M 2014

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

ESTIMATING FUEL CONSUMPTION FROM GPS DATA

AFONSO VILAÇA BASTOS SILVA

DISSERTAÇÃO DE MESTRADO APRESENTADA
À FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO EM
ENGENHARIA DA INFORMAÇÃO

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Estimating Fuel Consumption from GPS Data

Afonso Vilaça Bastos Silva

Master in Information Engineering

Supervisor: Professor Ana Aguiar

Second Supervisor: Professor Carlos Soares

October 31, 2014

Estimating Fuel Consumption from GPS Data

Afonso Vilaça Bastos Silva

Master in Information Engineering

Approved by . . . :

President: Professor António Pedro Aguiar
Referee: Professor Paulo Cortez
Referee: Professor Carlos Soares

October 31, 2014

Resumo

Uma grande fonte de emissão de CO_2 para a atmosfera terrestre é o consumo de combustível por parte de veículos rodoviários. Este fenómeno deve-se muito ao facto do automóvel ser o meio de transporte preferido para curtas e médias distâncias para a generalidade da população mundial, dada a sua comodidade e flexibilidade. Com o desenvolvimento da tecnologia móvel e da sua propagação no mercado, têm vindo a ser desenvolvidas muitas aplicações que, através de dados de mobilidade, dão um grande suporte ao utilizador em termos de orientação e gestão das suas viagens.

Sendo o GPS a principal fonte de dados de localização utilizada em smartphones, soluções que sugiram ao condutor percursos ou comportamentos que diminuam do consumo de combustível podem ser desenvolvidas tendo como base um modelo que estime o consumo de combustível de forma instantânea através de dados da sua mobilidade.

Através de uma recolha de dados com a utilização de smartphones com GPS embebido e em comunicação com um dispositivo externo (OBD) que fornece informação sobre o consumo de combustível, foi desenvolvido um modelo de regressão que estima o consumo de combustível instantâneo para veículos ligeiros movidos a gasolina ou diesel, onde as principais variáveis são a velocidade e a aceleração instantâneas do veículo e a inclinação da estrada.

O modelo é fiável e robusto a diferentes tipos de veículos e percursos urbanos, podendo o utilizador ter resultados mais precisos, caso sejam fornecidas algumas especificações básicas do veículo.

Abstract

A great source of CO_2 emission for the terrestrial atmosphere is the fuel consumption by road vehicles. This phenomenon owes very much to the fact of the car being the preferred mode of transportation for short and medium distances, for the majority of the world population, given its convenience and flexibility. With the development of mobile technology and its spread in the market, many applications have been developed that, from mobility data, give a great user support in terms of guidance and management of their travels.

Being GPS the main source of location data used in smartphones, solutions that suggest to the driver paths or behaviours that reduce fuel consumption can be developed based on a model that estimates fuel consumption instantaneously across his data mobility.

Through a data gathering with the use of smartphones with embedded GPS in communication with an external device (OBD) that provides information on fuel consumption, it was developed a regression model for light-duty vehicles moved by gasoline or diesel, that estimates the instantaneous fuel consumption, having as main variables the instantaneous speed and acceleration of the vehicle and the road gradient.

The model is reliable and robust to different types of vehicles and urban routes, and the user can get more accurate results if some simple vehicle specifications are provided.

Acknowledgments

In first place I want to thank my supervisors. I thank Professor Ana Aguiar because she was who proposed me the development of this project and gave me the fundamental orientation, especially in the startup phase. To Professor Carlos Soares for any tips, suggestions and explanations during all the data mining process. Without him this thesis would not have as much scientific depth. I also thank Vitor Ribeiro. He started this project a year before and was the one who explained the process of data collection to me. I have a special thanks to Instituto de Telecomunicações - Porto, for supplying materials for data collection and for my integration in SenseMyCity project. I thank João Rodrigues for technical assistance in the data collection phase and all the problems he could solve. I also thank him for having taught me to work with the database. I owe a thank to my professors of the Masters in Information Engineering (MEINF), which gave me the foundation for the development of this dissertation, and showed me the enjoyment that can be drawn from a data mining project.

This work exists only through the volunteering of several people who collected data during their daily trips: colleagues, friends, friends of friends, family and neighbours. I thank each of them very much by the good will and withstand at my insistence. I have a particular gratitude to my fellows from MEINF, specially to those who accompanied me in the second year. Carlos, that participated in the data collection and was giving me advice during the thesis processes. Mariana, with whom I began to sift through the data for the project of Data Mining course. And Majid, for all the precious coffee breaks we had while working at FEUP. I also thank my friend Pedro for the revision of the English text.

During this thesis, many friends, almost innumerable, accompanied me and supported me with words of encouragement, with advice, with patience while I was in the "burrow", with moments of fun, with prayers and in moments that strengthened my faith. To each of them I thank very much. Finally I have to thank my family for all the support. I thank each member, they all gave me strength. But some are worth mentioning. My brothers, Carlota and Leonardo, for our unity in every moment. My grandmother, that host me with huge treat in moments of intensive work. And my parents, for their support not only during this thesis, but throughout the investment and dedication they had with my education and for the love they have always given me. To them I dedicate this dissertation.

Afonso Vilaça

"It is not the amount of knowledge that makes a brain. It is not even the distribution of knowledge. It is the interconnectedness."

James Gleick, *The Information: A History, a Theory, a Flood*

Contents

List of Figures

List of Tables

Abbreviations and Symbols

A	Acceleration
AANN	Average Artificial Neural Networks
AL	Absolute Load
ANN	Artificial Neural Networks
BE	Baseline Error
CAFR	Corrected Air to Fuel Ratio
CART	Classification and Regression Tree
CL	Calculated Load
CMAF	Corrected Mass Air Flow
FC	Fuel Consumption
FD	Fuel Density
FEV	Fully Electric Vehicle
GPS	Global Positioning System
I	Inclination
IAT	Intake Air Temperature
IT	Information Technology
LPG	Liquefied Petroleum Gas
MAE	Mean Absolute Error
MAF	Mass Air Flow
MAP	Manifold Absolute Air Pressure
MARS	Multivariate Adaptive Regression Spline
MOE	Measure of Effectiveness
OBD	On Board Diagnostics
PPR	Projection Pursuit Regression
R^2	coefficient of determination
RBF	Radial Basis Function
RF	Random Forest
RMSE	Root Mean Squared Error
RPM	Revolutions Per Minute
S	Speed
SP	Specific Power
SVM	Support Vector Machines
VIN	Vehicle Identification Number

Chapter 1

Introduction

Today's society is enormously dependent on technology. This phenomenon is particularly present in mobility. Mobility has an impact in the daily life of individuals and companies. A type of mobility is related to data. The growing variety of ways to access the internet, to connect to a Global Positioning System (GPS), to be in contact with another person that is far away, to be informed by some sensors system about the status of a personal property, like a car or a house, or even to be able to access a TV channel, are indicators of how information is everywhere and increasingly accessible to more and more people.

Another type of mobility is related to people. The variety and quality of transportation alternatives and infrastructures are a main issue in urban management and development. This favours not only the mobility for long or medium periods of time, like migrations or vacations, but also daily life transportation. The technological growth should be accompanied by solutions that improve the quality of life both on personal and collective level. Mobility directly implies energy consumption. Technological and mobility growth must be controlled in order to be in harmony with a sustainable environment. Actually that is one of their great challenges.

Mobile phones are one of the key actors on the expansion of information. It is predictable that in 2014, the world will have more cell phones accounts than people (Silicon-India-Magazines (2013)). Particularly, smartphones are conquering the phone market, and these devices provide different types of information. For that reason there is an opportunity on the information technology (IT) field to maximize the offer of applications that can benefit individual and communities daily life, namely in people's daily mobility.

The automobile is the most used mode of transport worldwide, with 16,000 billion passenger km (and Transport-DG (2000)). Since the transportation sector is responsible for 22% of the CO_2 emissions from fossil fuel consumption, which in turn occupies 87% of the human sources of CO_2 (whatsyourimpact.org), incentives should be given to people in order to manage their travelling with environmental concerns. An extra motivation on Fuel Consumption (FC) savings is fuel price increase. In Portugal, gasoline average price rose 0.22 Eur/L and diesel's 0.18 Eur/L from January 2008 to July 2014 (maisgasolina.com).

1.1 Motivation

One way to travel by automobile spending less fuel can be made by planning the trip through the most economic path in terms of fuel savings, instead of the typical shortest or fastest path proposed by many GPS systems. Another option may be that of changing the behaviour of drivers avoiding conducts which imply an instantaneous raise in the FC. This motivates the development of software that provides the user information about FC in every moment of his trip as well as the idea of how FC varies with speed, or with change of speed or even with road inclination. Such software must be based on an accurate FC model which the user can rely on.

Smartphones with embedded GPS can be the ideal instruments for travel management with a fuel saving support. However, an additional information system is needed, that sends information about instantaneous FC. That job can be done by On-Board Diagnostics (OBD) devices. Recent vehicles have mandatory specific OBD protocols, namely OBD-II and EOBD, which provide enough information to measure FC second-by-second. The SenseMyCity project [1] from Instituto de Telecomunicações - Porto [2] developed an application that is able to gather instantaneous FC from OBD data and synchronize them with instantaneous speed, acceleration and inclination from GPS data.

There are already some models that predict FC instantaneously, using input variables that can be taken from GPS data. However, as it will be shown in chapter 2, some improvements and new approaches can be done, namely on the type of variables used, on the number of traffic situations and mainly on the regression algorithms used during the machine learning process.

The initial motivation of this thesis came from the opportunity to be able to promote a data collection of typical trips from drivers with smartphone.

1.2 Goals

The main objective of this thesis is to develop an accurate regression model for instantaneous fuel consumption, that should fulfil some requirements:

- use input variables that can be taken from GPS data, namely instantaneous speed, acceleration and inclination;
- take advantage of additional information concerning vehicle specifications;
- be reliable on different situations of driving in a urban scenario;
- be valid for, at least, light-duty vehicles having gasoline or diesel as fuel type.

[1] http://futurecities.up.pt/site/crowdsensor-sensemycity-prototype-and-testbed/
[2] http://www.it.up.pt/

1.3 Outline

This introduction explained the purpose and motivation for the development of this dissertation. Next, chapter 2, explores some of the related work, namely on fuel consumption estimation. A review of several proposed models is made, for instantaneous and aggregated FC estimation. Some examples of their applicability are also referred. This chapter also gives an overview on some algorithms and regression processes used in this thesis. Chapter 3 exposes the methodology followed. This chapter is divided into three sections. The first explains how data was gathered and filtered, and briefly explores the computation of variables. The second section is an exploratory analysis of the data where some statistics are done. The third part explains the experimental procedure. Chapter 4 show all the main results from the three steps of the experimental procedure. Finally, in chapter 5 conclusions are drawn and also some suggestions for further work and applications are made.

Chapter 2

State of the Art

This chapter presents the state of the art on estimating fuel consumption from vehicles. Additionally, it provides some background on the data mining techniques and machine learning algorithms used in this dissertation.

2.1 Predicting fuel consumption in automotive vehicles

This section will give an overview on the work already done on estimating fuel consumption and CO_2 emission from vehicles, using mobility data, possibly collected using a GPS device. The variables were typically speed, acceleration and road inclination. Most of the studies take place in urban environments with light-duty vehicles. However in some cases heavy-duty vehicles are included, too. The motivation is common: the creation of models to be applied in traffic management or route optimization that should be used in an urban planning with environmental concerns.

Some work focus on finding a model that gives the instantaneous fuel consumption and gas emission (fine granularity), with respect to some input variables. Others give results after aggregating points on time intervals or after averaging or summing consecutive points with FC information (coarse granularity). A selection of some important and influential work was done. A list is given in table 2.1.

First, some comments about the type of variables used and about the granularity of the predictions will be made. In the last subsection the types of models used are analysed.

2.1.1 Variables

From table 2.1 it is seen that there are four types of variables. Some are related to the route, considering vehicles movement and road characteristics. Others are variables related to the vehicle. Less common is the use of variables from environmental measures and from the time the collection occurred.

The most used variables are speed (S) and acceleration (A). Speed is easy to measure and acceleration is its derivative, which explains the common use of both. Inclination (I) is also used

Authors	Granularity	Vehicle	Variables
Bowyer et al. (1985)	fine	light	S, A, I
Joumard et al. (1995)	fine	light	S, A
Jimenez-Palacios (1999)	fine	light and heavy	S, A, I, S_w, vehicle characteristics
Ahn et al. (2002)	fine	light	S, A
Cappiello et al. (2002)	fine	light	S, A, weight
Rakha et al. (2004)	fine	light and heavy	S, A
Lei et al. (2010)	fine	light	S, A
Ribeiro et al. (2013)	fine	light	S, A, I
Pelkmans et al. (2004)	coarse	light and heavy	S, I, road class
Ericsson et al. (2006)	coarse	light	vehicle characteristics
Song et al. (2009)	coarse	light	A, S
Tavares et al. (2009)	coarse	heavy	S, I, load
André et al. (2009)	coarse	light and heavy	S, A, vehicle characteristics, road class, hour
Masikos et al. (2014)	coarse	light fully electric	I, weight, road class, battery status, electric auxiliaries status, temperature, humidity, date, hour.

Table 2.1: Fuel consumption estimation models.

in some cases. It can be measured locally or via GPS coordinates. Besides the inclination, some studies make a classification of roads, using each class as a variable. To characterize each road, its type is important (arterial road, motorway, driveway, regional road, street, etc.) and also other factors like traffic intensity, number of traffic lights, etc.

All the models have a validation for a specific type of vehicles. Most of them are developed for light-duty vehicles, although there are also models for heavy-duty vehicles. In terms of energy source, gasoline and diesel are the most common. However Song et al. (2009) also include liquefied petroleum gas (LPG) and Masikos et al. (2014) uses fully electric vehicles (FEV). Some models go further on vehicle characteristics and they consider the weight or the load as variables. Weight is a static variable, while load can change with the time. For example, Tavares et al. (2009) works on solid waste collection, so the load grows during a trip. Other vehicles specification are sometimes considered, usually related to the engine. Jimenez-Palacios (1999) use those parameters directly on FC computation while Ericsson et al. (2006) and André et al. (2009) use them to categorize the vehicles, producing different results for each category. Masikos et al. (2014), use the battery status and electric auxiliaries status (lights, heating, air-conditioning, radio and wipers) as input variables, since their work is with FEV.

From the work presented here, two of them consider variables related to external environment. Jimenez-Palacios (1999) includes the headwind velocity and Masikos et al. (2014) introduces the temperature and the humidity. The capture of these values requires the use of suitable sensors and a more sophisticated apparatus for the data collection.

Finally, in two of the works time is also considered as variable. André et al. (2009) consider the hour slot and Masikos et al. (2014) add the day of the week and the month. The time at which the trip occurs is an important variable, since the traffic varies from hour to hour, from week days

to weekends or even from month to month. Time variables have associated with them the traffic volume.

2.1.2 Granularity of prediction

It is possible to separate the granularity of the prediction into two levels, coarse and fine. Fine-grained models estimate instantaneous fuel consumption, while coarse-grained models estimate aggregated fuel consumption.

Instantaneous FC estimation results from training and testing models on datasets composed by vectors with data collected at each instant of time. The frequency is usually 1 Hz.

Aggregated FC estimation results from models that improve instantaneous FC models or take some information from them to make better predictions on total FC over a specific route or a specific road. Some of these models estimate FC after analysing several trips from different vehicles and drivers. Road characterization is usually included in these models.

2.1.3 Models

In this subsection the models presented in table 2.1 are discussed in some detail. Their goals are also referred.

Bowyer et al. (1985) aims to improve traffic management. They start by proposing a simple method for instantaneous fuel consumption:

$$FC = \begin{cases} 0.444 + 0.090\,R_T\,v + [0.054\,A^2\,S]_{A>0} & R_T > 0 \\ 0.444 & R_T \leq 0 \end{cases} \tag{2.1}$$

where R_T is considered to be the total tractive force required to drive the vehicle and it is computed by the following expression:

$$R_T = 0.333 + 0.00108\,S^2 + 1.200A + 0.118\,G, \tag{2.2}$$

where G is the percent grade (negative for downhill), i.e. $G = \frac{10}{9}I\%$. This expression computes instantaneous FC in mL/s and it was developed to be applied in road intersections or road small sections where inclination and speeds can be accurately measured.

Ten years later Joumard et al. (1995) did an intensive data collection from 150 vehicles, representing the European 1995 fleet, recording instantaneous emissions of CO, HC, NO_x and CO_2. Their proposed model is given in the form of a two-dimensional junction with the variables S and $S*A$.

A very detailed work was also done by Jimenez-Palacios (1999), where a model is not designed to estimate the fuel consumption, but rather to estimate the Specific Power (SP). SP is the instantaneous power per unit mass. They argue that SP is directly related to the engine load. SP depends on vehicle instantaneous speed and acceleration, road inclination and headwind velocity

(S_w):

$$SP = S(1.1A + 9.81I + 0.132) + 0.000302(S + S_w)^2 \tag{2.3}$$

The model proved to be reliable, depending however in many vehicle specification besides SP, in order to predict FC.

Another FC model using as explanatory variables instantaneous S and A is the one developed by Ahn et al. (2002). It is based on a dataset with five light-duty vehicles and three light-duty trucks. The approach was done with numerous polynomial combinations until the 4th degree. To estimate emission rates, they use a target variable called measure of effectiveness (MOE). The final model is a hybrid regression model with log transformation of a third degree polynomial and a stepwise function with two sub-expressions according to the sign of the acceleration:

$$ln(MOE) = \begin{cases} \sum_{i=0}^{3}\sum_{j=0}^{3}(L_{i,j}S^iA^j) & A \geq 0 \\ \sum_{i=0}^{3}\sum_{j=0}^{3}(M_{i,j}S^iA^j) & A < 0 \end{cases} \tag{2.4}$$

where i and j are the powers of S and A, respectively, and $L_{i,j}$ and $M_{i,j}$ are constants that depend on which fuel type is being used. This model is very accurate (coefficient of determination from 0.92 to 0.99) but it is built over data under hot stabilized conditions, not considering vehicle start effects and many other conditions existing in a urban context. Yet, the predictions are done over speed and accelerations levels, i.e. there is a discretization of the explanatory variables.

This last model was called VT-Micro and some the authors improved it (Rakha et al. (2004)), by increasing the dataset in terms of the number of vehicles and applying clustering techniques to make homogeneous categories.

In 2002, a more sophisticated model was also built by Cappiello et al. (2002), calling it EMIT (EMIssions from Traffic). The purpose of this model is to estimate tailpipe emissions, but has a first module where the fuel rate is computed based on instantaneous speed and acceleration for the vehicle category. Variables S and A are the main actors to calculate one of the intermediate variables, the tractive power:

$$P_{tract} = aS + bS^2 + cS^3 + mAS + mg\sin(I)S \tag{2.5}$$

where a is the rolling resistance coefficient, b the speed correction to rolling resistance coefficient, c the air drag resistance coefficient, m the vehicle mass and g the gravitational constant. The road inclination is also present in this expression, however the model was developed assuming its value is 0^o. This is one of its weaknesses. Other weaknesses include the data gathering done under very specific conditions. However EMIT is frequently cited and is a reference in the development of other models.

More recently Lei et al. (2010) take advantage of the historical acceleration. They developed a model calibrated by the multivariate least-squares method for two types of light-duty vehicles.

The proposed expression is similar to the one used in VT-micro, having at each instant t:

$$FC_t = \begin{cases} exp(\sum_{i=0}^{3}\sum_{j=0}^{3}(\lambda_{i,j} S_t^i \overline{A}_t^j)) & \overline{A}_t \geq 0 \\ exp(\sum_{i=0}^{3}\sum_{j=0}^{3}(\gamma_{i,j} S_t^i \overline{A}_t^j)) & \overline{A}_t < 0 \end{cases} \tag{2.6}$$

with $\overline{A}_t$ being composite acceleration given as $\overline{A}_t = \alpha A_t + (1-\alpha)\sum_{i=k}^{9}\frac{A_{t-k}}{9}$; i and j are the powers of S_t and $\overline{A}_t$, respectively, and $\lambda_{i,j}$ and $\gamma_{i,j}$ are the coefficients.

The final instantaneous FC model presented here is the one from Ribeiro (2013). Their work is the base of this dissertation, since it is based on the same dataset, although in this work we have used a more recent version. A multivariate least squares regression was done, using instantaneous speed, acceleration and road inclination, resulting in the two expressions:

$$FC_1 = 5.31 + 3.99A + 0.431I + 0.00213S^2 \tag{2.7}$$

$$FC_2 = 5.55 + 4.08A + 0.0329SI + 5.50E^{-5}S^3 \tag{2.8}$$

As it was already mentioned, coarse-grained models are developed to estimate total fuel consumptions in sets of trips. A program largely used, particularly in simulations, is VeTESS, developed by Pelkmans et al. (2004). It is a tool that saves driver behaviours and speed profiles. Some profiles are saved as references. One profile can be associated to a specific user by comparison, after gathering data from his routes. Based on these profiles, for example, the user can predict the fuel consumption for different possible routes.

To evaluate traffic effects on fuel consumption, Song et al. (2009) developed an aggregate FC model. They relate SP (from Jimenez-Palacios (1999)) with several driving activities from 26 gasoline and liquefied petroleum gas-fuelled light-duty vehicles. The estimation of the aggregated FC comes from a variable called fuel consumption indicator (FCI), taken from the sum of the instantaneous speed and from a derivation of SP:

$$FCI = \frac{\beta \sum_{t=1}^{T} SP' + T}{\gamma \frac{\sum_{t=1}^{T} S_t}{\varepsilon}} \tag{2.9}$$

where T is the total time, β, γ and ε are coefficients and SP' equals SP if SP is positive, otherwise equals 0. The inclination is assumed to be 0.

Another model for aggregated FC is proposed by Tavares et al. (2009) for route optimization with the objective of minimizing FC during solid waste collection. In every street they take into account the average speed and the inclination. The load of the trucks is another variable used. The

FC per unit distance is computed as follows:

$$\overline{FC} = 1068.4\overline{S}^{-0.4905}(1+0.36\frac{Load-50}{100})0.41\exp 0.18I \tag{2.10}$$

There are many other aggregated models. Bowyer et al. (1985) also use their instantaneous FC model to developed models for aggregated FC over specific roads, considering the number of stops and starts. A reference on estimating transport pollutants emissions in urban areas is the European project ARTEMIS, developed by André et al. (2009). Their model uses vehicles speed and accelerations; classifies vehicles considering their characteristics and characterizes roads in various types. One more variable that is used is the hourly traffic volume for each specific road or urban area. Another work on route optimization to reduce FC was done by Ericsson et al. (2006). They created 22 street classes with associated FC for at-peak and off-peak hours, for 3 types of vehicles with 2 fuel types. As last example, more recently Ferreira and d'Orey (2012) used the EMIT model to understand the impact of virtual traffic lights on carbon emissions, concluding from simulations that the reduction can achieve 20% under high-density traffic.

The work done by Masikos et al. (2014) is very reliable. They use artificial neural networks (ANN) for energy consumption estimation. By tuning the number of layers and the number of hidden units per layer, they achieve accurate results taking into account a large range of variables like road class, vehicle weight, battery status, electric auxiliaries status (lights, A/C, etc.), environmental temperature and humidity, and date and hour of the trip. Yet a driver profile was also included as variable, computed from his average energy consumption on previous trips.

In summary, both instantaneous and aggregated approaches have used some models that are very sophisticated, having as input many variables according to vehicles specifications, while others are more simple and accurate under specific conditions. Instantaneous speed and acceleration are two variables that are broadly used. On the other hand, inclination is less often used. Much of the work shown here reveals a great effort in getting variables, namely on the accuracy of them. Still, there is a concern with the type of road on which the data are collected. However, the regression is done by resorting to simple methods, typically multivariate linear least square regression.

2.2 Regression

Prediction is a process used in supervised learning problems, i.e., problems where each observation is a vector with values from input variables and target variables. When the target variable describes discrete categories, the learning process is classification. If the pretended output is continuous, the learning process is called regression.

In this section a general regression model is formalized, the evaluation of a regression output is discussed and some algorithms are presented.

2.2.1 Formalization

A regression model $\hat{Y}$ can be represented by a function f of the independent variables X and of the unknown parameters β:

$$\hat{Y} = f(X, \beta) \tag{2.11}$$

The model $\hat{Y}$ approximates the dependent variable Y knowing the values of the independent ones:

$$E(Y|X) = f(X, \beta) \tag{2.12}$$

Depending on the regression method, f can be expressed by an analytical expression or by a specific structure, for example, a regression tree. One of the tasks during model construction is the determination of the β parameters. One of the essential conditions to find β as the vector of parameters that minimizes the distance between $\hat{Y}$ and Y, with some exception, is that the number of observations N should be larger than the number of parameters k.

2.2.2 Evaluation

The performance of a model should be evaluated in a dataset different from the training set. What is commonly done is a separation of the total dataset into a training set, used to develop the model, and a test set, used to check the accuracy. When the data is large enough (this is very relative, it depends on the algorithm, on the application, on the quality of the data, etc.) the test set can also be divided into growing set and validation set, where the model can be tuned in some iterations before the final model is generated. A typical division is 70% for training and 30% for testing. Other method for training and testing (or growing and validating inside training) is cross-validation. In this method data is first split into k subsets of equal size. In each of k iterations each subset is used as testing set and the remaining as training set.

When the data is a time series, some particular issues should be considered in the evaluation process. Since to predict values in the past based on future data is nonsense, the testing set should include data recorded after the data from the training set. The same should be applied to the growing set and the validation set from the training set. With time series, instead of the cross-validation, there are two similar methods where the validation is always made in future data. One is the sliding window, the other is the growing window. In both there is an initial window with size L and that window is divided in training set, with size L_1, and testing set, with size L_2. In sliding window methods, in each training iteration, that window slides some distance s, where a new training and testing process is made, until the window reaches the end of the dataset. In the growing window method the testing sets are exactly the same, but the training sets are accumulative, i.e., each training set contains all the data previous to the testing set.

Also important to define in a regression process is the evaluation metric. The metric is a function that should be minimized or maximized when the best model is being looked for. One

metric can be the mean absolute error (MAE):

$$MAE = \frac{\sum_{t=1}^{n} |\hat{Y}_t - Y_t|}{n}, \tag{2.13}$$

where Y_t are values from the target variable from the testing set or from the validation set with size n, and $\hat{Y}_t$ are the correspondent predictions. This error measure estimates the typical error. Other possible metric is the root mean squared error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n} (\hat{Y}_t - Y_t)^2}{n}}. \tag{2.14}$$

This measure gives more importance to larger errors because of the quadratic exponent. The squared root implies that the dimension of the error is the same of the dependent variable. A different metric is the coefficient of determination (R^2):

$$R^2 = cor(\hat{Y}, Y)^2 = \frac{(\sum_{t=1}^{n} (\hat{Y}_t - \overline{\hat{Y}})(Y_t - \overline{Y}))^2}{\sum_{t=1}^{n} (\hat{Y}_t - \overline{\hat{Y}})^2 \sum_{t=1}^{n} (Y_t - \overline{Y})^2}, \tag{2.15}$$

where the $\overline{Y}$ and $\overline{\hat{Y}}$ are the average value for the target variable and for the predicted values, respectively. R^2 is the square of the correlation between the target variable and the predicted variable, and its value is between 0 and 1. While MAE and RMSE are metrics that are pretended to be minimized, R^2, since it is a correlation measure, should be maximized.

2.2.3 Algorithms

In this subsection the algorithms used in this dissertation are presented.

2.2.3.1 Linear Regression

One of the simplest models broadly use is the linear regression. This models has the vector form:

$$\hat{Y} = w^T X + w_0. \tag{2.16}$$

w and w_0 values depend on the metric. The most used approach is the least square error estimation. This approach looks for the w parameters that minimizes the expected MSE (Theodoridis and Koutroumbas (2009), pages 103-104):

$$J(w) = E[|Y - X^T w|^2], \tag{2.17}$$

$$\hat{w} = \underset{w}{\operatorname{argmin}} J(w). \tag{2.18}$$

The solution is easily found by annulling the gradient of $J(w)$. For simple problems with an approximated linear solution, or not demanding very accurate results, this simple algorithm is used because of its fast computation and easy understanding. Although it does not produce accurate results for non-linear problems, it is very used for comparison of results.

2.2.3.2 Support Vector Machine

A popular algorithm is the Support Vector Machine (SVM). It is one of the most standard algorithms on machine learning, either in regression or in classification (Bishop (2006), section 7.1 and kn:Theodoridis2009, section 3.7). In SVM for regression, a "tube" is defined mathematically. The points outside that tube are penalized, i.e., they do not contribute so much for the definition of the regression curve (figure 2.1). The regularized error function to be minimized is:

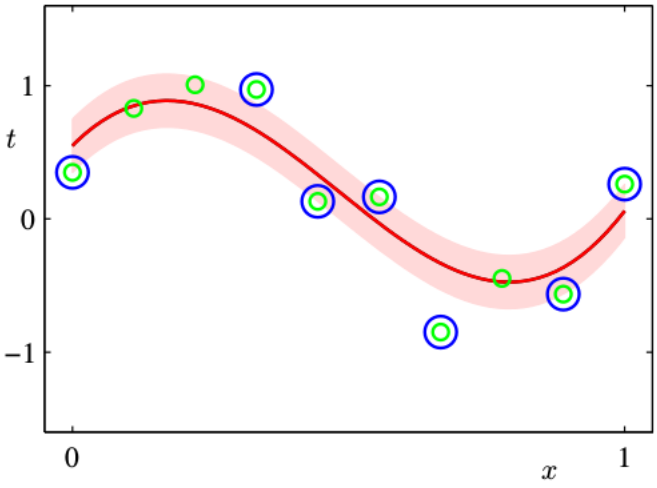


Figure 2.1: Result of a SVM for regression, from Bishop (2006)

$$C\sum_{t=1}^{N} E_\varepsilon(Y_t - \hat{Y}_t) + \frac{1}{2}||w||^2, \tag{2.19}$$

where C is the cost for regularization, and ε is the control factor from the epsilon-insensitive loss function:

$$E_\varepsilon(Y_t - \hat{Y}_t) = \begin{cases} 0, & |Y_t - \hat{Y}_t| < \varepsilon; \\ |Y_t - \hat{Y}_t| - \varepsilon, & otherwise. \end{cases} \tag{2.20}$$

The solution is not shown here. It implies the introduction of slack variables and Lagrange multipliers, followed by the optimization of the Lagrangian (annulling its gradient). The result has the form:

$$\hat{Y}(X) = \sum_{t=1}^{N} (a_t - \hat{a}_t)\, K(X, X_t) + w_0, \tag{2.21}$$

where a_t and $\hat{a}_t$ are Lagrange multipliers constrained to:

$$0 \leq a_t \leq C \tag{2.22}$$

$$0 \leq \hat{a}_t \leq C \tag{2.23}$$

and $K(x,x_t)$ is called the kernel function. In the linear case it has the form:

$$K_{Linear}(x,x_t) = x^T x_t = < x,x_t > . \tag{2.24}$$

But there are other types of kernel functions that map the independent variables to a different space. The choice of the "right" kernel function can greatly improve the fitting. Two very used kernels are the polynomial come:

$$K_{Poly}(x,x_t) = (scale < x,x_t > + offset)^{degree}, \tag{2.25}$$

and the Gaussian kernel, also known as radial basis function:

$$K_{Radial}(x,x_t) = exp(-\sigma||x-x_t||^2), \tag{2.26}$$

where $scale$, $offset$, $degree$ and σ are specific parameters from these kernel functions.

2.2.3.3 Artificial Neural Networks

Another classic algorithm in Machine Learning is artificial neural networks (ANN), used on classification and regression problems (Bishop (2006), chapter 7.1 and kn:Theodoridis2009, chapter 4). A basic ANN is a series of functional transformations. It is a network of conceptual neurons (also called units), organized by layers (figure 2.2). Each neuron of the first layer contains a quantity called activation, that is a linear combination of the input variables. The activation is transformed by a nonlinear activation function. In regression problems, that function is the identity. Each neuron of the second layer does a similar calculation, using as input variables the outputs of the first layer. This happens until the last layer (output layer), where the prediction of the dependent variable is obtained. The links between neurons of two consecutive layers are directed and weighted. Their weights are the coefficients of the linear combinations. Besides weights there is a bias in each neuron. The architecture of an ANN is defined by the number of hidden layers, the number of neurons in each layer and the connections between the different layers of neurons. The learning process for updating the weights of the interconnections also defines the ANN. The weights are updated according to a learning rate (η) and a weight decay (λ), in the direction of gradient steepest descent in $E(w)$, the error function:

$$w_i \leftarrow w_i - \eta \frac{\partial E}{\partial w_i} - \eta \lambda w_i. \tag{2.27}$$

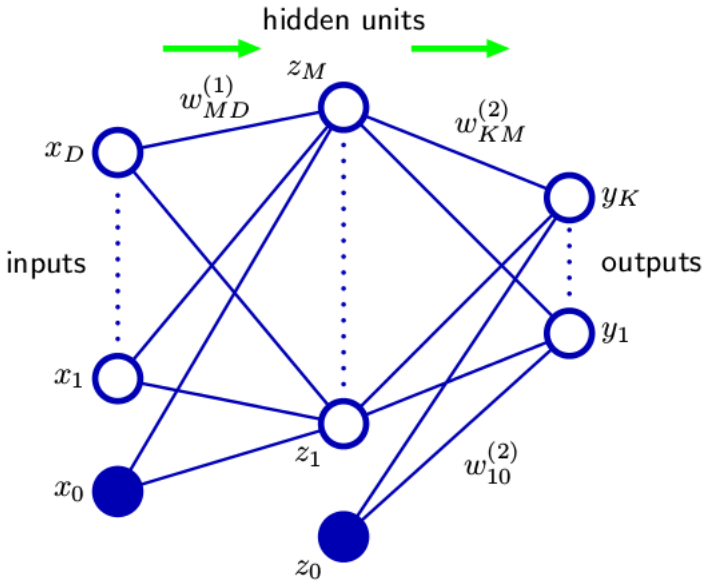


Figure 2.2: Example of an ANN, from Bishop (2006)

ANN are known for their use in a wide variety of problems. However, a good understanding of their behaviour is fundamental to get a robust model.

2.2.3.4 Projection Pursuit Regression

Another algorithm that should be mentioned is the Projection Pursuit Regression (PPR) (Friedman and Stuetzle (1981)). It is a model that consists of linear combinations of non-linear transformations (although they were linear in the initial works) of linear combinations of the explanatory variables. The model is described by the following equation:

$$\hat{Y}_t = w_0 + \sum_{j=1}^{r} f_j(w_j^T X_t), \tag{2.28}$$

where w_j is a vector of unknown parameters associated to the observation X_t and r defines the number of non-parametric functions f_j. This is another method that is flexible to many problems but the choice of the type of functions f_j and the selection of their smoothing parameters and of the number of terms is not trivial.

2.2.3.5 Multivariate Adaptive Regression Splines

One more algorithm that plays with linear combination is the Multivariate Adaptive Regression Splines (MARS) (Friedman (1991)). It consists in a linear combination of basis functions. A basis function can be a constant, a hinge function of a single variable or a product of two or more hinge functions, enabling the interaction between several variables. A hinge function has the form

$max(0, x - const)$ or $max(0, const - x)$. This results in a junction of several hyperplanes segments. The parameters of this model are the number of terms and the degree of the model function. This algorithm is flexible and easy to understand. It is also computationally fast.

2.2.3.6 Classification and Regression Tree

A type of algorithms known as recursive partitioning are also used in many data mining applications. A famous one is the Classification And Regression Tree (CART). A CART is the general term used to decision trees, that can be adaptable to classification or regression problems. However, our focus is on regression trees. A decision tree is a multistage decision system. Starting with the whole training set (root), data is split according to a rule, forming two new nodes. This is done to each node recursively until a stopping criterion is met. Each node has a specific test on the values of a variable and a splitting criterion. This step depends on the specific algorithm, but the decision is made based on the computation of probabilities and information gain. Some CART algorithms do a post-pruning based on results from testing on the validation set. The nodes on the extremes of the tree are called leaves. Each leaf corresponds to a prediction of the dependent variable (figure 2.3), usually the average of all the values that finish there. So, following the

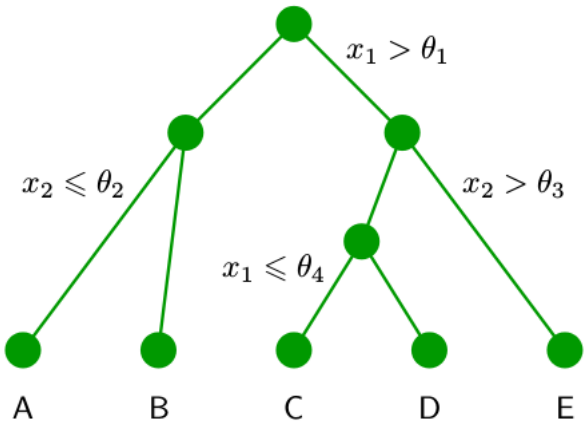


Figure 2.3: Example of a CART, from Bishop (2006)

rules from the root until a leaf, a value for the target variable is estimated, from any vector with the independent variables. Decision trees are easily interpretable and it is also easy to understand the important variables in decision making. Besides that, its learning is fast. However, without a pruning process or other criterion to stop splitting, the tree can overfit the training data. Other difficulty with CART is to find the best tree (between immense possibilities). Another limitation is the fact of only being possible axis-aligned splits.

2.2.3.7 M5

There are many variations of CART. One is the algorithm created by Quinlan (1992). He proposed a construction of piecewise linear models from a tree concept. This method is called M5. Basically the proposed idea was the modification of a classification and regression tree, where the leaves are values, to a similar tree but where the leaves are multivariate linear models. Smoothing and pruning processes are also suggested for this algorithm. M5 showed to improve results of some CART models.

2.2.3.8 Bagged CART

Sometimes it is possible to improve results by combining multiple learning algorithms. This is done by ensemble methods. One famous ensemble method is bagging. Bagging, also called bootstrap aggregating, is a technique where, from a training set of size n, m new training sets are generated by sampling it uniformly and with replacement (Han and Kamber (2006), section 6.14.1 and Witten et al. (2011), chapter 8). This kind of sample is known as a bootstrap sample. From the new m training sets, m models are fitted. A final model is generated by averaging the output of the models.

2.2.3.9 Random Forest

An ensemble model that uses bagging is the Random Forest (RF) (Breiman (2001)). RF also builds multiple decision trees. The difference from bagged CART is on the splitting process. At each candidate split, RF makes its selection from a random subset of features. This reduces correlation between generated trees, something that happens when one or few variables are very strong predictors.

2.2.3.10 Boosted Tree

Another ensemble method is boosting. Boosting uses weak learners to produce a strong learner. At each boosting iteration m, the base-learner outputs a prediction $\hat{Y}_m(X)$. Here we consider as base learner a CART (like Friedman (1999)) and an algorithm called Gradient Boosting. This algorithm requires as input a differentiable loss function $L(Y, f(X))$, where $f(X)$ is the model function, and the number of iterations M. For a tree with J leaves, J disjoint partitions of the input space are created: $R_{1,m}$, ..., $R_{J,m}$. A constant $b_{j,m}$ is predicted for each region:

$$h_m(X) = \sum_{j=1}^{J} b_{j,m} I(X \in R_{j,m}) \tag{2.29}$$

where I is the indicator operator. The model starts to be a constant:

$$f_0(X) = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{n} L(Y_i, \gamma). \tag{2.30}$$

In each iteration m there is an update of the model in the implicit anti-gradient direction, defined by:

$$f_m(X) = f_{m-1}(X) + \gamma_m h_m(x), \gamma_m = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{n} L(Y_i, F_{m-1}(X_i) + \gamma h_m(X_i)). \tag{2.31}$$

The output of the boosted tree is $F_M(x)$. Many other boosting algorithms exist. They proved to have improved the results in many problems and to be reliable in several applications. However they can have difficulties to handle noisy data.

2.2.3.11 Average Artificial Neural Networks

Finally, there are simple ensembles methods that just average the output of multiple models. One example is the Average Artificial Neural Networks. As the name suggests, it builds a model averaging the outputs of several ANN. Some methods do both bagging and averaging.

Chapter 3

Methodology

In this chapter the methodology followed is presented, starting by a description of the data used followed by the exploratory analysis and finishing with the implementation used to get the regression models of light-duty vehicles fuel consumption.

3.1 Data

As it happened in the master thesis of Ribeiro (2013), this work had the collaboration of some volunteers for data gathering. Each volunteer used an Android smartphone, running the application SenseMyCity (figure 3.1), developed at the Instituto de Telecomunicações - Porto, under the European project Future Cities, being carried in the city of Porto [1]. The majority of those trips occurred in the Porto region.

This application was developed to gather mobility data, from GPS and from various types of sensors. Some of those sensors can be external to the phone. The sensor used in this work is an On Board Diagnostics (OBD) scanner. This device is connected to the vehicle and communicates with the phone by a bluetooth connection giving it access to automotive information. Figure 3.2 shows one of those devices. It is a system commonly used to analyse the state of the health of the vehicle, having the capability of identifying malfunctions.

The most recent OBD protocol (OBD2) provides enough information to compute the instantaneous (1 Hz) fuel consumption. Ribeiro et al. (2013) present four ways how the fuel consumption can be computed using this system, depending on how the Corrected Mass Air Flow (CMAF) is computed. All the variables and formulas are explained by the authors. Here the reason of showing how FC is computed is to have a notion of the complexity and of the number of variables and constants:

$$FC = \frac{CMAF \times 100 \times 3600}{CAFR \times FD \times Speed} \tag{3.1}$$

where CAFR is the Corrected Air to Fuel Ratio and FD is the Fuel Density, a constant for each vehicle depending on fuel type. The authors suggest to give preference to the models with fewer

[1] http://futurecities.up.pt/site/

Figure 3.1: SenseMyCity: Android application for gathering GPS and sensors data.

variables, since they tend to have lower error. On gasoline vehicles, if Mass Air Flow (MAF) is provided, CMAF is obtained directly from it. For diesel vehicles a correction is made by multiplying it by a Calculated Load (CL). CL is a derivation of what the authors call Absolute Load (AL). AL is a percentage indicating the current air mass in the cylinders divided by the maximum air mass at standard temperature and atmospheric pressure. CL gives a similar measure, but does not take into account the current temperature, pressure, nor the engine volumetric efficiency. When MAF is not provided, it is necessary to resort to a model that depends on 4 parameters. In addition to CL, the engine Revolutions Per Minute (RPM), the Manifold Absolute Air Pressure (MAP) and the Intake Air Temperature (IAT) are necessary to calculate CMAF. The fourth way to compute the fuel consumption requires the value of AL. Since AL is not available for any of the vehicles monitored it is not presented here. The three equations used to compute the CMAF considered here are:

$$CMAF = \begin{cases} MAF & gasoline \\ MAF \times CL & diesel \\ \alpha \frac{CL \times RPM \times MAP}{IAT + 273.15} & gasoline\,or\,diesel \end{cases} \tag{3.2}$$

where α is a constant obtained from air properties and from some vehicle details (it is explained in detailed Ribeiro et al. (2013)).

From the volunteers that collaborate in this data collection process using the OBD device, 17 had a car transmitting successfully enough information to compute fuel consumption. It was explained to each user the importance of running the data collection process always with the same vehicle. Thus, although some OBD devices do not always transmit information about vehicle model and Vehicle Identification Number (VIN), each user id is always associated to the same vehicle. From the 17 light-duty vehicles, 9 of them use diesel fuel and 8 use gasoline. Besides the

Figure 3.2: Example of an OBD device.

fuel type, the users provided information about the vehicle make and model, engine displacement, engine power and vehicle weight (table 3.1). To better visualize the distribution of vehicles in relation to displacement, weight and power, a 3D scatter was produced (figure 3.3). Since the three variables corresponding to vehicles characteristics have different ranges, a mapping was made, providing a better sense of how close a vehicle is to the others when computing the distances in this new space. The transformation to every variable x is done by replacing x_i with x'_i:

$$x'_i = 1 - \frac{max(x) - x_i}{max(x) - min(x)} \tag{3.3}$$

The scatter on this space is in figure 3.4.

User ID	Model	FuelType	Displacement (cm³)	Weight (kg)	Power (kW)
11	Audi A4	diesel	1896	1450	96
23	Opel Corsa	gasoline	1199	1430	55
25	Volkswagen Polo	gasoline	1198	1550	51
56	Opel Corsa	gasoline	1229	1455	59
57	Nissan Qashqai	diesel	1461	2170	76
59	Mercedes C200	diesel	2143	2175	100
60	BMW S1	diesel	1600	1475	172
76	Volkswagen Passat	diesel	1968	1585	158
78	Dacia Sandero	gasoline	898	1520	66
84	Citroen Xsara Picasso	gasoline	1587	1790	70
107	Mazda 2	gasoline	1349	955	63
108	Renault Clio	gasoline	1200	990	54
109	Audi A4	gasoline	1600	1520	60
202	Peugeot 207	diesel	1560	1500	66
204	Renault Clio	diesel	1461	1515	49
205	Fiat Punto	diesel	1248	1205	66
206	Volkswagen Golf 6	diesel	1598	1314	77

Table 3.1: Vehicles characteristics.

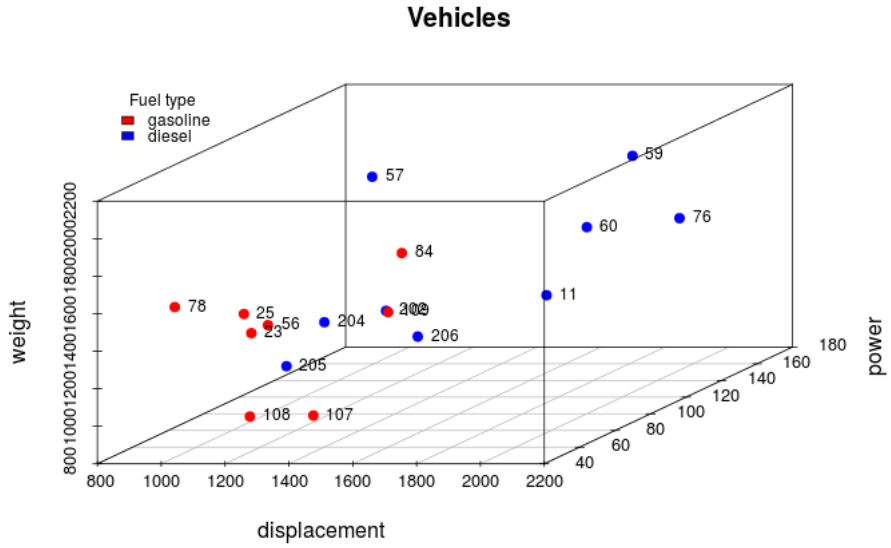


Figure 3.3: Scatter of the vehicles according to their displacement, power and weight.

As it was mentioned, the way CMAF is calculated varies from vehicle to vehicle, depending on the information provided by the OBD. It was sought to have for every vehicle the most accurate value possible. Users 23, 56 and 107 had vehicles providing the MAF, so the first expression of equation 3.2 was used. Diesel vehicles from users 11, 57, 59, 60, 76, 202, 205 and 206 also supply MAF, and the second expression was considered. For the rest of the vehicles (users 25, 78, 84, 108, 109 and 204) CMAF was computed from the third expression due to lack of information for calculating MAF.

Besides collecting fuel consumption data from the OBD device, the mobile application SenseMyCity uses the GPS receiver in-built in the equipment to collect information about position and speed, among others. Joining these two variables to the time it is possible to obtain the instantaneous acceleration and the road inclination. A synchronization is made providing for each second of a trip (when certain conditions are verified) the logging of the fuel consumption, speed, acceleration and inclination. Ribeiro (2013) exposes thoroughly all treatment operations carried out on the data captured by the SenseMyCity app.

The units used in this work were meters per second (m/s) for speed, meters per second squared (m/s^2) for acceleration, degrees (^{o}) for inclination and litres per one hundred kilometres (L/100Km) for fuel consumption. The quantity of points from each user varies. After missing values removal it ranges from 6081 (user 84) to 129310 (user 204). It was established as requirement of a minimum of 5000 instances by user, after a proper filtering. After that filtering, explained next, the number of points per user goes from 5773 (user 84) to 128688 (user 204). Figure 3.5 shows the number of points collected from each user.

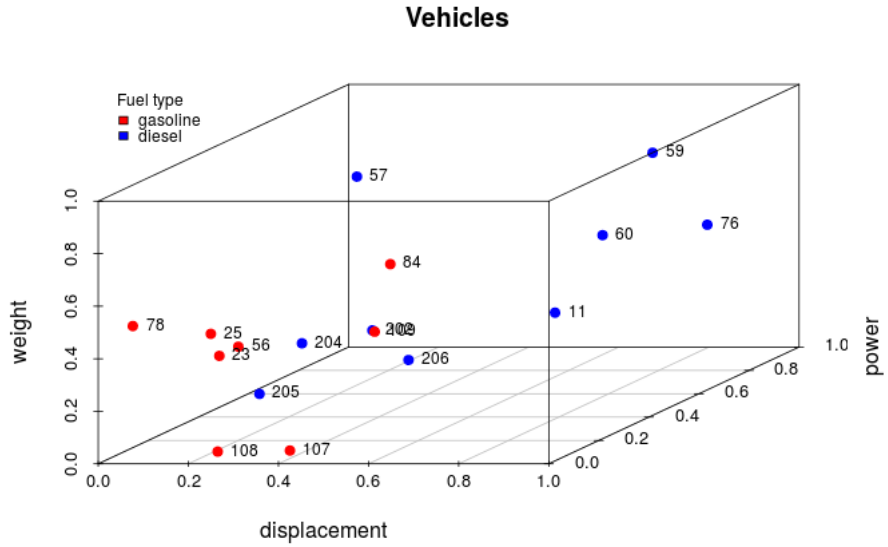


Figure 3.4: Scatter of the vehicles according to their displacement, power and weight after mapping.

The filtering was done to eliminate outliers, missing values, physically unrealistic points, unreliable data and ranges with a small number of points. The removal criteria were:

- points with missing values;
- points with GPS accuracy < 50 m;
- points with inclination out of the interval [-15; 15] ^{o};
- points with acceleration out of the interval [-4; 4] m/s^2;
- points with fuel consumption > 100 L/100Km;
- points where the difference of speeds given by GPS and OBD is greater than 3 m/s.

Elimination was the criterion chosen to deal with missing data, due to its uselessness for the regression methods. Another possibility would be to assign the missing attribute the average of the points with the remaining attributes in a given interval, where the point in consideration was included. Given the large amount of data, the use of this criterion was found not to be rewarding. GPS has an interval censoring when it determines positions. To improve the accuracy of the data, those points with accuracy (according to the phone's GPS) less than 50 m were also removed. The agreement between speed given by GPS and speed given by OBD is also important to ensure accuracy. Finally, the cut in the acceleration, inclination and fuel consumption ranges is due to the small quantity of points outside that range, and also by considering what they represent in the reality, since inclinations above 15^{o} are extremely rare in Portugal and large accelerations and fuel

Figure 3.5: Size of data by user, before and after filtering.

consumptions seem to be rare for the vehicles being used. Rarity makes it hard for regression models to explain them and can actually hurt their ability to learn a useful model.

3.2 Exploratory Data Analysis

An exploratory analysis was done over the four individual variables and on the relationship of each variable with the others. An important task is to analyse the distribution of each variable. To do it, the histogram of each one was built for all the vehicles.

Figure 3.6 shows the histograms of speed for user 11. The histograms of the remaining vehicles are shown in appendix A, in figure A.1. The speed is the most uniformly distributed variable. Still, the most typical values are between 5 and 15 m/s, which are typical speeds in urban roads. Some users have a second peak in the histogram on large speeds (near 30 m/s), probably from collections during highway travels.

Figure 3.7 shows the histogram of acceleration for user 11 and figure A.2 for all the remaining ones. A big percentage of the data have low absolute values of acceleration, typically between -0.5 and 0.5 m/s^2. The number of instances decreases with the growth of the acceleration, in modulus, resembling a Gaussian distribution. This provide further evidence to what it was already said about the reasons to exclude points with large absolute acceleration.

The histogram of inclination for user 11 is presented in figure 3.8, the remaining are in figure A.3. As in the case of the distribution of accelerations, the data has a majority of instances with low absolute inclination values. The bins with larger frequency are the ones in the middle ($-3^o \leq I \leq 3^o$). Again, the decay of the number of points with increasing inclination justifies the exclusion of values with $|I| > 15^o$.

Finally, figures 3.9 and A.4 show the histograms of fuel consumption. Typically the bins with the highest frequency are the first and the second. These correspond to values within the range 0 < FC < 10 L/100km. This fact is not surprising, since the average of fuel consumption of most

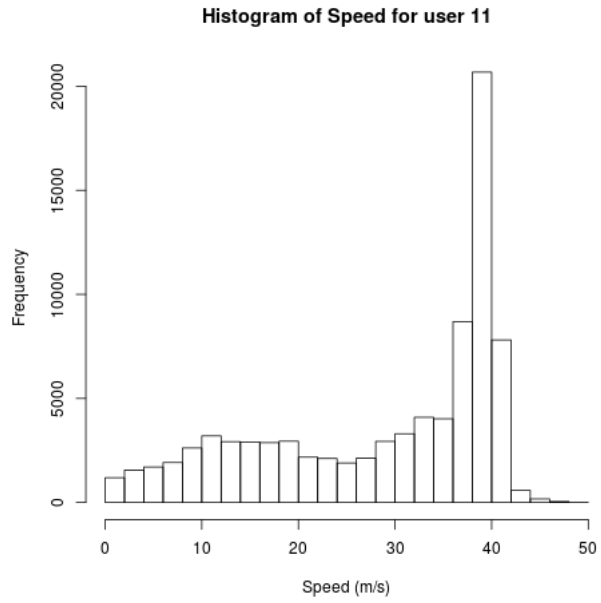


Figure 3.6: Histogram of speed for user 11.

light-duty vehicles is typically between 5 and 10 L/100km. The exceptions are the vehicles from users 78 and 109, which have the highest frequencies in the third and fourth bins, i.e. where $10 < FC < 20$ L/100km.

Besides the distribution of points for each variable, it is important to have a notion about how they are related, in order to understand dependencies and infer some possible causes. The plots that relate any two variables were built and are presented here.

Plots of acceleration vs speed, inclination vs speed and acceleration vs inclination were first generated, in order to infer potential relationships and correlations between the independent variables. In figures 3.10 and A.5 the relation between acceleration and speed is shown. The distribution of points has a shape of a lying menhir, with its bases in the origin. This plot represents well the fact of the data acquisition being made mainly in urban roads, where low accelerations and speeds are more typical.

Figures 3.11 and A.6 presents the distributions of points according to their inclination and speed. The shape of these plots is like a lying tower with a wide basis, meaning a large density on small speeds and absolute values of inclination. This is explained again by the type of roads, as most of them have small inclinations. For those with higher inclination, the speed is low. A city like Porto have many tight and sloping streets, where the speed limit is low and the larger roads are not so sloping.

Finally, the distribution of points according to their values of inclination and acceleration are presented. Figures 3.12 and A.7 show the plots of inclination vs acceleration. There is a high density of points for small absolute values of A and I, indicating uncorrelation between both variables. For large I values, A is typically small anyway.

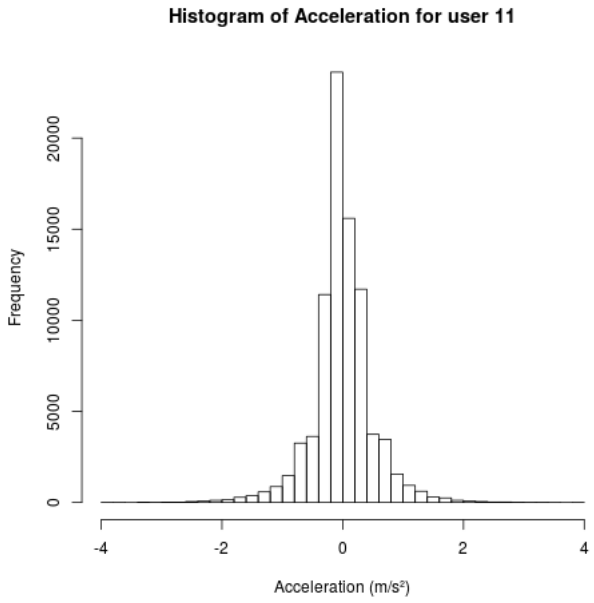


Figure 3.7: Histogram of acceleration for user 11.

The following is the analysis of the relationship between the dependent variable and each independent variable. This discussion starts from the plot of fuel consumption vs speed (figures 3.13 and A.8). There are some high FC values for low speeds, having a progressive decrease, approximately exponential. The density of those high values varies from vehicle to vehicle. Some low speeds can be associated with car boot and high acceleration, where the motor does more effort.

The relation between fuel consumption and acceleration was also studied. It is possible to observe it in figures 3.14 and A.9. There are dense clouds of points where acceleration is between -2 and 2 m/s^2 and fuel consumption is between 0 and 30 L/100km. This cloud is wider for some vehicles and narrower for others. There are some large FC values for small absolute accelerations yet. It should be noted that the cloud is not symmetric. It is skewed to the right, which means larger FC values for positive A. This fact complies with the effort an engine does when it is accelerating: higher effort on positive accelerations than on negative accelerations.

In the plot of fuel consumption vs inclination (figures 3.15 and A.10), there is a dense cloud, in most cases with a triangular shape, for points with inclination between -7^o and 7^o and fuel consumptions from 0 to 30 L/100km. Again, a light right skew is noticed, which represents larger car effort on positive inclinations. The triangular shape is not present in two vehicles (users 78 and 109), having bigger FC values on larger inclinations. In these two cases there is a second dense area for values of FC approximately equal to 50 L/100Km (user 78) or 40 L/100Km (user 109). It is not easy to find a non speculative explanation for these two deviant distributions.

The analysis presented so far provides already a notion of how the data is distributed in the 4-dimensional space. However to have an idea about the evolution of the target variable, fuel

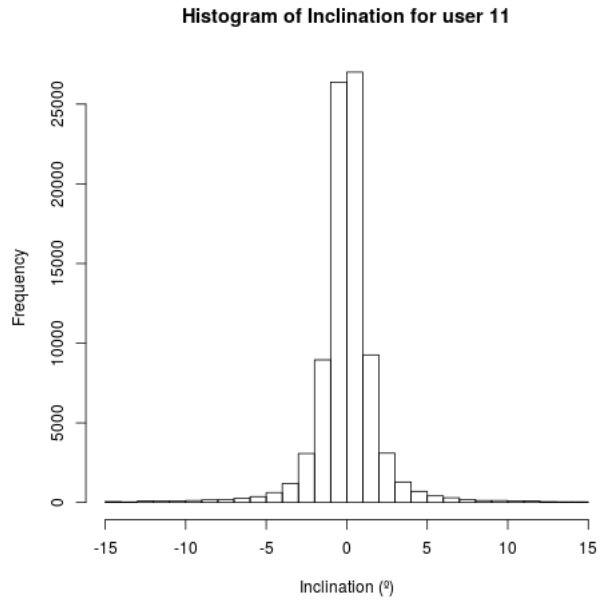


Figure 3.8: Histogram of inclination for user 11.

consumption, with the other ones, curves of the mean fuel consumption were built. First, the means and the standard deviations of consecutive small ranges of speed, acceleration and inclination were computed. Figures 3.16, 3.17 and 3.18 represent box plots of the fuel consumption for user 11. The remaining are presented in figures A.11, A.12 and A.13. The whiskers go up to 3 times the interquartile range from the box and the outliers were omitted from the plot. It is interesting to observe how spread is the data in each range, by the height of the box and the length of the whiskers. The points are more disperse (larger standard deviation) for low speeds and for positive acceleration. The variance is also higher for large absolute inclinations. It is expected to be harder to fit a model in those ranges.

As the final step of the exploratory analysis, curves with the mean fuel consumption of several bins of the independent variables were made. The outliers identified with the boxplot were not included in the calculus of the mean. This was done for all users and curves from vehicles with different fuel types were split into different graphics. Figure 3.19 shows the graphic of the mean fuel consumption vs speed. Looking at both curves, one notes different behaviour from diesel vehicles to gasoline vehicles. Diesel vehicles have lower mean FC on low speeds than gasoline vehicles. They also have a progressive increase of FC above the 20 m/s, while gasoline vehicles seem to stabilize the mean FC above 15 m/s. For diesel vehicles the curve looks like a parabola, while for gasoline vehicles it looks like a negative exponential. Another point to highlight is the fact that while for diesel vehicles the curves are all very close to each other, for gasoline vehicles two of them (users 78 and 109) have a positive shift in the mean FC relatively to the others. From vehicle characteristics we cannot draw any conclusion that could justify this phenomenon. In figure 3.4 we can see that they have no particular resemblance or common characteristic that

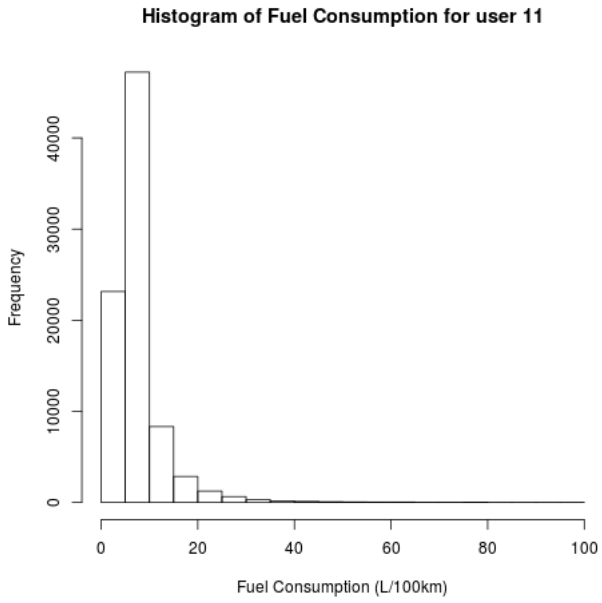


Figure 3.9: Histogram of fuel consumption for user 11.

distinguishes them from other vehicles.

Figure 3.20 presents the curves of the mean fuel consumption for each bin of accelerations. In these graphics there are not many differences between diesel vehicles and gasoline vehicles. Average FC is approximately constant for negative accelerations, values from -3 to -0.5 m/s^2 approximately. Then it seems to have a linear grow until about 1 m/s^2, where it stabilizes again. Vehicles from users 78 and 109 are the exceptions, having a positive vertical shift of more than the double of the others.

Finally, figure 3.21 contains the same curves but for various inclination ranges. The curves for diesel vehicles are similar to the curves for gasoline vehicles, once again. In all the cases there is a critical point. To the left of that point, the variation of FC is negative, while to its right, it is positive. This turning point is between -2.5º and 0º, depending on the vehicle. It can be concluded that increasing inclination will cause an increase of FC, whether the vehicle is climbing or coming down. However, the impact on the FC is higher on positive inclinations (faster increasing). As it was predictable, vehicles from users 78 and 109 have a large vertical shift.

From this exploratory data analysis it is possible to understand that fuel consumption, since it is an effect of a complex process, is not trivially predictable from the three explanatory variables. However, the shape of the distributions and the behaviour of the fuel consumption with the change of speed, acceleration and inclination are similar for different vehicles, with some significant differences on the relation between FC and speed for vehicles with different fuel type. Vehicles from users 78 and 109 have a much higher average fuel consumption, 21.4 and 23.6 L/100Km respectively, caused by unknown factors. These values are far from the average fuel consumption

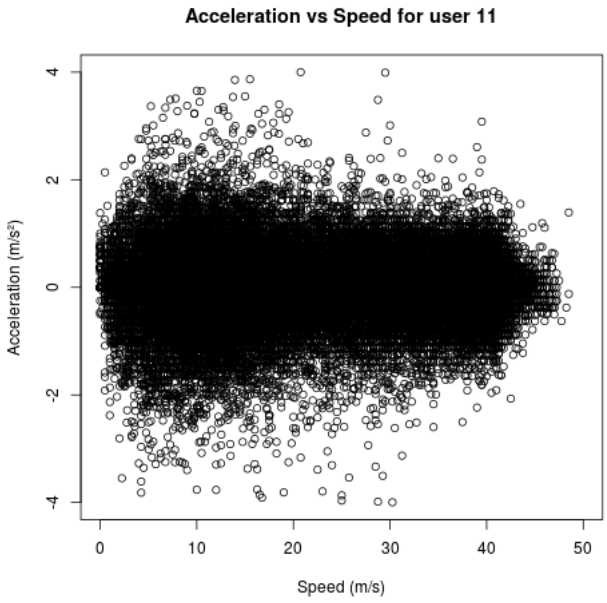


Figure 3.10: Acceleration vs Speed for user 11.

indicated by some trusty sources: 5.0 L/100Km for Dacia Sandero [2] and however 10.2 and 13.0 L/100Km for Audi A4 [3]. Nevertheless these two vehicles were not excluded from the next step (regression), since the shape of their mean FC curves are comparable with the remaining ones.

3.3 Experimental Procedure

In order to achieve the goal of this project, which is to find a model that accurately predicts the instantaneous fuel consumption of light-duty vehicles in a urban scenario, several regression methods were explored. An inductive reasoning was followed, i.e., moving from the specific to the general. More specifically, three hypotheses were investigated:

- **Hypothesis 1:** data from a trip allows the prediction of FC on later moments of that same trip (within trip prediction);
- **Hypothesis 2:** data from trips from a specific vehicle allows the prediction of FC on later trips from the same vehicle (across trips within vehicle prediction);
- **Hypothesis 3:** data from a vehicle allows prediction of FC in a different vehicle (across vehicles prediction);

Table 3.2 outlines the three hypotheses.

The tool used was the R software (R), in particular the caret package (Kuhn).

[2] http://www.dacia.co.uk/
[3] http://www.fuelly.com/

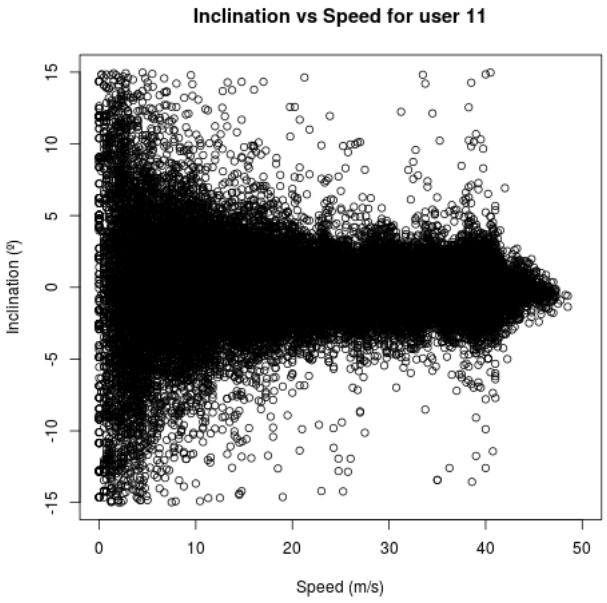


Figure 3.11: Inclination vs Speed for user 11.

Figure 3.22 is a scheme that shows how the data was partitioned for the first hypothesis. For every datasets formed by data from trips (called sessions) from a particular vehicle, several models were produced, each of them corresponding to a session. Only sessions with more than 30 points were included. Since any of these datasets is a time series, for each process the training was done with points that were collected before the points used to test that model. This is an important aspect for the validation of the model, since it does not make sense to use data to build a model to predict events that occurred in the past. The separation was done so that the training set included 70% of the session points and the test set the remaining 30%. To estimate the model performance during the training process, typical methodologies are sliding window or growing window (Hyndman and Athanasopoulos and R-caret). The sliding window method was chosen, using the implementation available in caret R package. The window size was 20% of the training set and validation was performed in a second sliding window of 10% of the same set, containing the points that followed the first window. To find the best parameters for each model, the training was submitted to pass through a tuning process.

Before advancing to the application of regression models, it should be mentioned that it is important to have a very basic model as reference. That model is called the baseline, and one considers that any model with higher error than the baseline is not learning any relevant patterns. The mean of FC of the training set was chosen as baseline:

$$FC_{baseline} = \frac{\sum_{i=1}^{N} FC_i}{N} \tag{3.4}$$

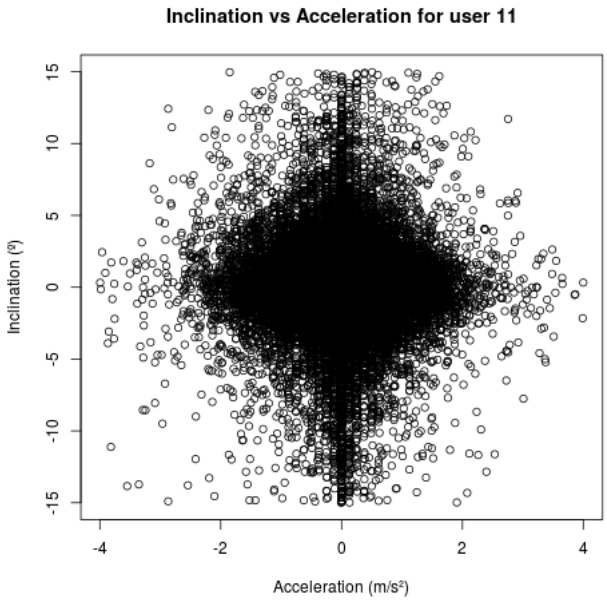


Figure 3.12: Inclination vs Acceleration for user 11.

where N is the number of points in the training set.

The metrics used for evaluation were the root mean squared error (2.14), and the coefficient of determination R^2 (eq. 2.15).

The metric chosen for parameter tuning was R^2, however, since RMSE has the same dimension of the fuel consumption, it will be more referred and represented during the results analysis.

Eleven algorithms were applied, although some of them are similar. Table 3.3 has the name of each method used, the name in the **caret R** package and the parameters for which **caret** allows tuning. In machine learning, it is advisable to start to find a model by simple methods. That allow us to quickly have results, from which we can make some initial considerations. So, the first method applied was a linear regression. This algorithm does not have parameters to tune. It just finds the hyperplane that minimizes the mean squared error. The linear regression model available in **caret** is called **lm**.

The second was SVM. It is known for often improving results when compared to those from linear regression, because it is robust to outliers (the support vectors are the determining elements in fit) and its kernel function can increase flexibility. Two kernel functions were used, representing two different methods: SVM with polynomial kernel (**svmPoly** algorithm in **caret**) and SVM with Radial Basis function kernel (**svmRadial** algorithm in **caret**). Independently from the kernel function, SVM has two parameters: C, the cost for regularization, and ε, the control factor from the epsilon-insensitive loss function (eq. 2.20). C was tuned between the values 0.1, 1 and 10. An higher value of C forces the creation of a model with higher training accuracy, while a model with a smaller value is more tolerant to misclassification. The selected values allow to test the performance of models more or less tolerant to misclassified points. The default value for ε

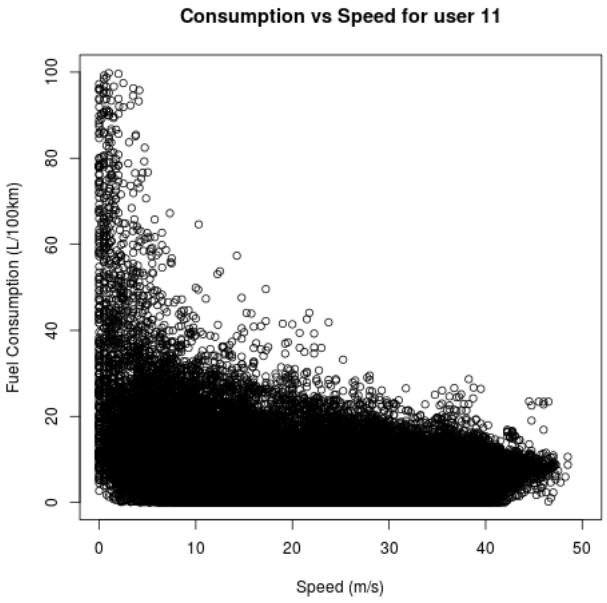


Figure 3.13: Fuel Consumption vs Speed for user 11.

was used, 0.1. Apart from these two generic parameters, other parameters specific to the kernel functions have to be considered. The polynomial kernel has the degree, the scale and the offset (equation 2.25). The radial basis function has the σ parameter (equation 2.26). The offset of the polynomial kernel was set to the default value of 1, the scale was tuned between the values 0.5, 1 and 2, and the degree varied from 1 to 4. With the *svmRadial* method the values of σ for the tuning process were 0.1, 0.5, 1, 2 and 3.

An ANN was applied using the **nnet** implementation from **caret**. It is a feed-forward neural network with a single hidden layer. **nnet** does not do on-line learning and does not have a learning rate. It uses a general quasi-Newton optimization procedure, the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm (Nocedal and Wright (2006), section 6.1). The number of hidden units was tuned from 1 to 6. The weights decay was selected between 0.1, 0.5, 1 and 1.5.

M5, a tree of models, as it is explained in the regression section of chapter 2, was also tested. In **caret**, there are three boolean parameters: *pruned*, *smoothed* and *rules*, as one wishes or not a pruned and/or smoothed tree, and also the use or not of the modified model M5-rules, proposed by Holmes et al. (1999). Parameters were tuned over all possible values.

Another algorithm explored was Projection Pursuit Regression (PPR). In R, the algorithm name is **ppr** and it uses linear functions (original version of the algorithm). It has as parameters the number of terms (*nterms*) which, in this experiment, was varied between 1 and 6.

One more method used was the Multivariate Adaptive Regression Spline (MARS). The parameters are the number of terms (*nprune*) and the product *degree*. During the tuning process, *nprune* was varied between the values 5, 10, 15 and 20, and *degree* went from 1 to 4.

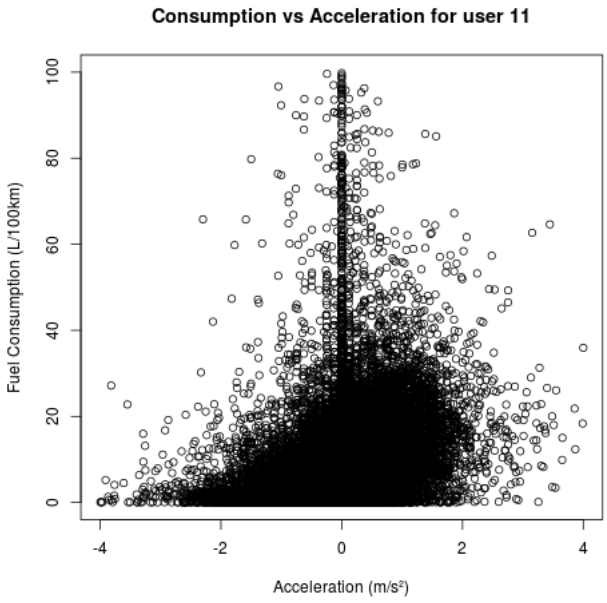


Figure 3.14: Fuel Consumption vs Acceleration for user 11.

In addition to these models, the problem was addressed with some ensemble methods, in order to also evaluate the success of this type of learning. The first one was the bagged Classification And Regression Tree (CART) (Breiman (1996)). The **caret treebag** algorithm was used . This method, where multiple trees are generated by bagging the dataset, has no parameters available for tuning.

A random forest (RF) was also used. The **caret** model used was **rf**, providing the possibility of tuning the number of randomly selected variables at each level (*mtry*). The tuning was done between 1 and 3.

Still in tree-based models, the boosted tree algorithm was used. It is based in gradient boosting where regression trees are utilized as base-learners. An implementation available on **caret** was used, called **blackboost**. The parameters are the number of boosting iterations (*mstop*) and the maximum tree depth (*maxdepth*). *mstop* was tuned between ten values, from 500 to 5000 with a step of 500, and *maxdepth* tuning values were the integers from 1 to 5.

The last ensemble model utilized was the average artificial neural network (AANN), a process that combines multiple ANN models, averaging the outputs of all the networks. The **caret** method used was **avNNet**. It generates multiple ANN similar to the one from **nnet**, using different random number seeds for weights initialization. With this algorithm, it is also possible to do bagging of ANNs. The parameters are the number of hidden units (*size*), the weights *decay* and the option of doing bagging or not (*bag*). The *size* was tuned between 1 and 6, the *decay* between 0.1, 0.5, 1 and 1.5, and *bag* between TRUE or FALSE.

After the parameter tuning and the analysis of the results for the first hypothesis, we moved to the second hypothesis: across trips within vehicle prediction. The plan was to build models for

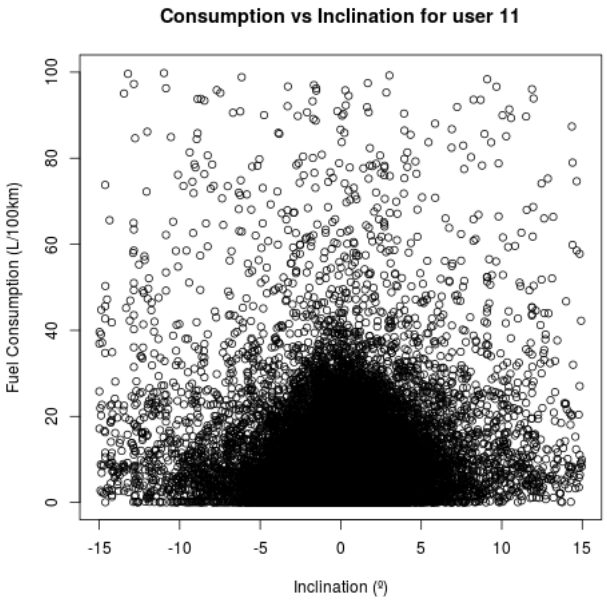


Figure 3.15: Fuel Consumption vs Inclination for user 11.

each vehicle using the algorithms with better results on the first hypothesis. Each vehicle dataset was separated in training and testing subsets. As before, the training set was formed by sessions corresponding to the first 70% of the data. The testing set was constituted by the remaining. In the model performance estimation, during training, the sliding window method was followed, with a sliding window with a size of 20% of the training set and a validation window with a size of 10%.

In the third hypothesis, we investigated the use of models from a vehicle in a different one. This is important for one of the goals of the project, which is to have models to be applied in new vehicles.

In this case, we tried two different scenarios. In the first all cars were combined, independently of the type of fuel. In second, we separated gasoline and diesel cars. Figure 3.23 is a scheme that summarizes all the process.

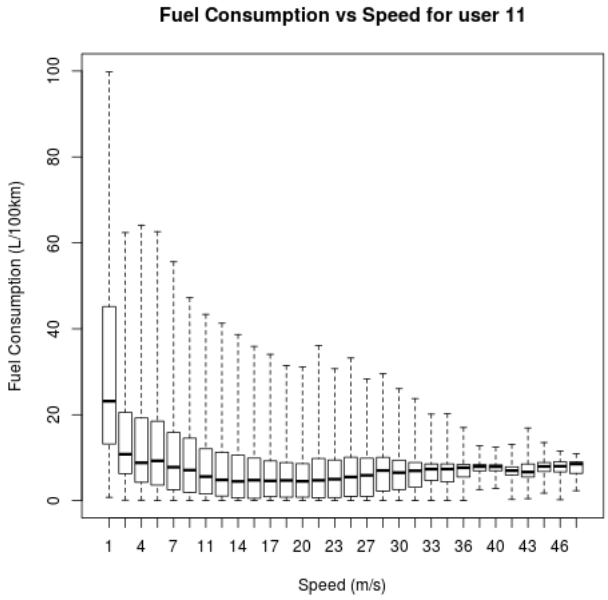


Figure 3.16: Box plot of fuel consumption vs speed for user 11.

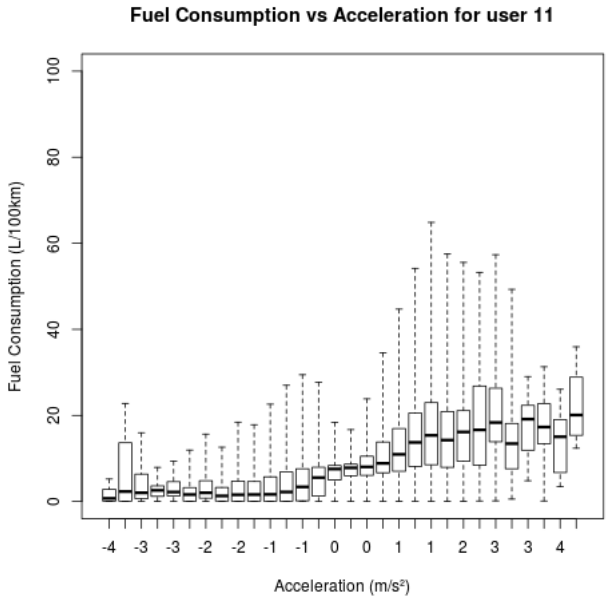


Figure 3.17: Box plot of fuel consumption vs acceleration for user 11.

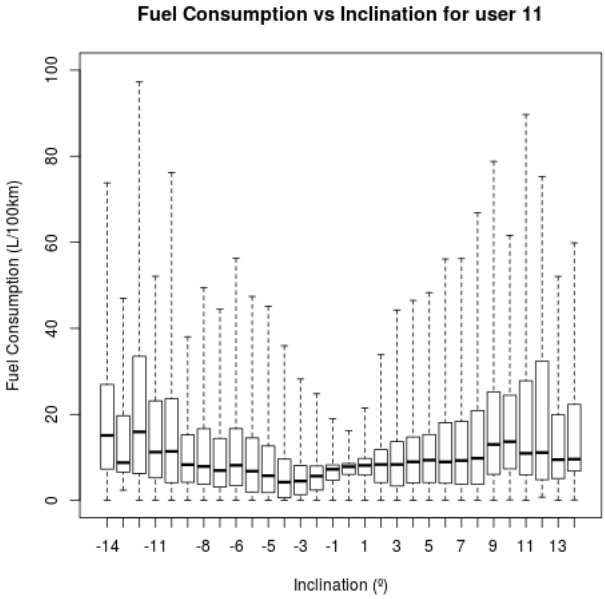


Figure 3.18: Box plot of fuel consumption vs inclination for user 11.

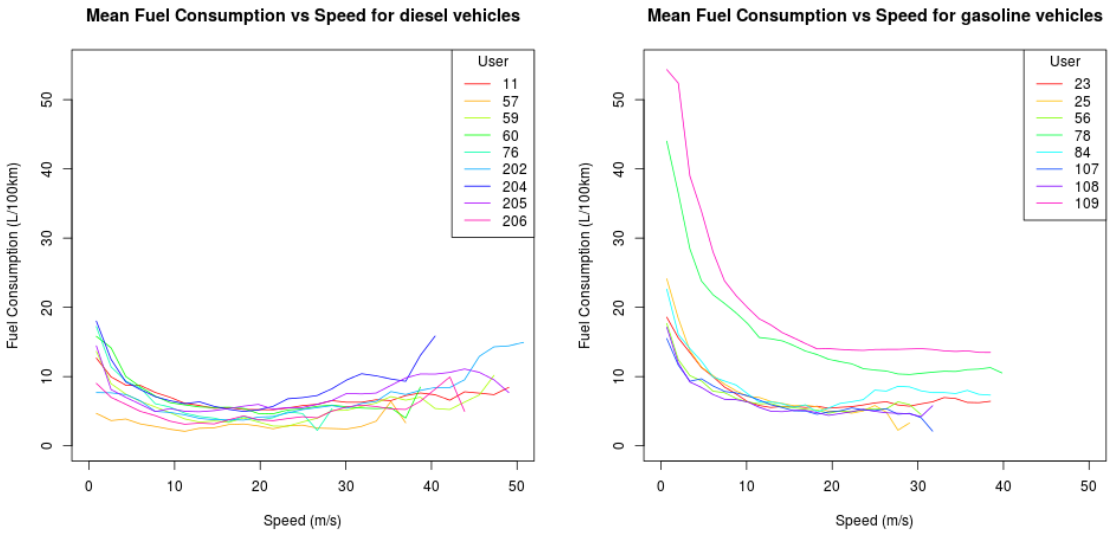


Figure 3.19: Mean fuel consumption vs speed.

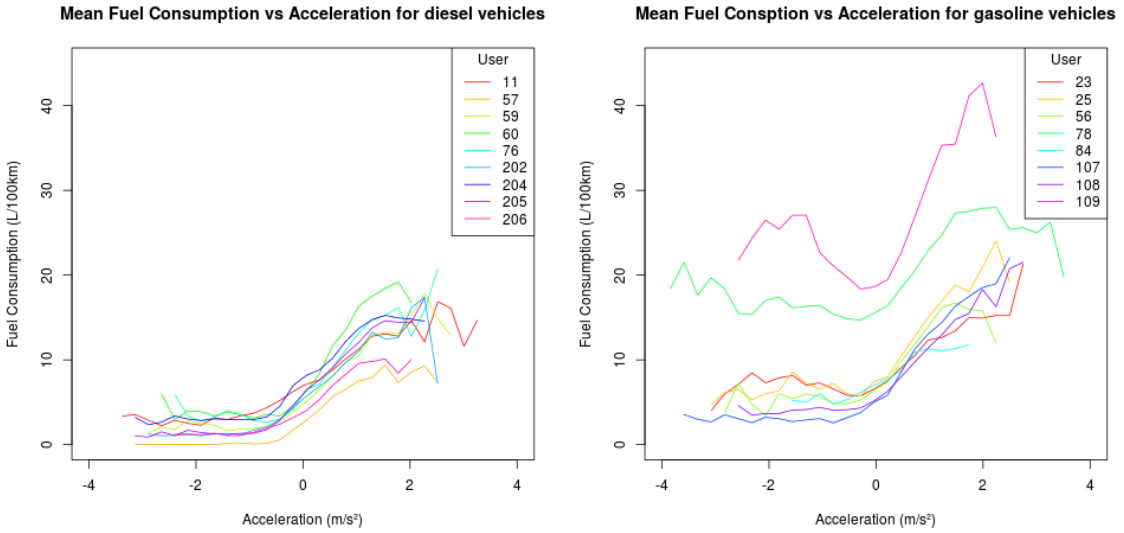


Figure 3.20: Mean fuel consumption vs acceleration.

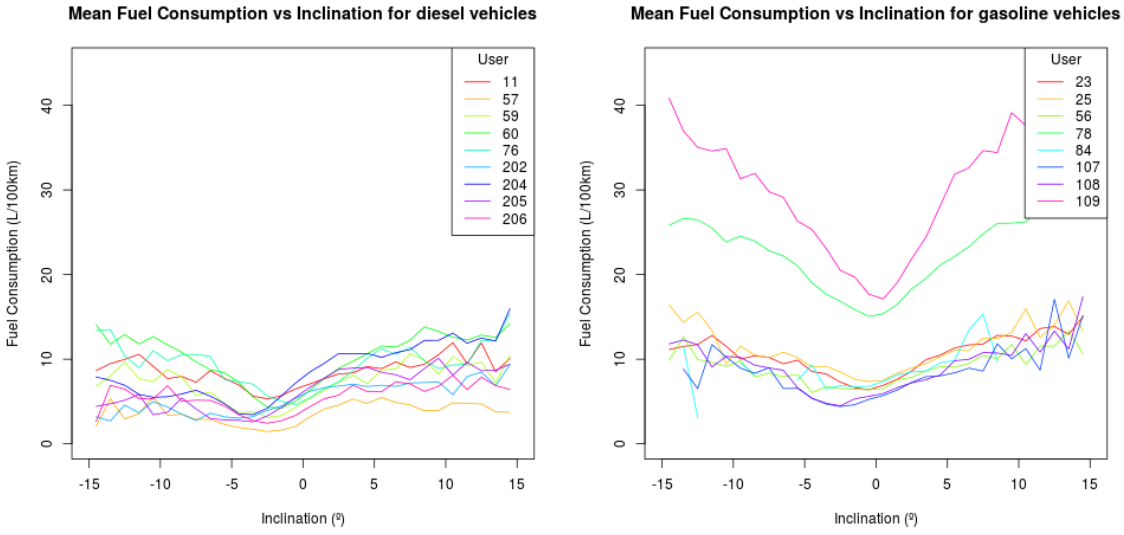


Figure 3.21: Mean fuel consumption vs inclination.

		Trip	
		Within	**Across**
Vehicle	**Within**	Hypothesis 1	Hypothesis 2
	Across		Hypothesis 3

Table 3.2: Hypothesis for prediction.

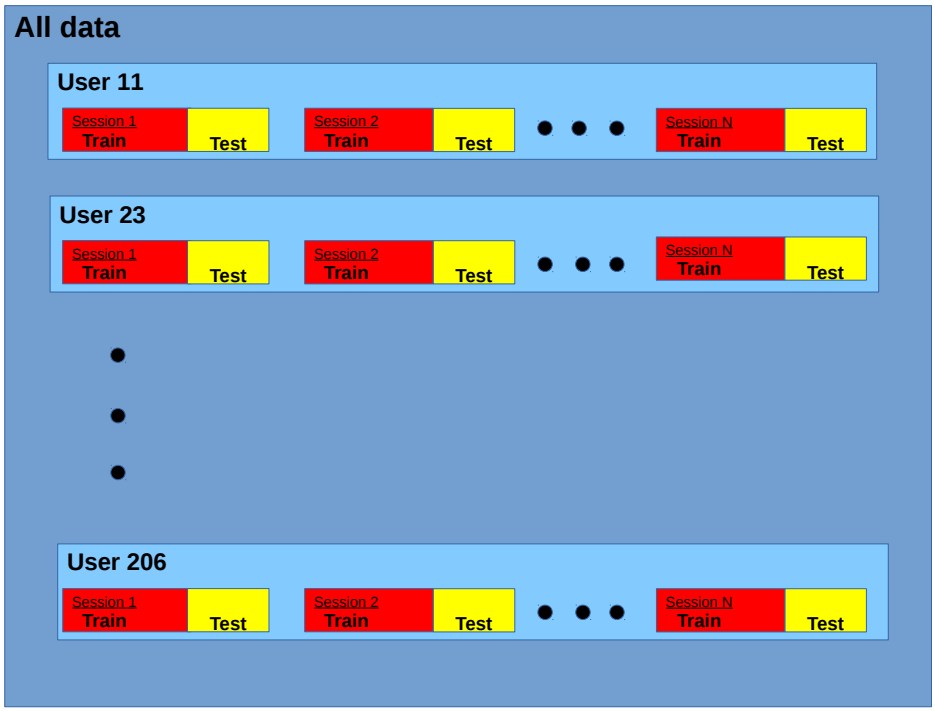


Figure 3.22: Data partition for the first hypothesis.

Model	Name in caret	Tuning Parameters
Linear Regression	**lm**	None
Support Vector Machines with Polynomial Kernel	**svmPoly**	*degree*, *scale*, *C*
Support Vector Machines with Radial Basis Function (RBF) Kernel	**svmRadial**	*sigma*, *C*
Neural Network	**nnet**	*size*, *decay*
Model Tree	**M5**	*pruned*, *smoothed*, *rules*
Projection Pursuit Regression	**ppr**	*nterms*
Multivariate Adaptive Regression Spline	**earth**	*nprune*, *degree*
Bagged CART	**treebag**	None
Random Forest	**rf**	*mtry*
Boosted Tree	**blackboost**	*mstop*, *maxdepth*
Model Averaged Neural Network	**avNNet**	*size*, *decay*, *bag*

Table 3.3: Regression Models.

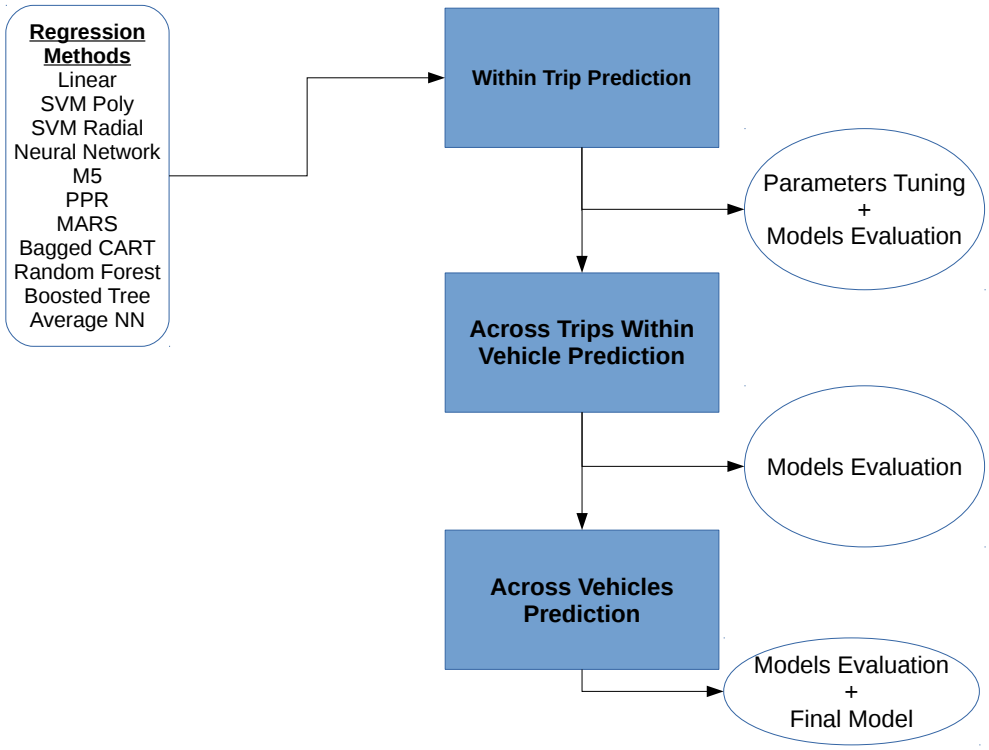


Figure 3.23: Experimental set-up diagram.

Chapter 4

Results

In this chapter the results are presented. As already shown in the previous chapter, the experimental process had three phases. The results of each are displayed in separated sections. It starts with within trip prediction and parameter tuning. It is followed by the construction of models for every car, to investigate across trips within vehicle prediction. After that, models from a vehicle are tested in datasets of different vehicles, referred to as across vehicles prediction. Finally, specific models are developed for each type of fuel.

4.1 Within Trip Prediction

4.1.1 Parameter tuning

As mentioned in the previous chapter, the model generation for each individual trip comprised parameter tuning for the algorithms. In appendix B a bar chart for each parameter is presented. Each bar corresponds to a user in which it is represented the number of sessions where each specific value was selected. Table 4.1 has a summary of the selected values for each parameter.

Figure B.1 represents the results for SVM algorithm with a polynomial kernel function. It is clear that the cost value with better results is 0.1 and for the scale it is 0.5. About the degree, in most sessions the chosen value was 3, however 2 is also well represented and for some vehicles it is preferable, specially for those with a diesel engine.

Parameter selection for SVM with the RBF kernel function is shown in figure B.2. Now, the selected cost value is typically different from the previous ones. The value with better results is mostly 10 but 1 was also chosen many times, being preferable for some diesel vehicles. Since the selected costs are typically higher, **svmRadial** proved to be more intolerant to points with larger error than **svmPoly**. Concerning σ, it is evident that the value with better results is 0.1. Larger σ in RBF kernels lead to a smoother regression, sometimes causing underfitting.

For ANN, the selected parameters are shown in figure B.3. The number of hidden units (*size*) with best results is generally 6, while the most advisable weight decay is 1.5. These results leave some doubt about the certainty of the choice of the parameter. This indicates that it is important to test a larger range of both parameters with more values of decay and size.

The results of parameter tuning for the next model, **M5**, are in figure B.4. In the plots, a value of 1 means "Yes" and 2 means "No". The selection of the parameters leaves no doubt. The model is more successful if is smoothed but without pruning, and rules. The choice of smoothing is not surprising, since smoothness usually avoids overfitting. The pruning should always be balanced between overfitting (no pruning at all) and underfitting (too much pruning), but in this case there was no way to control it in a finer way.

Figure B.5 presents the results for PPR, with only one parameter, the number of terms. There is a large variation in the chosen ones, the most frequent are the values 1, 2 and 6. With these results, it is difficult to infer about the ideal number of terms for our specific problem.

The results of the parameter tuning on MARS are shown in figure B.6. The best degree value is 1 for most sessions. However 2 is selected many times, in particular for user 204. The preferential number of terms (*nprune*) is 5, however with many wins with values 10 and 15, too. It is interesting to see that increasing the degree and the number of terms does not necessarily improve the model performance.

Concerning the ensemble models, the first one tested was the bagged CART (**treebag**). As it was already mentioned, there are not parameters to tune. So, concerning the random forest algorithm, it is seen from figure B.7 that in the majority of the sessions the model works better with only 1 randomly selected predictor. However, that is not a rule for all the vehicles. Indeed for most of gasoline vehicles, the number of selected predictors with more success is 2.

Figure B.8 presents the bar plots with parameter count for **blackboost**, a boosted tree algorithm. The maximum tree depth (*maxdepth*) selected values are divided by all the values but 5 is the typical one. Only in the case of user 59 the winner is 4 and not 5. The number of boosting iterations, or equivalently the number of trees, with more success is 500. Possibly, a much higher number can lead to overfitting.

Finally, figure B.9 shows the results of the parameter tuning for the average artificial neural networks. The most frequently selected values coincide with the ones for the simple ANN. Bagging the model does not necessarily improve the results, given the frequency of the TRUE and FALSE values for *bag* parameter is balanced.

4.1.2 Final Results

Now, it is important to analyse the results in terms of the prediction error in the test set of each session from each vehicle. To do this, box plots of RMSE were built for all the vehicles, where all the models are compared. The limits of RMSE box plot of the baseline model are also present. Figure 4.1 displays those results for vehicle 11. The box plots for the remaining vehicles are in figure **??**.

It is desirable that the RMSE of a model is as much as possible under the baseline RMSE, which we will refer to as baseline error (BE). A model that does real learning should have the median of RMSE below the BE. Not only the median is important, but also box limits and the whiskers length. In other words, they should present results statistically significantly better than the baseline, preferably with a smaller variance. In practical terms, in those plots, the median of

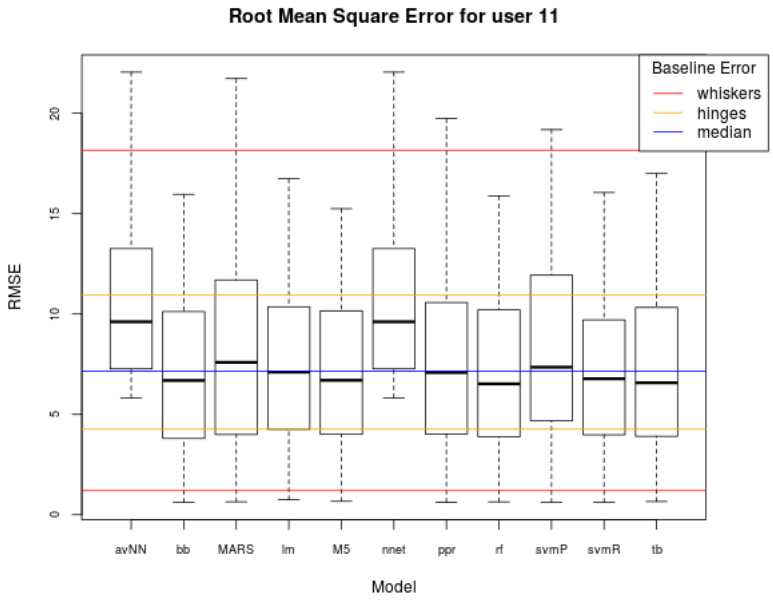


Figure 4.1: Box plot of RMSE of each model for user 11.

the RMSE should be below the blue line, the top hinge of a box (upper quartile) should be below the yellow line, and the top whisker limit (3 times interquartile range out) should be below the red line.

A first conclusion is that neural network models are not learning a useful model, since their values are clearly above the limits for all users. The cause of such bad results is the choice of parameters. These results show that a deeper understanding about the operation of a ANN would be necessary in order to choose some appropriate values for parameter tuning, which is not a trivial task. As expected from the output of the parameter tuning process, it was concluded that the number of units in the hidden layer should be larger for this concrete problem and other values of weight decay should be pre-selected.

Looking at all the box plots, it is noticed that the most successful models are the tree-based ones, namely **treebag**, **blackboost**, RF and **M5**, which obtain the most robust results, in general. Not only their median RMSE are substantially below the median BE, but also their whiskers are not very long, meaning that there are not many sessions with a large RMSE.

On the other hand, some other models, like MARS and SVM, although they present good results for some vehicles, for others their RMSE distribution has the median above the median BE. Furthermore it is noticed the existence of some very high errors for those cases. MARS models for users 76, 107 and 108 show good results, considering its small variance and small median RMSE, but for users 11 and 202 they are worse than the baseline. Identical situations happen with SVM models. By the same evidences, **svmRadial** has positive results for users 202 and 205, but for user 25 there exist very large errors and for user 206 the median is worse than the median of BE. In some cases this is also true for **svmPoly**, for example, for users 11, 57 and 84.

PPR is another model with good results for many vehicles. However, some cases are exceptions, like users 206 and 57, where the median RMSE of the model is larger than the median BE, once more. It is admissible to argue that if PPR and MARS had a larger range and variation of values for tuning parameters, better results could appear. However, parameter tuning requires a high computational effort translated into a very long time, so the number of values for parameter tuning was limited. Finally it is noticed that the improvement of the use of regression models in relation with the baseline varies from vehicle to vehicle and the learning is harder when the mean BE is smaller. For example, for user 204 the mean BE is 5.58 L/100Km and the best result is by RF with mean RMSE equal to 4.22 L/100Km (a difference of 1.36 L/100Km). Other similar cases are users 57, where the best model has a mean RMSE of only 0.41 below the mean BE or user 205 where the improvement in the error is only 0.95 L/100Km. On the other hand, for vehicles with large mean BE, the improvement is much larger. As an example, in user 109, the mean RMSE of **M5** is 9.59 L/100Km while the BE is 17.15 L/100Km (a difference of 7.56 L/100Km) and for user 25 the same model has an improvement of 6.39 L/100Km (from 16.43 to 10.04). Still, even when there is a big improvement in problems with large BE, apparently representing datasets that are harder to fit, the values obtained are still substantially larger than the mean RMSE for more predictable datasets.

4.2 Across Trips Within Vehicle Prediction

Since the second step of the regression process was to build a model by vehicle, the 3 models with best performance from the previous experiments were selected. They were the boosted tree (**blackboost**), the random forest (**rf**) and the model tree (**M5**). The parameters were the ones with higher frequency for each user. Figure 4.2 is a bar plot with the results in terms of RMSE for each vehicle and model.

The performances of the three models are similar. That is, the testing error of each one does not differ much from the others, in general. The results are positive, considering the difference in the RMSE in comparison with the baseline. Again, this decrease is larger when the BE is larger, which provides a big improvement in many cases. Still, for some users with a large BE the progress is not very significant, as in the case of user 76. In 88% of the cases, **blackboost** is the model with the best performance. However, the differences do not seem to be significant.

4.3 Across Vehicles Prediction

To know if it is possible to apply a model from a car on a different one, the models already obtained were tested in all the other vehicles. This time the test was done on the complete dataset for every user. In figure B.11 seventeen graphics present bar plots with the RMSE from all vehicle models tested on a particular vehicle. For comparison, the previous results (test in the same user of training) are also included. In this section it is presented, as illustration, the graphic with the errors of testing on user 11 dataset (4.3).

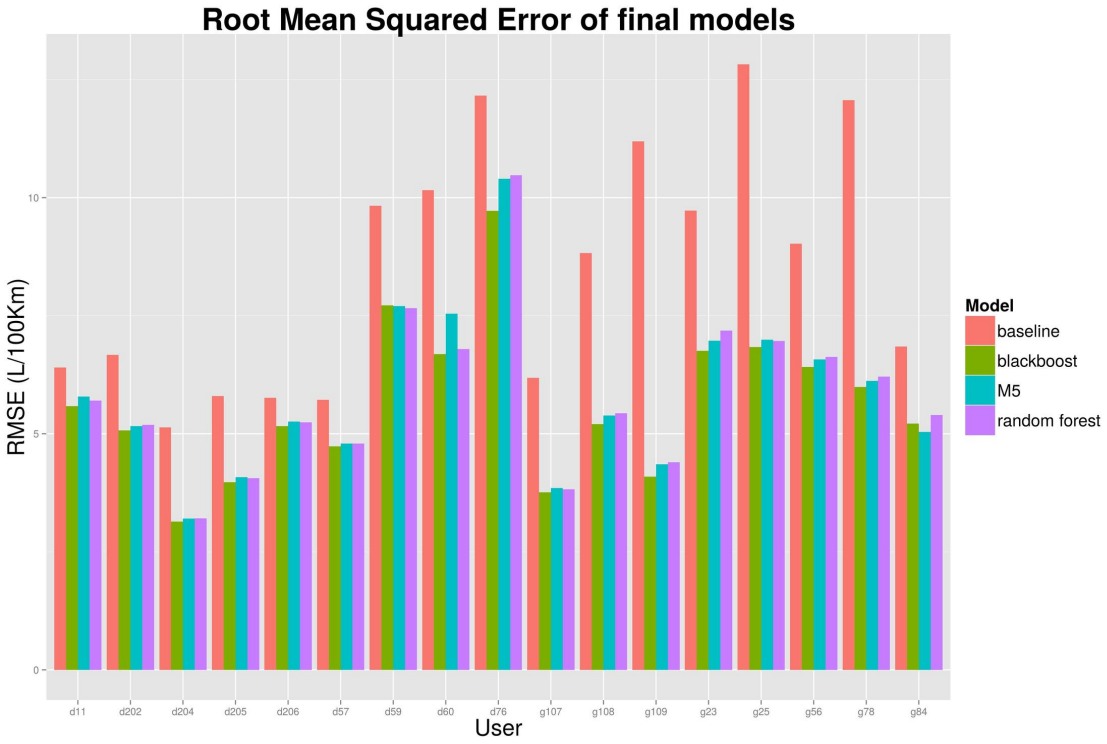


Figure 4.2: RMSE of models from each vehicle.

We can take some notes from these graphics. One is that in some cases, models from other users have better results than the model obtained from training with the dataset from the target car. That happens for users 59, 76 and 84. Analysing the fuel type from the vehicle that provides the model with the lowest error, we observe that, although the fuel type is the same in most cases, that is not true every time. The vehicles from users 11 and 59 (diesel) are better modelled by the models from users 23 and 107 (gasoline), and user 107 (gasoline) is well modelled by user 60 (diesel).

There are two cases with bad results: users 78 and 109. As it was seen, they have atypical curves of the mean fuel consumption, with much higher values of FC. Except for these two cases, the remaining models show flexibility, having relatively close RMSE values when applied to a specific vehicle. The method with better results in the majority of the cases is the **blackboost**. However, again, differences do not seem to be significant.

Looking again at table 3.1 and figure 3.4, where it is possible to measure vehicles similarities, three very similar vehicles stand out: those from users 23, 25 and 56. Their values of engine displacement, power and weight are very close. Also from their graphics in figure B.11 it is seen that one model trained in any of those vehicles predicts particularly well the FC on the others. Apart from these three vehicles, in some other cases there are models that work well on vehicles with similar displacement, power and weight, like the models from users 59, 60 and 76. However, there is not a clear pattern that emerges. As an example, the vehicle from user 11 is well modelled by the vehicles from users 23, 56 and 60, that have similar weights, but it is also well modelled

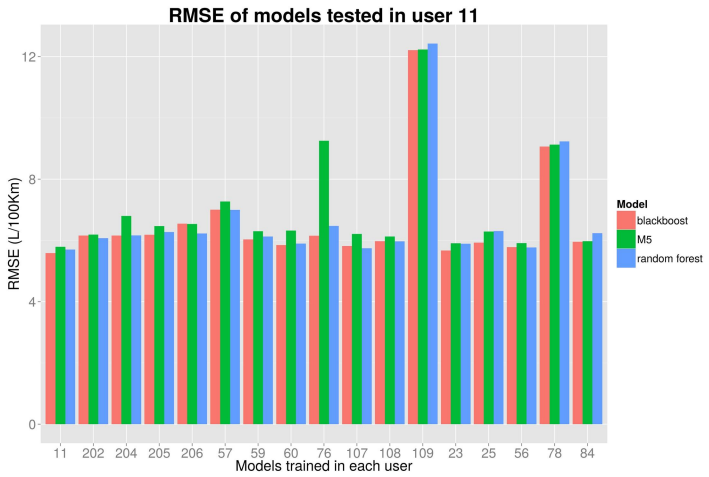


Figure 4.3: RMSE of models from different vehicles tested in user 11.

by user 107 that has a much lighter vehicle. Also, the fact that the model of a vehicle accurately explains the behaviour of another that is identical to it, does not mean that the same is true for the opposite. As an example, the model from user 107 predicts well the FC of user 108 (close weight and power) with a similar RMSE to the one obtained with the model generated from its own data (a difference of 0.38 L/100Km), but the opposite is not true, since the model from user 108 is not one of the best models for user 107 (difference on error of 1.68 L/100Km).

4.3.1 Generic Model by Fuel Type

In section 3.2, it was concluded that the similarity on fuel type leads to similar curves of the mean FC with respect to the explanatory variables. That is reflected in the predictions of models from one vehicle on another with the same fuel type, as it was shown in the previous section. This fact motivates the generation of a more general model, that should be applied to a vehicle with a specific fuel type, in this case diesel or gasoline.

Data from users 78 and 109 was neglected, considering the high deviance of their behavior (large average FC). The new training sets are the merging of the training sets of the previous process, separating gasoline vehicles from diesel vehicles. The same applies to the testing sets. In the training process the sliding window method was used once more, with a sliding window length of 10% the size of the training set and a validation window of 5%. The chosen regression algorithm was **blackboost**, since it proved to be the most accurate when models from a user are applied to others. The parameters used were the same: 500 boosting iterations and maximum tree depth equal to 5.

The RMSE was computed on each testing set. Figure 4.4 presents a bar plot where it is possible to compare them with the baseline and with the model trained in the same vehicle, here referred to as the individual model.

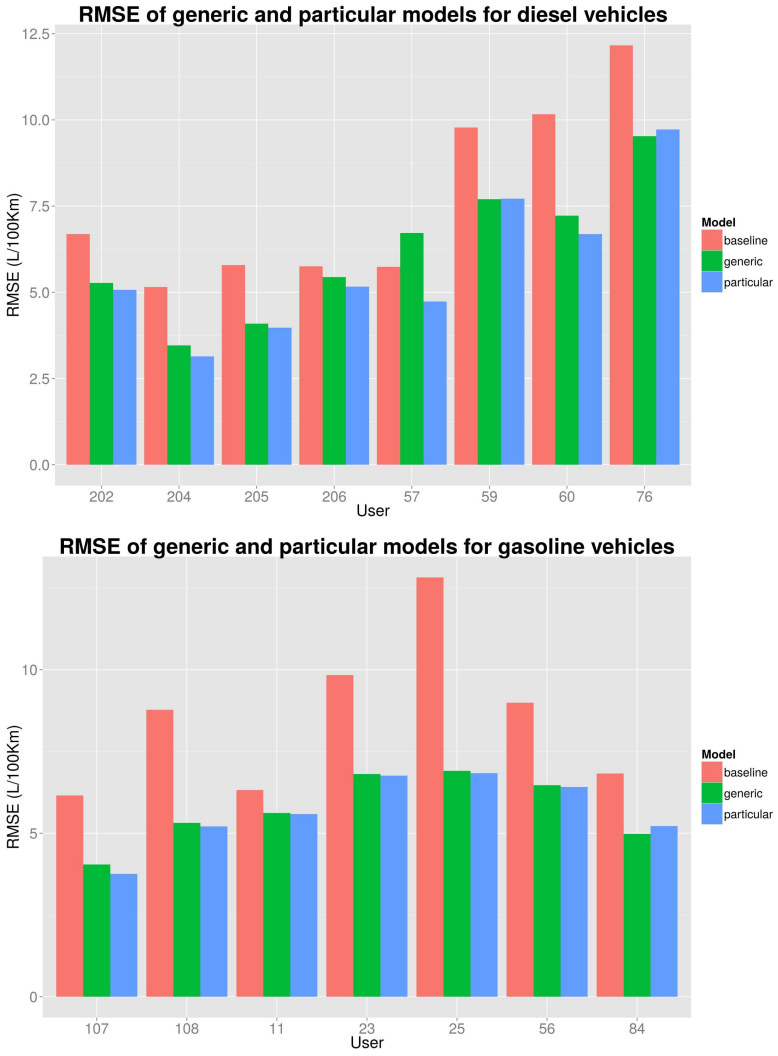


Figure 4.4: RMSE for generic and particular models.

For the majority of the vehicles, the model predicts relatively well, i.e. with an error not far from the one for the individual model. In some cases the generic model has even a lower error than the individual one, as in the cases of users 59, 76 and 84. There is one exception where the model fails in the prediction, having a RMSE higher than BE. It is the case of user 57. Looking at the mean FC curves, in figures 3.19, 3.20 and 3.21, it is evident the lower mean FC for this user. This is the most reasonable explanation for the unsuitability of the general model.

To have a visualization of the predicted mean FC curves in relation with the explanatory variables, the graphics from figures B.12, B.13 and B.14 were drawn, including the curves from the real values. For illustration, the curves for user 11 are presented here: figures 4.5, 4.6 and 4.7.

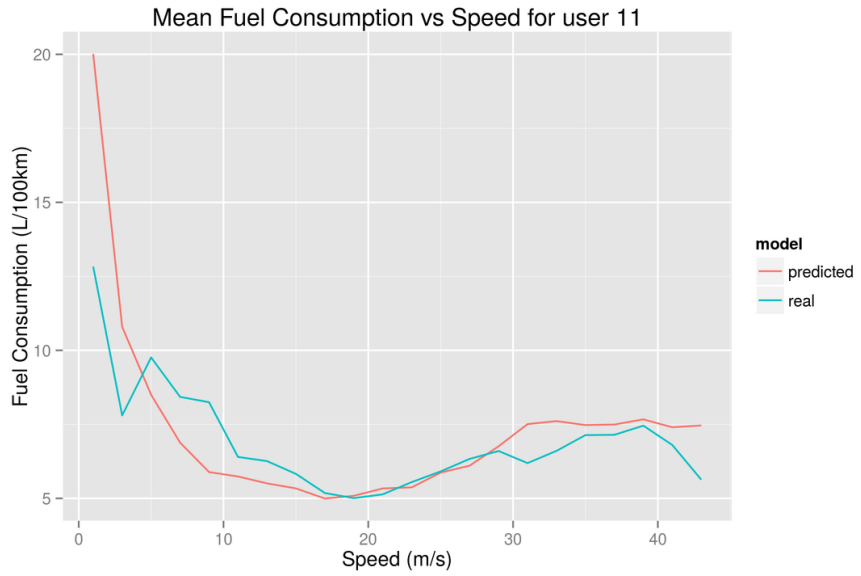


Figure 4.5: Mean fuel consumption vs speed for user 11. Real and predicted curves from the general model.

4.4 Final remarks

Given that the main motivation for this work is to offer drivers with a smartphone the possibility of get information about the instantaneous fuel consumption of their vehicles, the following alternatives are proposed, considering the only mandatory information is the fuel type:

- if information about engine displacement and power and vehicle weight is provided and there is on our database at least one vehicle with the same fuel type, at a distance less than δ in the $d' * p' * w'$ volume, then the **blackboost** model from closest vehicle should be used;
- otherwise, the generic **blackboost** model corresponding to the vehicle fuel type applies.

The volume $d' * p' * w'$ is the one obtained by the transformation in eq. 3.3 on all the vehicle variables. To propose a value for δ with guarantees of success, a study with a larger amount of vehicles is required. Still, from this study one can argue that the three closest vehicles with same fuel consumption have models with capability of predicting the FC of the others (users 23, 25 and 56). The maximum distance between them is 0.105. Other vehicles in similar conditions are 107 and 108, with a distance of 0.143, or 60 and 76, having a distance of 0.329, or yet 202 and 206, with a distance of 0.179. So, the proposed value for δ is 0.35.

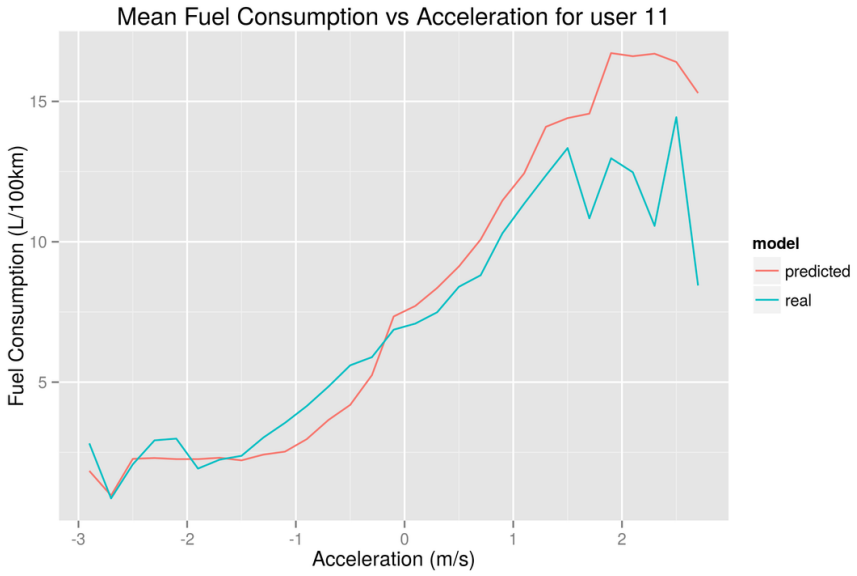


Figure 4.6: Mean fuel consumption vs acceleration for user 11. Real and predicted curves from the general model.

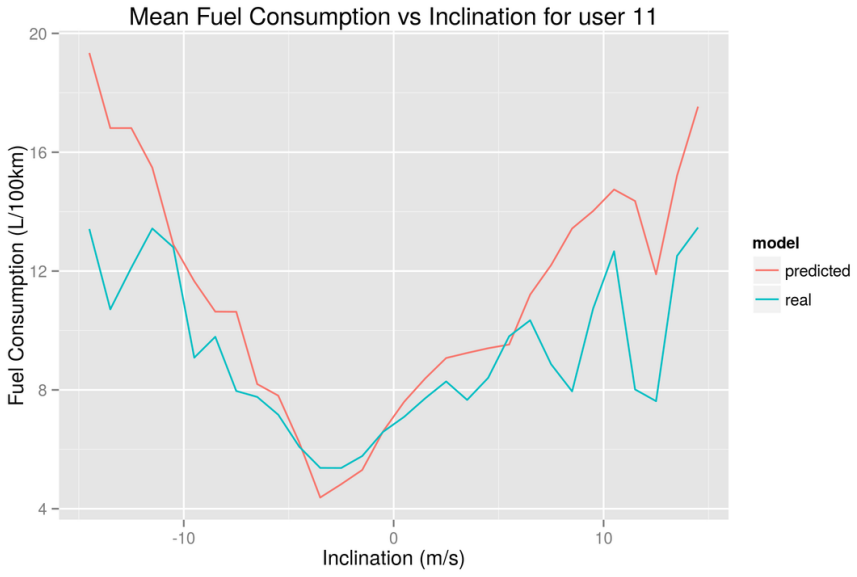


Figure 4.7: Mean fuel consumption vs inclination for user 11. Real and predicted curves from the general model.

Algorithm	Parameter	User																
		11	**23**	**25**	**56**	**57**	**59**	**60**	**76**	**78**	**84**	**107**	**108**	**109**	**202**	**204**	**205**	**206**
SVM with Polynomial Kernel	Cost	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	Degree	2	3	3	3	3	3	3	3	3	2, 3	3	3	3	2	3	2	3
	Scale	0.5	0.5	0.5	0.5	0.5	0.5	1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
SVM with Radial Kernel	Cost	1	10	10	10	1	1	10	1	10	10	10	10	10	1	10	10	1
	Sigma	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Artificial Neural Network	Decay	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	0.1	1.5	1.5	1.5
	Size	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
M5	Pruned	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No
	Smoothed	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	Rules	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No
Projection Pursuit Regression	Nterms	1	2	2	1	1	6	2	2	2	6	6	6	1	1	6	5	1
MARS	Degree	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	1	1
	Nprune	5	5	5	5	15	10	5	5	5	10	5	5	5	5	10	5	15
Random Forest	#RSP	1	2	2	1	1	1	1	1	2	2	1	1	2	1	1	1	1
Boosted Tree	Max Tree Depth	5	5	5	5	5	4	5	5	5	5	5	5	5	5	5	5	5
	#Trees	500	500	500	500	500	500	500	500	500	5000	500	500	500	500	500	500	500
Average ANN	Decay	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5
	Size	6	6	6	6	6	6	6	6	6	6	6	6	6	1	1	6	6
	Bagging	F	F	T	F	F	F	T	T	F	T	F	F	F	T	F	T	F

Table 4.1: Selected values from parameter tuning.

Chapter 5

Conclusions

This thesis started with the motivation of looking for a regression model of instantaneous fuel consumption for light-duty vehicles. This model should be flexible enough so it does not need too many variables, but sufficiently accurate to be useful to its users, having enough information to help manage their trips and driving behaviour, in order to save fuel. This leads to less CO_2 emissions and less travel costs.

Starting from this problem, some steps of a data mining process came up, where there was the possibility of studying the performance of some regression models, adding the understanding of the importance of their parameters. We organized the study into the investigation of 3 different hypotheses.

First, several algorithms were used in the generation of models for each individual trip, where a tuning of the parameters was done. It was concluded that the performance on predicting instantaneous FC is different for the various algorithms. Tree-based models proved to be robust, providing accurate models for several vehicles. Three of them stand out. One that is a tree of multivariate linear models - **M5**, and two ensemble models, one based on bagging - random forest, other based on boosting - **blackboost**.

Those three algorithms were used in the second hypothesis: to build a model for each individual vehicle, based on data from its trips. After building the models for each vehicle using these 3 methods, they were tested in trips from other cars, proving that the application of a model from a vehicle to predict the FC of another one is feasible.

The **blackboost** algorithm, was also used to generate two more general models, one for diesel vehicles, and one for gasoline vehicles. Predictions are considered successful. However the model is only valid for diesel vehicles with mean FC between 5 and 9 L/100Km and gasoline vehicles with mean FC between 7 and 12 L/100Km.

The final model proposes the use of a particular model (from a specific vehicle) or the generic one, depending on the information provided and on how similar the user vehicle is to the vehicles from the database.

The main goal was achieved: single users or corporations can use the proposed model in route management, driving control, studies of the environmental impact or urban planing.

However, improvements can be made to the model, namely if the number of vehicles in the database grows. If this comes about, the global model may be further specified, by taking into account more detailed vehicle characteristics, including models built from some dataset of a specific car or, alternatively, by including those characteristics as explanatory variables in the training process. Another possibility of improvement could be to consider as input variable the mean fuel consumption provided by the vehicle on-board computer.

A better understanding of the data and the behaviour of the algorithms on this problem was also achieved, which is an important goal for such a study.

Appendix A

Additional Graphics and Plots from Exploratory Data Analysis

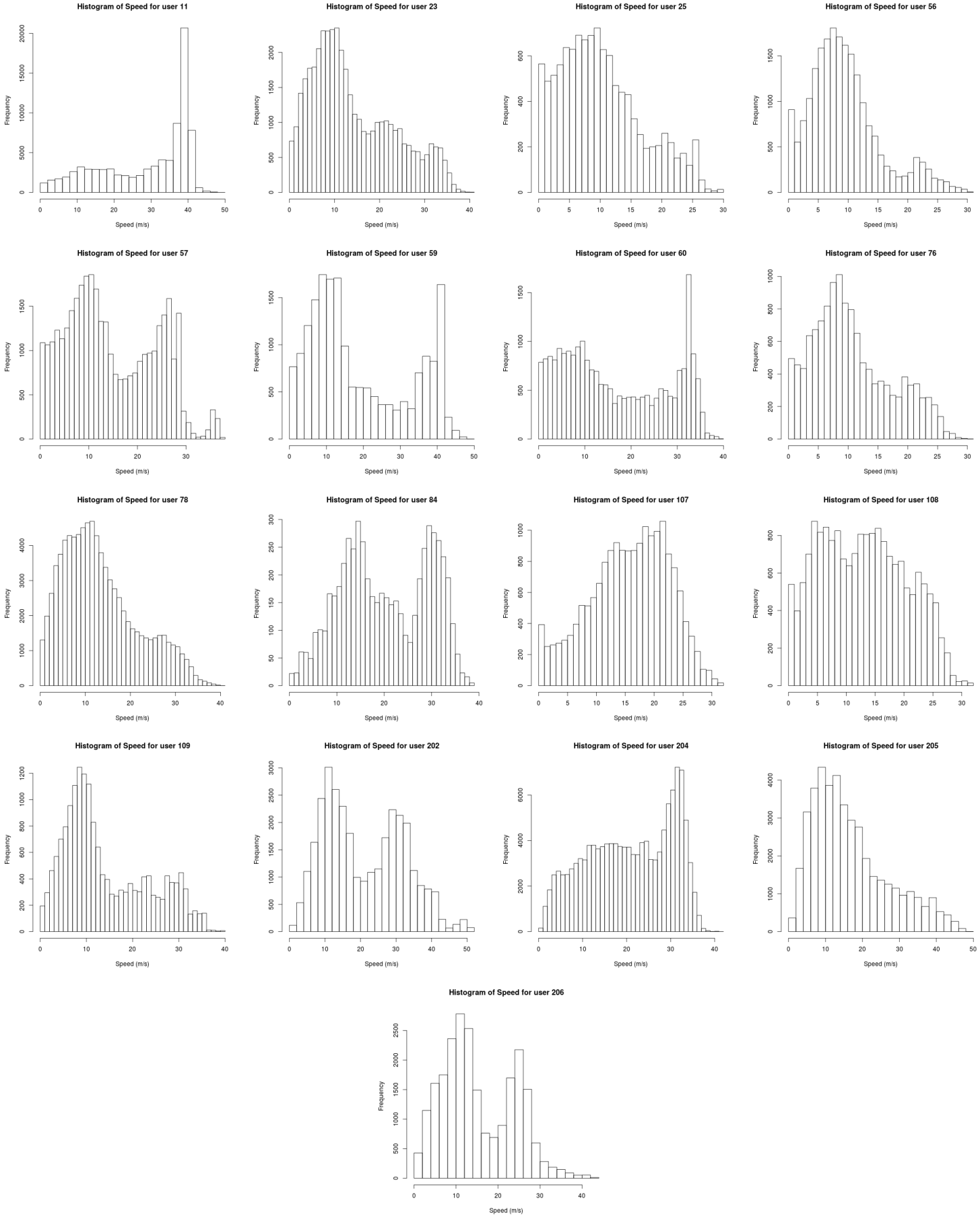


Figure A.1: Histograms of speed.

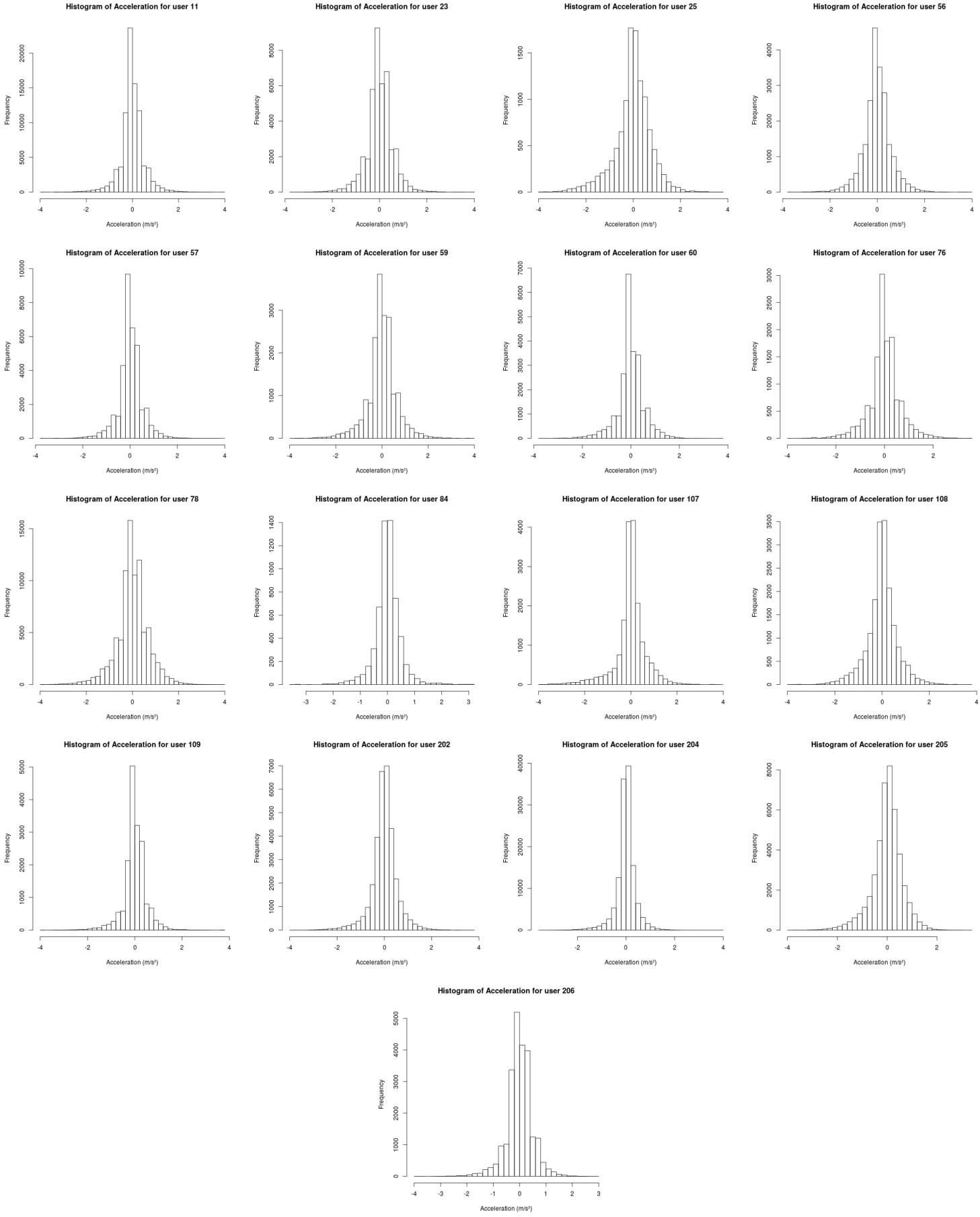


Figure A.2: Histograms of acceleration.

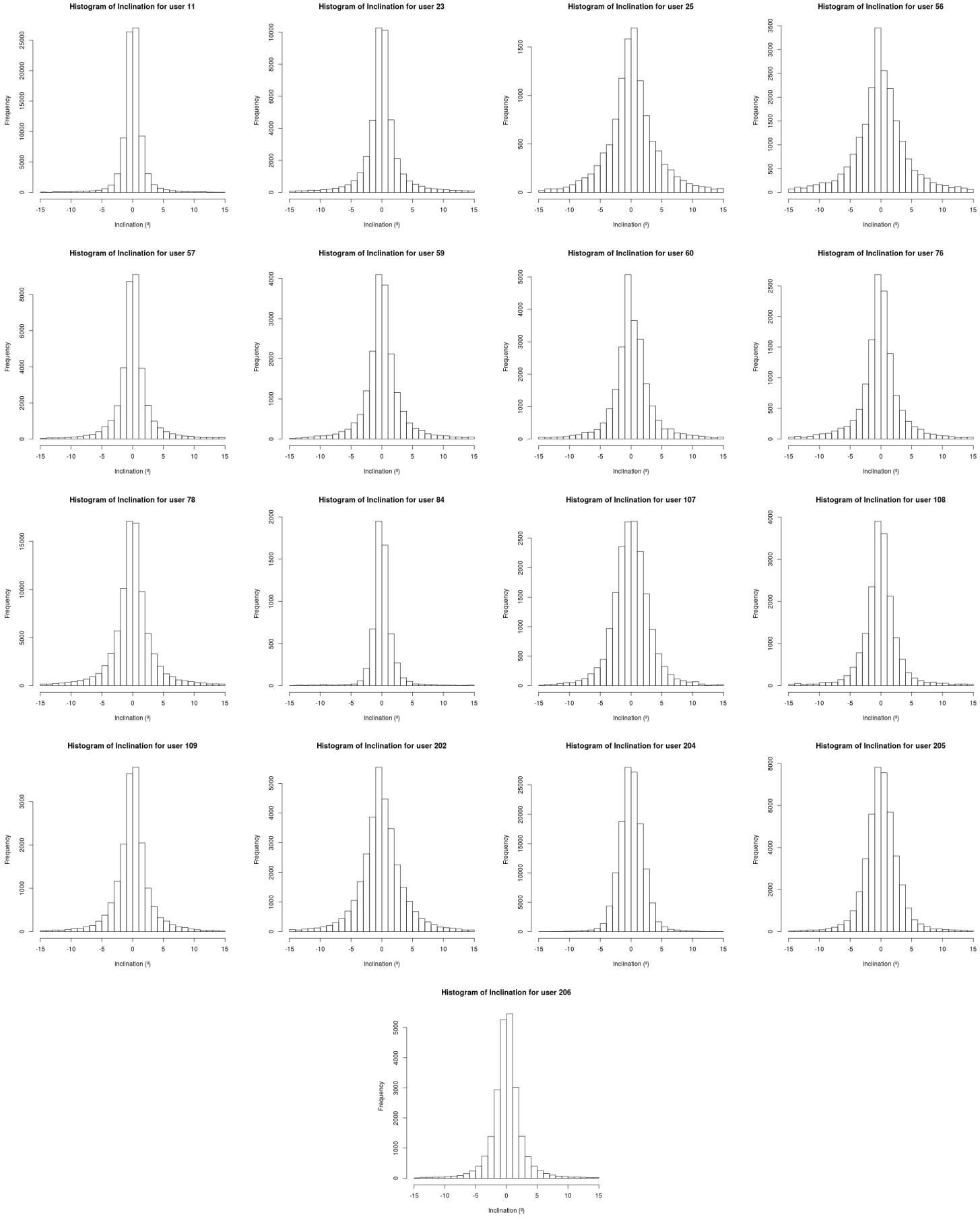


Figure A.3: Histograms of inclination.

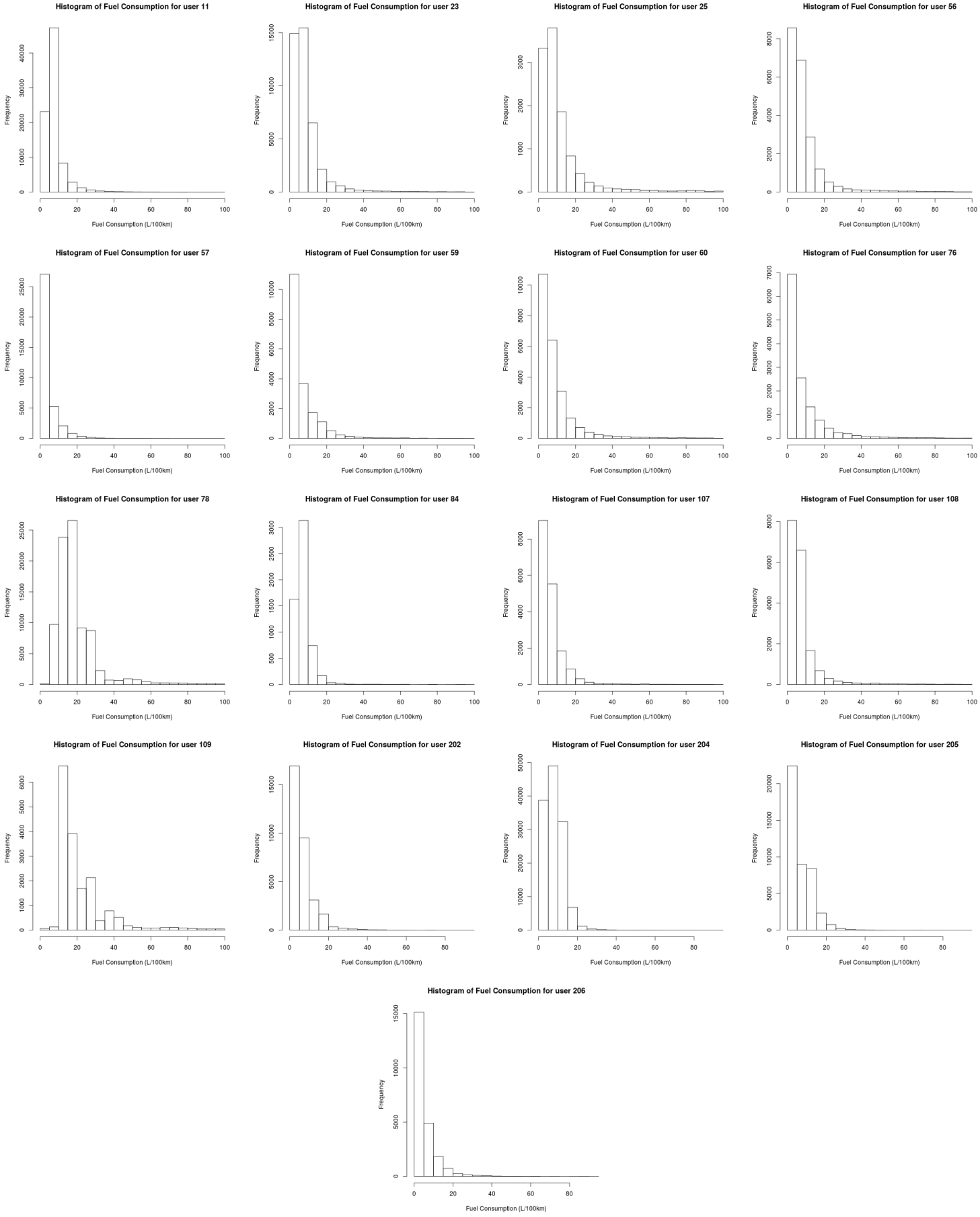


Figure A.4: Histograms of fuel consumption.

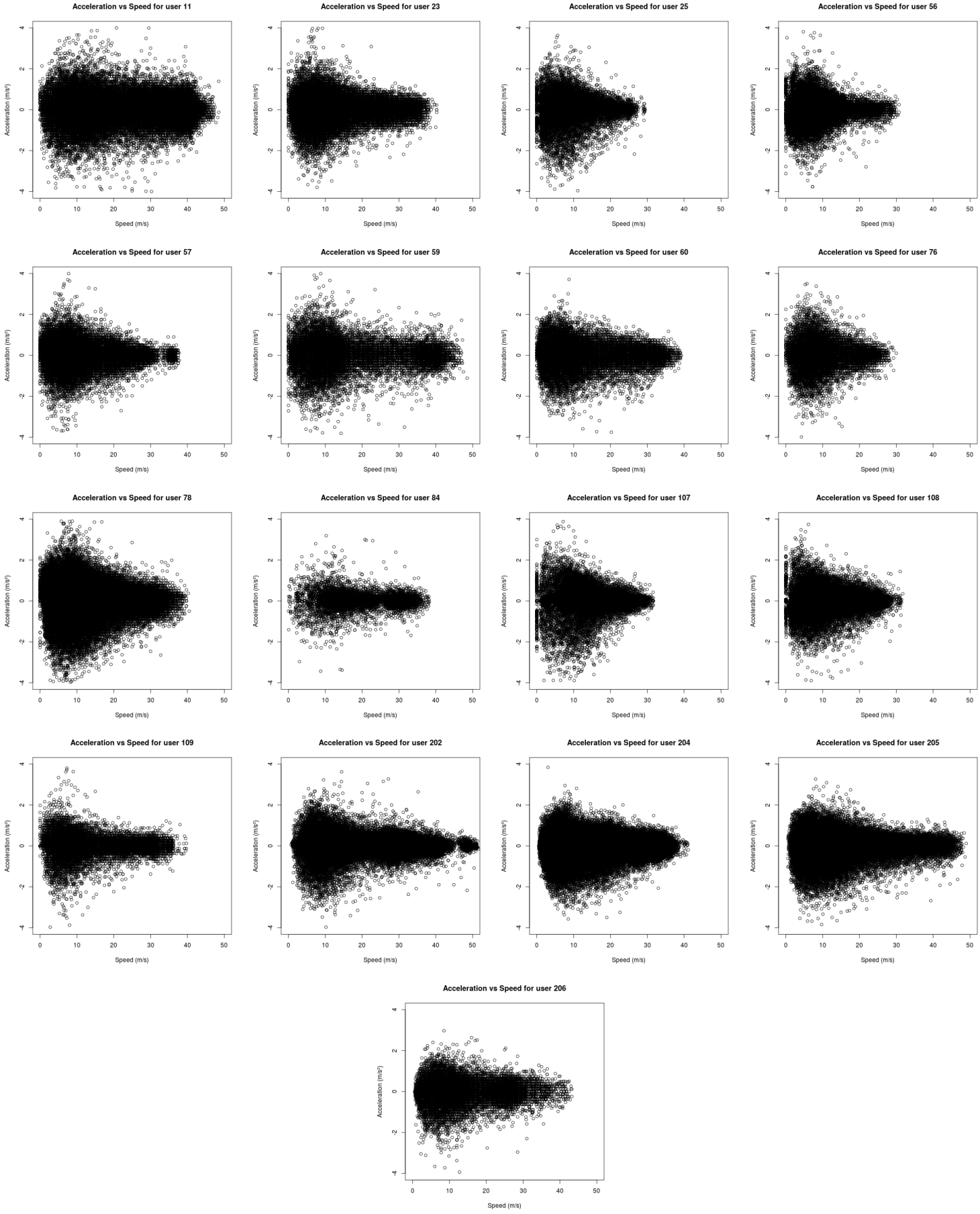


Figure A.5: Acceleration vs speed.

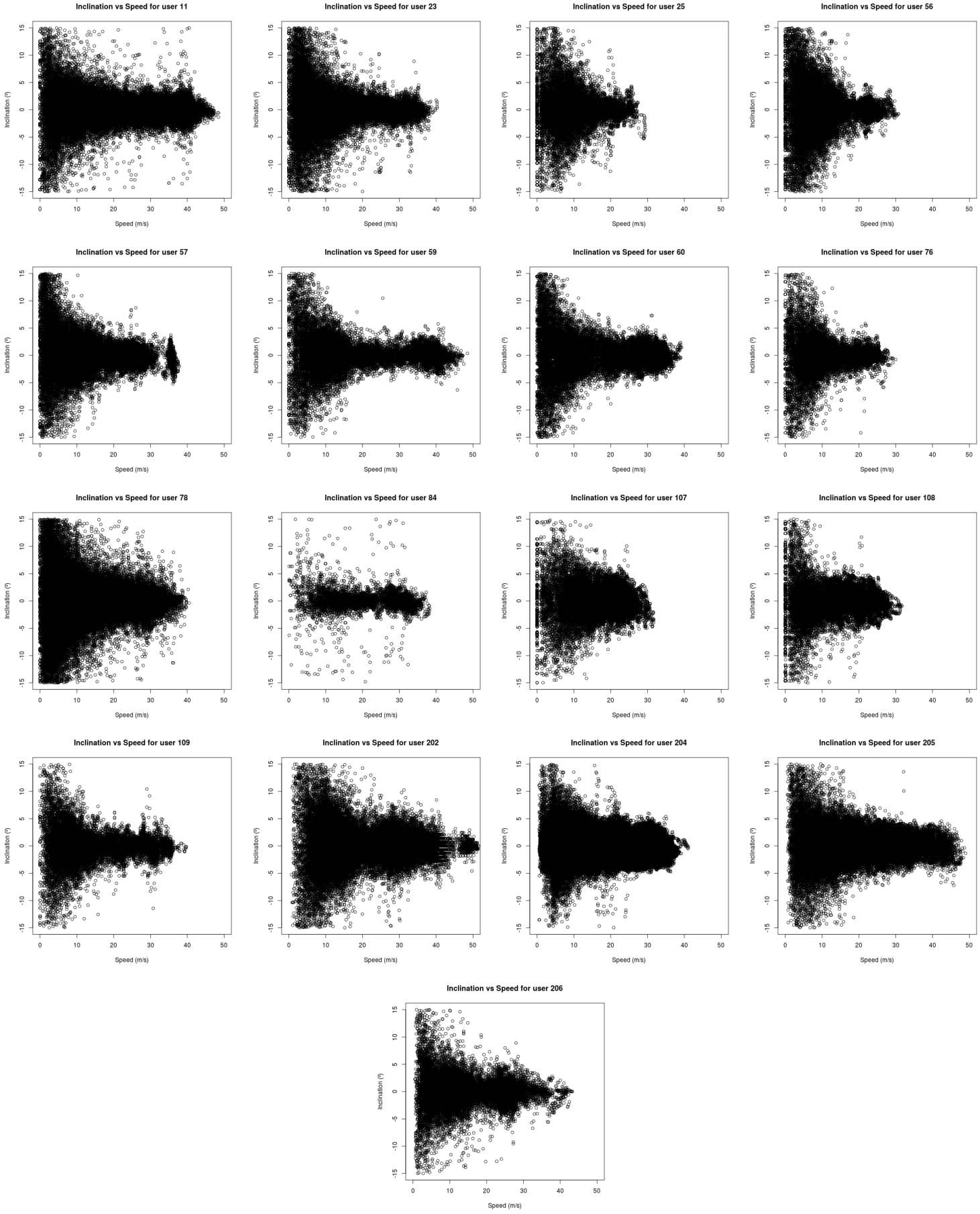


Figure A.6: Inclination vs speed.

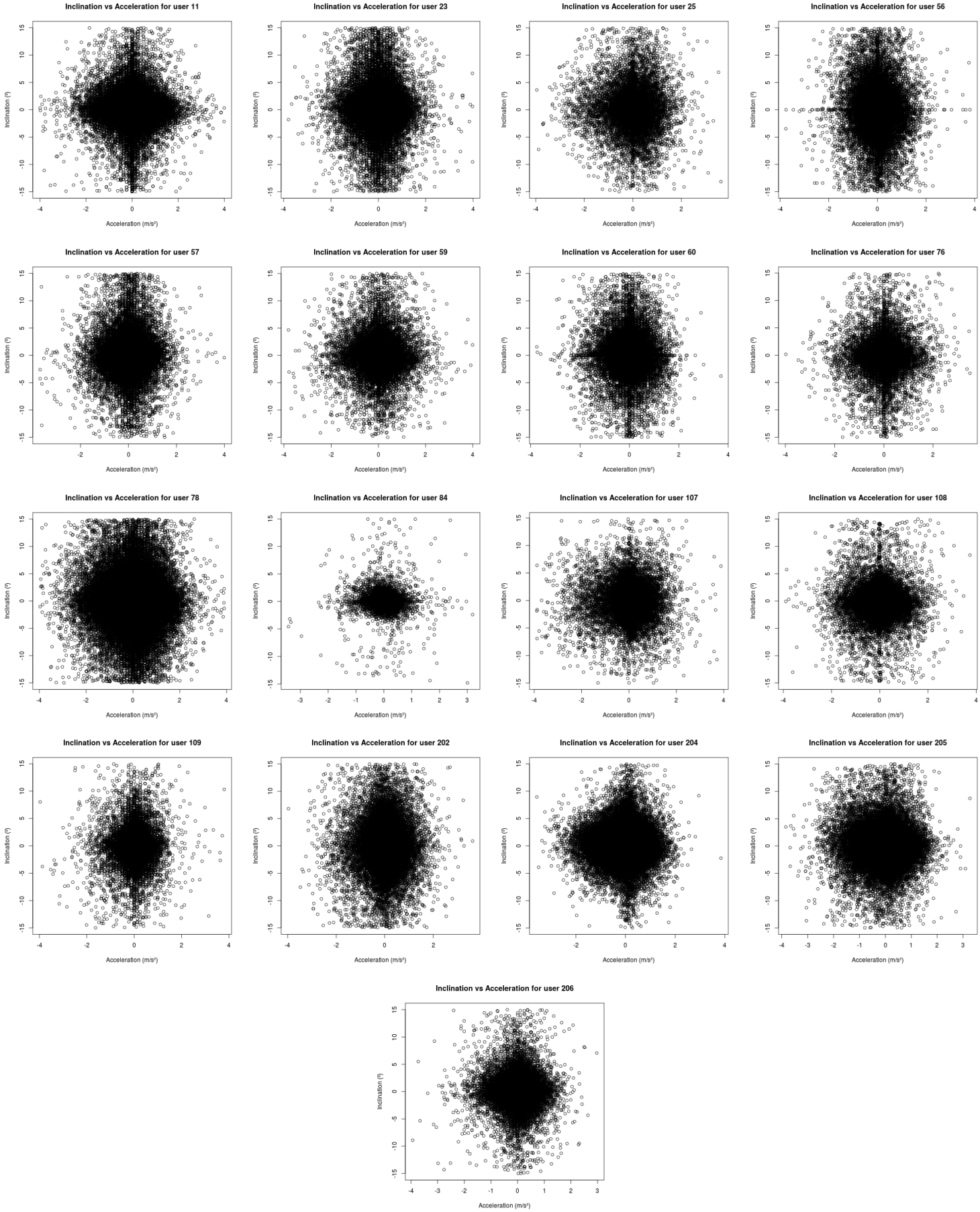


Figure A.7: Inclination vs acceleration.

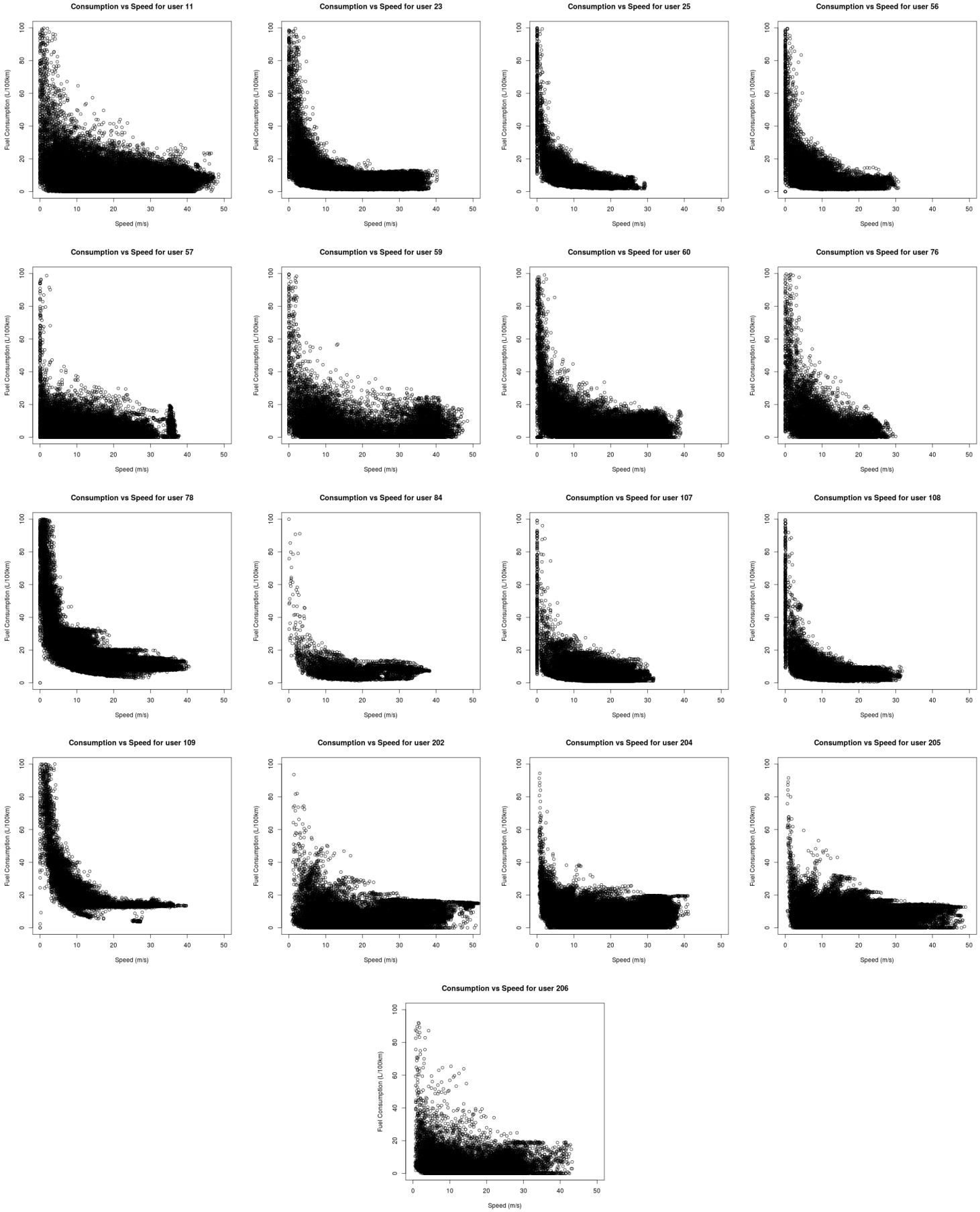


Figure A.8: Fuel consumption vs speed.

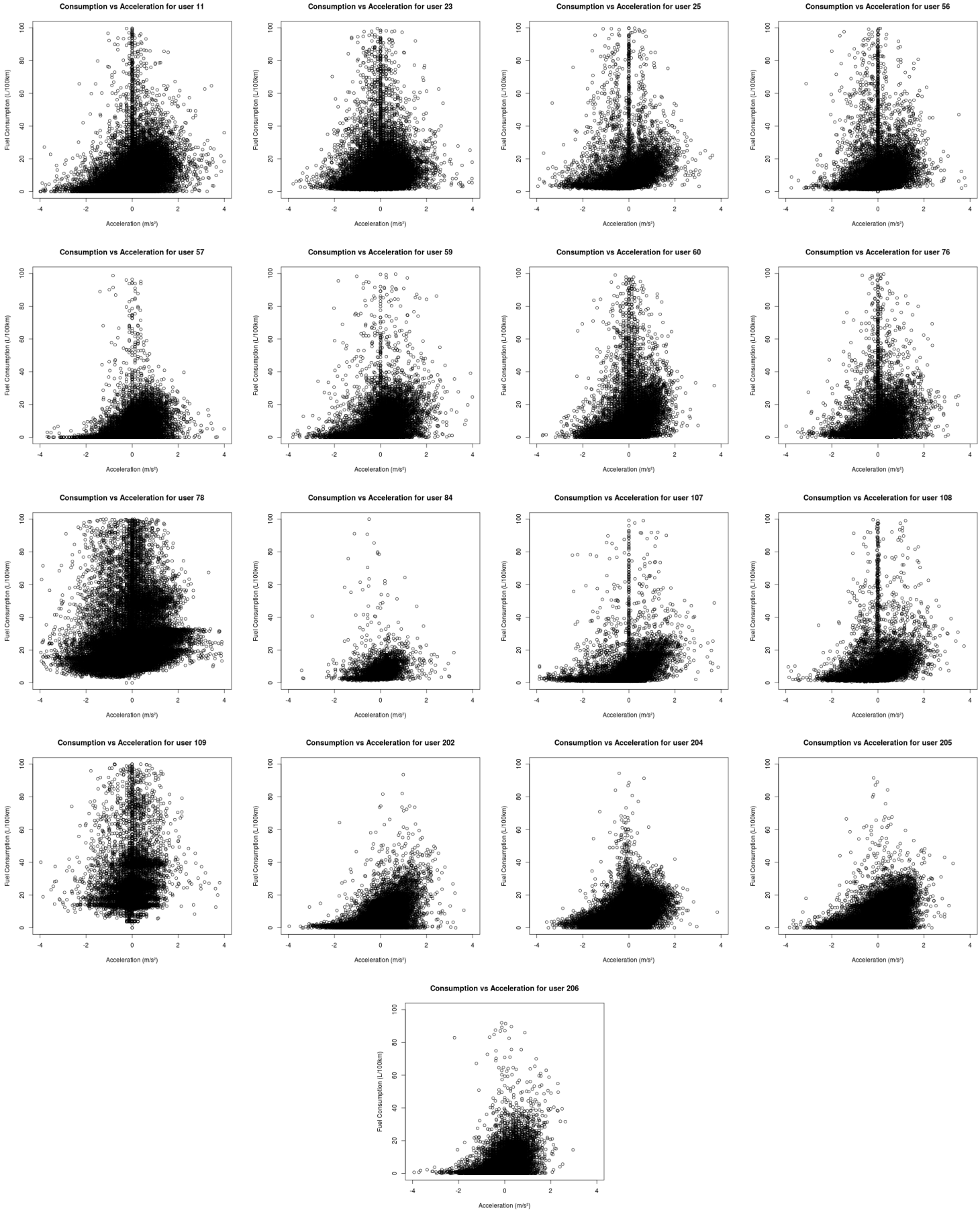


Figure A.9: Fuel consumption vs acceleration.

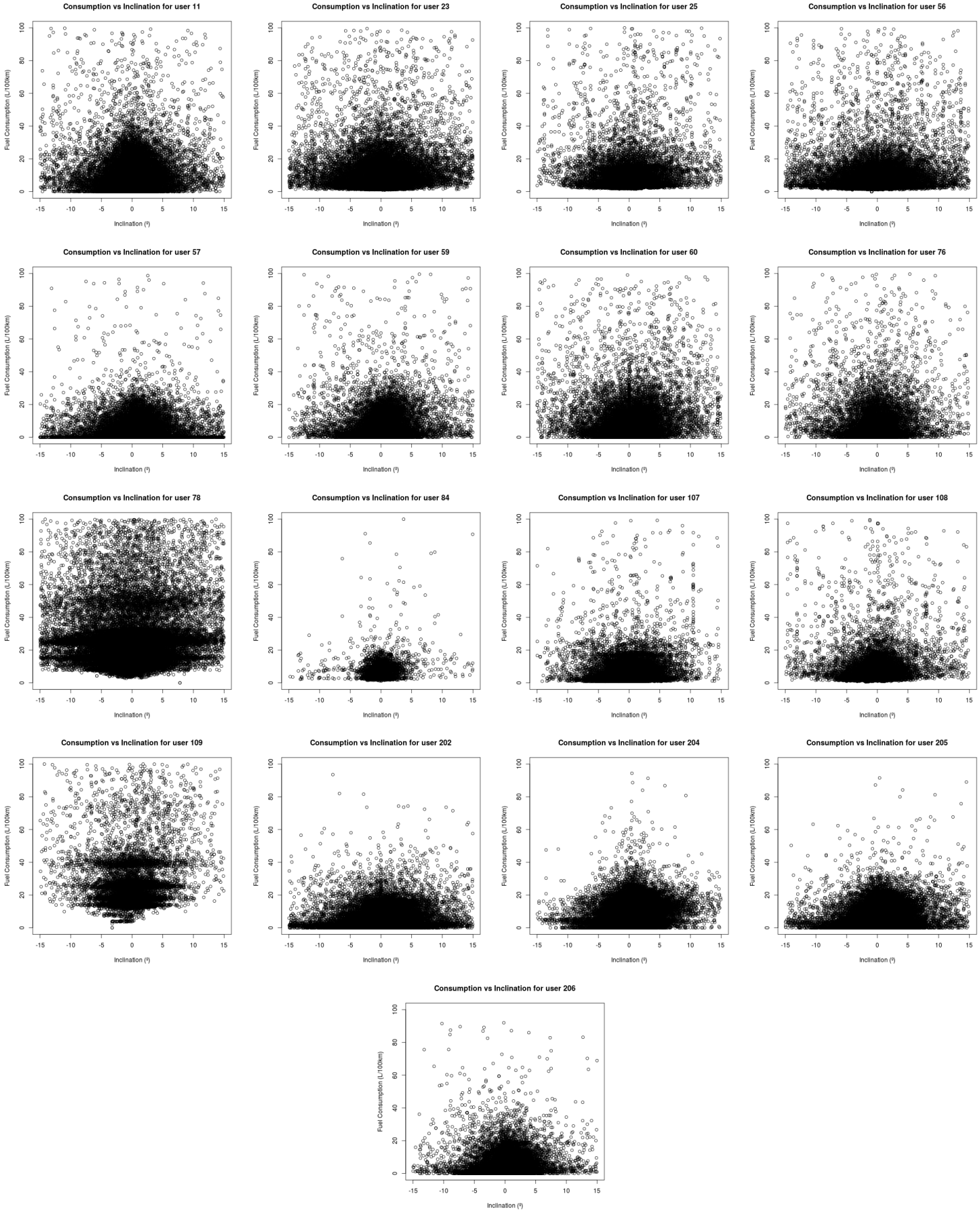


Figure A.10: Fuel consumption vs inclination.

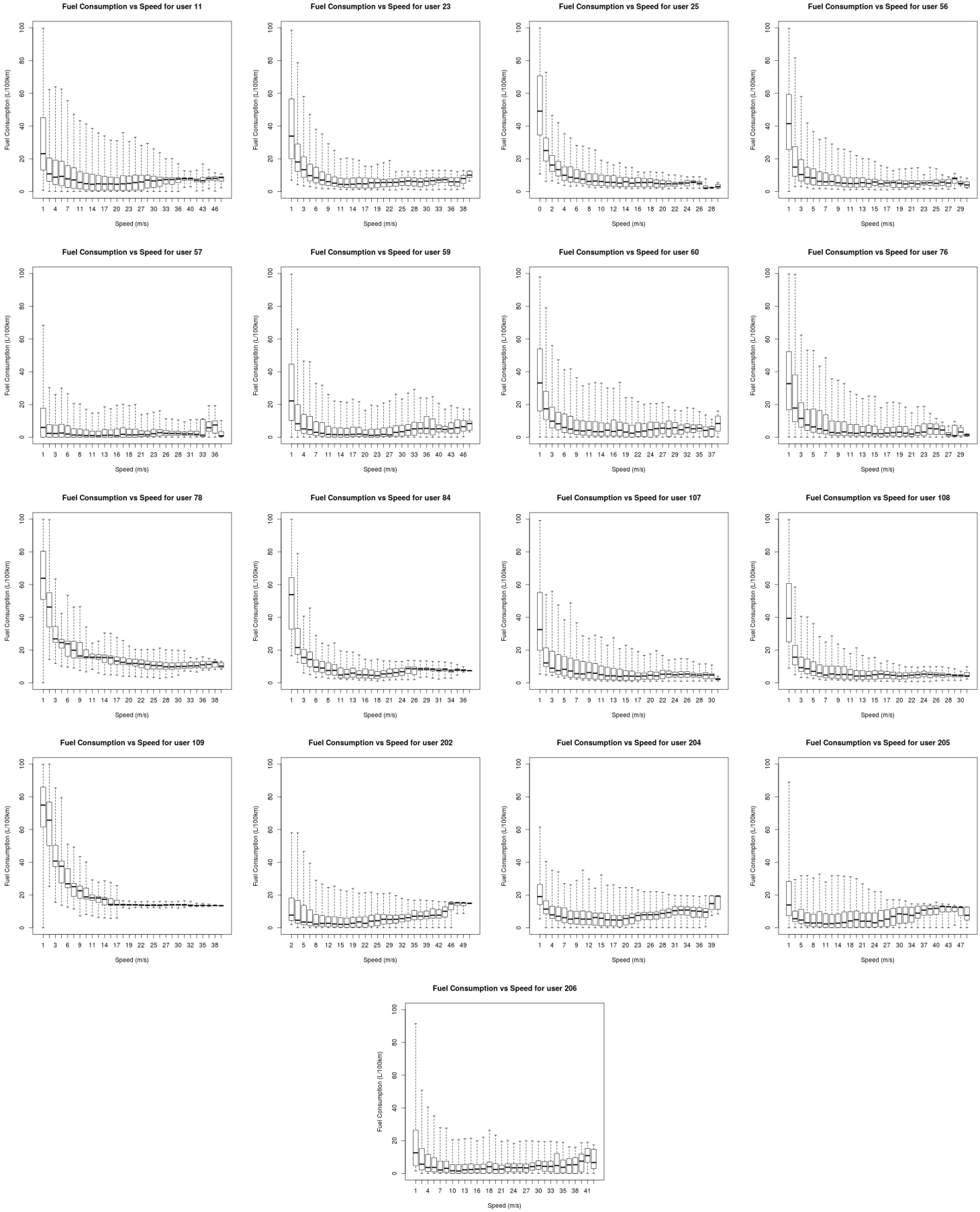


Figure A.11: Box plot of fuel consumption vs speed.

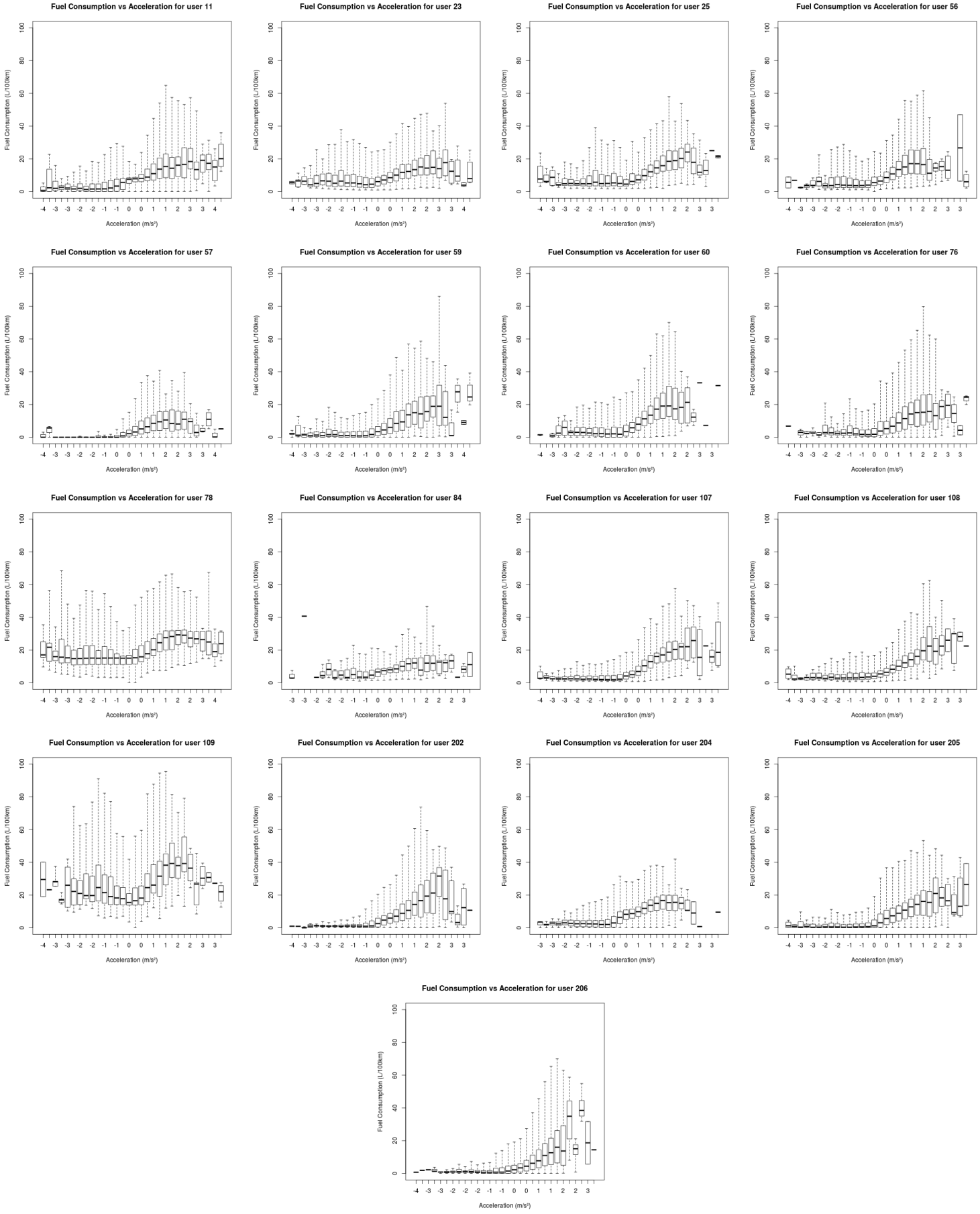


Figure A.12: Box plot of fuel consumption vs acceleration.

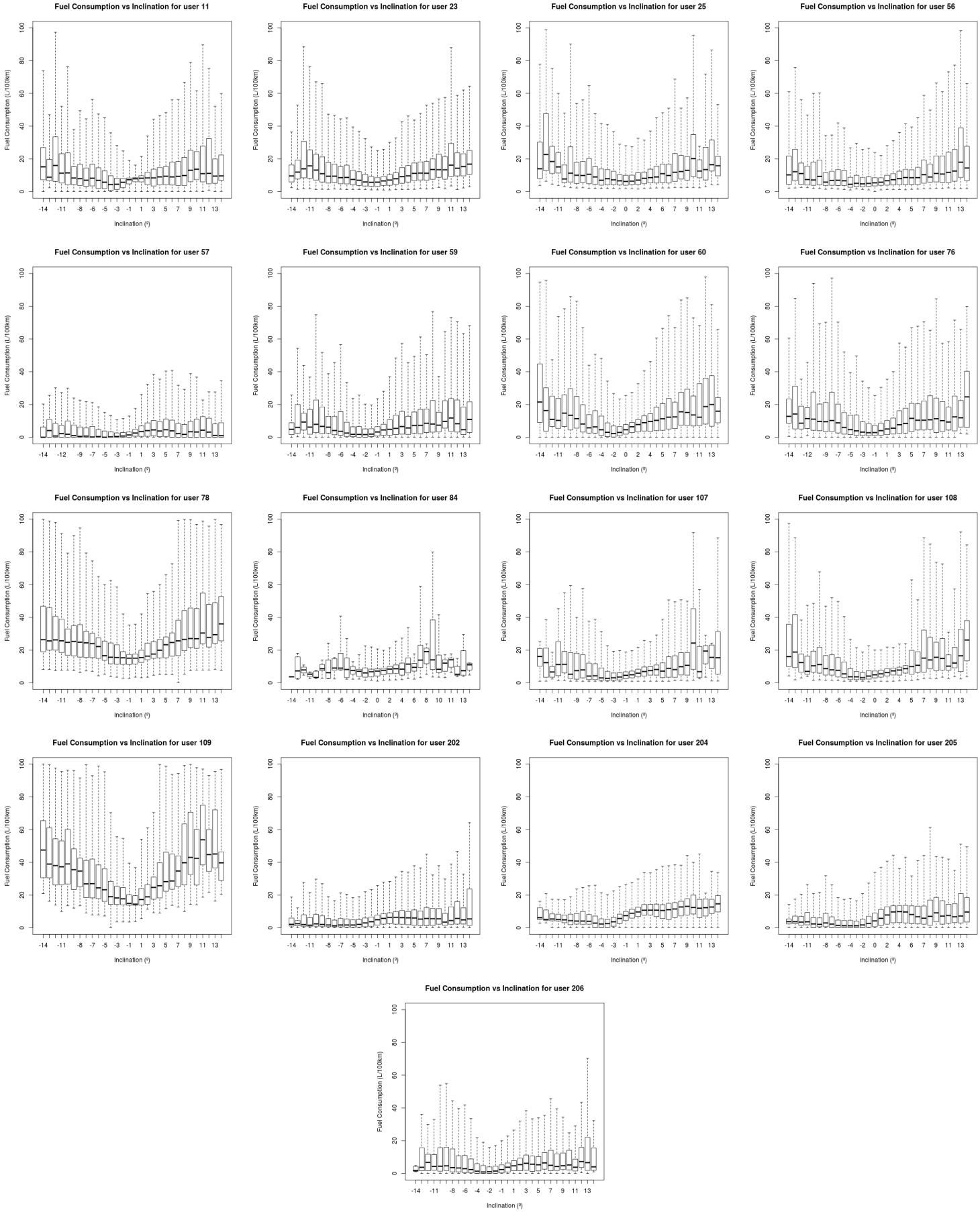


Figure A.13: Box plot of fuel consumption vs inclination.

Appendix B

Additional Graphics and Plots from Results

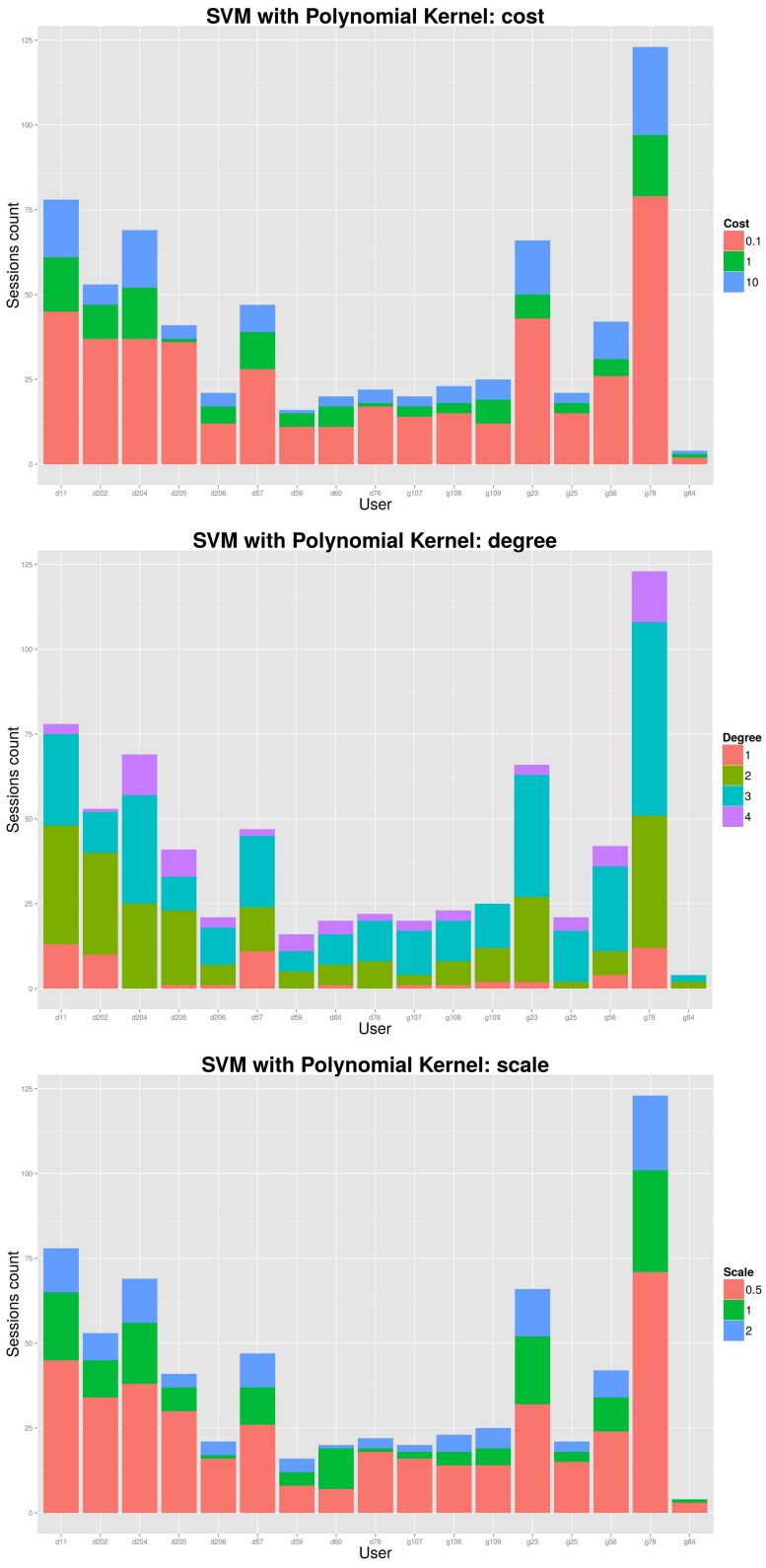


Figure B.1: Parameter selection for SVM with polyniomial kernel function.

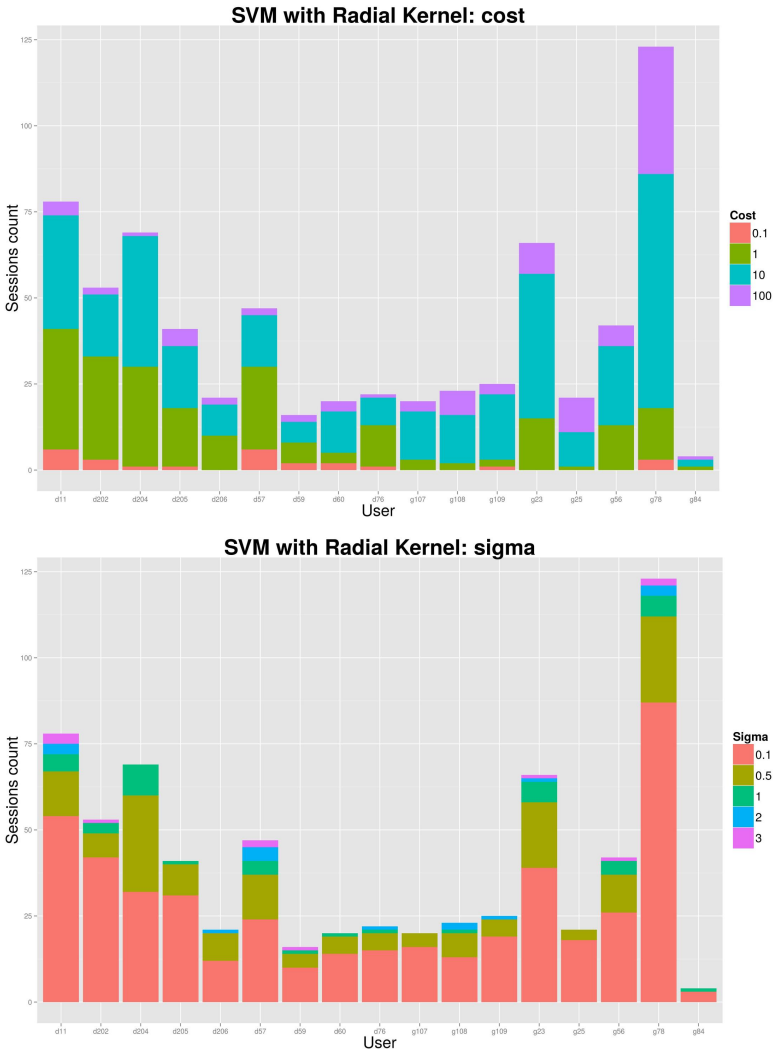


Figure B.2: Parameter selection for SVM with RBF kernel function.

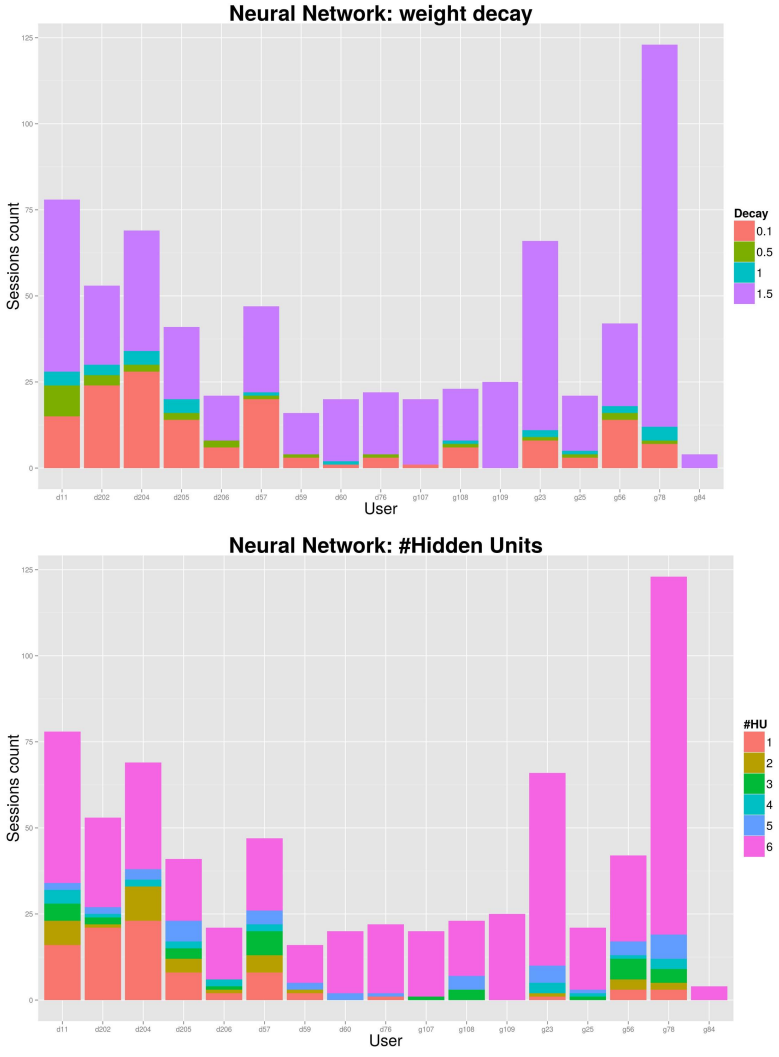


Figure B.3: Parameter selection for ANN

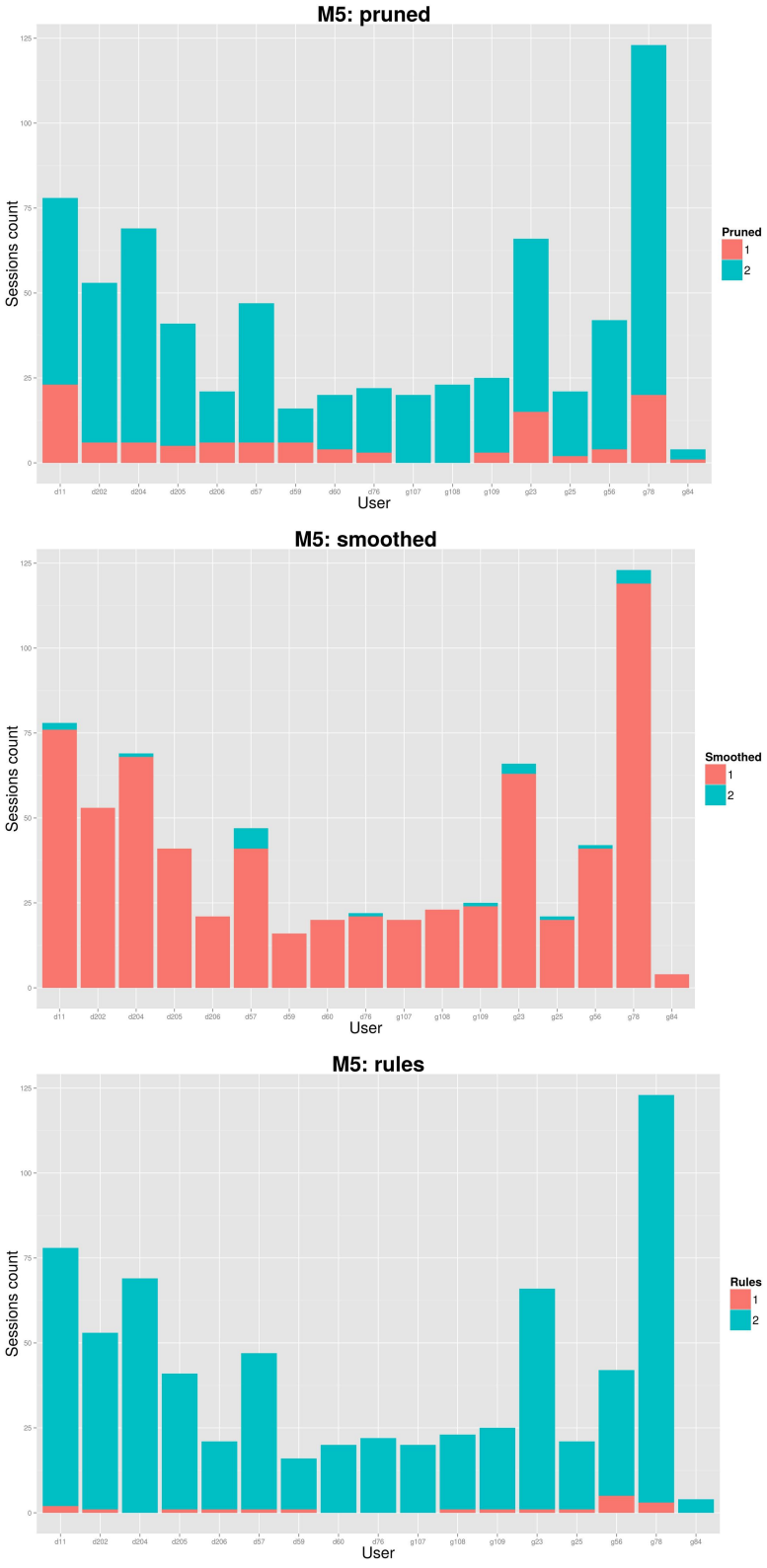


Figure B.4: Parameter selection for M5.

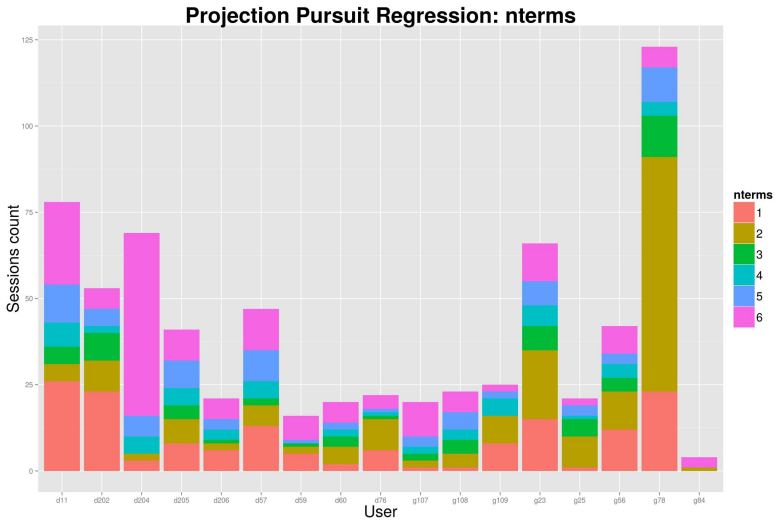


Figure B.5: Parameter selection for PPR.

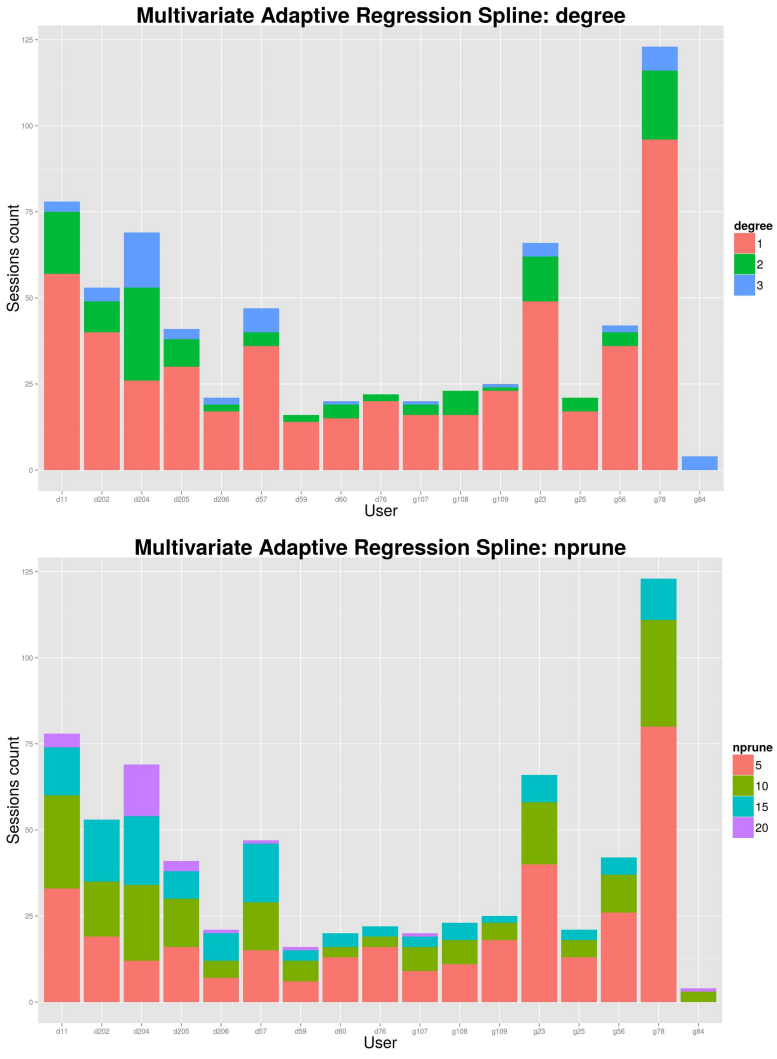


Figure B.6: Parameter selection for MARS.

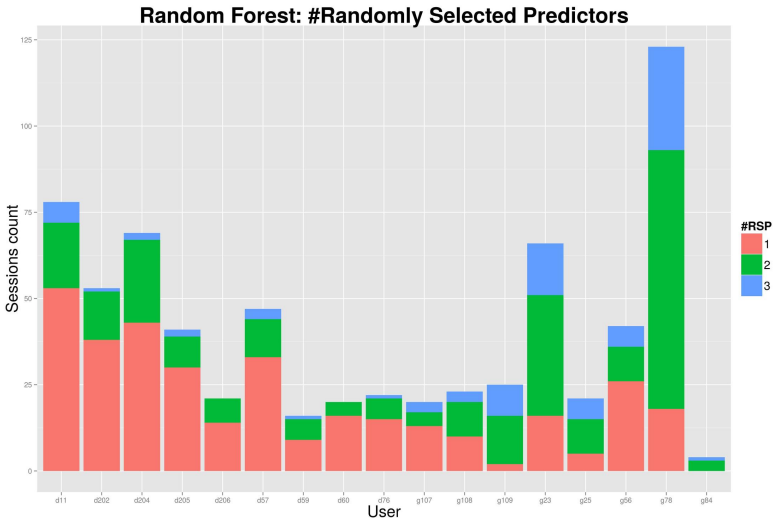


Figure B.7: Parameter selection for RF.

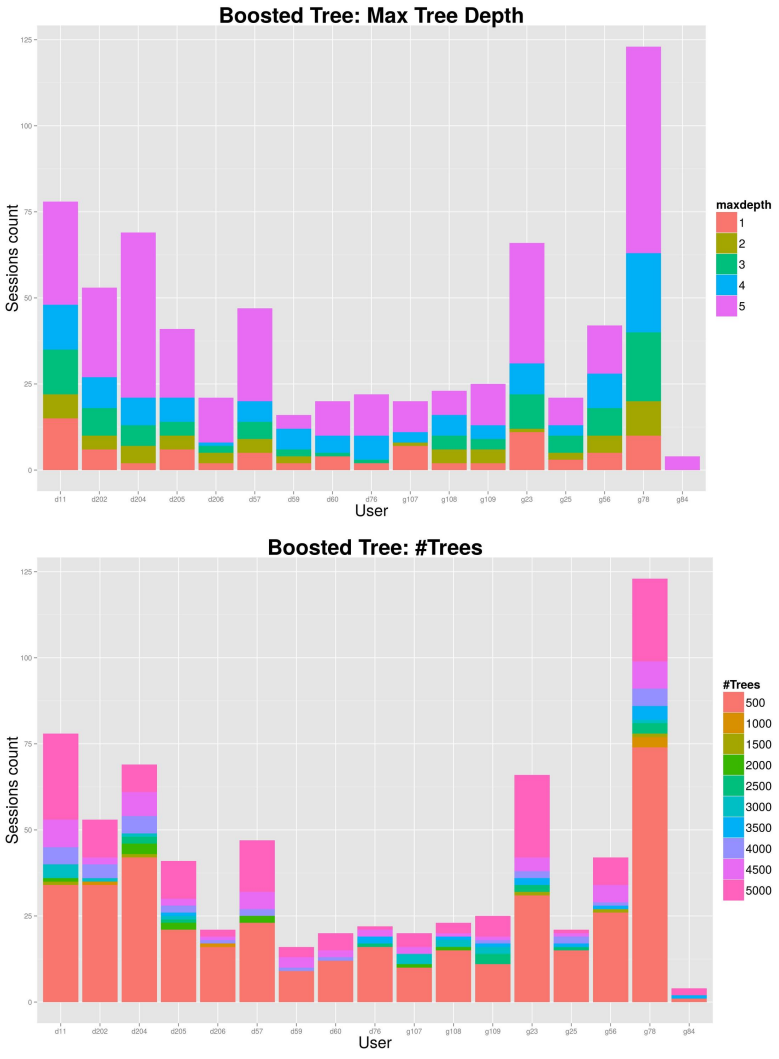


Figure B.8: Parameter selection for Boosted Tree.

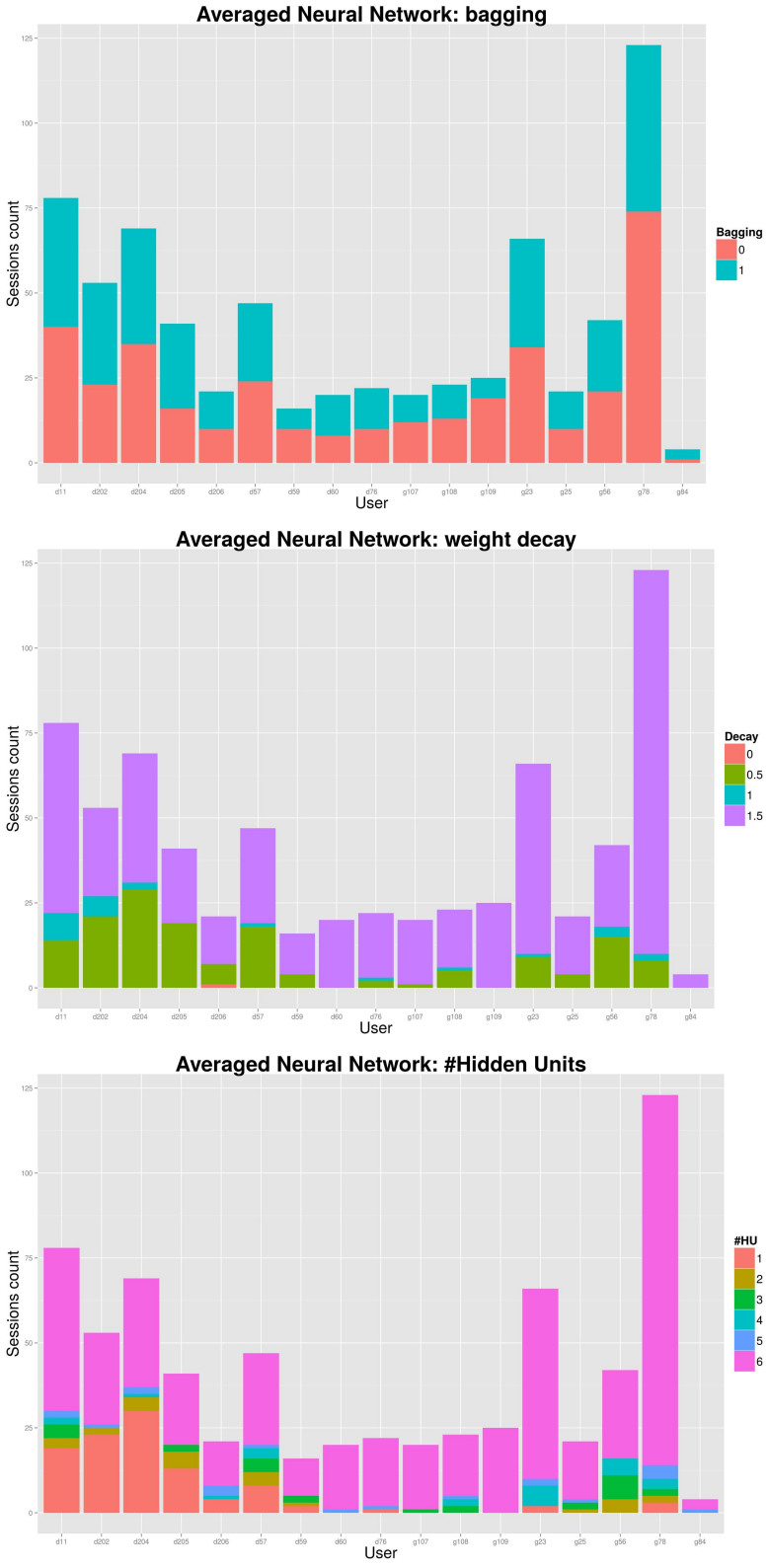
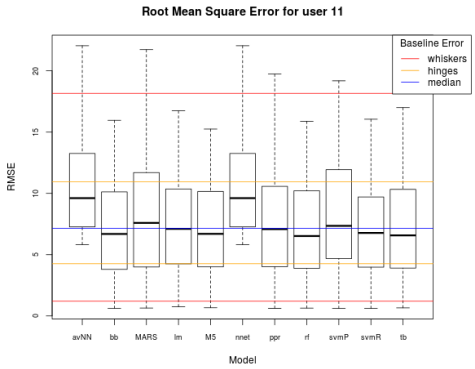
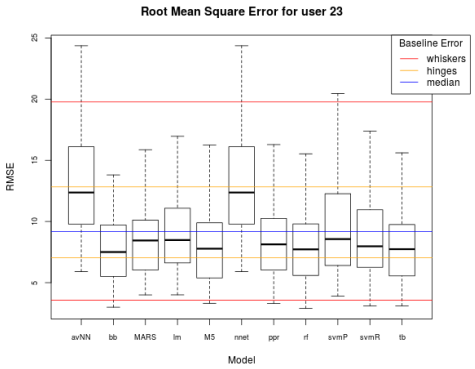
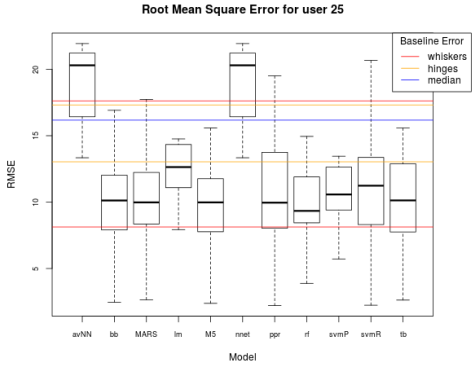
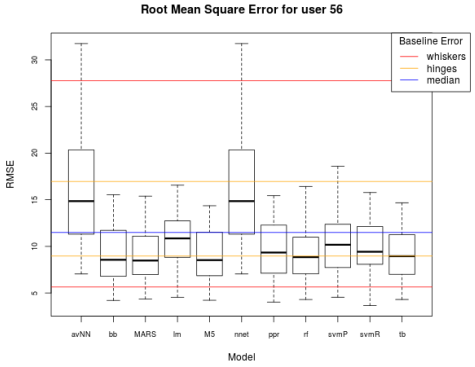


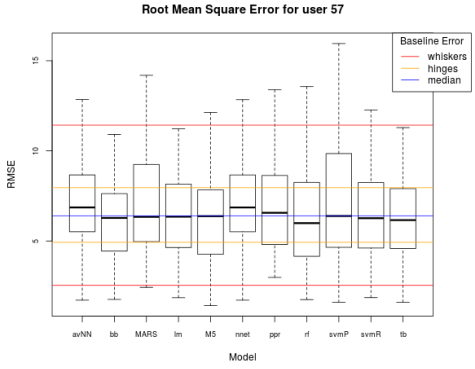
Figure B.9: Parameter selection for Average Artifitial Neural Networks.

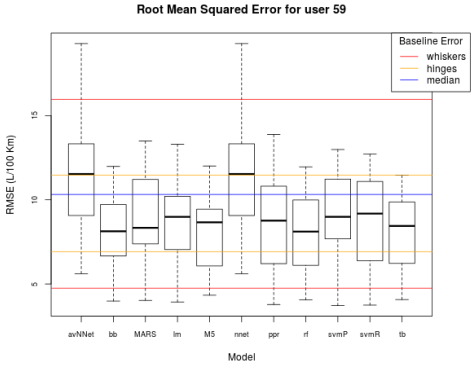

Root Mean Square Error for user 11
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb

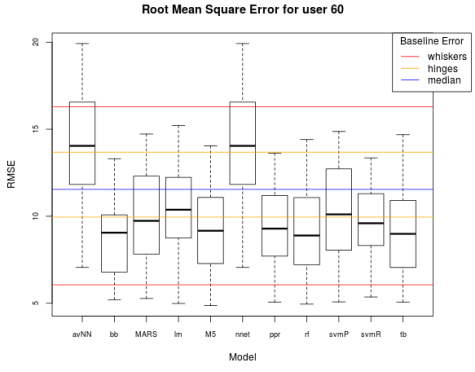

Root Mean Square Error for user 23
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb

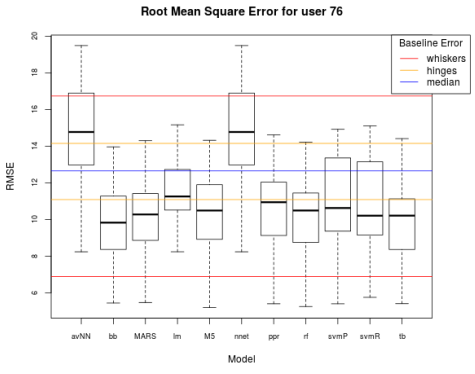

Root Mean Square Error for user 25
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb

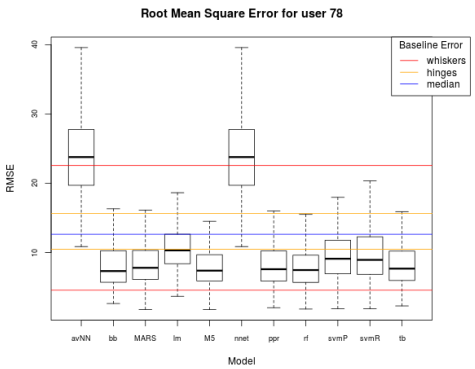

Root Mean Square Error for user 56
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb


Root Mean Square Error for user 57
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb


Root Mean Squared Error for user 59
Baseline Error
whiskers
hinges
median
RMSE (L/100 Km)
Model
avNNet bb MARS lm M5 nnet ppr rf svmP svmR tb


Root Mean Square Error for user 60
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb


Root Mean Square Error for user 76
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb


Root Mean Square Error for user 78
Baseline Error
whiskers
hinges
median
RMSE
Model
avNN bb MARS lm M5 nnet ppr rf svmP svmR tb

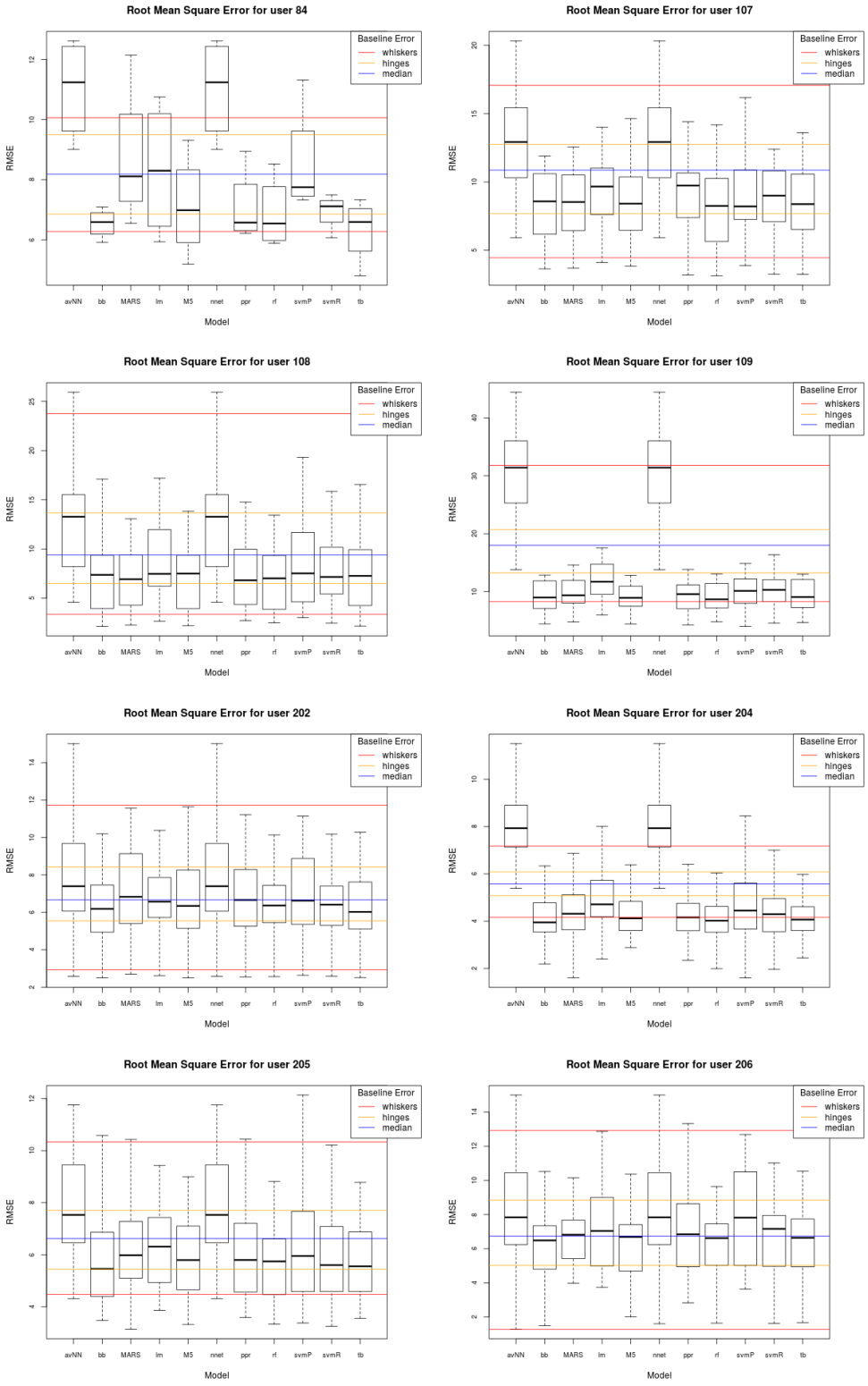
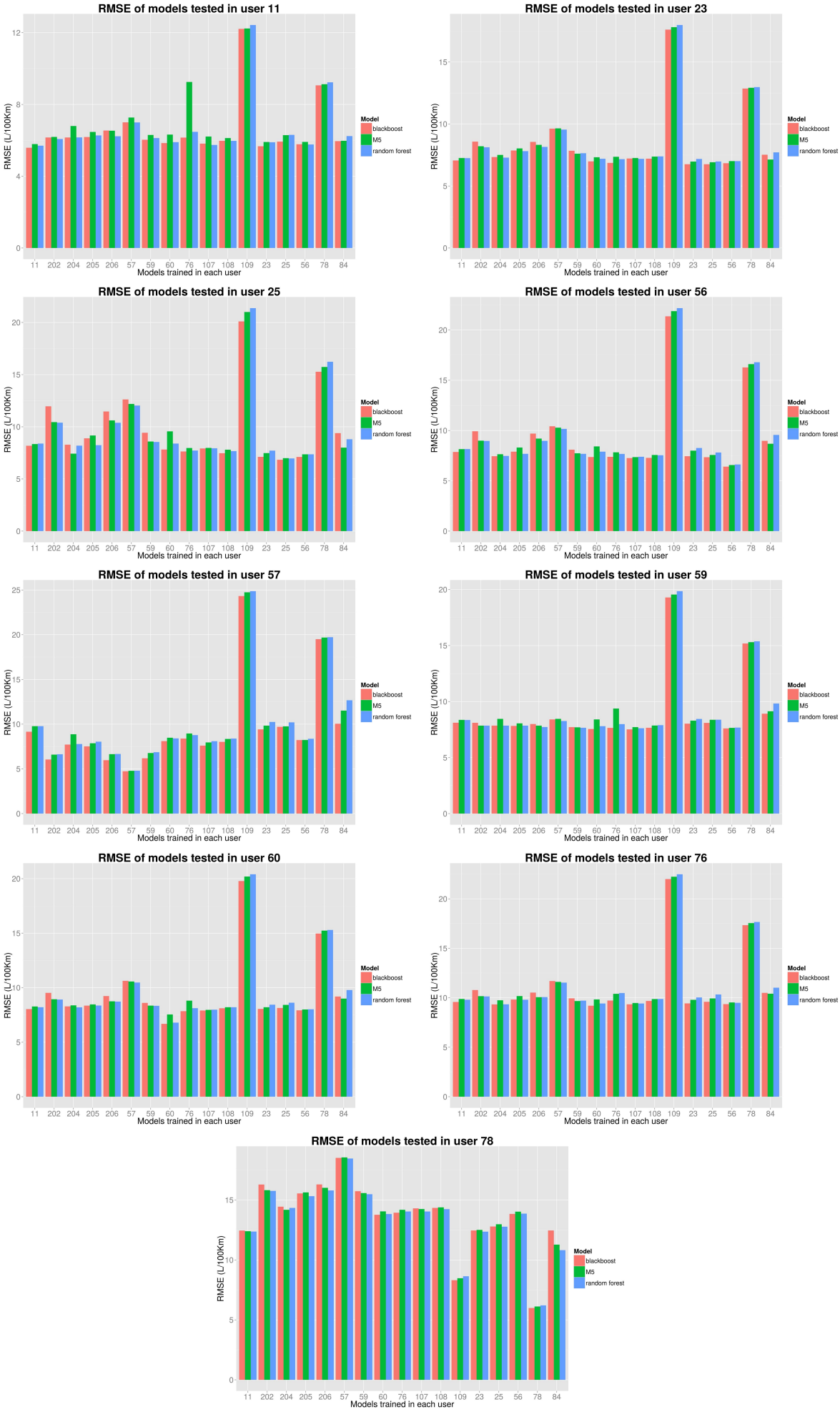


Figure B.10: Box plots of RMSE for each model.


RMSE of models tested in user 11
RMSE of models tested in user 23
RMSE of models tested in user 25
RMSE of models tested in user 56
RMSE of models tested in user 57
RMSE of models tested in user 59
RMSE of models tested in user 60
RMSE of models tested in user 76
RMSE of models tested in user 78
RMSE (L/100Km)
Models trained in each user
11 202 204 205 206 57 59 60 76 107 108 109 23 25 56 78 84
Model
blackboost
M5
random forest

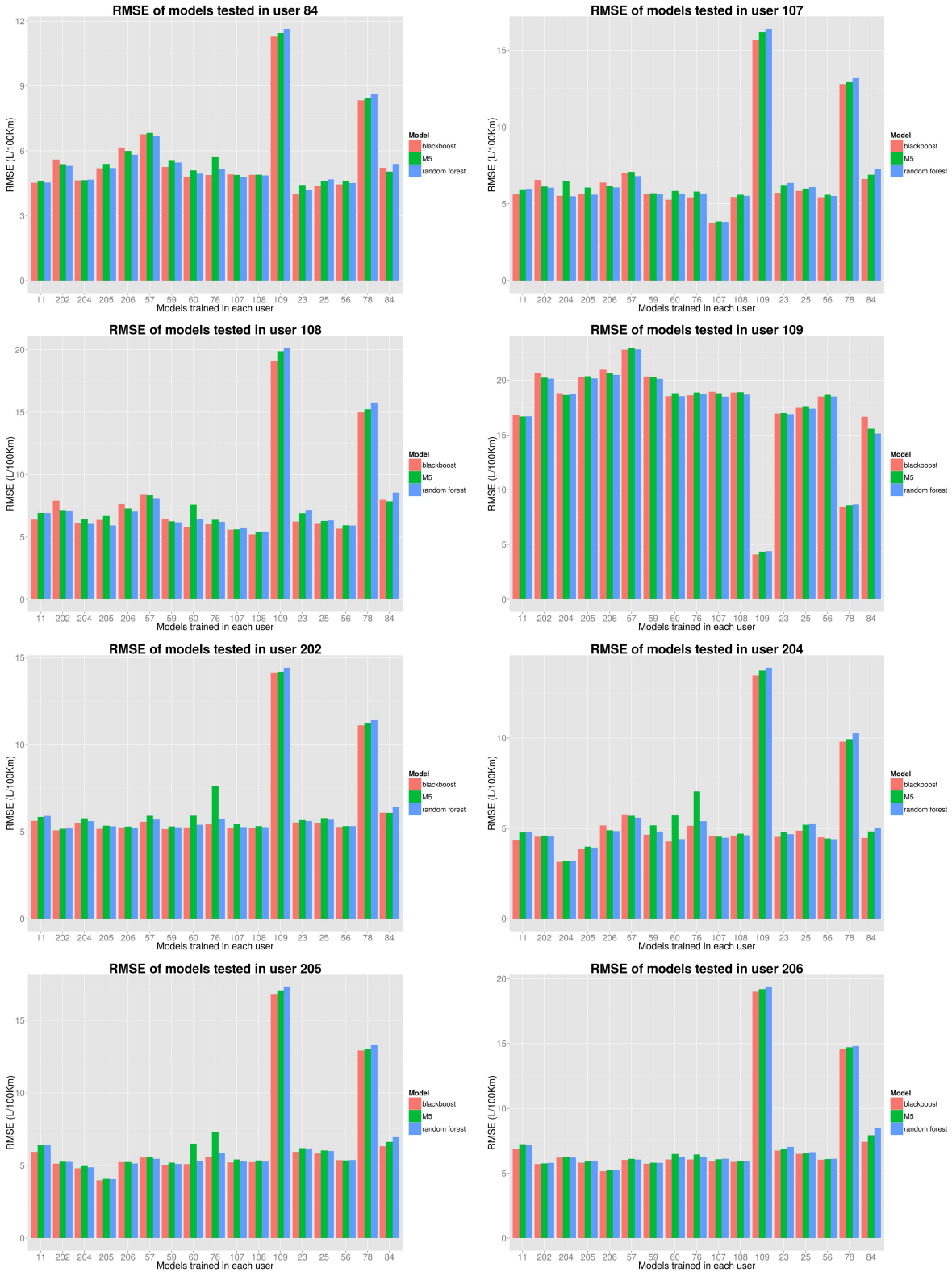


Figure B.11: RMSE of models from different vehicles tested in each vehicle.

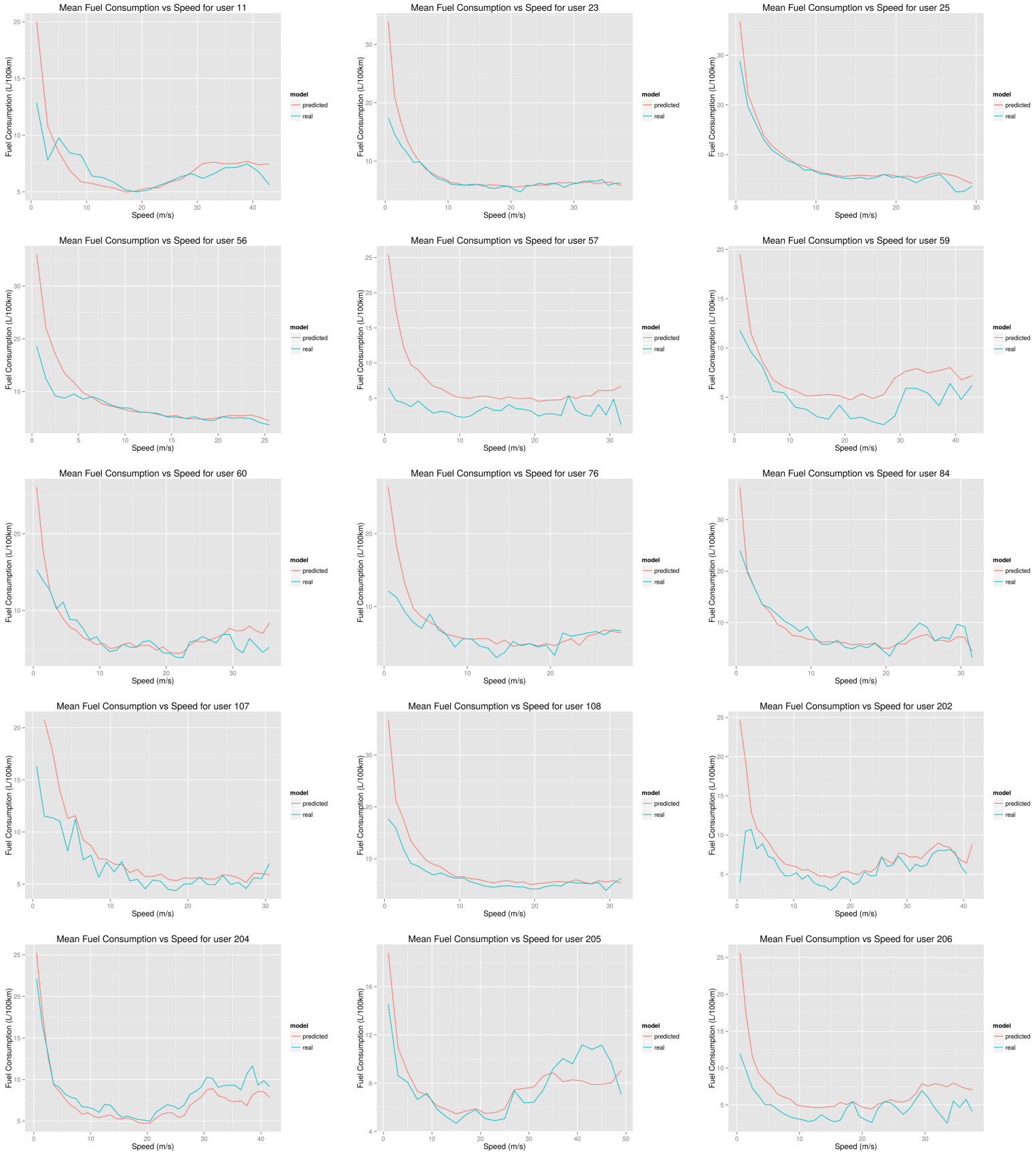


Figure B.12: Mean fuel consumption vs speed. Real and predicted curves from the general model.

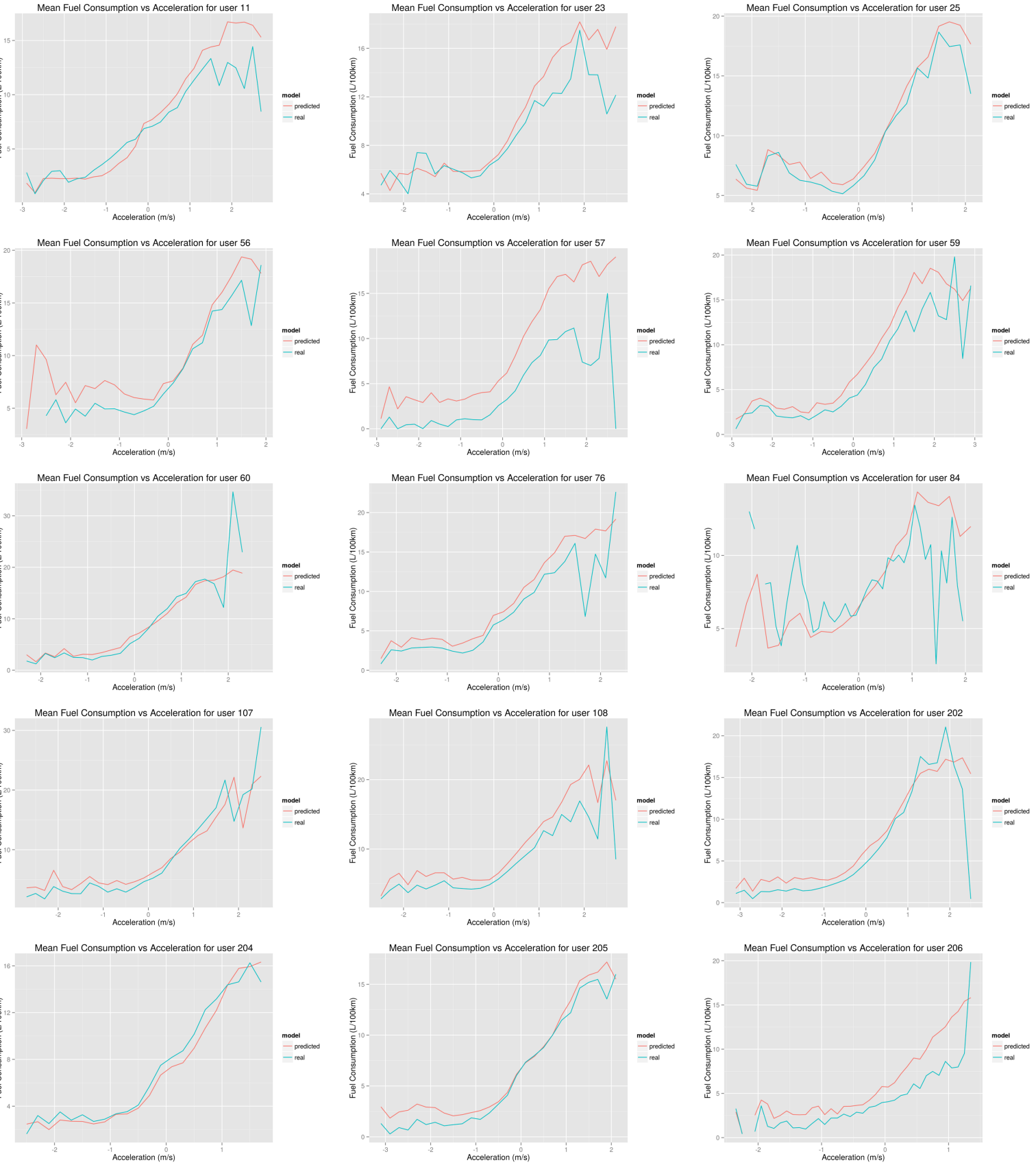


Figure B.13: Mean fuel consumption vs acceleration. Real and predicted curves from the general model.

Figure B.14: Mean fuel consumption vs inclination. Real and predicted curves from the general model.

References

Kyoungho Ahn, Hesham Rakha, Antonio Trani, and Michel Van Aerde. Estimating vehicle fuel consumption and emissions based on instantaneous speed and acceleration levels. *Journal of Transportation Engineering*, 2002.

European-Commission Energy and Transport-DG. Eu tranport in figures, January 2000. http://www.uni-mannheim.de/edz/pdf/2000/transstat.pdf.

Michel André, Mario Keller, Åke Sjödin, Marie Gadrat, Ian Mc Crae, and Panagiota Dilara. The artemis european tools for estimating the transport pollutant emissions. *18th International Emission Inventories Conference p. 1-10*, 2009.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science + Business Media, LLC, 2006.

D. P. Bowyer, R. Akçelik, and D. C. Biggs. Guide to fuel consumption analysis for urban traffic management. *Special Report SR No. 32. ARRB Transport Research Ltd, Vermont South, Australia*, 1985.

Leo Breiman. Bagging predictors. *Machine Learning, 24, 123–140, Kluwer Academic Publishers, Boston*, 1996.

Leo Breiman. Random forests. *Machine Learning, 45*, 2001.

Alessandra Cappiello, Ismail Chabini, Edward K. Nam, Alessandro Luè, and Maya Abou Zeid. A statistical model of vehicle emissions and fuel consumption. *IEEE 5th International Conference on Intelligent Transportation Systems*, 2002.

Eva Ericsson, Hanna Larsson, and Karin Brundell-Freij. Optimizing route choice for lowest fuel consumption – potential effects of a new driver support tool. *Transportation Research Part C 14*, 2006.

Michel Ferreira and Pedro M. d'Orey. On the impact of virtual traffic lights on carbon emissions mitigation. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 13, NO. 1*, 2012.

Jerome H. Friedman. Multivariate adaptive regression splines. *The Annals of Statistics, Vol. 19, No. 1, pp. 1-67*, 1991.

Jerome H. Friedman. Stochastic gradient boosting. *Stanford University*, 1999.

Jerome H. Friedman and Werner Stuetzle. Projection pursuit regression. *Journal of the American Statistical Association, Vol. 76, No. 376, pp. 817-823*, 1981.

Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, Elsevier, 2nd edition, 2006.

Geoffrey Holmes, Mark Hall, and Eibe Prank. Generating rule sets from model trees. *12th Australian Joint Conference on Artificial Intelligence, AI'99 Sydney, Australia*, 1999.

Rob J Hyndman and George Athanasopoulos. Forecasting: principles and practice, 2.5 evaluating forecast accuracy. https://www.otexts.org/fpp/2/5, YEAR=2012.

Jose Luis Jimenez-Palacios. *Understanding and Quantifying Motor Vehicle Emissions with Vehicle Specific Power and TILDAS Remote Sensing*. PhD thesis, Massachusetts Institute of Technology, 1999.

Robert Joumard, Peter Jost, and Dieter Hassel. Hot passenger car emissions modelling as a function of instantaneous speed and acceleration. *The Science of the Total Environment 169*, 1995.

Max Kuhn. *caret: Classification and Regression Training*. R project. http://cran.r-project.org/web/packages/caret/index.html.

Wei Lei, Hui Chen, and Lin Lu. Microscopic emission and fuel consumption modeling for light-duty vehicles using portable emission measurement system data. *World Academy of Science, Engineering and Technology*, 2010.

maisgasolina.com. Evolução dos preços médios dos combustíveis em portugal continental. https://www.maisgasolina.com/estatisticas-dos-combustiveis/.

Michail Masikos, Konstantinos Demestichas, Evgenia Adamopoulou, and Michael Theologou. Emission factors for hdv and validation by tunnel measurements. *Mesoscopic forecasting of vehicular consumption using neural networks*, 2014.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer-Verlag, 2nd edition, 2006.

Luc Pelkmans, Patrick Debal, Tom Hood, Gunther Hauser, and Maria-Rosa Delgado. Development of a simulation tool to calculate fuel consumption and emissions of vehicles operating in dynamic conditions. *SAE Technical Paper*, 2004.

J. R. Quinlan. Learning with continuous classes. *Adams and Sterling, Eds, 343-348, Singapore: World Scientific*, 1992.

R. The r project for statistical computing. http://www.r-project.org/.

R-caret. Data splitting. http://topepo.github.io/caret/splitting.html.

Hesham Rakha, Kyoungho Ahn, and Antonio Trani. Development of vt-micro model for estimating hot stabilized light duty vehicle and truck emissions. *Transportation Research Part D: Transport and Environment, 9(1):49–74*, 2004.

Vitor Ribeiro. Mining geographic data for fuel consumption estimation. Master's thesis, Faculdade de Engenharia da Universidade do Porto, 2013.

Vitor Ribeiro, João Rodrigues, and Ana Aguiar. Mining geographic data for fuel consumption estimation. *16th International IEEE Annual Conference on Intelligent Transportation Systems*, 2013.

Silicon-India-Magazines. World to have more cell phone accounts than people by 2014, January 2013. http://www.siliconindia.com/magazine_articles/World_to_have_more_cell_phone_accounts_than_people_by_2014-DASD767476836.html.

Guohua Song, Lei Yu, and Ziqianli Wang. Aggregate fuel consumption model of light-duty vehicles for evaluating effectiveness of traffic management strategies on fuels. *JOURNAL OF TRANSPORTATION ENGINEERING*, 2009.

G. Tavares, Z. Zsigraiova, V. Semiao, and M.G. Carvalho. Optimisation of msw collection routes for minimum fuel consumption using 3d gis modelling. *Waste Management 29*, 2009.

Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition*. Academic Press, Elsevier, 4th edition, 2009.

whatsyourimpact.org. What are the main sources of carbon dioxide emissions? http://whatsyourimpact.org/greenhouse-gases/carbon-dioxide-sources.

Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, Elsevier, 3rd edition, 2011.