

2011  
2012

## FACULTY OF BUSINESS ECONOMICS

*Master of Management: Management Information Systems*

## Masterproef

*Sensitivity analysis of the Feathers activity-based  
model for Flanders*

Promotor :  
Prof.dr.ir Tom BELLEMANS

Ward Vanderheyden

*Master Thesis nominated to obtain the degree of Master of Management , specialization  
Management Information Systems*

2011  
2012

# FACULTY OF BUSINESS ECONOMICS

*Master of Management: Management Information Systems*

## Masterproef

*Sensitivity analysis of the Feathers activity-based model for Flanders*

Promotor :  
Prof.dr.ir Tom BELLEMANS

Ward Vanderheyden

*Master Thesis nominated to obtain the degree of Master of Management , specialization  
Management Information Systems*



---

## **Foreword**

---

This master thesis will form the last step in my degree in Master of Management: Management Information Systems (MIS) at Hasselt University.

Making this thesis was a demanding job, but at the same time it was an opportunity to explore new fields that I was not familiar with before. There is no doubt that I learned a lot during the entire process.

Without the help of others, I would not have been able to complete research. Therefore, I wish to thank everyone who helped me. I would like to specifically thank Prof. Dr. Ir. Tom Bellemans, my supervisor, and dr. ir. Bruno Kochan, who assisted me throughout the entire process. Their vision and professional advice were crucial in the realization of this master thesis. I would also like to thank friends and family for their support.

---

## Summary

---

The transportation of goods and people is a key element of the economy. With the ever increasing intensity of road traffic, several issues arise that need to be addressed. Policy makers require reliable forecasts of traffic behaviour in order to develop an effective policy. Activity-based models are best suited to make the predictions used for these policies.

This master thesis will investigate the performance of one such activity-based transportation model, FEATHERS, when presented with decreasing amounts of training data. This study will attempt to do this by methods of progressive sampling, evaluating both the value of parameters derived from the output and the performance of the decision trees, which form the fundamentals of the decision process in the activity-based model. The goal is to be able to report on a minimum amount of training data required for the model to function at an acceptable level.

**Chapter one** forms the introduction to this master thesis. The background of the research is given and the research objectives are formulated. One central research questions and several more specific questions will guide the research. The central research question is as follows: "How is the performance of the FEATHERS model affected by a decrease in training data?"

**Chapter two** is the first of several chapters that form the literature study. This chapter gives a brief introduction to microsimulation models. First, a general overview of microsimulation models is given. Next, a more specific look is taken at activity-based models. It becomes clear that they have distinct advantages over other types of models and therefore are best suited to make predictions concerning travel behaviour. **Chapter three** gives an overview of the ALBATROSS model, which is an activity-based model

used for transportation research. It forms the basis of the FEATHERS model that forms the basis of this research. The conceptual framework and scheduling model are described and the process model is discussed.

**Chapter four** gives more information on the 'Onderzoek VerplaatsingsGedrag Vlaanderen' (OVG). The training data analyzed in this master thesis was derived from the OVG research. The concept and methodology of the OVG research are explained in this chapter. The FEATHERS framework, which is analyzed in this study, is described in **chapter five**. This section describes how the ALBATROSS model was incorporated in the FEATHERS framework, and which adjustments had to be made to allow for this. The statistical instruments used in the analysis are described in **chapter six**. The instruments used are the box plot, the 'Sequence Alignment Method' (SAM) and the 'Confusion Matrix Accuracy' (CMA).

**Chapter seven** is the actual analysis performed in this master thesis. First, the method used and process followed is extensively explained. Next, the analysis is performed in 3 major steps: an analysis by parameter, a SAM analysis and the analysis of the decision trees using CMA.

The discussion of the results of this study follows in **chapter eight**, where conclusions are drawn. It is concluded that the model suffers quickly from a decrease in training data, and it is recommended that the full training set be used for optimal model performance. The use of the model with 50% or less of the training data set used is strongly discouraged.

---

## Table of contents

---

### Contents

Foreword .....	- 1 -
Summary.....	- 2 -
Table of contents .....	- 4 -
List of figures.....	- 6 -
List of tables.....	- 9 -
Chapter 1: Introduction .....	- 11 -
1.1 Background .....	- 11 -
1.2 Research objective .....	- 12 -
Chapter 2: Microsimulation models.....	- 15 -
2.1 General .....	- 15 -
2.2 Activity-based modeling .....	- 17 -
Chapter 3: The ALBATROSS model .....	- 19 -
3.1 Conceptual framework .....	- 19 -
3.2 Scheduling model .....	- 22 -
3.3 Process model (scheduler).....	- 22 -
Chapter 4: 'Onderzoek VerplaatsingsGedrag Vlaanderen' (OVG) .....	- 29 -
4.1 Concept.....	- 29 -
4.2 Methodology.....	- 30 -
Chapter 5: The FEATHERS framework.....	- 31 -
Chapter 6: Statistical instruments.....	- 35 -
6.1 The box plot .....	- 35 -
6.2 'Sequence Alignment Method' (SAM) .....	- 36 -

6.3	'Confusion Matrix Accuracy' (CMA).....	- 38 -
	Chapter 7: Analysis.....	- 41 -
7.1	Method .....	- 41 -
7.2	Analysis by parameter .....	- 49 -
7.3	SAM analysis .....	- 62 -
7.4	CMA analysis .....	- 66 -
	Chapter 8: Discussion .....	- 73 -
8.1	Conclusion.....	- 73 -
	References .....	- 75 -



---

## List of figures

---

Figure 1: Process model for fixed activity patterns .....	25 -
Figure 2: Process model for predicting locations of fixed and flexible activities .....	26 -
Figure 3: Process model for flexible activity patterns.....	27 -
Figure 4: Anatomy of a box plot .....	36 -
Figure 5: Step 1 of the training process in the FEATHERS configuration file: creating ObservedFiles	44
-	
Figure 6: Step 2 of the training process in the FEATHERS configuration file: creating PADTdataBIN files .....	44 -
Figure 7: Step 3 of the training process in the FEATHERS configuration file: creating new bins .....	44 -
Figure 8: Step 4 of the training process in the FEATHERS configuration file: example of text file with new bins .....	45 -
Figure 9: Step 4 of the training process in the FEATHERS configuration file: putting the new bins in the clasmod module .....	46 -
Figure 10: Step 5 of the training process in the FEATHERS configuration file: creating the decision trees .....	46 -
Figure 11: Configuring FEATHERS for predictions: PopMod module.....	47 -
Figure 12: Configuring FEATHERS for predictions: PredictedFile submodule .....	47 -
Figure 13: Figure 5: Box plot of the amount of trips per person per day, fracs 1 to 16 .....	50 -
Figure 14: Deviation (%) in average amount of trips per person per day compared to reference value, fracs 1 to 16.....	50 -
Figure 15: Box plot of the amount of kilometres travelled per person per day, fracs 1 to 16.....	51 -
Figure 16: Deviation (%) in average amount of kilometres travelled per person per day compared to reference value, fracs 1 to 16.....	51 -

Figure 17: Box plot of the amount of trips per person per day by car (as driver), fracs 1 to 16 .....	52 -
Figure 18: Deviation (%) in average amount of trips per person per day by car (as driver) compared to reference value, fracs 1 to 16 .....	52 -
Figure 19: Box plot of the amount of trips per person per day on foot or by bicycle, fracs 1 to 16 ..	53 -
Figure 20: Deviation (%) in average amount of trips per person per day on foot or by bicycle compared to reference value, fracs 1 to 16 .....	54 -
Figure 21: Box plot of the amount of trips per person per day by public transport, fracs 1 to 16 ...	55 -
Figure 22: Deviation (%) in average amount of trips per person per day by public transport compared to reference value, fracs 1 to 16 .....	55 -
Figure 23: Box plot of the amount of trips per person per day by car as a passenger, fracs 1 to 16 ..	56 -
Figure 24: Deviation (%) in average amount of trips per person per day by car as a passenger compared to reference value, fracs 1 to 16 .....	56 -
Figure 25: Box plot of the amount of kilometres travelled by car as a driver, fracs 1 to 16 .....	57 -
Figure 26: Deviation (%) in average amount of kilometres travelled by car as a driver per person per day compared to reference value, fracs 1 to 16 .....	57 -
Figure 27: Box plot of the amount of kilometres travelled on foot or by bicycle, fracs 1 to 16 .....	58 -
Figure 28: Deviation (%) in average amount of kilometres travelled per person per day on foot or by bicycle compared to reference value, fracs 1 to 16 .....	59 -
Figure 29: Box plot of the amount of kilometres travelled by public transport, fracs 1 to 16 .....	59 -
Figure 30: Deviation (%) in average amount of kilometres travelled per person per day by public transport compared to reference value, fracs 1 to 16 .....	60 -
Figure 31: Box plot of the amount of kilometres travelled by car as a passenger, fracs 1 to 16 .....	61 -
Figure 32: Deviation (%) in average amount of kilometres travelled per person per day by car as a passenger compared to reference value, fracs 1 to 16 .....	61 -
Figure 33: Average SAM values for training and validation data sets .....	64 -
Figure 34: Average difference in SAM values between validation and training sets .....	64 -

Figure 35: Difference in SAM values between validation and training sets for each subset, fracs 1 to 8-  
65 -

Figure 36: Difference in SAM values between validation and training sets for each subset, frac 16- 65 -

Figure 37: Learning curve ..... - 67 -

Figure 38: Learning curve of decision tree 1 (inclusion work) ..... - 71 -

Figure 39: Learning curve of decision tree 18 (Timing)..... - 71 -

Figure 40: Learning curve of decision tree 5 (location, order) ..... - 72 -

---

## List of tables

---

Table 1: Example of a confusion matrix .....	38 -
Table 2: Attributes of the FEATHERS input files .....	42 -
Table 3: Size and subsets of the input data sets.....	43 -
Table 4: Number of households in the training and validation data sets for fracs 1 to 16.....	63 -
Table 5: Normalised average CMA values, for each progressive sampling step from 10% till 90% (in %) .....	69 -
Table 6: Percentage of the number of samples with a tangent smaller than 0.25 degrees, for progressive sampling step from 10% till 90%.....	70 -



---

## **Chapter 1: Introduction**

---

### **1.1 Background**

The transport of people and goods is a key part of the economy. With the strong increase in road traffic, several issues arise, which can impose substantial costs on the economy. These issues range from traffic congestions to infrastructure and road maintenance, ecological problems concerning CO<sub>2</sub> emissions and even health issues due to fine particles in the air (Proost et al., 2011). In order to deal with these issues, two elements are key: reducing traffic volume to a level where conditions do not vary much from day to day, and providing alternatives to reduce the intensity of use of sparse road space in congested conditions (Goodwin, 2004). In order for policy makers to address the concerns they have about traffic congestions, emissions and infrastructure, they need to have reliable forecasts of travel behaviour. This means understanding the transportation mode people choose to use, but also the reasons behind that choice. With that understanding, policy makers can effectively influence travel behaviour with the various instruments they have at their disposal, while also getting a better idea of possible secondary effects their decisions might have.

Activity-based models are best suited to predict travel behaviour, because of their ability to answer “what if” questions. However, they require different data than more conventional models. In order to build an activity-based model, data on activity patterns are required. For conventional models, there are several types of travel surveys that could be employed to estimate the necessary data. Given the specific needs of the activity-based modeling approach, however, the travel survey to be used has to include measurements of activities at the end of trips and how and when the respondents chose to do them. This implies that the data needed has to be collected on an individual basis. It is an extensive and expensive procedure. It is thus important to keep the amount of

data required to a minimum, while making sure that the model remains adequately accurate in its predictions. This thesis will investigate the effects of a reduced amount of data on the accuracy and will try to determine the minimum amount of data needed for the model to remain at an acceptable level of accuracy.

## 1.2 Research objective

As explained above, microsimulation models are an important tool for policy makers in the decision making process concerning transportation. One of the main models for this in Flanders is the FEATHERS model.

The FEATHERS model operates with input data derived from a Flemish study called 'Onderzoek VerplaatsingsGedrag Vlaanderen' (OVG). This survey is a trip-based survey method, with additional information on trip purposes, providing information on activities in between the trips. This makes it particularly suitable to be used as input data for the FEATHERS model. 8800 persons were selected based on a random sample of the population and were interviewed face-to-face. Clearly, this procedure requires a lot of time and effort, which could be well spent elsewhere.

This master thesis will investigate the possibility of training the model with less data, and will analyze the performance of the model when trained with smaller data sets. Additionally, the minimum amount of training data needed for the model to function properly will be investigated. This will be done in the first place by running the model with progressively smaller fractions of the training data set. The distribution and average values of several important output parameters will be analyzed for each of the data sets. In a next step, the performance of the model in predicting both seen and unseen data sets will be analyzed. We will check whether the model suffers significantly in this regard from a decreased amount of training data, and if so, at which stage this happens. For a more in-depth analysis, we will look at the performance of the decision trees underlying the prediction process. This analysis should show whether specific parts of the prediction

process suffer more than others. If this is the case, these different areas will be identified.

First, however, a literature review will give some more information on microsimulation models in general, and the ALBATROSS model & FEATHERS framework in more detail. The statistical instruments used in the analysis will also be discussed.

The research objective of this master thesis can be summarized by the following central research question:

**How is the performance of the Feathers model affected by a decrease in training data?**

Additionally, following specific research questions will further guide the research process:

- What does the architecture of the FEATHERS framework look like and how does the prediction process work?
- Which statistical instruments can be used to analyze the model performance?
- What is the minimum amount of training data required for the FEATHERS model to function properly?
- Are there certain aspects of the model that suffer more than others from the decreased amount of training data?





## **Chapter 2: Microsimulation models**

---

### **2.1 General**

Microsimulation has been in existence since the 1950s, but it wasn't used widespread until much later, due to several reasons, such as limited computing power. Microsimulation models generate data on social or economic units. Often, these units are drawn from data based on surveys. Because the individual level is used as the basis of the model, microsimulation allows for the analysis of the distribution of resources across different groups. This distinguishes microsimulation models from other models that try to simulate systems as a whole. Additionally, microsimulation enables the exploration of heterogeneity and diversity within the simulated population (Zaidi et al., 2001).

Microsimulation models have proven to be particularly useful as tools for policy analysis. This is mainly due to their potency in answering "what if?" questions. Furthermore, they allow analyzing the impact of prospective models on the individual level. Microsimulation models are dynamic in nature. Aside from simulating a policy environment, like static models, they also incorporate behavioural response to these policies.

The dynamic component of a microsimulation model exists in the changes in behaviour that are applied to individual cases or groups. This is called the "ageing" of a case. There are 2 approaches for such ageing: static and dynamic ageing. Static ageing involves the re-weighting of the data after every period, based on an external data source. Dynamic ageing simulates new attributes for each person, using the attributes of the previous period. The key difference is that static ageing adapts a sample to external estimates at one point in time, essentially ignoring the processes that generated the individual observations in this sample. This implies that external estimates exist (which isn't the case for prospective models) and that they are accurate. Dynamic ageing generates

underlying social processes, which opens up greater opportunities for research, in addition to addressing policy concerns. In reality, however, most models use a combination of both types of ageing in their procedures.

A second important distinction exists between deterministic and stochastic processes within dynamic modeling. In a deterministic model, the relationships are determined by parameters defined within the model. A stochastic model incorporates random processes to either reflect the random nature of the relationship or to account for random influences. Most dynamic microsimulation models use a combination of stochastic and deterministic simulation processes.

Many microsimulation models have been constructed and modified over the years, such as DYNASIM, CORSIM, DYNAMOD & MOSART. Some important lessons have been learned from the use of these models:

- A successful model requires **clear objectives**.
- Model builders need to be sensitive to the **shortcomings of data used in estimating model parameters. Sensitivity analysis is essential** in gauging the impact of particular parameters on the output of the model.
- The model should be **flexible** enough to incorporate the most recent and robust data.
- **Innovation** in model building may be desirable, but it involves taking **risks**.
- There may be **questions about the feasibility and costs** of running a microsimulation model. Sometimes **simpler solutions may be preferable**.
- Producing output that covers the **short and medium term, as well as the longer term**, ensures that the model remains credible.
- The choice of **base data** is an early but very important decision in the model building process.

## 2.2 Activity-based modeling

Before activity-based modeling became popular, trip-based models were the conventional method of travel demand forecasting. The trip-based models, however, always lacked a valid representation of underlying travel behaviour. For a period these models were sufficient to assess the relative performance of transportation alternatives, but due to fundamental changes in urban, environmental and energy policy in the 1970s, a different approach to travel forecasting was needed. Many theories and frameworks were developed, but they all shared a common philosophy: travel was now analyzed as "daily or multi-day patterns of behaviour, related to and derived from differences in lifestyles and activity participation among the population" (Jones et al, 1990). This common philosophy is known as "the activity-based approach". The fundamental idea is that travel behaviour is a consequence of activity behaviour, and thus the understanding of travel behaviour is secondary to the understanding of activity behaviour. Travel becomes a derived demand, based on the need to pursue activities distributed in space (Recker, 1995).

The activity-based models have an important advantage over other types of models. They are better at understanding the direct impact of transportation policies on travelers, allowing them to better predict the effect of these policies. Furthermore, activity-based models are also able to take secondary effects of transportation policies into account, which is impossible for many other models.



---

## **Chapter 3: The ALBATROSS model**

---

ALBATROSS stands for 'A Learning Based Transportation Oriented Simulation System'. It is an activity-based model that predicts which activities are conducted, when, where, for how long, with whom, and the transport mode involved. It was originally developed for the Dutch Ministry of Transportation, Public Works and Water Management, to explore possibilities of a rule-based approach and develop a travel demand model for policies impact analysis.

### **3.1 Conceptual framework**

It is postulated that activity participation, allocation and implementation fundamentally take place at the household level. At that level, activities are performed and decisions are made regarding what activities to conduct. The generation of activity calendars covers several time frames.

Some long term decisions at the household level will strongly influence the composition of the activity calendars. Decisions regarding marriage and children are irreversible in the short term and thus have a strong impact on the kinds of activities that can and need to be performed within a household. Other decisions, like choice of work and the purchase of a car, can theoretically be changed in the short term, but in general these represent big choices that are only made after a longer period of consideration. Hence, these decisions have a major influence on the possible activity patterns, since the location of the residence and the workplace are the main locations for activities, and along with the transportation system available, they form the cornerstones of the decision process. It is up to the household to allocate activities to household members. The allocation

mechanism will depend on several factors, such as time and gender-specific roles (Arentze et al., 2000).

An individual activity program is then derived from the household activity calendar. This process depends on the nature of the activity, the urgency of the activity and the desire to meet activity and time-related objectives. Once this individual program is generated, the next step is to schedule the activities.

Albatross uses a sequential decision process to produce the daily activity schedules of individuals in a household. A priority-based scheduling process is applied, where mandatory activities are scheduled first and discretionary activities are scheduled next. Additionally, timing and trip-chaining decisions have priority over location decisions, which in turn have priority over decisions regarding transport mode. The result is an activity schedule, which describes for a given day which activities are conducted, when, for how long, where, with whom and the transport mode involved. The model does take interactions between individuals into account. The scheduling processes run parallel and decisions are alternated, with each individual taking the current state of the schedule of the other into account when making a decision. This implies that scheduling decisions of one individual may put constraints on choice options of the other. This happens, for example, when there are more driving licenses than cars available in the household.

The actual process of scheduling activities can be conceptualized as a process in which an individual attempts to realize particular goals, given a variety of constraints that limit the number of feasible activity patterns.

Several types of constraints can be identified (Arentze et al., 2000):

- Situational: A person and transport mode (and perhaps other resources) cannot be at different locations at the same time.
- Institutional: Opening hours influence the possible times some activities can be implemented.
- Household: Children have to be brought to school, so other activities cannot be performed at that time.
- Spatial: Some activities cannot be performed at particular locations. Also, individuals may have incomplete information about the opportunities that certain locations offer.
- Time: Limits the number of activities because there is a minimum duration to some activities, and total time is limited.
- Spatial-temporal: The specific interaction between an individual's activity program, the individual's cognitive space, the institutional context and the transportation environment may imply that a person cannot be at a particular location at the right time to conduct a certain activity

The next question is how individuals choose between feasible activity patterns. Unlike other models, the ALBATROSS model assumes that continuous interaction with the environment results in choice heuristics that individuals and households apply when faced with a choice. These choice rules are continuously adapted through learning. The actual execution of activity programs is monitored on a real-time basis. This means that individuals are constantly faced with the decision whether or not to reschedule activities during execution, when activities cannot be performed as expected. This could happen when the use and speed of transportation networks is not as envisioned. Overall, the learning theory on which Albatross is based implies that rules governing choice behavior are heuristic, context-dependent and adaptive in nature.



### 3.2 Scheduling model

The decision trees used in Albatross are derived from observations in activity diaries, using a CHAID-based induction method. This method aims to find the smallest tree that best explains a sample of observations by recursively splitting the sample based on attribute variables. This allows taking a large set of attribute variables into account in each scheduling decision. In the decision tree induction process, segmentation and derivation of decision rules are done simultaneously. Because all earlier decisions are considered as attribute variables consistent for a current decision, the model is able to take interactions between activity-travel choices into account, both within and between individuals (Arentze et al, 2000).

The sensitivity to small changes in price variables and travel time variables of the decision trees over a continuous range is a concern. For that reason, the Parametric Action Decision Tree (PADT) is introduced. A conventional decision tree is replaced by a PADT for each decision that has travel-costs and travel-time implications. This allows Albatross to reproduce price and time elasticities for many choice facets of an activity pattern.

Decision trees offer the advantage of being able to take discontinuous, non-linear effects on choice behavior into account. A disadvantage of decision trees is that they tend to be very extensive and complex when derived from empirical data, which makes the analysis harder.

### 3.3 Process model (scheduler)

Two major components define the schedules for each individual and each day. One component generates an activity skeleton containing fixed activities, along with their duration and starting time. The second component handles the flexible activities, along

with duration, time-of-day and other travel characteristics. Both components use the same location model and assume a sequential decision process.

Figures 1, 2 and 3 schematically present the structure of each of the main components of the process model. Each numbered rectangle represents a decision tree derived from activity diary data. The indices used in the figure are defined as follows:

$i$  = index of activity in order of priority,  $i=1...I$

$j$  = index of episode of activity  $i$  in order of start time,  $j = 1...J$

$k$  = index of tour in order of start time,  $k = 1...K$

Decisions 1 to 13 comprise the skeleton components, decisions 14-20 are used for the location component for both types of activities and decisions 21-27 represent the flexible activity component.

There are 2 key principles to the process model. The first is that qualitative decisions are preferred and made as much as possible. Decisions such as selecting an activity or trip linkages between activities are made separately for this reason. Second, the sequencing of identified key choices is based on assumptions regarding priority of decisions. This is why the skeleton is determined before the flexible parts of the schedule, and start time decisions are made before location and transport mode decisions.

### **The skeleton**

The skeleton component determines activity patterns on a continuous scale. The subprocesses are as follows:

1. Determining the sleep activities pattern
2. Determining the primary work/school activity pattern
3. Determining the secondary fixed activities pattern
4. Determining the location of each fixed activity

The model chooses a start and end time for the sleep activity. For simplification, it ignores the cases where there is no sleep activity (e.g. night shift). Sleep activities during the day are not considered separately but are considered within the in-home activities category.

The primary work/school activity has a maximum of 2 episodes, with a minimum duration of 1 hour each. Activities with a shorter duration are not ignored but considered separately as a category of secondary fixed activities. The location component chooses locations in descending order of priority of fixed activities. The model chooses by increasingly narrowing down the choice set. The first tree determines whether or not the activity is conducted within the home municipality of the individual. In case of the latter option, the choice of municipality depends on a choice of an order and distance band. There are five orders based on population size. First-order municipalities have an above regional function, second-order municipalities have a regional function, third-order municipalities have a local core function and fourth and fifth-order municipalities correspond to small towns. Once the order is chosen, the choice of a distance band follows. The combination of order and distance band reduces the choice set sharply. However, if there are still multiple alternatives left, the model chooses semi-randomly. Less distant locations have a higher chance of being selected. For the selection of a zone within a municipality, the same process is followed. The model thus takes into account that location selection has a random component, but also correlates the choice with distance and order.

### **Flexible part**

Figure 3 represents the second part of the model, handling the flexible part of the schedules. The subprocesses go as follows:

1. Determining selection, travel party and duration of flexible activities
2. Determining start time and trip chaining
3. Determining the location of each flexible activity

#### 4. Determining the transport mode of each tour in the schedule

Compared to the process in the previous component, the order of mode and location decisions has been switched. Modeling location before order greatly simplified computational procedures in this section. An exception is made for the transport mode for work/school trips, because it is assumed that the order of decisions is dependent on motive. The same location model that was used for the skeleton, is used for the flexible activities.

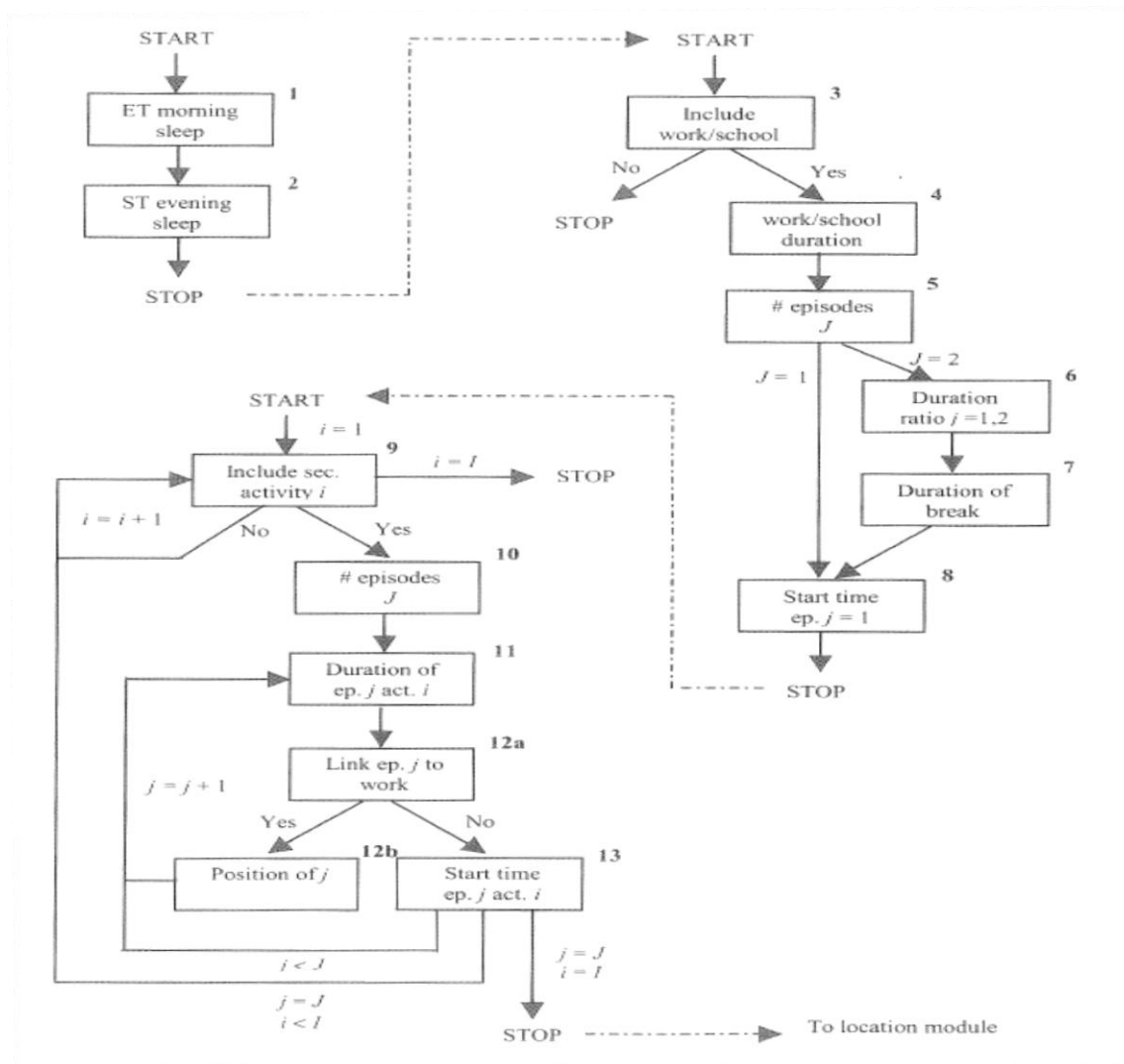


Figure 1: Process model for fixed activity patterns

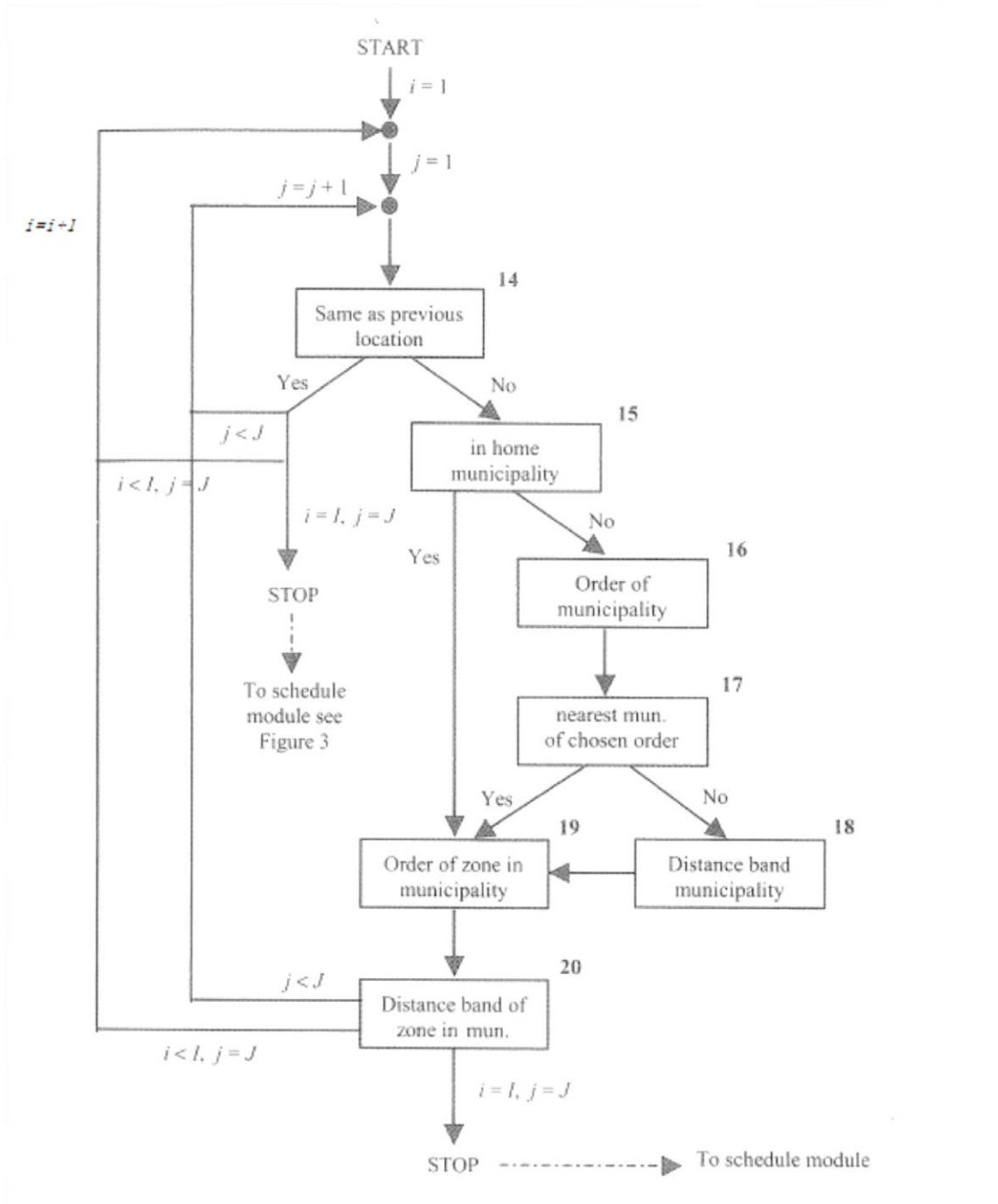


Figure 2: Process model for predicting locations of fixed and flexible activities

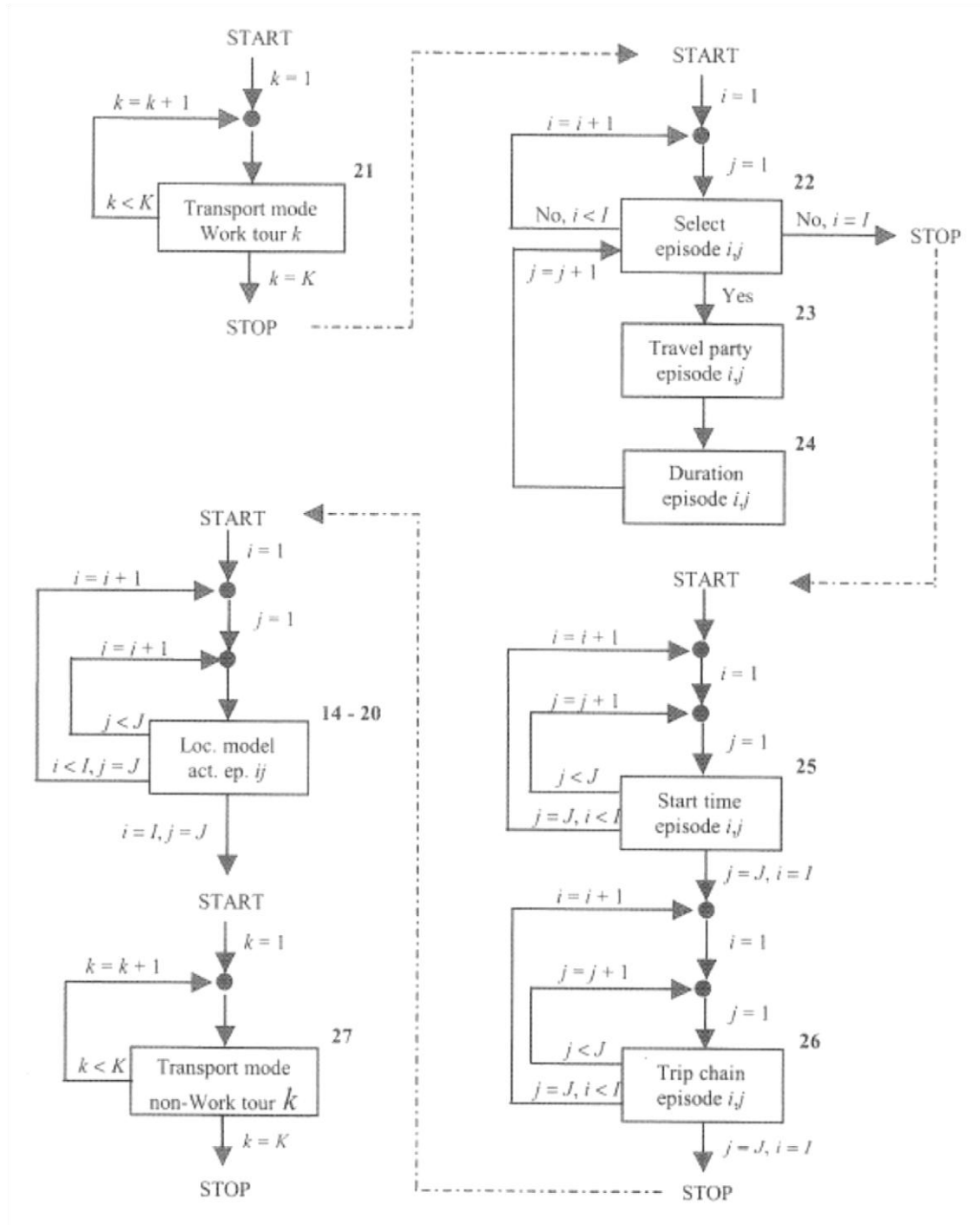


Figure 3: Process model for flexible activity patterns



---

## **Chapter 4: 'Onderzoek VerplaatsingsGedrag Vlaanderen' (OVG)**

---

Since the analysis of training data for the FEATHERS framework is an essential part of this master thesis, some information concerning the source of this data is warranted.

### **4.1 Concept**

'Onderzoek VerplaatsingsGedrag Vlaanderen' (OVG) is a Flemish study carried out by the 'Transportation Research Institute' (IMOB), part of the University Hasselt. There currently are 3 versions of the OVG. The first one was carried out between 1994 and 1995, the second between 2000 and 2001, and the third between 2007 and 2008. Data from this third version are used in the FEATHERS framework. The 3 parts were carried out at 3 specific intervals, meaning they are a form of discontinuous research. Currently, a 4<sup>th</sup> OVG is being carried out. Contrary to the previous parts, the 4<sup>th</sup> OVG is a continuous research that started in 2008 and will be finished by 2013. All surveys were carried out in the whole Flemish region (Janssens et al, 2009).

The goal of the OVG research is to get a comprehensive view of several attributes of households and individuals concerning mobility. On the household level, this mainly concerns the attributes of transport modes the household possesses. On the individual level, it mostly concerns the actual trips people make in their daily lives. Additionally, several sociologic and demographic attributes of households and individuals are investigated, to allow for a meaningful and comprehensive analysis.



## 4.2 Methodology

The research was carried out through a survey of 8800 individuals, starting at the age of 6 years old. They were selected through a sample of the national register. The participants were questioned through a face-to-face interview. Questions were asked concerning family, mobility and personal attributes.

Additionally, each person was asked to keep a travel diary, where they made note of every trip they made on a randomly chosen day. In a next step, these data were entered in the computer during another face-to-face interview. It is important to note that the behaviour of people is not investigated through objective observation in this study. Rather, people are asked to comment on their own mobility behaviour. This gives a more in-depth set of data but also adds uncertainty: the behaviour of people may not actually be the same as the behaviour they report from themselves.

Note that the methodology described concerns the 3<sup>rd</sup> OVG. Some fundamental changes were made compared to the first 2 OVGs. The most important difference is the change from surveys through telephone and post to face-to-face interviews. This was necessary to guarantee the quality of the survey. The main factors that determine the quality of social surveys are 1) being able to contact the participants and 2) getting their cooperation. Because fewer and fewer people possess telephone connected to a fixed line, more surveys would have to be done by post. However, the response rate for postal surveys is much lower than for telephone surveys. Additionally, this accessibility (or lack of it) is very selective: some social groups are overrepresented while others are missing almost entirely. This would have negative implications for the quality of the research. For this reason, the decision was made to opt fully for face-to-face interviews, which resulted in a higher quality of data than was the case for the previous OVGs (Janssens et al, 2009).

---

## Chapter 5: The FEATHERS framework

---

FEATHERS is a modular activity-based model of transport demand framework. The activity-based scheduling model present within FEATHERS is the ALBATROSS scheduling model, discussed in the previous chapter. The framework provides the tools needed for models to be created, maintained and updated. It has both tailored memory structures and a database structure allowing activity-based models to be developed, assimilated and modified inside FEATHERS. A number of steps were needed to incorporate the ALBATROSS model into the FEATHERS framework:

### I. Tailoring the ALBATROSS model to the Flemish situation

The original ALBATROSS model was specifically designed for the Netherlands, so naturally several changes have to be made for the model to be usable in Flanders.

Firstly, each of the 26 decision trees have to be replaced by new trees, derived from corresponding activity diary data gathered in Flanders.

Secondly, because the model inside FEATHERS uses a lot of continuous condition variables that are discretized into nominal attributes before training the trees, a list of discretizing bins has to be defined.

Finally, the variables related to the transport system deserve special attention. One of the most important changes is the re-implementation of the calculation of travel costs by particular modes. Since these were based on the Netherlands, some of the assumptions were invalid for Flanders. Therefore, some of these costs have to be recalculated.

## II. Preparing input data for the ALBATROSS model

Several data layers inside the FEATHERS database system have to be prepared in order to run the ALBATROSS activity-based scheduler. Schedule information, a synthetic population data set and environment information about the study area in terms of zoning system, land use and transportation system have to be processed.

- Schedule data: Given the needs of the activity-based modeling approach, specific data is required. The "Onderzoek Verplaatsingsgedrag Vlaanderen (OVG)" travel survey contains all that data needed in the model. It is essentially a trip-based survey, but information about the trip purposes and thus about the activities in between the trips, is also available. This makes it particularly suitable to be used in the FEATHERS framework.
- Synthetic population data: An important element in activity-based models is detailed information on household and person demographics. In Flanders, the gathering of personal data from administrative registers is prohibited for privacy reasons. A synthetic population data set has to be generated to compensate for the missing data. This synthetic population is a statistical duplicate of the actual population. For each household and person, important attributes are generated.
- Environment data: The **zoning system** represents the geographical component in the model. FEATHERS uses a hierarchy of three geographical layers: Superzones, Zones and Subzones. Superzones correspond with municipalities, Zones correspond with administrative units and Subzones consist of virtual areas constructed based on homogeneous characteristics. The **land use system** provides sector-specific data on the availability and attractiveness of locations for conducting specific activities. These data are available at different levels of the zoning system. The **transportation system** contains information about distances, travel times and access times through Level OF Service (LOS) matrices. These are divided by transport mode. The different modes considered are car

(driver & passenger), public transport and slow mode (on foot or by bicycle).

Additionally, different travel times are calculated to account for peak traffic.



---

## Chapter 6: Statistical instruments

---

### 6.1 The box plot

The box plot has become the standard technique to present the so-called 5-number summary, consisting of the minimum and maximum values, the upper and lower quartiles and the median. It is seen as a good way to summarize the distribution of a dataset and is a straightforward way to compare different datasets (Potter, 2006).

The typical construction of a box plot divides the data distribution into quartiles, four subsets with equal size. The box indicates the lower and upper quartiles, the interior of the box consists of the innerquartile range, which is the area between the upper and lower quartiles and consists of 50% of the distribution. The box is intersected by a crossbar, which is drawn at the median of the dataset. Whiskers on both sides of the box represent minimum and maximum values in the dataset. Sometimes the whiskers represent a multiple of the innerquartile range to remove extreme outliers, which are then presented separately through other symbols. In this case, the plot is referred to as a schematic plot rather than a box plot. Figure 4 summarizes the anatomy of a box plot (Potter, 2006).

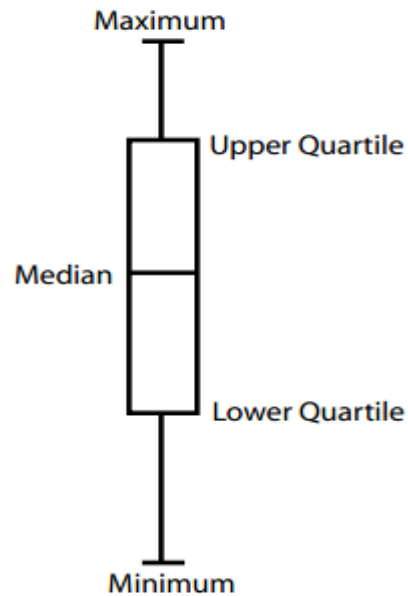


Figure 4: Anatomy of a box plot

## 6.2 'Sequence Alignment Method' (SAM)

Another important way to evaluate the model performance is assessing the goodness-of-fit. Because FEATHERS predicts activity patterns, it is important that a goodness-of-fit measure can capture the multi-faceted aspect of activity patterns. Additionally, it is crucial that the measure is also flexible in allowing the inclusion of categorical and sequential information. Most of the facets of activity patterns have a categorical nature, but the facet of activity scheduling implies sequential information.

Most similarity measures in transport science are insensitive to sequential information and are insensitive to activity patterns of unequal length. This makes them not well suited to evaluate the performance of our model. The 'Sequence Alignment Method' (SAM), introduced by Wilson (1998), can overcome these shortcomings. The method originally stems from molecular biology, where it measured biological distance between DNA & RNA strings (Arentze et al, 2000).

## **Background**

It is assumed that activity patterns can be represented as a string of information, for example by letters representing activity types. Such a string is called a sequence, because the order of the letters has a meaning. The SAM method compares two sequences by defining a source sequence and a target sequence, and calculating the total amount of effort required to equalize the source sequence with the target sequence. For this, SAM distinguishes several operations: identity, substitution, insertion and deletion. Each operation requires a certain amount of effort. The substitution operation can be thought of as the sum of the deletion and insertion operations.

There are many ways of applying operations in order to change a sequence into another sequence. These ways all lead to different computational costs. It is thus important to define an additional operational decision to define the similarity measure. SAM is based on the Levenshtein distance, which is defined as the smallest number of substitutions, insertions and deletions required to change the source sequence into the target sequence (Doran et al, 2010). When there are several choice alternatives, the smallest cost alternative is always chosen. By employing the Levenshtein distance to SAM, similarity is defined as the smallest sum of operation weighting values required to change the source sequence into the target sequence.

As mentioned, an important feature of SAM is that it captures sequential similarities, which methods based on Euclidian distance are incapable of. This is because SAM distinguishes between "wrong position but same order" and "wrong position and different order" cases. For example, consider sequences "ABCDE" (target) and "AFBCDE" (source). The conventional method counts the number of elements that are different. This would give a distance of 4 units in this case ( $B \neq F$ ,  $C \neq B$ ,  $D \neq C$  and  $E \neq D$ ). SAM would give a distance of one, deleting F from the source sequence, putting a high value on the order of elements.



Additionally, SAM also captures dissimilarity by different length. For example, consider sequences "ABC" (target) and "ABCDEF" (source). The conventional method would give a distance of zero units, because  $A = A$ ,  $B = B$  and  $C = C$ . SAM, however, would give a distance of 3 units, deleting D, E and F from the source sequence.

### 6.3 'Confusion Matrix Accuracy' (CMA)

A more in-depth analysis of the model performance means taking a look at the performance of the individual decision trees that form the basis of the scheduling process. This requires an accuracy indicator for the performance of the decision trees in the model. A confusion matrix summarizes the results of the testing of an algorithm of a decision tree. Table 1 gives an example of a confusion matrix. Assume the decision tree in the example predicts the transport mode of a trip. The possible transport modes here are car, public transport and bike. The rows give the actual transport mode that was used, while the columns give the mode predicted by the decision tree. All correct predictions are in the diagonal of the table, making it easy to spot errors.

		predicted class		
		car	public transport	Bike
actual class	Car	5	3	0
	public transport	2	3	1
	Bike	0	2	11

**Table 1: Example of a confusion matrix**

In this example, 8 cars were actually used. The decision tree predicted 5 cars correctly, while it assigned 3 trips incorrectly to public transport. Of the 6 actual public transports, 3 were correctly assigned. The bike was correctly predicted as transport mode 11 out of

13 times. It is clear that this decision tree has problems distinguishing between cars and public transport, while it can predict the choice for a bike fairly well.

The Confusion Matrix Accuracy (CMA) summarizes the information in the confusion matrix into one value that gives the accuracy of the decision tree. It is determined by calculating the ratio of correct predictions, which is given by a fraction where the denominator is the sum over all cells in the confusion matrix of the decision tree, and the numerator is the sum over all diagonal cells of the confusion matrix. For this decision tree, the CMA equals 70,3% (19/27) (Kohavi et al, 1998).



---

## Chapter 7: Analysis

---

This chapter will discuss the actual research performed for this master thesis. In the first part, the method will be outlined. In the second part, the results will be analyzed and results will be presented.

### 7.1 Method

The goal of the research is to determine how much input data FEATHERS need in order to make accurate predictions. To do this, we need to gather data by training and running the model for different training sets decreasing in size.

#### **Preparing training data sets**

The first step is creating the training data sets to be used as input for the model. FEATHERS uses a series of 5 text files as input data:

- Activities
- Households
- Journeys
- Lags
- Persons

Together, these 5 files contain all the information of the activities of a household, as they are kept in an activity diary. Each line of code in these files represents a separate activity, household, journey, lag or person. Every line starts with a unique ID number, along with other information. The files are semicolon delimited, which means that the different numbers in one line are separated by semicolons, effectively dividing them into

columns. Below a few lines of code are shown for one of the activities files. Table 1 gives an overview of the information contained in the different files.

```
0;0;7;0;300;1440;2055
1;1;4;0;300;300;-1204
2;1;4;6;835;95;953
3;1;4;0;1042;978;-1204
```

activities	Households	Journeys	Lags	Persons
Activity ID	Household ID	Journey ID	Lag ID	Person ID
Person ID	PDA?	Person ID	Journey ID	Household ID
Day	Location	Day	Car	Age
Type	Composition	Begin time	Lag number	Work status
Begin time	Income	Duration	Waiting time	Gender
Duration	Age	Transport mode	Duration	Driving license?
Location ID	Children	Start location	Transport mode	Postcode
	#cars	End location		
	#people			
	Type			
	Weight			

Table 2: Attributes of the FEATHERS input files

As we can see, the files are interlinked. For example: an activity entry contains a Person ID that refers to the person carrying out the activity.

The research will use the household level as basis because FEATHERS operates on that level. We want to gradually decrease the amount of households in the training set by a factor 2, effectively dividing each set in half. We refer to the smaller data sets as a "frac" (fraction) of the original data set. Thus, a data set of frac 4 is half the size of a frac 2 data set, and a quarter the size of the original data set, which is a frac 1. Naturally, when dividing a data set in half, we end up with 2 smaller data sets. We refer to these as

subsets of a frac. Because each subset has to be equal in size, a few entries might go unused in higher fracs. Table 2 gives an overview of the fracs, their subsets and the size.

<b>Frac</b>	<b>size</b>	<b>subsets</b>
frac 1	6266	1
frac 2	3133	2
frac 4	1566	4
frac 8	783	8
frac 16	391	16

**Table 3: Size and subsets of the input data sets**

Since FEATHERS operates on the household level and all files are interlinked, it is not possible to simply cut the 5 input files in half. Instead, after creating the household file for a frac and subset, entries from the other files have to be selected based on the relevant IDs. A small program was written to automate this procedure.

Unfortunately, all FEATHERS input files have to start with an ID of 0. This makes the procedure more complex, because once the base IDs in the first column are renamed, the IDs referring to it in other files have to be renamed as well. Because of the complexity, this was done manually. Every text file was imported in Microsoft Excel, using the semicolon delimiter to separate the data into columns. Then, the IDs were renamed manually and the file was saved as a comma delimited file (.csv). Finally, the commas were replaced by semicolons and the file was saved as a text file, now ready to be used in FEATHERS.

## **Training FEATHERS**

With the training data sets ready, the next step is the actual training of the model. The training is a step-by-step process, selecting the right modules in the configuration file at each step and then running the executable file.

- Step 1: The training data sets are put in the DatMod module. The ObservedFiles submodule is activated in the AlbExpMod module to create an observed file from the training data set. Figure 5 shows the configuration file for this step.

```
1384 <module name="AlbExpMod">
1385   <param name="Active" value="1"/>
1758   <submodule name="ObservedFiles">
1759     <param name="Active" value="1"/>
1760     <outputfile name="ObservedSurveyFile">
1761       <param name="Path" value="C:\Feathers\MobilityPlan\ObservedFile\ObservedSurveyOVG_frac_1_subset_1.obs"
1762     </outputfile>
```

Figure 5: Step 1 of the training process in the FEATHERS configuration file: creating ObservedFiles

- Step 2: The PADTdataBIN submodule is activated in the AlbExpMod module. The number of household files in the observed file is entered and A PADTdataBIN file is created, using the observed file as input. Figure 6 gives the configuration file for step 2.

```
1469   <submodule name="PADTdataBIN">
1470     <param name="Active" value="1"/>
1471     <param name="NrOfObservedHouseholdSchedules" value="6266"/> <!-- 6494 -->
1472     <inputfile name="ObservedFile">
1473       <param name="Path" value="C:\Feathers\MobilityPlan\ObservedFile\ObservedSurveyOVG_frac_1_subset_1.obs"
1474     </inputfile>
1475     <outputfile name="PADTdataFile">
1476       <param name="Path" value="C:\Feathers\MobilityPlan\DT\PADT\PADTdata_frac_1_subset_1.bin"/>
1477     </outputfile>
```

Figure 6: Step 2 of the training process in the FEATHERS configuration file: creating PADTdataBIN files

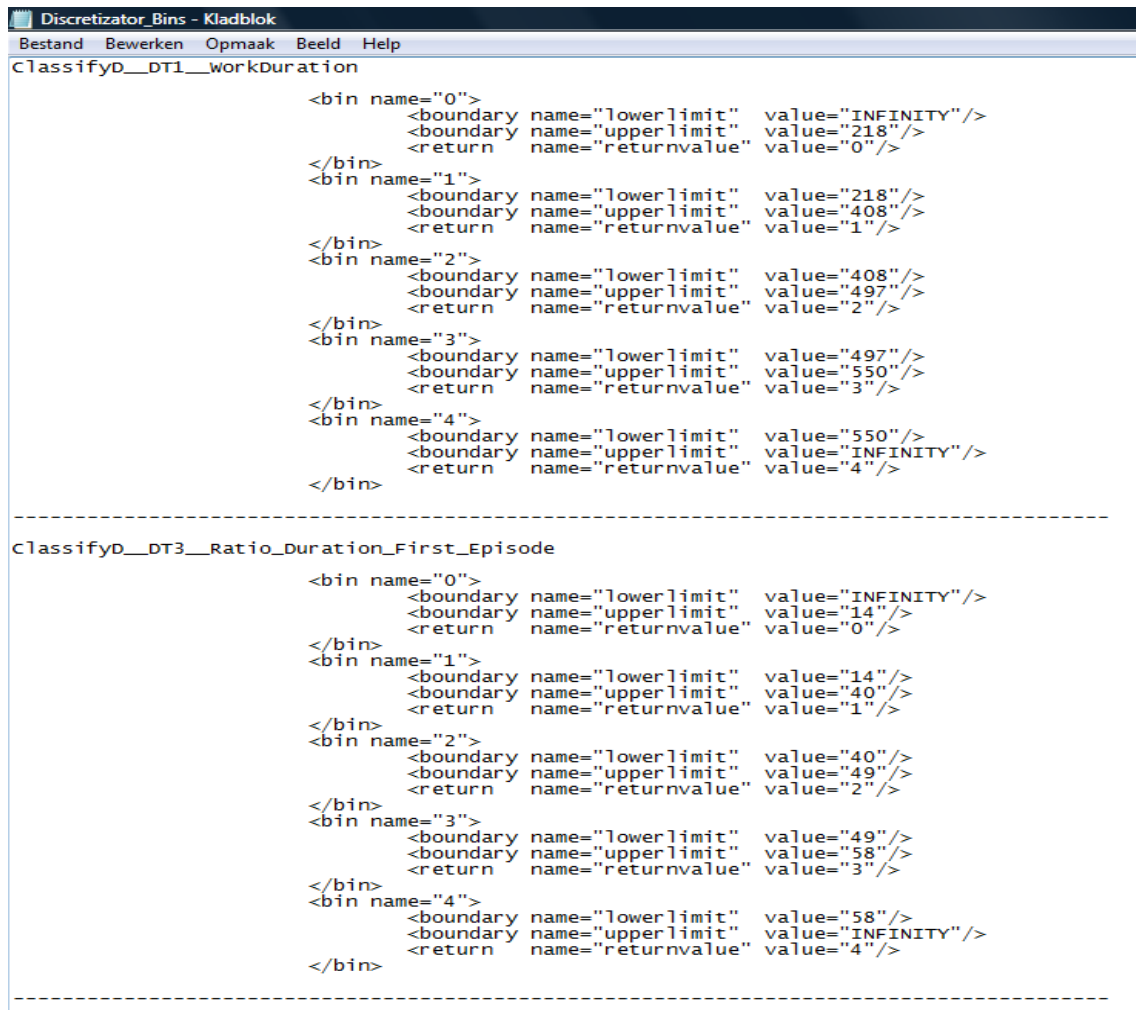
- Step 3: New discretizator bins have to be created. This is done by activating "create bins" in the DT submodule of the AlbExpMod module.

```
1512   <submodule name="DT">
1513     <param name="Active" value="1"/>
1514     <param name="NrOfObservedHouseholdSchedules" value="6266"/> <!-- 6494 -->
1515     <param name="CreateBins" value="1"/> <!-- 1 = Create bin boundaries, 0 = train decision t
1535     <inputfile name="ObservedFile">
1536       <param name="Path" value="C:\Feathers\MobilityPlan\ObservedFile\ObservedSurveyOVG_frac_1_subset_1.obs"/>
```

Figure 7: Step 3 of the training process in the FEATHERS configuration file: creating new bins

- Step 4: The previous step has created 9 text files containing new bins for several parameters in the ClasMod module. These have to be copied manually into the ClasMod, overwriting the previous bins. Figure 8 gives an example of a text file

with new bins. Figure 9 shows where the bins have to be put in the configuration file.



```
Discretizator_Bins - Kladblok
Bestand Bewerken Opmaak Beeld Help
ClassifyD__DT1__workDuration

    <bin name="0">
        <boundary name="lowerlimit" value="INFINITY"/>
        <boundary name="upperlimit" value="218"/>
        <return name="returnvalue" value="0"/>
    </bin>
    <bin name="1">
        <boundary name="lowerlimit" value="218"/>
        <boundary name="upperlimit" value="408"/>
        <return name="returnvalue" value="1"/>
    </bin>
    <bin name="2">
        <boundary name="lowerlimit" value="408"/>
        <boundary name="upperlimit" value="497"/>
        <return name="returnvalue" value="2"/>
    </bin>
    <bin name="3">
        <boundary name="lowerlimit" value="497"/>
        <boundary name="upperlimit" value="550"/>
        <return name="returnvalue" value="3"/>
    </bin>
    <bin name="4">
        <boundary name="lowerlimit" value="550"/>
        <boundary name="upperlimit" value="INFINITY"/>
        <return name="returnvalue" value="4"/>
    </bin>

-----
ClassifyD__DT3__Ratio_Duration_First_Episode

    <bin name="0">
        <boundary name="lowerlimit" value="INFINITY"/>
        <boundary name="upperlimit" value="14"/>
        <return name="returnvalue" value="0"/>
    </bin>
    <bin name="1">
        <boundary name="lowerlimit" value="14"/>
        <boundary name="upperlimit" value="40"/>
        <return name="returnvalue" value="1"/>
    </bin>
    <bin name="2">
        <boundary name="lowerlimit" value="40"/>
        <boundary name="upperlimit" value="49"/>
        <return name="returnvalue" value="2"/>
    </bin>
    <bin name="3">
        <boundary name="lowerlimit" value="49"/>
        <boundary name="upperlimit" value="58"/>
        <return name="returnvalue" value="3"/>
    </bin>
    <bin name="4">
        <boundary name="lowerlimit" value="58"/>
        <boundary name="upperlimit" value="INFINITY"/>
        <return name="returnvalue" value="4"/>
    </bin>

-----
```

Figure 8: Step 4 of the training process in the FEATHERS configuration file: example of text file with new bins



```

2960 <ClassifierType name="ClassifyD">
2961   <Classifier name="DT1"> <!--Flemish: Work duration -->
2962     <bin name="0">
2963       <boundary name="lowerlimit" value="INFINITY"/>
2964       <boundary name="upperlimit" value="218"/>
2965       <return name="returnvalue" value="0"/>
2966     </bin>
2967     <bin name="1">
2968       <boundary name="lowerlimit" value="218"/>
2969       <boundary name="upperlimit" value="408"/>
2970       <return name="returnvalue" value="1"/>
2971     </bin>
2972     <bin name="2">
2973       <boundary name="lowerlimit" value="408"/>
2974       <boundary name="upperlimit" value="497"/>
2975       <return name="returnvalue" value="2"/>
2976     </bin>
2977     <bin name="3">
2978       <boundary name="lowerlimit" value="497"/>
2979       <boundary name="upperlimit" value="550"/>
2980       <return name="returnvalue" value="3"/>
2981     </bin>
2982     <bin name="4">
2983       <boundary name="lowerlimit" value="550"/>
2984       <boundary name="upperlimit" value="INFINITY"/>
2985       <return name="returnvalue" value="4"/>
2986     </bin>
2987   </Classifier>

```

Figure 9: Step 4 of the training process in the FEATHERS configuration file: putting the new bins in the clasmod module

- Step 5: With the bins modified, the next step is to create the decision trees. This is done in the DT submodule of the AlbExpMod module. Figure 10 shows this process.

```

1512 <submodule name="DT">
1513   <param name="Active" value="1"/>
1514   <param name="NrOfObservedHouseholdSchedules" value="6266"/> <!-- 6494 -->
1515   <param name="CreateBins" value="0"/> <!-- 1 = Create bin boundaries, 0 = train decision tr
1535   <inputfile name="ObservedFile">
1536     <param name="Path" value="C:\Feathers\MobilityPlan\ObservedFile\ObservedSurveyOVG_frac_1_subset_1.obs",
1537   </inputfile>
1559   <inputfile name="PADTdataFile">
1560     <param name="Path" value="C:\Feathers\MobilityPlan\DT\PADT\PADTdata_frac_1_subset_1.bin"/>
1561   </inputfile>
1562   <outputfile name="DtreesFile">
1563     <param name="Path" value="C:\Feathers\MobilityPlan\DT\Estimation\dtrees-VL_frac_1_subset_1.dta"/>
1564   </outputfile>

```

Figure 10: Step 5 of the training process in the FEATHERS configuration file: creating the decision trees

- The model is now trained and ready to make predictions. In the final step, the PredictedFile submodule is activated in the AlbExpMod module. The PADTdataBIN & decision trees files created in the previous steps are used as input here. The PopMod module is also activated, this provides the synthetic population. A frac 2 of the synthetic population for Flanders and Brussels is used, containing 1449213

households. Figure 11 shows the configuration in the PopMod module; figure 12 shows the AlbExpMod module.

```

378 <module name="PopMod">
379 <!-- TODO: Check the assignment of names to the param values. Only for the attributes should there be numbers assigned -->
380 <param name="Active" value="1"/>
396 <inputfile name="Households">
397 <param name="Path" value="C:\Feathers\MobilityPlan\PopMod\Households_VLBXL_frac2.txt"/><!-- PopulationHH_frac1 -->
431 <inputfile name="Persons">
432 <param name="Path" value="C:\Feathers\MobilityPlan\PopMod\Persons_VLBXL_frac2.txt"/><!-- PopulationGL_frac1 -->

```

Figure 11: Configuring FEATHERS for predictions: PopMod module

```

649 <submodule name="PredictSchedules">
650 <param name="Active" value="1"/>
651 <param name="Day" value="1"/> <!-- [ 1-7 ] 1 = Monday -->
652 <param name="NrOfHouseholdSchedules" value="1449213"/>
653 <param name="Iterations" value="1"/>
671 <outputfile name="PredictedFile">
672 <param name="Path" value="C:\Feathers\MobilityPlan\Prediction\PredictedFile_frac_1_subset_1.prd"/>
673 </outputfile>
704 <projectfile name="PADTdataFile">
705 <param name="Path" value="C:\Feathers\MobilityPlan\DT\PADT\PADTdata_frac_1_subset_1.bin"/>
706 </projectfile>
707 <projectfile name="DTrees">
708 <param name="Path" value="C:\Feathers\MobilityPlan\DT\Estimation\dtrees-VL_frac_1_subset_1.dta"/>
709 </projectfile>

```

Figure 12: Configuring FEATHERS for predictions: PredictedFile submodule

Because FEATHERS is a microsimulation model, more than one run is needed to assess the performance. To determine the required number of model runs, the following formula was used (US department of transportation, 2004):

$$CI_{1-\alpha\%} = 2 * t_{(1-\alpha/2), N-1} \frac{s}{\sqrt{N}}$$

Where:

$CI_{(1-\alpha)\%}$  = (1-alpha%) confidence interval for the true mean, where alpha equals the probability of the true mean not lying within the confidence interval

$t_{(1-\alpha/2), N-1}$  = Student's t-statistic for the probability of a two-sided error summing to alpha with N-1 degrees of freedom, where N equals the number of repetitions.

S = standard deviation of the model results

When solving this equation for N, it is necessary to iterate until the estimated number of repetitions matches the number of repetitions assumed when looking up the t statistic. It was found that the model required a minimum of 8 runs to obtain a statistically valid result. To provide a small extra margin, 10 model runs are performed for each subset.

### **Statistics**

When a simulation is done, FEATHERS does not provide a readily readable file. FEATHERS generates a prediction in .prd format, from which we can draw statistics. The model provides a statistical module to do this. Using the statmod2 module, we can generate a "statisticsline", which is a text file containing the following numbers:

- Number of trips per person per day
- Kilometres travelled per person per day
- Number of trips by car as a driver per person per day
- Number of trips on foot or by bicycle per person per day
- Number of trips by public transport per person per day
- Number of trips by car as a passenger per person per day
- Kilometres travelled by car as a driver per person per day
- Kilometres travelled on foot or by bicycle per person per day
- Kilometres travelled by public transport per person per day
- Kilometres travelled by car as a passenger per person per day

## 7.2 Analysis by parameter

The first results we will analyze are the number of trips and kilometres per person per day for each transport mode. The main interest is on the distribution of the predictions within each frac. For a straightforward interpretation of the results, a box plot was selected as method of representation. In general, we would expect the spread and range to steadily increase, creating a broader peak in distribution as we move to higher fracs and thus smaller sets of training data. However, the distribution alone does not tell the entire story. The average values for each frac are analyzed for each frac, in comparison to the result for frac 1. A deviation below 5% of the frac 1 result is deemed acceptable, anything higher is seen as too strong a deviation.

### #trips

Figure 13 gives the box plot for amount of trips per person per day, for fracs 1, 2, 4, 8 and 16. With a full training data set, there is almost no spread in the results, they remain at 2,87 trips per person per day with no outliers. As the size of the training set is reduced, spread grows, as we would expect it to happen. The size of the box and the whiskers increase and the results become less reliable. There is a big increase in spread from frac 2 to frac 4. The box plots for fracs 4 & 8 indicate that the results are positively skewed. Frac 16 has, rather unexpectedly, a smaller spread and more symmetric distribution than frac 8. Figure 14 shows that the average values are all fairly close to frac 1's reference value. Frac 16, however, just exceeds to 5% threshold that was set earlier.

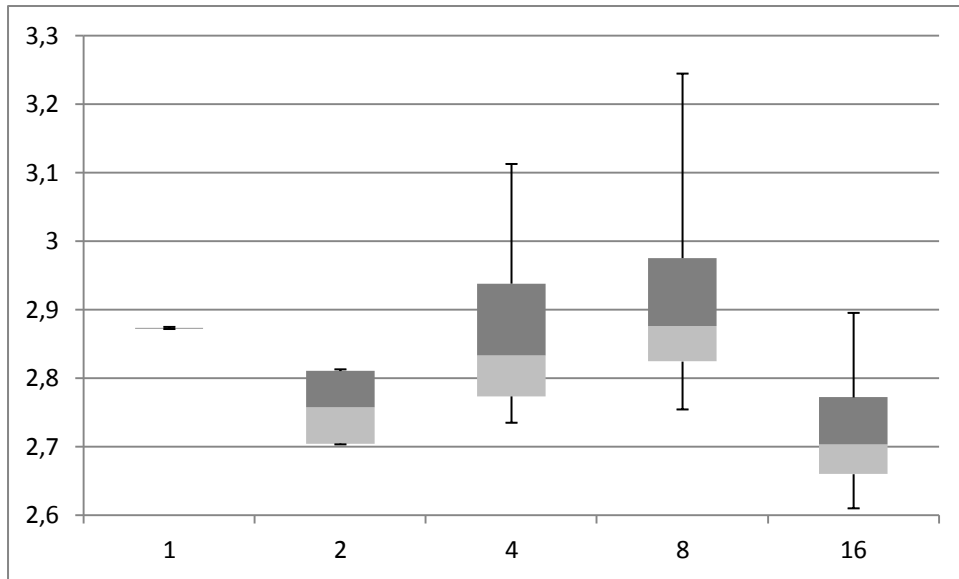


Figure 13: Figure 5: Box plot of the amount of trips per person per day, fracs 1 to 16

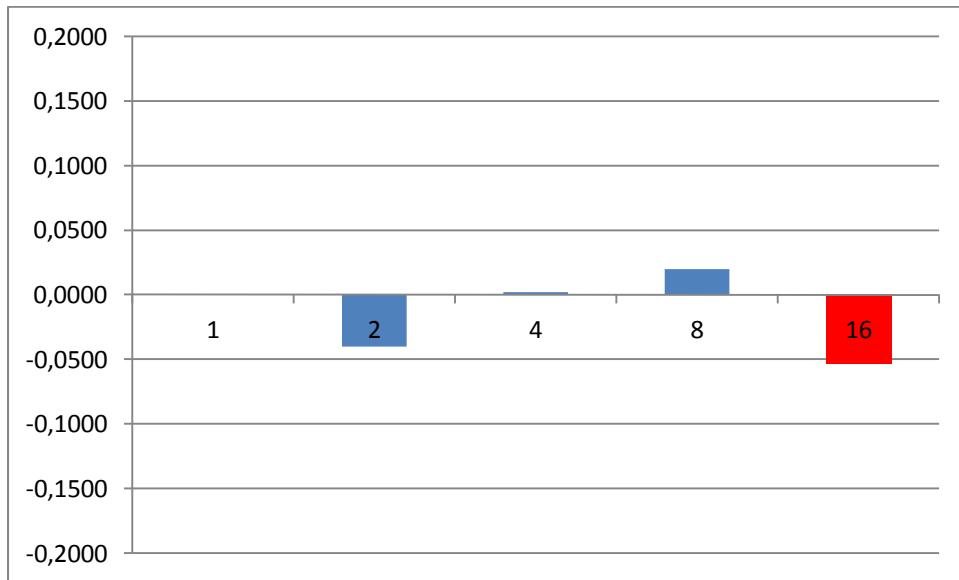


Figure 14: Deviation (%) in average amount of trips per person per day compared to reference value, fracs 1 to 16

### kilometres travelled

Figure 15 shows that the results are fairly consistent from fracs 1 to 4, with slight increases in variation. When frac 8 is reached, however, the results become very unreliable, but huge variation and outliers. We also note a positive skewedness of the data in frac 8. The distribution of frac 16 shows a smaller peak but has extreme outliers.

Figure 16 shows deviations above the threshold value for fracs 8 & 16, with the average from frac 8 showing over 10% deviation from the reference value.

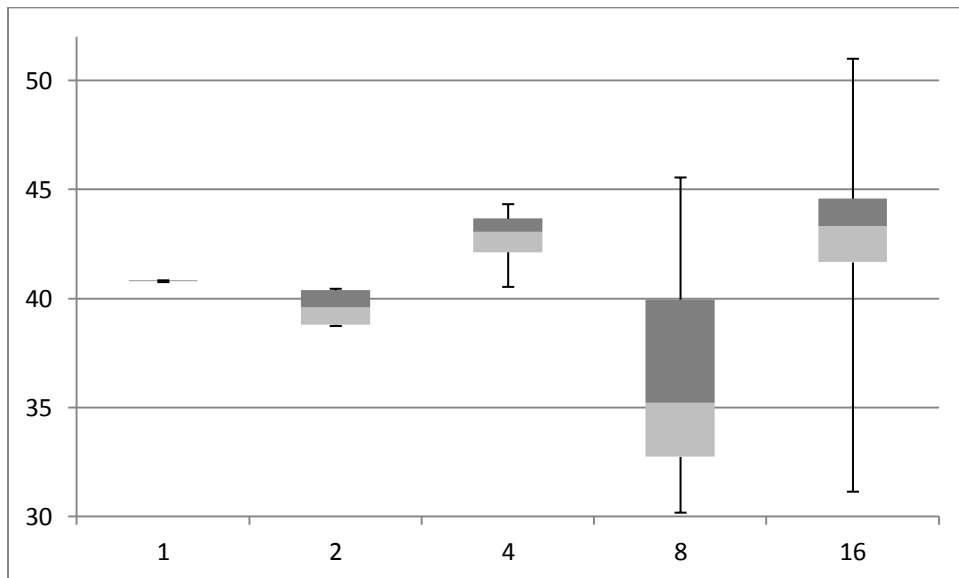


Figure 15: Box plot of the amount of kilometres travelled per person per day, fracs 1 to 16

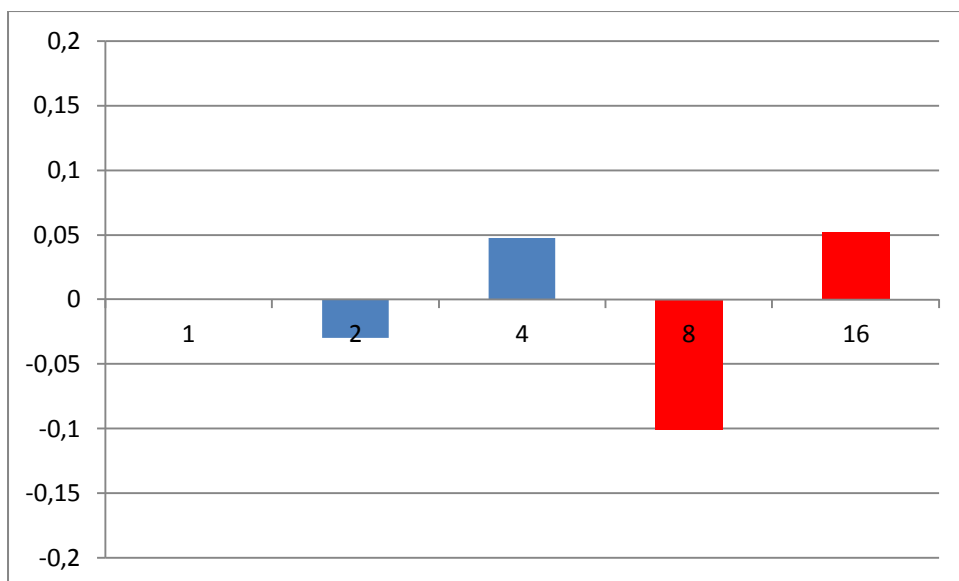


Figure 16: Deviation (%) in average amount of kilometres travelled per person per day compared to reference value, fracs 1 to 16

#### #trips by car (driver)

For this parameter, the results become unreliable starting at frac 4. As we can see in figure 17, the range of the box plot increases almost tenfold from frac 2 to frac 4, the

box size triples indicating a much broader peak in the distribution, and the results are strongly positively skewed. These results stabilise for fracs 8 & 16. As we can see in figure 18, the average values are fairly stable, all staying within a 5% range from the reference value.

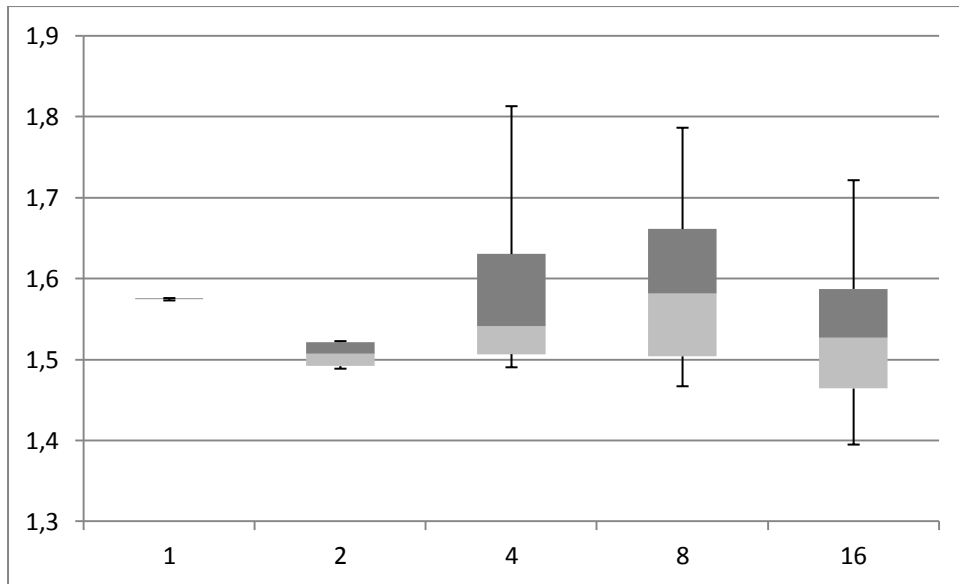


Figure 17: Box plot of the amount of trips per person per day by car (as driver), fracs 1 to 16

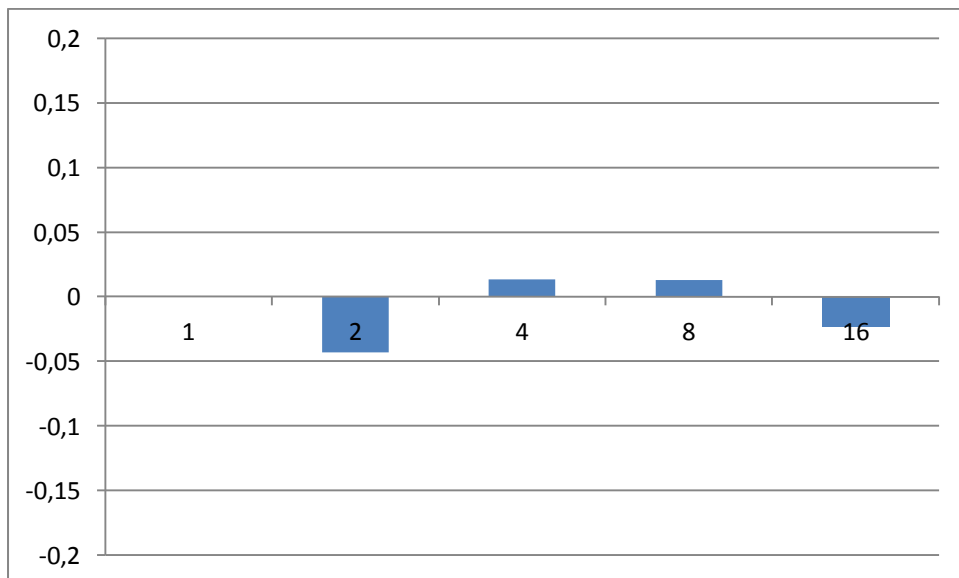


Figure 18: Deviation (%) in average amount of trips per person per day by car (as driver) compared to reference value, fracs 1 to 16

### #trips on foot or by bicycle

The results for this parameter remain fairly reliable up until frac 4, with a small increase in spread in frac 2 and even a decrease in frac 4. However, as figure 19 shows, in frac 8 we see a huge increase in spread and a major skewedness in the results. The median is almost located at the bottom of the box, indicating that the results are strongly positively skewed. The results for frac 16 are similar, with less extreme outliers but also a strong positive skewedness. Additionally, figure 20 shows that the average values for fracs 8 & 16 strongly deviate from the average found in frac 1, with both straying over 10% from the reference value. It is also interesting to note that the average found for frac 8 is over 10% *higher* than that found for frac 1, while the average for frac 16 is almost 15% *lower*. This indicates that the model is not reliable predicting this parameter with an 8<sup>th</sup> (or less) of training data.

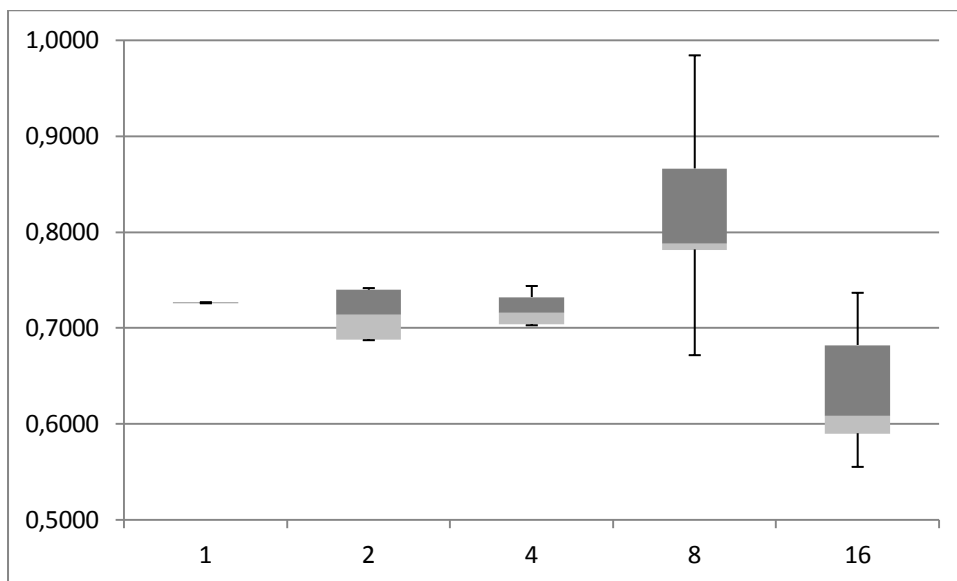
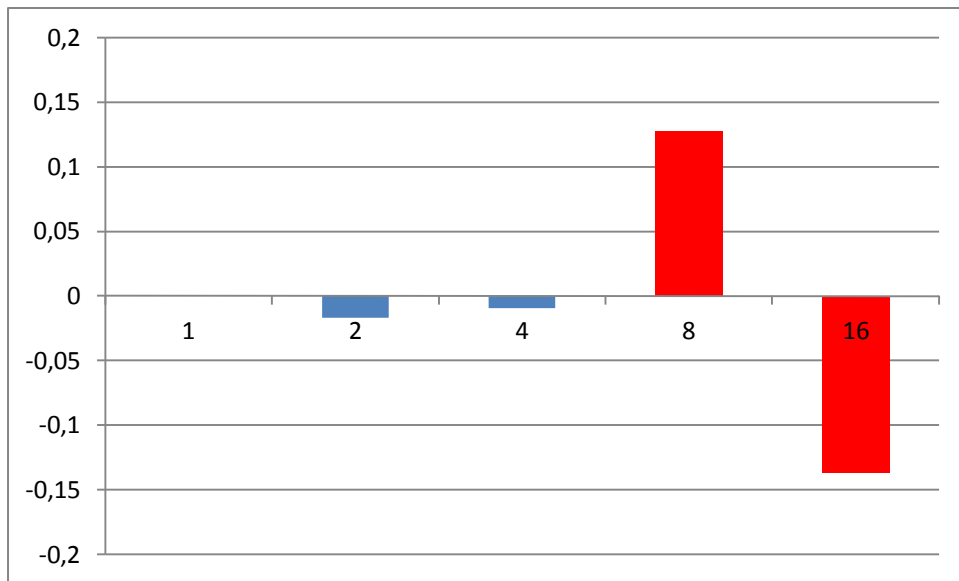


Figure 19: Box plot of the amount of trips per person per day on foot or by bicycle, fracs 1 to 16





**Figure 20: Deviation (%) in average amount of trips per person per day on foot or by bicycle compared to reference value, fracs 1 to 16**

#### #trips by public transport

Figure 21 shows a symmetric distribution with limited spread only until frac 2. From frac 4 on, the spread increases strongly and the distribution becomes asymmetric, showing a strong negative skewedness for frac 4. Frac 8 on the other hand is positively skewed, while frac 16 is again negatively skewed. However, figure 22 shows that the values deviate strongly from the reference value. Even for frac 2, the average value is already over 10% lower than the average for frac 1. Fracs 8 & 16 show results over 15% below the reference value.

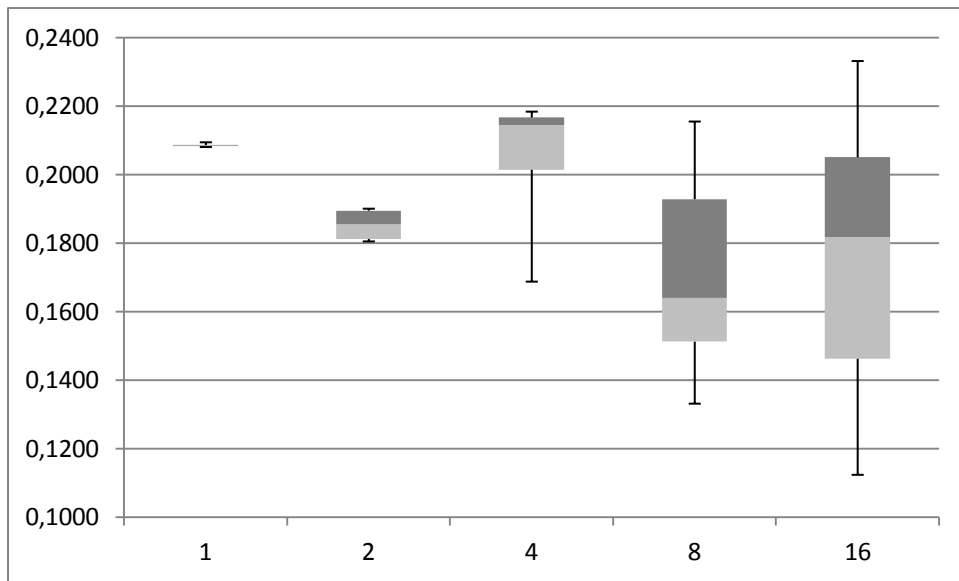


Figure 21: Box plot of the amount of trips per person per day by public transport, fracs 1 to 16

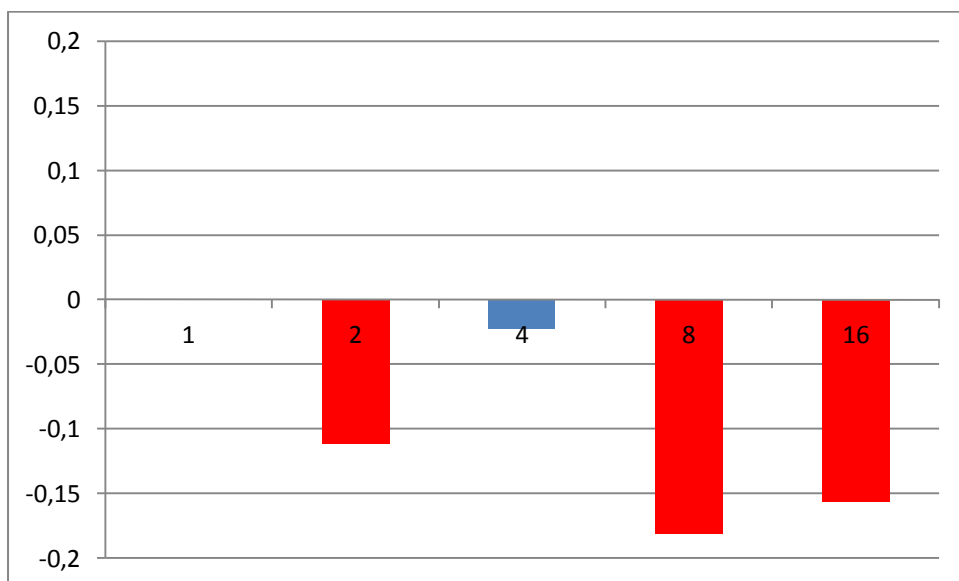


Figure 22: Deviation (%) in average amount of trips per person per day by public transport compared to reference value, fracs 1 to 16

### #trips by car (passenger)

Figure 23 shows a fairly large spread for frac 2, but the distribution is very symmetric. Frac 4 has a smaller spread but is strongly negatively skewed. Frac 8 has a very large spread and is strongly skewed in the positive direction. The range is similar for frac 16

but the data here is negatively skewed. Figure 24 shows only frac 8 just crossing the threshold value, with an average that deviates just over 5% from the frac 1 average.

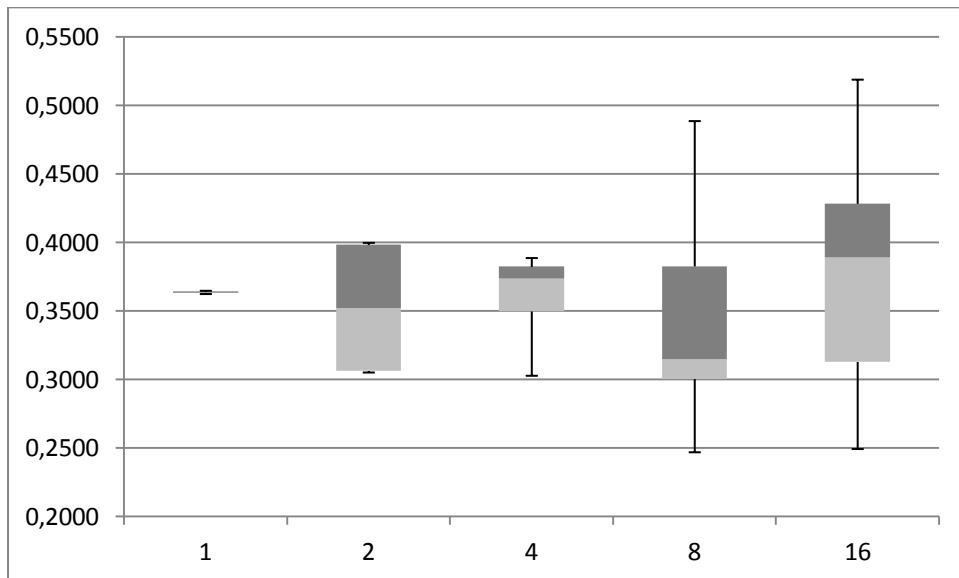


Figure 23: Box plot of the amount of trips per person per day by car as a passenger, fracs 1 to 16

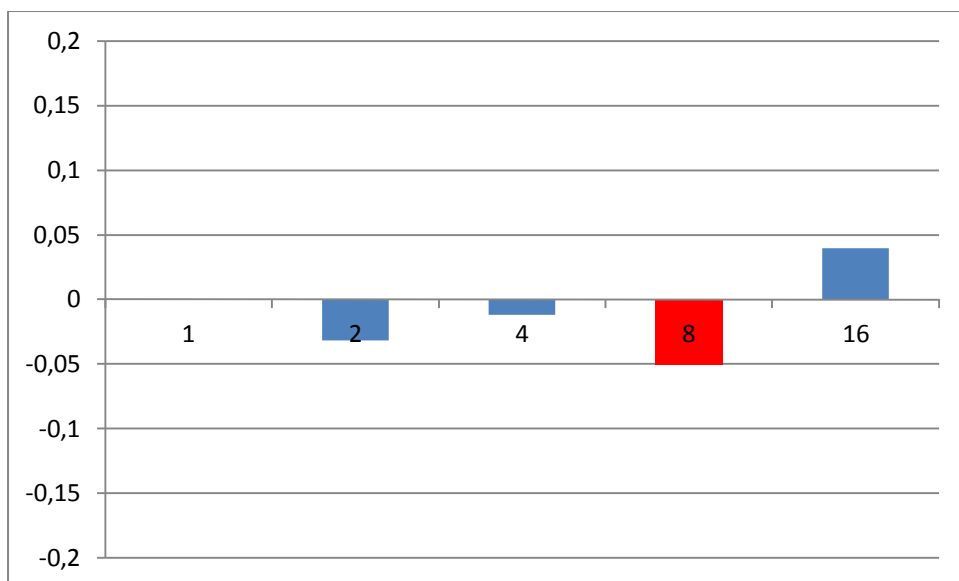


Figure 24: Deviation (%) in average amount of trips per person per day by car as a passenger compared to reference value, fracs 1 to 16

#### kilometres travelled by car (driver)

Figure 25 shows a symmetric distribution and a small spread for frac 2, which is remarkable compared to the frac 2 distribution in other parameters. We also note an

ever increasing spread going towards frac 16, combined with a positive skewedness in results. Frac 16 also displays major outliers. Figure 26 shows a high deviation in average value for frac 8, clearly above the threshold value.

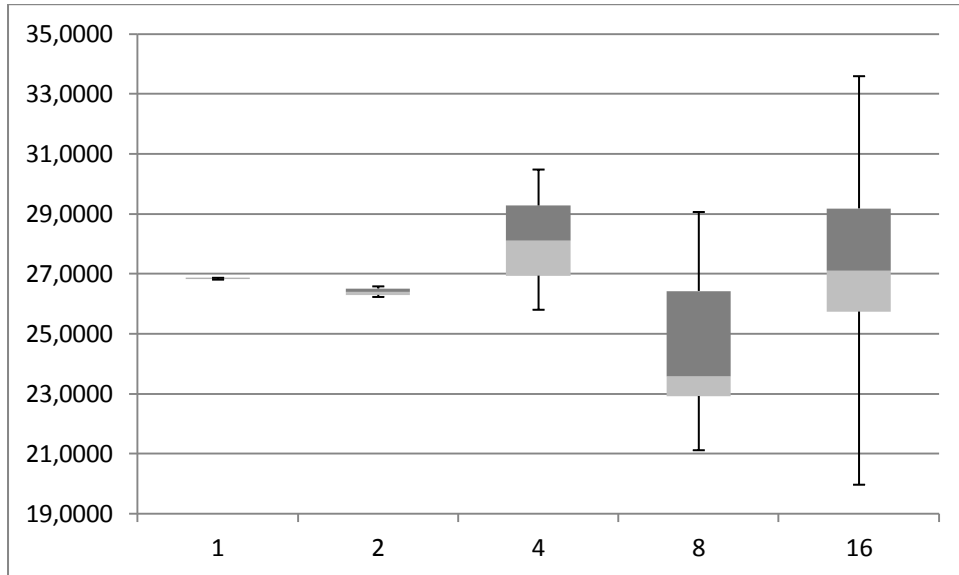


Figure 25: Box plot of the amount of kilometres travelled by car as a driver, fracs 1 to 16

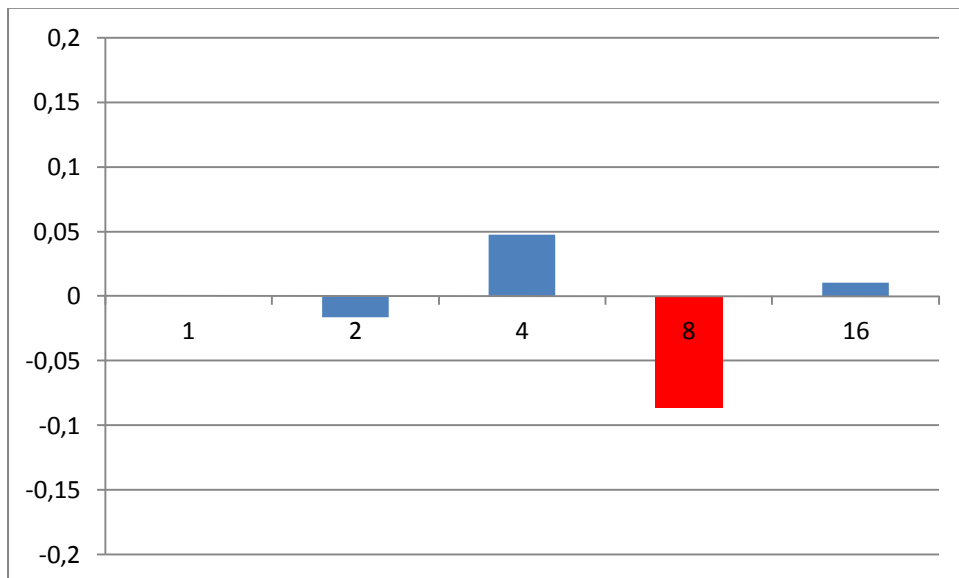
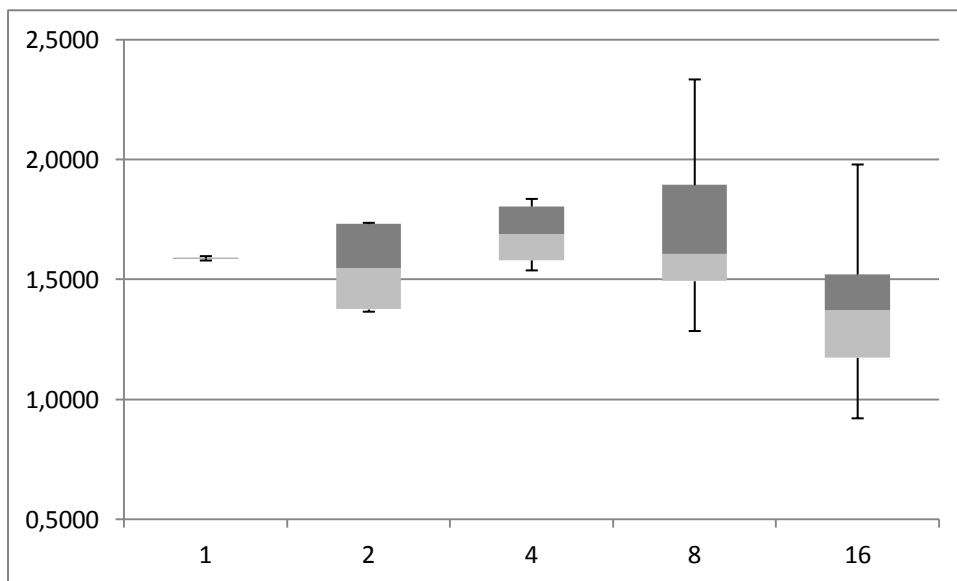


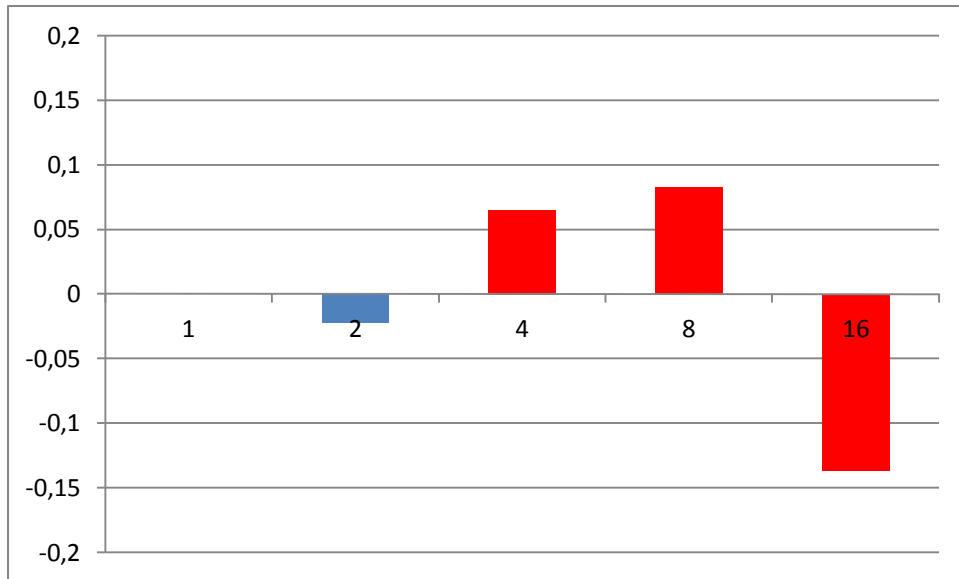
Figure 26: Deviation (%) in average amount of kilometres travelled by car as a driver per person per day compared to reference value, fracs 1 to 16

kilometres travelled on foot or by bicycle

Figure 27 shows a rather large but symmetric spread for frac 2. The range actually decreases in frac 4, but frac 8 displays a major increase in its spread and is positively skewed. Frac 16 has a similar spread as frac 8. The average values are unreliable starting from frac 4, as we can see in figure 28. Only frac 2 shows an average that is close to that of frac 1, the others are clearly above the threshold value, with the average of frac 16 deviating almost 15% from the reference value.



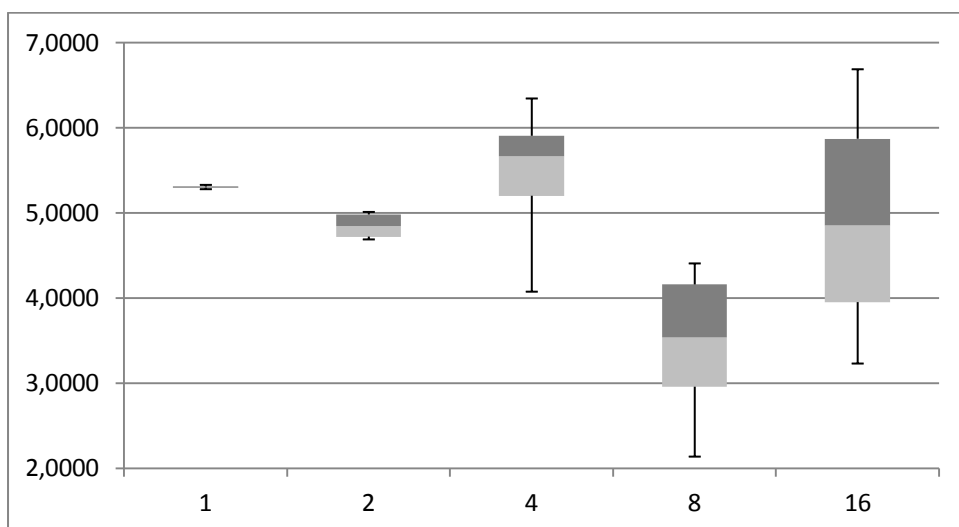
**Figure 27: Box plot of the amount of kilometres travelled on foot or by bicycle, fracs 1 to 16**



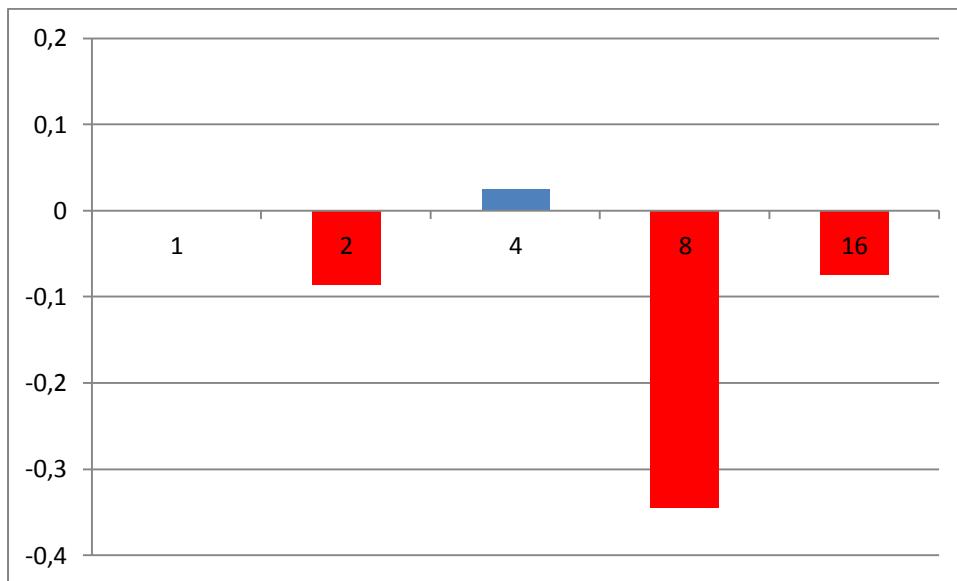
**Figure 28: Deviation (%) in average amount of kilometres travelled per person per day on foot or by bicycle compared to reference value, fracs 1 to 16**

#### kilometres travelled by public transport

As figure 29 shows, the spread increases sharply with frac 4. Furthermore, the results are asymmetric and show a negative skewedness. The distribution is comparable for frac 8, while frac 16 shows a broader peak and stronger outliers. While the distributions for frac 4 and 8 are comparable, figure 30 shows that the average value is quite reliable for frac 4, while that of frac 8 is extremely unreliable, deviating over 30% from the reference value. The average values of fracs 2 and 16 exceed the threshold as well.



**Figure 29: Box plot of the amount of kilometres travelled by public transport, fracs 1 to 16**



**Figure 30: Deviation (%) in average amount of kilometres travelled per person per day by public transport compared to reference value, fracs 1 to 16**

#### kilometres travelled by car (as passenger)

The predictions for this parameter remain fairly accurate up until frac 4, with a limited spread and a symmetric box plot, as figure 31 shows. With a frac 8, however, extreme outliers appear, and frac 16 shows a distribution with an extremely broad peak and has a strong negative skewedness. Furthermore, figure 32 shows that both frac 8 and 16 have unacceptable average values that deviate over 20% from the frac 1 average. The average value of frac 4 also falls just above the threshold value.

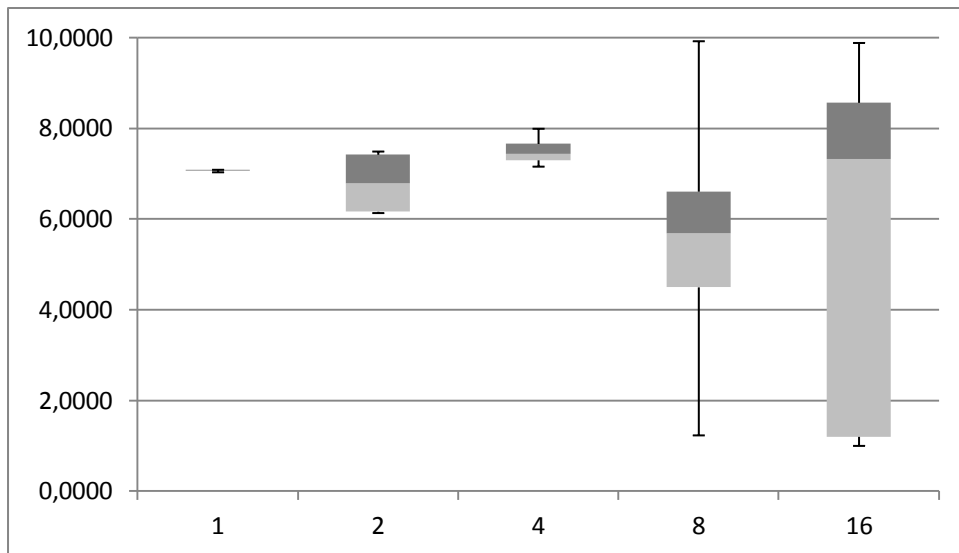


Figure 31: Box plot of the amount of kilometres travelled by car as a passenger, fracs 1 to 16

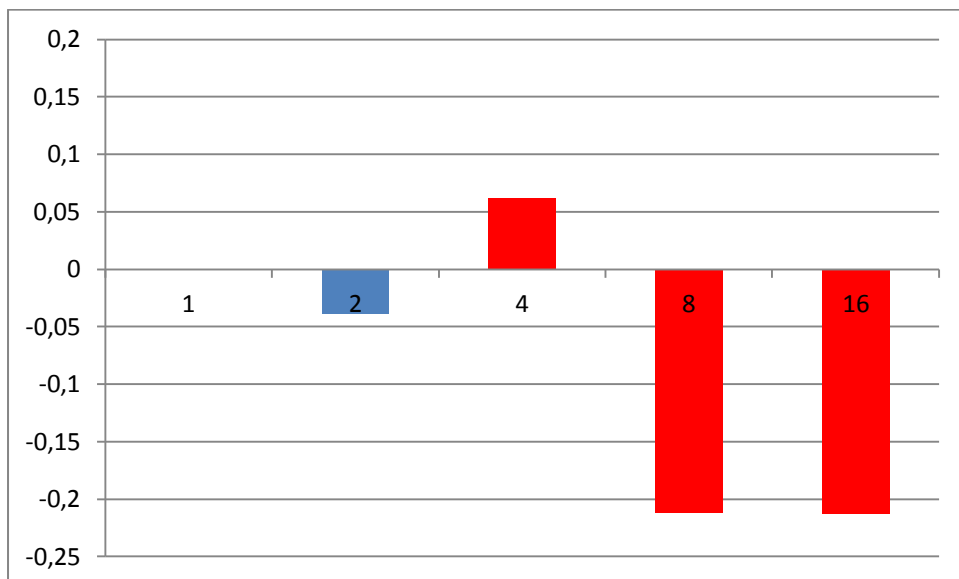


Figure 32: Deviation (%) in average amount of kilometres travelled per person per day by car as a passenger compared to reference value, fracs 1 to 16

In general, we can see that the model is less accurate for kilometres travelled than it is for amount of trips. Furthermore, it is clear that the model struggles most with predicting trips that are carried out by public transport or slow (bike, on foot) transport. Both for amount of trips and kilometres travelled, the model performs noticeably better predicting car use, compared to other transport modes. This could be explained by the fact that the car is the dominant mode of transport, by quite a big margin (Steg, 2003). This means



that the car would be present in the training data set much more than public or slow transport. It is logical that predicting public and slow transport use, both already with only a limited presence in the original data set, would suffer more from a decrease in training data than predicting car use.

### 7.3 SAM analysis

The statistics module in FEATHERS allows us to easily calculate the SAM statistic, by activating the SAM submodule. However, the input data needed to calculate SAM requires some preparation.

First, the input data sets (already divided into fracs and subsets) are split 75-25 into training data sets and validation data sets. ObservedFile (.obs) files are then created, using the training and validation data sets, respectively. Next, predictions are made using the PADTdataBIN & decision tree files that were created when training the model with the full subsets. These predictions are made with the training and validation data sets as synthetic population in the popmod module. We set the amount of iterations to 100 here, so FEATHERS repeats this process a 100 times. Finally, the ObservedFile is used as input in the statistics module, along with the predicted file, and the SAM submodule calculates how well the model was able to reproduce the observed files in the prediction. This is done for both training and validation data sets for each frac and subset. Because the SAM module analyses how well the model can reproduce the observed data, it is important that the observed and predicted data sets are of identical size. FEATHERS controls this by asking for the number of households in every step of the process. Table 4 shows the number of households for each data set and frac used. Because we used 100 iterations when making the predictions, these numbers need to be multiplied by 100 when entered in the statistical module.

Frac	# of households	
	Training	Validation
1	4698	1566
2	2348	783
4	1173	391
8	586	195
16	293	97

**Table 4: Number of households in the training and validation data sets for fracs 1 to 16**

The SAM submodule renders a text file with 3 values. The first value gives the length of the activity sequence averaged over all entries. The second value is the SAM value averaged over all entries. The third value gives a normalised SAM value, dividing the second value by the first value. We will use this normalised SAM value for our analysis.

We will analyze both the absolute SAM values and the difference between training and validation results. We would expect the SAM values to increase as the size of the data sets decreases. By comparing the SAM values of the training data with the validation data for each subset, we can analyze how good the model is predicting data it has never seen, compared to its performance predicting data that was used in training the model. We would also expect this difference to increase as smaller data sets are used.

Figure 33 shows the average SAM values per frac for both training and validation data sets. The SAM values of the validation data behave more or less as expected, clearly increasing when confronted with small input data sets in fracs 8 and 16. The average values for the training data are more erratic, decreasing and increasing in turn, but not showing major differences.

Figure 34 shows the average difference per frac between the validation and training SAM values (validation value – training value). One interesting observation is that the difference in SAM values in frac 2 is significantly lower than in frac 1. The SAM values for the validation data in frac 2 are actually lower than the SAM values for the training data, meaning that the model performs better predicting unseen data than when reproducing data that was used for training. The results for fracs 8 and 16 are more in line with

expectations: a strong increase is noted in the discrepancy between validation and training values, indicating that the models ability to predict unseen data suffers significantly when the data sets become very small.

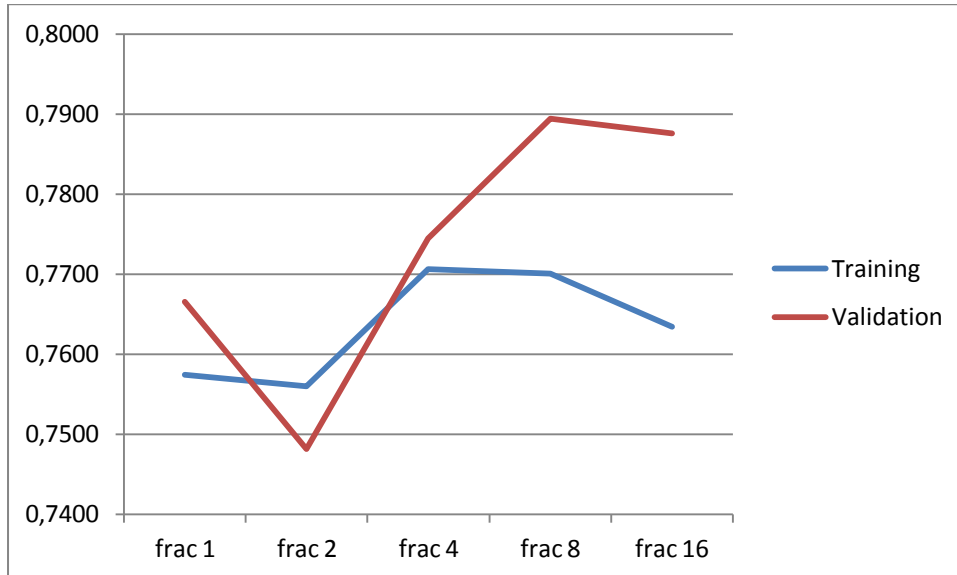


Figure 33: Average SAM values for training and validation data sets

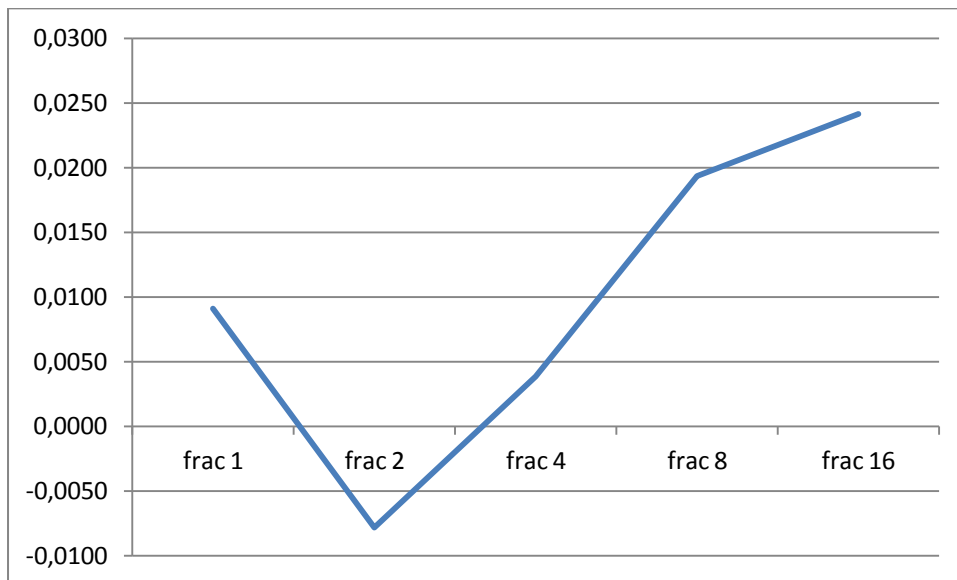


Figure 34: Average difference in SAM values between validation and training sets

Analysis of the individual difference between validation and training SAM values for each subset again shows the inconsistent and unreliable performance of the model at fracs 8 and 16. Figures 34 and 35 show the difference between validation and training SAM values for fracs 1 to 8 and frac 16, respectively.

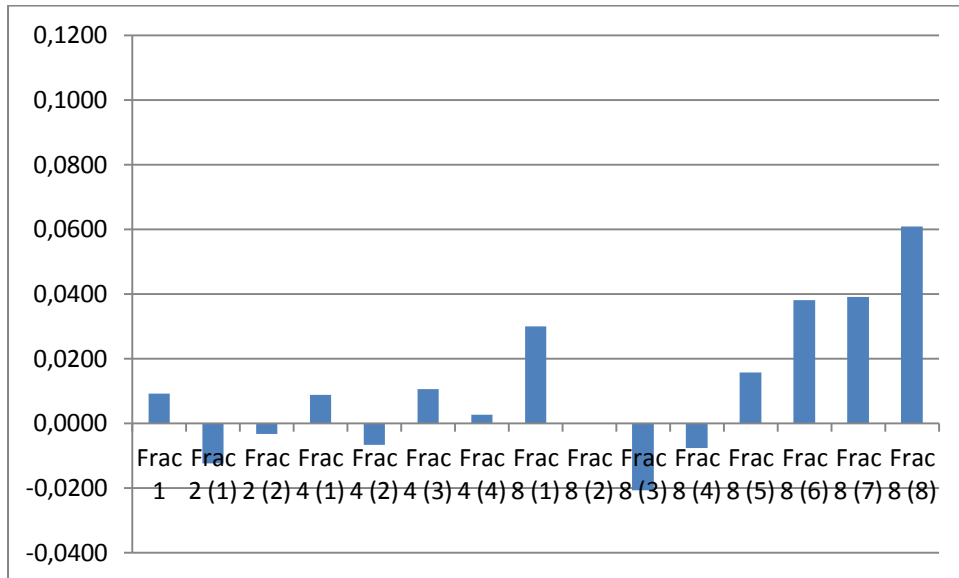


Figure 35: Difference in SAM values between validation and training sets for each subset, fracs 1 to 8

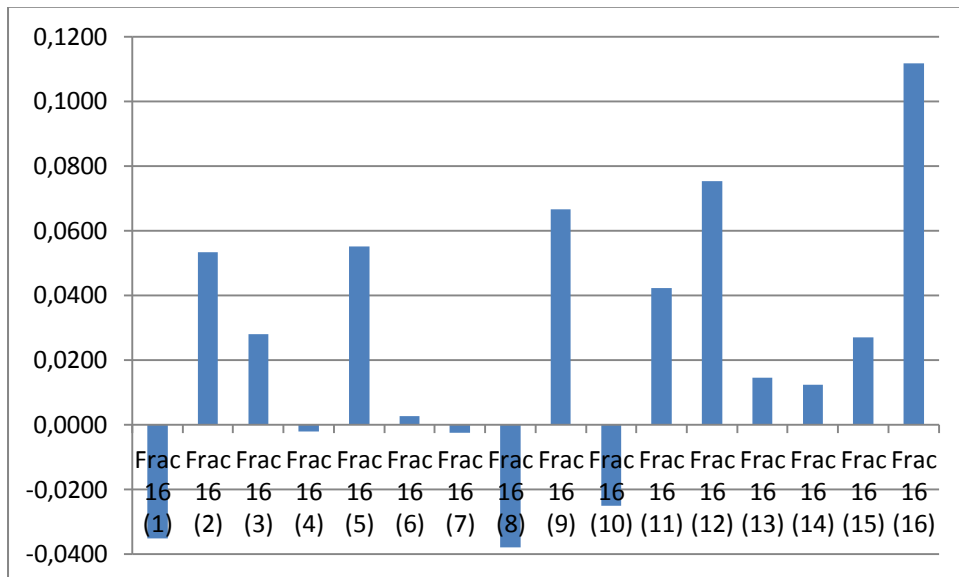


Figure 36: Difference in SAM values between validation and training sets for each subset, frac 16

As we can see, for fracs 1 to 4, the difference nowhere even comes close to 0,02. When only an 8<sup>th</sup> of the data is used, we see that the range of the results sharply increases, with outliers to 0,04 and even 0,06. Frac 16 shows major outliers in positive and negative direction, with a range from -0,04 to 0,11 SAM.

The SAM analysis further confirms the inconsistent and unreliable performance of FEATHERS when trained with an 8<sup>th</sup> or less of the regular input data set.

#### 7.4 CMA analysis

After focusing on the output of the model in the previous sections, it is useful to take a deeper look at the scheduling process. This means taking analyzing the performance of the decision trees, which form the core of the scheduling process. Some of the trends we discovered in the previous sections should also be visible in analysis of the decision trees.

A method of progressive sampling is applied, meaning that progressively larger samples of the original training data set are used until no more improvement is seen. The relationship between sample size and model accuracy is depicted by a learning curve. Figure 37 gives an example of a learning curve. The horizontal axis represents the sample size, varying from 0 to N. The vertical axis gives the accuracy of the decision tree algorithm for a given training set size.

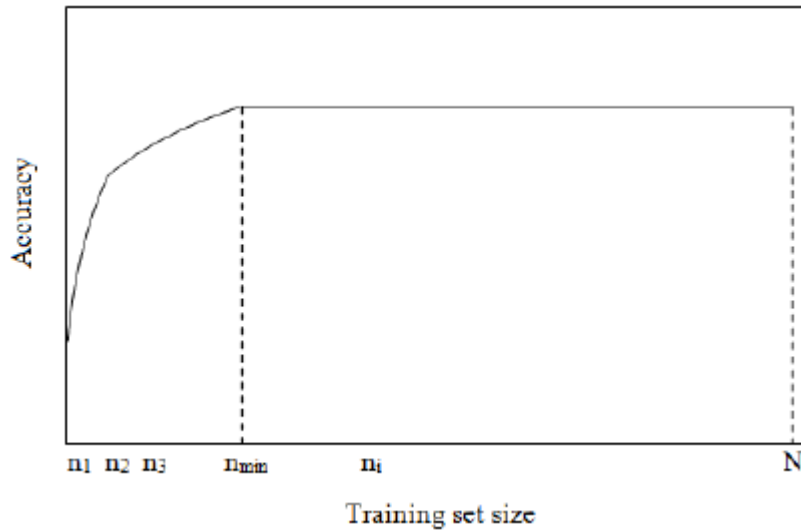


Figure 37: Learning curve

Learning curves typically have a steep section early in the curve, a decreasing slope in the middle portion and a plateau in the last part. The middle part can be missing in some curves or extremely large in others. When the plateau is reached, the accuracy does not improve further. It is possible that the plateau is never reached when the sample size is too small. For a very large sample size, it is possible that the plateau is reached very early in the curve. It is generally assumed that learning curves behave consistently and that the slope does not increase (Catlett, 1991).

When a learning curve reaches its plateau, it has converged. At this point, a smaller data set would result in lower accuracy, while a bigger data set would not result in an increased accuracy. We label this size of data set,  $n_{min}$ , as the smallest sufficient training set. By applying progressive sampling, we will attempt to determine  $n_{min}$  empirically.

Different kinds of progressive sampling are possible. Arithmetic sampling uses the following sequence:

$$S_a = n_0 + (i \cdot n_\delta) = \{n_0, n_0 + n_\delta, n_0 + 2 \cdot n_\delta, \dots, n_0 + k \cdot n_\delta\}$$

An example of arithmetic sampling is  $\{100, 200, 300, \dots, n_k\}$ .

An alternative sampling sequence is called geometric sampling. It uses the following sequence:

$$S_g = a^i.n_0 = \{n_0, a.n_0, a^2.n_0, \dots, a^k.n_0\}$$

Arithmetic sampling is chosen because it is more straightforward and because we are not dealing with very large data sets.

Because we will not be working with subsets here, a different method of sampling is required, as one can obtain many samples of the same size by randomly selecting instances of the original data set. Therefore, 30 different samples of the same size are compiled for each step of the sampling procedure.

The 'Confusion Matrix Accuracy' (CMA), explained in section 5.3, will be used to evaluate the performance of the decision trees. Using 30 samples per step means that each sample size will generate 30 different decision trees and thus 30 different CMAs. A learning curve will be created by chaining the averages of the normalized values of these 30 CMAs. The normalization is necessary to compare learning curves. Table 4 gives a summary of all normalized average CMA values. The points on the learning curve are used to estimate a tangent line for each of the 30 samples. The slope of this tangent line will be compared to zero, because the tangent line of a learning curve is approximately zero when the plateau is reached.

The following criterion is set to determine where convergence is reached: convergence is reached when more than 90% of the 30 sample CMA values at a given point on the learning curve have a tangent smaller than or equal to 0.25 degrees. Table 5 gives the convergence percentage of the learning curves for all discrete decision trees.

nr	decision tree	10%	20%	30%	40%	50%	60%	70%	80%	90%
1	Inclusion work	100,1	100,3	100,0	100,0	100,0	100,1	100,1	100,1	100,0
2	Number of episodes	97,0	97,7	97,3	97,8	98,4	98,3	98,5	99,1	99,6
3	Location, same as previous	84,3	86,5	92,4	93,8	95,5	97,1	97,8	98,9	99,5
4	Location, in/out home	92,9	94,6	96,2	96,5	97,6	98,2	98,4	99,1	99,6
5	Location, order (1)	87,0	90,7	91,7	93,5	95,2	96,0	97,5	97,9	98,6
6	Location, nearest of order	92,2	95,0	95,7	96,6	96,5	97,5	98,1	98,5	99,1
7	Location, distance band (1)	77,1	79,4	85,1	87,5	89,7	92,4	94,4	96,1	96,5
8	Location, order (2)	85,5	89,2	92,0	93,1	94,0	96,2	97,6	98,6	99,7
9	Location, distance band (2)	78,1	84,6	89,0	93,3	95,3	96,6	97,8	98,4	99,5
10	Transport mode (1)	90,6	93,3	93,8	94,8	95,3	96,8	97,1	97,7	99,3
11	Inclusion fixed	99,2	99,7	99,7	99,8	99,8	99,8	99,9	99,9	100,0
12	number of episodes	94,4	96,7	98,3	98,2	98,3	98,9	99,4	100,0	100,5
13	Chaining, work	92,5	94,4	95,5	96,8	98,1	98,9	99,6	100,0	100,9
14	Location, same as previous	96,9	97,0	99,2	99,9	100,1	100,2	100,3	100,3	100,4
15	Location, distance-size class	75,7	83,0	89,5	93,8	97,5	100,4	100,6	101,2	101,6
16	Inclusion flexible	99,2	99,3	99,4	99,5	99,6	99,7	99,8	99,8	99,9
17	Duration	93,8	95,9	97,3	98,3	99,2	99,7	99,8	100,0	100,3
18	Timing	85,7	93,3	96,4	98,3	99,3	99,6	99,5	99,7	99,7
19	Chaining	94,2	98,7	99,3	99,7	100,2	100,1	100,1	100,2	100,1
20	Transport mode (2)	88,5	93,1	95,8	97,5	98,6	99,7	99,7	99,7	99,7

**Table 5: Normalised average CMA values, for each progressive sampling step from 10% till 90% (in %)**



nr	decision tree	10%	20%	30%	40%	50%	60%	70%	80%	90%
1	Inclusion work	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0
2	Number of episodes	90,0	100,0	97,0	97,7	100,0	100,0	97,7	100,0	100,0
3	Location, same as previous	60,0	13,3	53,3	70,0	63,3	77,7	86,7	96,7	100,0
4	Location, in/out home	73,3	80,0	100,0	96,7	93,3	100,0	100,0	100,0	100,0
5	Location, order (1)	40,0	60,0	63,3	63,3	80,0	70,0	93,3	86,7	76,7
6	Location, nearest of order	50,0	86,7	93,3	100,0	100,0	100,0	100,0	100,0	100,0
7	Location, distance band (1)	60,0	10,0	50,0	56,7	50,0	56,7	53,3	76,7	30,0
8	Location, order (2)	36,7	43,3	80,0	83,3	63,3	63,3	96,7	96,7	100,0
9	Location, distance band (2)	20,0	16,7	23,3	73,3	73,3	90,0	100,0	86,7	100,0
10	Transport mode (1)	50,0	86,7	90,0	100,0	76,7	100,0	96,7	93,3	100,0
11	Inclusion fixed	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0
12	number of episodes	53,3	56,7	96,7	96,7	90,0	93,3	96,7	93,3	100,0
13	Chaining, work	53,3	66,7	70,0	76,7	76,7	96,7	96,7	86,7	100,0
14	Location, same as previous	90,0	63,3	100,0	100,0	100,0	100,0	100,0	100,0	100,0
15	Location, distance-size class	6,7	16,7	23,3	26,7	40,0	93,3	86,7	83,3	100,0
16	Inclusion flexible	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0
17	Duration	60,0	90,0	93,3	96,7	100,0	96,7	100,0	100,0	100,0
18	Timing	3,3	26,7	76,7	96,7	100,0	100,0	100,0	100,0	100,0
19	Chaining	10,0	93,3	100,0	100,0	100,0	100,0	100,0	100,0	100,0
20	Transport mode (2)	16,7	40,0	63,3	90,0	90,0	100,0	100,0	100,0	100,0

**Table 6: Percentage of the number of samples with a tangent smaller than 0.25 degrees, for progressive sampling step from 10% till 90%**

Some interesting things stand out when looking at table 5. Firstly, 3 decision trees reach convergence immediately at only 10% of the total training data set. Apparently, these decision trees need very little data in order to accurately make predictions. These 3 trees constitute the inclusion decision trees (inclusion work, inclusion fixed, inclusion flexible). Figure 38 gives the learning curve of one of these 3 decision trees, the inclusion work tree. It is clear that the learning curve is nearly flat, indicating that the plateau is reached almost instantly, and additional training data does not result in an improved CMA.

Most other learning curves reach their plateau somewhere between sample fractions of 10% and 90%. Figure 39 shows an example of this with decision tree 18, which determines timing. Finally, there are two decision trees that do not reach convergence at

all. These decision trees need more data than available in order to reach their maximum accuracy. Figure 40 shows an example of this with decision tree 5. As we can see, the learning curve keeps on elevating and no plateau is reached. The two decision trees that do not reach are both decision trees that make location choices. This is not surprising, considering the difficult nature of location choices compared to other choices, such as inclusion.

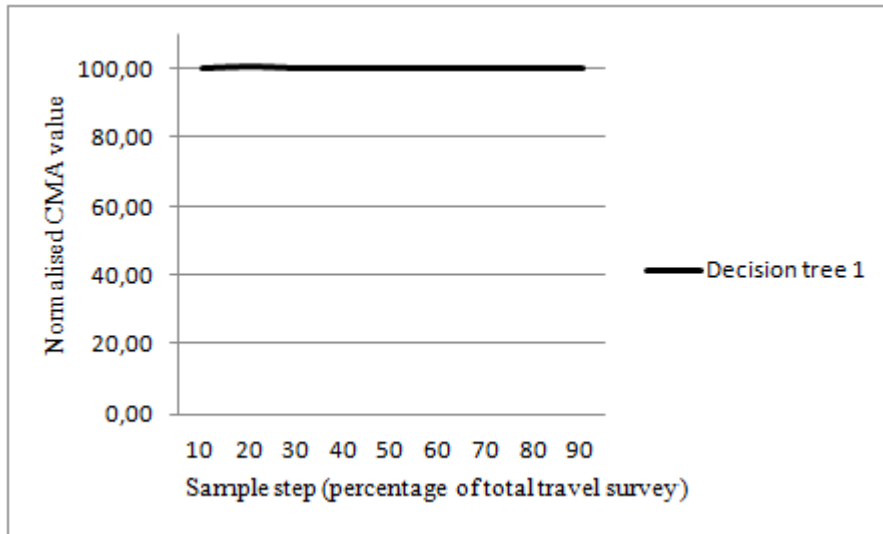


Figure 38: Learning curve of decision tree 1 (inclusion work)

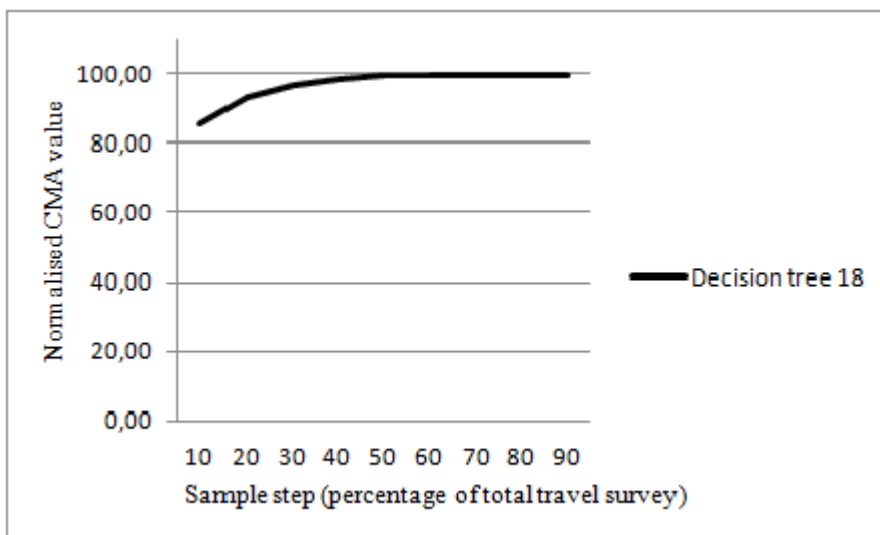


Figure 39: Learning curve of decision tree 18 (Timing)

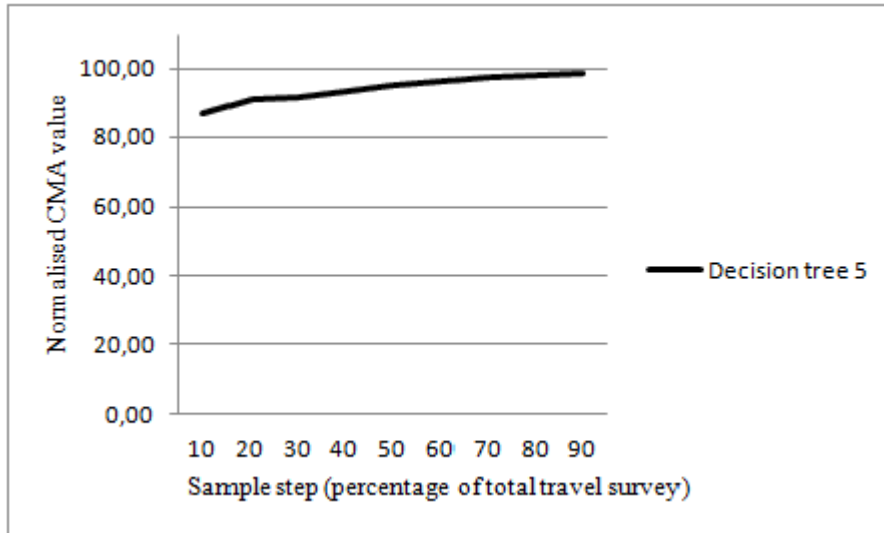


Figure 40: Learning curve of decision tree 5 (location, order)

---

## Chapter 8: Discussion

---

### 8.1 Conclusion

The goal of this master thesis was to investigate the performance of the FEATHERS model when trained with smaller amounts of training data. This was done by a form of progressive sampling where increasingly smaller fractions (fracs) of the original data set were used to train the model. Several important parameters were generated from the output and the distribution and goodness-of-fit were analyzed. A more in-depth analysis was performed on the performance of the decision trees.

The analysis by parameter shows that the model performance gets progressively worse as the training data set gets smaller, as would be expected. This decrease in accuracy is moderate when a frac 2 (50% of the data set) is used. The distribution shows an increase in range but remains fairly symmetric, and the average values stay below 5% deviation for all but 2 of the 10 parameters. With this amount of training data, the model could be useful, depending on the purpose. At frac 4 (25% of the data set), however, the model performance starts to suffer heavily. The distributions become skewed, often show a broad peak and big outliers are no exception. The average value of the parameters remains acceptable however, with again only 2 parameters showing a deviation over 5%. The model performance really crashes from frac 8 (12,5% of the data set) onward. The distributions are strongly skewed, extreme outliers are observed for every parameter and the average values show major deviations, some even up to 30%. It is clear that the model is totally inconsistent and unreliable at this stage.

The SAM analysis further confirms the findings of the parameter analysis. At fracs 8 and 16, the SAM values become very large, indicating that the model cannot reproduce the training data well. Additionally, the difference between the training and validation values becomes bigger and bigger, which shows that the model gets progressively worse at predicting data it has not yet seen before. Big differences are also observed between the different subsets of fracs 8 and 16, once again confirming the highly inconsistent performance of the model with such a limited amount of training data.

The analysis of the decision trees clearly shows that some trees can function very well with a very limited amount of training data, while others struggle even with extensive amounts of data. The inclusion trees, responsible for deciding whether or not to include an activity in the schedule, appear to need very little training data and perform perfectly with only 10% of the training data. The opposite is true for the location trees, which decide on the location of activities and trips. Their performance is not satisfactory even when trained with a large amount of training data. The other investigated decision trees reach their maximum accuracy at various points between 10% and 90% of the training data set.

In general, we would advise against the use of smaller data sets to train the FEATHERS model. The model accuracy suffers quite quickly from a decrease of training data. While the initial loss of accuracy is rather moderate for most parameters, the location model is the first to suffer. Since this location model is an essential part of most activity-based models, given their use in transportation research, this would not be acceptable. Many of these models are used by policy makers, and only the highest quality of performance is acceptable when important policy decisions are to be based upon the predictions of the model. For specific purposes that do not require accurate location estimates, the model performance could be deemed acceptable when trained with 50% of the training data. However, it is unlikely that many applications requiring such data exist. We would advise against any use of the model when trained with 25% or less of the training data set. The results indicate highly unstable and inconsistent performances at that stage.

---

## References

---

Proost, S. And K. Van Dender, 2011, *What long-term road transport future? Trends and policy options*, Review of Environmental Economics and Policy, Volume 5, issue 1

Goodwin, P., 2004, *The economic costs of road traffic congestion*, UCL (University College London), The Rail Freight Group: London, UK.

Zaidi, A. and K. Rake, 2001, *Dynamic microsimulation models: a review and some lessons for SAGE*, Department of social policy London school of economics

Arentze, T.A. and H.J.P. Timmermans, 2000, *Albatross: A learning-Based Transportation Oriented Simulation System*, European Institute of Retailing and Services Studies, 2. conceptual considerations

Arentze, T.A. and A. Schoemakers, 2004, *Albatross: gevoeligheden van een nieuw activiteiten – verplaatsingsmodel in kaart gebracht*, Colloquium Vervoersplanologisch Speurwerk 2004, 25th & 26th of November, Zeist, The Netherlands

Potter, K., 2006, *Methods of presenting statistical information: the box plot*, University of Utah

Joh, C-H., T. Arentze, H. J.P. Timmermans, 2001, *Multidimensional Sequence Alignment Methods for Activity-Travel Pattern Analysis: a Comparison for Dynamic Programming and Genetic Algorithms*, Geographical Analysis, Volume 33, Issue 3, p. 247-270, July 2001

Arentze, T., F. Hofman, H. Van Mourik, H. Timmermans, 2000, *ALBATROSS multiagent, rule-based model of activity pattern decisions*, Transportation Research Record 1706, Volume 1706/2000, p. 136-144

Doran, H.C. and P.B. Van Wamelen, 2010, *Application of the Levenshtein distance metric for construction of longitudinal data files*, Educational Measurement: Issues and practice, Volume 29, No. 2, p. 13-23

Kohavi, R. and F. Provost, 1998, *Glossary of terms: Machine Learning*. Kluwer Academic Publishers, Boston

Catlett, J., 1991, *Megainduction: Machine learning on very large databases*, PhD thesis, School of Computer Science, University of Technology, Australia, Sydney

Janssens, D., E. Moons, E. Nuyts, G. Zwerts, 2009, *Onderzoek verplaatsingsgedrag Vlaanderen 3 (2007 - 2008)*, Universiteit Hasselt: Instituut voor Mobiliteit (IMOB)

Steg, L., 2003, *Can public transport compete with the private car?*, IATSS research, Vol. 27, No. 2, p.27-35

## **Auteursrechtelijke overeenkomst**

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:

**Sensitivity analysis of the Feathers activity-based model for Flanders**

Richting: **Master of Management-Management Information Systems**

Jaar: **2012**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

**Vanderheyden, Ward**

Datum: **21/08/2012**