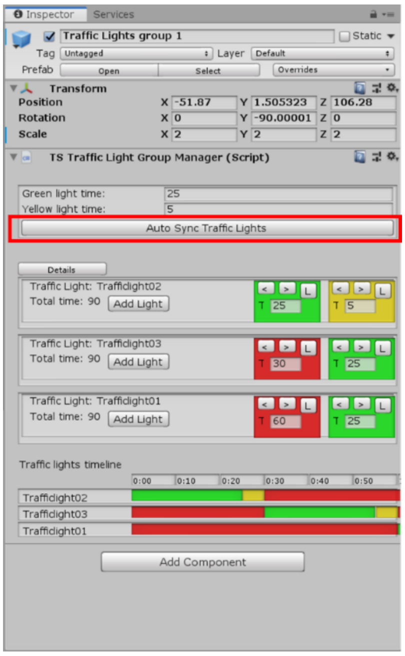


Activate Traffic Light Behaviour

In order to ensure that the traffic lights operate consistently and in sync with each other, the following steps should be carried out:

1) In the Hierarchy pane, click:
City-Maker > Double-Block-06(Clone) > Traffic-Lights > Traffic Lights group 1
2) In the Inspector Window, click on **Auto Sync Traffic Lights.**
3) Repeat steps 1 and 2 for every traffic lights group in Double-Block-06(Clone) and Double-Block-07(Clone).





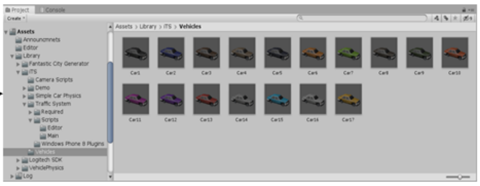
4) Now you can run the game and the changes will be incorporated.

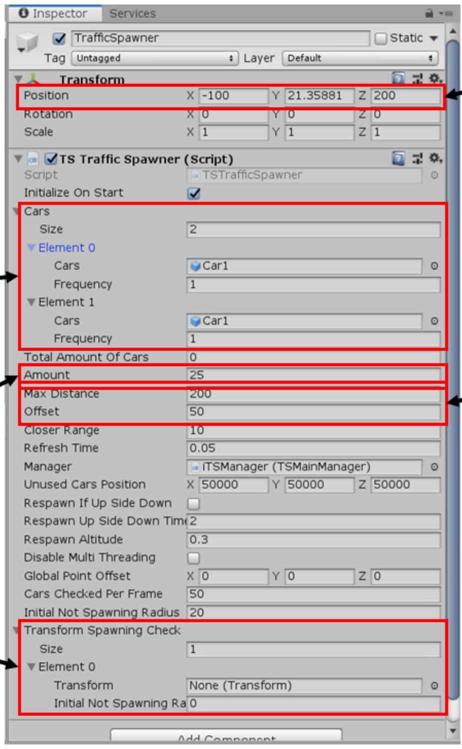
Add Traffic

1) In the Hierarchy Pane, click:
System Managers > Traffic Spawner

2) In the Inspector Window, make sure that the checkbox beside "**TrafficSpawner** is checked. If it is not, check it.

3) In the Inspector Window, you can also change the following settings (see image below).





Change x-, y- and z- values in order to position the traffic spawner over the part(s) of the scene you want traffic on

Here you can specify the type and frequency of the cars that comprise the traffic. *Size* refers to the number of types and generates the same number of elements. Under each element, drag a car from 'Vehicles' (see above) to the Cars field.

Amount refers to the total number of cars on the road. Increasing this number will increase overall traffic density within the traffic spawner area specified.

Max Distance determines the maximum area that the traffic will cover. Offset determines the inner green radius. Cars will only spawn in between the larger and smaller green radii.

This can be used to ensure that cars do not spawn on and directly around your car. Increase Size to '1', drag the Sedan game object from the Hierarchy pane into the Transform field and set the *initial not spawning radius* to a desired number.

4) Now you can run the game and the traffic will be added.

Starting an experimental run

These steps should be followed in the correct order as they are explained when starting an experimental run.

Calibrate: Logitech G920 Controller

These steps should be followed only at the start of your first experimental run of the day, or when you notice that the steering wheel is not in line with the animation.

- Find the Search bar located in the bottom-left corner of your screen beside the Windows Logo.
- In the Search bar, type ‘**Set up USB game controllers**’ and select this option.
- Click:

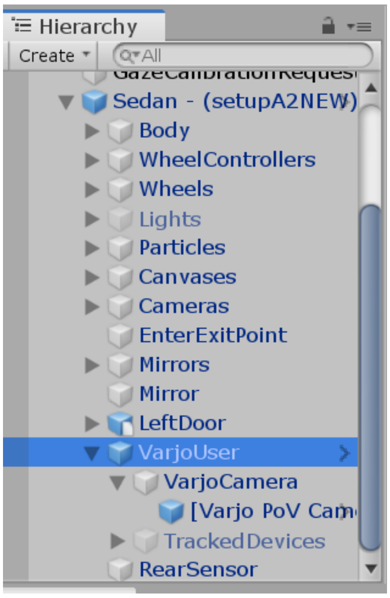
  ‘Properties’ -> ‘Settings’ -> ‘Reset to default’ -> ‘Calibrate’
- Follow the instructions presented to you.
- When finished, click ‘**Apply**’.

Calibrate: Varjo Camera Location (special case only)

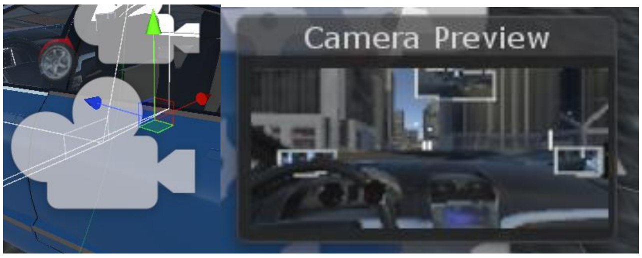
Every time (a) the simulator or (b) the VR cameras physically move in the room, the view that is seen through the VR headset when Unity is run might change.

Check to see if the view is still in place by wearing the VR headset and running the scene.

If it is not in place:



- Determine in which way the camera needs to be moved in order for it to be correctly positioned.
- End Scene.
- Find and click on the GameObject ‘**VarjoUser**’ in the Hierarchy Tab. Place your cursor on the Scene view and press F to find the object on the Scene.
- Click on the  button (two arrows intersecting each other) along the top or press ‘W’ – three-colored arrows will appear around the object you have selected.
- Select the appropriate arrow (based on the axis along which you want to move the object) and change the position of the camera.
- Run Scene and check to see if the position is now correct. Repeat until satisfied.

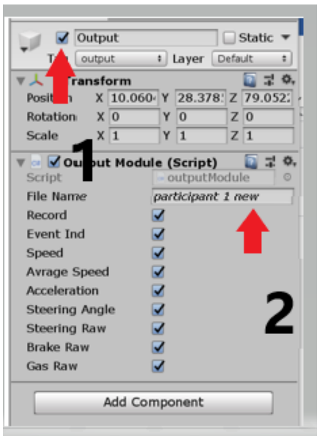


Output Log

From Unity, you get the data from the behaviour of the car during the experiment. You find the output log in the

'Hierarchy' tab -> System Managers -> Output.

Within the '**Inspector**' window you must rename your output in 'File Name' and click all the logs you want to have output from, for example, '**Speed'**, '**Steering Angle**' and '**Gas Raw'**.
Change the file name to something appropriate EVERY time you run a new session!
The next time you run the game, the data will be recorded.



Set-up: iMotions (optional: in case of eye-tracking)

In case you use eye-tracking, follow these steps EVERY time you start running a new session!

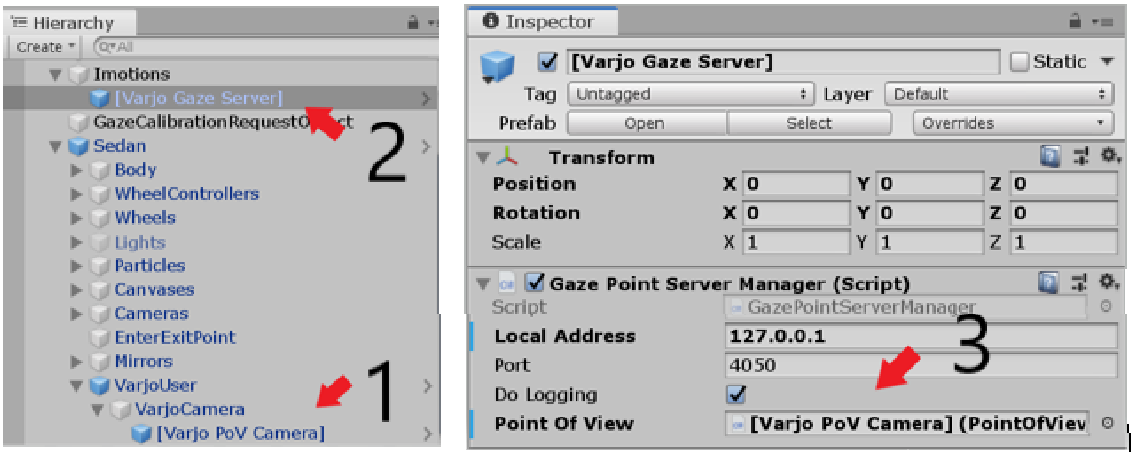
For every time you add another prefab car in your scene, you need to do these steps:

- Navigate to:

  Sedan -> VarjoUser -> VarjoCamera -> [Varjo PoV Camera]

- **Imotions -> [Varjo Gaze Server] -> 'Inspector' window -> Point of View**

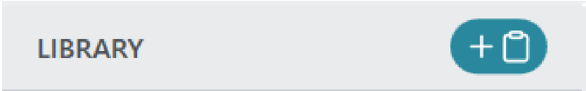
- Drag the [Varjo PoV Camera} into the '**Point of View'** in the **Inspector** tab



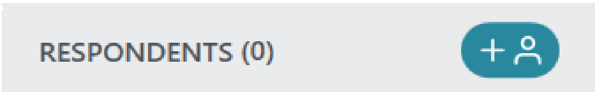
At the beginning of the study:

- Start iMotions software
- Create a study: In the left side of the screen

  Click on the **blue '+' -> Name your study -> Click next -> Click add**



- Create a respondent: In the right side of the screen click on the **blue '+'** -> Name your respondent and add demographic information



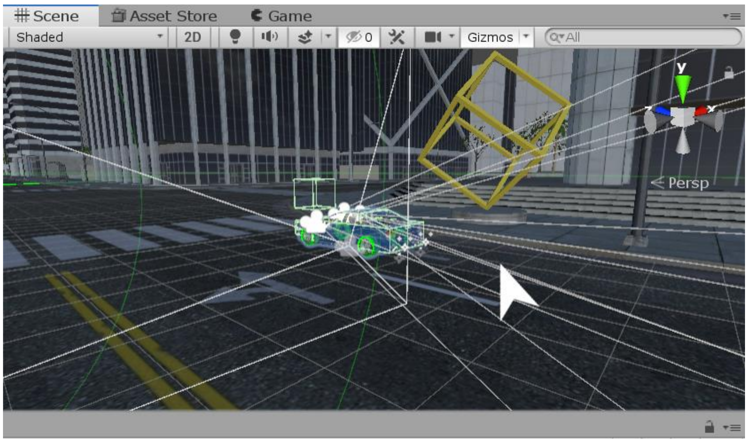
- Create a stimulus: **Click on the blue '+'** in the right upper corner. In case of working with the driving simulator, click on Screen Recording -> Click Add

For each participant:

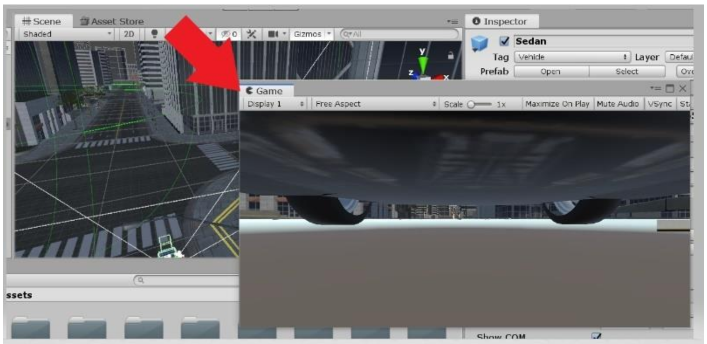
- Run scene on Unity
- On the iMotions application, click on:

  'preferences' -> 'reconnect sensors'.
- Calibrate the Varjo: If using eye-tracking, the Varjo headset needs to be calibrated every time the headset is removed. There are two ways to calibrate the eye-tracker:

- Method 1: Through VarjoBase:
  - Click on the **Unity scene**
  - Press the space bar. The calibration process will be activated by itself and the user can calibrate the eye-tracker by following the instructions on the VR. The left button (Application button) on the headset is 'Okay'.



- Method 2: Through the headset:
  - Place the headset on your head.
  - Press the right button (System button) on the headset.
  - A navigation menu appears on the VR, click on **Calibration -> Eye tracking -> Start**
  - The calibration process starts.



- Start recording in iMotions
- Drag the Game Scene of Unity to the screen that will be recorded by the iMotions (Preferably the second screen).
- Go back to Unity, and leave Unity opened during the entire recording, so the participant can keep the car in control.
- When the session is finished, stop the recording and the recording will be saved in your study.

Run: Unity Scene

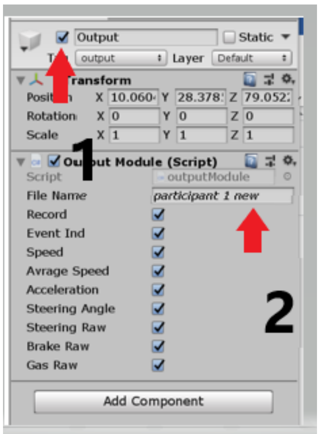
- Press the Play button and click on the Game tab in order to make sure the participant can control the car.

Output unity

From Unity, you get the data from the behaviour of the car during the experiment. You find the output log in the

‘Hierarchy’ tab -> System Managers -> Output.

Within the ‘**Inspector**’ window you must rename your output in ‘File Name’ and click all the logs you want to have output from, for example, ‘**Speed**’, ‘**Steering Angle**’ and ‘**Gas Raw**’.
Change the file name to something appropriate EVERY time you run a new session!
The next time you run the game, the data will be recorded.



File

After running the game, the data file can be found in the project folder under ***‘Assets' -> ‘Log'***

Variables

The following data can be gathered.

- Timestamps in seconds
- Events (can be recorded manually, by pressing the z key, or automatically)
- Speed in Km/h
- Average speed in Km/h (Avg speed)
- Acceleration in m/s
- Steering Angle
- Raw steering values
- Raw gas values
- Raw brake values

After running the game, the data is saved in an excel file.

	A	B	C	D	E	F	G	H	I
1	TimeStam	Event	Speed	AvgSpeed	Acceleration	Steering Angle	SteeringRaw	GasRaw	BrakeRaw
2	0		0	0	0	0	0	0	0

Each value is linked to a timestamp which is displayed in the first column in the excel file. The timestamp in column 1 is connected to the data in the other columns. For example, the speed at 63,6 seconds in the game was approximately 45,70 km/h (856, column C).

A858 | 63,6

	A	B	C	D	E	F	G	H	I	J
856	63,51		45,70372	0,520665	2,520879	0,259191	351	18061	32767	
857	63,555		46,12461	0,530863	2,635702	0,160624	218	18061	32767	
858	63,6		46,56333	0,541107	2,74334	-0,02646	-36	18061	32767	
859	63,66		47,17379	0,55146	2,874247	-0,24913	-340	18061	32767	
860	63,705		47,64914	0,561964	2,962685	-0,4093	-566	18061	32767	

Events

Manually recorded

By pressing the M key, it will stop printing the event in the excel file. This way you can see at which time stamps the event was happening. Here the event was triggered between the Z key was pressed at 77,28 second and finished when the M key was pressed at 77,895 seconds (row 1137 till 1156)

A1137 | 77,28

	A	B	C	D	E	F	G	H	I
1137	77,28	initiate lar	49,30546	3,540991	-0,9411494	2,942832	4062	32767	32767
1138	77,325	initiate lar	49,1592	3,548383	-0,9353002	2,900906	4001	32767	32767
1139	77,37	initiate lar	49,01362	3,555619	-0,9302775	2,853822	3933	32767	32767
1140	77,415	initiate lar	48,86868	3,562611	-0,9250641	2,7747	3821	32767	32767
1141	77,46	initiate lar	48,72451	3,569873	-0,9175619	2,744859	3777	32767	32767
1142	77,505	initiate lar	48,58096	3,577201	-0,9150823	2,741869	3770	32767	32767
1143	77,55	initiate lar	48,43773	3,584509	-0,9138743	2,7345	3757	32767	32767
1144	77,595	initiate lar	48,29481	3,591656	-0,9112676	2,700889	3708	32767	32767
1145	77,64	initiate lar	48,15241	3,598616	-0,9062449	2,640246	3622	32767	32767
1146	77,685	initiate lar	48,01073	3,605723	-0,8998235	2,5474	3492	32767	32767
1147	77,715	initiate lar	47,9181	3,613078	-0,8589427	2,442825	3347	32767	32767
1148	77,75999	initiate lar	47,77636	3,620196	-0,8818945	2,32926	3189	32767	32767
1149	77,805	initiate lar	47,63747	3,627049	-0,876236	2,29516	3140	32767	32767
1150	77,85	initiate lar	47,49902	3,633931	-0,8752823	2,296113	3139	32767	32767
1151	77,895	initiate lar	47,36074	3,640925	-0,875028	2,297795	3139	32767	32767
1152	77,93999		47,22268	3,647734	-0,8733749	2,299475	3139	32767	32767
1153	77,985		47,08492	3,654522	-0,8714041	2,301151	3139	32767	32767
1154	78,03		46,94748	3,661192	-0,8691788	2,297687	3127	32767	32767
1155	78,075		46,81039	3,667828	-0,8667628	2,281732	3108	32767	32767
1156	78,13499		46,62891	3,673779	-0,8302689	2,24644	3051	32767	32767

Make sure that you first press the Z key (row 4730), then press the B key (several times) (row 4742) and lastly press the M key to finish the event (row 4746). The distance from the back of the car and the vehicle behind here was 83,30m when the B key was pressed.

A1137 | fx | 77,28

	A	B	C	D	E	F	G	H	I
4730	243,36	initiate lar	49,09599	11,12641	-0,8503596	1,235895	1704	32767	32767
4731	243,42	initiate lar	48,91204	11,12691	-0,8464813	1,104977	1522	32767	32767
4732	243,48	initiate lar	48,72895	11,12737	-0,8437475	1,096609	1510	32767	32767
4733	243,525	initiate lar	48,5918	11,12809	-0,8516312	1,131592	1560	32767	32767
4734	243,57	initiate lar	48,45488	11,12906	-0,8511862	1,139695	1566	32767	32767
4735	243,615	initiate lar	48,31818	11,13012	-0,8496603	1,14271	1569	32767	32767
4736	243,66	initiate lar	48,18183	11,13108	-0,8473715	1,148641	1581	32767	32767
4737	243,705	initiate lar	48,04581	11,13216	-0,8454641	1,188857	1630	32767	32767
4738	243,735	initiate lar	47,95564	11,1334	-0,8347194	1,200372	1645	32767	32767
4739	243,78	initiate lar	47,81949	11,13433	-0,8430481	1,201971	1646	32767	32767
4740	243,825	initiate lar	47,68417	11,13531	-0,8415222	1,192603	1631	32767	32767
4741	243,87	initiate lar	47,54918	11,13611	-0,8389791	1,175175	1606	32767	32767
4742	243,915	83,30491	47,41457	11,1367	-0,836436	1,127714	1541	32767	32767
4743	243,975	83,30491	47,23587	11,13718	-0,8237839	1,011581	1374	32767	32767
4744	244,02	83,30491	47,10224	11,1379	-0,82709	0,8920814	1217	32767	32767
4745	244,065	83,30491	46,96923	11,1385	-0,8229574	0,7988166	1089	32767	32767
4746	244,125	83,30491	46,79262	11,13901	-0,8159002	0,7173297	977	32767	32767

Automatically recorded

Events can also be recorded automatically with the use of the *Checkpoint Manager* (see ''*preparation of your experiment''*) in Unity. When the Checkpoint System is used, the Checkpoints will be printed in the data file when these are hit in the scene.

Raw steering values

In column SteeringRaw, column G, the raw values of the steering wheel are printed. The range of these values is from –32767 to 32767. The maximum negative value is recorded when the physical steering wheel is totally turned to the left. The maximum positive value is recorded when the physical steering wheel is totally turned to the right.

To get the valuable steering angle out of these numbers, it is possible to rescale them in excel. The physical steering wheel has the possibility to turn 900° in total, so 450° to the left and 450° to the right. Therefore, the scale should be between –450 and 450. Steps in excel:

- Make a new column next to the SteeringRaw column and give this a preferred name.
- Type an '=' in the first cell of this new column
- Click the cell of the old column next to this

G	H
SteeringRa	SteeringSc
-12	=G2

- Type a new formula in the bar above the table: **=(*value*/32767)*450**

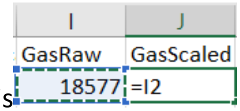
`=(G2/32767)*450`

- Push 'Enter'
- Click the right bottom corner of the cell with the new value and drag this down for the entire column. The other values will be calculated automatically.

Raw gas and brake values

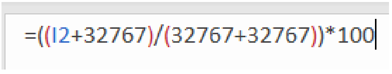
In the column GasRaw and BrakeRaw, the raw values of the physical gas and brake throttles are printed. Just as the SteeringRaw, the range of these values is from –32767 to 32767. **Pay attention: -32767 means that the throttle is maximally pushed and 32767 means that the throttle is not pushed at all.**

Also, these values can be scaled to a valuable scale. In this case, the range from 0 to 100 would be valuable, in which 0 means that the throttle is maximally pushed and 100 means that the throttle is not pushed at all. Similar steps should be taken as in scaling the steering angle:

- Make a new column next to the GasRaw or BrakeRaw column and give this a preferred name.
- Type an '=' in the first cell of this new column
- Click the cell of the old column next to this 

I	J
GasRaw	GasScaled
18577	=I2

- Type a new formula in the bar above the table: **=((*value*+32767)/(32767+32767))*100**

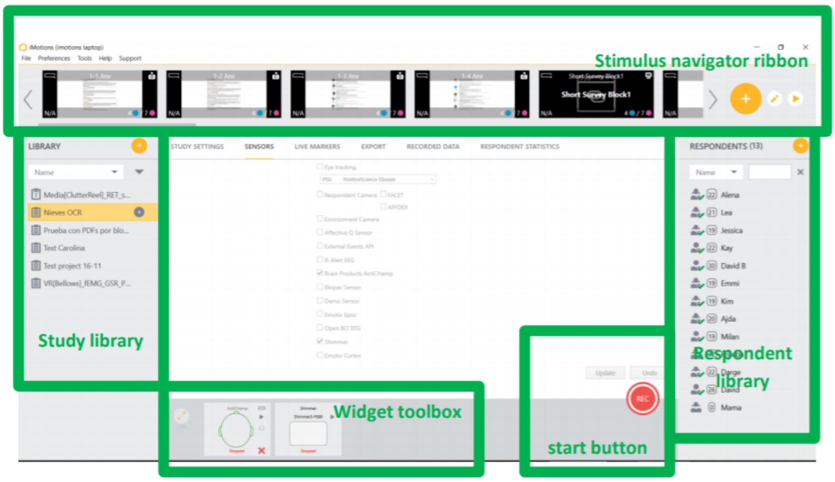


- Push 'Enter'
- Click the right bottom corner of the cell with the new value and drag this down for the entire column. The other values will be calculated automatically.

Exporting and analysing Imotions data

Exporting data

Right-click on the study you want to use in the study library -> click on **export sensor data** -> select the element you want to export.



Analysing

For information on analysing data following link can be consulted:

help.imotions.com/hc/en-us/articles/213046585-Gaze-Mapping-

For the creation of AOI the following link can be consulted

:https://help.imotions.com/hc/en-us/articles/205144072-Automated-Areas-of-Interest-AOIs-

Appendix F

NASA TLX

Figure 8.6

NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.

Name	Task	Date

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low — Very High

Appendix G

User Experience Questionnaire

Please make your evaluation now.

For the assessment of the product, please fill out the following questionnaire. The questionnaire consists of pairs of contrasting attributes that may apply to the product. The circles between the attributes represent gradations between the opposites. You can express your agreement with the attributes by ticking the circle that most closely reflects your impression.

Example:

attractive	○ ⊗ ○ ○ ○ ○ ○	unattractive

This response would mean that you rate the application as more attractive than unattractive.

Please decide spontaneously. Don’t think too long about your decision to make sure that you convey your original impression.

Sometimes you may not be completely sure about your agreement with a particular attribute or you may find that the attribute does not apply completely to the particular product. Nevertheless, please tick a circle in every line.

It is your personal opinion that counts. Please remember: there is no wrong or right answer!

Please assess the product now by ticking one circle per line.

not understandable	○ ○ ○ ○ ○ ○ ○	understandable
easy to learn	○ ○ ○ ○ ○ ○ ○	difficult to learn
obstructive	○ ○ ○ ○ ○ ○ ○	supportive
complicated	○ ○ ○ ○ ○ ○ ○	easy
secure	○ ○ ○ ○ ○ ○ ○	not secure
meets expectations	○ ○ ○ ○ ○ ○ ○	does not meet expectations
clear	○ ○ ○ ○ ○ ○ ○	confusing
Predictable	○ ○ ○ ○ ○ ○ ○	Unpredictable

Appendix H

Confidence Questionnaire

Confidence questionnaire

Participant no.________

Setup ______

1. On a scale from 1 till 10, how confident did you feel while making the lane changes?

____

2. On a scale from 1 till 10, how confident did you feel while parking?

____

Appendix I

Simulator Sickness Questionnaire

SIMULATOR SICKNESS QUESTIONNAIRE

Kennedy, Lane, Berbaum, & Lilienthal (1993)***

Instructions : Circle how much each symptom below is affecting you right now.

1. General discomfort	None	Slight	Moderate	Severe
2. Fatigue	None	Slight	Moderate	Severe
3. Headache	None	Slight	Moderate	Severe
4. Eye strain	None	Slight	Moderate	Severe
5. Difficulty focusing	None	Slight	Moderate	Severe
6. Salivation increasing	None	Slight	Moderate	Severe
7. Sweating	None	Slight	Moderate	Severe
8. Nausea	None	Slight	Moderate	Severe
9. Difficulty concentrating	None	Slight	Moderate	Severe
10. « Fullness of the Head »	None	Slight	Moderate	Severe
11. Blurred vision	None	Slight	Moderate	Severe
12. Dizziness with eyes open	None	Slight	Moderate	Severe
13. Dizziness with eyes closed	None	Slight	Moderate	Severe
14. *Vertigo	None	Slight	Moderate	Severe
15. **Stomach awareness	None	Slight	Moderate	Severe
16. Burping	None	Slight	Moderate	Severe

* Vertigo is experienced as loss of orientation with respect to vertical upright.

** Stomach awareness is usually used to indicate a feeling of discomfort which is just short of nausea.

Last version : March 2013

***Original version : Kennedy, R.S., Lane, N.E., Berbaum, K.S., & Lilienthal, M.G. (1993). Simulator Sickness Questionnaire: An enhanced method for quantifying simulator sickness. *International Journal of Aviation Psychology, 3*(3), 203-220.

Appendix J

Demographic Questionnaire

Questionnaire participant no. ______

1. What is your age ?

______ Year

2.What is your gender?

Male / Female

3. What is your country of birth?

____

4. Do you possess a driving license?

YES / NO

5. How many years of driving experience do you have?

____

Appendix K

Informed Consent

Dear Participant,

You are invited to participate in the study called; "Future of car displays". This research is being done as part of my Master's thesis for the University of Twente. The aim of the research is to gain more insight into the use of digital displays as a replacement for traditional mirrors. This research has been approved by the ethics committee of the faculty; "Behavioural Management and social sciences".

The study will start with questions about personal data (place of residence and age etc.). Then, I will ask you to take a seat in the driving simulator in which you will perform several tasks. These tasks will include that you are driving around on a circuit and at a certain moment that a signal is given, you will change lanes as quickly as possible. At the end you are asked to do a parking task. We will repeat this several times with different setups in which the position of the mirrors differ. In between, there will be a 10 minute break. The experiment will take approximately 60 minutes and will be recorded.

Your participation in this study is entirely voluntary and you can stop at any time. You are free to refuse to answer questions.

Remember that we are investigating how the system works. Poor performance will only indicate shortcomings in the system's usability.

As virtual reality system can in some cases give you dizziness or other issues known as cyber-sickness which may include nausea and discomfort, if you suffered of previously induced motion sickness from virtual reality or simulators, or if you are (possibly) pregnant or suffering from any conditions (e.g., epilepsy, etc.) that may expose you to an adverse reaction to an immersive experience in a virtual reality setting we strongly advise against participating in this study.

We believe that there are no further risks associated with this investigation. However, we will do our best to ensure that your answers remain confidential and will not be used for other purposes.

Informed consent.

By signing this document you indicate that you are at least 16 years old; that you are well informed about the research and the way in which the research data is collected, used and treated and what possible risks are involved by participating in this research.

If you had any questions, indicate when signing that you were able to ask these questions and that these questions were answered clearly. You indicate that you voluntarily agree with your participation in this study.

This research project is guided by Dr. Borsci. The purpose of this document is to establish the conditions of my participation in the project.

	Yes	No
1. I never suffered of cyber-sickness or motion sickness due to the use of simulators and virtual reality		
2. I am not aware of any physical or medical condition that prevent me to be part of this experiment.		
3. I have the right to stop even without asking to perform the experiment at any time if I feel that I am dizzy or not comfortable for any reason.		
4. I have received sufficient information about this research project. The purpose of my participation as a participant in this project has been clearly explained to me and I know what this means for me.		
5. My participation as a participant in this project is voluntary. There is no explicit or implicit compulsion to participate in this study.		
6. The experiment will take 60 minutes. I give the researcher permission to make screen and eye movement recordings and take written notes during the research. It is clear to me that if I object to certain issues, I can stop my participation at any time, without giving a reason.		
7. I have received the guarantee from the investigator that the investigator ensures that I am not identifiable in data, reports or articles brought forward by the investigation. My privacy is guaranteed as a participant in this study.		
8. I have been guaranteed that this research project has been reviewed and approved by the ethics committee of the BMS Ethics Committee. For objections with regard to the design and / or implementation of the research, I can contact the Secretary of the Ethics Committee of the faculty of Behavioural, Management and Social Sciences at the University of Twente via ethicscommittee-bms@utwente.nl.		
9. I have read and understood this form. All my questions have been answered to my satisfaction and I voluntarily agree to participate in this study.		

________________________ ________________________ _________

Name of participant Signature Date

________________________ ________________________ _________

Name Researcher Signature Date

Appendix L

Analysis Package

Performance data

Sarah Koning

10-4-2020

```
library(tidyverse)

## -- Attaching packages -------------------------------------------------------------------------
------------------------------------------ tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ----------------------------------------------------------------------------------
------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(gridExtra)

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine

library(rstanarm)

## Loading required package: Rcpp

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo

## rstanarm (Version 2.19.2, packaged: 2019-10-01 20:20:33 UTC)

## - Do not expect the default priors to remain the same in future rstanarm versions.

## Thus, R scripts should specify priors explicitly, even if they are just the defaults.

## - For execution on a local, multicore CPU with excess RAM we recommend calling

## options(mc.cores = parallel::detectCores())

## - bayesplot theme set to bayesplot::theme_default()

##    * Does _not_ affect other ggplot2 plots

##    * See ?bayesplot_theme_set for details on theme setting

library(bayr)
```

```
## Registered S3 methods overwritten by 'bayr':
##   method          from
##   coef.stanreg    rstanarm
##   predict.stanreg rstanarm

##
## Attaching package: 'bayr'

## The following objects are masked from 'package:rstanarm':
##
##     fixef, ranef
```

Data analysis package of performance data

The following code is needed for the data analysis that should be performed on the performance data (data gathered from the unity). Make sure that the initial data set should have rows containing :

- The observation number
- The corresponding participant number (Observation)
- The number indicating the number of lane changes performed per participant for one setup.(Lanechangenumber)
- The corresponding lanechange time for each lane change. (Lanechangetime)
- The corresponding headway distance for each lane change.(Lanechangedistance)
- The mirror setup that was used (Group)
- The Speed at the moment that the car ended the lane change (Speed)
- The number indicating how many lane changes have been performed by the one participant (Participantlanechange)
- The years of driving experience (Driving experience)
- Age of the corresponding participant (age)

The theme_apa can be used to add an apa theme to your graph. Just install the library(jtools) and library(car), and add: geom_line(aes(linetype = Group))+ theme_apa() + theme(text=element_text(family="Times")) + theme(legend.position = c(0.9, 0.8)) + scale_linetype_manual(values=c("solid","dashed", "dotted")) to the code of your graph.

```
windowsFonts(Times=windowsFont("Times new roman"))

theme_apa(
  legend.pos = "right",
  legend.use.title = FALSE,
  legend.font.size = 12,
  x.font.size = 12,
  y.font.size = 12,
  facet.title.size = 12,
  remove.y.gridlines = TRUE,
  remove.x.gridlines = TRUE
)
```

Loading data set

Code for loading the dataset:

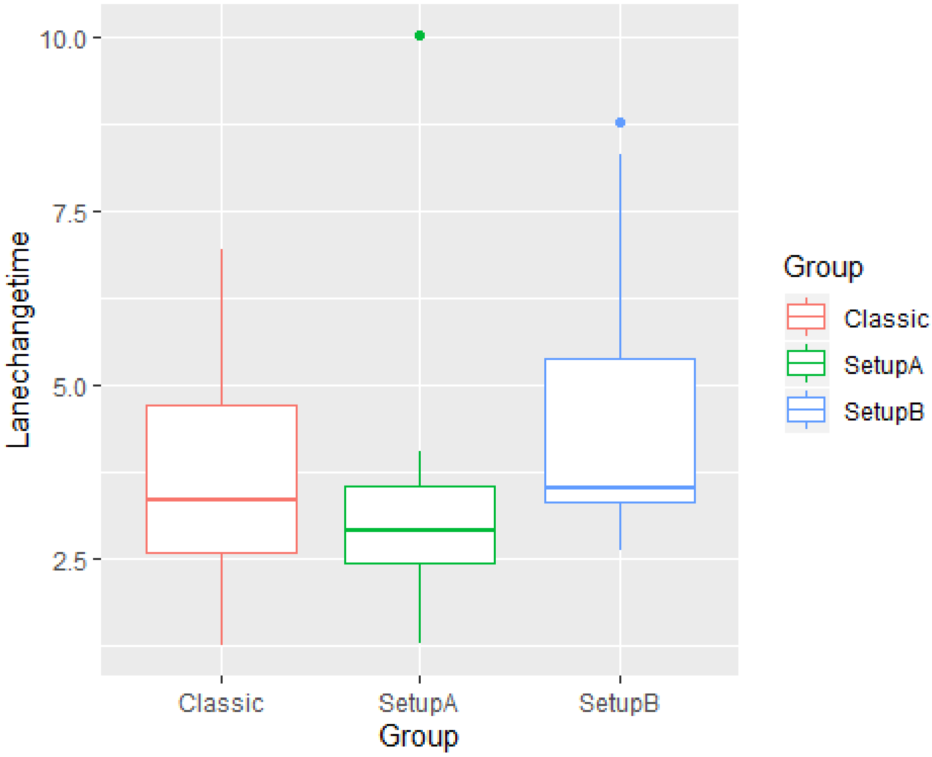
```
DF_1 <-
  read.csv2("performancedata.csv") %>%
  filter(Lanechangenumber < 6) %>%
  mutate(Partipant  = as.factor(Participant),
         Observation = as.factor(Observation),
         Lanechangenumber = as.numeric(Lanechangenumber),
         Participantslanechange = as.numeric(Participantslanechange))

summary(DF_1)

##   Observation  Participant   Lanechangenumber Lanechangetime
##  1      : 1   Min.   :1.00   Min.   :1.00     Min.   : 1.245
##  2      : 1   1st Qu.:1.00   1st Qu.:2.00     1st Qu.: 2.610
##  3      : 1   Median :1.00   Median :3.00     Median : 3.390
##  4      : 1   Mean   :1.44   Mean   :2.64     Mean   : 3.972
##  5      : 1   3rd Qu.:2.00   3rd Qu.:4.00     3rd Qu.: 4.410
##  6      : 1   Max.   :2.00   Max.   :5.00     Max.   :10.020
##  (Other):19
##  Lanechangedistance     Group       Speed       Participantslanechange
##  Min.   :  8.764    Classic:9   Min.   :15.41   Min.   : 1.00
##  1st Qu.: 19.921    SetupA :8   1st Qu.:46.66   1st Qu.: 4.00
##  Median : 29.571    SetupB :8   Median :56.73   Median : 7.00
##  Mean   : 36.924                Mean   :54.01   Mean   : 7.64
##  3rd Qu.: 38.099                3rd Qu.:64.77   3rd Qu.:10.00
##  Max.   :126.404                Max.   :73.96   Max.   :19.00
##
##  Drivingexperience      Age         Partipant
##  Min.   : 1.00     Min.   :19.00   1:14
##  1st Qu.: 1.00     1st Qu.:19.00   2:11
##  Median :22.00     Median :41.00
##  Mean   :12.76     Mean   :31.32
##  3rd Qu.:22.00     3rd Qu.:41.00
##  Max.   :22.00     Max.   :41.00
##
```

Data summary

A boxplot can be created to get the corresponding lane change time and lanechange distance for each setup.

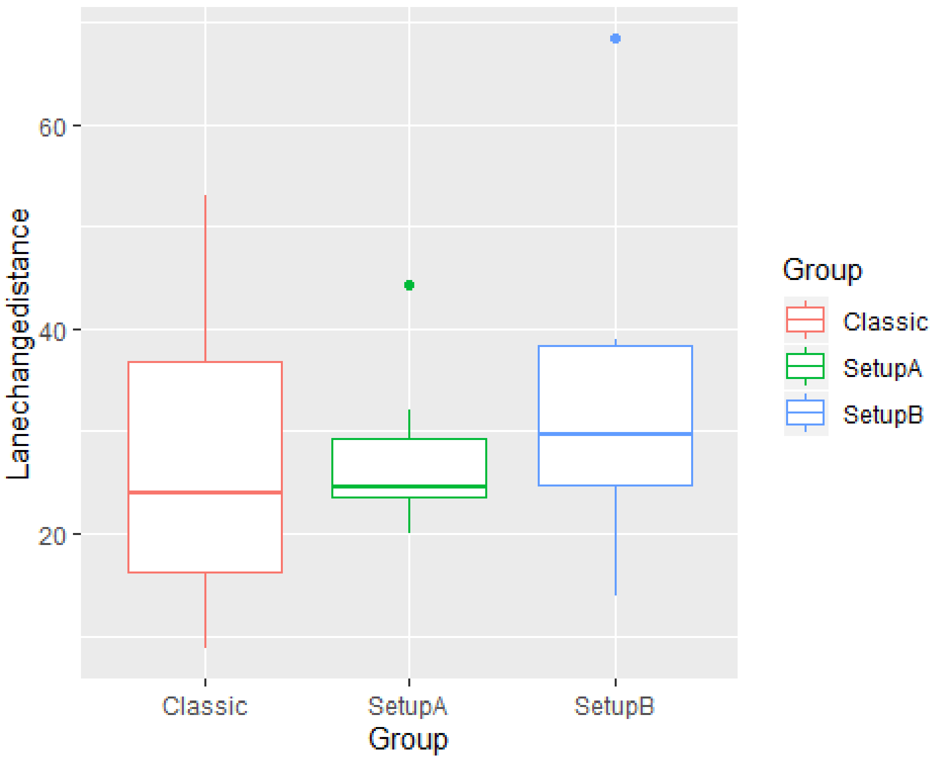
Time

```
DF_1 %>%
  ggplot(aes(x = Group,color = Group, y = Lanechangetime)) +
  geom_boxplot()
```



Distance

```
DF_1 %>%
  filter(Lanechangedistance < 100) %>%
  ggplot(aes(x = Group, colour = Group,  y = Lanechangedistance)) +
  geom_boxplot()
```

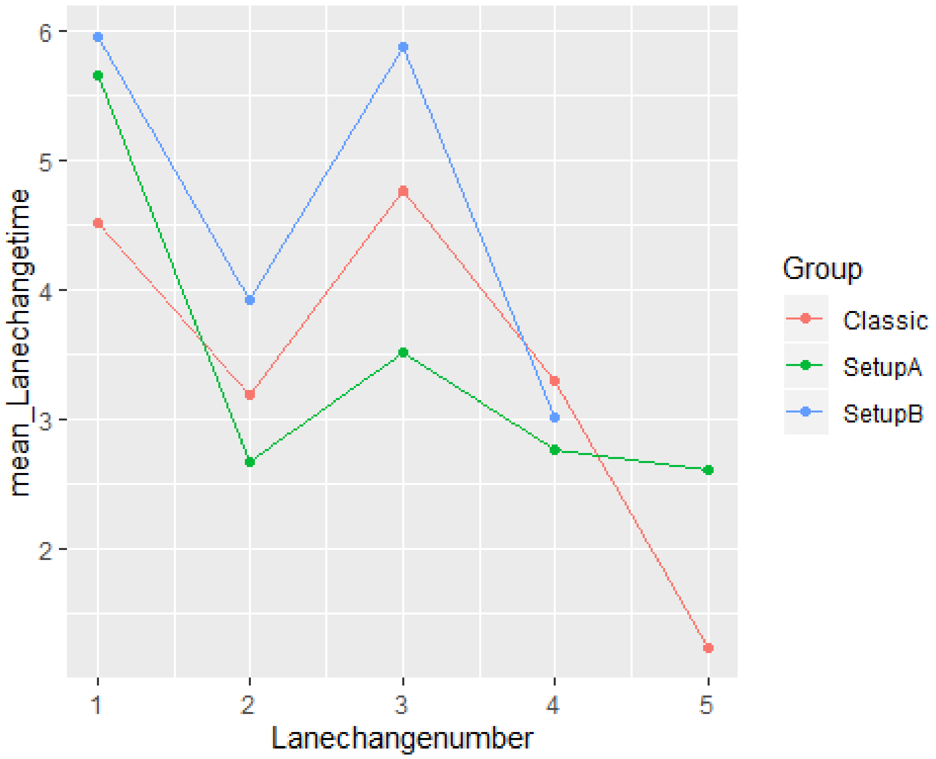


Graphs

Underneath, the code is presented that will show the mean lanechange time and lane change distance for each setup per Lane change that was performed in a graph.

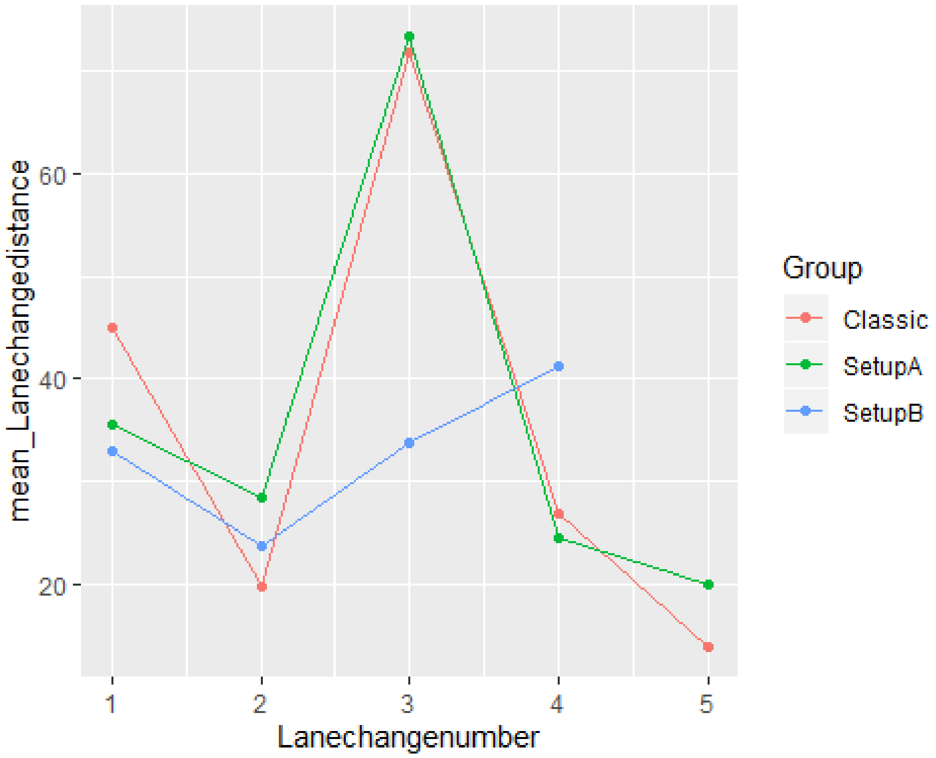
Mean time

```
DF_1 %>%
  filter(Lanechangenumber < 6) %>%
    group_by(Group, Lanechangenumber) %>%
  summarize(mean_Lanechangetime = mean(Lanechangetime, na.rm = T),
            sd_Lanechangetime = sd(Lanechangetime, na.rm = T)) %>%
  ggplot(aes(color = Group,
             x = Lanechangenumber,
             y = mean_Lanechangetime)) +
  geom_point() +
  geom_line(aes(group = Group))
```



Mean distance

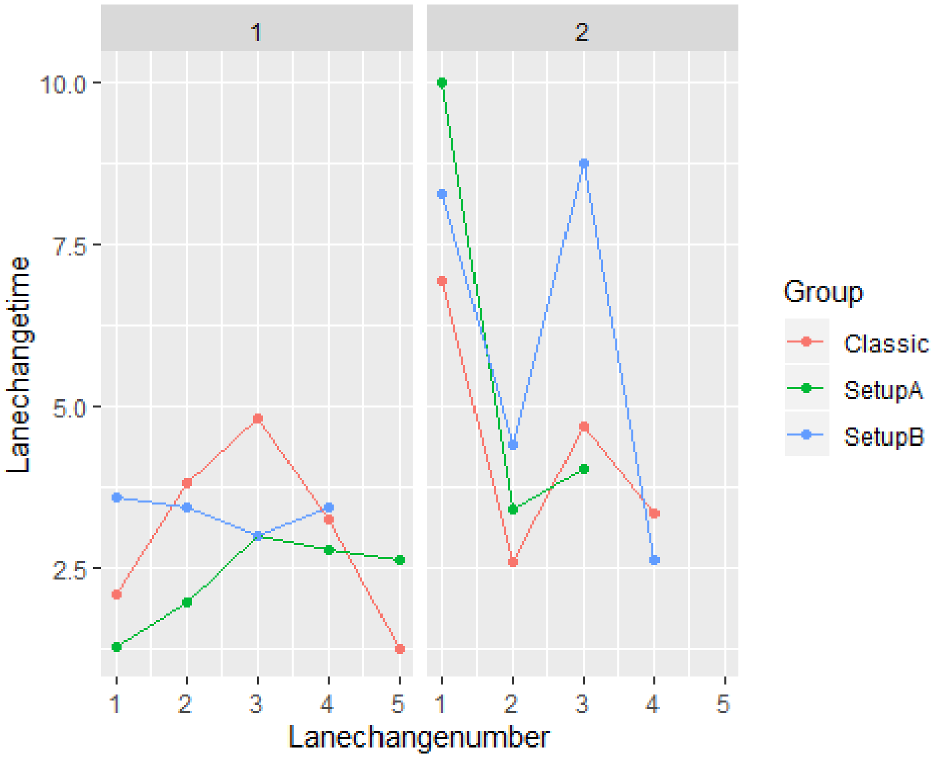
```
DF_1 %>%
  filter(Lanechangenumber < 6) %>%
    group_by(Group, Lanechangenumber) %>%
  summarize(mean_Lanechangedistance = mean(Lanechangedistance, na.rm = T),
            sd_Lanechangedistance = sd(Lanechangedistance, na.rm = T)) %>%
  ggplot(aes(color = Group,
             x = Lanechangenumber,
             y = mean_Lanechangedistance)) +
  geom_point() +
  geom_line(aes(group = Group))
```



Time per participant

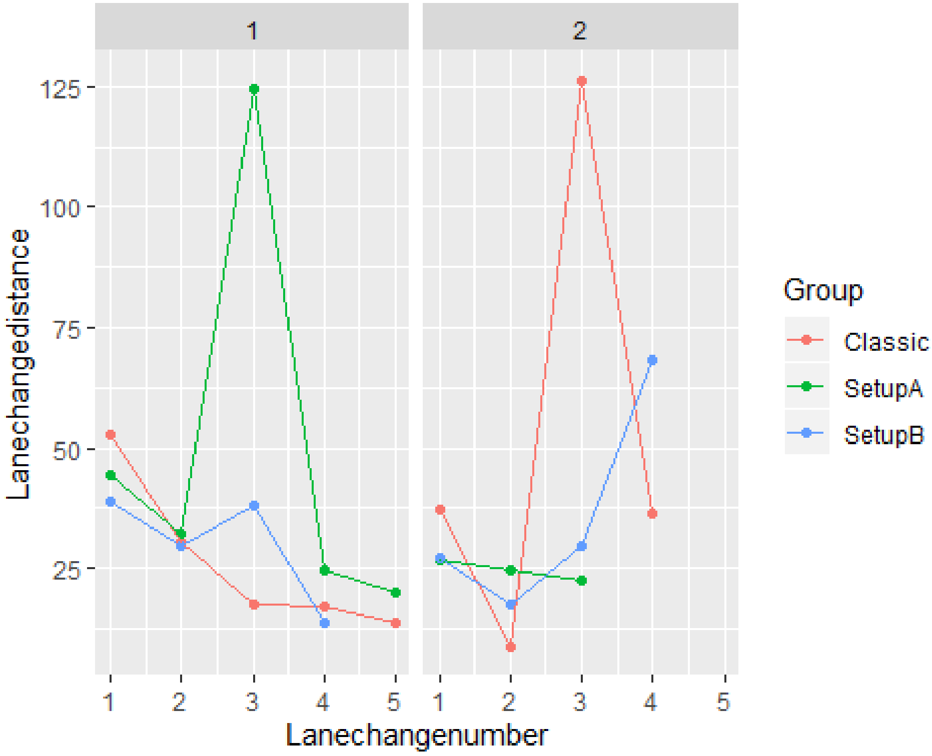
The following code will give you the graphs for each participant

```
DF_1 %>%
  filter(Lanechangenumber < 6) %>%
  group_by(Group, Lanechangenumber) %>%
  ggplot(aes(color = Group,
             x = Lanechangenumber,
             y = Lanechangetime)) +
  facet_wrap(~Participant) +
  geom_point() +
  geom_line(aes(group = Group))
```



Distance per participant

```
DF_1 %>%
  filter(Lanechangenumber < 6) %>%
  group_by(Group, Lanechangenumber) %>%
  ggplot(aes(color = Group,
             x = Lanechangenumber,
             y = Lanechangedistance)) +
  facet_wrap(~Participant) +
  geom_point() +
  geom_line(aes(group = Group))
```



Modeling

The code for the regression analysis can be found underneath to see if performance is expected to increase. The intercept of model 2 represents the time that people needed to perform the lane change tasks with the classic setup when they made the first lane change. GroupSetup A and GroupSetup B need to be added to this number to get the expected lane change time for the first session for Setup A and B . Subsequently, per session the time will decrease or increase for the classic setup with the number presented by the lanechangenumber. The same counts for Setup A and B represented by GroupSetupA:Lanechangenumber and GroupSetupB:Lanechangenumber.

Time

```
MODEL_1 <- stan_glm(Lanechangetime ~ 1 + Group + Lanechangenumber, data = DF_1)
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.091 seconds (Warm-up)
## Chain 1:                0.095 seconds (Sampling)
## Chain 1:                0.186 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.083 seconds (Warm-up)
## Chain 2:                0.083 seconds (Sampling)
## Chain 2:                0.166 seconds (Total)
```

```
## Chain 2:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.117 seconds (Warm-up)
## Chain 3:                0.073 seconds (Sampling)
## Chain 3:                0.19 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.082 seconds (Warm-up)
## Chain 4:                0.071 seconds (Sampling)
## Chain 4:                0.153 seconds (Total)
## Chain 4:
```

```
summary(MODEL_1)
```

```
##
## Model Info:
##  function:     stan_glm
##  family:       gaussian [identity]
##  formula:      Lanechangetime ~ 1 + Group + Lanechangenumber
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 25
##  predictors:   4
##
## Estimates:
```

```
##                    mean   sd   10%   50%   90%
## (Intercept)       5.4    1.3  3.7   5.4   7.0
## GroupSetupA      -0.1    1.1 -1.5  -0.1   1.3
## GroupSetupB       0.8    1.1 -0.5   0.9   2.2
## Lanechangenumber -0.6    0.4 -1.1  -0.6  -0.1
## sigma             2.3    0.4  1.9   2.2   2.7
##
## Fit Diagnostics:
##            mean   sd   10%   50%   90%
## mean_PPD 4.0    0.7  3.2   4.0   4.8
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (f
or details see help('summary.stanreg')).
##
## MCMC diagnostics
##                  mcse Rhat n_eff
## (Intercept)      0.0  1.0  2866
## GroupSetupA      0.0  1.0  2814
## GroupSetupB      0.0  1.0  2720
## Lanechangenumber 0.0  1.0  3315
## sigma            0.0  1.0  2897
## mean_PPD         0.0  1.0  3713
## log-posterior    0.0  1.0  1630
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective s
ample size, and Rhat is the potential scale reduction factor on split chains (at convergence Rhat=
1).
```

```
fixef(MODEL_1)
```

fixef	center	lower	upper
Intercept	5.3565110	2.826167	7.8932750
GroupSetupA	-0.1151628	-2.302571	2.0306623
GroupSetupB	0.8554638	-1.429531	3.0639058
Lanechangenumber	-0.6012728	-1.360115	0.1144867

```
MODEL_2 <- stan_glm(Lanechangetime ~ Group + Lanechangenumber + Group:Lanechangenumber, data = DF_
1)
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.229 seconds (Warm-up)
## Chain 1:                0.185 seconds (Sampling)
## Chain 1:                0.414 seconds (Total)
```

```
## Chain 1: 
## 
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2: 
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2: 
## Chain 2: 
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2: 
## Chain 2:  Elapsed Time: 0.202 seconds (Warm-up)
## Chain 2:                0.208 seconds (Sampling)
## Chain 2:                0.41 seconds (Total)
## Chain 2: 
## 
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3: 
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3: 
## Chain 3: 
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3: 
## Chain 3:  Elapsed Time: 0.201 seconds (Warm-up)
## Chain 3:                0.246 seconds (Sampling)
## Chain 3:                0.447 seconds (Total)
## Chain 3: 
## 
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4: 
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4: 
## Chain 4: 
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
```

```
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.185 seconds (Warm-up)
## Chain 4:                0.196 seconds (Sampling)
## Chain 4:                0.381 seconds (Total)
## Chain 4:
```

```
summary(MODEL_1)
```

```
##
## Model Info:
##  function:     stan_glm
##  family:       gaussian [identity]
##  formula:      Lanechangetime ~ 1 + Group + Lanechangenumber
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 25
##  predictors:   4
##
## Estimates:
##                    mean   sd   10%   50%   90%
## (Intercept)       5.4    1.3  3.7   5.4   7.0
## GroupSetupA      -0.1    1.1 -1.5  -0.1   1.3
## GroupSetupB       0.8    1.1 -0.5   0.9   2.2
## Lanechangenumber -0.6    0.4 -1.1  -0.6  -0.1
## sigma             2.3    0.4  1.9   2.2   2.7
##
## Fit Diagnostics:
##            mean   sd   10%   50%   90%
## mean_PPD 4.0    0.7  3.2   4.0   4.8
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (f
or details see help('summary.stanreg')).
##
## MCMC diagnostics
##                  mcse Rhat n_eff
## (Intercept)      0.0  1.0  2866
## GroupSetupA      0.0  1.0  2814
## GroupSetupB      0.0  1.0  2720
## Lanechangenumber 0.0  1.0  3315
## sigma            0.0  1.0  2897
## mean_PPD         0.0  1.0  3713
## log-posterior    0.0  1.0  1630
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective s
ample size, and Rhat is the potential scale reduction factor on split chains (at convergence Rhat=
1).
```

```
summary(MODEL_2)
```

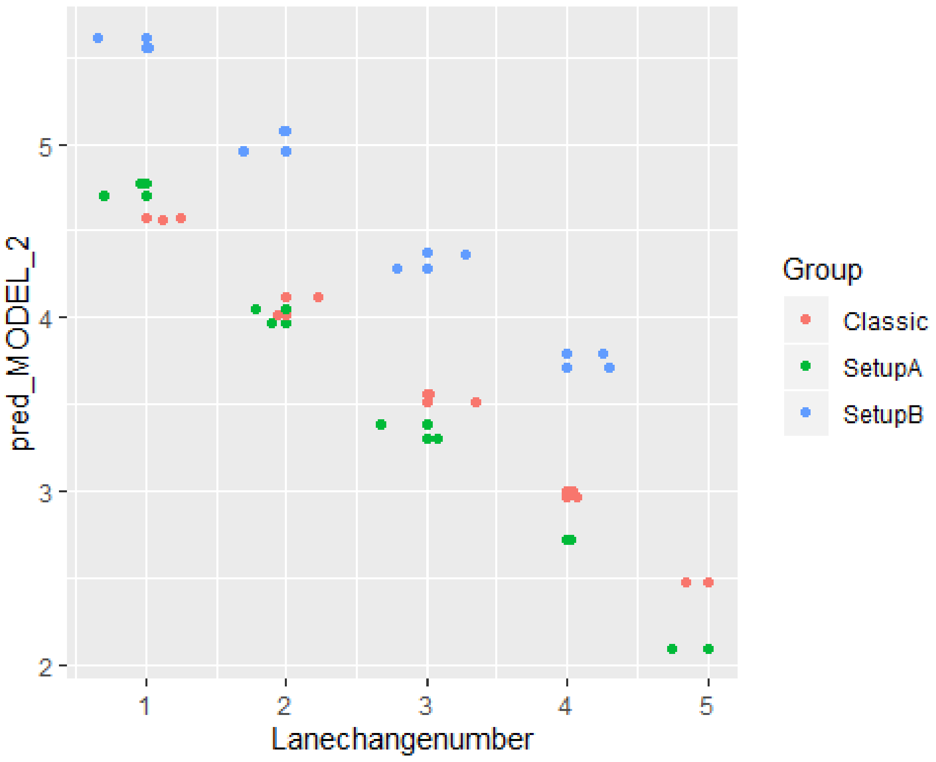
```
##
## Model Info:
##  function:     stan_glm
##  family:       gaussian [identity]
##  formula:      Lanechangetime ~ Group + Lanechangenumber + Group:Lanechangenumber
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 25
##  predictors:   6
##
```

```
## Estimates:
##                                mean   sd   10%   50%   90%
## (Intercept)                   5.1    1.7  3.0   5.1   7.3
## GroupSetupA                   0.2    2.3 -2.7   0.3   3.0
## GroupSetupB                   1.1    2.3 -1.8   1.1   4.1
## Lanechangenumber             -0.5    0.5 -1.2  -0.5   0.2
## GroupSetupA:Lanechangenumber -0.1    0.8 -1.1  -0.1   0.9
## GroupSetupB:Lanechangenumber -0.1    0.8 -1.2  -0.1   1.0
## sigma                         2.4    0.4  1.9   2.3   2.9
##
## Fit Diagnostics:
##            mean   sd   10%   50%   90%
## mean_PPD 4.0    0.7  3.2   4.0   4.8
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (f
or details see help('summary.stanreg')).
##
## MCMC diagnostics
##                              mcse Rhat n_eff
## (Intercept)                  0.0  1.0  1841
## GroupSetupA                  0.1  1.0  1734
## GroupSetupB                  0.1  1.0  1962
## Lanechangenumber             0.0  1.0  1838
## GroupSetupA:Lanechangenumber 0.0  1.0  1693
## GroupSetupB:Lanechangenumber 0.0  1.0  1862
## sigma                        0.0  1.0  2194
## mean_PPD                     0.0  1.0  3110
## log-posterior                0.1  1.0  1269
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective s
ample size, and Rhat is the potential scale reduction factor on split chains (at convergence Rhat=
1).
```

```
bind_rows(
    posterior(MODEL_1),
    posterior(MODEL_2)) %>%
  fixef()
```

model	fixef	center	lower	upper
MODEL_1	Intercept	5.3565110	2.826167	7.8932750
MODEL_1	GroupSetupA	-0.1151628	-2.302571	2.0306623
MODEL_1	GroupSetupB	0.8554638	-1.429531	3.0639058
MODEL_1	Lanechangenumber	-0.6012728	-1.360115	0.1144867
MODEL_2	Intercept	5.1313414	1.945467	8.4964925
MODEL_2	GroupSetupA	0.2758619	-4.170254	4.7523961
MODEL_2	GroupSetupB	1.0761396	-3.513137	5.8297656
MODEL_2	Lanechangenumber	-0.5231351	-1.654257	0.5209199
MODEL_2	GroupSetupA:Lanechangenumber	-0.1450172	-1.611781	1.4189035
MODEL_2	GroupSetupB:Lanechangenumber	-0.1015895	-1.808949	1.5623952

```
DF_1 %>%
  mutate(pred_MODEL_2 = predict(MODEL_2)$center) %>%
  ggplot(aes(x = Lanechangenumber, col = Group , y = pred_MODEL_2)) +
  geom_jitter() +
  geom_point()
```



The same holds for the lanechangedistance;

Distance

```
MODEL_3 <- stan_glm(Lanechangedistance ~ 1 + Group + Lanechangenumber, data = DF_1)

##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.134 seconds (Warm-up)
## Chain 1:                0.078 seconds (Sampling)
## Chain 1:                0.212 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
```

```
## Chain 2: Adjust your expectations accordingly!
## Chain 2: 
## Chain 2: 
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2: 
## Chain 2:  Elapsed Time: 0.142 seconds (Warm-up)
## Chain 2:                0.074 seconds (Sampling)
## Chain 2:                0.216 seconds (Total)
## Chain 2: 
## 
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3: 
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3: 
## Chain 3: 
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3: 
## Chain 3:  Elapsed Time: 0.11 seconds (Warm-up)
## Chain 3:                0.082 seconds (Sampling)
## Chain 3:                0.192 seconds (Total)
## Chain 3: 
## 
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4: 
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4: 
## Chain 4: 
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4: 
```

```
## Chain 4:  Elapsed Time: 0.139 seconds (Warm-up)
## Chain 4:                0.091 seconds (Sampling)
## Chain 4:                0.23 seconds (Total)
## Chain 4:
```

```
summary(MODEL_3)
```

```
##
## Model Info:
##  function:     stan_glm
##  family:       gaussian [identity]
##  formula:      Lanechangedistance ~ 1 + Group + Lanechangenumber
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 25
##  predictors:   4
##
## Estimates:
##                    mean   sd    10%   50%   90%
## (Intercept)       40.8   18.0  18.4  40.7  63.9
## GroupSetupA        1.9   15.6 -17.7   1.7  21.3
## GroupSetupB       -5.1   15.4 -24.6  -5.5  14.5
## Lanechangenumber  -1.1    5.3  -7.9  -1.1   5.6
## sigma             32.7    5.3  26.4  32.0  39.7
##
## Fit Diagnostics:
##            mean   sd   10%   50%   90%
## mean_PPD 36.9    9.2 25.1  37.1  48.6
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (f
or details see help('summary.stanreg')).
##
## MCMC diagnostics
##                  mcse Rhat n_eff
## (Intercept)      0.3  1.0  3168
## GroupSetupA      0.3  1.0  3266
## GroupSetupB      0.3  1.0  3160
## Lanechangenumber 0.1  1.0  3643
## sigma            0.1  1.0  3194
## mean_PPD         0.1  1.0  4030
## log-posterior    0.0  1.0  1377
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective s
ample size, and Rhat is the potential scale reduction factor on split chains (at convergence Rhat=
1).
```

```
fixef(MODEL_3)
```

fixef	center	lower	upper
Intercept	40.672617	5.427809	77.017891
GroupSetupA	1.749755	-28.698550	33.032645
GroupSetupB	-5.471005	-34.935443	25.003247
Lanechangenumber	-1.116519	-11.406943	9.554451

```
MODEL_4 <- stan_glm(Lanechangedistance ~ Group + Lanechangenumber + Group:Lanechangenumber, data =
 DF_1)
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
```

```
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.322 seconds (Warm-up)
## Chain 1:                0.216 seconds (Sampling)
## Chain 1:                0.538 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.35 seconds (Warm-up)
## Chain 2:                0.234 seconds (Sampling)
## Chain 2:                0.584 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
```

```
## Chain 3:
## Chain 3:  Elapsed Time: 0.265 seconds (Warm-up)
## Chain 3:                0.19 seconds (Sampling)
## Chain 3:                0.455 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.223 seconds (Warm-up)
## Chain 4:                0.206 seconds (Sampling)
## Chain 4:                0.429 seconds (Total)
## Chain 4:
```

```
summary(MODEL_4)
```

```
##
## Model Info:
##  function:     stan_glm
##  family:       gaussian [identity]
##  formula:      Lanechangedistance ~ Group + Lanechangenumber + Group:Lanechangenumber
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 25
##  predictors:   6
##
## Estimates:
##                                mean   sd    10%   50%   90%
## (Intercept)                   45.3   23.6  15.8  45.1  75.6
## GroupSetupA                   -1.8   31.9 -41.6  -2.3  38.9
## GroupSetupB                  -18.1   33.0 -59.5 -17.6  22.5
## Lanechangenumber              -2.9    7.9 -12.8  -2.9   6.9
## GroupSetupA:Lanechangenumber   1.6   10.9 -12.1   1.8  15.4
## GroupSetupB:Lanechangenumber   5.4   11.8  -9.2   5.4  20.0
## sigma                         33.7    5.6  27.1  33.0  40.9
##
## Fit Diagnostics:
##            mean   sd   10%   50%   90%
## mean_PPD 37.0    9.4 25.3  37.1  48.8
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (f
or details see help('summary.stanreg')).
##
## MCMC diagnostics
##                              mcse Rhat n_eff
## (Intercept)                  0.6  1.0  1651
## GroupSetupA                  0.8  1.0  1668
## GroupSetupB                  0.8  1.0  1655
```

```
## Lanechangenumber              0.2  1.0  1565
## GroupSetupA:Lanechangenumber 0.3  1.0  1556
## GroupSetupB:Lanechangenumber 0.3  1.0  1506
## sigma                         0.1  1.0  2278
## mean_PPD                      0.2  1.0  3100
## log-posterior                 0.1  1.0  1250
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective s
ample size, and Rhat is the potential scale reduction factor on split chains (at convergence Rhat=
1).
```

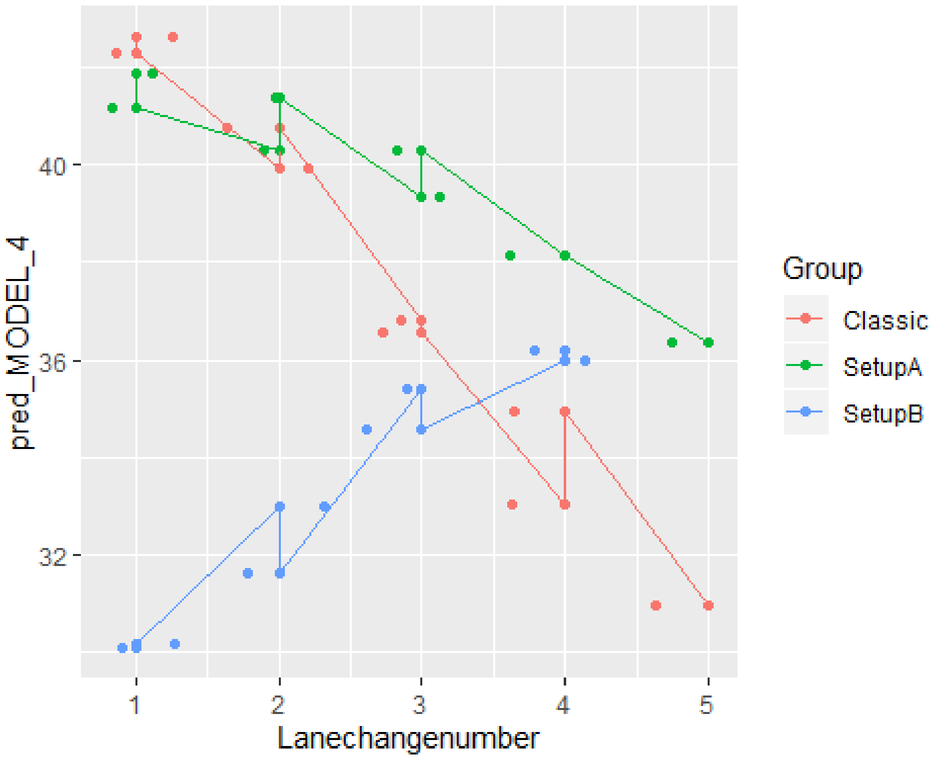
```
summary(MODEL_4)
```

```
##
## Model Info:
##  function:     stan_glm
##  family:       gaussian [identity]
##  formula:      Lanechangedistance ~ Group + Lanechangenumber + Group:Lanechangenumber
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 25
##  predictors:   6
##
## Estimates:
##                                mean   sd    10%   50%   90%
## (Intercept)                   45.3   23.6  15.8  45.1  75.6
## GroupSetupA                   -1.8   31.9 -41.6  -2.3  38.9
## GroupSetupB                  -18.1   33.0 -59.5 -17.6  22.5
## Lanechangenumber              -2.9    7.9 -12.8  -2.9   6.9
## GroupSetupA:Lanechangenumber   1.6   10.9 -12.1   1.8  15.4
## GroupSetupB:Lanechangenumber   5.4   11.8  -9.2   5.4  20.0
## sigma                         33.7    5.6  27.1  33.0  40.9
##
## Fit Diagnostics:
##            mean   sd   10%   50%   90%
## mean_PPD 37.0    9.4 25.3  37.1  48.8
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (f
or details see help('summary.stanreg')).
##
## MCMC diagnostics
##                              mcse Rhat n_eff
## (Intercept)                  0.6  1.0  1651
## GroupSetupA                  0.8  1.0  1668
## GroupSetupB                  0.8  1.0  1655
## Lanechangenumber             0.2  1.0  1565
## GroupSetupA:Lanechangenumber 0.3  1.0  1556
## GroupSetupB:Lanechangenumber 0.3  1.0  1506
## sigma                        0.1  1.0  2278
## mean_PPD                     0.2  1.0  3100
## log-posterior                0.1  1.0  1250
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective s
ample size, and Rhat is the potential scale reduction factor on split chains (at convergence Rhat=
1).
```

```
bind_rows(
    posterior(MODEL_3),
    posterior(MODEL_4)) %>%
  fixef()
```

model	fixef	center	lower	upper
MODEL_3	Intercept	40.672617	5.427809	77.017891
MODEL_3	GroupSetupA	1.749755	-28.698550	33.032645
MODEL_3	GroupSetupB	-5.471005	-34.935443	25.003247
MODEL_3	Lanechangenumber	-1.116519	-11.406943	9.554451
MODEL_4	Intercept	45.126568	-2.305253	92.369922
MODEL_4	GroupSetupA	-2.273670	-64.132257	60.034851
MODEL_4	GroupSetupB	-17.590545	-81.681366	48.460745
MODEL_4	Lanechangenumber	-2.926208	-18.663295	12.837666
MODEL_4	GroupSetupA:Lanechangenumber	1.787422	-20.116442	22.409607
MODEL_4	GroupSetupB:Lanechangenumber	5.418148	-17.940773	28.807589

```
DF_1 %>%
  mutate(pred_MODEL_4 = predict(MODEL_4)$center) %>%
  ggplot(aes(x = Lanechangenumber, col = Group , y = pred_MODEL_4)) +
  geom_jitter() +
  geom_point() +
  geom_line()
```

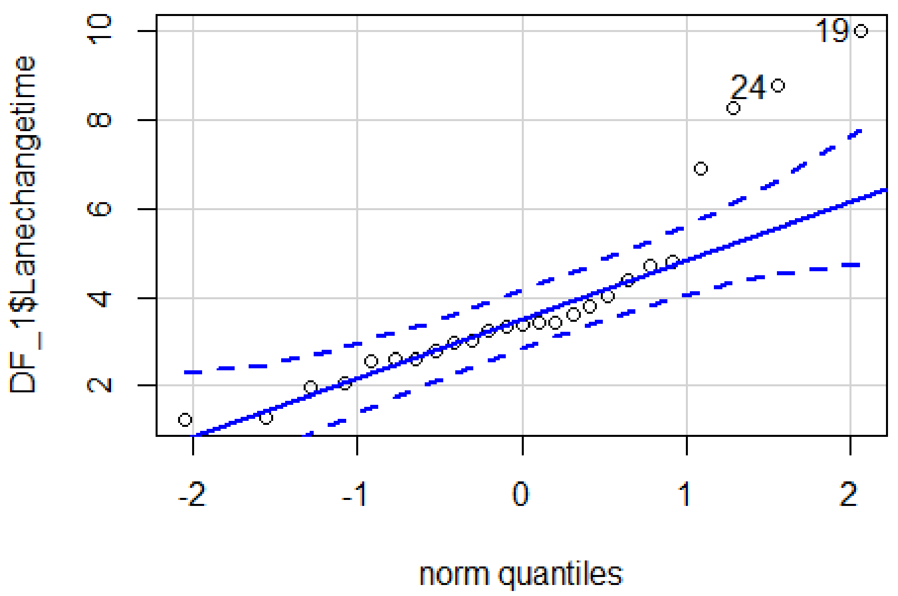


Test for group differences

A Group difference test can be performed to see if there is a difference in lane change time and lane change distance for each group. At first a normality test should be performed to see if the data is normally distributed. At first the qqPlot can be used. If all the point fall approximately along the reference line we assume that the data is normally distributed. Then, a Shapiro-Wilk normality test should be performed. If the p-value obtained after performing the test is higher than 0,05, this will imply that the distribution of the data is not significantly different from a normal distribution and normality can be assumed. In this case, an one-way anova test can be performed to see if there are significant differences between the groups. Else, use a non-parametric test like the Kruskal-Wallis test.

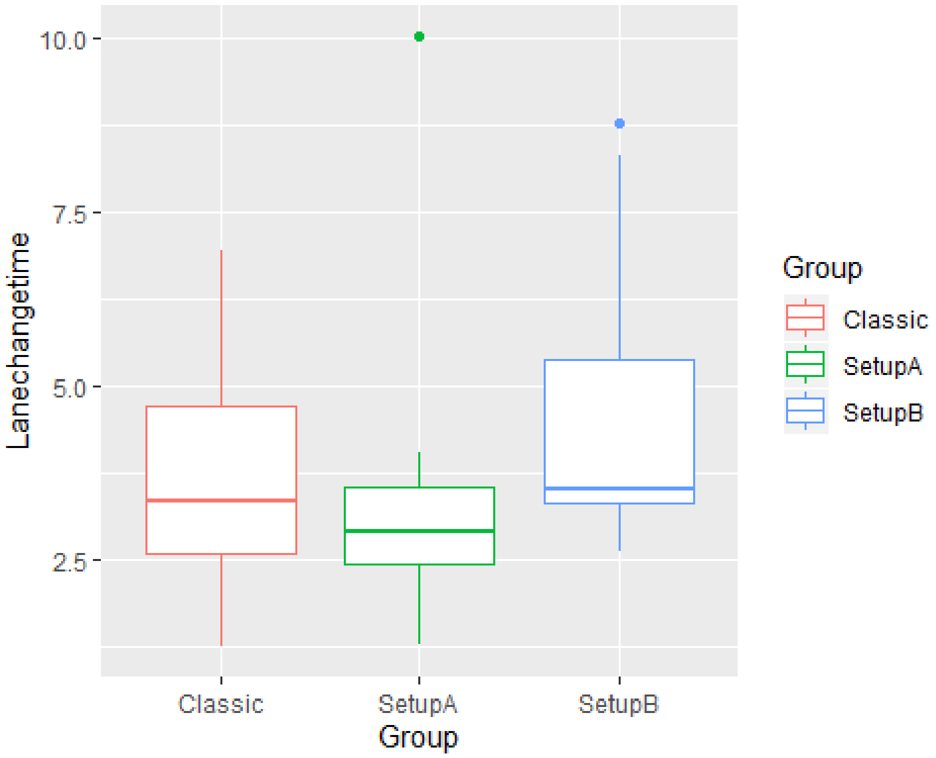
Time

```
qqPlot(DF_1$Lanechangetime)
```



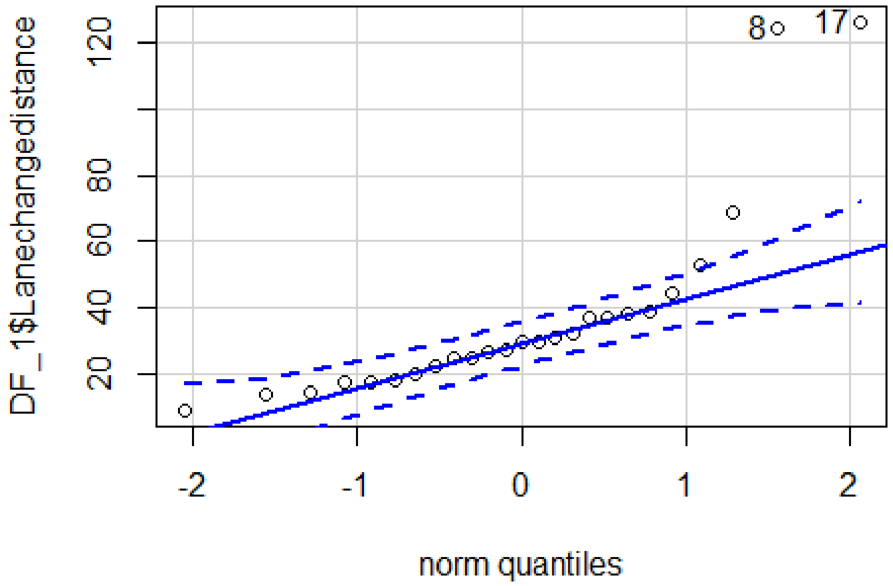

```
## [1] 19 24

shapiro.test(DF_1$Lanechangetime)

##
##  Shapiro-Wilk normality test
##
## data:  DF_1$Lanechangetime
## W = 0.83214, p-value = 0.000822

kruskal.test(Lanechangetime ~ Group, data = DF_1)

##
##  Kruskal-Wallis rank sum test
##
## data:  Lanechangetime by Group
## Kruskal-Wallis chi-squared = 1.9917, df = 2, p-value = 0.3694
```

```
DF_1 %>%
  ggplot(aes(x = Group, col = Group,y = Lanechangetime)) +
  geom_boxplot()
```

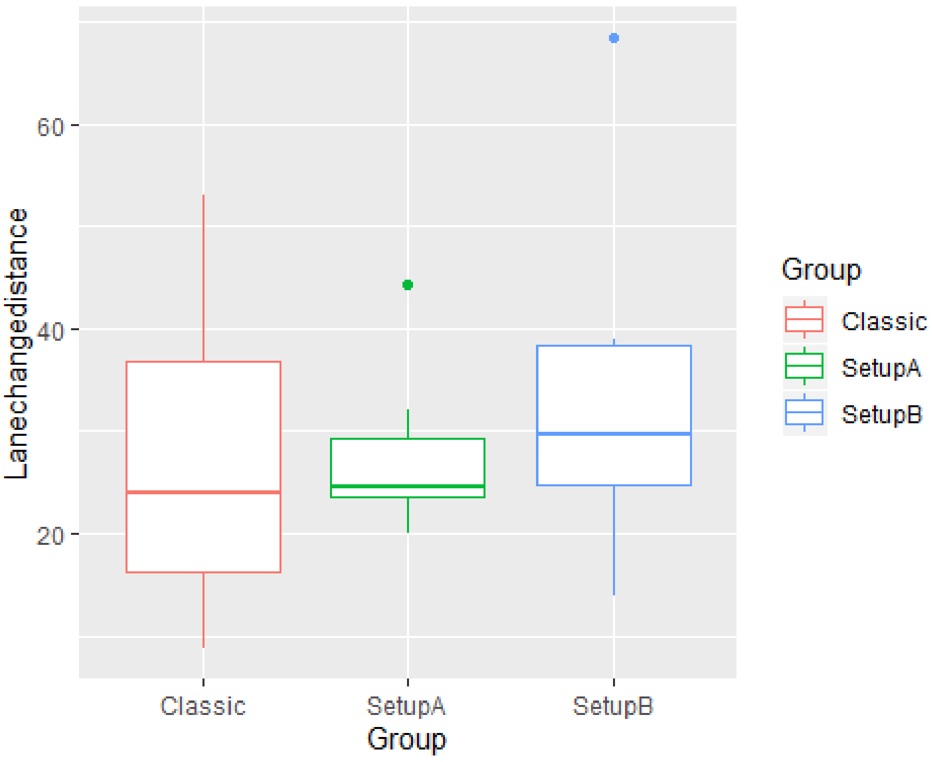


Distance

```
shapiro.test(DF_1$Lanechangedistance)

##
##  Shapiro-Wilk normality test
##
## data:  DF_1$Lanechangedistance
## W = 0.6997, p-value = 7.221e-06

qqPlot(DF_1$Lanechangedistance)
```




```
## [1] 17  8

kruskal.test(Lanechangedistance ~ Group, data = DF_1)

##
##  Kruskal-Wallis rank sum test
##
## data:  Lanechangedistance by Group
## Kruskal-Wallis chi-squared = 0.22359, df = 2, p-value = 0.8942

DF_1 %>%
  filter(Lanechangedistance < 100) %>%
  ggplot(aes(x = Group, col = Group,y = Lanechangedistance)) +
  geom_boxplot()
```


Lanechangedistance
60
40
20
Classic
SetupA
SetupB
Group
Group
Classic
SetupA
SetupB

subjective

Sarah Koning

16-4-2020

```
library(tidyverse)

## -- Attaching packages ------------------------------------------------------------------------
---------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ---------------------------------------------------------------------------------
---------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(gridExtra)

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine

library(rstanarm)

## Loading required package: Rcpp

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo

## rstanarm (Version 2.19.2, packaged: 2019-10-01 20:20:33 UTC)

## - Do not expect the default priors to remain the same in future rstanarm versions.

## Thus, R scripts should specify priors explicitly, even if they are just the defaults.

## - For execution on a local, multicore CPU with excess RAM we recommend calling

## options(mc.cores = parallel::detectCores())

## - bayesplot theme set to bayesplot::theme_default()

##    * Does _not_ affect other ggplot2 plots

##    * See ?bayesplot_theme_set for details on theme setting

library(bayr)

## Registered S3 methods overwritten by 'bayr':
##   method          from
##   coef.stanreg    rstanarm
##   predict.stanreg rstanarm

##
## Attaching package: 'bayr'
```

```
## The following objects are masked from 'package:rstanarm':
##
##     fixef, ranef

library(reshape2)

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

Data analysis package subjective data

This part contains the code that can be used for the data analysis of the subjective data. Make sure that the data set contains.

- Rows with each individual score on the Nasa TLX
- The scores on Confidence for both the parking and lane change tasks (LaneConfidence & ParkConfidence)
- the individual scored on the UEQ (Perspicuity and dependability)
- The participant number (Participant)
- The Setup that was used (Group)

Loading the data

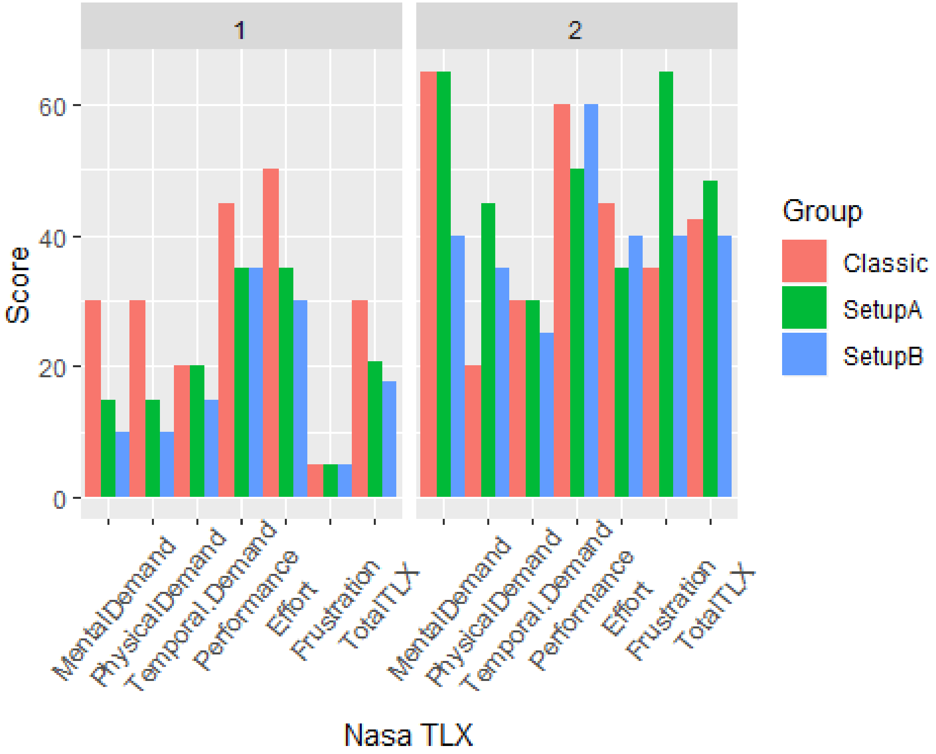
Code for loading the dataset:

```
DF_2 <- read.csv2("subjective.csv")

summary(DF_2)

##   MentalDemand   PhysicalDemand  Temporal.Demand  Performance       Effort
##  Min.   :10.00   Min.   :10.00   Min.   :15.00   Min.   :35.0   Min.   :30.00
##  1st Qu.:18.75   1st Qu.:16.25   1st Qu.:20.00   1st Qu.:37.5   1st Qu.:35.00
##  Median :35.00   Median :25.00   Median :22.50   Median :47.5   Median :37.50
##  Mean   :37.50   Mean   :25.83   Mean   :23.33   Mean   :47.5   Mean   :39.17
##  3rd Qu.:58.75   3rd Qu.:33.75   3rd Qu.:28.75   3rd Qu.:57.5   3rd Qu.:43.75
##  Max.   :65.00   Max.   :45.00   Max.   :30.00   Max.   :60.0   Max.   :50.00
##   Frustration       TotalTLX      LaneConfidence  ParkConfidence
##  Min.   : 5.00   Min.   :17.50   Min.   :7.000   Min.   :2.000
##  1st Qu.: 5.00   1st Qu.:23.12   1st Qu.:7.625   1st Qu.:3.000
##  Median :20.00   Median :35.00   Median :8.250   Median :6.500
##  Mean   :25.83   Mean   :33.19   Mean   :8.083   Mean   :5.333
##  3rd Qu.:38.75   3rd Qu.:41.88   3rd Qu.:8.500   3rd Qu.:7.000
##  Max.   :65.00   Max.   :48.33   Max.   :9.000   Max.   :8.000
##   Perspicuity    Dependability     Participant      Group
##  Min.   :1.000   Min.   :0.2500   Min.   :1.0   Classic:2
##  1st Qu.:1.375   1st Qu.:0.3125   1st Qu.:1.0   SetupA :2
##  Median :2.000   Median :0.5000   Median :1.5   SetupB :2
##  Mean   :1.917   Mean   :0.7500   Mean   :1.5
##  3rd Qu.:2.438   3rd Qu.:1.2500   3rd Qu.:2.0
##  Max.   :2.750   Max.   :1.5000   Max.   :2.0
```

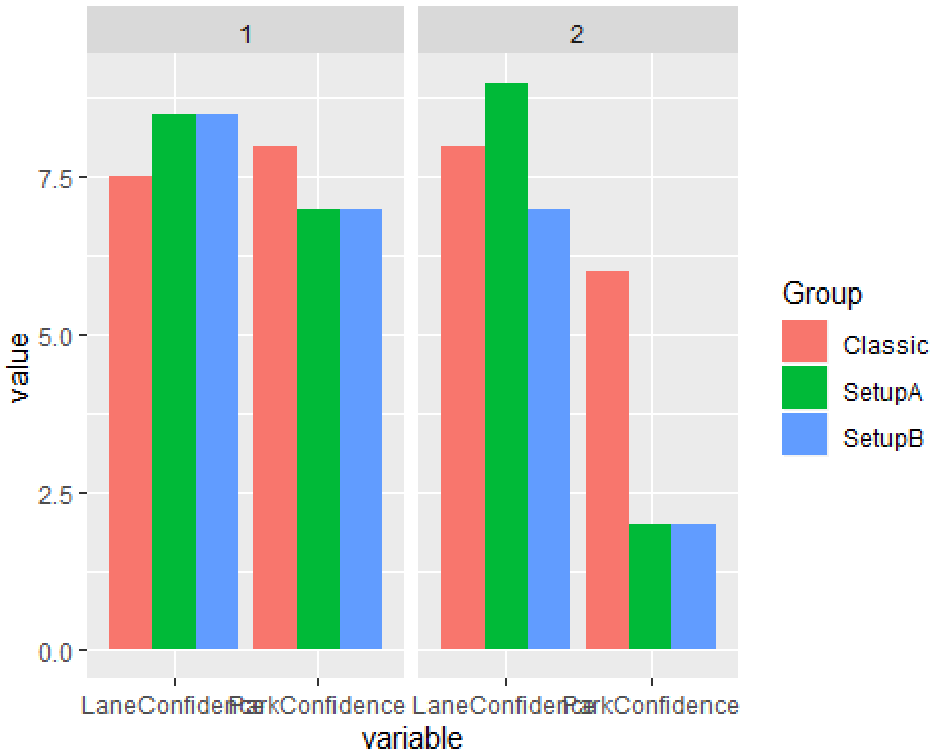
Data analysis

This code will give you a histogram per participant on the different scores of the NasaTLX, confidence scores and UEQ questionnaire per Setup:

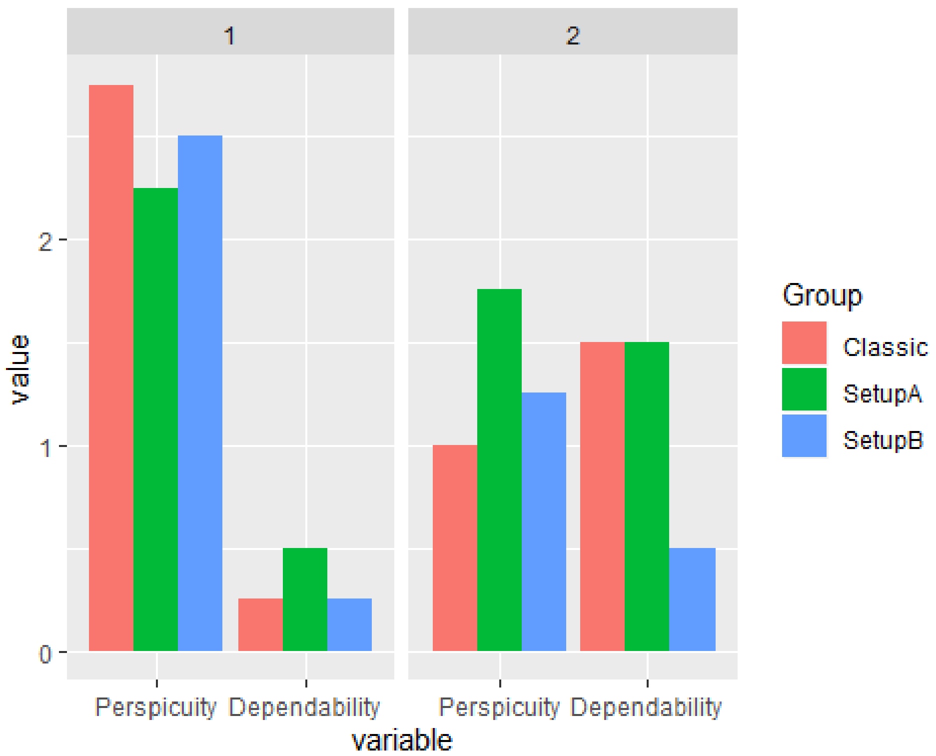
```
DF_2 %>%
  select(Group,Participant,MentalDemand,PhysicalDemand,Temporal.Demand,Performance,Effort,Frustrat
ion,TotalTLX) %>%
    melt(id = c("Group","Participant")) %>%
      ggplot(aes(x=variable, y=value, fill=Group,height = 1 , width = 1,)) +
        theme(axis.text.x=element_text(angle=50,vjust = 0.6,size = 10)) +
          labs(x="Nasa TLX", y = "Score") +
            facet_wrap(~Participant) +
              geom_bar(stat="identity", position=position_dodge())
```




```
DF_2 %>%
  select(Group,Participant, LaneConfidence, ParkConfidence) %>%
  melt(id = c("Group","Participant")) %>%
  ggplot(aes(x=variable, y=value, fill=Group)) +facet_wrap(~Participant) +
   geom_bar(stat="identity", position=position_dodge())
```




```
DF_2 %>%
  select(Group,Participant, Perspicuity, Dependability) %>%
  melt(id = c("Group","Participant")) %>%
  ggplot(aes(x=variable, y=value, fill=Group)) +facet_wrap(~Participant) +
   geom_bar(stat="identity", position=position_dodge())
```

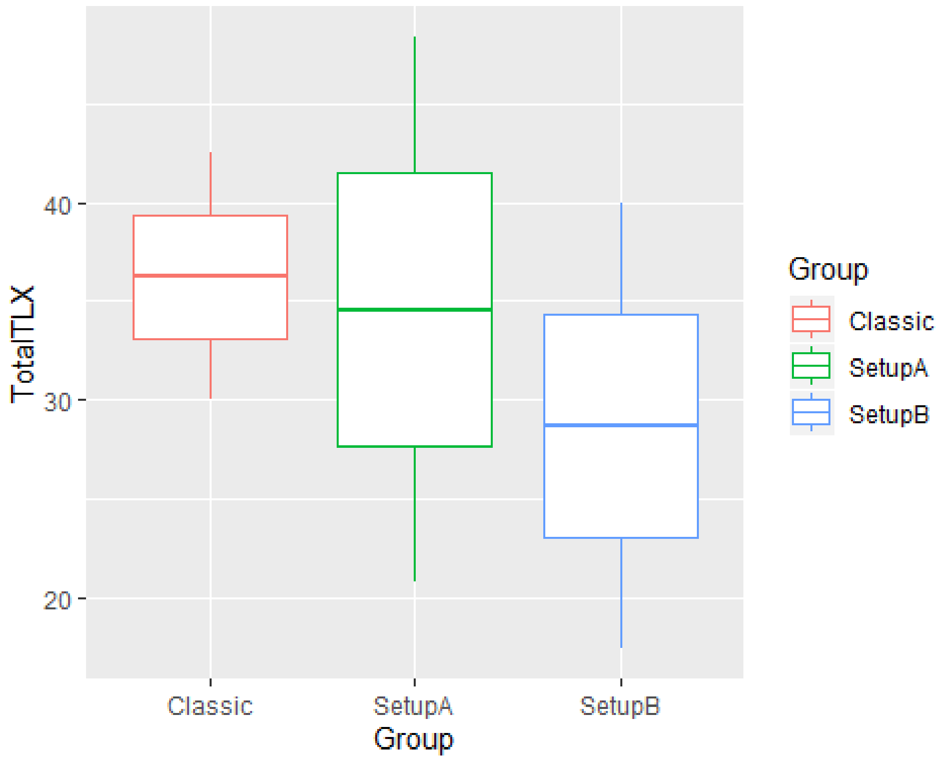


Group differences

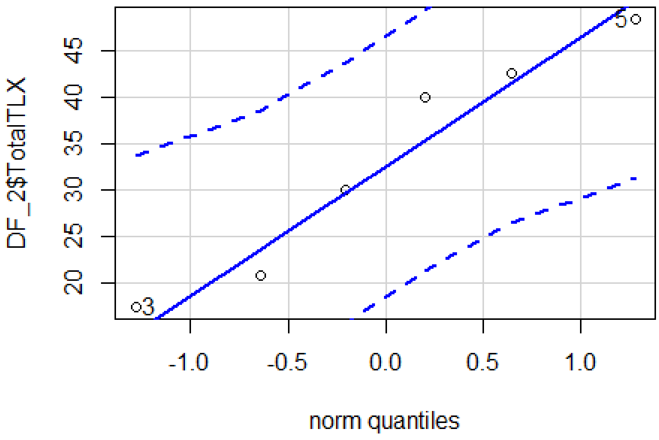
In the following section contains the code to obtain information on group differences for the NasaTLX, Confidence scores and UEQ . At first a normality test should be performed to look if the data is normally distributed. At first the qqPlot can be used. If all the point fall approximately along the reference line we assume that the data is normally distributed. Then, a Shapiro-Wilk normality test should be performed. If the p-value obtained after performing the test is higher than 0,05, this will imply that the distribution of the data is not significantly different from a normal distribution and normality can be assumed. In this case, an one-way anova test can be performed to see if there are significant differences between the groups. Else, use a non-parametric test like the Kruskal-Wallis test.

Nasatlx

```
DF_2 %>%
  ggplot(aes(x = Group,color = Group, y = TotalTLX)) +
  geom_boxplot()
```



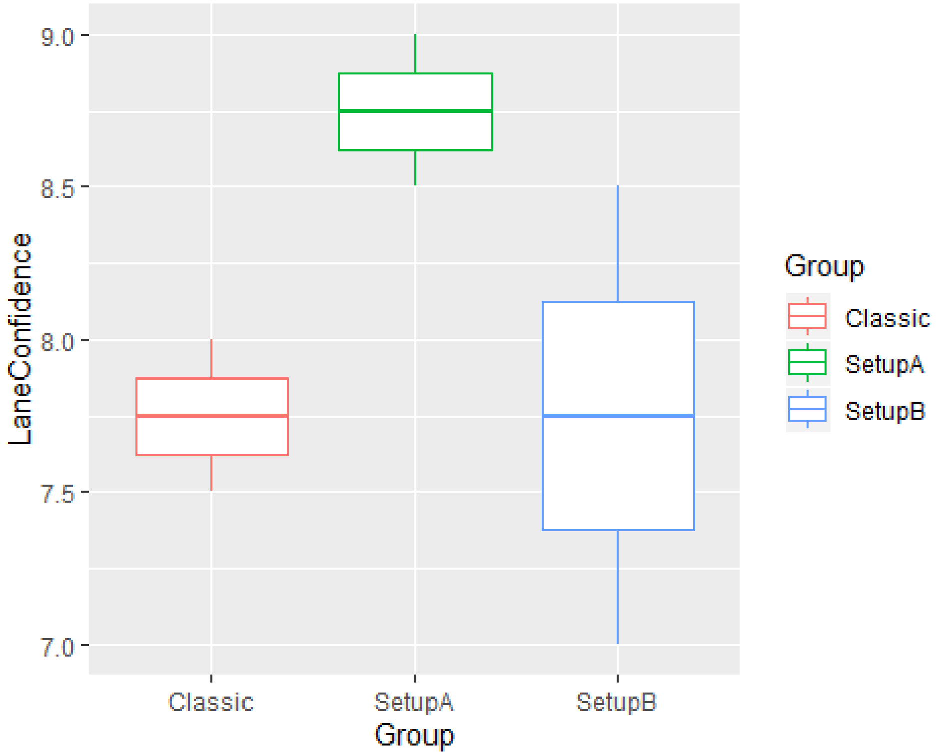

```
qqPlot(DF_2$TotalTLX)
```



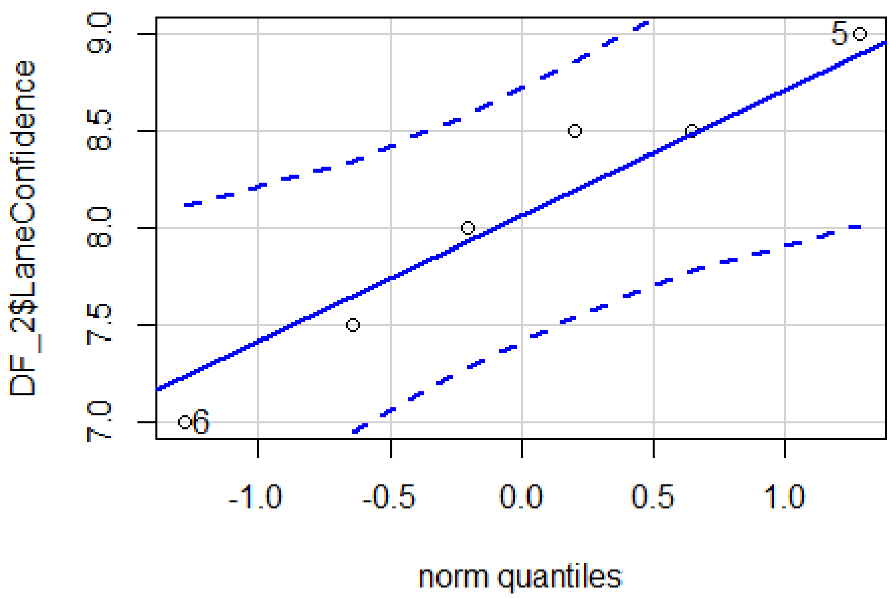

```
## [1] 3 5

shapiro.test(DF_2$TotalTLX)

##
##  Shapiro-Wilk normality test
##
## data:  DF_2$TotalTLX
## W = 0.93035, p-value = 0.5828

resTLX.aov <- aov(TotalTLX ~ Group, data = DF_2)
summary(resTLX.aov)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Group        2   62.0   31.02   0.131  0.882
## Residuals    3  709.4  236.46
```

Confidence

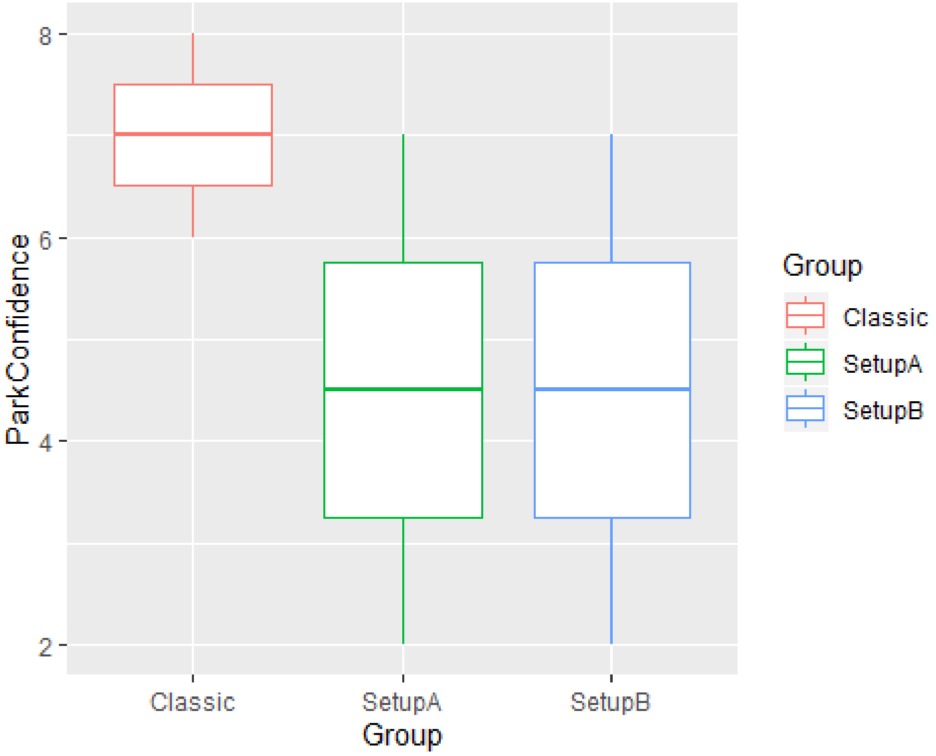
```
DF_2 %>%
  ggplot(aes(x = Group,color = Group, y = LaneConfidence)) +
  geom_boxplot()
```



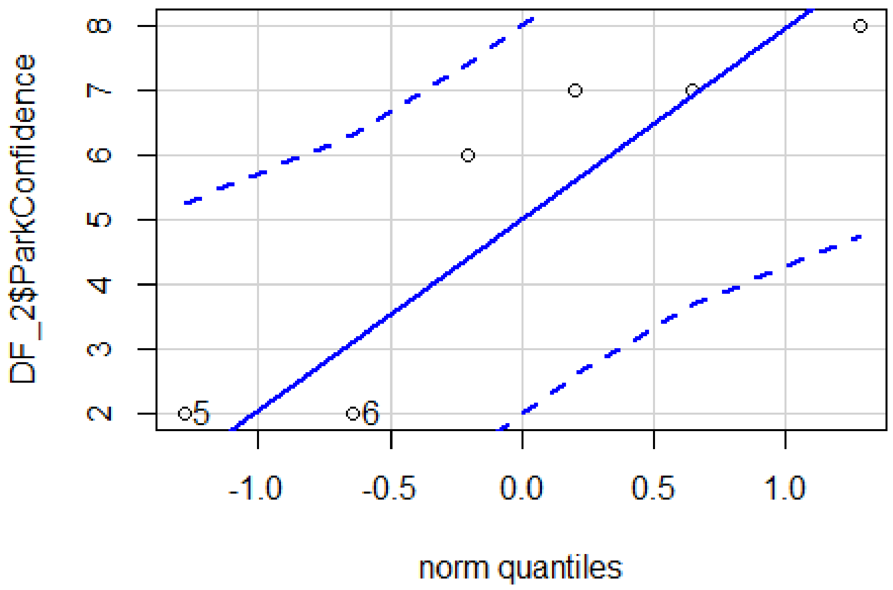

```
qqPlot(DF_2$LaneConfidence)
```



```
## [1] 6 5

shapiro.test(DF_2$LaneConfidence)

##
##  Shapiro-Wilk normality test
##
## data:  DF_2$LaneConfidence
## W = 0.95801, p-value = 0.8043

resLaneconf.aov <- aov(LaneConfidence ~ Group, data = DF_2)
summary(resLaneconf.aov)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Group        2  1.333  0.6667   1.455  0.362
## Residuals    3  1.375  0.4583

DF_2 %>%
  ggplot(aes(x = Group,color = Group, y = ParkConfidence)) +
  geom_boxplot()
```



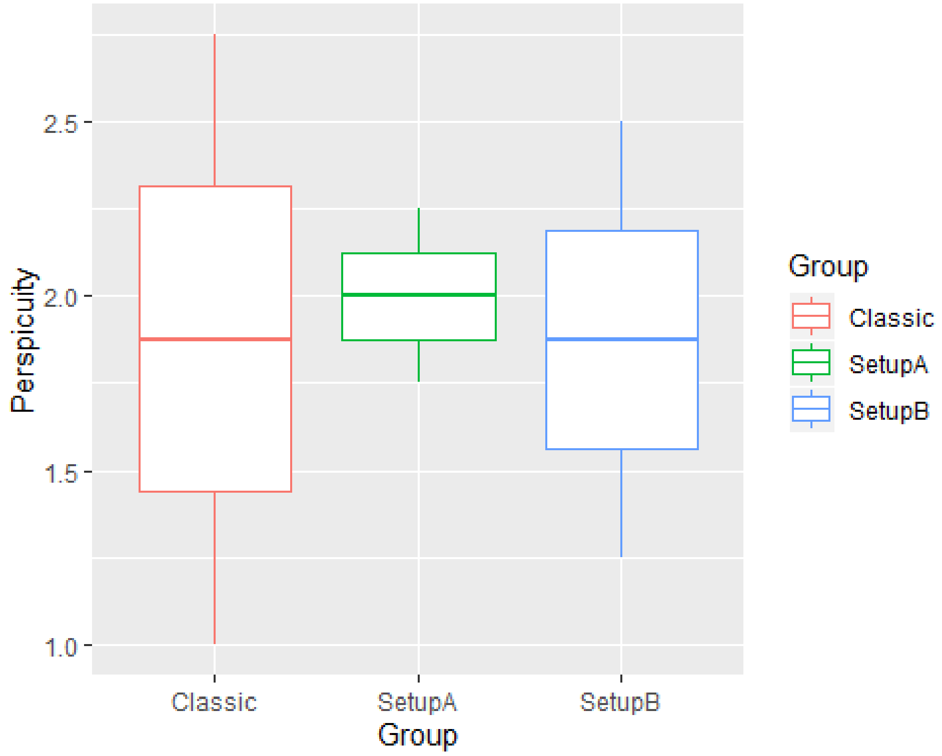

```
qqPlot(DF_2$ParkConfidence)
```



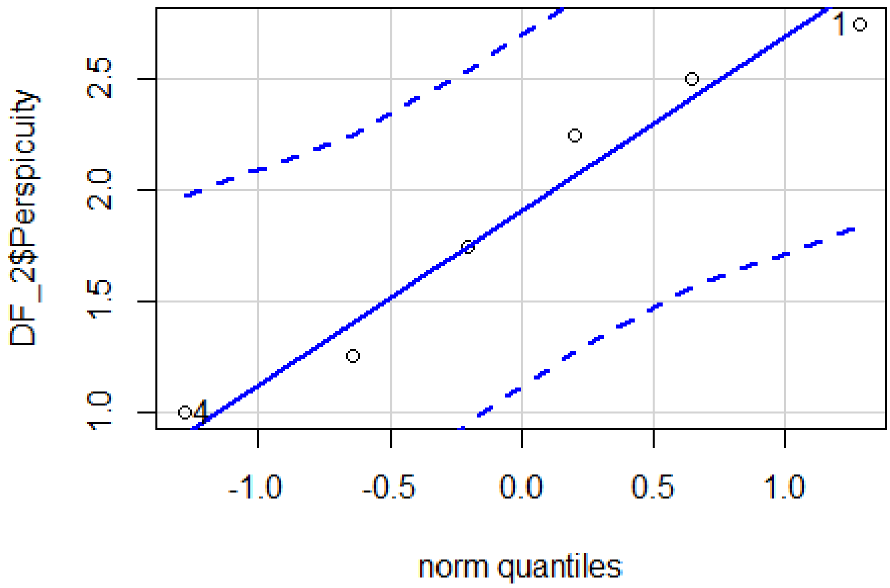

```
## [1] 5 6

shapiro.test(DF_2$ParkConfidence)

##
##  Shapiro-Wilk normality test
##
## data:  DF_2$ParkConfidence
## W = 0.80996, p-value = 0.07211

resParkconf.aov <- aov(ParkConfidence ~ Group, data = DF_2)
summary(resParkconf.aov)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Group        2  8.333   4.167   0.463  0.668
## Residuals    3 27.000   9.000
```

UEQ

```
DF_2 %>%
  ggplot(aes(x = Group,color = Group, y = Perspicuity)) +
  geom_boxplot()
```

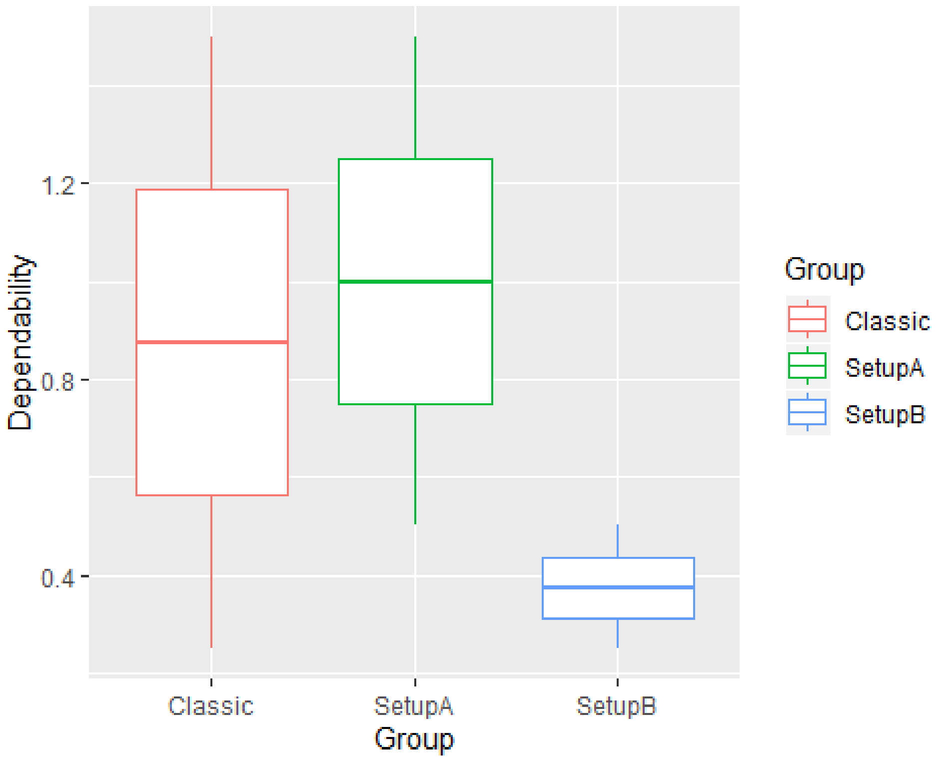



```
qqPlot(DF_2$Perspicuity)
```

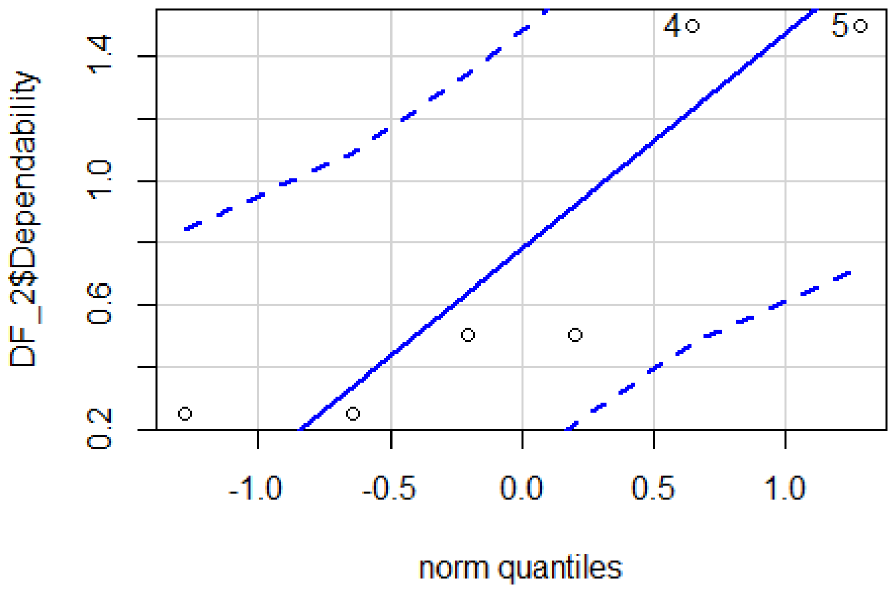



```
## [1] 4 1
```

```
shapiro.test(DF_2$Perspicuity)

##
##  Shapiro-Wilk normality test
##
## data:  DF_2$Perspicuity
## W = 0.94009, p-value = 0.6599

resPers.aov <- aov(Perspicuity ~ Group, data = DF_2)
summary(resPers.aov)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Group        2 0.0208  0.0104   0.013  0.987
## Residuals    3 2.4375  0.8125

DF_2 %>%
  ggplot(aes(x = Group,color = Group, y = Dependability)) +
  geom_boxplot()
```




```
qqPlot(DF_2$Dependability)
```




```
## [1] 4 5

shapiro.test(DF_2$Dependability)

##
##  Shapiro-Wilk normality test
##
## data:  DF_2$Dependability
## W = 0.76178, p-value = 0.02591

kruskal.test(Dependability ~ Group, data = DF_2)

##
##  Kruskal-Wallis rank sum test
##
## data:  Dependability by Group
## Kruskal-Wallis chi-squared = 1.25, df = 2, p-value = 0.5353
```

Eye-tracking data

Sarah Koning

28-4-2020

```
library(tidyverse)

## -- Attaching packages --------------------------------------------------------------------------
----------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts -----------------------------------------------------------------------------------
----------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(gridExtra)

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine

library(rstanarm)

## Loading required package: Rcpp

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo

## rstanarm (Version 2.19.2, packaged: 2019-10-01 20:20:33 UTC)

## - Do not expect the default priors to remain the same in future rstanarm versions.

## Thus, R scripts should specify priors explicitly, even if they are just the defaults.

## - For execution on a local, multicore CPU with excess RAM we recommend calling

## options(mc.cores = parallel::detectCores())

## - bayesplot theme set to bayesplot::theme_default()

##    * Does _not_ affect other ggplot2 plots

##    * See ?bayesplot_theme_set for details on theme setting

library(bayr)

## Registered S3 methods overwritten by 'bayr':
##   method          from
##   coef.stanreg    rstanarm
##   predict.stanreg rstanarm

##
## Attaching package: 'bayr'
```

```
## The following objects are masked from 'package:rstanarm':
##
##     fixef, ranef

library(reshape2)

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

Eye-tracking data

This part contains the code that is need to perform the analysis on the eye-tracking data. You should be able to export metrics on “Moving AOI results” from the eye motion software and load it into R. Most important here is that the average fixation duration for each AOI is present in the dataset. It can be useful to add some columns to this sheet on;

- The Setup that was used.(Setup)
- The % of the total time that was spent in an AOI by dividing the duration by the total time (timepercentoftotal)
- The participant number (Participant)

Loading the data

Code for loading the dataset:

```
DF_3 <-
  read.csv2("MovingAOIResult2.csv") %>%
  filter(Type == "Dynamic AOI")

summary(DF_3)

##       Study.name          Respondent.Name  Gender        Age        Group
##  study Sarah:18   P.3.3Btherightone:3      Male:18   Min.   :0   Default:18
##                   P2.1Classic      :3                1st Qu.:0
##                   P2.2A            :3                Median :0
##                   P2.3B            :3                Mean   :0
##                   P3.2A            :3                3rd Qu.:0
##                   P3.3classic      :3                Max.   :0
##
##          Type                        Label      Start..ms.      Duration..ms.
##  Dynamic AOI:18   leftmirrora           :2   Min.   :  6967   Min.   : 52188
##  Stimulus   : 0   leftmirrorb           :2   1st Qu.:  8416   1st Qu.:255373
##                   leftmirrorclassic     :2   Median : 11064   Median :298178
##                   rearviewmirrora       :2   Mean   : 20820   Mean   :281181
##                   rearviewmirrorb       :2   3rd Qu.: 13973   3rd Qu.:332954
##                   rearviewmirrorclassic:2   Max.   :123643   Max.   :416796
##                   (Other)               :6
##           ParentStimulus ParentStimulus_Start..ms.
##                   : 0     Min.   :770
##  ScreenRecording-1:18     1st Qu.:783
##                           Median :797
##                           Mean   :801
##                           3rd Qu.:822
##                           Max.   :837
##
##  ParentStimulus_Duration..ms. Hit.time.G..ms.  Time.spent.G..ms.
##  Min.   :334587               Min.   :  8133   Min.   : 4453
##  1st Qu.:341488               1st Qu.: 18059   1st Qu.: 7636
##  Median :365377               Median : 37570   Median : 9584
##  Mean   :390362               Mean   : 43788   Mean   :10244
##  3rd Qu.:463083               3rd Qu.: 48993   3rd Qu.:13630
##  Max.   :472260               Max.   :123089   Max.   :16701
##
##  Time.spent.G.... Respondent.ratio.G Revisit.G..Revisitors.
##  Min.   : 2.000   Min.   :1          Min.   :1
##  1st Qu.: 3.000   1st Qu.:1          1st Qu.:1
##  Median : 4.000   Median :1          Median :1
##  Mean   : 4.278   Mean   :1          Mean   :1
##  3rd Qu.: 4.750   3rd Qu.:1          3rd Qu.:1
##  Max.   :11.000   Max.   :1          Max.   :1
##
```

```
##  Revisit.G..Visitors. Revisit.G..Revisits.  TTFF.F..ms.     Time.spent.F..ms.
##  Min.   :1            Min.   :11.00        Min.   :  8169   Min.   : 3633
##  1st Qu.:1            1st Qu.:14.50        1st Qu.: 18147   1st Qu.: 6284
##  Median :1            Median :18.50        Median : 26508   Median : 7976
##  Mean   :1            Mean   :19.94        Mean   : 42372   Mean   : 8280
##  3rd Qu.:1            3rd Qu.:25.50        3rd Qu.: 49026   3rd Qu.:11177
##  Max.   :1            Max.   :36.00        Max.   :123106   Max.   :12878
##
##  Time.spent.F.... Revisit.F..Revisitors. Revisit.F..Visitors.
##  Min.   :1.000    Min.   :1              Min.   :1
##  1st Qu.:2.000    1st Qu.:1              1st Qu.:1
##  Median :3.000    Median :1              Median :1
##  Mean   :3.389    Mean   :1              Mean   :1
##  3rd Qu.:4.000    3rd Qu.:1              3rd Qu.:1
##  Max.   :9.000    Max.   :1              Max.   :1
##
##  Revisit.F..Revisits. Fixations.Count First.Fixation.Duration..ms.
##  Min.   :13.00        Min.   :27.00   Min.   : 11.0
##  1st Qu.:16.00        1st Qu.:39.50   1st Qu.: 34.0
##  Median :20.00        Median :49.50   Median : 72.0
##  Mean   :21.72        Mean   :50.11   Mean   :142.0
##  3rd Qu.:27.00        3rd Qu.:53.75   3rd Qu.:212.8
##  Max.   :37.00        Max.   :88.00   Max.   :468.0
##
##  Average.Fixations.Duration..ms.  Mouse.Clicks timepercentoftotal  Participant
##  Min.   : 93.0                   Min.   :0     Min.   :1.248      Min.   :1.0
##  1st Qu.:126.2                   1st Qu.:0     1st Qu.:1.758      1st Qu.:1.0
##  Median :160.5                   Median :0     Median :2.627      Median :1.5
##  Mean   :166.7                   Mean   :0     Mean   :2.684      Mean   :1.5
##  3rd Qu.:208.0                   3rd Qu.:0     3rd Qu.:3.506      3rd Qu.:2.0
##  Max.   :243.0                   Max.   :0     Max.   :4.992      Max.   :2.0
##
##      Setup
##  Classic:6
##  SetupA :6
##  SetupB :6
##
##
##
##
```

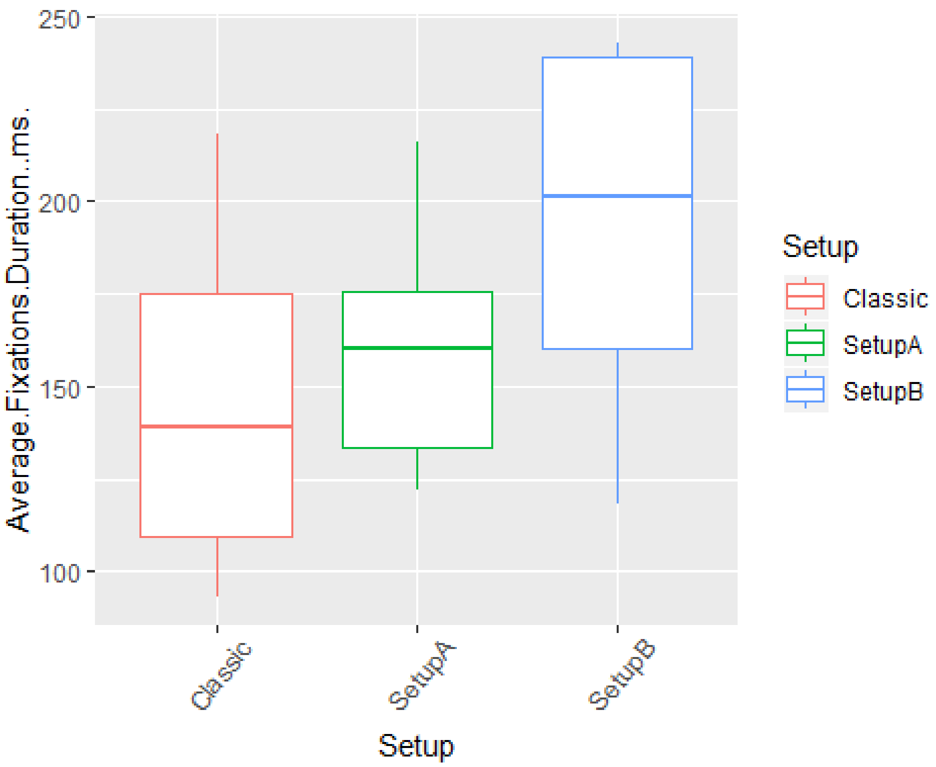
Average Fixation duration

This code will give an overview of the mean and sd of the different groups.

```
DF_3 %>%
group_by(Setup) %>%
  summarise(
    count = n(),
    mean = mean(Average.Fixations.Duration..ms., na.rm = TRUE),
    sd = sd(Average.Fixations.Duration..ms., na.rm = TRUE)
  )

## # A tibble: 3 x 4
##   Setup   count  mean    sd
##   <fct>   <int> <dbl> <dbl>
## 1 Classic     6  146   48.1
## 2 SetupA      6  161.  35.2
## 3 SetupB      6  194.  53.4
```

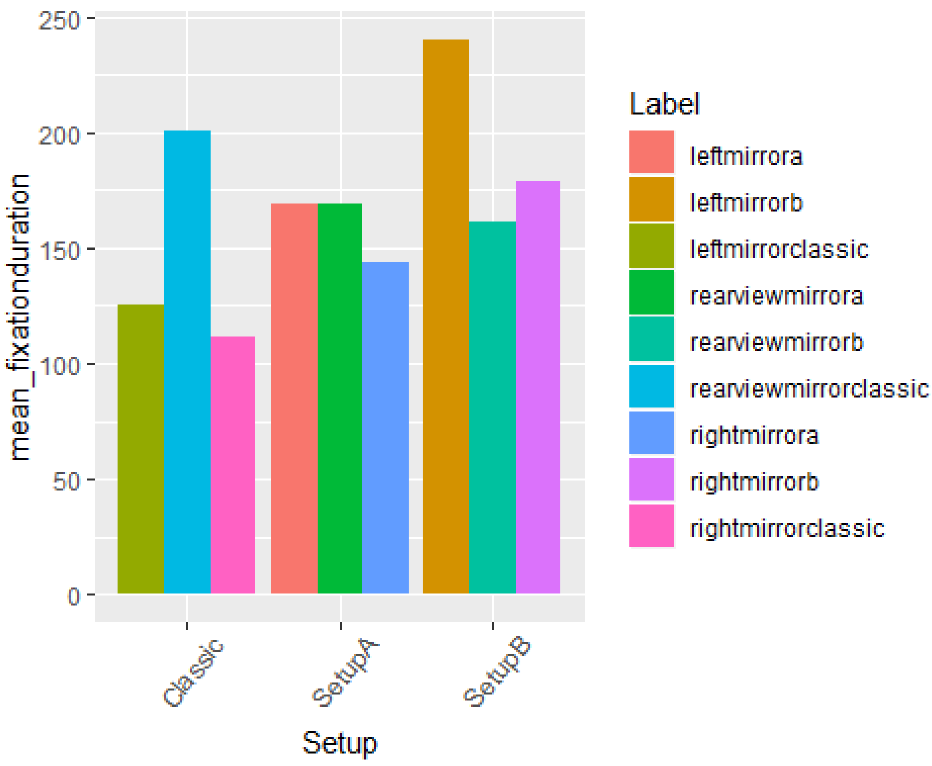
This code will give a boxplot for the different Setups:

```
DF_3 %>%
  filter(Type == "Dynamic AOI") %>%
  ggplot(aes(x = Setup,color = Setup, y = Average.Fixations.Duration..ms.)) +
   theme(axis.text.x=element_text(angle=50,vjust = 0.6,size = 10)) +
  geom_boxplot()
```



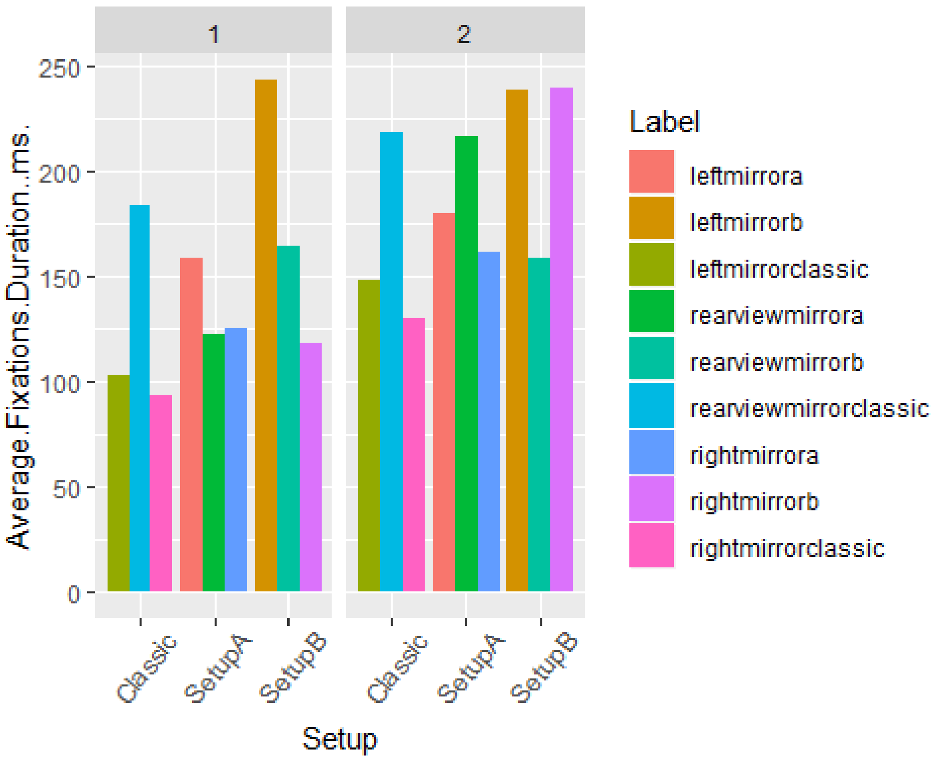
This code will give an histogram For the average fixation duration for each mirror per Setup:

```
DF_3 %>%
  group_by(Label, Setup) %>%
  summarize(mean_fixationduration = mean(Average.Fixations.Duration..ms., na.rm = T),
            sd_fixationduration = sd(Average.Fixations.Duration..ms., na.rm = T)) %>%
  ggplot(aes(x=Setup, y=mean_fixationduration, fill= Label)) +
  theme(axis.text.x=element_text(angle=50,vjust = 0.6,size = 10)) +
  geom_bar(stat="identity", position=position_dodge())
```



This code will give an histogram for the average fixation duration for each mirror per Setup per participant:

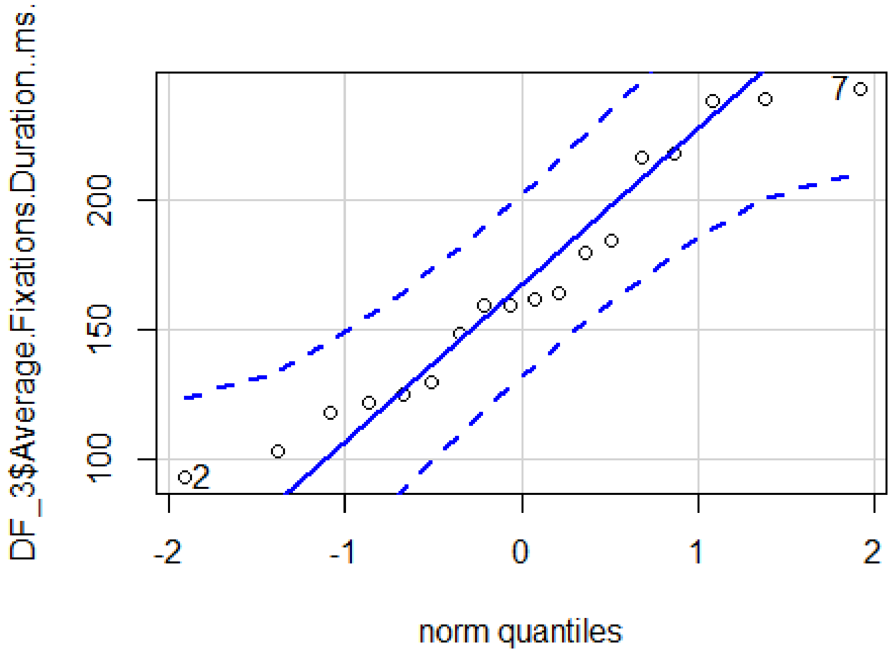
```
DF_3 %>%
  filter(Type == "Dynamic AOI") %>%
  ggplot(aes(x=Setup, y=Average.Fixations.Duration..ms., fill=Label)) +
  facet_wrap(~Participant)+
  theme(axis.text.x=element_text(angle=50,vjust = 0.6,size = 10)) +
  geom_bar(stat="identity", position=position_dodge())
```



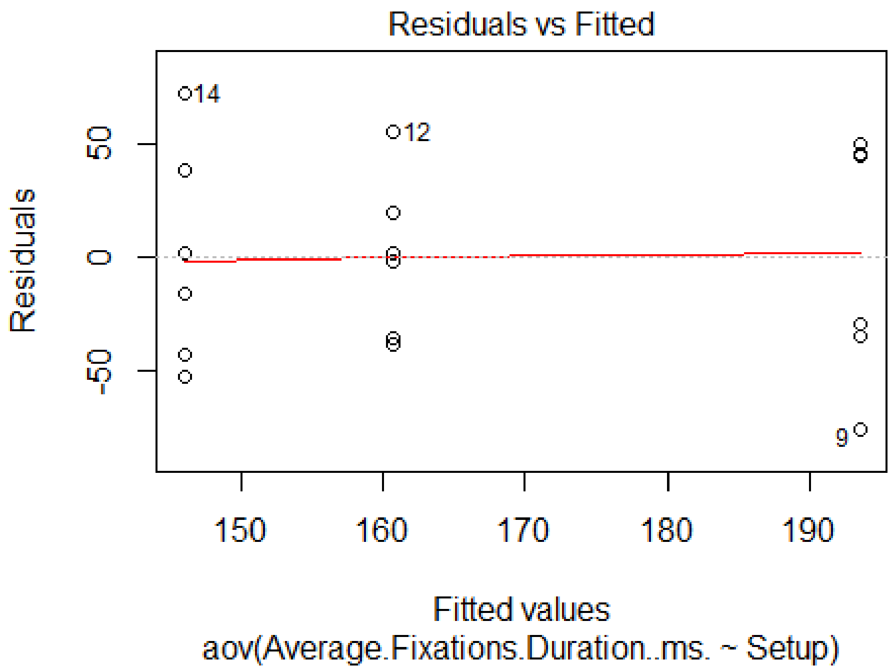
Group differences

Underneath is calculated if there was a significant difference between the Setups on average fixation duration. At first a normality test should be performed to see if the data is normally distributed. At first the qqPlot can be used. If all the point fall approximately along the reference line we assume that the data is normally distributed. Then, a Shapiro-Wilk normality test should be performed. If the p-value obtained after performing the test is higher than 0,05, this will imply that the distribution of the data is not significantly different from a normal distribution and normality can be assumed. In this case, an one-way anova test can be performed to see if there are significant differences between the groups. Else, use a non-parametric test like the Kruskal-Wallis test.

```
res.aov <- aov(Average.Fixations.Duration..ms. ~ Setup, data = DF_3)
summary(res.aov)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Setup        2   7099    3549   1.663  0.223
## Residuals   15  32015    2134

shapiro.test(DF_3$Average.Fixations.Duration..ms.)

##
##  Shapiro-Wilk normality test
##
## data:  DF_3$Average.Fixations.Duration..ms.
## W = 0.93747, p-value = 0.2621

qqPlot(DF_3$Average.Fixations.Duration..ms.)
```




```
## [1] 7 2
plot(res.aov, 1)
```



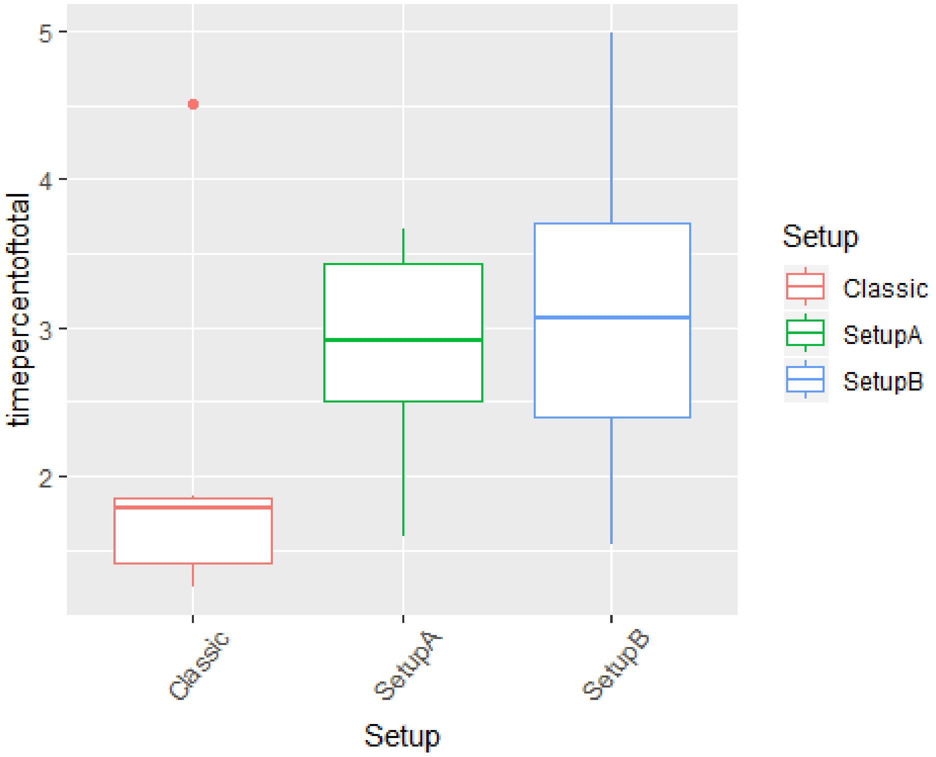
Time spent %

This code will give an overview of the mean and sd of the different groups.

```
DF_3 %>%
group_by(Setup) %>%
  summarise(
    count = n(),
    mean = mean(timepercentoftotal, na.rm = TRUE),
    sd = sd(timepercentoftotal, na.rm = TRUE)
  )

## # A tibble: 3 x 4
##   Setup   count  mean    sd
##   <fct>   <int> <dbl> <dbl>
## 1 Classic     6  2.08 1.22
## 2 SetupA      6  2.85 0.776
## 3 SetupB      6  3.12 1.22
```

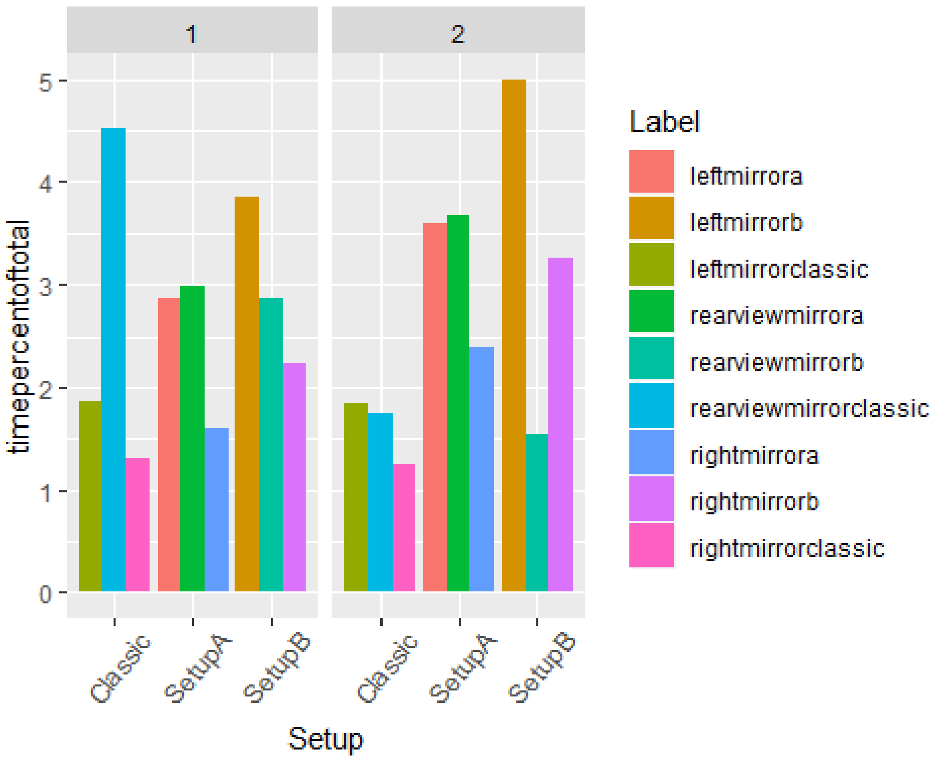
This code will give a boxplot for the diffrent Setups:

```
DF_3 %>%
  filter(Type == "Dynamic AOI") %>%
  ggplot(aes(x = Setup,color = Setup, y = timepercentoftotal)) +
   theme(axis.text.x=element_text(angle=50,vjust = 0.6,size = 10)) +
  geom_boxplot()
```



This code will give an histogram For the percentage of time spent for each mirror per Setup for each participant:

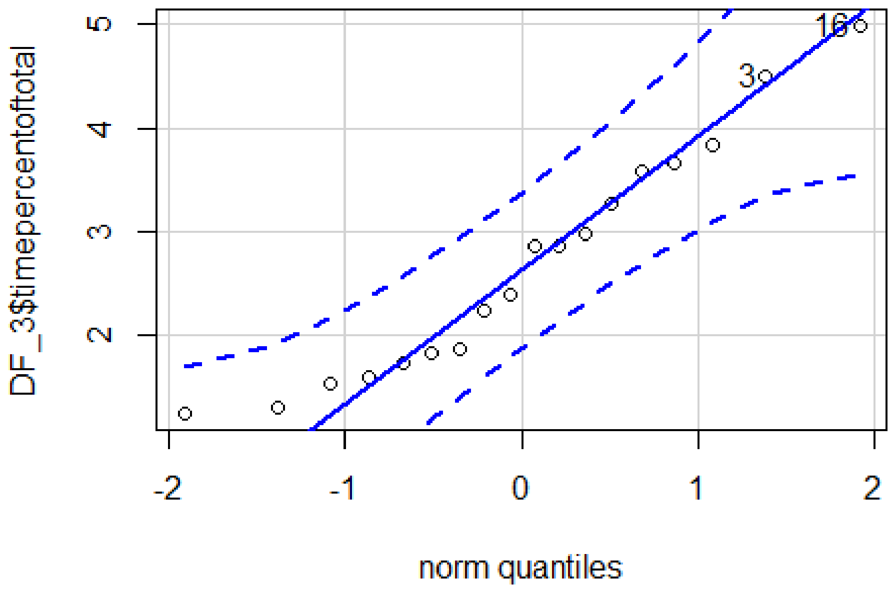
```
DF_3 %>%
  filter(Type == "Dynamic AOI") %>%
  ggplot(aes(x=Setup, y=timepercentoftotal, fill=Label)) +
  facet_wrap(~Participant)+
  theme(axis.text.x=element_text(angle=50,vjust = 0.6,size = 10)) +
  geom_bar(stat="identity", position=position_dodge())
```



Group differences

Underneath is calculated if there was a significant difference between the Setups on percentage of time spent in Mirrors. At first a normality test should be performed to see if the data is normally distributed. At first the qqPlot can be used. If all the point fall approximately along the reference line we assume that the data is normally distributed. Then, a Shapiro-Wilk normality test should be performed. If the p-value obtained after performing the test is higher than 0,05, this will imply that the distribution of the data is not significantly different from a normal distribution and normality can be assumed. In this case, an one-way anova test can be performed to see if there are significant differences between the groups. Else, use a non-parametric test like the Kruskal-Wallis test.

```
res1.aov <- aov(timepercentoftotal ~ Setup, data = DF_3)
summary(res1.aov)

##             Df Sum Sq Mean Sq F value Pr(>F)
## Setup        2  3.489   1.745   1.466  0.262
## Residuals   15 17.845   1.190

shapiro.test(DF_3$timepercentoftotal)

##
##  Shapiro-Wilk normality test
##
## data:  DF_3$timepercentoftotal
## W = 0.94048, p-value = 0.2954

qqPlot(DF_3$timepercentoftotal)
```




```
## [1] 16  3
plot(res1.aov, 1)
```