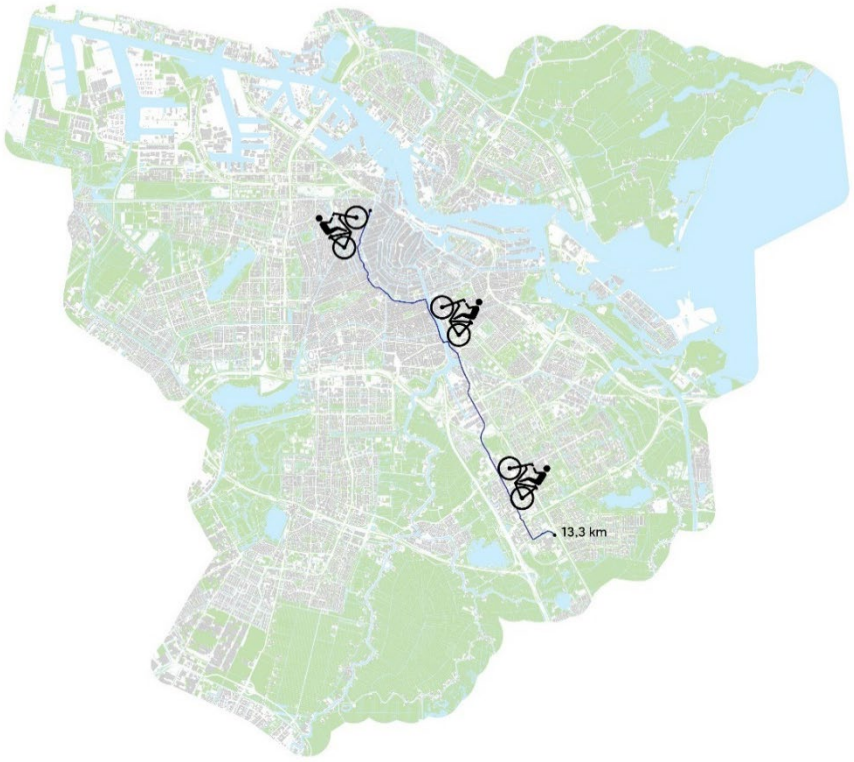
MSc Thesis

Cycling Distance and the Built Environment:

An investigation to what extend cycling distance is influenced by the built environment of Amsterdam and surroundings



Written by:

S.J.A. (Simon) de Haas

Master of Science

Metropolitan Analysis Design and Engineering

At the Wageningen University,

December 2020.

Supervised by:

dr. C. (Kees) Maat & dr.ir. D. (Danique) Ton

WAGENINGEN
UNIVERSITY & RESEARCH

TU Delft

AMS
AMSTERDAM INSTITUTE FOR
ADVANCED METROPOLITAN SOLUTIONS

Thesis title:	Cycling Distance and the Built Environment
Subtitle:	An investigation to what extend cycling distance is influenced by the built environment of Amsterdam and surroundings
Course:	MSc Thesis Metropolitan Analysis, Design and Engineering
Course code:	YMS-80330
Credits:	30 ECTS
Study programme:	MSc Metropolitan Analysis, Design and Engineering
Institution:	Wageningen University & Research Delft University of Technology Amsterdam Institute for Advanced Metropolitan Solutions
Place:	Beusichem
Date:	09/12/2020
Student:	S.J.A. (Simon) de Haas
Student number:	-
First examiner:	dr. C. (Kees) Maat
Second examiner:	dr.ir. D. (Danique) Ton
Third examiner:	dr.ir. K.B.M. (Karin) Peters

Preface

Dear reader,

The objective of this master thesis is building an understanding of the relationship between cycling distance and built environment. I have written this thesis report as part of my master's programme Metropolitan Analysis, Design and Engineering, which is a joint degree program of Wageningen University and University of Technology Delft, facilitated by the Amsterdam Institute for Advanced Metropolitan Solutions in Amsterdam. I have selected the topic based on my interest in mobility flows, built environment and data science, an area relatively new for me. I am delighted I was able to complete this thesis, and therefore my master's programme. I would like to express my gratitude to the people that have supported me and believed in me.

First, I would like to thank my supervisors Kees Maat and Danique Ton. Their expertise in the topic, professional supervision, and constructive feedback helped me to write this thesis report. Writing this thesis and conducting a data analysis was relatively new for me. Without their guidance, I would not have accomplished as much as I have done now. I have enjoyed the discussions that gave me a much better understanding of the topic, even though almost all our meetings had to be done online, due to restrictions related to the COVID-19 pandemic. Secondly, I want to special thank Berjan Mensink and Jan-Lieuwe de Vries, who were always on standby to help and support during the executing.

Last but not least, I want to thank my family and friends that have supported me from the beginning. They have showed interest in my progression and they have supported me throughout the entire process: from inception to completion of this thesis. Wim and Berjan for proofreading this thesis and my partner Esther, for always being supportive and proud during the ups and downs I have faced during this period. Thank you very much!

Simon de Haas
Beusichem, December 2020

Abstract

In order to stop climate change, it is inevitable to transition to more environmental-friendly ways of transportation, such as cycling. An important criterion for selecting the bicycle as a modality is the distance to the destination. An increase in travel distance coincides with a decrease in modal share of bicycles compared to alternative modalities. Literature does not explain why cycling is preferred for certain distances and not for others, except for some practical reasons.

This thesis investigates the built environment in relation to cycling distance. Literature indicates the built environment as being important for destination choice, mode choice and route choice. Moreover, literature also indicates a relationship between cycling distance and these decision choices. As urbanization of the built environment increases, more destinations are within reach and this tends to reduce travel distance. At the same time, network friction reduces due to increasing network density. Finally, preferences to cycle through or avoid certain characteristics of the built environment may result in detours, which increase travel distance.

People's decisions on travel destination, transportation mode and route appear to be crucial for choosing the bicycle as mode of transport. Humans make these decisions simultaneously, but in this thesis, it is assumed that it is done in the order: destination, mode, and route. By making this assumption, it is possible to say that (a) destination choice is connected to Euclidean distance, (b) mode choice is connected to network friction, and (c) route choice is connected to behavioural detour.

The 7D's framework of Ewing & Cevero (2010) has been used to quantify the built environment. It will turn out that only the first three D's (Design, Density and Diversity) are influencing cycling distance. The choices for destination, mode and route choice are also influenced by the following elements: network density, cycle paths, green environments, smooth surface, built environment density and the mixture of functions in the built environment.

This research has been conducted with bicycle trip data within the municipalities of Amsterdam, Amstelveen, Diemen and Ouder-Amstel. Those municipalities offer a fitting case study, since the 'Greater Amsterdam' has a varied built environment and has a need to reduce car traffic significantly. In this thesis, the relation of the elements with cycling distance are statistically tested with a multiple linear regression analysis. In order to do so, elements have to been quantified and combined with the bicycle trips. Explanatory models are developed to explain cycling distance, Euclidean distance, and detour distance with the elements of the built environment.

The models constructed in this thesis show that the elements green environments, waterbodies, smooth surface material, and cycle paths have a positive increasing impact on cycling distance, while network density and built environment have a negative decreasing impact. No clear impact on travel distance is found for the element mixture of functions. This thesis also show that the built environment is more important for determining the Euclidean distance than its importance detour distance.

The results from this thesis confirm the relationship between built environment and cycling distance. To be specific, the results show that the built environment density is most important for Euclidean distance and therefore destination choice. Green environments and waterbodies are incentive elements of the built environment to overcome the distance added by the network friction. The two elements can also affect the cycling distance due to the choice of behavioural detour. This shows that bicycle use for longer distances can be stimulated by adding more green and potential waterbodies in the built environment.

Glossary

Term	Description
7D's	7D's are referring to the framework of Ewing & Cevero (2010) were built environment is categorised as Design, Density, Diversity, Distance to transit, Destination accessibility, Demand management and Demographics.
BAG	Basisregistratie Adressen en Gebouwen, a dataset including addresses, buildings and building function.
BCW	Bicycle Counting Week, a dataset including bicycle trips
Behavioural detour	Behavioural detour is that part of a cycling distance that makes the trip longer than the shortest path.
BGT	Basisregistratie Grootschalige Topografie, a dataset including all kind of objects in the built environment
Built environment	Built environment refers to the environment created for humans for human activities by humans. In the Netherlands, almost no space has been left untouched.
Cycling distance	The distance between origin and destination that has been travelled by bike. Cycling distance consists of the Euclidean distance, network friction and behavioural detour.
Destination choice	Destination choice is choosing one destination among several alternatives.
Detour	Detour is a different or less direct route to a destination that is used to avoid a problem, to make a visit or perform an activity on the way (Cambridge University, n.d.).
Detour distance	The cycling distance minus the Euclidean distance. The distance that includes network friction and behavioural detour.
Euclidean distance	The 'crow flies' distance or celestial distance
Link	The polyline between two intersections of the bicycle counting week data.
Mode choice	The consideration between different transport modes.
Network friction	The friction of the infrastructure or network in the built environment that makes it less possible to travel like the crow flies.
Route choice	Route choice is the decision between different routes to reach the destination.
Segment length	The length of the trip segment.
Trip	Your journey from origin to destination (Cambridge University, n.d.)
Trip segment	One link that is used specific for a trip. More trip segments form the trip

Contents

Preface 3
Abstract 4
Glossary 5
List of figures 8
List of tables 8
1 Introduction 9
1.1 Motivation 9
1.2 Case Study 11
1.3 Research Objectives and Questions 11
1.4 Thesis Outline 12
2 Theoretical Background 13
2.1 Cycling Distance 13
2.2 Destination, Mode and Route Choices 13
2.3 7D's of the Built Environment 14
2.3.1 Design 15
2.3.2 Density 16
2.3.3 Diversity 16
2.3.4 Distance to Transit 16
2.3.5 Destination Accessibility 16
2.3.6 Demand Management 16
2.3.7 Demographics 17
2.4 Elements Related to Cycling Distance 17
2.5 Conceptual Framework 18
2.6 Hypotheses 18
3 Methodology 20
3.1 Analysis Method: Multiple Linear Regression 20
3.2 Data Collection 20
3.2.1 Bicycle Trip Data 20
3.2.2 Design Elements of the Built Environment 21
3.2.3 Built Environment Density and Mixture 21
3.3 Data Operationalisation 21
3.3.1 Operationalisation 21
3.4 Data Processing 23
3.5 Variable Creation 24
3.5.1 Cycling Distance 24

3.5.2 Percentage along Trees 25
3.5.3 Percentage along Water 25
3.5.4 Percentage along Vegetation 26
3.5.5 Percentage over Smooth Surface Material 26
3.5.6 Percentage over Cycle Paths 27
3.5.7 Network Density 27
3.5.8 Average Built Environment Density 27
3.5.9 Average Mixture of Functions 28
3.5.10 Euclidean Distance 29
3.5.11 Detour Distance 29
3.5.12 Summary of Created Variables 29
4 Results 30
4.1 Descriptive Statistics 30
4.1.1 Trip Visualisation 32
4.1.2 Relation between Cycling Distance Euclidean Distance and Detour Distance 32
4.1.3 Correlation of the Variables 33
4.2 Modelling 35
4.2.1 Results Interpretation and Hypotheses Testing 37
5 Conclusion & Discussion 42
5.1 Conclusion 42
5.2 Discussion 43
5.3 Recommendations 45
5.3.1 Research Recommendations 45
5.3.1 Policy Recommendations 46
6 References 47
7 Annexes 49

List of figures

Figure 1.1 Modal split in relation to distance (CBS, 2018) 9
Figure 2.1 Visualisation of the Euclidean distance, network friction and behavioural detour 13
Figure 2.2 The assumed order of choices for this research 14
Figure 2.3 Conceptual model for researching cycling distance 18
Figure 3.1 Data Action Model 23
Figure 3.2 Borders of top 5 municipalities with trips to Amsterdam (CBS, 2016) 24
Figure 3.3 Location of trees within the research area 25
Figure 3.4 Visualisation of how geo information of trees is quantified. 25
Figure 3.5 Location of water within the research area 25
Figure 3.6 Location of vegetation within the research area 26
Figure 3.7 Visualisation how information of the surface material is add to the trip segment 27
Figure 3.8 Visualisation how functions have been counted around trip segments 28
Figure 4.1 Distribution of cycling distance in the dataset categorised by origin municipality. 31
Figure 4.2 Visualisation of an average trip in the dataset 32
Figure 4.3 Euclidean distance and detour distance in relation with cycling distance 33
Figure 4.4 Correlation matrix of the variables in the dataset 34

List of tables

Table 3.1 Table explaining measurement criteria for the different hypothesis 22
Table 3.2 Number of trips including categorised in origin and destination municipalities 24
Table 3.3 Summary of the variables that have been created for the analysis 29
Table 4.1 Cross table showing the number of trips available in the dataset 30
Table 4.2 Descriptive statistics of the dataset 31
Table 4.3 Information about the variables of the average trip in Figure 4.5 32
Table 4.4 Model one: cycling distance - built environment 35
Table 4.5 Model two: detour distance - built environment 36
Table 4.6 model three: Euclidean distance - built environment 36
Table 4.7 Model four: Cycling distance - built environment + Euclidean distance 37

1 Introduction

Policymakers in several countries, including the Netherlands, are showing an increasing interest in cycling. For example, Dutch government wants to decrease the number of kilometres driven by cars by eight billion kilometres by 2030 to decrease emissions. They encourage and invest in the use of the bicycles (Ministry of Economic Affairs and Climate Policy, 2019). Governmental institutions, ranging from municipalities to national governments, are exploring all opportunities to increase bicycle transportation and to reduce motorized traffic and its resulting air pollution (Strauss et al., 2015). Other reasons for encouraging cycling as a means of transport is the potential benefit for individual health, it could reduce chronic diseases (Roever et al., 2019; Ton et al., 2019), and it is a cheaper way of transportation compared to car usage and public transport (Harms & Kansen, 2017; Heinen et al., 2010). Besides policymakers, researchers are showing an increasing interest in bicycle research (Heinen et al., 2010). The need to switch to more environmental-friendly ways of transportation and increased interest in the topic have led to more research into travel behaviour of cyclist and the relation to wider scale of influencing factors.

1.1 Motivation

The Netherlands is a cycling rich country. Inhabitants together own around 22 million bicycles on 17 million inhabitants. On average a Dutch person travels 10,000 kilometres a year, where nearly 900 kilometres (nine percent) is done by bicycle(Ministry of Infrastructure and Water Management, 2019). In comparison, in Denmark the average travel distance per person is 14,200 kilometres, where 650 kilometres (almost five percent) is done by bicycle (Ministry of Transport Denmark, 2012). Additionally, 25% of the trips made in the Netherlands is done by bike. In comparison, in the United States and Australia this percentage is less dan one. (Harms & Kansen, 2017).

These high numbers can be attributed to a geographical advantage, a high-quality bicycle infrastructure and increased traffic safety for cyclists (Schepers et al., 2017). Nevertheless, many ‘short-distance’ trips in the Netherlands are still made by car. From all car trips that are made in the Netherlands 50% are less than 7.5 kilometres and about 40% of the car trips are less than 5 kilometres, see Figure 1.1 (CBS, 2018). Distances less than 7.5 kilometres are excellent distances to cycle, according to State Secretary Van Veldhoven (NOS, 2018). Therefore, there is still potential to increase the cycling share of total travelled kilometres per year and decrease the kilometres travelled by car.

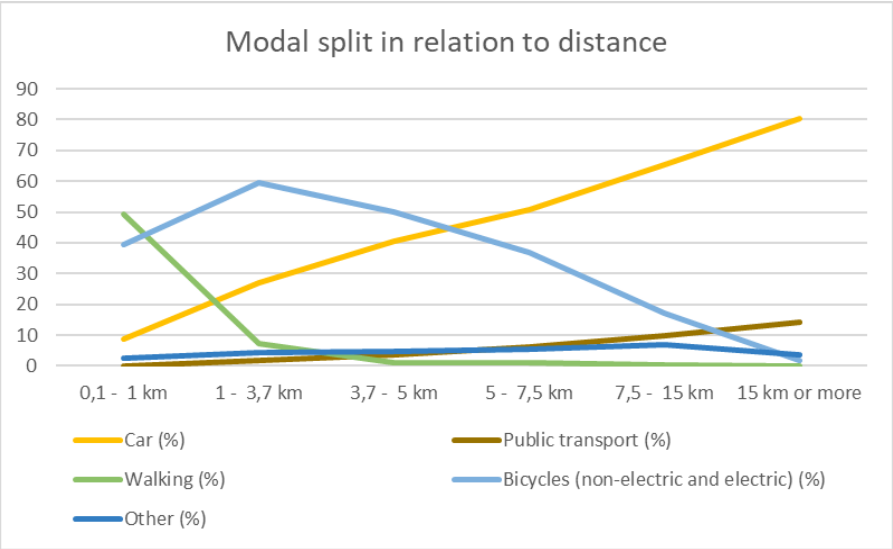


Figure 1.1 Modal split in relation to distance (CBS, 2018)

While most bicycle kilometres are travelled for leisure and the commuting purposes in the Netherlands (Ministry of Infrastructure and Water Management, 2019), some people are selecting other modes of transport above cycling. Reasons for not choosing to cycle are the difficulty of carrying loads, weather conditions, the slower speed of cycling outside urban areas, compared to motorized traffic, the lack of facilities at the destination and the physical effort. An increase in distance is directly an increase in travel time and physical effort (Heinen et al., 2010). This explains why bicycle usage decreases with longer distances in graph 1.1 above. The growth in the share of E-bikes from 12% in 2013 to 18% in 2017, may change the limitation to short distances, since longer distances become within reach (Ministry of Infrastructure and Water Management, 2019). The combination of E-bikes with cycle highways can take the cycling policy of cities to a new level and encourage a larger part of the population to change their commuting habits. (INTREG, n.d.).

This study has investigated the cycling distance, as it plays an important role in a person's decision to cycle to a certain destination or not. Studies show that distance is a determinant factor for destination selection, mode, and route choice (Heinen et al., 2010; Pritchard et al., 2019; Ton et al., 2018). However, hardly any reasons are given - except for some influencing factors - why certain distances are cycled while others are not. One important characteristic found for destination selection, mode and route choice is 'built environment'. Ewing & Cevero (2010) show that 'built environment' affects travel time and distance. They developed a way to measure the impact of 'built environment' in 5D's: Design, Density, Diversity, Distance to transit, Destination accessibility. Other researchers also add Demand management and Demographics as additional D's to measure the built environment (Ewing & Cervero, 2010). The relationship between built environment and each of those choices has been investigated, but what the relationship between built environment and cycling distance is not explained.

It is therefore important to get a better understanding of cycling distance in relation to 'built environment', since understanding the effects of 'built environment' on cycling behaviour can lead to policy designs that incorporates stimulating elements into the built environment. This could make the built environment more appealing for bicycle usage for greater distances and stimulate people to switch from car to bicycle for distances less than 7.5 kilometres. Many variables of the built environment related to cycling and the three choices have been identified, but no explanation has been given yet as to how those variables correlate to cycling distance.

The objective of this thesis is to solve this problem and close the gap by researching the significance of elements of the built environment by analysing observed bicycle travel data. This thesis is built upon findings from the fields of destination, mode and route choice and it uses the framework of Ewing and Cevero (2010) to quantify elements of the built environment, so they can be examined in relation to cycling distance.

Travel distance widely researched and plays a role in destination, mode and route choice (Heinen et al., 2010; Pritchard et al., 2019; Ton et al., 2018). The distance between origin and destination is essentially the 'crow flies' distance, also known as the Euclidean distance. However, the actual travel distance is generally longer. This thesis distinguishes two travel distance-increasing components:

1. Friction caused by travelling via the road network; this network friction adds additional distance.
2. Behavioural detours, where travellers often decide not to take the shortest route but make detours.

In this research, it is assumed that the built environment affects the three components of the distance travelled. As the built environment becomes more urban, more destinations are within reach and this

tends to reduce travel distances. At the same time, network friction reduces because network density increases with urbanisation. The relationship between behavioural detour and the built environment is more complex (see next chapter). The built environment can be quantified by using the framework of Ewing & Cervero (2010). After quantification, it is possible to analyse the built environment with cycling trips, creating an understanding of the built environment in relation to cycling distance.

1.2 Case Study

The Netherlands is widely known as the bicycle country of the world. People use the bicycle for many purposes inside and outside urban areas. In Amsterdam, the capital of the Netherlands, there are many modes used to travel to destinations both inside and outside the city border. The city is placed in the top five of best bicycle cities in the world (Coya AG, 2019). This makes Amsterdam and its suburbs ideal for finding out how the built environment influences cycling distance.

Amsterdam is an old city designed in the 16th century. The available modes of that age determined the design. Nowadays, many cars go through the streets of Amsterdam, but the street design is not suitable for this mode of transportation. As a result, many conjunctions and parking problems occur within the city centre. Moreover, there is no available space left to improve the infrastructure for cars. The most obvious way out is stimulating the usage of bicycles.

1.3 Research Objectives and Questions

The main objective of this thesis is to analyse the impact of the built environment on cycling distance. This will help to get a better understanding of why people cycle certain distances. The city of Amsterdam is used as case study. Knowing the associated elements of the built environment will help local policy makers to invest in better cycling facilities. This will be investigated using the following research question (RQ):

"To what extent does the built environment explain cycling distances within the municipality of Amsterdam?"

It is assumed that elements of the built environment influence cycling distance. However, there are several other factors influencing cycling distance. To understand what the relationship between built environment and cycling distance is, it is necessary to understand those other impacting factors. To be able to answer the main research question of this thesis, the following sub questions (SQ) are raised:

SQ1. "Which elements of the built environment are associated to cycling distance and how can those elements be measured according to the literature?"

SQ2. "To what extent does the built environment explain the total cycling distance?"

SQ3. "To what extent does the built environment explain the detour distance?"

SQ4. "To what extent does the built environment explain the Euclidean distance?"

SQ5. "To what extent does the Euclidean distance contribute to the cycling distance?"

The first sub-question will be answered by reviewing the academic literature on this subject. The first sub-question helps in the search for specific elements of the built environment that are related to the cycling distance, and it helps in finding ways to operationalise the elements for analysis. The remaining four sub-questions help to analyse the built environment in relation to the cycling distance. A

quantitative methodology is used to operationalise the research and used to test the developed hypotheses using existing cyclist travel data.

1.4 Thesis Outline

This thesis consists of five chapters. Chapter 2 provides a theoretical background for the following notions: (a) distance, (b) the decisions made to go cycling, and (c) which elements of the built environment have a connection to cycling distance. A conceptual model on which the methodology is based follows. Chapter 3 describes the research method and explains the implementation and quantification of the elements of the built environment. Chapter 4 describes the data and presents the results. Finally, chapter 5 provides the conclusions and recommendations.

2 Theoretical Background

Section 2.1 explains the first theoretical notion of 'cycling distance'. In section 2.2, the notions of 'destination', 'mode', and 'route choice' are introduced and there is explained how intertwined they are. Section 2.3 explains how the built environment can be measured with the 7D's. In addition, it discusses which elements are available in the literature about destination, mode and route choice.

2.1 Cycling Distance

The cycling distance is the distance someone needs to travel by bicycle from origin to destination. This distance includes the Euclidean distance, network friction and behavioural detour. Euclidean distance is the distance as the crow flies and is the simplest to calculate. The built environment prevents people to go in a straight line; instead, people use the infrastructure of the bicycle network.

The bicycle network adds to the complexity to cycle to your destination, this is called network friction. The network friction can differ for each type of mode. For example, in one-way streets cars can go in one direction only, while cyclists can go in both directions. The Euclidean distance together with the network friction is the shortest route.

Finally, people can cycle a route deviating from the shortest path. They may not be aware of the shortest path or are taking consciously another route to avoid places or just to cycle through pleasant areas in the built environment. This is called behavioural detour. The combination network friction and behavioural detour is also referred as the detour distance. Normally the route choice is included in the Euclidean distance and network friction, as people take into account their knowledge of the different routes when choosing the destination and mode. The knowledge component makes the concept of route choice more complex to understand. Someone's preference to cycle a certain distance through the built environment is what this research is about. Figure 2.1 shows the three types of distances, line A is the Euclidean distance, Line B is the Euclidean distance and network friction (shortest path) together and line C depicts the combination of Euclidean distance, network friction and behavioural detour.

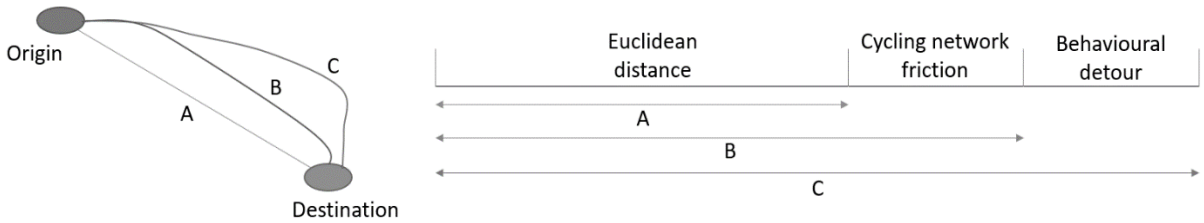


Figure 2.1 Visualisation of the Euclidean distance, network friction and behavioural detour

2.2 Destination, Mode and Route Choices

In the state-of-the-art destination choice (Clifton, Singleton, Muhs, & Schneider, 2016; Kitamura, Yoshii, & Yamamoto, 2009), mode choice (Olde Kalter, Geurs, & Hoogendoorn-Lanser, 2015; Manville, 2017; Ton, Duives, Cats, Hoogendoorn-Lanser, & Hoogendoorn, 2019; Whalen, Páez, & Carrasco, 2013) and route choice (Ghanayim & Bekhor, 2018; Prato, Halldórsdóttir, & Nielsen, 2018; Pritchard, Frøyen, & Snizek, 2019; Ton, Cats, Duives, & Hoogendoorn, 2017; Ton et al., 2018) have been widely investigated. Each study mentions that distance and built environment are factors in these choices. The facts that both are a factor in every decision and that decisions are taken simultaneously, makes it difficult to separately discuss their influence on the cycling distance. In this research, is assumed that choices are made in the order of destination choice, mode choice and route choice, see Figure 2.2.

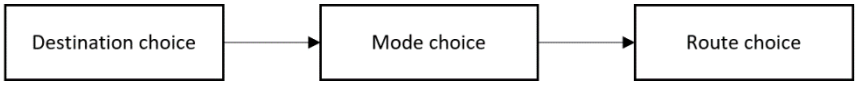


Figure 2.2 The assumed order of choices for this research

Destination choice in its simplest form is choosing one destination from several alternatives. The distance to each of these destinations is a factor in making this decision. This assumption is based on the fact that the starting location is determined for a longer period of time, such as home. Destinations that have alternatives to choose from are so-called flexible destinations (e.g. shopping, leisure, sports). These destinations are spread throughout the built environment. For these destinations, travel distance is decisive. For fixed destinations, such as work and education, the location in the built environment is already fixed and the travel distance cannot change this (Chowdhury, 2017). If people choose such a destination, the Euclidean distance of the journey does not change, regardless of the mode of transport and the choice of route.

The decision to choose a certain mode of transport (e.g. walking, cycling, public transport, car) is referred to as mode choice in the literature. The mode determines the network that can be used and this influences the distance through network friction. For example, there are paths through rural areas for cycling and walking which are not accessible by car. This means that the shortest path for cyclists will be different for car drivers. The notion of shortest path is important as an increase in distance for active modes also increases journey time and efforts. (Heinen et al., 2010).

The last component that influences cycling distance is route choice. Normally, the available networks and the available routes influence the destination choice and the mode choice, but for better understanding, these have been defined already. In this way, the route choice is only influenced by the behavioural detour. Pritchard (2019) has found out that cyclist most often do not take the shortest path. Instead they take detours that make the cycling distance approximately 1.2 times longer (Pritchard et al., 2019). The built environment might not be appealing enough for cycling, or an alternative (longer) route in the built environment is more pleasant to cycle through. Both are behavioural detours. There is also the possibility that the cyclist is not aware of the shortest path, as most daily made trips are not checked by the cyclist for alternatives (Rosman, 2015).

Distance and built environment are influencing the destination choice, mode choice and route choice and these decisions are again affecting the distance. The built environment is the context in which people make the choices affecting the cycling distance. In this research, it is assumed that the choices are made in the following order: destination, mode, and route choice. This helps to get a better understanding of the relationship between built environment and cycling distance.

2.3 7D's of the Built Environment

The previous section explained how the built environment is influencing cycling distance. This section will break down the built environment into elements that can be analysed in relation to cycling distance. The literature divides elements of the built environment into positive and negative elements that influence cycling. The framework Ewing & Cervero (2010) categorises the elements into the 7D's: Design, Density, Diversity, Distance to transit, Destination accessibility, Demand management and Demographics. This categorisation helps to make the elements measurable.

Each of the following subsections starts with an explanation of what the D entails, how it measures and quantifies elements in the category, and how it affects cycling distance. Subsequently, the elements relating to destination, mode and route choice in relation to cycling are discussed.

2.3.1 Design

Design of the built environment includes street network characteristics within an area. These can vary from high-density urban grid networks like straight streets and blocks to curving streets forming loops. Design can be measured by the number of four-way crossings, the number of intersections per square kilometre, sidewalk coverage, average building setback, block size, average street widths, number of pedestrian crossings, presence of street trees, or other physical variables characterising the environment (Ewing & Cervero, 2010). The design defines the network layout and characterizes the urban form. Design mainly influences network friction and behavioural detour.

Within urban designs, a dense and continuous infrastructure is positively associated with active modes (Ton et al., 2017). A denser infrastructure decreases the network friction and results in shorter cycling distances. The disadvantage is that the number of intersections and turns usually increase. These are negatively associated to cycling route choice (Prato et al., 2018; Ton et al., 2017). This means that a denser network reduces friction in the network, while the discontinuous network increases the cycling distance because of behavioural detour.

Cyclist avoid uphill slopes and this also increases behavioural detour (Prato et al., 2018). Uphill slopes are less common in relatively flat countries like the Netherlands, but the preference of cyclist to have underpasses instead of bridges supports this tendency. For example, the municipality of Leeuwarden prefers to build cycling tunnels instead of bridges, because the speed built up when entering the tunnel can be used again when leaving it (Municipality of Leeuwarden, 2013). While bridges and tunnels are decreasing the network friction, bridges and tunnels can be the reason for a detour, increasing the cycling distance.

Another design element that is network related is the presence of (separate) cycle paths. Pritchard (2019) found out that cycling separated from other traffic has a positive influence on the cyclist route choice (Pritchard et al., 2019), meaning people would make a detour even if it increases the cycling distance. Dutch research has shown that separate cycle paths have no significant influence on route choice and mode, meaning separated cycle paths would not increase cycling distance. The reason why it is not significant in the Netherlands is that the Dutch guidelines for infrastructure as well as the way of dealing with cyclists are well established. Therefore it is less of a serious problem if cyclists join the same lane as other (motorised) traffic if the speed is maximum 50 km/h. Above this speed limit, separated cycle paths are included in the street design (Ton et al., 2017).

Surface quality is influencing cycling route choice (Hölzel et al., 2012; Pritchard et al., 2019). Research has shown that asphalt is most preferable for cycling. Asphalt is highly comfortable for cycling and it has a low rolling resistance. Concrete is the next preferred surface that has many benefits for maintenance, but concrete is not as comfortable as asphalt. The least preferred surfaces are self-binding gravel and cobblestone (Hölzel et al., 2012). This means that a smooth surface quality is positive for reaching further destinations by bike and consequently longer cycling distances. Moreover, it also means that people are making detours to avoid bad quality surfaces.

Literature refers to a final design element and this element is ‘aesthetics’ related. Park and street landscaping are positively associated to cycling (Fraser & Lock, 2011; Heinen et al., 2010), but no direct relation to cycling distance has been found. Parks and street plantations make it more attractive to choose a bicycle. Therefore, in view of network friction, it makes it pleasant to cycle a certain distance. A side effect could be that cyclists prefer a detour, because a greener street is more appealing than the shortest route.

2.3.2 Density

Density of the built environment may be the compactness of a variable within a defined area. Density is measurable as the variable of interest per unit of an area. This could be the population, dwelling units, employment, building floor area or other density related variables. Population and employment density can also be measured in activity density per areal unit (Ewing & Cervero, 2010).

The cycling distance is influenced by density since it determines how far destinations are located from each other (Ton et al., 2019). Density makes cycling distances shorter since the Euclidean distance decreases in higher density areas. Route choice impacts distance as well. Cyclists prefer routes through low dense areas and sport/scenic areas (Prato et al., 2018). A result of this preference is that cycling distance increases because of the detour cyclists are making.

2.3.3 Diversity

Diversity in the built environment is the number of different land uses or activities carried out within an area. Diversity is measurable by the degree of land use/activity per land area, floor area or employment. More diversity is seen as a higher value. Higher diversity can lead to closer flexible destinations, resulting in shorter cycling distances (Ewing & Cervero, 2010).

The presence of shops and higher mixed land use environment encourages active mode use, whereas low residential mixtures discourages active mode use (Heinen et al., 2010; Ton et al., 2019). Destinations with a higher mix of functions may be more cycle-able as the distance is shorter than in areas with a low mix. One could therefore assume that a higher mix of functions reduces cycling distance.

2.3.4 Distance to Transit

Distance to Transit is the distance from a location or area to the nearest public transport station/stop. It is commonly measured as the shortest route to the nearest train station, metro station or tram stop or bus stop. Alternatively it is measurable as the distance between stops or the number of stops within an area (Ewing & Cervero, 2010). The use of public transportation is positively associated with cycling as from origin to the station and station to final destination involves usage of active modes most of the time (Handy et al., 2014; Heinen et al., 2010; Rissel et al., 2012). Although distance to transit is positively associated to cycling, it does not affect cycling distance, since cycling is only a part of the whole trip. Distance to transit does not change Euclidean distance, network friction and behavioural detour either.

2.3.5 Destination Accessibility

Destination accessibility in the built environment is the ease of access to a location or area. It is regionally measured, usually through the amount of jobs available within certain travel time boundaries, or travel time to business areas (Ewing & Cervero, 2010). Handy (1993) defined it locally as the distance to the nearest shop. The nearest shop is already included in the definitions of density and diversity of the built environment, since a higher density and diversity shorten the distance to destinations, such as shops. There do not seem to be other relevant elements, as destinations that are not accessible by bicycle are outside the scope of this study.

2.3.6 Demand Management

Demand management in the built environment is managing travel activities by physical facilities or rules. Demand management is measurable by, for example, the amount of parking space or height of the parking costs (Ewing & Cervero, 2010). Demand management influences the mode choice for cycling positively by making bicycle parking spots available (Heinen et al., 2010), or by facilitating showers and lockers at work (Ton et al., 2019). This makes a destination more suitable for cycling but

does not change the cycling distance as Euclidean distance, network friction, and behavioural detour stay the same.

2.3.7 Demographics

Demographics is seen as the seventh D, and in travel studies considered as an influencing factor (Ewing & Cervero, 2010). Within travel studies, users are asked to record the demographics of their trips. In the built environment, demographics can also be used as descriptive for the kind of people living in an area. Demographics of an individual may affect cycling distance as someone's fitness level may influence the ability to cycle certain distances. Household characteristics that may influence cycling are the number of children and income of a household (Handy et al., 2014; Heinen et al., 2010; Mitra, 2013; Muñoz et al., 2016). These variables are person-specific and cannot be generalised for the built environment. Other elements related to cycling distance are not found for this category.

2.4 Elements Related to Cycling Distance

To understand how cycling distance is interwoven with destination choice, mode choice and route choice, the distance is divided into three components: Euclidean distance, network friction and behavioural detour. In reality the decision for destination, mode and route are made in parallel. For research purposes, it is assumed that the choices are made in the following order: destination choice, mode choice, route choice, see Figure 2.2.

If the destination, influenced by the built environment, has been chosen, the Euclidean distance is set and will not change. Next in line is the decision for a mode, again influenced by built environment. When the decision is made, both the network (shortest path from origin to destination) and the network friction distance are known, see Figure 2.1 line B. The last decision is route choice, where preferences to cycle through certain environment may result in a detour, line C in Figure 2.1, that increases the cycling distance.

With this basic understanding of cycling distance, it is possible to finalise this review of the literature on destination, mode and route choice for cycling. Elements have been categorised in the framework of Ewing & Cevero (2010), to make them measurable and see how they affect cycling distance. The conclusion is that only the first three D's: Design, Density and Diversity of the built environment are influencing cycling distance:

1. *Design*:
    - Density network layout
    - Cycle paths
    - Green environment
    - Surface quality
2. *Density:*
    - Built environment density
3. *Diversity*:
    - Mixture of functions

For the other D's of the built environment no influencing elements on travel distance are found.

4. *Distance to transit*: the cycle trip is part of a larger trip to a destination. Only the Euclidean distance is different, but the cycling distance is not.
5. *Destination accessibility*: this is locally measured as distance to closest store and as such already captured by the built environment density and mixture of functions.

6. *Demand management*: this affects the choice to cycle or not, but is does not change the Euclidean distance, network friction and behavioural detour, and therefore does not impact the cycling distance.
7. *Demographics*: this may influence an individual's activity level, but this element cannot be generalized for groups or an entire population.

2.5 Conceptual Framework

From the reviewed literature the following conceptual model could be developed, see Figure 2.3. The model gives a simplified representation of reality. In reality, people make decisions on destination, mode and route in parallel. But this makes it harder to understand and visualise the relationship between built environment and cycling distance. Therefore, this model assumes the following order: destination, mode and route. Figure 2.3 shows a diagram about the relationship between built environment and cycling distance is defined.

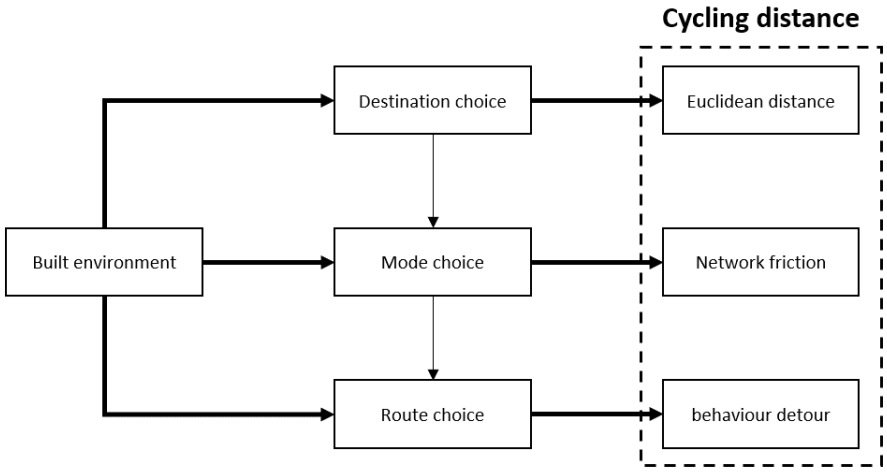


Figure 2.3 Conceptual model for researching cycling distance

Cycling distance is the result of the Euclidean distance, network friction and behavioural detour, see Figure 2.1. The built environment via destination choice influences the Euclidean distance; the built environment through mode choice influences the network friction, and route choice influences behavioural detour. For this thesis, the destination, mode, and route are already known. This way, the influence of the built environment that has been included in the decisions, can be analysed in relation to the cycling distance.

2.6 Hypotheses

The relationship between built environment and cycling distance outlined in the conceptual model leads to the following general hypothesis:

The design, density, and diversity of the built environment influence cycling distances

The main hypothesis can be subdivided into seven smaller hypotheses based on the elements named in section 2.4:

Design

1. The denser the network, the shorter the cycling distance
2. The higher the use of cycle paths, the longer the cycling distance
3. The greener the environment, the longer the cycling distance

4. The smoother the street surface, the longer the cycling distance
5. The more along waterbodies, the longer the cycling distance (sub-hypothesis of the greener the environment, the longer the cycling distance)[1]

Density

6. The denser the built environment, the shorter the cycling distance

Diversity

7. The larger the mixture of functions, the shorter the cycling distance

[1] *Amsterdam is used as a case study in this thesis and this city has many waterbodies. Since water is part of the green/natural environment and since it stimulates cycling, it is interesting to investigate if there is a relation between cycling distance and the presence of waterbodies.*

3 Methodology

This chapter discusses the methodological design of this thesis. The research is empirical in nature. The hypotheses listed in section 2.6 are tested through a statistical analysis. Between the elements of the built environment and cycling is looked for correlations. There are two types of data needed for this research: observed data and data about the environment. The observed dataset consists of bicycle trip data. The second dataset contains information on the design, boundaries and the usage of the built environment. The datasets are processed and combined into one dataset for the statistical analysis.

The first section outlines that the multiple linear regression method is used to analyse the correlation between the elements of the built environment and cycling distance. The next section describes the available datasets and the required elements in that set. Section 3.3 describes the operationalisation, on how these elements and hypotheses can be tested. The last section describes how the elements are quantified and it introduces the available variables to be used in the multiple linear regression.

3.1 Analysis Method: Multiple Linear Regression

The elements of the built environment in relation to cycling distance will be tested by multiple linear regression. Multiple linear regressions enable us to estimate the relation between a continuous dependent variable and a set of explanatory variables. The equation is:

$$Y = \beta 0 + \beta 1X1 + \beta 2X2 + \cdots \beta pXp$$

Y = Dependent variable
β0 = Constant (known as the Y-intercept)
βpXp = Independent variables

The purpose of the multiple linear regression may vary. Most common are the predictive model and explanatory models. For predictive modelling, the regression is used to develop a model that accurately predicts values of the response variable based on values of the predictors, for the future. For the predictive model, it is important that the model is not overfitting or has a multicollinearity issue. The second one is the explanatory model. This model shows the relationships between the explanatory variables and the dependent variable, taking influences on each other into account. For this model, multicollinearity is less of an issue. This research is looking at how elements coherently influence cycling distance. For this reason, an explanatory model is most appropriate.

3.2 Data Collection

The thesis examines the relation between built environment and cycling distance. There are two types of data necessary: observed data, data that include trips of cyclists and data about the environment. This chapter explores the internet in search of data needed to investigate the relationship between the built environment and cycling distance.

3.2.1 Bicycle Trip Data

The Bicycle Counting Week (BCW), in Dutch known as the “fietstelweek”, is a national examination on cycling in the Netherlands and it gathers data on bicycle trips. Bicycle counting has been executed in 2015 and 2016. The BCW was an initiative of the Fietsersbond, Keypoint, NHTV, Beaumont Communicatie & Management and Mobidot, commissioned by Dutch government bodies. The last count covers the period of 19 to 26 September 2016. Almost 30,000 cyclists participated and they completed 416,376 bicycle trips. Participants had to download an app, especially created for this count, on their mobile phone to keep track of their trips. Afterwards the data are anonymized.

The open data of BCW consist of links, polylines and a table including route ID’s and link numbers. The table provides information on which links are used for a trip that can be identified by the route ID.

Combining these two datasets results in trip segments when joined based on route ID form one trip. There are no demographic variables available and begin and end point deviate 100 to 300 metres from the origin. Finally, it is not possible to see which trips the same person made. The data include speed, starting day + hour, intensity of the link. In short, the following information is available:

- List including route ID's and link numbers
- Link intensity
- Average cycle speed on the link
- Cycle trips (path)
- Average cycle speed
- Date & time of the trip

3.2.2 Design Elements of the Built Environment

The 'Basisregistratie Grootschalige Topografie' (BGT) provides the elements of the built environment. The BGT is a detailed digital map of the Netherlands. On this map you can find all physical objects such as buildings, roads, water, railways and (agricultural) terrains (Kadaster, n.d.-b). Municipalities constantly update the data; thus, the actual data is available. From this map the following information of the built environment is required:

- Roads
- Trees
- Vegetation surfaces
- Waterbodies

3.2.3 Built Environment Density and Mixture

The dataset 'Basisregistratie Adressen en Gebouwen' (BAG) is the best source for extracting the built environment density and mixture of functions. The BAG is a dataset including all buildings of the Netherlands. This dataset has information about the year of construction, usage, and square metres (Kadaster, n.d.-a). There are 11 function categories available:

- Meeting function
- Healthcare
- Office
- Education
- Prison
- Industry
- Sport
- Living
- Accommodation
- Shop
- Other

3.3 Data Operationalisation

This section describes how the hypotheses can be measured, which of the earlier mentioned datasets are required, and which data action model is used to visualise the steps.

3.3.1 Operationalisation

To analyse the built environment in relation to cycling distance, elements of the built environment need to be quantified. For example, which trees along the trip should be taken into account, only the trees within a 15-metre radius, or within a 30-metre radius? This subsection clarifies these decisions. Chapter 2 described how each of the 7D's are measurable and this helps to quantify the elements found in literature. For each hypothesis, it is defined how it will be measured based on the literature review. If no definition is provided measurement is based on availability of data.

Table 3.1 Table explaining measurement criteria for the different hypothesis

Hypothesis	ELEMENT	MEASURED IN	DATASET
Design			
1. The denser the network, the shorter the cycling distance	Dense network layout	Average length of the trip segment	Bicycle Counting Week 2016
2. The higher the use of cycle paths, the longer the cycling distance	cycle path	Percentage of the trip going over cycle paths	Bicycle Counting Week 2016
3. The greener the environment, the longer the cycling distance	Green environment	Percentage of the trip going along trees or vegetation areas	BGT
4. The smoother the street surface, the longer the cycling distance	smooth surface material	Percentage of the trip including asphalt/concrete (closed surface material)	BGT
5. The more along waterbodies, the longer the cycling distance	Waterbodies	Percentage of the trip going along water	BGT
Density			
6. The denser the built environment, the shorter the cycling distance	Density of the built environment	Average built environment density measured in functions per hectare	BAG
Diversity			
7. The larger the mixture of functions, the shorter the cycling distance	Mixture of functions	Average number of unique functions along the trip	BAG

3.4 Data Processing

The dataset combines data of the built environment and trip data. Further in this section, is explain how the variables are generated. Figure 3.1 visualises the data steps taken to combine environment data with the BCW data.

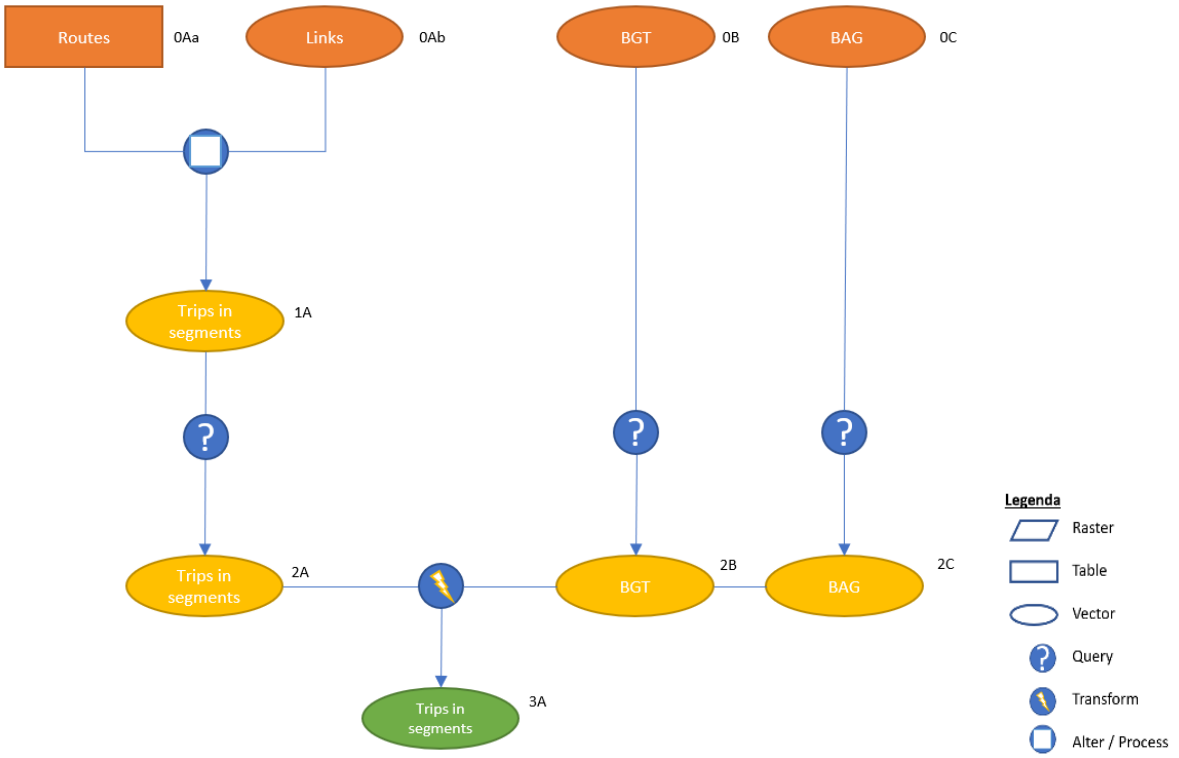


Figure 3.1 Data Action Model

The BCW data is a table with information about the trips (0Aa) and which links have been used. The links are in another dataset (0Ab) and consist of polylines that together form the network. These have been combined to generate trip segments (1A), see Figure 3.1. The joined trip segments, which are based on the route ID, form the trip.

The municipality of Amsterdam is not one region. It comprises a large area around the city centre and a small enclave in the southeast under Diemen, see Figure 3.2. The small area is known as Amsterdam-Southeast. That is why the boundaries of the research area need to be redefined, as some journeys went from Amsterdam-Southeast to Amsterdam, crossing some other municipalities. For each municipality it was counted how many starting points and end points of the journey were within the boundaries of that municipality, see Table 3.2 for the top 5. The table shows that many trips have a start and end in Amsterdam, but a significant number of trips came from other municipalities traveling to or from Amsterdam. Based on these numbers and the geographic location of these municipalities, it has been decided to include the municipalities of Amstelveen, Ouder-Amstel, and Diemen into the research area.

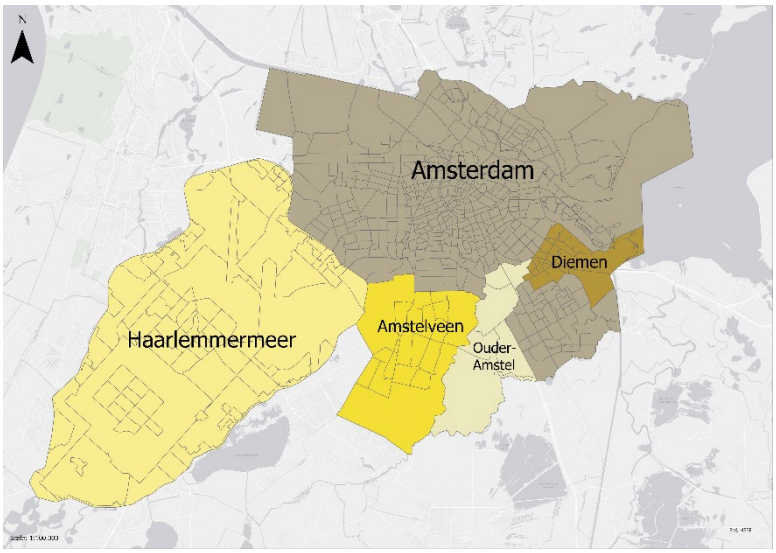


Figure 3.2 Borders of top 5 municipalities with trips to Amsterdam (CBS, 2016)

Table 3.2 Number of trips including categorised in origin and destination municipalities

Origin municipality	Destination municipality	No. of trips
Amsterdam	Amsterdam	27,287
Amsterdam	Amstelveen	721
Amstelveen	Amsterdam	671
Amsterdam	Diemen	317
Diemen	Amsterdam	277
Ouder-Amstel	Amsterdam	264
Amsterdam	Ouder-Amstel	219
Amsterdam	Haarlemmermeer	188
Haarlemmermeer	Amsterdam	165

Based on the research area and the trip segments, the BGT and BAG have been queried to derive the datasets 2A, 2B, and 2C in Figure 3.1. Finally, the datasets have been joined into one dataset.

The next section explains how the variables have been combined with the trip segments and how the final variables have been generated for the multiple linear regression analysis. All taken steps to create the trip segments and how data of the BGT and BAG are combined with the trip segments can be found in detail in Annex A – Data Action Steps ArcGIS Pro.

3.5 Variable Creation

This section explains how the variables have been created from the geo information and how it is transformed into a variable that can be used in multiple linear regression model. One part is conducted in ArcGIS Pro and the other part in Python. Detailed steps can be found in Annex A – Data Action Steps ArcGIS Pro and Annex B – Data Analysis Notebook. The following variables will be created: cycling distance, percentage along trees, percentage along water, percentage along vegetation, percentage over smooth surface material, percentage over cycle paths, network density, average built environment density, average mixture of functions, Euclidean distance and detour distance.

3.5.1 Cycling Distance

The cycling distance is the total trip distance. The cycling distance is the sum of the segment lengths with the same route ID.

3.5.2 Percentage along Trees

The variable percentage along trees represents the occurrence of trees along the network segments. The trees data are taken from the BGT dataset.

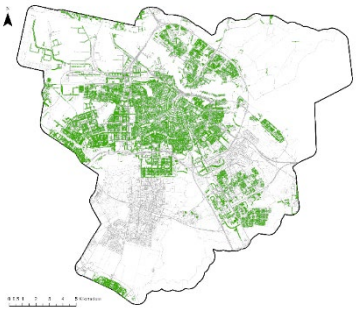


Figure 3.3 Location of trees within the research area

Figure 3.3 visualises the location of the trees within the research area. It shows that there are many trees within the city of Amsterdam compared to the countryside. The variable is created by placing a point on the segment every five metres. From each point, trees are searched for within a radius of 15 metres. If a tree is within reach, the outcome is 'True', otherwise 'False'. An example is shown in Figure 3.4, where eight out of twenty points have a tree in reach.

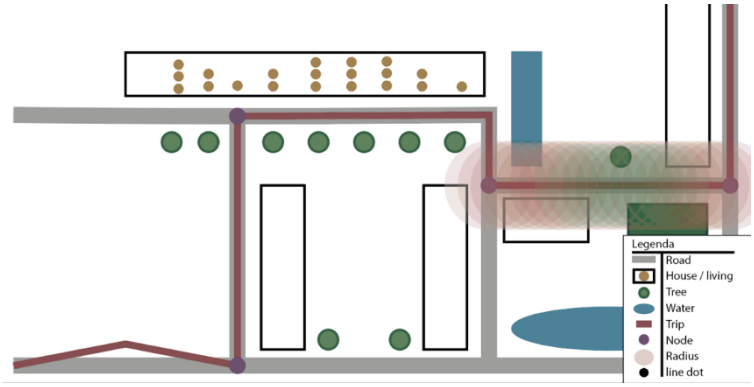


Figure 3.4 Visualisation of how geo information of trees is quantified.

This analysis has been done for each network segment. This results in a total number of points along the network segment and number of points with trees in reach. By adding these numbers together based on the route ID, the total number of points along the route and the number of points with trees within reach is derived. The percentage along trees variable is calculated by dividing the number of points with trees with the total number of points along the trip.

$$Percentage\ along\ trees = \frac{Points\ with\ trees\ in\ presence}{Total\ number\ of\ points}$$

3.5.3 Percentage along Water

The variable percentage along water represents the occurrence of waterbodies along the trip segments. Data about waterbodies are taken from the BGT dataset.

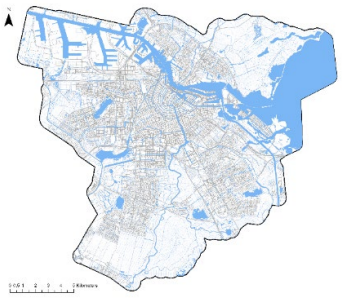


Figure 3.5 Location of water within the research area

Figure 3.5 shows the waterbodies within the research area. The variable is created by placing a point on the segment every five metres. From each point, water bodies are searched within a radius of 15 metres. If a waterbody is present, the outcome is 'True', otherwise 'False'. The variable is created in the same way as the variable for trees.

This results in a total number of points and number of points where waterbodies are present. By adding these numbers together based on the route ID, the total number of points along the journey and the number of points including water bodies are found. The percentage along water is calculated by dividing the number of points with waterbodies nearby with the total number of points along the trip.

$$Percentage\ along\ water\ = \frac{Points\ with\ waterbodies\ in\ presence}{Total\ number\ of\ points}$$

3.5.4 Percentage along Vegetation

The variable percentage along vegetation represents the occurrence of vegetation surfaces along the trip segment. The data about vegetation surfaces are taken from the BGT dataset.

Figure 3.6 shows the location of the vegetation surfaces. The variable is created by placing a point on the segment every five metres. From each point, vegetation surfaces are searched for within a radius of 15 metres. When a vegetation surface is found, the outcome is 'True', otherwise 'False'. The variable is created in the same way as the variable for trees.

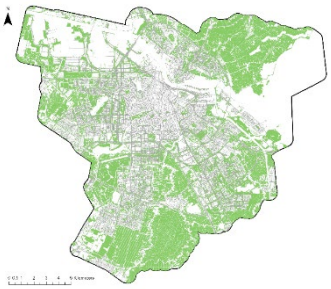


Figure 3.6 Location of vegetation within the research area

This results in a total number of points and number of points where vegetation surfaces are present. By adding these numbers together based on route ID, the total number of points along the trip and the number of points including vegetation surfaces are found. The percentage along vegetation is calculated by dividing the number of points with vegetation nearby with the total number of points along the trip.

$$Percentage\ along\ vegetaiton = \frac{Points\ with\ vegetation\ in\ presence}{Total\ number\ of\ points}$$

3.5.5 Percentage over Smooth Surface Material

The percentage over closed surface material represents the part of the trip that is over asphalt or concrete, the so called smooth surface materials. The surface materials are taken from the BGT dataset. Each trip segment is assigned one of the following surfaces:

- Transition (combination of more than one material)
- Open surface material (i.e., paving stones)
- Closed surface material (i.e. concrete or asphalt)
- Half surface material (i.e. gravel)
- Unpaved (i.e. dirt)

To join the data, 10 points have been generated along the trip segments. The trip segment is assigned to the surface material that was found most. It turned out that the trip segments of the BCW (2016) were not exactly coinciding with the road data of the BGT (2020). In that case, the nearest road segment to the point was used to assign a surface material. Figure 3.7 visualises how the data has been aggregated.

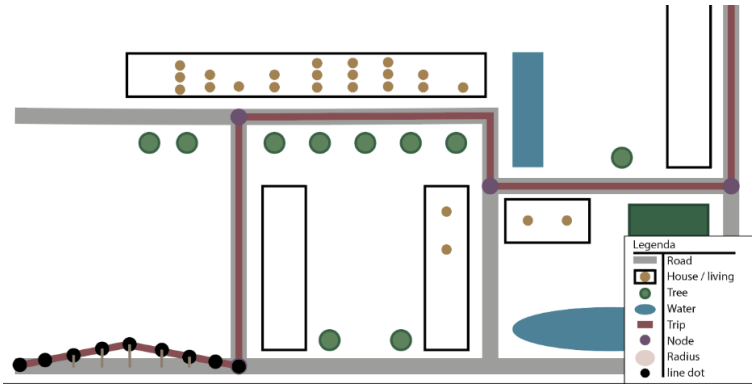


Figure 3.7 Visualisation how information of the surface material is add to the trip segment

The percentage over smooth surface material is calculated by dividing the total number of metres over smooth surface material by the total cycling distance.

$$Percentage\ over\ closed\ surface\ material = \frac{Meters\ over\ closed\ surface\ material}{Cycling\ distance}$$

3.5.6 Percentage over Cycle Paths

The percentage over cycle paths represents the part of the trip that is over cycle paths. The data is already included in the BCW 2016. The percentage over cycle path is the sum of the trip segments over cycle paths divided by the total cycling distance.

$$Percentage\ over\ cycle\ paths = \frac{Meters\ over\ cycle\ paths}{Cycling\ distance}$$

3.5.7 Network Density

The network density is determined by calculating the average distance between crossings or intersections. The smaller the distance between crossings, the higher the network density. Each cycling trip consists of multiple trip segments. The average distance between intersections has been calculated by dividing the cycling distance by the number of trip segments longer than 10 metres. Segments shorter than 10 metres are located at the same crossing and are usually used to cross a street or tramway. The formula looks as follows:

$$Network\ density = \frac{Cycling\ distance}{Number\ of\ segments\ longer\ than\ 10\ meter}$$

3.5.8 Average Built Environment Density

The density of the built environment of the bicycle trip forms the average density of the built environment. There are multiple ways to calculate the built environment density along the trip. In this research, it has been decided to measure it by the number of functions per hectare. For each trip segment the density of the built environment is calculated, and these numbers are used to calculate the average density of the entire trip. The information on functions is taken from the BAG dataset.

For each trip segment, the number of functions within a radius of 30 metres is counted. For example, Figure 3.8 shows 19 functions (dots) within a radius of 30 metres.

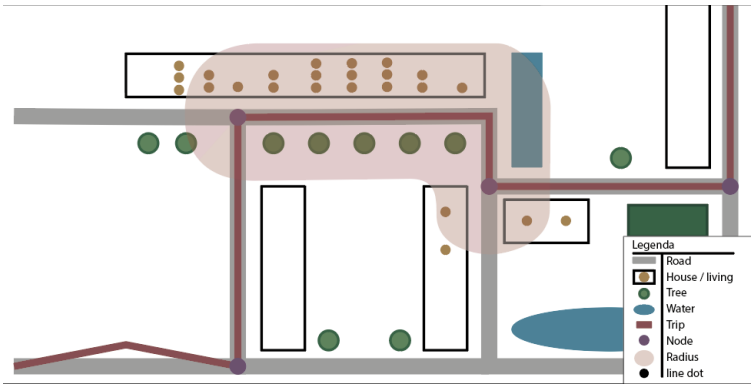


Figure 3.8 Visualisation how functions have been counted around trip segments

The built environment density is calculated by the number of functions within the trip segment divided by the area of the search radius. The result is multiplied by 10,000 to turn it into the number of functions per hectare. To calculate the average density for the entire trip, the density of the trip segment is multiplied by the segment length first, in order to divide it by the total cycling distance afterwards.

$$Segm_{funcdens} = \frac{Number\ of\ functions}{(Segm_{length} \times 2r) + \left(\frac{1}{2} \times \pi \times r^2 \times Segm_{length}\right)} \times 10{,}000 \times Segm_{length}$$

$Segm_{funcdens}$ = function density of the segment (function / Ha)
$Segm_{length}$ = the length of the segment (m)
r = radius that has been used to search for functions (30 metres)

The average built environment density along the trip is calculated by adding together all density values with the same route ID and divide these by the corresponding cycling distance.

$$Average\ built\ environment\ density = \frac{segm_{funcdens}}{Cycling\ distance}$$

3.5.9 Average Mixture of Functions

The average mix of functions is formed by the mixture of the environment of the trip. For each trip segment, the unique functions within a radius of 30 metres are counted. The lowest possible number is 0 (no functions) and the highest is 11 (all possible functions are present). Figure 3.8 shows how the count has been conducted. This example only includes one colour, meaning the trip segment gets the value 1 for mixture of functions.

The number of unique functions within a trip segment is multiplied by the segment length.

$$Segm_{uniquefunctions} = Number\ of\ unique\ functions \times segm_{length}$$

$Segm_{uniquefunctions}$ = the number of unique functions for the segment multiplied by the segment length.

The average mixture of functions along the trip is determined by adding together the total mixture of function values with the same route ID and divide this number by the corresponding cycling distance.

$$Average\ mixture\ of\ functions = \frac{Segm_{uniquefunctions}}{Cycling\ distance}$$

3.5.10 Euclidean Distance

The Euclidean distance is the celestial (as the crow flies) distance between origin and destination. The Euclidean distance is calculated by using the Pythagoras theorem and the start and end coordinates of every trip.

$$Euclidean\ distance = \sqrt{(startX - endX)^2 + (startY - endY)^2}$$

3.5.11 Detour Distance

The detour distance represents the network friction distance plus the behavioural detour distance. The detour distance is the cycling distance minus the Euclidean distance.

$$Detour\ distance = Cycling\ distance - Euclidean\ distance$$

3.5.12 Summary of Created Variables

Table 3.3 provides on overviews of the variables that have been created for the analysis.

Table 3.3 Summary of the variables that have been created for the analysis

Variable	unit	Description
Cycling distance	m	The total distance of the trip
Percentage along trees		Percentage of the trip along trees
Percentage along water		Percentage of the trip along water
Percentage along vegetation		Percentage of the trip along vegetation
Percentage over smooth surface material		Percentage of the trip over asphalt/concrete
Percentage over cycle pats		Percentage of the trip over cycle paths
Network density	m	Average distance between crossings or intersections
Average built environment density	f/Ha	Average built environment density measured in functions per hectare
Average mixture of functions	-	The average number of unique functions along the trip
Detour distance	m	The sum of network friction and behavioural detour
Euclidean distance	m	The celestial (as the crow flies) distance between origin and destination

4 Results

This chapter presents and discusses the research results. The first section provides a description of the data. The next section describes four models and their outcomes. The outcomes are compared with the findings in the literature and compared with the hypotheses defined in chapter 2.

4.1 Descriptive Statistics

This section describes the dataset that is used for modelling. A total of 30,137 trips are still available for the model. This is 873 trips less than the raw data we started with. Some trips have been removed since the coordinates for origin and destination were at the same location, suggesting the trip was made to cycle for leisure. Trips that return to their origin and do not aim to reach a different destination are excluded.

The included trips took place in the muncipalities of Amsterdam, Amstelveen, Ouder-Amstel and Diemen. Most trips start in the municipality of Amsterdam and stay within its boundaries, see Table 4.1 and Figure 4.1. After Amsterdam, most trips come from Amstelveen, Ouder-Amstel and Diemen respectively. or the municipality of Amstelveen, most trips were limited to Amstelveen, only a minority went to Amsterdam. Short trip distances are overrepresented compared to long bike rides. The travel distance for trips from Amsterdam is harder to determine than from other municipalities.

Table 4.1 Cross table showing the number of trips available in the dataset from origin to destination of each municipality

	Destination	Amsterdam	Amstelveen	Ouder-Amstel	Diemen	**Sum of the origin**
Origin	Amsterdam	26,450	707	216	305	27,678
	Amstelveen	653	936	41	5	1,635
	Ouder-Amstel	258	37	69	14	378
	Diemen	262	8	23	153	446
	Sum of the destination	27,623	1,688	349	477	

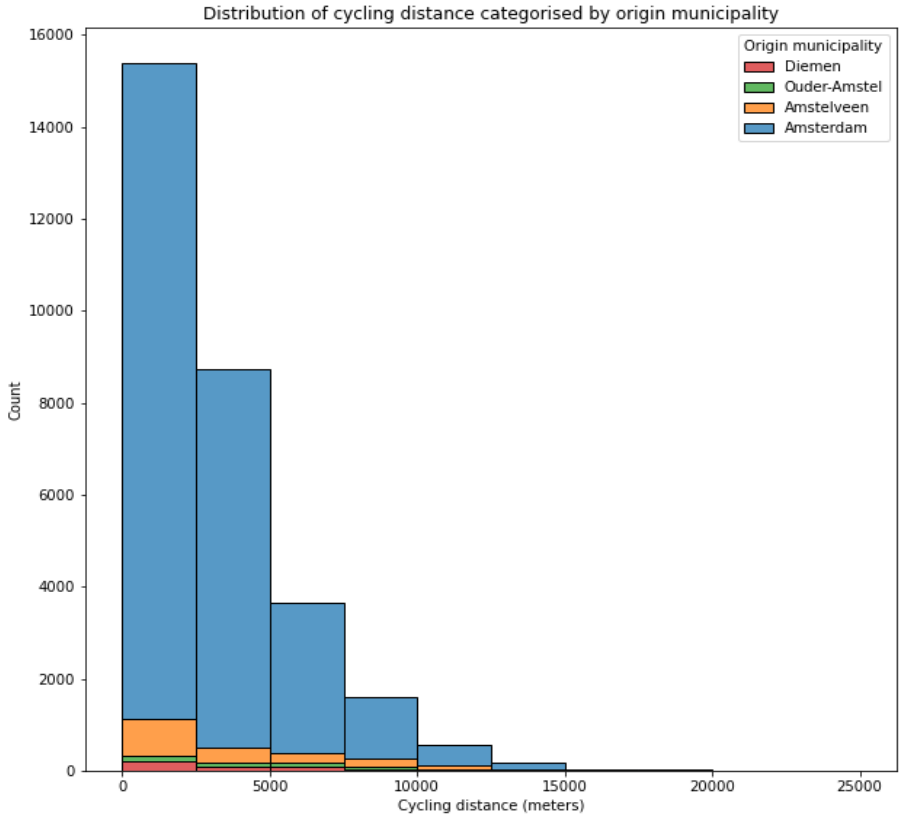


Figure 4.1 Distribution of cycling distance in the dataset categorised by origin municipality.

Table 4.2 gives insight in the spread of the variables and gives the mean, standard deviation, and range.

Table 4.2 Descriptive statistics of the dataset

Variable	mean	std	Range
Cycling distance (m)	3,241	2,590	500 - 26,176
Euclidean distance (m)	2532	2003	261 - 15,505
Percentage along trees	0.53	0.23	0.00 - 1.00
Percentage along water	0.27	0.23	0.00 - 1.00
Percentage along vegetation	0.44	0.31	0.00 - 1.00
Part of trip going over smooth surface material	0.42	0.22	0.00 - 1.00
Part of trip going over cycle paths	0.52	0.28	0.00 - 1.00
Network density (m)	67	23	30 - 1,199
Built environment density (functions/ha)	108	77	0.00 - 483
Mixture of functions	2.67	1.40	0.00 - 7.40

Below are a few key findings:

- The 30,137 trips have a cycling distance between 0.5 kilometres and 26.18 kilometres with an average cycling distance of 3,241 metres.
- On average 53% of the trips is along trees, 27% along water and 44% along vegetation. Some trips do not include any of these, while a few trips are constantly along trees, water, or vegetation.
- On average 42% of the trips is over smooth surface materials like asphalt and concrete.

- On average 52% of the trips is over cycle paths.
- The network density, measured in average distance between crossings, varies between nearly 30 metres and almost 1,200 metres, where the average network density is 67 metres.
- The average built environment density is almost 108 functions per hectare and varies between 0 and 483 functions per hectare.
- The average number of functions along the trip is 2.67 and ranges from 0 to 7.4 functions.

4.1.1 Trip Visualisation

Figure 4.2 shows trip number 209351 from Spijtellaantje, Amsterdam (left) eastward to Strawinskylaan, Amsterdam (right). This trip is representative for the average trip in the dataset. The blue line represents the trip itself, and the red border represents the 30-metre radius. The trip length is 2,509 metres, see Table 4.3. This is slightly less than the average cycling distance of 3,241 metres.

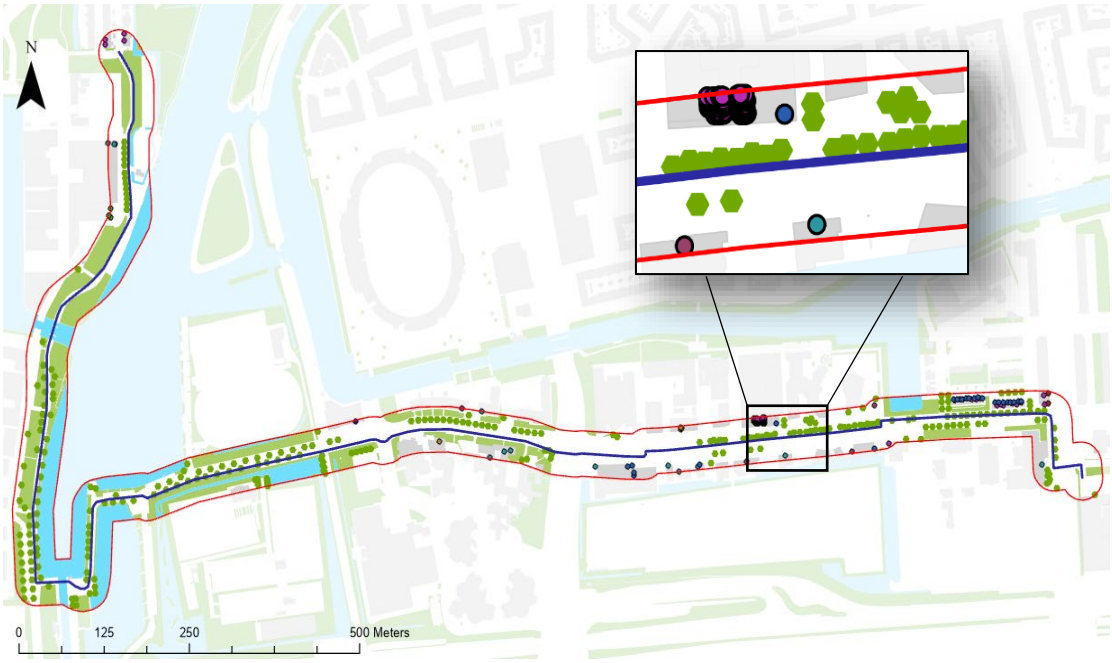


Figure 4.2 Visualisation of an average trip in the dataset

To highlight one indicator, water is found along the first part of the trip and along the end. In total 43% of the trip is going along water. The zoomed-in trip image shows that locations with trees are not necessary indicated with vegetation surfaces. Table 4.3 displays the values of the other variables.

Table 4.3 Information about the variables of the average trip in Figure 4.5

CYCLING DISTANCE	DETOUR DISTANCE	EUCLIDEAN DISTANCE	NETWORK DENSITY	PERCENTAGE ALONG TREES	PERCENTAGE ALONG WATER	PERCENTAGE ALONG VEGETATION	PERCENTAGE OVER SMOOTH SURFACE MATERIAL	PERCENTAGE OVER CYCLE PATHS	BUILT ENVIRONMENT DENSITY	MIXTURE OF FUNCTIONS
2509	987	1,522	70	0.74	0.43	0.76	0.19	0.69	22	1.46

4.1.2 Relation between Cycling Distance Euclidean Distance and Detour Distance

The cycling distance is the sum of the Euclidean distance, network friction and behavioural detour. Figure 4.3 is made with the Seaborn library and visualises the Euclidean distance and detour distance. As cycling distance increases, the spread of Euclidean and detour distance becomes wider. It shows

that for longer cycling distances the influence of the network friction and behavioural detour increases. The graph also shows that the cycling distance is largely determined by the Euclidean distance and less by the detour distance. The mean and standard deviation for Euclidean distance and detour distance are shown in Table 4.2 above.

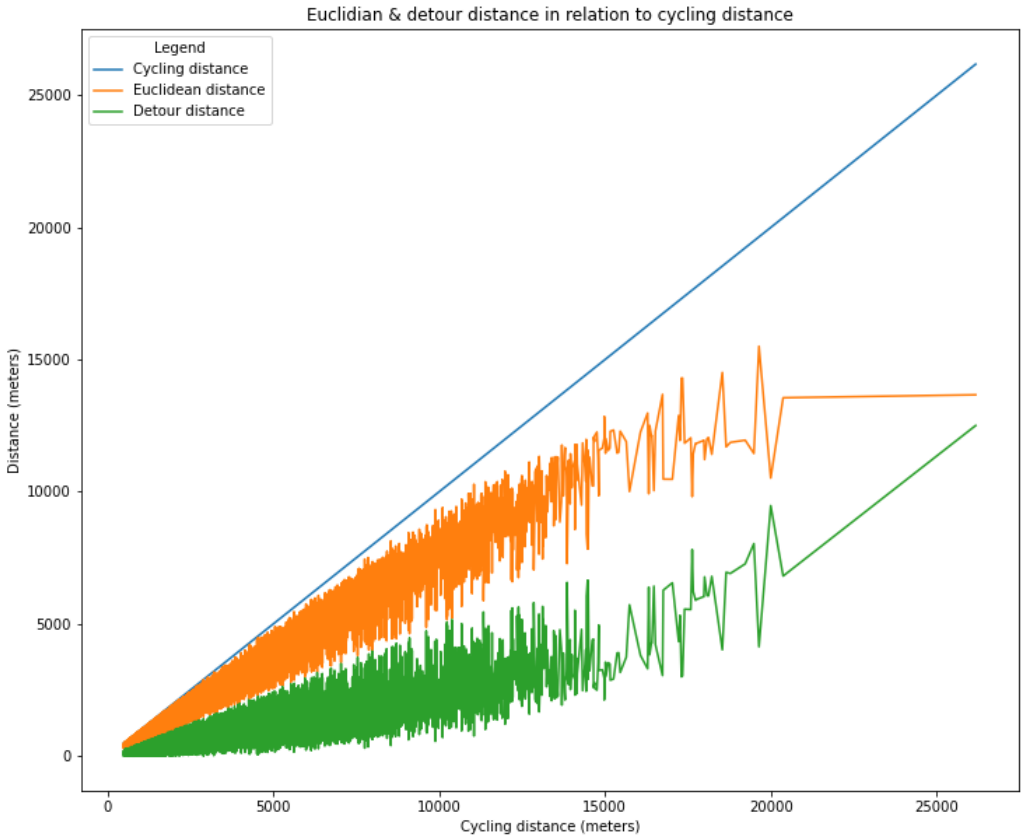


Figure 4.3 Euclidean distance and detour distance in relation with cycling distance

4.1.3 Correlation of the Variables

This research investigates the relation between the cycling distance and elements of the built environment. The relation has been explored by using scatter plots and a correlation matrix. The scatter plots in Annex C – Scatter Plots & Distribution, visualise the relationship between one variable and another, where the correlation matrix only gives us the correlation number.

The correlation matrix in Figure 4.4 shows the relation between cycling distance and the other variables in the left most column. It shows some very weak and some very strong relations with the dependent variable, both negatively and positively. Detour distance and Euclidean distance turn out to have a very strong relation with cycling distance, because these give the cycling distance.

The correlation matrix shows that there are independent variables correlating with each other. A very strong negative correlation is found between the percentage of vegetation and built environment density. This is expected since a larger built area leaves less space for vegetation; see Figure 4.2 and Table 4.3. Locations with more buildings have less vegetation. A strong positive correlation is found between mixture of functions and built environment density. It is expected to find more unique functions in higher built environment densities than in lower built environment densities. The zoomed frame of Figure 4.2 shows four different functions are present in this part. The correlations between independent variables indicate that there is a chance for multicollinearity, but this is not a problem for the explanatory models developed.

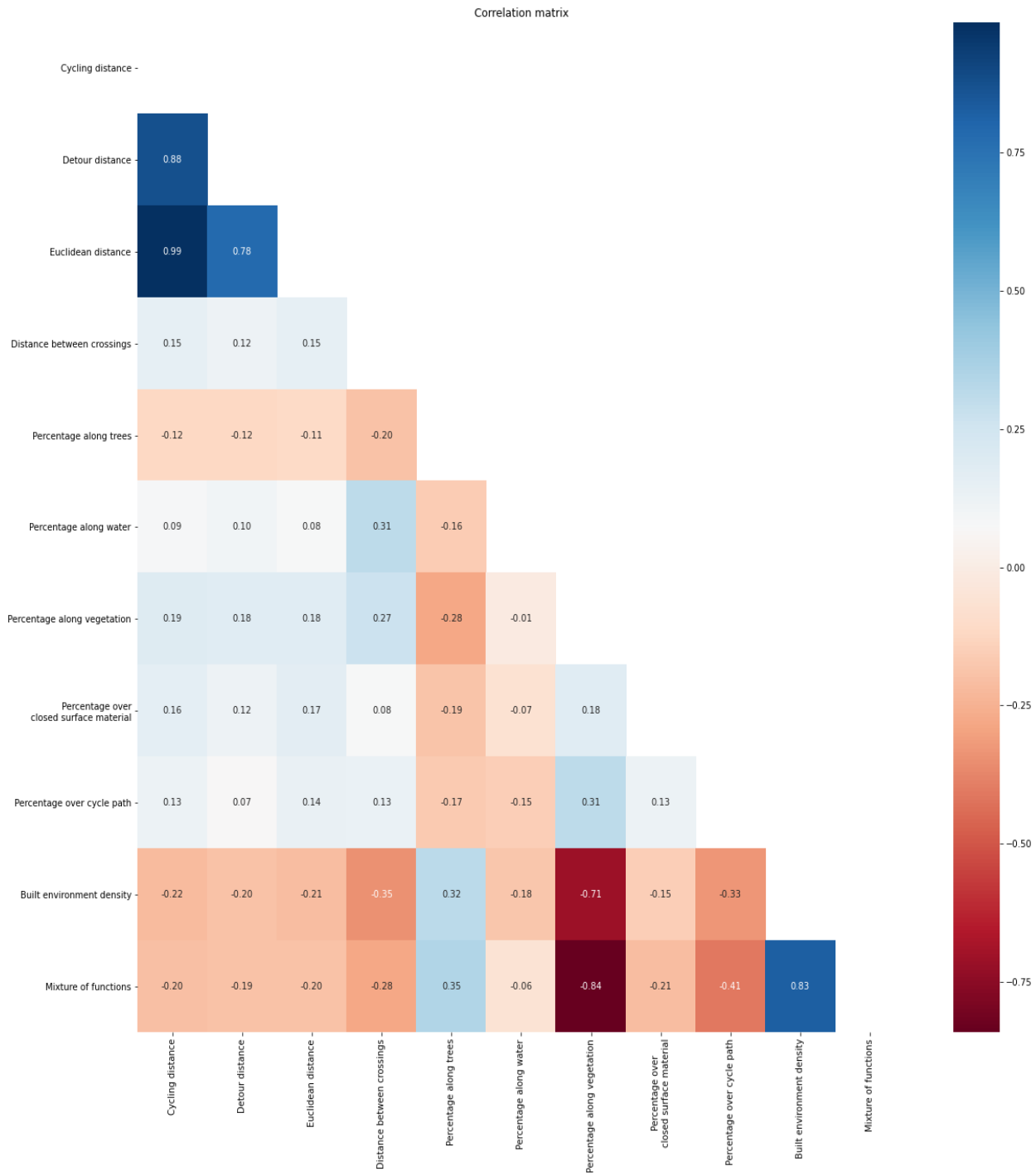


Figure 4.4 Correlation matrix of the variables in the dataset

4.2 Modelling

This section describes the explanatory models. Where the previous section described how the data looks like, this section explores the relation of cycling distance with elements of the built environment by putting all predictors into one model. This shows how well able the elements of the built environment can explain cycling distance and this shows the direction and magnitude of the variable's coefficients.

Model 1: cycling distance ~ built environment

In this model, cycling distance is the dependent variable, while the elements of built environment are the predicting variables. The cycling distance is the total distance from origin to destination. This model shows that the built environment determines 7.7% of the variance in cycling distance; see Table 4.4. All variables have a positive coefficient with cycling distance, except for the percentage along trees and built environment density. The Beta indicates that smooth surface material and built environment density have the strongest influence. One percent increase for smooth surface material increases the cycling distance by 15.21 metres. This supports the hypothesis that cyclist prefer to cycle over asphalt. An increase of one function per hectare decreases the cycling distance with 3.64 metres. This means distances are shorter in more urbanised environments. The Beta indicates smooth surface material and built environment density are the most influencing variables.

Table 4.4 Model one: cycling distance - built environment

Variable	Coeff	Std. Error	Beta	p-value
Constant	1,738.409	122.587		0.000
Percentage along trees	-169.413	67.267	-0.015	0.012
Percentage along water	760.093	70.996	0.067	0.000
Percentage along vegetation	565.562	88.401	0.067	0.000
Percentage over smooth surface material	1,521.249	68.590	0.127	0.000
Percentage over cycle path	603.367	58.210	0.065	0.000
Network density	6.927	0.709	0.061	0.000
Built environment density	-3.643	3,465.268	-0.108	0.000
Mixture of functions	40.625	24.814	0.022	0.102

N = 30.137, R^2 = 0,077, Adj. R^2 = 0,077

Model 2: detour distance ~ built environment

The model explaining detour distance, the cycling distance minus the Euclidean distance, shows an adjusted R-squared of 5.8%; see Table 4.5. This is slightly lower than the R-squared of model one. In this second model all variables have a positive coefficient, except for percentage along trees, built environment density and mixture of functions. The coefficients magnitudes are smaller for all variables than in the first model. This is expected as the detour distance is a smaller part of the cycling distance. The Beta show that the following three variables are the greatest influencers: (a) percentage along water, (b) percentage over smooth surface material and (c) percentage along vegetation. This means cyclist are making detours to cycle through greener environments, along water and over asphalt. In relation to the first model, it is expected that the Euclidean distance only explains 2% of the cycling distance.

Table 4.5 Model two: detour distance - built environment

Variable	Coeff	Std. Error	Beta	p-value
Constant	515.937	33.527		0.000
Percentage along trees	-83.627	18.397	-0.028	0.000
Percentage along water	257.776	19.417	0.083	0.000
Percentage along vegetation	187.307	24.178	0.082	0.000
Percentage over smooth surface material	267.203	18.759	0.083	0.000
Percentage over cycle path	17.525	15.920	0.007	0.271
Network density	0.936	0.194	0.030	0.000
Built environment density	-0.625	947.751	-0.069	0.000
Mixture of functions	-12.345	6.787	-0.025	0.069

N = 30,137, R^2 = 0.059, Adj. R^2 = 0.058

Model 3: Euclidean distance ~ built environment

The third model explains Euclidean distance with the built environment as explanatory factor. The built environment accounts for 7.8% of the variance in Euclidean distance; see Table 4.6. This is almost identical to model one. The coefficient directions and magnitudes are the same in model one. However, for mixture of functions the magnitude has increased compared to model one. The percentage over smooth surface material and built environment have the highest influence according to the Beta.

Table 4.6 model three: Euclidean distance - built environment

Variable	Coeff	Std. Error	Beta	p-value
Constant	1222.472	94.791		0.000
Percentage along trees	-85.7855	52.015	-0.010	0.099
Percentage along water	502.3164	54.898	0.057	0.000
Percentage along vegetation	378.2554	68.357	0.058	0.000
Percentage over smooth surface material	1254.046	53.038	0.136	0.000
Percentage over cycle path	585.8418	45.012	0.081	0.000
Network density	5.9906	0.548	0.068	0.000
Built environment density	-3.0185	0.268	-0.116	0.000
Mixture of functions	52.9699	19.188	0.037	0.006

N = 30,137, R^2 = 0.078, Adj. R^2 = 0.078

Model 4: cycling distance ~ built environment + Euclidean distance

In this model, the cycling distance is modelled based on the variables built environment and Euclidean distance. Euclidean distance is added as an independent variable, so the model corrects itself for the destination that has been chosen. Destination is the first choice to make and is known for the trips. By including Euclidean distance in the model, the influence of the built environment on destination choice is covered by this variable. The included built environment variables only need to account for the remaining unexplained distance: the detour distance.

Model four explains 97.2% of the variance for cycling distance; see Table 4.7, showing that the Euclidean distance largely determines the cycling distance. Model three shows the influence of the built environment on the Euclidean distance. This means for this model; the cycling distance is influenced by built environment through Euclidean distance. In addition, cycling distance is influenced by the effect of the built environment through network friction and behavioural detour.

Compared to model one, the variables percentage over smooth surface material, percentage over cycle paths and mixture of functions become negative, while built environment density becomes positive. These aspects will be discussed in subsection 4.2.1.

Compared to a bivariate model, where the Euclidean distance as the one explanatory factor gives an R-squared of 97.2%, see Annex B – Data Analysis Notebook. This leaves 2.8% of the cycling distance unexplained and the explanation should come from the detour distance. However, this is not correct, since the other bivariate model shows that detour distance is able to explain 76.9% of the variance in cycling distance. The conclusion is therefore justified that other influencing factors determine cycling distance. These factors are most likely identical to the ones explaining Euclidean distance and detour distance.

Table 4.7 Model four: Cycling distance - built environment + Euclidean distance

Variable	Coeff	Std. Error	Beta	P-value
Constant	181.387	21.299		0.000
Percentage along trees	-60.151	11.656	-0.005	0.000
Percentage along water	120.309	12.318	0.011	0.000
Percentage along vegetation	83.791	15.325	0.010	0.000
Percentage over smooth surface material	-75.987	11.994	-0.006	0.000
Percentage over cycle path	-142.800	10.114	-0.015	0.000
Network density	-0.703	0.123	-0.006	0.000
Built environment density	0.202	601.688	0.006	0.001
Mixture of functions	-26.841	4.300	-0.015	0.000
Euclidean distance	1.274	0.001	0.985	0.000

N = 30.137, R^2 = 0,972, Adj. R^2 = 0,972

4.2.1 Results Interpretation and Hypotheses Testing

Model one shows that the variance for cycling distance can be explained for 7.7% without including Euclidean distance as predictive variable. Model two shows that the built environment explains for 5.8% of the detour distance. This suggests that Euclidean distance would only explain for 2% of the variance. However, if the Euclidean distance is added to the equation, this increases to 97.2% of the variance. Bivariate models with cycling distance as dependent variable and Euclidean distance and detour distance as independent variables show an R-squared of 97.2% and 76.9% respectively. The models show that the built environment is influencing the total cycling distance through Euclidean distance, network friction and behavioural detour.

In the models the constants, known as the Y-intercepts, differ from model one to four. In none of the models, the explanatory factors can be zero. Therefore, there is no interest in discussing these constants.

Green Environments

According to literature, green environments, like parks and plantation along streets are positively associated to cycling (Fraser & Lock, 2011; Heinen et al., 2010). The green environments have been measured by the presence of vegetation and trees within a specified radius along the trip. The presence of trees showed a negative correlation with cycling distance, while vegetation along the trip has a positive influence. An explanation for trees is that the data for rural areas are not accurate, see Figure 3.3. Acknowledging the variable trees is not a good indicator, the hypothesis: "The greener the environment, the longer the cycling distance is" can be confirmed based on the variable percentage along vegetation.

Percentage along trees indicates the effect of trees along the trip on the response variable. Something noticeable is that the percentage of trees has a negative correlation in all four models. The coefficient in the first model on cycling distance is -169.41 if the trip goes 1% more along trees. Only the Beta in model three shows it does not have a big impact in comparison to the other variables. Looking at model 4, the impact on cycling distance is almost three times smaller as the coefficient is -60.15. Trees also have a negative impact on the detour distance, while assumed more trees would make it more interesting to detour. Model three shows trees also have a negative influence on Euclidean distance. Comparing it with the variable vegetation, it has the opposite effect, but it is possible that the data for trees differ from that of vegetation and that the surface below trees has not be indicated as vegetation surface.

Percentage along vegetation indicates the effect of vegetation surfaces along the trip on the response variable. Vegetation has a positive relation to cycling distance, detour distance and Euclidean distance. The coefficient of vegetation is 83.79 in model four. This is about 7 times smaller than in the first model. It follows the same pattern as water; only the influence on the cycling distance and detour distance is smaller than in the case of water. It could be that the area around water has many vegetation surfaces. However, this would be different within the city of Amsterdam with its numerous canals. In comparison with the variable trees, it has the opposite effect. It could be that the surface near the trunk of the tree has not be characterised as vegetation surface. The figures 3.3 and 3.4 show that many trees are found in the city, while other vegetation surfaces are primarily found outside the city.

The literature states that green environments encourage people to cycle. The effect of trees on cycling distance is only negative. Meaning the presence of trees would decrease the cycling distance. An explanation for this effect could be that cyclists take shortcuts, where they go through a park instead of going around it, having trees around the path. A second explanation could be that data about trees are not accurate for places with low built environment density, like rural areas. Figure 3.3 partially confirms this second explanation. The map shows there are almost no tree data in rural areas and the municipality of Amstelveen.

Green environments would stimulate cycling according to the literature. In our models this is supported for vegetation surfaces. Even if the Euclidean distance is added to the equation, the effect stays positive, only the impact is much smaller. Model two shows that vegetation also has effect on detour distance and could influence network friction or behavioural detour.

Percentage along Waterbodies

Amsterdam is rich of canals and waterways along the streets. Like green environments, water could also have a positive effect on cycling. The presence of water is measured within a specified radius along the trip. A higher percentage of water was found along longer cycling distances. This makes hypothesis 5: "The more along waterbodies, the longer cycling distance is" plausible. However, the effect is much

smaller if Euclidean distances is included in the hypotheses. A reason could be that waterways cause higher network friction. It could also be a reason for someone to detour.

Percentage along water indicates the effect of waterbodies along the trip on the response variable. A higher percentage of water along the trip increases the cycling distance in models one and three and the detour distance in model two. The coefficient in model two is 258, with a Beta of 0.083. This is the second highest coefficient and highest Beta in the model. Water is significantly influencing the network friction and behavioural detour. This is confirmed by model four, where water has again second highest coefficient (120.31).

Water has a high impact on cycling distance even if Euclidean distance has been added to the model. Predicting Euclidean distance, the variable water has also a high coefficient. For this reason, water could be causing a higher network friction, since you can only cross it at bridges, or people really enjoy cycling along waterways and add more behavioural detour distance to the total cycling distance.

Percentage over Cycle Paths

Separated cycle paths are in many countries linked to more use of bicycles. In the Netherlands, more mixed result occurs. Our data provide information on cycle paths but not if they were separated. The distance over cycle paths has been added together and a percentage of the trip over cycle paths has been calculated. A relation has been found that longer cycling distances are made over cycle paths. This confirms hypothesis 2: “The more use of cycle paths, the longer the cycling distance “.

Percentage over cycle paths indicates the effect of part of the trip going over cycle paths on the response variable. Trips going over cycle paths increase the cycling distance in model one. The coefficient in this model is 603.37. However as soon as Euclidean distance is added to the equation, the effect becomes negative 142.80. The percentage over cycle paths has a coefficient of 17.53 in the model of detour distance. It shows cycle paths do not have a big effect on detour distance, and with a p-value of 0.271 not significant. The negative effect in model three could be explained by the fact that cycle paths can be shortcuts in the built environment, comparable to parks.

For cycle paths mixed results were found if it would stimulate cycling or not in the Netherlands. The models confirm that there is no clear effect of cycle paths on travel distance. The variable is not significant in model two and has a very small effect in model four. This could mean that the paths are used as shortcuts through the built environment. In support of this explanation is the fact that the built environment density is increasing travel distance in model four.

Percentage over Smooth Surface Material

Cyclists prefer a smooth surface quality, like asphalt and concrete. High comfort and lower rolling resistance account for this (Hölzel et al., 2012). The distance over smooth surface material have been added together for calculating the percentage of the trip is over this type of surface material. A higher percentage of asphalt along the trip results in longer cycling distances. This confirms hypothesis 4: “The more over smooth hardening, the longer the cycling distance”.

Percentage over smooth surface material indicates the effect of part of the trip going over smooth surface material on the response variable. Percentage over smooth surface material, that is in the Netherlands asphalt or concrete, is positive in the first three models and negative in the fourth model, like cycle paths. The first model shows a coefficient of 1521.25 that is the highest of all. The effect on detour distance is already 5 times smaller for surface material 267.20. This means asphalt or concrete is found more for longer cycling distances than for shorter distances. The effect becomes negative as soon as the Euclidean distance is added to the equation. Possibly cycle paths, which are also negative, are made more often of asphalt than other pavement materials.

The literature states that asphalt and concrete are preferred by cyclists as it increases the comfort of cycling. It decreases the rolling resistance meaning it takes less effort to cycle over asphalt than other pavements. It is found that longer cycling distances are cycled over asphalt.

Network Density

A higher network density results in shorter distances as the network distance becomes closer to the distance as the crow flies (Heinen et al., 2010). Network density is measured as the distance between crossings, excluding segments that were shorter than 10 metres. The models do not indicate a strong relation between network density and cycling distance. Therefore, hypothesis 1: The denser the network, the shorter cycling distance" cannot be confirmed.

The network density indicates the effect of network density on the response variable. The network density is measured as the average distance between crossings. In the first model the coefficient is 6.93, meaning 1.00 metres increase in network density increases the cycling distance by 7 metres. It has almost no effect on the detour distance (0.94) and in model four, which includes the Euclidean distance, it is negative -0.70. Compared to the other variable in the model, the effect is small.

The literature states that a denser network makes it more attractive for people to cycle, as the cycling distance may become closer to the Euclidean distance. The side effect was that it increases the number of crossings and turns that have a negative impact on someone's route choice. While network density is significant in all four models, it has a small impact on cycling distance, detour distance, and Euclidean distance.

Built Environment Density

The built environment density was found to be related to cycling. A higher built environment density is positively related to selecting the bicycle and it makes cycling distances shorter as everything is compacter and closer to one another. The number of functions per square metre along the trip measures the built environment density. A higher built environment density decreases the cycling distance. This confirms hypothesis 6: "The denser the built environment, the shorter the cycling distance". This could be due the fact destination is closer to the origin in high dense areas than at location with a low dense built environment. However, within cities, the built environment may be reason to detour, as it is less possible to cycle in a straight line to the destination.

The built environment density indicates the effect of the density of the built environment on the response variable. The coefficient of built environment density in the first model is -3.643 on cycling distance. In the second model it is -0.625 on detour distance. A higher built environment density decreases the cycling distance. An increase of the built environment density decreases the cycling distance and even the detour distance. In model four, where Euclidean distance is added to the equation the effect becomes the opposite and the coefficient becomes positive 0.20.

The literature states that a higher built environment density is found positive to cycling as the destination may become closer to origin. It also states that cyclists choose their routes through low dense areas. The first three models found that the relation of the built environment density is negatively related to cycling distance and detour distance. So, distances become shorter if the built environment density increases. In the fourth model the relation becomes positive with built environment density. This means cycling distance becomes longer if average density increases. This could mean the built environment adds distance due to network friction, or people making behavioural detours confirming that they may choose for areas with a low built environment density.

Mixture of Functions

A high mixture of functions encourages the use of active modes while low mixture discourage. (Heinen et al., 2010; Ton et al., 2019). In areas with higher mixture of functions it is more likely to find flexible destination closer to the origin. The mixture of functions is measured as the number of different functions along the trip. Varied results have been found for mixture of functions, therefore the hypothesis 7: “The larger mixture of functions, the shorter the cycling distance” cannot be confirmed.

The mixture of functions indicates the effect of average number of unique functions along the trip on the response variable. In the first model the coefficient is 40.63, meaning if the average mixture of functions increases by 1, the cycling distances increases with 40.63 metres. However, the models show the variable is not significant. If the Euclidean distance is included, the relation is even negative -26.84. Making the cycling distance shorter if the average mixture of function increases. In the second model it has a negative coefficient of -12.35, even in this model it is shown that the variable is not significant. The average mixture of functions can range from 0 to 7.40, see Table 4.2. This means the cycling distance could only increase by 300 metres or decrease by nearly 200 metres. The effect on cycling distance and detour distance is quite small compared to other variables.

A higher mixture of functions is related to shorter cycling distances when the destination is closer to the origin. The third model shows that a higher average mixture of functions along the trip decreases the cycling distance and confirms what is stated in the literature. However, the impact of mixture of functions on cycling distance stays small.

5 Conclusion & Discussion

This thesis examined the impact of built environment on cycling distance. In general, this study has contributed to the existing body of knowledge on how characteristics of the built environment can influence the distance cyclists' travel. The case study area was Amsterdam and its neighbouring municipalities Amstelveen, Diemen, and Ouder-Amstel. This chapter finalizes this research by answering the research questions and provides recommendations for further research and policy makers.

5.1 Conclusion

Cycling distance and built environment are important factors for the decision to cycle to a destination. It is known that the built environment affects cycling distance. When the built environment is more urban, more destinations are nearby, which tends to reduce the Euclidean distance. At the same time, network density increases with urbanisation reducing network friction. Finally, the built environment influences someone's choice to cycle a certain route, affecting the distance. There are three major decisions made before and during a cycling trip: destination choice, mode choice, and route choice. People make these choices simultaneously, but in this thesis the fixed order of destination, mode, and route choice is assumed for better understanding. With such a fixed order, it is possible to say that (a) destination choice is connected to Euclidean distance, (b) mode choice is connected to network friction, and (c) route choice is connected to behavioural detour. Knowing the influence of the built environment on cycling distance helps to design an environment that is positively influencing cycling.

The first sub-question: "Which elements of the built environment are associated to cycling distance according to the literature and how can they be measured?"

Six elements related to cycling distance are found in the literature: green environments, cycle paths, smooth surface material, network density, built environment density and mixture of functions. Waterbodies has been added as a seventh element, since waterways and canals are typical for the spatial context of Amsterdam. Green environments and smooth surface material are associated with longer cycling distances. On the other hand, a mixture of functions and a higher network and built environment density are associated with a shorter cycling distance. For waterbodies it is assumed that it has the same positive effect as green environments. Cycle paths would not have a notable effect on the cycling distance. The elements network density, mixture of function and built environment density are measured as an average along the trip. The elements water, green environments, smooth surface material and cycle paths are measured as a percentage of the trip going along or over those elements. A specific description of the elements can be found in chapter three Table 3.3.

The second sub-question: "To what extent does the built environment explain the total cycling distance?"

This thesis shows that the elements of the built environment that have been analysed explain 7.7% of the variance in cycling distance. Green environments, water, cycle path and smooth surface materials increase the cycle distance. A higher built environment density and network density shortens the cycling distance. An increase in the mixture of functions is found to increase the cycling distance. However, the effect of mixture of functions is negligible (200-300 metres only). Surface quality and built environment density turn out to be the most influencing factors.

The third sub-question: "To what extent does the built environment explain the detour distance?"

Detour distance is the distance that includes network friction and behavioural detour. Model 2 showed that the built environment can explain 5.8% of the variance in detour distance. The detour distance increases if higher levels of green environments, water bodies, cycle paths, and smooth surface material are available along the road. The detour distances decline when the built environment,

network density and mixture of functions increase. The most influencing elements are green environments, water, and smooth surface material.

The fourth sub-question: "To what extent does the built environment explain the Euclidean distance?"
The built environment can account for 7.8% of the variance in Euclidean distance. Destinations may become further away if the trip becomes greener, along waterways; have cycling paths and smooth surface material. The Euclidean distance also increases when a higher mixture of functions is found along the trip, but the effect is small. The Euclidean distance decreases when built environment and network density increase. The most influencing elements are smooth surface material and built environment density.

The fifth sub-question: "To what extent does the Euclidean distance contribute to the cycling distance?"
The Euclidean distance can explain almost everything for the cycling distance, compared to the other variables in model 4. The model explains 97.2% of the variance in cycling distance that means it can explain almost the entire distance. However, it is important to keep in mind that the built environment is influencing Euclidean distance itself, as is explained in the previous paragraph.

The main research question: "To what extent does the built environment explain cycling distances within the municipality of Amsterdam?"
This thesis investigated the effects of the built environment on cycling distance in the Amsterdam area. Cycling distance is the Euclidean distance, network friction and behavioural detour combined. In this thesis the effects of the built environment have been tested on the total cycling distance and have been split up into Euclidean distance and detour distance. The models suggest a clear influence of the built environment on cycling distance. The effect has also been tested on Euclidean distance and detour distance and shows a slightly larger effect on Euclidean distance than on detour distance. This means that the built environment influences the destination choice more than route choice.

The built environment explains for 7.7% of the variance in cycling distance, see Table 4.4. This is quite large regarding to only seven elements have been researched in this thesis. Green environments and waterbodies along the trip are found to stimulate cycling for longer distances. The built environment and network density are found as factors for decreasing the cycling distance. The mixture of functions did not give a clear effect. The expectation was that it would make distances shorter, but the opposite was found as well. For cycle paths similar mixed results are found. Smooth surface materials like asphalt are found along longer cycling distances. The cycling infrastructure in the Netherlands is of high quality and it is therefore logic to find smooth surface materials along longer cycling distances. The most striking results are that built environment density is most important for Euclidean distance, while green environments and waterbodies are most important for detour distance. This shows that the built environment density is more important for the destination choice and that in more urbanized environments, destinations are closer to origin. Furthermore, green environments and waterbodies are most important for detour distance and therefore incentive elements of the built environment to overcome the distance added by the network friction or provide reasons for behavioural detour. This shows that bicycle use for longer distances can be stimulated by adding more green and potential waterbodies in the built environment.

5.2 Discussion

This section offers a reflection of the study. This is divided into four parts, starting with a discussion what the results mean. Next, the quality of the data. Thirdly, the data gathering itself and finally a review of the data analysis method.

Results Interpretation

50% of the car trips made are shorter than 7.5 kilometres and the question is how to decrease this percentage in favour of traveling by bicycle. This thesis investigated what elements of the built environment are found along longer cycling distances. Some positive results are found. Changing the built environment and investing in green, cycle paths and asphalt could stimulate the use of bicycles for longer cycling distances. There are many other factors influencing cycling, but it would still be an improvement to invest in the built environment.

The average cycling distance is 3.2 kilometres. Trips are going for 44% through green environments, 27% along water, 42% over asphalt, 52% over cycle paths and the average distance between crossings is 67 metres. The built environment that is cycled through has an average of 108 functions per hectare with an average of 3 different functions, see Table 4.2. The results show that investing in greener environments and water along the trip, cycle paths and asphalt make destinations further away more attractive for the use of bicycles. A side effect could be that it will stimulate people, who currently use their car for distance up to 3.2 kilometres, to go cycling. The results show that investing in the bicycle network makes it more attractive to start using the bicycle for certain distances. The distance between crossings could be increased by removing intersections where cyclist need to interact with other traffic. The results of this thesis support the investments in bicycle highways, as those stimulate cycling for longer cycling distances.

Quality

A big part of this research relies on the quality of the data. Data came mainly from two sources: Bikeprint that provided Bicycle Counting Week (BCW) data including trips that were made by cyclists in 2016 and Kadaster that provided data about the built environment. The data of BCW consist of anonymized trips that were provided voluntary. This could indicate that a lot of the data has been gathered by bicycle enthusiasts and is not completely representative of the population. Due to the anonymization, it was impossible to see which trips were made by one person or what the demographics of this person were. The real origin and destination of each trip has been cut of randomized between 100 and 300 metres, so it was not possible to know from where the trip was coming from or going to. From the data about the built environment only the location was available. The quality of the element is unknown. For example, it is known where vegetation surfaces were located, but unknown what sort of vegetation was located there, and which is preferred by cyclists. This could influence someone's choice to select a certain route. Finally, data of the built environment in the same timeframe (2016) as that of the BCW could not be found. This means the environment where the trips were made could have changed over time. Nevertheless, this study still indicates there is a relation between elements of the built environment and cycling distance, but there is room for data quality improvements.

Gathering, Analysis and Results

There are many methods to add information of the built environment to the trips. There were two ways selected to add data to trips. The first method is based on points generated every five metres along the trip segments. From each point within 15 metres radius a search is done for water, vegetation, and trees. The first flaw is that points generated along the trip segments could be placed closer together. This might have given a more precise view if these elements were present or not. It was not possible to set the distance between points closer as it extended the computing time exponentially. By looking at distances between points of 15 metres and 5 metres, the time increased from 10 minutes to 2,5 hours to analyse one element. In addition, the search areas had already a lot of overlap with each other making it not necessary to decrease the distance between points. The search radius has been set to 15 metres, because this includes most parts of the streets and public

space, without including the areas behind buildings. The second method used is to collect data within a radius of 30 metres around trip segments to count functions. This has been chosen to be the method to indicate what the function density is around the segments and what type of functions are present. A radius of 30 metres is selected as it includes buildings from both sides of the street and excludes buildings along parallel oriented streets.

The analysis only looks at the average, vegetation, trees, water, function density and mixture of functions. It is unknown how these elements are spread along the trip. This means we cannot determine if a trip is going through high dense green environment like a forest or park. Those parks and forests may be the reason that trips are longer. Furthermore, the built environment density can be calculated in many ways. In this thesis density is calculated with functions, while it is also possible to do it with the number of buildings, or buildings square metres. The density is calculated per trip segment first and next an average was calculated for the entire trip. The problem with the method is that search areas are overlapping at crossings. A qualitative better option is to count all functions within a trip and divide it by the search area, but due the increase in computing time it was not feasible in the set time frame of this research. In this thesis the detour distance calculated with the use of Euclidean distance. This made it not possible to distinguish the influence of network friction and behavioural detour separately. Another method was to calculate the shortest path for every trip. Finally, Model one explains 7,7% of the variance in cycling distance with the elements of the built environment. This means there are other factors explaining cycling distance than the factors used in this study. Future research should search for these other explaining factors.

General Execution

In general, the scope of this study could have been more restricted. To be specific, the approach from a wider perspective of elements associated to cycling behaviour funnelled to elements influencing cycling distance, could have been specified to only those elements influencing cycling distance. It took a lot of time to understand and get the focus on the important parts of this research. The approach from destination, mode and route choice first make it harder to understand and get the focus on what was most important in this research: cycling distance. Nevertheless, due the stepwise approach that has been adopted a wider understanding of the topic has been obtained. This has led in qualitative results which form a basis for further research.

5.3 Recommendations

The following subsections contain recommendations for researchers that want to conduct additional research on this topic, as well as recommendations for policy makers who are trying to stimulate bicycle use for longer cycling distances.

5.3.1 Research Recommendations

This research gives insight in some elements of the built environment and how they are correlated with cycling distance. The results are promising and open opportunities for follow-up research. This can be split into improvements and extensions. This research analysed the elements waterbodies and green environments in a way it has looked in the present from a point within a radius of 15 metres. The quality of the vegetation surface or waterbodies is unknown. For example, vegetation surfaces could be field of grass or a perk of flowers, or waterbodies could be a ditch or a river. It would be an addition to know the quality of the surfaces and bodies. Secondly, it would be good to search for other elements of the built environment that may influence cycling distance. This could be elements of the built environment for the categories Design, Density or Diversity. For example, other aesthetic elements or traffic densities. Lastly, this research investigated the elements along the trip, while origin and destination are even important for someone to cycle certain distance. The investigating origin and

destination it could become interesting to look at elements in the categories of the other four D's: Distance to transit, Destination accessibility, Demand management and Demographics. Last, it would be interesting to repeat this research with a different study area to find out if results are matching and policy recommendations would be applicable for other places.

5.3.1 Policy Recommendations

Based on the conclusions, it is highly recommended that policy makers and practitioners remain investing in bicycle infrastructure. Cyclists prefer to use cycle paths, for their route choice. Usage of cycle path' share increases as trips become longer. Cyclists prefer smooth surfaces for cycling as it increases the comfort and decreases physical effort. Therefore, it is recommended to use asphalt as surface material for bicycle routes. Within cities, trip distances are found to be shorter due the fact destinations are closer to origin. To stimulate bicycle usage through the built environment it would be helpful to invest in green and waterbodies in the city. Outside the cities, the increasing share of e-bikes (Ministry of Infrastructure and Water Management, 2019) makes it more interesting to invest in bicycle highways. The recommendation is to make those highways of asphalt, invest in green along the highway and keep the number of intersections to a bare minimum, because lower network density has been found along longer cycling trips.

The choice to cycle instead of using the car or public transport is highly dependent on the cycling distance. On average, the more compact the environment is, the shorter the distances to the destination are. People are willing to cycle for better infrastructure and a nicer environment. However, this does extend the cycling distance, so it is not unlikely that poor infrastructure and an unattractive environment will deter people from cycling as well. For policy makers who want to encourage cycling, this is important to realise.

6 References

Cambridge University. (n.d.). *Cambridge Dictionary*. Retrieved October 28, 2020, from https://dictionary.cambridge.org/

CBS. (2018). *4 procent lopend naar het werk*. https://www.cbs.nl/nl-nl/nieuws/2018/14/4-procent-lopend-naar-het-werk

Chowdhury, S. (2017). *Faculty of Engineering and Technology , Civil Engineering and Management Intrapersonal Variation in Destination Choice* . 55. http://purl.utwente.nl/essays/73775

Coya AG. (2019). *Global Bicycle Cities Index 2019*. https://www.coya.com/bike/index-2019

Ewing, R., & Cervero, R. (2010). Travel and the built environment. *Journal of the American Planning Association*, *76*(3), 265–294. https://doi.org/10.1080/01944361003766766

Fraser, S. D. S., & Lock, K. (2011). Cycling for transport and public health: A systematic review of the effect of the environment on cycling. *European Journal of Public Health*, *21*(6), 738–743. https://doi.org/10.1093/eurpub/ckq145

Handy, S., van Wee, B., & Kroesen, M. (2014). Promoting Cycling for Transport: Research Needs and Challenges. *Transport Reviews*, *34*(1), 4–24. https://doi.org/10.1080/01441647.2013.860204

Harms, L., & Kansen, M. (2017). Fietsfeiten. *Kennisinstituut Voor Mobiliteitsbeleid KiM*, 16.

Heinen, E., van Wee, B., & Maat, K. (2010). Commuting by bicycle: An overview of the literature. *Transport Reviews*, *30*(1), 59–96. https://doi.org/10.1080/01441640903187001

Hölzel, C., Höchtl, F., & Senner, V. (2012). Cycling comfort on different road surfaces. *Procedia Engineering*, *34*, 479–484. https://doi.org/10.1016/j.proeng.2012.04.082

INTREG. (n.d.). *Why investing in cycle highways?* Retrieved November 16, 2020, from https://cyclehighways.eu/about/why-investing-in-cycle-highways.html

Kadaster. (n.d.-a). *Dataset: Basisregistratie Adressen en Gebouwen (BAG)*. Retrieved September 4, 2020, from https://www.pdok.nl/introductie/-/article/basisregistratie-adressen-en-gebouwen-ba-1

Kadaster. (n.d.-b). *Dataset: Basisregistratie Grootschalige Topografie (BGT)*. Retrieved June 12, 2020, from https://www.pdok.nl/introductie/-/article/basisregistratie-grootschalige-topografie-bgt-

Ministry of Economic Affairs and Climate Policy. (2019). *Klimaatakkoord*. 250. https://www.klimaatakkoord.nl/binaries/klimaatakkoord/documenten/publicaties/2019/06/28/klimaatakkoord/klimaatakkoord.pdf

Ministry of Infrastructure and Water Management. (2019). *Mobiliteitsbeeld 2019*. 204. https://www.kimnet.nl/publicaties/%0Ahttps://www.kimnet.nl/publicaties/rapporten/2019/11/12/mobiliteitsbeeld-2019-vooral-het-gebruik-van-de-trein-neemt-toe

Ministry of Transport Denmark. (2012). *The Danish Transport System, Facts and Figures*.

Mitra, R. (2013). Independent Mobility and Mode Choice for School Transportation: A Review and Framework for Future Research. *Transport Reviews*, *33*(1), 21–43. https://doi.org/10.1080/01441647.2012.743490

Municipality of Leeuwarden. (2013). *Richtingwijzer fiets*. https://www.crow.nl/downloads/documents/kpvv-beleidsdocumenten/richtingwijzer-fiets-2006

Muñoz, B., Monzon, A., & Daziano, R. A. (2016). The Increasing Role of Latent Variables in Modelling Bicycle Mode Choice. *Transport Reviews*, *36*(6), 737–771. https://doi.org/10.1080/01441647.2016.1162874

NOS. (2018). *Woon je dicht bij je werk? Toch 19 cent fietsvergoeding, wil staatssecretaris | NOS*. https://nos.nl/artikel/2236131-woon-je-dicht-bij-je-werk-toch-19-cent-fietsvergoeding-wil-staatssecretaris.html

Prato, C. G., Halldórsdóttir, K., & Nielsen, O. A. (2018). Evaluation of land-use and transport network effects on cyclists' route choices in the Copenhagen Region in value-of-distance space. *International Journal of Sustainable Transportation*, *12*(10), 770–781. https://doi.org/10.1080/15568318.2018.1437236

Pritchard, R., Frøyen, Y., & Snizek, B. (2019). Bicycle level of service for route choice—A GIS evaluation of four existing indicators with empirical data. *ISPRS International Journal of Geo-Information*, *8*(5), 1–20. https://doi.org/10.3390/ijgi8050214

Rissel, C., Curac, N., Greenaway, M., & Bauman, A. (2012). Physical activity associated with public transport use-a review and modelling of potential benefits. *International Journal of Environmental Research and Public Health*, *9*(7), 2454–2478. https://doi.org/10.3390/ijerph9072454

Roever, L., Tse, G., & Biondi-Zoccai, G. (2019). Walking or cycling to work to prevent myocardial infarction: Hope or hype? *European Journal of Preventive Cardiology*, 2047487319880365. https://doi.org/10.1177/2047487319880365

Rosman, S. (2015). *Planning for cyclists.* https://dspace.library.uu.nl/handle/1874/320194

Schepers, P., Twisk, D., Fishman, E., Fyhri, A., & Jensen, A. (2017). The Dutch road to a high level of cycling safety. *Safety Science*, *92*, 264–273. https://doi.org/10.1016/j.ssci.2015.06.005

Strauss, J., Miranda-Moreno, L. F., & Morency, P. (2015). Mapping cyclist activity and injury risk in a network combining smartphone GPS data and bicycle counts. *Accident Analysis and Prevention*. https://doi.org/10.1016/j.aap.2015.07.014

Ton, D., Cats, O., Duives, D., & Hoogendoorn, S. (2017). How do people cycle in amsterdam, Netherlands?: Estimating cyclists' route choice determinants with GPS data from an Urban area. *Transportation Research Record*, *2662*(1), 75–82. https://doi.org/10.3141/2662-09

Ton, D., Duives, D. C., Cats, O., Hoogendoorn-Lanser, S., & Hoogendoorn, S. P. (2019). Cycling or walking? Determinants of mode choice in the Netherlands. *Transportation Research Part A: Policy and Practice*, *123*(August 2018), 7–23. https://doi.org/10.1016/j.tra.2018.08.023

Ton, D., Duives, D., Cats, O., & Hoogendoorn, S. (2018). Evaluating a data-driven approach for choice set identification using GPS bicycle route choice data from Amsterdam. *Travel Behaviour and Society*, *13*(February), 105–117. https://doi.org/10.1016/j.tbs.2018.07.001

7 Annexes

A. Data Action Steps ArcGIS Pro
B. Data Analysis Notebook (file formats: pdf & html)
C. Variables Scatter & Distribution Plot

Annex A - Data Action Steps ArcGIS Pro

Thesis: Cycling Distance and the Built Environment - Simon de Haas

Date: 09-12-2020

Data Actionmodel step	substep	Function	dataset	data		description what is done
				Input	Output	
0Aa	1	import	Fietstelweek	netwerk-2016-28992.shp	netwerk-2016-28992	Import links into the map
0Ab	2	import	Fietstelweek	routes2016.csv	routes.csv	Import routes2016 into the map
0A	3	Table to table	Fietstelweek	routes.csv	A0_routes_fietstelweek2016	Create from a csv a table
0B	4	Features to Features	Fietstelweek	netwerk-2016-28992	B0_links_fietstelweek2016	Copies the features in the shape file to new file for edditing
1A	5	Make Query Table	Fietstelweek	A0_routes_fietstelweek2016, B0_links_fietstelweek2016	QueryTable_A1_trips_in_segements	Makes a query table that connect the links to the route data
1A	6	Copy Features	Fietstelweek	QueryTable_A1_trips_in_segements	A1_trips_in_segements	Copies the data of the query table into a shape file
1A	7	select	Fietstelweek	A1_trips_in_segments	A1_forwards_trips_in_segments	Select the trip segments that are digitalised from start to end
1A	8	select	Fietstelweek	A1_trips_in_segments	A1_backwards_trips_in_segments	select the trip segments that are digitalised from end to start
1A	9	flip line	Fietstelweek	A1_backwards_trips_in_segments	A1_backwards_trips_in_segments	De lines are flipped from end to start --> start to end (name incorrect)
1A	10	merge	Fietstelweek	A1_forwards_trips_segments, A1_backwards_trips_in_segments	A1_sameDirected_trips_in_segments	Merge the data of the two datasets into one file
1A	11	pairwise dissolve	Fietstelweek	A1_sameDirected_trips_in_segments	A1_trips_combined_for_points	The segments are combined based on route Id, this layer is created to find the route id's of the trips that are having a start or end point within the borders of Amsterdam
1A	12	Add Geometry	Fietstelweek	A1_trips_combined_for_points	A1_trips_combined_for_points	The start, mid and end coordinates atributes are add to the trips
1B	13	XY Table To Point	Fietstelweek	A1_trips_combined_for_points	A1_start_points_of_trips	Created points from the start X and Y coordinate of all trips
1B	14	XY Table To Point	Fietstelweek	A1_trips_combined_for_points	A1_end_points_of_trips	Created points from the end X and Y coordinate of all trips
0D	15	import	cbs wijk en buurt kaart 2016	buurt_2016	D0_CBS_buurt_2016	import all neighbourhoods of the Netherlands
1A	16	Spatial Join	Fietstelweek, CBS	B1_start_points_of_trips, B1_end_points_of_trips, A1_sameDirected_trips_in_segments, D0_CBS_buurt_2016	A1_start_points_of_trips_gmbuNaam, A1_end_points_of_trips_gmbuNaam, A1_sameDirected_trips_inclGmBuNaam	Added the geography location via CBS data to the trip segment and trip origin and destination
1A	17	Join field	Fietstelweek	B1_start_points_of_trips_gmbuNaam, B1_end_points_of_trips_gmbuNaam, A1_sameDirected_trips_inclGmBuNaam	A1_sameDirected_trips_inclGmBuNaam	Joined the neighbourhood and municipality name from the starting and ending location with the trip segment data based route id
1D	18	Dissolve	cbs wijk en buurt kaart 2016	D0_cbs_buurt_2016	StudyArea_boundary	Created one boundary including the municipalities: Amstelveen, Amsterdam, Ouder-Amstel, Diemen
1A	19	Select	Fietstelweek	A1_sameDirected_trips_inclGmBuNaam	A1_sameDirected_trips_inclGmBuName_StudyArea	Selected all trip segments that are needed for trips going to and from the municipalities: Amstelveen, Amsterdam, Diemen and Ouder-Amstel. The municipalities selecting are based on the number of trips coming, going and within those municipalities, see Annex XX
1A	20	pairwise dissolve	Fietstelweek	A1_sameDirected_trips_inclGmBuName_StudyArea	A1_trips_TFW_studyArea	Created trips from the selection above
1A	21	Select by location	Fietstelweek	A1_trips_TFW_studyArea	A1_trips_TFW_studyArea	Selected only trips that are completely within the studyArea. All other trips are deleted (going from 31.121 to 30.010 trips)
1A	22	Calculate field	Fietstelweek	A1_trips_TFW_studyArea	A1_trips_TFW_studyArea	Added a Column with comWith_StudyArea 'yes'
1A	23	Join field	Fietstelweek	A1_trips_TFW_studyArea, A1_sameDirected_trips_inclGmBuName_StudyArea	A1_sameDirected_trips_inclGmBuName_StudyArea	Joined a column with Yes if the segment is needed for a trip completely within the studyArea
2A	24	Select	Fietstelweek	A1_sameDirected_trips_inclGmBuName_StudyArea	A2_tripSegments_within_studyArea	Only select trip segments that are needed and are completly within studyArea
2A	25	Copy Features, Delete Identical	Fietstelweek	A2_tripSegments_within_studyArea	A2_links_within_studyArea	Copied and deleted the identical trip segments to make the proces faster of adding data to link numbers faster
	26	Buffer	StudyArea	StudyArea_boundary	StudyArea_boundary_buffer600m	Created a bufferzone arround the studyArea to make it possible to analyse all trips even if they are close at the studyArea_boundary
1B	27	Clip	BGT, StudyArea	wegdeel_v, Boundary_studyArea_buffer600m	B1_BGT_wegdeel_studyArea	Clipped all objects from the BGT wegdeel into a new feature class
1B	28	Import	BGT	wegdeel_v_pdok	B1_BGT_wegdeel_verharding	Imported a gml file from pdok were surface material is included
1B	29	Clip	BGT, StudyArea	vegetatieobject_p, Boundary_studyArea_buffer600m	B1_BGT_vegetatieobjecten_StudyArea	Clipped all vegetatieobjecten within the studyArea
1E	30	Import	BAG	verblijfsobjecten	E1_BAG_verblijfsobjecten_studyArea	
2A	31	genererate points	Fietstelweek	A2_links_within_studyArea	A2_points_along_links_within_studyArea	Generated 10 points along with even wide distance between them
2A	32	Spatial join	Fietstelweek	A2_points_along_links_within_studyArea, B1_BGT_wegdeel_verharding	A2_points_along_links_within_studyArea_surfMatFun	Combined the closest surface material to the points along the links. Those combined will be used to see wat surface material and function the link possible had
2A	33	Summary Statistics	Fietstelweek	A2_points_along_links_within_studyArea_surfMatFun	A2_table_SurfaceMaterialFunction_links	Generated a table that include the max and minimum surface material and function of the links
2A	34	genererate points		A2_links_within_studyArea	A2_links_within_studyArea_points15meters	Generated a point every 15 meter on the link, including a point at the end of the line.
2A	35	Summarize Nearby	Fietstelweek, BGT	A2_pointsAlongLinks_studyArea_D15m, B1_BGT_BegroeidTerreindeel_studyArea, B1_BGT_waterdeel_studyArea	A2_pointsAlongLinks_studyArea_D15m_SN15m_Waterdeel, A2_pointsAlongLinks_studyArea_D15m_SN10m_Begroeiddeel	Searched for water nearby (15 meters) and vegitated area (10 meters) around the points
2A	36	Calculate field	Fietstelweek, BGT	A2_pointsAlongLinks_studyArea_D15m_SN15m_Waterdeel, A2_pointsAlongLinks_studyArea_D15m_SN10m_Begroeiddeel	A2_pointsAlongLinks_studyArea_D15m_SN15m_Waterdeel, A2_pointsAlongLinks_studyArea_D15m_SN10m_Begroeiddeel	Added a count column, if water or vegetation was nearby 1, if not 0
2A	37	Summarize statistics	Fietstelweek, BGT	A2_pointsAlongLinks_studyArea_D15m_SN15m_Waterdeel, A2_pointsAlongLinks_studyArea_D15m_SN10m_Begroeiddeel	A2_table_vegetatedpart_links, A2_table_waterPart_links	Summarized the data based on link nummer (going from points to links). Count if its nearby, sum the size of the surface and take the mean of the surface (bigger means its on average nearby)
2A	38	Summarize Nearby	Fietstelweek, BGT, BAG	B1_BGT_vegetatieobjecten_StudyArea, E1_BAG_verblijfsobjecten_studyArea	A2_links_studyArea_SN15m_TreesCount, A2_links_studyArea_SN30m_verblijfsobjecten	Counted trees (15 meters) around the links and verblijfsobjecten (30 meters) around the links
3A	39	Join Field	Fietstelweek, BGT, BAG	A2_links_within_studyArea, A2_table_waterPart_links, A2_table_vegetatedpart_links, A2_table_SurfaceMaterialFunction_links, A2_links_studyArea_SN15m_TreesCount, A2_links_studyArea_SN30m_verblijfsobjecten	A3_links_with_WaVeTrVbSf	Joined the data into one link file. The following has been add: Count_NearbyWater, Mean_NearbyWaterArea, Sum_NearbyWaterArea, count_NearbyVegetation, Mean_NearbyVegetationArea, Sum_NearbyVegetationArea, Number_of_Trees_Along, Num_ - bijeenkomstfunctie, celfunctie, gezondheidszorgfunctie, industriefunctie, kantoorfunctie, logiesfunctie, onderwijsfunctie, overige_gebruiksfunctie, sportfunctie, winkelfunctie, woonfunctie, SUM_SurfaceVerblijf, Total_num_points, Max_SurfaceMat, MAX_function
4A	40	Join field	Fietstelweek	A3_links_with_WaVeTrVbSf	A4_tripSegments_with_dataWaVeTrVbSf	Joined the data of the links with the segments of the trips
4A	41	Table to table	Fietstelweek	A4_tripSegments_with_dataWaVeTrVbSf	dataset_cdbe_v07.csv	Exported the data into a csv file

Annex B - Data Analysis Notebook

Thesis: Cycling Distance and the Built Environment - Simon de Haas

Date: 09-12-2020

This notebook is created with Jupyter Notebook. The following sections explain the actions that are taken, the code that is used and the output generated by this code. Those notebooks are normally not shared in the file format: PDF. The HTML file that has been add to the annex of this thesis shows this Notebook in a better form.

Sections

The notebook consist of the following sections

- 1 Libraries import
- 2 Data loading
- 3 Dummy variables
- 4 Making trips (groupby)
- 5 Final variables
- 6 Check Multicollinearity & describe dataset
- 7 Building models

Section 01: Importing libraries & global functions

In [1]:
```
#Import the needed packages
import pandas as pd
import numpy as np
import seaborn as sns
import pingouin as pg
import matplotlib.pyplot as plt

import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor


from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split

from scipy import stats

from mlxtend.feature_selection import SequentialFeatureSelector as SFS
from mlxtend.plotting import plot_sequential_feature_selection as plot_sfs

%matplotlib inline
```

```
C:\Users\simon\anaconda3\envs\python37\lib\site-packages\outdated\utils.py:18: OutdatedPackageWarning: Th
e package pingouin is out of date. Your version is 0.3.7, the latest is 0.3.8.
Set the environment variable OUTDATED_IGNORE=1 to disable these warnings.
  **kwargs
```

In [2]:
```
#Function create model has been made to make the process of developing models faster.

def createmodel(X, Y):
    from sklearn.model_selection import train_test_split
    print('-'*80)
    # Split X and y into X_
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.20, random_state=1)

    # create a Linear Regression model object
    reg = LinearRegression()

    # pass through the X_train & y_train data set
    reg.fit(X_train, y_train)

    # let's grab the coefficient of our model and the intercept
    intercept = reg.intercept_[0]
    coefficent = reg.coef_[0][0]

    print("['" + str(Y.columns[0]) + "'] vs. " + str(X.columns.values.tolist()))

    print("\nThe intercept for our model is {:.4}".format(intercept))


    print("The Coefficient for our model is {:.2}\n".format(coefficent))

    # define our input
    X2 = sm.add_constant(X)

    # create a OLS model
    model = sm.OLS(Y, X2)

    # fit the data
    est = model.fit()


    print(est.summary())
    print('-'*80)
    return est
```

Section 02: Importing dataset and data cleaning

In this section the datafile (csv) exported from ArcGIS Pro loaded and cleaned. This file consist of 2,057,144 rows of trip segments and 51 columns. The columns are renamed for easier understanding and faster writing.

In [3]:
```
#import dataset
df_raw = pd.read_csv('dataset_cdbe_v07.csv', sep =';')

#Preview of the dataset and the first five rows
print("The number of rows and columns in the datafile: " + str(df_raw.shape))
display(df_raw.head())
```

```
The number of rows and columns in the datafile: (2057144, 60)
```

	OBJECTID	Join_Count	TARGET_FID	A0_routes_fietst	A0_routes_fiet_1	A0_routes_fiet_2	A0_routes_fiet_3	A0_routes_fiet_4
0	1	1	121	1106971	NaN	f	45873	19,381526455465600
1	2	1	122	1106972	NaN	f	45873	19,385452327285599
2	3	1	123	1106984	NaN	f	45873	15,660230374852199
3	4	1	124	1106985	NaN	f	45873	13,444885121840301
4	5	1	125	1106855	NaN	f	45873	13,448102483174299

5 rows × 60 columns

2.1 Changing column names

In [4]:
```
#changing column names
#Colum name dictionary
column_names = { 'OBJECTID':'objectid',
                 'Join_Count':'join_count',
                 'TARGET_FID':'target_fid',
                 'A0_routes_fietst':'routes_linknum',
                 'A0_routes_fiet_1':'routes_month',
                 'A0_routes_fiet_2':'routes_direction',
                 'A0_routes_fiet_3':'routes_routeid',
                 'A0_routes_fiet_4':'routes_speed',
                 'A0_routes_fiet_5':'routes_hour',
                 'A0_routes_fiet_6':'routes_wkdag',
                 'A0_routes_fiet_7':'routes_year',
                 'B0_links_fietste':'links_highway',
                 'B0_links_fiets_1':'links_intensity_01',
                 'B0_links_fiets_2':'links_intensity',
                 'B0_links_fiets_3':'links_linknum',
                 'B0_links_fiets_4':'links_objectid',
                 'B0_links_fiets_5':'links_speed_r',
                 'B0_links_fiets_6':'links_source',
                 'B0_links_fiets_7':'links_speed',
                 'B0_links_fiets_8':'links_target',
                 'BU_CODE':'links_buc',
                 'BU_NAAM':'links_bun',
                 'GM_NAAM':'links_gmn',
                 'BU_NAAM_start':'trip_bun_st',
                 'GM_NAAM_start':'trip_gmn_st',
                 'BU_NAAM_end':'trip_bun_en',
                 'GM_NAAM_end':'trip_gmn_en',
                 'CompWith_StudyAr':'compwith_s',
                 'Shape_Length':'segm_length',
                 'count_NearbyWate':'segm_c_water_d15',
                 'MEAN_NearbyWater':'segm_m_watera_d15',
                 'SUM_NearbyWaterA':'segm_s_watera_d15',
                 'count_NearbyVega':'segm_c_vega_d15',
                 'SUM_NearbyVegeta':'segm_s_vegaa_d15',
                 'MEAN_NearbyVeget':'segm_m_vegaa_d15',
                 'Number_of_Trees_':'segm_n_trees',
                 'Total_Num_points':'segm_t_func',
                 'Num_bijeenkomstf':'segm_n_meetf',
                 'Num_celfunctie':'segm_n_celf',
                 'Num_gezondheidsz':'segm_n_ghzf',
                 'Num_industriefun':'segm_n_indusf',
                 'Num_kantoorfunct':'segm_n_officef',
                 'Num_logiesfuncti':'segm_n_logiesf',
                 'Num_onderwijsfun':'segm_n_eduf',
                 'Num_overige_gebr':'segm_n_othf',
                 'Num_sportfunctie':'segm_n_sportf',
                 'Num_winkelfuncti':'segm_n_shopf',
                 'Num_woonfunctie':'segm_n_livef',
                 'SUM_SurfaceBuild':'segm_s_functionarea',
                 'MAX_function':'segm_r_function',
                 'MAX_surfaceMat':'segm_r_surfm',
                 'Points_along_lin':'segm_n_pointsalong_d15',
                 'SUM_NearbyWate_1':'segm_c_water_d5',
                 'Points_along_l_1':'segm_n_pointsalong_d5',
                 'SUM_NearbyVegaAr':'segm_c_vega_d5',
                 'SUM_NearbytreesD':'segm_c_trees_d5',
                 'START_X':'start_x',
                 'START_Y':'start_y',
                 'END_X':'end_x',
                 'END_Y':'end_y'
                }

#Executing renaming
df = df_raw.rename(columns = column_names)
```

In [5]:
```
#creating the dataframe with only the nescarry data
df_a = df[['objectid','routes_routeid','links_linknum','segm_length','trip_bun_st',
       'trip_gmn_st', 'trip_bun_en', 'trip_gmn_en',
          'links_highway', 'segm_n_pointsalong_d15','segm_c_water_d15',
       'segm_c_vega_d15', 'segm_n_trees',
       'segm_t_func', 'segm_n_meetf', 'segm_n_celf', 'segm_n_ghzf',
       'segm_n_indusf', 'segm_n_officef', 'segm_n_logiesf', 'segm_n_eduf',
       'segm_n_othf', 'segm_n_sportf', 'segm_n_shopf', 'segm_n_livef',
       'segm_s_functionarea', 'segm_r_function', 'segm_r_surfm',
       'segm_n_pointsalong_d5', 'segm_c_water_d5', 'segm_c_vega_d5',
       'segm_c_trees_d5', 'start_x', 'start_y', 'end_x', 'end_y']]

#Set the index to objectid
df_a.index = df_a['objectid']
df_a = df_a.drop('objectid', axis = 1)
```

2.2 Changing the datatype of the columns

In this section, the datatype has been changed from object to float and strings.

In [6]:
```
#Showing datatype of the columns
df_a.dtypes
```

Out[6]:
```
routes_routeid              int64
links_linknum              object
segm_length                object
trip_bun_st                object
trip_gmn_st                object
trip_bun_en                object
trip_gmn_en                object
links_highway              object
segm_n_pointsalong_d15      int64
segm_c_water_d15           object
segm_c_vega_d15            object
segm_n_trees                int64
segm_t_func                 int64
segm_n_meetf               object
segm_n_celf                object
segm_n_ghzf                object
segm_n_indusf              object
segm_n_officef             object
segm_n_logiesf             object
segm_n_eduf                object
segm_n_othf                object
segm_n_sportf              object
segm_n_shopf               object
segm_n_livef               object
segm_s_functionarea        object
segm_r_function            object
segm_r_surfm               object
segm_n_pointsalong_d5      object
segm_c_water_d5           float64
segm_c_vega_d5             object
segm_c_trees_d5            object
start_x                    object
start_y                    object
end_x                      object
end_y                      object
dtype: object
```

In [7]:
```
#Splitting the data at the Comma and changing the decimal symbol.
df_a['links_linknum'] = df_a['links_linknum'].str.split(',').str[0]
df_a['segm_length'] = df_a['segm_length'].str.replace(',','.')
df_a['segm_c_water_d15'] = df_a['segm_c_water_d15'].str.split(',').str[0]
df_a['segm_c_vega_d15'] = df_a['segm_c_vega_d15'].str.split(',').str[0]
df_a['segm_n_meetf'] = df_a['segm_n_meetf'].str.split(',').str[0]
df_a['segm_n_celf'] = df_a['segm_n_celf'].str.split(',').str[0]
df_a['segm_n_ghzf'] = df_a['segm_n_ghzf'].str.split(',').str[0]
df_a['segm_n_indusf'] = df_a['segm_n_indusf'].str.split(',').str[0]
df_a['segm_n_officef'] = df_a['segm_n_officef'].str.split(',').str[0]
df_a['segm_n_logiesf'] = df_a['segm_n_logiesf'].str.split(',').str[0]
df_a['segm_n_eduf'] = df_a['segm_n_eduf'].str.split(',').str[0]
df_a['segm_n_othf'] = df_a['segm_n_othf'].str.split(',').str[0]
df_a['segm_n_sportf'] = df_a['segm_n_sportf'].str.split(',').str[0]
df_a['segm_n_shopf'] = df_a['segm_n_shopf'].str.split(',').str[0]
df_a['segm_n_livef'] = df_a['segm_n_livef'].str.split(',').str[0]
df_a['segm_s_functionarea'] = df_a['segm_s_functionarea'].str.split(',').str[0]
df_a['segm_n_pointsalong_d5'] = df_a['segm_n_pointsalong_d5'].str.split(',').str[0]
df_a['segm_c_vega_d5'] = df_a['segm_c_vega_d5'].str.split(',').str[0]
df_a['segm_c_trees_d5'] = df_a['segm_c_trees_d5'].str.split(',').str[0]
df_a['start_x'] = df_a['start_x'].str.replace(',','.')
df_a['start_y'] = df_a['start_y'].str.replace(',','.')
df_a['end_x'] = df_a['end_x'].str.replace(',','.')
df_a['end_y'] = df_a['end_y'].str.replace(',','.')
```

In [8]:
```
#Checking if the previous data processing went well
df_a.head()
```

Out[8]:

objectid	routes_routeid	links_linknum	segm_length	trip_bun_st	trip_gmn_st	trip_bun_en	trip_gmn_en	links_highway	se
1	45873	1106971	27.970244681051227	Spuistraat Noord	Amsterdam	Plantage	Amsterdam	unclassified	
2	45873	1106972	33.324512957758223	Spuistraat Noord	Amsterdam	Plantage	Amsterdam	unclassified	
3	45873	1106984	107.799887481712986	Spuistraat Noord	Amsterdam	Plantage	Amsterdam	unclassified	
4	45873	1106985	5.072759267251489	Spuistraat Noord	Amsterdam	Plantage	Amsterdam	unclassified	
5	45873	1106855	22.054028200640737	Spuistraat Noord	Amsterdam	Plantage	Amsterdam	cycleway	

5 rows × 35 columns

In [9]:
```
#Converting dictionary to change datatype from objects to floats and strings.
convert_dict = {'routes_routeid':int,
                'links_linknum':float,
                'segm_length':float,
                'trip_bun_st':str,
                'trip_gmn_st':str,
                'trip_bun_en':str,
                'trip_gmn_en':str,
                'links_highway':str,
                'segm_n_pointsalong_d15':float,
                'segm_c_water_d15':float,
                'segm_c_vega_d15':float,
                'segm_n_trees':float,
                'segm_t_func':float,
                'segm_n_meetf':float,
                'segm_n_celf':float,
                'segm_n_ghzf':float,
                'segm_n_indusf':float,
                'segm_n_officef':float,
                'segm_n_logiesf':float,
                'segm_n_eduf':float,
                'segm_n_othf':float,
                'segm_n_sportf':float,
                'segm_n_shopf':float,
                'segm_n_livef':float,
                'segm_s_functionarea':float,
                'segm_r_function':str,
                'segm_r_surfm':str,
                'segm_n_pointsalong_d5':float,
                'segm_c_water_d5':float,
                'segm_c_vega_d5':float,
                'segm_c_trees_d5':float,
                'start_x':float,
                'start_y':float,
                'end_x':float,
                'end_y':float
               }

df_a = df_a.astype(convert_dict)

df_a = df_a.reset_index()

#Checking if datatypes have been changed. The data types are correctly changed.
# df_a.dtypes  // The line has ben outcommented to keep the notebook readable.
```

Section 03: Categorising data

In this section, the quantification of the elements have been finalised and combined with the bicycle trips.

3.1 Mixture of functions and function density

In [10]:
```
#Calculating the total number of functions along every segment.

df_a['segm_t_func'] = (df_a['segm_n_meetf'] + df_a['segm_n_celf'] +
                       df_a['segm_n_ghzf'] + df_a['segm_n_indusf'] +
                       df_a['segm_n_officef'] + df_a['segm_n_logiesf'] +
                       df_a['segm_n_eduf'] + df_a['segm_n_othf'] +
                       df_a['segm_n_sportf'] + df_a['segm_n_shopf'] +
                       df_a['segm_n_livef'])
```

In [11]:
```
df_a['segm_m_meetf']  = np.where(df_a['segm_n_meetf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_celf'] = np.where(df_a['segm_n_celf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_ghzf'] =  np.where(df_a['segm_n_ghzf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_indusf'] =  np.where(df_a['segm_n_indusf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_officef'] = np.where(df_a['segm_n_officef'] > 0, df_a['segm_length'], 0)
df_a['segm_m_logiesf'] = np.where(df_a['segm_n_logiesf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_eduf'] =   np.where(df_a['segm_n_eduf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_othf'] =  np.where(df_a['segm_n_othf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_sportf'] =  np.where(df_a['segm_n_sportf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_shopf'] =   np.where(df_a['segm_n_shopf'] > 0, df_a['segm_length'], 0)
df_a['segm_m_livef'] = np.where(df_a['segm_n_livef'] > 0, df_a['segm_length'], 0)
```

In [12]:
```
#function created to check if a function is present or not. If present its value becomes 1, otherwise 0.
#The outcome of the function is a added up number with the number of different function along the segment.

def difffunc(a,b,c,d,e,f,g,h,i,j,k):
    a = np.where(a > 0, 1, 0)
    b = np.where(b > 0, 1, 0)
    c = np.where(c > 0, 1, 0)
    d = np.where(d > 0, 1, 0)
    e = np.where(e > 0, 1, 0)
    f = np.where(f > 0, 1, 0)
    g = np.where(g > 0, 1, 0)
    h = np.where(h > 0, 1, 0)
    i = np.where(i > 0, 1, 0)
    j = np.where(j > 0, 1, 0)
    k = np.where(k > 0, 1, 0)

    numfunc = a + b + c + d + e + f + g + h + i + j + k
    return numfunc
```

In [13]:
```
dft = df_a
r = 30
pi = 3.141592653589793

#With the use of the function difffunc, the number of unique functions along every segment has been create
d.
dft['segm_func_diff'] = difffunc(dft['segm_n_meetf'],
       dft['segm_n_celf'], dft['segm_n_ghzf'], dft['segm_n_indusf'],dft[ 'segm_n_officef'],
       dft['segm_n_logiesf'], dft['segm_n_eduf'], dft['segm_n_othf'], dft['segm_n_sportf'],
       dft['segm_n_shopf'], dft['segm_n_livef'])

#The mixture of functions has been multiplied by the length of the segment. This is needed to
#calculated the average number of unique functions along the trip.

dft['segm_func_diff'] = dft['segm_func_diff'] * dft['segm_length']

#Calculating the function density along every segment.
dft['segm_func_dens'] = (((dft['segm_t_func'] / ((dft['segm_length']*(2*r))+(0.5*pi*(r^2))))*10000)*dft['s
egm_length'])

#showing the created columns and the first five rows.
display(dft[['segm_func_dens', 'segm_t_func', 'segm_length']].head())

df_a = dft
```

	segm_func_dens	segm_t_func	segm_length
0	5197.128094	32.0	27.970245
1	3913.905932	24.0	33.324513
2	8442.590459	51.0	107.799887
3	3494.961180	24.0	5.072759
4	806.525791	5.0	22.054028

3.2 Surface material

In [14]:
```
#Changing the categorical data of surface material into five unique columns, with the segment length as va
lue.

dft = df_a

dft['segm_surfm_transitie'] = np.where(dft['segm_r_surfm'] == 'transitie', dft['segm_length'],0)
dft['segm_surfm_openverh'] = np.where(dft['segm_r_surfm'] == 'open verharding', dft['segm_length'],0)
dft['segm_surfm_geslotenverh'] = np.where(dft['segm_r_surfm'] == 'gesloten verharding', dft['segm_length'
],0)
dft['segm_surfm_halfverh'] = np.where(dft['segm_r_surfm'] == 'half verhard', dft['segm_length'],0)
dft['segm_surfm_onverh'] = np.where(dft['segm_r_surfm'] == 'onverhard', dft['segm_length'],0)

#Showing the created columns. Only one of the five columns may have a value greater than zero.
display(dft[['segm_surfm_transitie', 'segm_surfm_openverh',
             'segm_surfm_geslotenverh', 'segm_surfm_halfverh',
             'segm_surfm_onverh' ]])

df_a = dft
```

	segm_surfm_transitie	segm_surfm_openverh	segm_surfm_geslotenverh	segm_surfm_halfverh	segm_surfm_onverh
0	0.0	0.000000	27.970245	0.0	0.0
1	0.0	0.000000	33.324513	0.0	0.0
2	0.0	0.000000	107.799887	0.0	0.0
3	0.0	0.000000	5.072759	0.0	0.0
4	0.0	0.000000	22.054028	0.0	0.0
...	...	...	...	...	...
2057139	0.0	0.000000	3.935699	0.0	0.0
2057140	0.0	0.000000	2.852945	0.0	0.0
2057141	0.0	29.094554	0.000000	0.0	0.0
2057142	0.0	18.894123	0.000000	0.0	0.0
2057143	0.0	51.214963	0.000000	0.0	0.0

2057144 rows × 5 columns

3.3 Road type

In [15]:
```
dft = df_a

road_func_list =  [
                    ['segm_linkfunc_cycle','links_highway','cycleway'],
                    ['segm_linkfunc_unclas','links_highway','unclassified'],
                    ['segm_linkfunc_pedes','links_highway','pedestrian'],
                    ['segm_linkfunc_secon','links_highway','secondary'],
                    ['segm_linkfunc_footway','links_highway','footway'],
                    ['segm_linkfunc_empty','links_highway',''],
                    ['segm_linkfunc_path','links_highway','path'],
                    ['segm_linkfunc_residen','links_highway','residential'],
                    ['segm_linkfunc_tertiar','links_highway','tertiary'],
                    ['segm_linkfunc_prim','links_highway','primary'],
                    ['segm_linkfunc_steps','links_highway','steps'],
                    ['segm_linkfunc_service','links_highway','service'],
                    ['segm_linkfunc_track','links_highway','track'],
                    ['segm_linkfunc_livestr','links_highway','living_street'],
                    ['segm_linkfunc_prim_link','links_highway','primary_link'],
                    ['segm_linkfunc_bridleway','links_highway','bridleway'],
                    ['segm_linkfunc_platform','links_highway','platform'],
                    ['segm_linkfunc_bus_stop','links_highway','bus_stop'],
                    ['segm_linkfunc_construction','links_highway','construction'],
                    ['segm_linkfunc_secon_link','links_highway','secondary_link'],
                    ['segm_linkfunc_tert_link','links_highway','tertiary_link'],
                    ['segm_linkfunc_elevator','links_highway', 'elevator'],
                    ['segm_linkfunc_road','links_highway', 'road']
                ]


length = len(road_func_list)
count = 0

while count < length:
    dft[road_func_list[count][0]] = np.where(dft[road_func_list[count][1]] == road_func_list[count][2], df
t['segm_length'],0)
    count += 1

dft.head()

df_a = dft
```

In [16]:
```
#creating column that indicates if a segment is shorter than 10 meters or not.
df_a['segm_short'] = np.where(df_a['segm_length'] < 10, 1,0)
```

Section 04: Creating trips

The following code groups the rows based on routeid, to create trip. Columns with only NaN values are deleted.

In [17]:

```
#Group by routes_routeid
agg_dict = {
          #General
                'objectid':['count'],
                'segm_length':['sum','mean'],
                'segm_short':['sum'],
                'trip_bun_st':['max'],
                'trip_gmn_st':['max'],
                'trip_bun_en':['max'],
                'trip_gmn_en':['max'],
                'links_highway':['max'],
                'segm_n_pointsalong_d5':['sum'],
                'segm_n_pointsalong_d15':['sum'],


          #Trees, water, vegatation
                'segm_n_trees':['sum'],
                'segm_c_trees_d5':['sum'],

                'segm_c_water_d5':['sum'],

                'segm_c_vega_d5':['sum'],



          #Function buildings
                'segm_m_meetf':['sum'],
                'segm_m_celf':['sum'],
                'segm_m_ghzf':['sum'],
                'segm_m_indusf':['sum'],
                'segm_m_officef' :['sum'],
                'segm_m_logiesf':['sum'],
                'segm_m_eduf':['sum'],
                'segm_m_othf':['sum'],
                'segm_m_sportf':['sum'],
                'segm_m_shopf':['sum'],
                'segm_m_livef':['sum'],


                'segm_func_diff':['sum'],
                'segm_func_dens':['sum'],

                'segm_t_func':['sum'],

          #Surface material
                'segm_surfm_transitie':['sum'],
                'segm_surfm_openverh':['sum'],
                'segm_surfm_geslotenverh':['sum'],
                'segm_surfm_halfverh':['sum'],
                'segm_surfm_onverh':['sum'],

          #Road function
                'segm_linkfunc_cycle':['sum'],
                'segm_linkfunc_unclas':['sum'],
                'segm_linkfunc_pedes':['sum'],
                'segm_linkfunc_secon':['sum'],
                'segm_linkfunc_footway':['sum'],
                'segm_linkfunc_empty':['sum'],
                'segm_linkfunc_path':['sum'],
                'segm_linkfunc_residen':['sum'],
                'segm_linkfunc_tertiar':['sum'],
                'segm_linkfunc_prim':['sum'],
                'segm_linkfunc_steps':['sum'],
                'segm_linkfunc_service':['sum'],
                'segm_linkfunc_track':['sum'],
                'segm_linkfunc_livestr':['sum'],
                'segm_linkfunc_prim_link':['sum'],
                'segm_linkfunc_bridleway':['sum'],
                'segm_linkfunc_platform':['sum'],
                'segm_linkfunc_bus_stop':['sum'],
                'segm_linkfunc_construction':['sum'],
                'segm_linkfunc_secon_link':['sum'],
                'segm_linkfunc_tert_link':['sum'],
                'segm_linkfunc_elevator':['sum'],
                'segm_linkfunc_road':['sum'],

        #Start- endlocation
                'start_x':['max'],
                'start_y':['max'],
                'end_x':['max'],
                'end_y':['max']
```

```
            }

df_g = df_a.groupby(['routes_routeid']).agg(agg_dict)
df_g.columns = df_g.columns.map('_'.join).str.strip('_')
df_g = df_g.rename(columns = {'objectid_count':'num_of_segments', 'segm_length_sum':'trip_length'})
df_g = df_g.round(4)

display(df_g.head())
```

routes_routeid	num_of_segments	trip_length	segm_length_mean	segm_short_sum	trip_bun_st_max	trip_gmn_st_max	trip_bun_en
3	122	6470.3288	53.0355	20	Lootsbuurt	Amsterdam	Te
6	20	1364.2225	68.2111	4	Woon- en Groengebied Sloterdijk	Amsterdam	E
8	19	765.5621	40.2927	7	Vondelpark Oost	Amsterdam	Vondelpar
21	41	1930.4358	47.0838	8	Leliegracht e.o.	Amsterdam	Oosterdoks
22	129	4990.9536	38.6896	37	Oosterdokseiland	Amsterdam	Stationsple

5 rows × 61 columns

Section 05: Calculating trip percentages

5.1 Calculating percentages

The percentage along trees, percentage along waterbodies, percentage along vegetation, average number of unique functions along the trip, average function density along the trip, percentage over closed (smooth) surface material, percentage over cycle paths and average distance between crossings.

In [18]:
```
df_perc = df_g

#percentage green and water and trees nearby along the trip
df_perc['trip_p_trees'] = ((df_perc['segm_c_trees_d5_sum'] / df_perc['segm_n_pointsalong_d5_sum']))
df_perc['trip_p_water'] = ((df_perc['segm_c_water_d5_sum'] / df_perc['segm_n_pointsalong_d5_sum']) )
df_perc['trip_p_vega'] = ((df_perc['segm_c_vega_d5_sum'] / df_perc['segm_n_pointsalong_d5_sum']))

#Calculating average number of unique functions along the trip
df_perc['trip_p_funcdiff'] = (((df_perc['segm_func_diff_sum'] / df_perc['trip_length'])))

#Calculating the function density (built environment density) along the trip
df_perc['trip_p_funcdens'] = (((df_perc['segm_func_dens_sum'] / df_perc['trip_length'])))

#Calculating the percentage of the trip going over closed (smooth) surface material
df_perc['trip_surfm_geslotenverh'] = (((df_perc['segm_surfm_geslotenverh_sum'] / df_perc['trip_length'])))

#Calculating the percentage of the trip going over cycle paths
df_perc['trip_linkfunc_cycle'] = (((df_perc['segm_linkfunc_cycle_sum'] / df_perc['trip_length'])))

#Calculating the average distance between crossings of every trip.
df_perc['trip_d_cross'] = ((df_perc['trip_length'] / (df_perc['num_of_segments'] - df_perc['segm_short_su
m'])))
```

In [19]:
```
#Calculating the Euclidean distance
df_perc['crow_length'] = (  np.sqrt(
                                    (abs((df_perc['start_x_max'] - df_perc['end_x_max']))**2 +
                                    abs((df_perc['start_y_max'] - df_perc['end_y_max']))**2)
                                )
                          )
#Calculating the percentage of detour
df_perc['detour_perc'] = ((np.round((df_perc['trip_length'] / df_perc['crow_length']),8)-1) *100)

#Calculating the detour distance
df_perc['detour_meters'] = ((np.round((df_perc['trip_length'] - df_perc['crow_length']),8)))
```

5.2 Creating final dataframe

The dataframe: df_m is used to develop the models and only has the columns that are needed.

In [20]:
```
#creating data frame and select the necessary columns.
df_m = df_perc[['trip_length',
                'detour_perc',
                'detour_meters',
                'crow_length',
                'trip_d_cross',
                'trip_gmn_st_max',
                'trip_gmn_en_max',
                'trip_p_trees',
                'trip_p_water',
                'trip_p_vega',
                'trip_surfm_geslotenverh',
                'trip_linkfunc_cycle',
                'trip_p_funcdens',
                'trip_p_funcdiff',
               ]]

#Drop trips that have a higher detour_perc of hunderd
df_m = df_m.loc[df_perc['detour_perc'] < 100]
df_m = df_m.drop(['detour_perc'], axis = 1)

display(df_m.shape)
df_m.head()
```

```
(30137, 13)
```

Out[20]:

routes_routeid	trip_length	detour_meters	crow_length	trip_d_cross	trip_gmn_st_max	trip_gmn_en_max	trip_p_trees	trip_p_water
3	6470.3288	2955.924432	3514.404368	63.434596	Amsterdam	Amsterdam	0.676113	0.311741
6	1364.2225	370.248768	993.973732	85.263906	Amsterdam	Amsterdam	0.885496	0.419847
8	765.5621	144.332550	621.229550	63.796842	Amsterdam	Amsterdam	0.800000	0.103448
21	1930.4358	531.172633	1399.263167	58.498055	Amsterdam	Amsterdam	0.421918	0.709589
25	2329.9306	554.600084	1775.330516	75.159052	Amsterdam	Amsterdam	0.774775	0.693694

Section 06: Check Multicollinearity & describe dataset

6.1 Correlation matrix

In [21]:
```
# calculate the correlation matrix
corr = df_m.corr()

# display the correlation matrix
corr_matrix = df_m.rcorr(stars=True)
display(corr_matrix)
```

	trip_length	detour_meters	crow_length	trip_d_cross	trip_p_trees	trip_p_water	trip_p_vega	trip_surfm_ge
trip_length	-	***	***	***	***	***	***	
detour_meters	0.877	-	***	***	***	***	***	
crow_length	0.986	0.784	-	***	***	***	***	
trip_d_cross	0.152	0.122	0.154	-	***	***	***	
trip_p_trees	-0.118	-0.116	-0.111	-0.197	-	***	***	
trip_p_water	0.087	0.103	0.076	0.313	-0.157	-	*	
trip_p_vega	0.189	0.183	0.18	0.272	-0.277	-0.013	-	
trip_surfm_geslotenverh	0.162	0.116	0.169	0.076	-0.187	-0.067	0.183	
trip_linkfunc_cycle	0.128	0.071	0.141	0.132	-0.17	-0.153	0.308	
trip_p_funcdens	-0.216	-0.196	-0.21	-0.345	0.317	-0.177	-0.712	
trip_p_funcdiff	-0.203	-0.193	-0.195	-0.281	0.347	-0.057	-0.841	

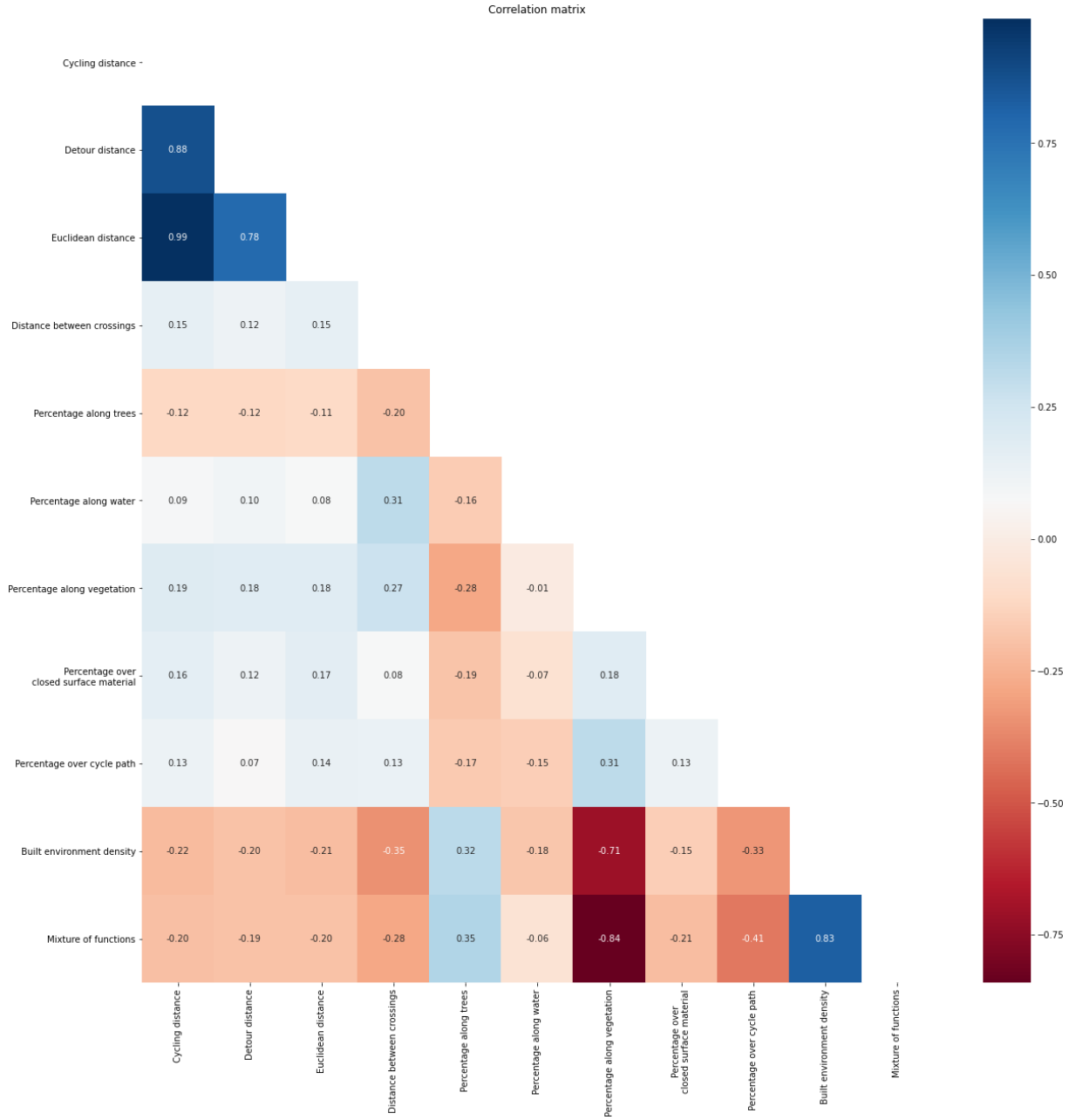
In [22]:

```
#Showing correlation matrix
mask = np.triu(np.ones_like(corr, dtype=np.bool))
plt.subplots(figsize=(20,20))
labels=['Cycling distance',
        'Detour distance',
        'Euclidean distance',
        'Distance between crossings',
        'Percentage along trees',
        'Percentage along water',
        'Percentage along vegetation',
        'Percentage over\nclosed surface material',
        'Percentage over cycle path',
        'Built environment density',
        'Mixture of functions',
       ]


ax = sns.heatmap(corr, xticklabels=(labels), mask=mask, yticklabels=(labels), cmap='RdBu', annot=True, fmt
=".2f")

ax.set_title('Correlation matrix')


plt.savefig('correlation matrix.png')
```

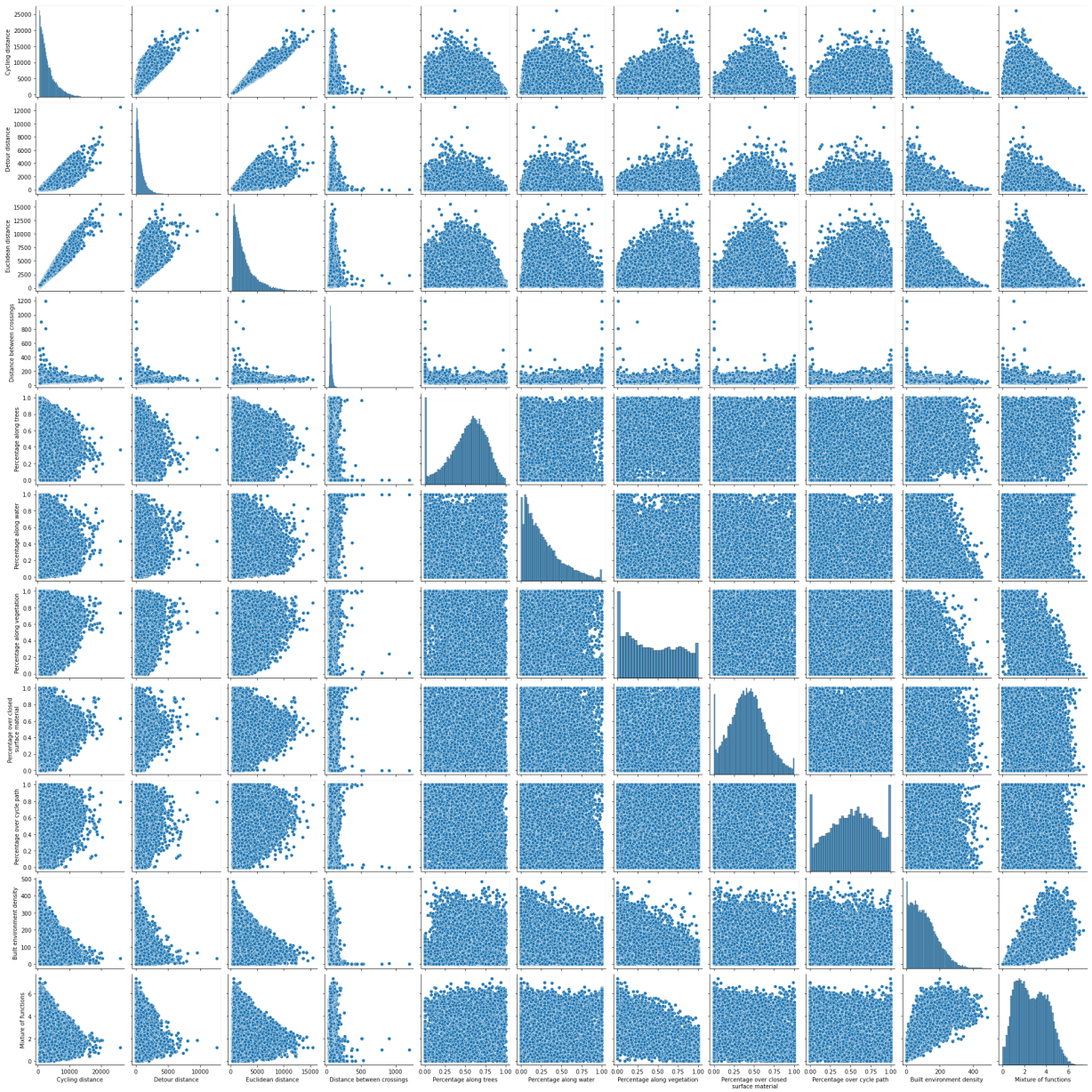


In [23]:
```
#Creation of the scatter and distribution plot.
df_pairplot = df_m

df_pairplot = df_pairplot.drop(['trip_gmn_st_max', 'trip_gmn_en_max'], axis = 1)

df_pairplot = df_pairplot.rename(columns={'trip_length':'Cycling distance',
                                          'detour_meters':  'Detour distance',
                                          'crow_length':'Euclidean distance',
                                          'trip_d_cross':'Distance between crossings',
                                          'trip_p_trees':   'Percentage along trees',
                                          'trip_p_water': 'Percentage along water',
                                          'trip_p_vega':'Percentage along vegetation',
                                         'trip_surfm_geslotenverh': 'Percentage over closed\n surface ma
terial',
                                         'trip_linkfunc_cycle':'Percentage over cycle path',
                                         'trip_p_funcdens':     'Built environment density',
                                         'trip_p_funcdiff':     'Mixture of functions',
                                })

pp = sns.pairplot(df_pairplot, aspect = 1)
# pp.savefig("19 scatter matrix.png")
```



6.2 Data description

In [24]:
```
# get the summary
desc_df = df_m.describe()

# add the standard deviation metric
desc_df.loc['+3_std'] = desc_df.loc['mean'] + (desc_df.loc['std'] * 3)
desc_df.loc['-3_std'] = desc_df.loc['mean'] - (desc_df.loc['std'] * 3)

# display it
desc_df
```

Out[24]:

	trip_length	detour_meters	crow_length	trip_d_cross	trip_p_trees	trip_p_water	trip_p_vega	trip_surfm_geslotenver
count	30137.000000	30137.000000	30137.000000	30137.000000	30137.000000	30137.000000	30137.000000	30137.00000
mean	3241.287510	708.816148	2532.471363	67.087490	0.530181	0.274329	0.439853	0.42138
std	2589.946760	701.340114	2003.359075	22.682391	0.232713	0.226952	0.305579	0.21693
min	500.032300	-0.000016	261.103095	29.625650	0.000000	0.000000	0.000000	0.00000
25%	1327.386300	230.977058	1054.604347	54.956120	0.390909	0.094463	0.161943	0.27047
50%	2441.291100	499.157543	1909.709540	62.819393	0.563218	0.215370	0.411765	0.41868
75%	4349.706300	953.699660	3375.817100	73.458278	0.700258	0.395659	0.707006	0.56613
max	26175.794400	12506.354415	15504.767216	1198.540950	1.000000	1.000000	1.000000	1.00000
+3_std	11011.127790	2812.836490	8542.548588	135.134663	1.228319	0.955186	1.356588	1.07219
-3_std	-4528.552769	-1395.204194	-3477.605863	-0.959684	-0.167957	-0.406528	-0.476883	-0.22942

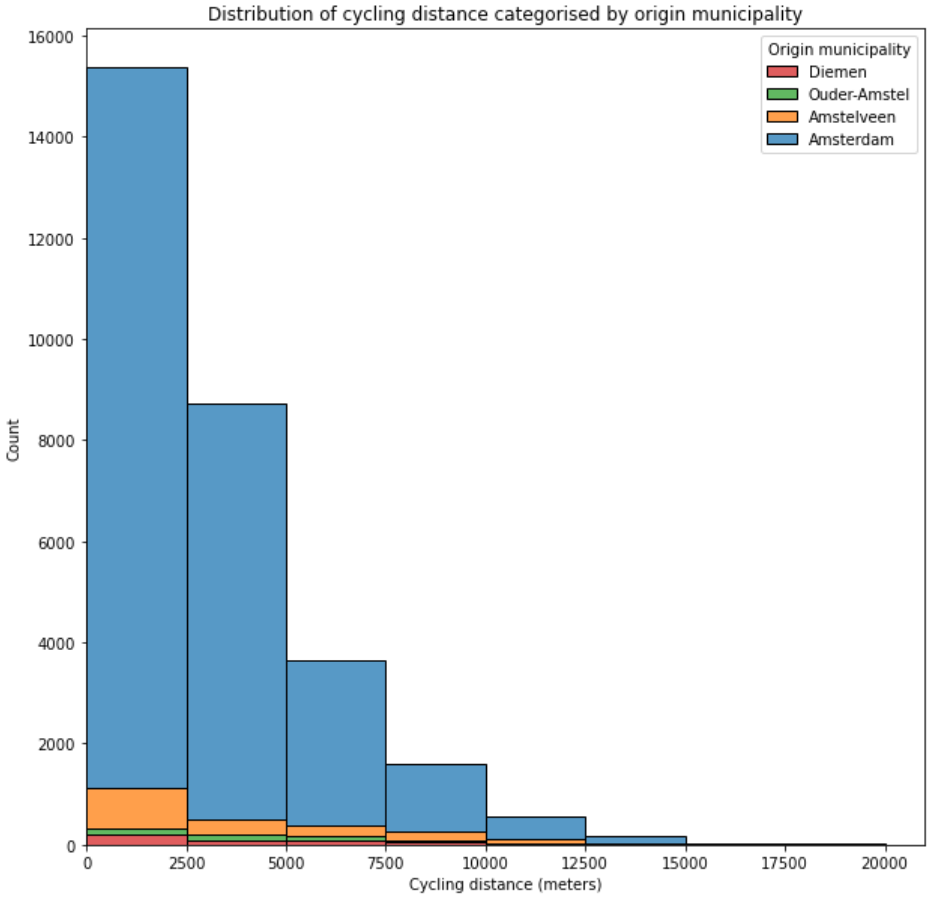
6.3 distribution of cycling distance

In [25]:
```
#Showing distribution of the cycle trips with origin as category.

plt.subplots(figsize=(10,10))
bins = [0, 2500,5000,7500,10000,12500,15000,17500,20000,22500,25000]
ax = sns.histplot(df_m, x='trip_length', hue='trip_gmn_st_max', multiple="stack", bins=bins)
ax.legend(['Diemen', 'Ouder-Amstel','Amstelveen','Amsterdam'], title='Origin municipality')

ax.set(ylabel='Count', xlabel='Cycling distance (meters)')
ax.set_xlim(0, 21000)
ax.set_title('Distribution of cycling distance categorised by origin municipality')

plt.savefig('histplot cyclingdistance origin.png')
```



6.4 Data of the most average trip

In [26]:
```
#Data of the most average trip. Found by the sum of the z-scores that was closest to zero
df_m.loc[[209351]]
```

Out[26]:

routes_routeid	trip_length	detour_meters	crow_length	trip_d_cross	trip_gmn_st_max	trip_gmn_en_max	trip_p_trees	trip_p_water
209351	2508.9128	987.276999	1521.635801	69.692022	Amsterdam	Amsterdam	0.738046	0.430353

6.5 Relation of Euclidean distance and detour distance with cycling distance

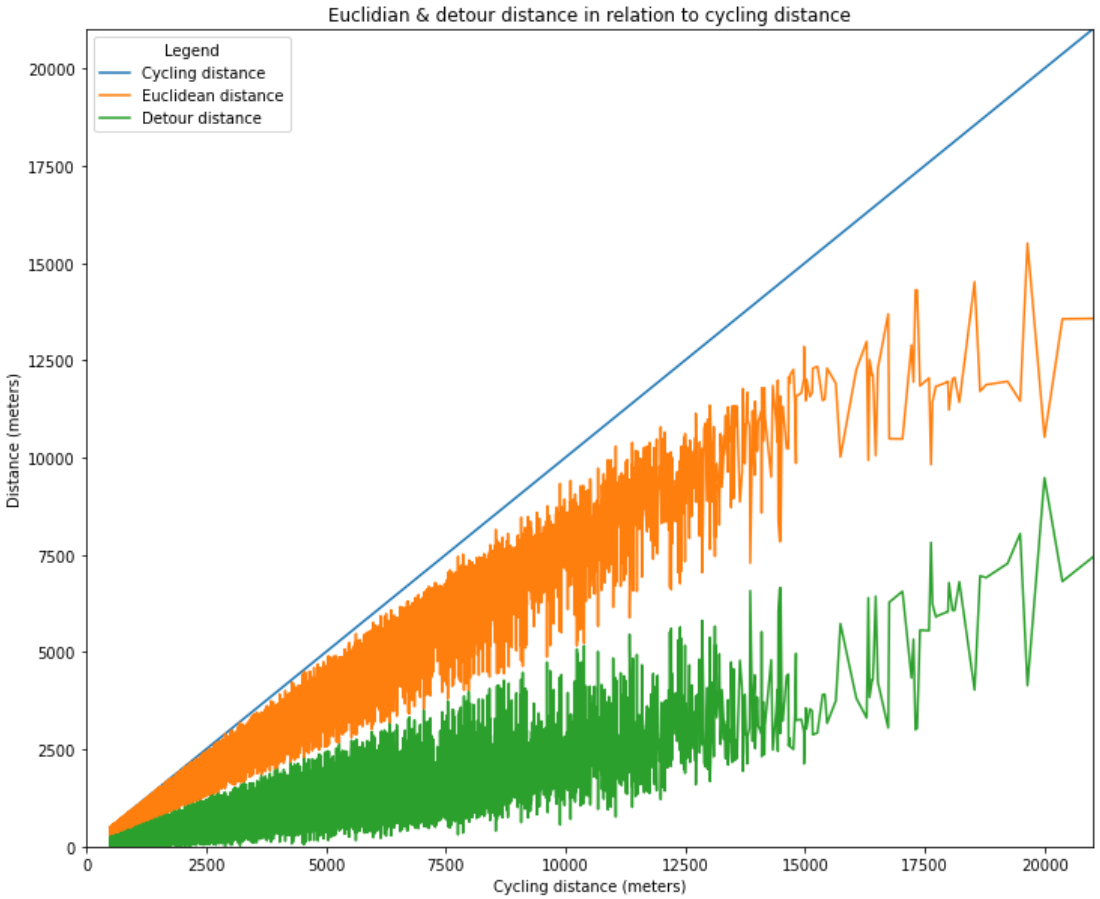
In [27]:
```
#showing the euclidean distance and detour distance in relation to cycling distance with a lineplot
plt.subplots(figsize=(12,10))

ax = sns.lineplot(data=df_m, x="trip_length", y="trip_length")
ax = sns.lineplot(data=df_m, x="trip_length", y="crow_length")
ax = sns.lineplot(data=df_m, x="trip_length", y="detour_meters")

ax.set_title('Euclidian & detour distance in relation to cycling distance')


ax.legend(['Cycling distance', 'Euclidean distance', 'Detour distance'], title='Legend')
ax.set(ylabel='Distance (meters)', xlabel='Cycling distance (meters)')
ax.set_xlim(0, 21000)
ax.set_ylim(0, 21000)

plt.savefig('lineplot distances original.png')
```



Section 07: Building models

bev = built environment variables

In [28]:
```
#Total cycling distance ~ bev
X1 = df_m[['trip_p_trees',
           'trip_p_water',
           'trip_p_vega',
           'trip_surfm_geslotenverh',
           'trip_linkfunc_cycle',
           'trip_d_cross',
           'trip_p_funcdens',
           'trip_p_funcdiff']]

Y1 = df_m[['trip_length']]

createmodel(X1,Y1)
```

```
--------------------------------------------------------------------------------
['trip_length'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_link
func_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff']

The intercept for our model is 1.796e+03
The Coefficient for our model is -2.1e+02

                            OLS Regression Results
==============================================================================
Dep. Variable:            trip_length   R-squared:                       0.077
Model:                            OLS   Adj. R-squared:                  0.077
Method:                 Least Squares   F-statistic:                     315.4
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:20   Log-Likelihood:            -2.7841e+05
No. Observations:               30137   AIC:                         5.568e+05
Df Residuals:                   30128   BIC:                         5.569e+05
Df Model:                           8
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                     1738.4087    122.587     14.181      0.000    1498.134    1978.684
trip_p_trees              -169.4126     67.267     -2.519      0.012    -301.259     -37.567
trip_p_water               760.0928     70.996     10.706      0.000     620.938     899.248
trip_p_vega                565.5620     88.401      6.398      0.000     392.293     738.831
trip_surfm_geslotenverh   1521.2488     68.590     22.179      0.000    1386.809    1655.689
trip_linkfunc_cycle        603.3665     58.210     10.365      0.000     489.272     717.461
trip_d_cross                 6.9268      0.709      9.773      0.000       5.538       8.316
trip_p_funcdens             -3.6430      0.347    -10.513      0.000      -4.322      -2.964
trip_p_funcdiff             40.6251     24.814      1.637      0.102      -8.012      89.262
==============================================================================
Omnibus:                     7286.487   Durbin-Watson:                   1.953
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17679.608
Skew:                           1.343   Prob(JB):                         0.00
Kurtosis:                       5.620   Cond. No.                     1.46e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.46e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
--------------------------------------------------------------------------------
```

Out[28]:
```
<statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bd4447488>
```

In [29]:
```
#detour distance ~ bev
X2 = df_m[['trip_p_trees',
           'trip_p_water',
           'trip_p_vega',
           'trip_surfm_geslotenverh',
           'trip_linkfunc_cycle',
           'trip_d_cross',
           'trip_p_funcdens',
           'trip_p_funcdiff']]

Y2 = df_m[['detour_meters']]

createmodel(X2,Y2)
```

```
--------------------------------------------------------------------------------
['detour_meters'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_li
nkfunc_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff']

The intercept for our model is 531.7
The Coefficient for our model is -9.1e+01

                            OLS Regression Results
==============================================================================
Dep. Variable:          detour_meters   R-squared:                       0.059
Model:                            OLS   Adj. R-squared:                  0.058
Method:                 Least Squares   F-statistic:                     235.0
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:            -2.3934e+05
No. Observations:               30137   AIC:                         4.787e+05
Df Residuals:                   30128   BIC:                         4.788e+05
Df Model:                           8
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                      515.9364     33.527     15.388      0.000     450.221     581.652
trip_p_trees               -83.6271     18.397     -4.546      0.000    -119.687     -47.567
trip_p_water               257.7764     19.417     13.276      0.000     219.718     295.835
trip_p_vega                187.3066     24.178      7.747      0.000     139.918     234.696
trip_surfm_geslotenverh    267.2028     18.759     14.244      0.000     230.434     303.972
trip_linkfunc_cycle         17.5247     15.920      1.101      0.271     -13.680      48.730
trip_d_cross                 0.9363      0.194      4.830      0.000       0.556       1.316
trip_p_funcdens             -0.6245      0.095     -6.589      0.000      -0.810      -0.439
trip_p_funcdiff            -12.3448      6.787     -1.819      0.069     -25.647       0.957
==============================================================================
Omnibus:                    15781.887   Durbin-Watson:                   1.989
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           188945.017
Skew:                           2.255   Prob(JB):                         0.00
Kurtosis:                      14.407   Cond. No.                     1.46e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.46e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
--------------------------------------------------------------------------------
```

Out[29]:
```
<statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bce89bc88>
```

In [30]:
```
#Euclidean distance ~ bev
X3 = df_m[['trip_p_trees',
           'trip_p_water',
           'trip_p_vega',
           'trip_surfm_geslotenverh',
           'trip_linkfunc_cycle',
           'trip_d_cross',
           'trip_p_funcdens',
           'trip_p_funcdiff']]

Y3 = df_m[['crow_length']]

createmodel(X3,Y3)
```

```
--------------------------------------------------------------------------------
['crow_length'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_link
func_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff']

The intercept for our model is 1.265e+03
The Coefficient for our model is -1.2e+02

                            OLS Regression Results
==============================================================================
Dep. Variable:            crow_length   R-squared:                       0.078
Model:                            OLS   Adj. R-squared:                  0.078
Method:                 Least Squares   F-statistic:                     318.1
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:            -2.7066e+05
No. Observations:               30137   AIC:                         5.413e+05
Df Residuals:                   30128   BIC:                         5.414e+05
Df Model:                           8
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                     1222.4722     94.791     12.896      0.000    1036.677    1408.267
trip_p_trees               -85.7855     52.015     -1.649      0.099    -187.737      16.166
trip_p_water               502.3164     54.898      9.150      0.000     394.714     609.919
trip_p_vega                378.2554     68.357      5.534      0.000     244.273     512.237
trip_surfm_geslotenverh   1254.0459     53.038     23.644      0.000    1150.089    1358.003
trip_linkfunc_cycle        585.8418     45.012     13.015      0.000     497.617     674.067
trip_d_cross                 5.9906      0.548     10.930      0.000       4.916       7.065
trip_p_funcdens             -3.0185      0.268    -11.265      0.000      -3.544      -2.493
trip_p_funcdiff             52.9699     19.188      2.761      0.006      15.361      90.579
==============================================================================
Omnibus:                     6824.829   Durbin-Watson:                   1.943
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            14829.605
Skew:                           1.313   Prob(JB):                         0.00
Kurtosis:                       5.217   Cond. No.                     1.46e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.46e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
--------------------------------------------------------------------------------
```

Out[30]:
```
<statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bce8a6788>
```

In [31]:
```
# Euclidean distance + bev
X4 = df_m[['trip_p_trees',
           'trip_p_water',
           'trip_p_vega',
           'trip_surfm_geslotenverh',
           'trip_linkfunc_cycle',
           'trip_d_cross',
           'trip_p_funcdens',
           'trip_p_funcdiff',
           'crow_length']]

Y4 = df_m[['trip_length']]



createmodel(X4,Y4)
```

```
--------------------------------------------------------------------------------
['trip_length'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_link
func_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff', 'crow_length']

The intercept for our model is 185.9
The Coefficient for our model is -5.9e+01

                            OLS Regression Results
==============================================================================
Dep. Variable:            trip_length   R-squared:                       0.972
Model:                            OLS   Adj. R-squared:                  0.972
Method:                 Least Squares   F-statistic:                 1.175e+05
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:            -2.2558e+05
No. Observations:               30137   AIC:                         4.512e+05
Df Residuals:                   30127   BIC:                         4.513e+05
Df Model:                           9
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                      181.3871     21.299      8.516      0.000     139.640     223.134
trip_p_trees               -60.1506     11.656     -5.161      0.000     -82.996     -37.305
trip_p_water               120.3094     12.318      9.767      0.000      96.165     144.454
trip_p_vega                 83.7909     15.325      5.468      0.000      53.753     113.828
trip_surfm_geslotenverh    -75.9871     11.994     -6.335      0.000     -99.497     -52.478
trip_linkfunc_cycle       -142.8004     10.114    -14.119      0.000    -162.625    -122.976
trip_d_cross                -0.7031      0.123     -5.714      0.000      -0.944      -0.462
trip_p_funcdens              0.2016      0.060      3.350      0.001       0.084       0.320
trip_p_funcdiff            -26.8409      4.300     -6.242      0.000     -35.269     -18.413
crow_length                  1.2737      0.001    986.607      0.000       1.271       1.276
==============================================================================
Omnibus:                    17308.000   Durbin-Watson:                   1.997
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           487328.204
Skew:                           2.246   Prob(JB):                         0.00
Kurtosis:                      22.181   Cond. No.                     3.30e+04
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 3.3e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
--------------------------------------------------------------------------------
```

Out[31]: <statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bce8b0f08>

7.1 Calculate Beta Coefficients of the models

The beta coefficients are calculated with the use of a standard scaler

In [32]:

```
import pandas as pd
from sklearn.preprocessing import MinMaxScaler, StandardScaler

# scaler = MinMaxScaler()
scaler = StandardScaler()

#Preparing a dateframe that has standarized values
df_betacoef = df_m.copy()

vlist = [
                'trip_length',
                'detour_meters',
                'crow_length',
                'trip_d_cross',
                'trip_p_trees',
                'trip_p_water',
                'trip_p_vega',
                'trip_surfm_geslotenverh',
                'trip_linkfunc_cycle',
                'trip_p_funcdens',
                'trip_p_funcdiff'
               ]

df_betacoef[vlist] = scaler.fit_transform(df_betacoef[vlist].to_numpy())
```

In [33]:
```
#Beta coefficients: Total cycling distance ~ bev
X1 = df_betacoef[['trip_p_trees',
            'trip_p_water',
            'trip_p_vega',
            'trip_surfm_geslotenverh',
            'trip_linkfunc_cycle',
            'trip_d_cross',
            'trip_p_funcdens',
            'trip_p_funcdiff']]

Y1 = df_betacoef[['trip_length']]

createmodel(X1,Y1)
```

```
--------------------------------------------------------------------------------
['trip_length'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_link
func_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff']

The intercept for our model is -0.001314
The Coefficient for our model is -0.019

                            OLS Regression Results
==============================================================================
Dep. Variable:            trip_length   R-squared:                       0.077
Model:                            OLS   Adj. R-squared:                  0.077
Method:                 Least Squares   F-statistic:                     315.4
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:                -41551.
No. Observations:               30137   AIC:                         8.312e+04
Df Residuals:                   30128   BIC:                         8.319e+04
Df Model:                           8
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                    1.492e-16      0.006    2.7e-14      1.000      -0.011       0.011
trip_p_trees               -0.0152      0.006     -2.519      0.012      -0.027      -0.003
trip_p_water                0.0666      0.006     10.706      0.000       0.054       0.079
trip_p_vega                 0.0667      0.010      6.398      0.000       0.046       0.087
trip_surfm_geslotenverh     0.1274      0.006     22.179      0.000       0.116       0.139
trip_linkfunc_cycle         0.0648      0.006     10.365      0.000       0.053       0.077
trip_d_cross                0.0607      0.006      9.773      0.000       0.048       0.073
trip_p_funcdens            -0.1083      0.010    -10.513      0.000      -0.129      -0.088
trip_p_funcdiff             0.0220      0.013      1.637      0.102      -0.004       0.048
==============================================================================
Omnibus:                     7286.487   Durbin-Watson:                   1.953
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17679.608
Skew:                           1.343   Prob(JB):                         0.00
Kurtosis:                       5.620   Cond. No.                         5.27
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
--------------------------------------------------------------------------------
```

Out[33]:
```
<statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bce8bb788>
```

In [34]:
```
#Beta coefficients: detour distance ~ bev
X2 = df_betacoef[['trip_p_trees',
            'trip_p_water',
            'trip_p_vega',
            'trip_surfm_geslotenverh',
            'trip_linkfunc_cycle',
            'trip_d_cross',
            'trip_p_funcdens',
            'trip_p_funcdiff']]

Y2 = df_betacoef[['detour_meters']]

createmodel(X2,Y2)
```

```
--------------------------------------------------------------------------------
['detour_meters'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_li
nkfunc_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff']

The intercept for our model is -0.002358
The Coefficient for our model is -0.03

                            OLS Regression Results
==============================================================================
Dep. Variable:          detour_meters   R-squared:                       0.059
Model:                            OLS   Adj. R-squared:                  0.058
Method:                 Least Squares   F-statistic:                     235.0
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:                -41850.
No. Observations:               30137   AIC:                         8.372e+04
Df Residuals:                   30128   BIC:                         8.379e+04
Df Model:                           8
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                     3.643e-17      0.006   6.52e-15      1.000      -0.011       0.011
trip_p_trees                -0.0277      0.006     -4.546      0.000      -0.040      -0.016
trip_p_water                 0.0834      0.006     13.276      0.000       0.071       0.096
trip_p_vega                  0.0816      0.011      7.747      0.000       0.061       0.102
trip_surfm_geslotenverh      0.0827      0.006     14.244      0.000       0.071       0.094
trip_linkfunc_cycle          0.0069      0.006      1.101      0.271      -0.005       0.019
trip_d_cross                 0.0303      0.006      4.830      0.000       0.018       0.043
trip_p_funcdens             -0.0686      0.010     -6.589      0.000      -0.089      -0.048
trip_p_funcdiff             -0.0246      0.014     -1.819      0.069      -0.051       0.002
==============================================================================
Omnibus:                    15781.887   Durbin-Watson:                   1.989
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           188945.017
Skew:                           2.255   Prob(JB):                         0.00
Kurtosis:                      14.407   Cond. No.                         5.27
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
--------------------------------------------------------------------------------
```

Out[34]:
```
<statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bce8c3b48>
```

In [35]:
```
#Beta coefficients: Euclidean distance - bev
X3 = df_betacoef[['trip_p_trees',
            'trip_p_water',
            'trip_p_vega',
            'trip_surfm_geslotenverh',
            'trip_linkfunc_cycle',
            'trip_d_cross',
            'trip_p_funcdens',
            'trip_p_funcdiff']]

Y3 = df_betacoef[['crow_length']]

createmodel(X3,Y3)
```

```
--------------------------------------------------------------------------------
['crow_length'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_link
func_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff']

The intercept for our model is -0.0008737
The Coefficient for our model is -0.013

                            OLS Regression Results
==============================================================================
Dep. Variable:            crow_length   R-squared:                       0.078
Model:                            OLS   Adj. R-squared:                  0.078
Method:                 Least Squares   F-statistic:                     318.1
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:                -41541.
No. Observations:               30137   AIC:                         8.310e+04
Df Residuals:                   30128   BIC:                         8.317e+04
Df Model:                           8
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                   -1.249e-16      0.006  -2.26e-14      1.000      -0.011       0.011
trip_p_trees              -0.0100      0.006     -1.649      0.099      -0.022       0.002
trip_p_water               0.0569      0.006      9.150      0.000       0.045       0.069
trip_p_vega                0.0577      0.010      5.534      0.000       0.037       0.078
trip_surfm_geslotenverh    0.1358      0.006     23.644      0.000       0.125       0.147
trip_linkfunc_cycle        0.0813      0.006     13.015      0.000       0.069       0.094
trip_d_cross               0.0678      0.006     10.930      0.000       0.056       0.080
trip_p_funcdens           -0.1160      0.010    -11.265      0.000      -0.136      -0.096
trip_p_funcdiff            0.0370      0.013      2.761      0.006       0.011       0.063
==============================================================================
Omnibus:                     6824.829   Durbin-Watson:                   1.943
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            14829.605
Skew:                           1.313   Prob(JB):                         0.00
Kurtosis:                       5.217   Cond. No.                         5.27
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
--------------------------------------------------------------------------------
```

Out[35]:
```
<statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bce8c8c88>
```

In [36]:
```
#Beta coefficients: Total cycling distance ~ euclidean distance + bev
X4 = df_betacoef[['trip_p_trees',
            'trip_p_water',
            'trip_p_vega',
            'trip_surfm_geslotenverh',
            'trip_linkfunc_cycle',
            'trip_d_cross',
            'trip_p_funcdens',
            'trip_p_funcdiff',
            'crow_length']]

Y4 = df_betacoef[['trip_length']]

createmodel(X4,Y4)
```

```
--------------------------------------------------------------------------------
['trip_length'] vs. ['trip_p_trees', 'trip_p_water', 'trip_p_vega', 'trip_surfm_geslotenverh', 'trip_link
func_cycle', 'trip_d_cross', 'trip_p_funcdens', 'trip_p_funcdiff', 'crow_length']

The intercept for our model is -0.0004537
The Coefficient for our model is -0.0053

                            OLS Regression Results
==============================================================================
Dep. Variable:            trip_length   R-squared:                       0.972
Model:                            OLS   Adj. R-squared:                  0.972
Method:                 Least Squares   F-statistic:                 1.175e+05
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:                 11277.
No. Observations:               30137   AIC:                        -2.253e+04
Df Residuals:                   30127   BIC:                        -2.245e+04
Df Model:                           9
Covariance Type:            nonrobust
================================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                    1.492e-16      0.001   1.56e-13      1.000      -0.002       0.002
trip_p_trees               -0.0054      0.001     -5.161      0.000      -0.007      -0.003
trip_p_water                0.0105      0.001      9.767      0.000       0.008       0.013
trip_p_vega                 0.0099      0.002      5.468      0.000       0.006       0.013
trip_surfm_geslotenverh    -0.0064      0.001     -6.335      0.000      -0.008      -0.004
trip_linkfunc_cycle        -0.0153      0.001    -14.119      0.000      -0.017      -0.013
trip_d_cross               -0.0062      0.001     -5.714      0.000      -0.008      -0.004
trip_p_funcdens             0.0060      0.002      3.350      0.001       0.002       0.009
trip_p_funcdiff            -0.0145      0.002     -6.242      0.000      -0.019      -0.010
crow_length                 0.9852      0.001    986.607      0.000       0.983       0.987
==============================================================================
Omnibus:                    17308.000   Durbin-Watson:                   1.997
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           487328.204
Skew:                           2.246   Prob(JB):                         0.00
Kurtosis:                      22.181   Cond. No.                         5.34
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
--------------------------------------------------------------------------------
```

Out[36]:
```
<statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bcea11888>
```

7.2 Some extra models that have been created

In [37]:
```
#Total cycling distance ~ Euclidean distance
X5 = df_m[['crow_length']]

Y5 = df_m[['trip_length']]

createmodel(X5,Y5)
```

```
--------------------------------------------------------------------------------
['trip_length'] vs. ['crow_length']

The intercept for our model is 12.9
The Coefficient for our model is 1.3

                            OLS Regression Results
==============================================================================
Dep. Variable:            trip_length   R-squared:                       0.972
Model:                            OLS   Adj. R-squared:                  0.972
Method:                 Least Squares   F-statistic:                 1.035e+06
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:            -2.2589e+05
No. Observations:               30137   AIC:                         4.518e+05
Df Residuals:                   30135   BIC:                         4.518e+05
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          13.9317      4.044      3.445      0.001       6.005      21.858
crow_length     1.2744      0.001   1017.550      0.000       1.272       1.277
==============================================================================
Omnibus:                    17212.874   Durbin-Watson:                   1.999
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           470895.569
Skew:                           2.238   Prob(JB):                         0.00
Kurtosis:                      21.841   Cond. No.                     5.20e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 5.2e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
--------------------------------------------------------------------------------
```

Out[37]: <statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bcea188c8>

In [38]:
```
#Total cycling distance ~ detour distance
X6 = df_m[['detour_meters']]

Y6 = df_m[['trip_length']]

createmodel(X5,Y5)
```

```
--------------------------------------------------------------------------------
['trip_length'] vs. ['crow_length']

The intercept for our model is 12.9
The Coefficient for our model is 1.3

                            OLS Regression Results
==============================================================================
Dep. Variable:            trip_length   R-squared:                       0.972
Model:                            OLS   Adj. R-squared:                  0.972
Method:                 Least Squares   F-statistic:                 1.035e+06
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:            -2.2589e+05
No. Observations:               30137   AIC:                         4.518e+05
Df Residuals:                   30135   BIC:                         4.518e+05
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          13.9317      4.044      3.445      0.001       6.005      21.858
crow_length     1.2744      0.001   1017.550      0.000       1.272       1.277
==============================================================================
Omnibus:                    17212.874   Durbin-Watson:                   1.999
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           470895.569
Skew:                           2.238   Prob(JB):                         0.00
Kurtosis:                      21.841   Cond. No.                     5.20e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 5.2e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
--------------------------------------------------------------------------------
```

Out[38]: <statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bce891448>

In [39]:
```
#Total cycling distance ~ detour distance + bev
X7 = df_m[['trip_p_trees',
           'trip_p_water',
           'trip_p_vega',
           'trip_surfm_geslotenverh',
           'trip_linkfunc_cycle',
           'trip_d_cross',
           'trip_p_funcdens',
           'trip_p_funcdiff',
           'detour_meters']]

Y7 = df_m[['trip_length']]



createmodel(X6,Y6)
```

```
--------------------------------------------------------------------------------
['trip_length'] vs. ['detour_meters']

The intercept for our model is 958.3
The Coefficient for our model is 3.2

                            OLS Regression Results
==============================================================================
Dep. Variable:            trip_length   R-squared:                       0.769
Model:                            OLS   Adj. R-squared:                  0.769
Method:                 Least Squares   F-statistic:                 1.005e+05
Date:                Wed, 09 Dec 2020   Prob (F-statistic):               0.00
Time:                        13:52:21   Log-Likelihood:            -2.5752e+05
No. Observations:               30137   AIC:                         5.151e+05
Df Residuals:                   30135   BIC:                         5.151e+05
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                    coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          945.5263     10.190     92.792      0.000     925.554     965.499
detour_meters    3.2389      0.010    316.947      0.000       3.219       3.259
==============================================================================
Omnibus:                     6017.919   Durbin-Watson:                   1.950
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            34993.972
Skew:                           0.837   Prob(JB):                         0.00
Kurtosis:                       8.006   Cond. No.                     1.42e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.42e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
--------------------------------------------------------------------------------
```

Out[39]: <statsmodels.regression.linear_model.RegressionResultsWrapper at 0x28bcea245c8>

Annex C - Scatter & Distribution Plots

Thesis: Cycling Distance and the Built Environment - Simon de Haas

Date: 09-12-2020