Predicting the presence of air pollutants using low cost sensors

Auteur : Mennicken, Luisa
Promoteur(s) : Timmermans, Catherine; Romain, Anne-Claude

Faculté : Faculté des Sciences
Diplôme : Master en sciences mathématiques, à finalité spécialisée en statistique
Année académique : 2017-2018
URI/URL : http://hdl.handle.net/2268.2/4935

University of Liège
Faculty of Sciences
Department of Mathematics
Academic year 2017 - 2018

Predicting the presence of air pollutants using low cost sensors

Preprocessing user interface and linear modelling approach

Thesis presented for the Master's degree
in Mathematical Sciences with focus in statistics

Realised by
Luisa Mennicken

Supervisors
Catherine Timmermans
Anne-Claude Romain

*At this point I would like to thank all those who supported
and motivated me during the preparation of
this Master's thesis and even, more broadly,
during these years of study.*

*I would first like to thank my supervisors,
Ms. Timmermans and Ms. Romain, for having offered me this topic,
giving me the opportunity to link theory and practice.
Thank you also for their valuable advice and suggestions
as well as for their availability throughout this work.*

*Finally, I especially thank my family, my friends and Dominik,
who made my studies only possible with their support.
They always had an open ear for my worries,
gave me fresh heart in difficult times,
shared my happiness about success
and believed in me over the duration
of my entire studies.*

Contents

Introduction

"Air pollution is causing damage to human health and ecosystems. Large parts of the population do not live in a healthy environment, according to current standards. To get on to a sustainable path, Europe will have to be ambitious and go beyond current legislation."

This statement originates from the Belgian scientist Hans Bruyninckx, Executive Director of the European Environment Agency and reveals a very important current topic [4]. According to an estimation of the World Health Organisation (WHO), around 4.2 million people die every year as a result of exposure to ambient outdoor air pollution [8]. By lowering levels of air pollution it is possible to improve the overall health of people around the world, asserts the WHO.

Since 1995, the ULiège Sensing of Atmospheres and Monitoring Laboratory (SAM) is developing low cost chemical sensors devices for monitoring environmental odours in the field. On the one hand, these sensors are advantageous for measurement of mixed compounds, but on the other hand they are unprivileged for single pollutant sensing. However, it is believed that the simultaneous use of several low cost sensors could result in a sufficient information about the concentration of several pollutants. Therefore, an experiment has been conducted in which sensors measurements are compared to the sensing performances of standardized chemical analysers. The analysers are certified and operated by the official Wallonia public scientific institute (ISSeP).

In 2010, the SAM laboratory started a long-term study in collaboration with ISSeP in order to compare the reference analysers with low cost chemical sensors. Both kind of devices worked simultaneously for real time measurements. Their special subject is the comparison of detecting malodours, but also chemical compounds are considered. In this work, we focus exclusively on the second monitoring of chemical target compounds which are methane, ammonia, hydrogen sulphide, and also non-methane and petroleum hydrocarbons like limonene, pinene, benzene, toluene, ethylbenzene and xylene.

The general objective of this study is to search for signatures of the named compounds identified by the reference analysers in all ULiège sensors. Which sensors contribute in the prediction of air pollutants and could be considered for a wide surveillance network?

For the analysis, Prof. Dr. Anne-Claude Romain, responsible for the project of the Department of Environmental Sciences ULiège, made available the real-measurements performed by sensors and analysers for the period 24.08.2016 - 30.01.2017. A data set about the meteorological conditions was also provided to improve the analysis.

This study comprehends two contributions: first, a preprocessing approach and a user interface is constructed for an appropriate pretreatment of the data. Secondly, a linear modelling is developed with its performances, limits and perspectives.

In Chapter 1 "Data collection" the functioning and installation of the sensors and analysers is explained. This section illustrates how the data have been collected by the two instruments. The response times of sensors and analysers are discussed as well.

Chapter 2 "Data preprocessing" consists on the process of pretreatment of the data to enable the comparison of data from sensors and analysers. It must be noticed that the complete preprocessing can be reproduced for new similar data.

Next, there will be a descriptive analysis of the available data in Chapter 3 "Data description and subsequent pretreatment". The measurements of each chemical sensor and each analyser will be explored and all anomalies or missing values will be detected.

Chapter 4 "Graphical user interface" illustrates the user interface created for the data preprocessing in a Shiny application.

Once the data set is complete, the search of a predictive model can begin. In Chapter 5 "Predictive model", the procedure of finding a model for every analyser as well as their qualities will be explained in detail.

In the last Chapter 6 "Conclusion and perspectives", we discuss about the obtained results and possible perspectives for a further study with these data.

Objective and scope of the study The aim of this work consists in a search for signatures of different compounds identified by reference analysers in the ULiège sensors. More in detail, we want to predict the concentrations of the components in the atmosphere. It is considered that the actual concentrations of these components are as reported by the analysers. For this work, we concentrate on linear models to predict with our multivariate sensor data. We illustrate this in general:

Let Y_{ana} be the variable containing the measurements from one of the analysers. We want to explain this variable Y_{ana} in terms of the data given by all sensors. In other words, we search a function f such that

$$Y_{ana} = f(S_1, S_2, ..., S_n)$$

where f is a linear function, $S_1, S_2, ..., S_n$ the measurements of all sensors and n the total number of sensors used in the prediction. More in detail, the search of the function f consists in a search of real coefficients $\alpha_1, ..., \alpha_n$ so that

$$\begin{aligned} Y_{ana} &= f(S_1, S_2, ..., S_n) \\ &= \alpha_1 S_1 + ... + \alpha_n S_n. \end{aligned}$$

The coefficients $\alpha_1, ..., \alpha_n$ will give the contribution of each sensor in the prediction. Moreover, we want to find such a function f, which could apply to all different analysers, hence for all specific chemical compounds. Then, the low cost chemical sensors would have the additional advantage of being able to predict specific compounds by the prediction model like the expensive analysers.

A good understanding of the data is necessary for the further preprocessing and analysis of them. Therefore, the installation and functioning of analysers and sensors is explained in detail in the next chapter.

Chapter 1

Data collection

The data for this study originate from two sources: sensors and analysers. The *sensors* are small, low cost devices developed by ULiège SAM Laboratory. The *analysers* are more sophisticated and expensive measuring devices operated by ISSeP. First, we describe the functioning, measuring and limits of the sensors. Then, we clarify how the analysers perform measuring. And finally, the installation of the two devices is explained.

1.1 Sensors

Material Sensors data were captured by six non specific sensors having particular properties and being identified by codes. Those sensors are non-specific, meaning that they are sensitive to the presence of multiple chemical components. Nevertheless, their chemical properties are such that we may expect each of them to be sensitive to some specific compounds. The specificities of every sensor are resumed in Table 1.1. They are not calibrated and return the signal of electrical resistance every minute [1].

Table 1.1: Table of sensors array

Sensor	Announced compounds selectivity	Observed measurement range
TGS2602	VOC, Hydrogen sulfide and ammoniac	6.13 – 17.81 kOhm
TGS2610	propane and butane	15.39 – 50.86 kOhm
TGS2611	methane	8.38 – 28.53 kOhm
TGS2620	organic solvents, alcohols and carbon monoxide	4.46 – 24.43 kOhm
GGS1330	combustibles gazes	6.62 – 59.82 kOhm
TGS2444	ammoniac	38.4 – 211.3 kOhm

[1] We note that several additional specific sensors dedicated to the detection of VOC, CO_2, H_2S and NH_3 where also installed, but they will not be used in this study. One of the reason is that these substances were not sufficiently present in the landfill during the experiment. A descriptive analysis of these signals can be found in Appendix B.1.

Concerning the device, the sensors array is composed of 6 metal oxide sensors TGS2602, TGS2610, TGS2611, TGS2620, GGS1330 and TGS2444 [2]. The sensors are placed inside a cylindrical chamber in PTFE (200 ml) whose inlet and outlet are respectively centred on the inferior and superior sides of the cylinder-forming size. The sensors are forming a circle perpendicular to the gas flow, to ensure that the same air goes through all the sensors simultaneously. The flow pump placed after the sensor chamber has a regulated flow rate of 250 ml/min. The chamber temperature is kept at 50°C by a heating resistor and natural cooling. Specific software [3] controls the hardware and allows the acquisition of the sensor signals resistance (in kOhm). The sensors resistance was measured every 10 seconds, averaged out in 1 minute steps and stored in local memory. On the following picture (see Figure 1.1), the chemical sensors are represented on the left, the sensors chamber on the right [14].

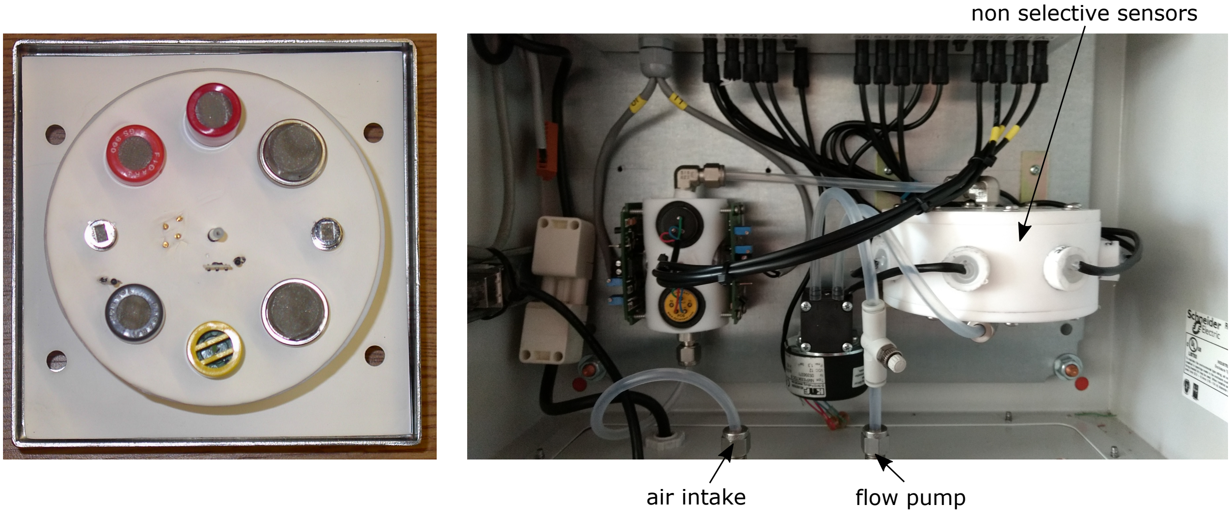


Figure 1.1: Chemical sensors and sensors chamber

Measurement principle The six sensors are metal oxide semiconductors. Oxygen oxidises the material and negative charges are blocked on the surface. As a result, the electric current does not flow. In other words, in the presence of oxygen the resistance is very high. However, if a pollutant is present, the oxygen oxidises more easily the pollutant than the material. There are more free electrons in the material of the semiconductor, the resistance decreases and the current goes faster. So, the appearance of a pollutant results in a decrease in electrical resistance.

Discussion Humidity and temperature may influence the resistance: the latter increases if the humidity and/or the temperature decrease. There may also be other influences related to events affecting the air quality that are detected by the sensors. Besides, given

[2] The five "TGS" sensors are of the brand FIGARO, the "GGS" sensor of the brand UST.

[3] The concerned software has been developed in LabView (NI instrument, USA).

the principle of metal oxides, only the molecules that get oxidised will cause resistance decreasing. The other ones have either no impact on the resistance (will remain stable), or will possibly increase the resistance (for example this is the case for O_3). In conclusion, we have to pay attention in interpreting resistance variations.

1.2 Analysers

Material Analysers data were captured by six analysers, operated by ISSeP. These analysers are specific and target chemical compounds whose concentrations are returned every half hour [4]. The specificities of these analysers are presented in Table 1.2.

Table 1.2: Table of analysers

Analyser	Return	Unit	Measurement range
RMHB09 CH4	concentration of methane	ppm	1.1 – 57.689
RMHB09 H2S	concentration of hydrogen sulphide	$\mu g/m^3$	1 – 341.653
RMHB09 NH3	concentration of ammoniac	$\mu g/m^3$	1 – 124.54
RMHB09 BENZ	concentration of benzene	$\mu g/m^3$	0.1 – 3.5
RMHB09 TOLU	concentration of toluene	$\mu g/m^3$	0.1 – 11.442
RMHB09 LIMO	concentration of limonene	ppm	0.1 – 9.442

Detection method The analysers rely on different detection methods:

- Flame ionization detector: The concentration of CH_4 in the air is measured using this method. The acquisition time of this analyser is equal to 10 seconds and has been averaged to the half hour.

- Fluorescence UV: The analyser dedicated to H_2S first converts H_2S into SO_2, which is then measured by the principle of fluorescence UV [5]. Like the CH_4 analyser before, the provided measurements for every 10 seconds have been averaged to the half hour afterwards.

- The concentration of NH_3 is converted into nitric oxide (NO) measured by the chemiluminescence reaction [6] with ozone. These signals are also averaged for every half hour.

[4] We note that the measurements from three analysers (MPXY, ETBZ and PINE) are also available, but not used in this study because of the too minimal concentrations. A descriptive analysis of these 3 analysers is described in Appendix B.2.

[5] This principle consists in the measurement of fluorescence emission in a wave length of 330 nm.

[6] Chemiluminescence is the emission of light (luminescence), as the result of a chemical reaction (https://www.sciencedirect.com/topics/neuroscience/chemiluminescence).

Analysers dedicated to benzene, toluene and limonene have another measurement principle: The air to be analysed passes through an adsorbent that will adsorb organic compounds during 12 minutes. Then, the adsorbent is heated up/pressed to release the whole fixed mass during 3 minutes. The values are the totals over the sample during 12 minutes. After 15 minutes, this process is repeated and so forth. Afterwards, these values are also averaged to the half hour. The detection process by the aid of an adsorbent is illustrated in the following Figure 1.2.

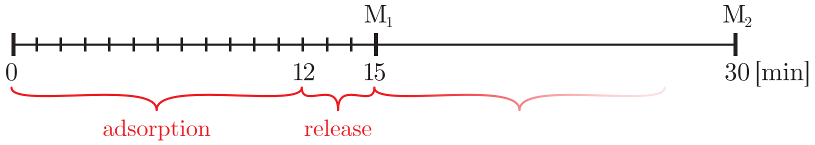


Figure 1.2: Measuring of the adsorbent

1.3 Installation of sensors and analysers

Mobile laboratory trailer Sensors and analysers have been placed within a mobile laboratory trailer. For this study, the trailer was installed at the site of a municipal solid organic waste landfill in a trailer of the ISSeP which was located at Habay (Province of Luxembourg, Wallonia, Belgium). On this location, identified as *Biogas*, sensors and analysers proceeded measuring simultaneously for the period of August 2016 - January 2017.

A weather station and a tube providing air intake are situated on the roof of the trailer. On Figure 1.3 we can see the trailer containing all the measurement instruments [14].

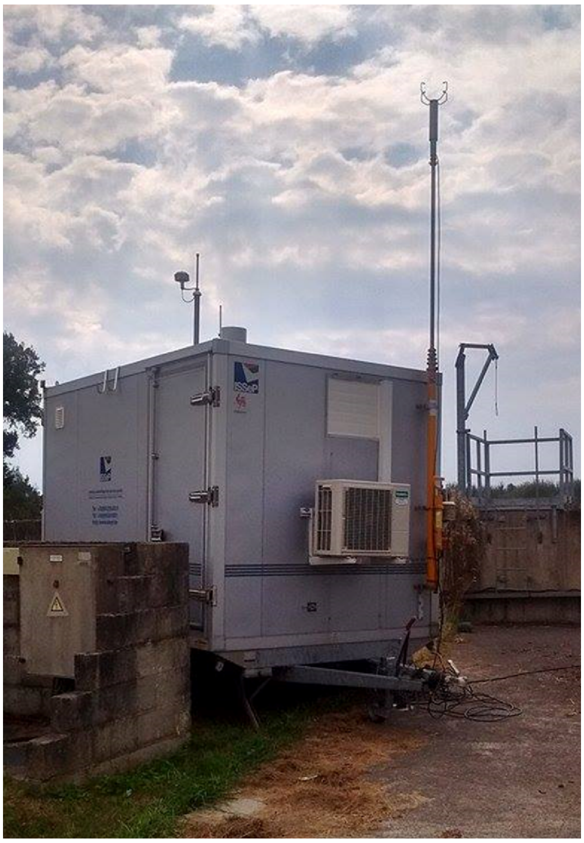


Figure 1.3: Trailer of ISSeP

Installation of the sensors and analysers It is important to mention that sensors and analysers have the same air intake. The air enters through a pipe at 2.8 m from the ground and passes through the chamber of the sensors and analysers respectively, as represented on the following Figure 1.4.

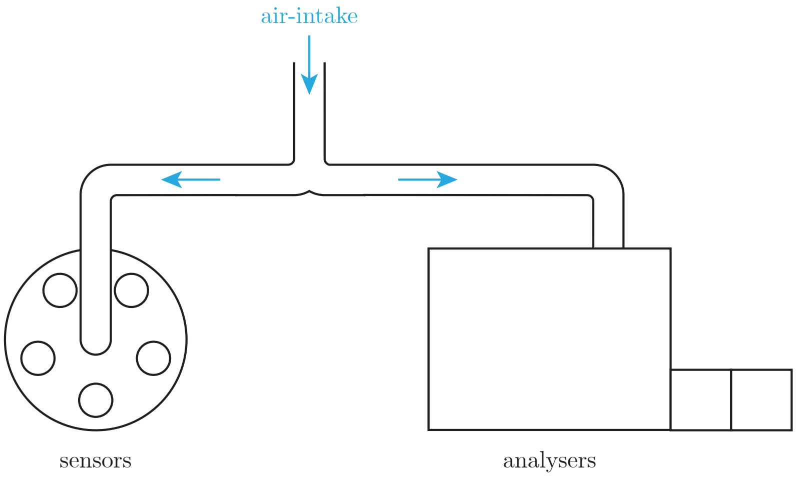


Figure 1.4: Installation of sensors and analysers

Parameters of control In addition to sensors and analysers data collection, several control parameters are measured, which are represented in the following Table 1.3.

Table 1.3: Table of control parameters

Name	**Return**	**Unit**	**Frequency**
Temp. Enc.	temperature around enclosure	°C	per minut
Temp. In.	temperature in enclosure target value : 50°C	°C	per minut
Hr. Enc.	relative humidity in enclosure	%	per minut
Hr. In.	relative humidity around enclosure	%	per minut
RMHB09 DV	wind direction	° 0°= East	per 30 minutes
RMHB09 HR	relative humidity outside	%	per 30 minutes
RMHB 09 PA	atmospheric pressure	hPa	per 30 minutes
RMHB09 TT	outside temperature	°C	per 30 minutes
RMHB09 VV	wind velocity	m/s	per 30 minutes

We call these measurements *parameters of control*, because they are not part of the direct databases produced by sensors and analysers, but more supplementary parameters. Only if special events will be detected in the sensors and analysers sensing, we will compare with the parameters of control, to check every influence.

The first four measurements include the temperature and relative humidity in the enclosure in which the sensors are located, as well as the same measurements around the enclosure for every minute. The temperature in the enclosure is maintained at 50°C with the aid of a heater. The last five measurements provide from a meteorological station from the ISSeP located on the mobile laboratory trailer. This station observed the wind direction, the relative humidity, the atmospheric pressure, the temperature and wind velocity from the laboratory location every half hour.

1.4 Response times

Influences on the response times The lengths and diameters of the pipes bringing air to the chambers of sensors and analysers respectively are not necessarily the same. It follows that the volumes of aspirated air and the response times are not necessarily identical for the two devices. Using volumes, physical and chemical properties, one can calculate the time to arrive at the two different instruments. Moreover, the time of measurement is not equivalent for sensors and analysers as explained in detail before (see Section 1.1 and Section 1.2). Finally, the measuring frequency of the sensors is one minute, whereas it is 30 minutes for the analysers.

Method for determining the response time of sensors and analysers The response time of sensors and analysers can be determined by several approaches. Concerning the data of this study, the response times have been verified and the reactions of both instruments were quasi-instantaneous and simultaneous.

Now, after having a better understanding of the functioning and procedures of measurements, we can carry on with the data preprocessing in the next chapter.

Chapter 2

Data preprocessing

Some preprocessing is needed before we can conduct any analysis on the collected data. First, measurements from sensors, analysers and control parameters were recorded in different files, with different formats and at different time frequencies. Therefore, an intermediate objective in this chapter is the fusion of these data files into a single working dataset. Secondly, some specific time periods cannot be considered for our analysis, as sensors were dedicated to another experiment at that time. Another objective is thus to subset the dataset so that it contains only usable data.

The chapter is organized as follows. We first define the necessary preprocessing for the sensors data file. Afterwards, the process for the analysers data set is explained. Finally, we proceed to the fusion and creation of one large data set, which will be analysed in the subsequent chapter.

At the end of the chapter, we will also introduce the preprocessing of the control parameters data, which is very similar to the one for the sensors and analysers data.

It has to be noticed that the preprocessing is a standard procedure that has to be conducted for any similar study resulting in similar data file structure. At SAM Laboratory, it used to be undertaken "manually", using Excel. On the opposite, we propose a sequence of semi-automatic functions in the software R (version R-3.4.3), which will resolve all intermediate problems. The functions are called *semi-automatic* because there is a necessary intervention of the user to enter data and options and react in case of warning. These functions are designed to be general enough so as to be applied to a dataset coming from similar studies. The R script of all functions of the preprocessing is available on the MatheO platform.

2.1 Summary of the data preprocessing

The complete preprocessing procedure is summarized in the following Figure 2.1.

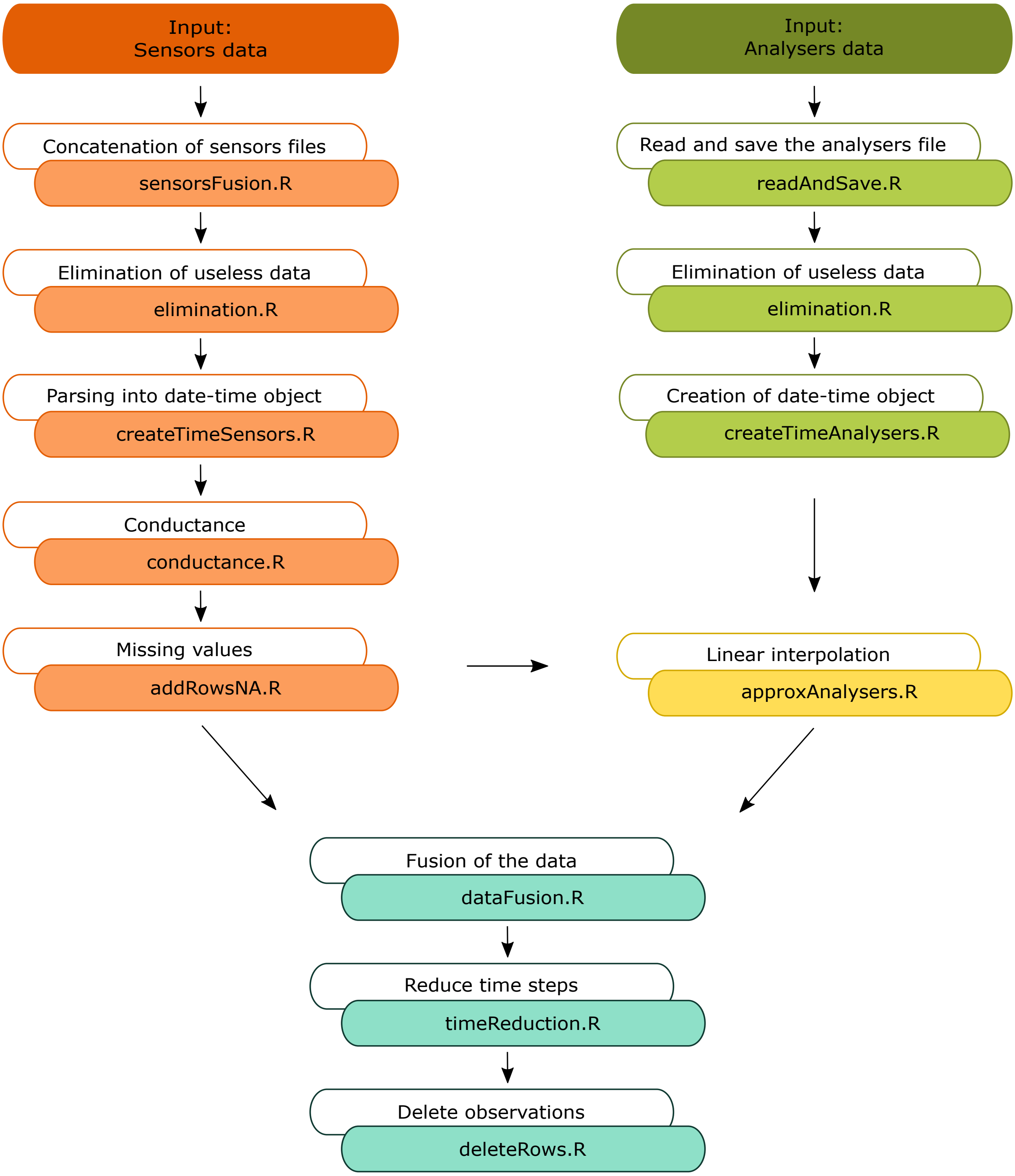


Figure 2.1: Summary of the data preprocessing

This process is general and can be applied to new datasets with a similar structure. Furthermore, a shiny interface is accessible to the user to proceed this preprocessing for new data (see Chapter 4).

2.2 Sensor data file

Sensors file formatting The data file of the sensors must conform certain conditions or have several characteristics for a smooth preprocessing:

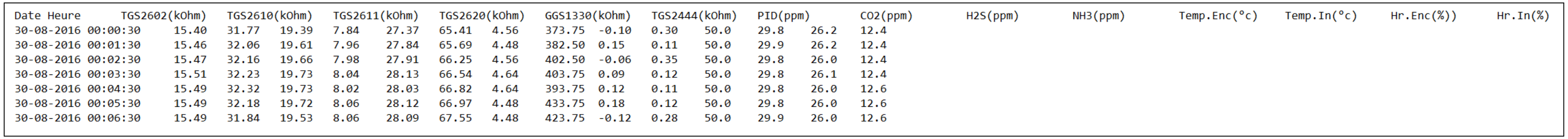
- One column of the data set contains date values, which must have the format day-month-year, e.g. 29.08.2016, the separator being arbitrary.
- Another column contains the time values, which are in the order hour-minute-second, e.g. 13:28:57, and also an arbitrary separator.
- For further preprocessing, the time zone "UTC" is the most adequate because there is no ambiguity about the time lag as for the "MET" time zone for example.
- The sensors measurements are returned as resistance measurements.
- Missing values and the corresponding measurement times are not written in the data set and imply a higher time difference between two consecutive measurements.

The following Table 2.1 contains further information about the datasets of sensors.

Table 2.1: Table of sensors data files

Content	one file per day
Format	text file (.txt)
Period	29.08.2016 13:28:57 - 30.01.2017 23:59:14
Frequency	per minute
Number of columns/sensors measurements	16

The available sensors data starts the 29.08.2016 at 13:28:57 and stops the 30.01.2017 at 23:59:14. The sensors provide measurements for each minute. One data file is created for each day of measurement. Figure 2.2 provides an example of the first lines of such a data file.



```
Date Heure       TGS2602(kOhm)   TGS2610(kOhm)   TGS2611(kOhm)   TGS2620(kOhm)   GGS1330(kOhm)   TGS2444(kOhm)   PID(ppm)        CO2(ppm)        H2S(ppm)        NH3(ppm)        Temp.Enc(°c)    Temp.In(°c)     Hr.Enc(%))      Hr.In(%)
30-08-2016 00:00:30     15.40   31.77   19.39   7.84    27.37   65.41   4.56    373.75  -0.10   0.30    50.0    29.8    26.2    12.4
30-08-2016 00:01:30     15.46   32.06   19.61   7.96    27.84   65.69   4.48    382.50  0.15    0.11    50.0    29.9    26.2    12.4
30-08-2016 00:02:30     15.47   32.16   19.66   7.98    27.91   66.25   4.56    402.50  -0.06   0.35    50.0    29.8    26.0    12.4
30-08-2016 00:03:30     15.51   32.23   19.73   8.04    28.13   66.54   4.64    403.75  0.09    0.12    50.0    29.8    26.1    12.4
30-08-2016 00:04:30     15.49   32.32   19.73   8.02    28.03   66.82   4.64    393.75  0.12    0.11    50.0    29.8    26.0    12.6
30-08-2016 00:05:30     15.49   32.18   19.72   8.06    28.12   66.97   4.48    433.75  0.18    0.12    50.0    29.8    26.0    12.6
30-08-2016 00:06:30     15.49   31.84   19.53   8.06    28.09   67.55   4.48    423.75  -0.12   0.28    50.0    29.9    26.0    12.6
```

Figure 2.2: Header of the sensors data in text editor

2.2.1 Concatenation of sensors files

The first step is to concatenate all the daily files in order to have all sensors measurements in only one file. This concatenation will be executed by the function `sensorsFusion()`:

```
sensorsFusion(files_location,save_location,date_column)
```

Description This function reads the data sets located on `files_location` one by one and creates the concatenation of them, which will be saved on the specified location `save_location` with the specified file name.

Arguments

- `files_location` is a string enclosed in quotation marks containing the path of the file where all the sensors data sets (and them only) are stored. The storage name of these files is not relevant, e.g.: "C:/Users/user/folder/sensors_files". The data sets that have to be concatenated by this function have to include the date values in the specified column in `date_column` to ensure a correct output data set.

- `save_location` is a string in quotation marks containing the desired path directory where the concatenation file will be saved, the name of this file and the extension .csv, e.g.: "C:/Users/user/folder/fileName.csv".

- `date_column` is a quoted string, specifying the exact name of the column containing the date values, e.g.: "Date".

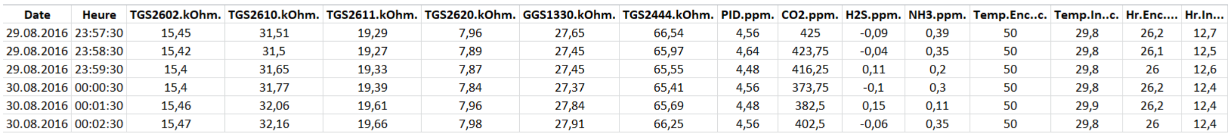
Value The concatenation of the sensors dataset files are written and saved in one CSV file named by the name in `save_location`.

Remarks

- By comparing the date of each file, we check if days are repeated in the `files_location` and if so, a warning message appears but the repeated day is not joined to the file "fileName.csv". If not, the measurements of this day are added to the file "fileName.csv".

- It is recommended to choose another directory in `save_location` as the one specified in `files_location` to keep the sensors files unchanged when the preprocessing will be re-executed afterwards.

The next Figure 2.3 shows a part of the new obtained file which is named "sensorsComplete.csv". We can see that the measurements from the 29.08.2016 are directly followed by the values from the 30.08.2016.



Date	Heure	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS2444.kOhm.	PID.ppm.	CO2.ppm.	H2S.ppm.	NH3.ppm.	Temp.Enc..c.	Temp.In..c.	Hr.Enc....	Hr.In...
29.08.2016	23:57:30	15,45	31,51	19,29	7,96	27,65	66,54	4,56	425	-0,09	0,39	50	29,8	26,2	12,7
29.08.2016	23:58:30	15,42	31,5	19,27	7,89	27,45	65,97	4,64	423,75	-0,04	0,35	50	29,8	26,1	12,5
29.08.2016	23:59:30	15,4	31,65	19,33	7,87	27,45	65,55	4,48	416,25	0,11	0,2	50	29,8	26	12,6
30.08.2016	00:00:30	15,4	31,77	19,39	7,84	27,37	65,41	4,56	373,75	-0,1	0,3	50	29,8	26,2	12,4
30.08.2016	00:01:30	15,46	32,06	19,61	7,96	27,84	65,69	4,48	382,5	0,15	0,11	50	29,9	26,2	12,4
30.08.2016	00:02:30	15,47	32,16	19,66	7,98	27,91	66,25	4,56	402,5	-0,06	0,35	50	29,8	26	12,4

Figure 2.3: Pass from first to second day

2.2.2 Elimination of useless data

One goal of the preprocessing is to keep the useful data and to delete those which are not informative or useful for the analysis. For instance, our sensors dataset contains measurements for the control parameters which are not preprocessed in exactly the same way as sensors measurements. Furthermore, the signals from some additional specific sensors, that is PID, CO_2, H_2S and NH_3, do not give any information (see Appendix B) and thus will be deleted as well. The function named `elimination()` has been written to do this task:

```
elimination(data,var_to_keep)
```

Description The function `elimination()` reads the input dataset `data` and creates a new dataset containing only those columns specified in the argument `var_to_keep`.

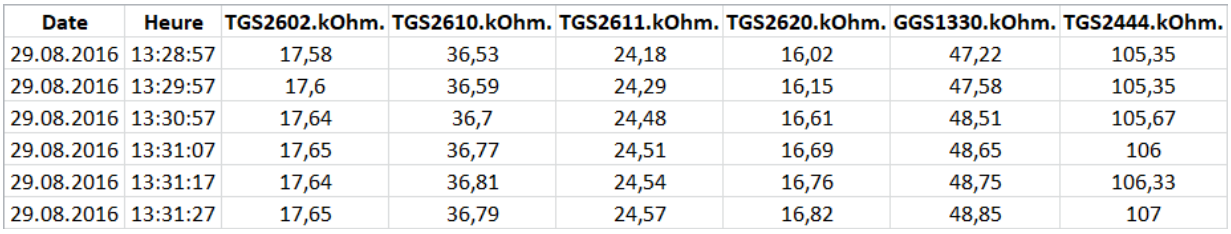
Arguments

- `data` is a data frame containing values we want to retrieve.
- `var_to_keep` is a vector of character strings containing the desired column names of the data set `data` we want to retain.

Value The function `elimination()` returns a new dataset, where the columns not referenced in `var_to_keep` are away. All other columns of `data` will remain in the new dataset without any change.

Remarks In case of misspelling or inserting a wrong column name, a warning message is written and the data set will remain unchanged. If we do not want to eliminate any column of the data set, we simply enter an empty string " " or -1 and the data set will not be changed.

The header of the new dataset is shown on the following Figure 2.4.



Date	Heure	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS2444.kOhm.
29.08.2016	13:28:57	17,58	36,53	24,18	16,02	47,22	105,35
29.08.2016	13:29:57	17,6	36,59	24,29	16,15	47,58	105,35
29.08.2016	13:30:57	17,64	36,7	24,48	16,61	48,51	105,67
29.08.2016	13:31:07	17,65	36,77	24,51	16,69	48,65	106
29.08.2016	13:31:17	17,64	36,81	24,54	16,76	48,75	106,33
29.08.2016	13:31:27	17,65	36,79	24,57	16,82	48,85	107

Figure 2.4: Data set after elimination

2.2.3 Parsing into date-time object

On the new data set (see Figure 2.4), the values in the columns `Date` and `Heure` are stored as factors, and not as date and time objects in R.

In this study, we use the R-package `lubridate` which includes functions to parse date-time data. More in detail, we use the `parse_date_time()` function of this package to parse an input vector into a POSIXct date-time object [6]. The POSIXct class allows handling dates and times with control for time zones, "ct" stands for calendar time. A POSIXct date-time object is stored as the number of days or seconds from some reference date.

To transform the separated date and time values to date-time objects in one column, the function `createTimeSensors()` has been created:

```
createTimeSensors(data,time_zone,date_column,time_column)
```

Description The function `createTimeSensors()` will read the input data set in the argument `data`. This data set must contain dates in the column named in the argument `date_column` in the order day-month-year (29.08.2016) and time values in the column named in `time_column` in the order hour-minute-second (13:28:57). The remaining columns of the input data set contain sensors measurements. The function concatenates the two first columns, then parses the concatenated values into POSIXct date-time objects and returns a new data set containing the date-time objects in the first column, and the remaining sensors measurements from the input data set.

Arguments

- `data` is a data frame containing date values in the column specified in `date_column`, time values in the column `time_column`.
- `time_zone` is a quoted character string that specifies the time zone with which to parse the dates, e.g.: "UTC" for *Coordinated Universal Time* or "MET" for *Middle European Time.*

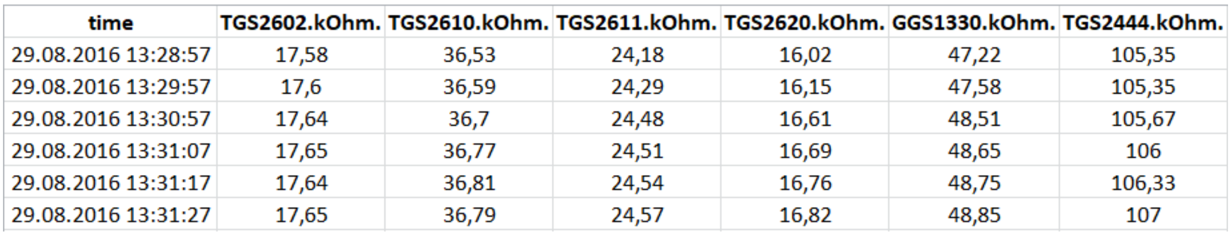
- `date_column` is a character string containing the name of the date column in quotation marks, e.g. "Date".

- `time_column` is a character string containing the name of the time column in quotation marks, e.g. "Heure".

Value The function `createTimeSensors()` has as output a new data set `newData`. The first column of `newData` contains the date-time objects in the specified time zone, the remaining columns the unchanged measurements of the sensors.

Remarks Concerning the time zone, we have to pay attention in which time zone the input dataset is defined. Furthermore, the MET time is also known as *Central European Time* "CET" and is the same as the UTC time with a time lag of one hour (MET=UTC+01:00). For our sensors data set, the time is defined in the time zone UTC. If no time zone is specified in the `createTimeSensors()`, the default "UTC" is considered.

The following Figure 2.5 represents the header of the output dataset after applying the `createTimeSensors()` function.



time	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS2444.kOhm.
29.08.2016 13:28:57	17,58	36,53	24,18	16,02	47,22	105,35
29.08.2016 13:29:57	17,6	36,59	24,29	16,15	47,58	105,35
29.08.2016 13:30:57	17,64	36,7	24,48	16,61	48,51	105,67
29.08.2016 13:31:07	17,65	36,77	24,51	16,69	48,65	106
29.08.2016 13:31:17	17,64	36,81	24,54	16,76	48,75	106,33
29.08.2016 13:31:27	17,65	36,79	24,57	16,82	48,85	107

Figure 2.5: Data set including date-time objects

2.2.4 Conductance

Up to present, our sensors data set contains resistance measurements. In Chapter 1, it is explained that the resistance decreases when a pollutant is present, that is to say the chemical concentration of the pollutant increases. So, resistance and concentration change in opposite way. To evade this, we can transform the sensors measurements into conductance, which is the inverse of resistance. On the following Figure 2.6, the resistance measurements of one sensor are represented on the left, the conductance measurements on the right.

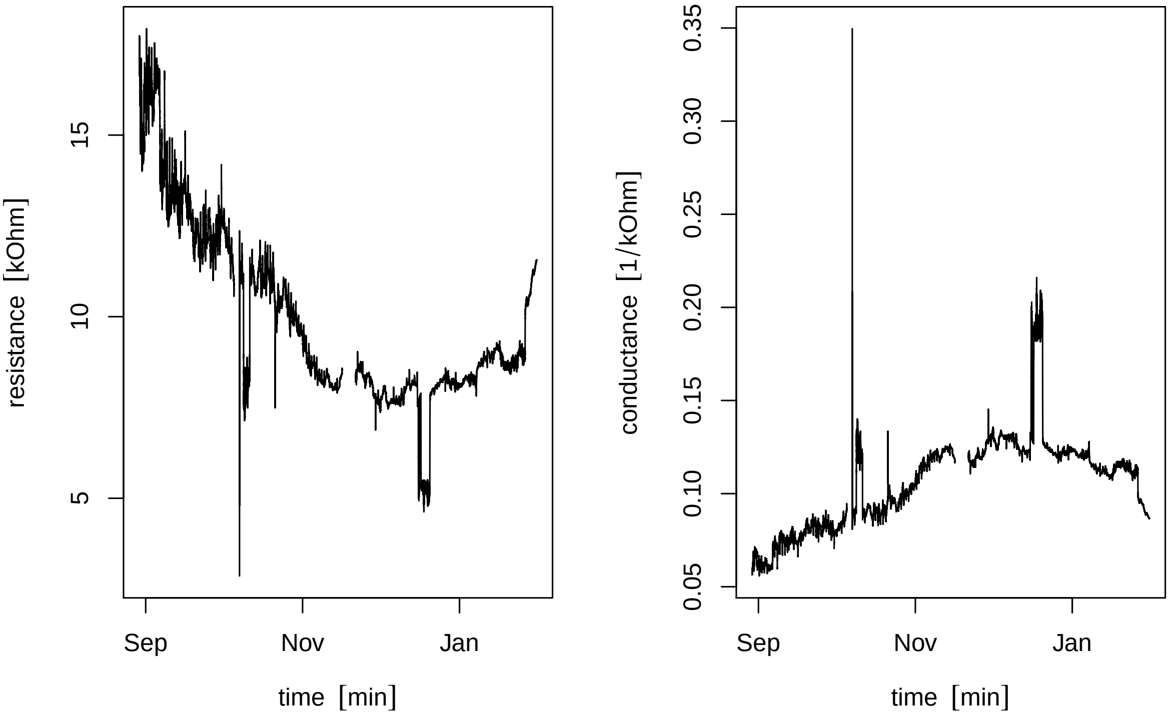


Figure 2.6: Resistance - Conductance of the TGS2602 sensor

The following function will compute this transformation, which will make the subsequent interpretation clearer:

```
conductance(data,columns)
```

Description The function `conductance()` transforms a data set by computing the inverse of the elements in the specified columns in the argument `columns`.

Arguments

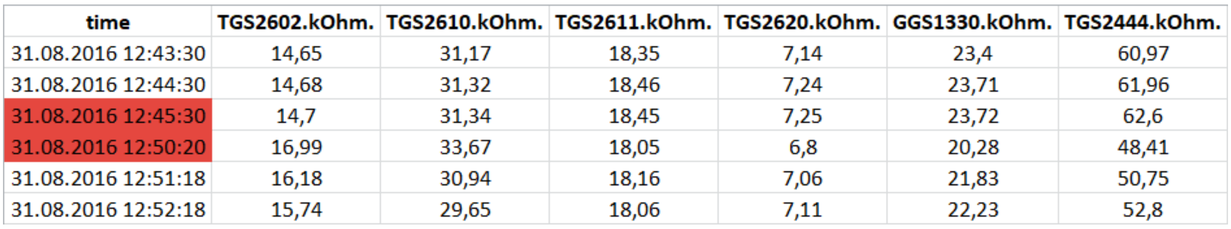
- `data` is a data set containing values to inverse.
- `columns` is a vector containing the column names of the data set in quotation marks which have to be inverted.

Value This function returns a new dataset `newData` containing the inverse values of the elements in the columns specified by `columns` in the dataset `data`.

Remark For our sensors dataset, we will apply the `conductance()` function to all columns in the dataset except the first one. This column contains the time values, which we do not want to be inverted.

2.2.5 Missing values

We have already seen that the measurement frequency of the sensors is one minute. When a value is missing in the data set, the respective measuring time is not written in the data set, thus the whole row in the data set is missing and a time difference of more than one minute appears. In the following Figure 2.7 a part of the data set with missing rows (indicated by the red cases) is illustrated.



time	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS2444.kOhm.
31.08.2016 12:43:30	14,65	31,17	18,35	7,14	23,4	60,97
31.08.2016 12:44:30	14,68	31,32	18,46	7,24	23,71	61,96
31.08.2016 12:45:30	14,7	31,34	18,45	7,25	23,72	62,6
31.08.2016 12:50:20	16,99	33,67	18,05	6,8	20,28	48,41
31.08.2016 12:51:18	16,18	30,94	18,16	7,06	21,83	50,75
31.08.2016 12:52:18	15,74	29,65	18,06	7,11	22,23	52,8

Figure 2.7: Part of data set with missing time values

To complete the data set by the missing rows, the function `addRowsNA` has been written:

```
addRowsNA(data)
```

Description The function `addRowsNA()` takes as argument the database to handle, calculates the time differences between each consecutive couple of rows, and if necessary, inserts a row filled with NA and the corresponding time value. The output is a new data set which corresponds to the input data set where the NA rows have been added for missing values.

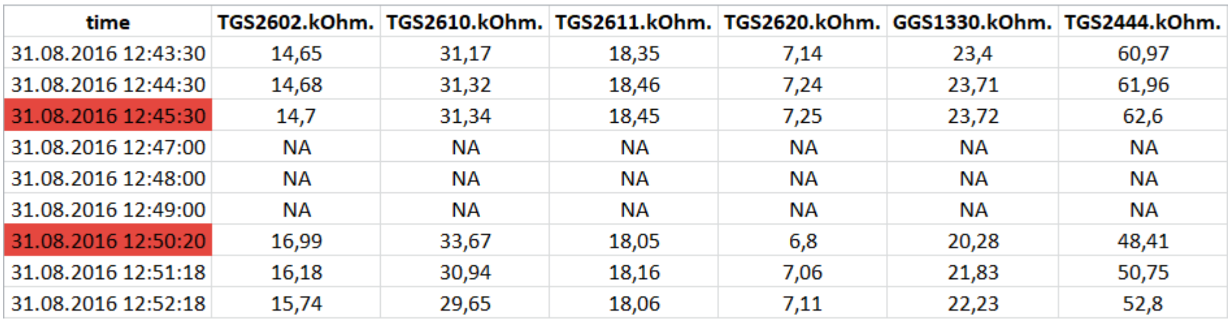
Arguments

- `data` is a data frame containing in the first column date-time objects.

Value The function `addRowsNA()` outputs a new dataset `newData`, which is the input data set where rows have been added when a sensor measurement is missing.

Remarks If each time difference calculated on the first column in the dataset `data` is already less or equal to one minute, the new dataset `newData` corresponds to the original dataset `data`. This is the case when there are no missing values.

Now, the new data set includes NA values when a sensor measurement is missing like shown in the subsequent Figure 2.8.



time	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS2444.kOhm.
31.08.2016 12:43:30	14,65	31,17	18,35	7,14	23,4	60,97
31.08.2016 12:44:30	14,68	31,32	18,46	7,24	23,71	61,96
31.08.2016 12:45:30	14,7	31,34	18,45	7,25	23,72	62,6
31.08.2016 12:47:00	NA	NA	NA	NA	NA	NA
31.08.2016 12:48:00	NA	NA	NA	NA	NA	NA
31.08.2016 12:49:00	NA	NA	NA	NA	NA	NA
31.08.2016 12:50:20	16,99	33,67	18,05	6,8	20,28	48,41
31.08.2016 12:51:18	16,18	30,94	18,16	7,06	21,83	50,75
31.08.2016 12:52:18	15,74	29,65	18,06	7,11	22,23	52,8

Figure 2.8: Part of data set with added NA values

The preprocessing for the sensors data set is finished here. We will now describe the preprocessing of the analysers data.

2.3 Analysers data file

Analysers file formatting As for the sensors data sets, several characteristics and formats have to been followed for a correct preprocessing:

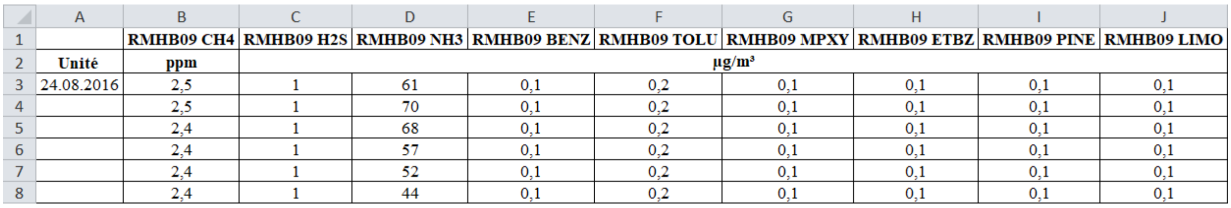
- The date values are stored in the format day.month.year, the separator being arbitrary.
- The time values are implemented in the preprocessing in the format hour:minutes:seconds, because the original data set does not include exact measuring times.
- After some manipulation in Excel, the data file must be in .csv format.
- The analysers data set presents two heading rows of the analysers data set, that will by merged to one header.

Table 2.2 provides general information about the analysers data file before pretreating it.

Table 2.2: Table of analysers data file

Content	one file
Format	excel file (.xlsx)
Period	24.08.2016 00:00:00 - 30.01.2017 23:30:00
Frequency	per 30 minutes
Number of columns/analysers measurements	10

The available analysers data starts the 24.08.2016 at midnight (00:00:00) and stopped the 30.01.2017 at 23:30:00. The analysers provided measurements for each half hour, which are stored in one excel file. The header of this data file is represented in Figure 2.9.



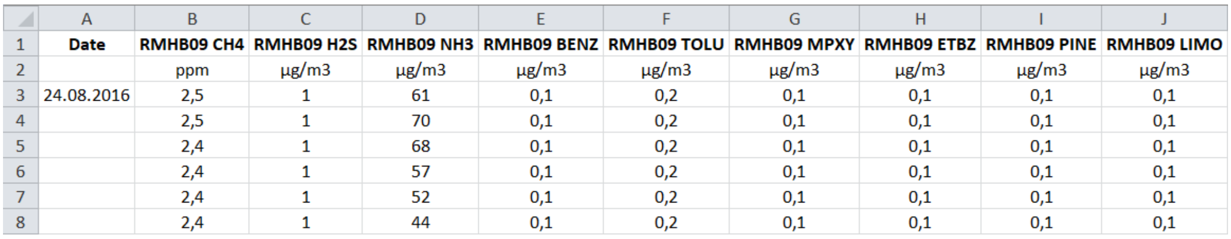
	A	B	C	D	E	F	G	H	I	J
1		RMHB09 CH4	RMHB09 H2S	RMHB09 NH3	RMHB09 BENZ	RMHB09 TOLU	RMHB09 MPXY	RMHB09 ETBZ	RMHB09 PINE	RMHB09 LIMO
2	Unité	ppm	µg/m³							
3	24.08.2016	2,5	1	61	0,1	0,2	0,1	0,1	0,1	0,1
4		2,5	1	70	0,1	0,2	0,1	0,1	0,1	0,1
5		2,4	1	68	0,1	0,2	0,1	0,1	0,1	0,1
6		2,4	1	57	0,1	0,2	0,1	0,1	0,1	0,1
7		2,4	1	52	0,1	0,2	0,1	0,1	0,1	0,1
8		2,4	1	44	0,1	0,2	0,1	0,1	0,1	0,1

Figure 2.9: Header of the analysers data file in Excel

The preprocessing concerning the analysers data starts with several manipulations in Excel:

- In the first excel cell `A1`, we add "Date".
- In the cell `A2`, we delete "Unité" so that this cell is empty.
- The cells `C2-J2` are merged to one cell and contain the unit "$\mu g/m^3$". We cancel the merge of these cells and repeat the concerned unit for each single cell.
- We save this modified data file with the CSV format at a specified location, e.g.: "C:/Users/user/analysers_file.csv".

The next Figure 2.10 illustrates this new data set "analysers_file.csv".



	A	B	C	D	E	F	G	H	I	J
1	Date	RMHB09 CH4	RMHB09 H2S	RMHB09 NH3	RMHB09 BENZ	RMHB09 TOLU	RMHB09 MPXY	RMHB09 ETBZ	RMHB09 PINE	RMHB09 LIMO
2		ppm	µg/m3	µg/m3	µg/m3	µg/m3	µg/m3	µg/m3	µg/m3	µg/m3
3	24.08.2016	2,5	1	61	0,1	0,2	0,1	0,1	0,1	0,1
4		2,5	1	70	0,1	0,2	0,1	0,1	0,1	0,1
5		2,4	1	68	0,1	0,2	0,1	0,1	0,1	0,1
6		2,4	1	57	0,1	0,2	0,1	0,1	0,1	0,1
7		2,4	1	52	0,1	0,2	0,1	0,1	0,1	0,1
8		2,4	1	44	0,1	0,2	0,1	0,1	0,1	0,1

Figure 2.10: Header of the modified analysers data file in Excel

The subsequent preprocessing of the analysers data set will be executed in R.

2.3.1 Read and save the data file

We can observe on the previous Figure 2.10 that there are two heading rows in the data set. Therefore, the following function `readAndSave()` has been created:

```
readAndSave(file_path)
```

Description The function `readAndSave()` reads the input data set whose file path is `file_path`, concatenates the two first lines of this data set and stores the output data set.

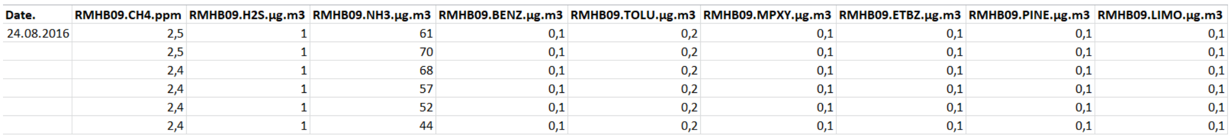
Arguments

- `file_path` is a string enclosed in quotation marks containing the file path of the analysers data set, e.g.: "C:/Users/user/analysers_file.csv".

Value The function `readAndSave()` creates a new data set `data_ana` containing the concatenation of the two first rows of the input data set as first row, and then the remaining values of this data set.

Remarks We concatenated the two first rows of the data set to keep the units of every analyser.

The header of the output data set is represented beneath (see Figure 2.11).



Date.	RMHB09.CH4.ppm	RMHB09.H2S.µg.m3	RMHB09.NH3.µg.m3	RMHB09.BENZ.µg.m3	RMHB09.TOLU.µg.m3	RMHB09.MPXY.µg.m3	RMHB09.ETBZ.µg.m3	RMHB09.PINE.µg.m3	RMHB09.LIMO.µg.m3
24.08.2016	2,5	1	61	0,1	0,2	0,1	0,1	0,1	0,1
	2,5	1	70	0,1	0,2	0,1	0,1	0,1	0,1
	2,4	1	68	0,1	0,2	0,1	0,1	0,1	0,1
	2,4	1	57	0,1	0,2	0,1	0,1	0,1	0,1
	2,4	1	52	0,1	0,2	0,1	0,1	0,1	0,1
	2,4	1	44	0,1	0,2	0,1	0,1	0,1	0,1

Figure 2.11: Header of `data_ana`

2.3.2 Elimination of useless data

Like in the preprocessing of the sensors data, we want to select only the columns of the studied analysers data and to delete those which are not useful. Therefore, we will use the same function as explained in the Section 2.2.2 to obtain a new data set. We want to delete the columns specifying the measurements of the three unusable analysers MPXY, ETBZ and PINE (see Appendix B.2).

2.3.3 Creation of date-time object

On the last Figure 2.11 we observe that the first column consists of date values in the form day.month.year followed by empty cells. We have to see this data set in blocks. Every block corresponds to one day, the first column contains first the date of the concerned day and then 47 empty cells are following for the remaining half hours of this day. The function `createTimeAnalysers()` will transform the date values into date-time objects so that afterwards, we have date-time values for every 30 minutes:

```
createTimeAnalysers(data,time_zone,date_column)
```

Description This function takes as argument the input dataset `data`, fragments this input data in blocks, one block corresponding to one day. The date of the current block will be repeated 48 times (because a day consists of 48 half hours). Then, an output dataset will be created, whose first column named `time` is a concatenation of the repeated dates and the half hours from 00:00:00 up to 23:30:00 for every block. These date-time values will than be parsed into date-time objects like got the sensors time measurements. The remaining columns contain the analysers measurements.

Arguments

- **data** is a data frame containing date values for each 48 cells, with intermediate empty cells in the first column and other values (e.g. analysers measurements) in the remaining columns.

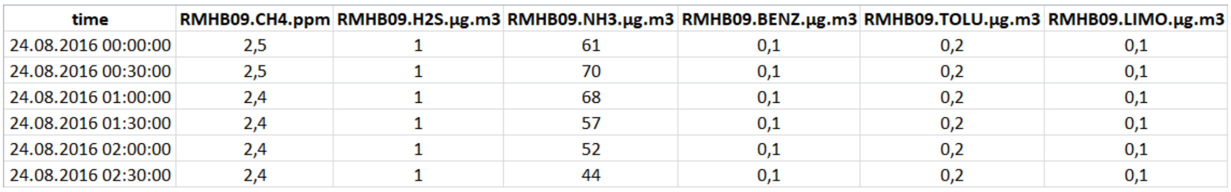
- **time_zone** is a quoted character string containing the time zone used for the time values, e.g: "UTC", which is also the default time zone.

- **date_column** is a string in quotation marks specifying the column name of the column containing the date values.

Value The function `createTimeAnalysers()` returns an output dataset `newData` whose first column contains date-time objects of the analysers measuring times and the remaining columns the respective measurements.

Remarks This function checks simultaneously if every block contains actually 48 rows, otherwise there are missing or too many measurements. If a block with more or less than 48 rows appears, a warning message will be written and the function stops executing. An `NULL` object will be returned.

The new data set has the following heading (see Figure 2.12).



time	RMHB09.CH4.ppm	RMHB09.H2S.µg.m3	RMHB09.NH3.µg.m3	RMHB09.BENZ.µg.m3	RMHB09.TOLU.µg.m3	RMHB09.LIMO.µg.m3
24.08.2016 00:00:00	2,5	1	61	0,1	0,2	0,1
24.08.2016 00:30:00	2,5	1	70	0,1	0,2	0,1
24.08.2016 01:00:00	2,4	1	68	0,1	0,2	0,1
24.08.2016 01:30:00	2,4	1	57	0,1	0,2	0,1
24.08.2016 02:00:00	2,4	1	52	0,1	0,2	0,1
24.08.2016 02:30:00	2,4	1	44	0,1	0,2	0,1

Figure 2.12: Head of `newData`

2.4 Fusion of the sensors and analysers data sets

After having finished the preprocessing of the sensors and analysers data sets separately, we can finally achieve the fusion of the two obtained data sets in this section, which is our intermediate objective.

Output file formatting The final output data file presents several characteristics in terms of the format:

- The date-time objects are written in the format year-month-day hour:minutes:seconds, for example: "2016-10-07 13:50:30" in the first column.
- The data file contains first the columns of the approximated data by interpolation, then the exact data. In our study thus, the approximated analysers data is followed by the exact sensors values.
- The column names can be chosen by the user for an agreeable use and readability afterwards.

2.4.1 Linear interpolation

To proceed on the fusion of the sensors and analysers data files, we should have the measurements from both instruments for the same time series. However, the analysers have a measuring frequency of half an hour and the sensors one of a minute. Furthermore, they did not start measurements at the same time. The sensors started to work on the 29.08.2017 at 13:28:57, but the analysers on the 24.08.2016 at 00:00:00. If we want to keep the exact measuring times of the sensors, we take as sensors values the measured values for the abscissa *time*. The analysers values will be the estimated values at the abscissa time after linear interpolation of the data in the window $[time - 30min; time + 30min]$, and NA otherwise. The following function, called `approxAnalysers()` will perform this approximation for the analysers:

```
approxAnalysers(data_analysers,data_sensors,
   time_step_analysers,time_step_sensors)
```

Description The function `approxAnalysers()` executes a linear interpolation of the measurements with the greater time step by taking the time series of the data set with smaller time step and writes the interpolated values in a new data set.

Arguments

- `data_analysers` is a data frame containing a first column `time` with date-time objects and measurements in the remaining columns, e.g. the data set found at the end of the analysers data set preprocessing.

- `data_sensors` is a data frame with the same specifications but with another time step in the first column. We take the data set obtained after preprocessing of the sensors data.

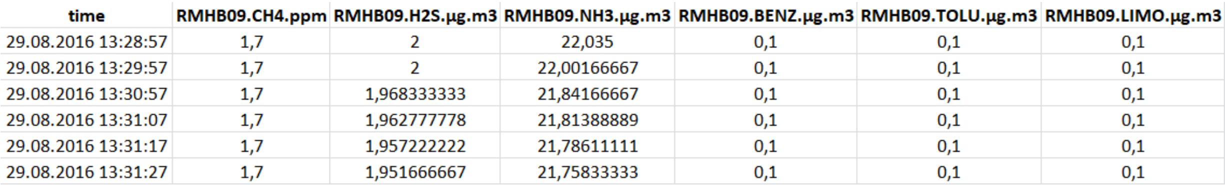
- `time_step_analysers` is an integer indicating which time difference (in minutes) is defined in the data set `data_analysers`.

- `time_step_sensors` is an integer indicating which time difference (in minutes) is defined in the data set `data_sensors`.

Value The function `approxAnalysers()` compares the time steps of both data sets in the input, then performs a linear interpolation on the data set with the larger time step and returns an output data set named `data_approx` which contains these interpolated measurements in terms of the smaller time steps.

Remarks The time series used to perform the interpolation, thus the smaller time steps defines the time period when the approximation takes place. If the time series to approximate exceeds the time period for the interpolation (before and/or after), these values will simply not be taken in the new data set. In the opposite way, if it falls short of the interpolation time period, the approximated data set is filled with NA values up to the moment where the two input data sets simultaneously contain measurements.

The head of the new data set `data_approx` is shown in the following Figure 2.13.



time	RMHB09.CH4.ppm	RMHB09.H2S.µg.m3	RMHB09.NH3.µg.m3	RMHB09.BENZ.µg.m3	RMHB09.TOLU.µg.m3	RMHB09.LIMO.µg.m3
29.08.2016 13:28:57	1,7	2	22,035	0,1	0,1	0,1
29.08.2016 13:29:57	1,7	2	22,00166667	0,1	0,1	0,1
29.08.2016 13:30:57	1,7	1,968333333	21,84166667	0,1	0,1	0,1
29.08.2016 13:31:07	1,7	1,962777778	21,81388889	0,1	0,1	0,1
29.08.2016 13:31:17	1,7	1,957222222	21,78611111	0,1	0,1	0,1
29.08.2016 13:31:27	1,7	1,951666667	21,75833333	0,1	0,1	0,1

Figure 2.13: Head of approximated analysers data set

2.4.2 Fusion of data

Finally, the function `dataFusion()` will proceed the fusion of the sensors data set with the analysers data set.

```
dataFusion(data_approximated,data_exact,colNames)
```

Description This function merges the data set containing the exact measured values with the approximated data set, created by the previous function `approxAnalysers()`.

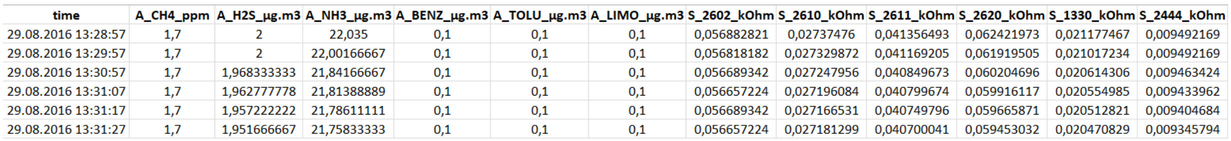
Arguments

- `data_approximated` is a data frame containing the approximated values by interpolation with first column `time`.
- `data_exact` is a data frame containing the real measured values with first column `time`.
- `colNames` is a character string containing the chosen column names for the output data.

Value The function `dataFusion()` returns the output data set `data_fusion` whose first column contains the date-time objects given by the first data set [1], the remaining columns contain the interpolated measurements from `data_approximated`, and then the exact measurements.

Remark The column names of the output data set can be chosen for the convenience and clearness for the subsequent work.

In the next Figure 2.14, we can see the header of the merged data set.



time	A_CH4_ppm	A_H2S_µg.m3	A_NH3_µg.m3	A_BENZ_µg.m3	A_TOLU_µg.m3	A_LIMO_µg.m3	S_2602_kOhm	S_2610_kOhm	S_2611_kOhm	S_2620_kOhm	S_1330_kOhm	S_2444_kOhm
29.08.2016 13:28:57	1,7	2	22,035	0,1	0,1	0,1	0,056882821	0,02737476	0,041356493	0,062421973	0,021177467	0,009492169
29.08.2016 13:29:57	1,7	2	22,00166667	0,1	0,1	0,1	0,056818182	0,027329872	0,041169205	0,061919505	0,021017234	0,009492169
29.08.2016 13:30:57	1,7	1,968333333	21,84166667	0,1	0,1	0,1	0,056689342	0,027247956	0,040849673	0,060204696	0,020614306	0,009463424
29.08.2016 13:31:07	1,7	1,962777778	21,81388889	0,1	0,1	0,1	0,056657224	0,027196084	0,040799674	0,059916117	0,020554985	0,009433962
29.08.2016 13:31:17	1,7	1,957222222	21,78611111	0,1	0,1	0,1	0,056689342	0,027166531	0,040749796	0,059665871	0,020512821	0,009404684
29.08.2016 13:31:27	1,7	1,951666667	21,75833333	0,1	0,1	0,1	0,056657224	0,027181299	0,040700041	0,059453032	0,020470829	0,009345794

Figure 2.14: Head of `data_fusion`

2.5 Preprocessing on the merged data set

Some manipulations must be realised on the fusion data set before we can use it in the model prediction.

[1] By the use of the `approxAnalysers()` function, the two time columns are exactly the same.

2.5.1 Time reduction

Up to now, the time frequency of our data set is one minute. With this time step, the measurements are very likely to be dependent with the time. Therefore we want to increase the time difference to at least 30 minutes to avoid this time dependence. The following function named `timeReduction()` is used:

```
timeReduction(data,step)
```

Description This function creates a dataset containing data with a greater time lag than the input dataset `data`. The number of minutes in the argument `step` is added to the first time value in the data set, and the smallest time value greater than this sum is defined as the next measuring time. This process continues until the end of the data set. This procedure ensures time differences of at least the number of minutes in `step` and therefore, the independence of the measurements in time.

Arguments

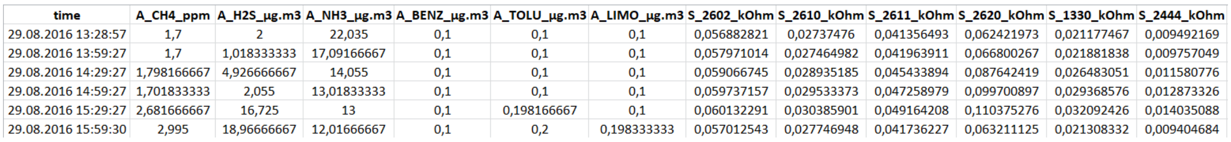
- `data` is a data frame containing the measurements of analysers and sensors and the first column defines the time series.

- `step` is a strictly positive integer specifying the desired number of minutes between each measurement (row).

Value The function `timeReduction()` returns the output dataset `smallData` whose first column contains the date-time objects whose differences are given by the argument `step`. All measurements between these minutes are deleted.

Remark When the sensors started measuring, there were sometimes 10 seconds time steps, so less than one minute between each measure. With this function, all measurements with a step smaller than it should be, will be eliminated.

In the next Figure 2.15, we can see the header of the data set after time reduction.



time	A_CH4_ppm	A_H2S_µg.m3	A_NH3_µg.m3	A_BENZ_µg.m3	A_TOLU_µg.m3	A_LIMO_µg.m3	S_2602_kOhm	S_2610_kOhm	S_2611_kOhm	S_2620_kOhm	S_1330_kOhm	S_2444_kOhm
29.08.2016 13:28:57	1,7	2	22,035	0,1	0,1	0,1	0,056882821	0,02737476	0,041356493	0,062421973	0,021177467	0,009492169
29.08.2016 13:59:27	1,7	1,018333333	17,09166667	0,1	0,1	0,1	0,057971014	0,027464982	0,041963911	0,066800267	0,021881838	0,009757049
29.08.2016 14:29:27	1,798166667	4,926666667	14,055	0,1	0,1	0,1	0,059066745	0,028935185	0,045433894	0,087642419	0,026483051	0,011580776
29.08.2016 14:59:27	1,701833333	2,055	13,01833333	0,1	0,1	0,1	0,059737157	0,029533373	0,047258979	0,099700897	0,029368576	0,012873326
29.08.2016 15:29:27	2,681666667	16,725	13	0,1	0,198166667	0,1	0,060132291	0,030385901	0,049164208	0,110375276	0,032092426	0,014035088
29.08.2016 15:59:30	2,995	18,96666667	12,01666667	0,1	0,2	0,198333333	0,057012543	0,027746948	0,041736227	0,063211125	0,021308332	0,009404684

Figure 2.15: Head of `small_Data`

2.5.2 Delete observations

For several days, there were odour bags connected to the instruments. We have to delete the rows containing the concerned measurements, otherwise we could misinterpret fluctuations of the signals. The next Table 2.3 contains the days, where odour bags were connected.

Table 2.3: Days of odour bags

Odour bag connection
27.10.2016
3.11.2016
14.11.2016
16.112016
23.11.2016
24.11.2016
25.11.2016
29.11.2016
30.11.2016

The following function `deleteRows()` will perform the elimination of specified observations:

```
deleteRows(data,start_rows,end_rows)
```

Description The function takes as argument an input data set, compares the column `time` of this data set with the time intervals given by `start_rows` and `end_rows` and deletes the rows corresponding to time values included in one of the intervals.

Arguments

- `data` is a data frame whose first column contains a time series, the remaining columns contain analysers and sensors measurements.

- `start_rows` is a vector containing character strings in quotation marks specifying the starting time of the intervals to delete, e.g. ("2016-10-09 01:00:00" , "2016-10-27 00:00:00").

- `save_location` is a vector containing character strings in quotation marks specifying the ending time of the intervals to delete, e.g. ("2016-10-11 11:00:00" , "2016-10-27 23:59:59").

Value The function `deleteRows()` returns an output dataset `newData` containing only the rows of the input dataset, whose measuring time is not between two respective starting and ending times of intervals. In our example described in the arguments, we would delete all measurements taken between "2016-10-09 01:00:00" and "2016-10-11 11:00:00" as well as the measurements from "2016-10-27 00:00:00" to "2016-10-27 23:59:59".

Remarks If the number of starting times do not equal those of the ending times, a warning message will appear and the input data set will not change.

2.6 Control parameters preprocessing

We want to use the control parameters if special events are observed in the sensors and analysers sensing, to check every influence. Therefore, we have to pre-process these data sets as well. The next Figure 2.16 presents the summary of the preprocessing of the control parameters.

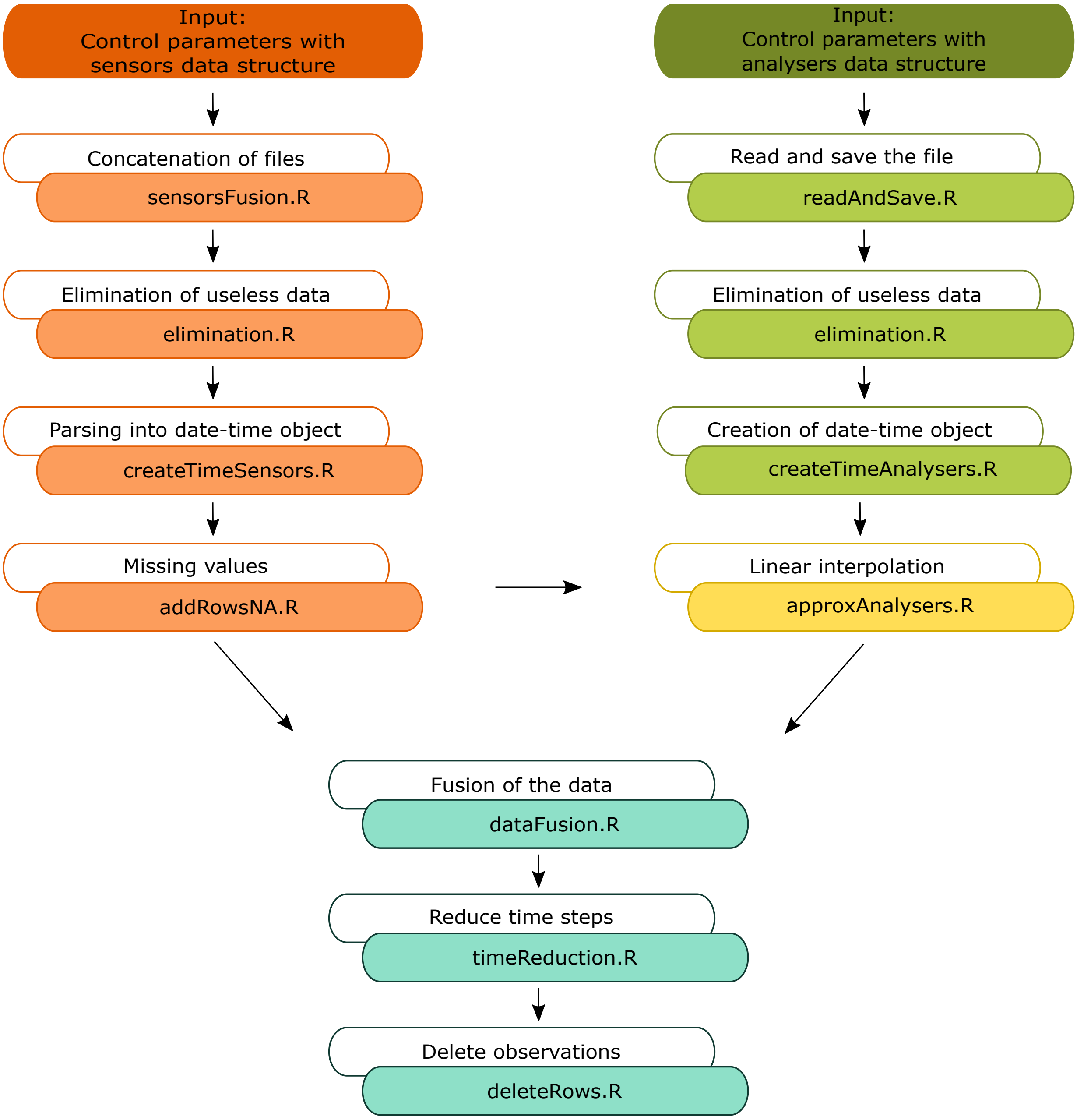


Figure 2.16: Summary of the control parameters preprocessing

We will look back on the table summarising the control parameters (see Table 2.4).

Table 2.4: Table of control parameters

Name	Return	Unit	Comments
Temp. Enc.	temperature around enclosure	°C	per minut
Temp. In.	temperature in enclosure	°C	per minut maintained at 50°C
Hr. Enc.	relative humidity in enclosure	%	per minut
Hr. In.	relative humidity around enclosure	%	per minut
RMHB09 DV	wind direction	°	per 30 minutes 0°corresponds to East
RMHB09 HR	relative humidity outside	%	per 30 minutes
RMHB 09 PA	atmospheric pressure	hPa	per 30 minutes
RMHB09 TT	outside temperature	°C	per 30 minutes
RMHB09 VV	wind velocity	m/s	per 30 minutes

The first four variables have their source in the same dataset as the sensors measurements. So, the preprocessing of these four variables has the same structure as the sensors data set and the same functions as explained in Section 2.2. The only modification is, that we do not need to apply the `conductance()` function, because it makes no sense to inverse temperature or humidity values.

The last five variables delivered by the meteorological station from the ISSeP are stored in a data set having the same structure as the original analysers data set. Thus, we can use the same functions as explained in Section 2.3.

Afterwards, exactly as for the sensors and analysers data set we can proceed a linear interpolation to receive the same time series before merging the two data sets. Then, we can also reduce the time step to take the same as for the fusion data set containing the sensors and analysers measurements and also delete the rows we want to exclude by the same reasons as before. Now, we can use the final control data set to check any influence when having results in further analyses.

2.7 Overview

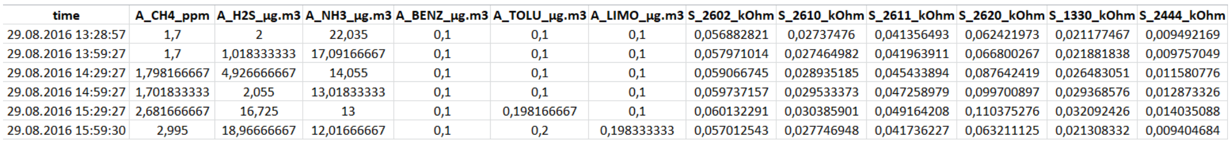
In general, this preprocessing can be reproduced for a new data set, with the same properties than the data sets in this study, e.g.: contain time values in one of the columns. When the preprocessing is terminated, we can begin the analysis of the final data set.

Chapter 3

Data description and subsequent pretreatment

In this chapter, we analyse the preprocessed data set to obtain a better understanding of the data. For every analyser and every sensor variable, we provide a data summary for all the variables separately in the data set, count the missing values, examine their ranges and observe atypical events.

The first lines of the preprocessed dataset are represented in Figure 3.1.



time	A_CH4_ppm	A_H2S_µg.m3	A_NH3_µg.m3	A_BENZ_µg.m3	A_TOLU_µg.m3	A_LIMO_µg.m3	S_2602_kOhm	S_2610_kOhm	S_2611_kOhm	S_2620_kOhm	S_1330_kOhm	S_2444_kOhm
29.08.2016 13:28:57	1,7	2	22,035	0,1	0,1	0,1	0,056882821	0,02737476	0,041356493	0,062421973	0,021177467	0,009492169
29.08.2016 13:59:27	1,7	1,018333333	17,09166667	0,1	0,1	0,1	0,057971014	0,027464982	0,041963911	0,066800267	0,021881838	0,009757049
29.08.2016 14:29:27	1,798166667	4,926666667	14,055	0,1	0,1	0,1	0,059066745	0,028935185	0,045433894	0,087642419	0,026483051	0,011580776
29.08.2016 14:59:27	1,701833333	2,055	13,01833333	0,1	0,1	0,1	0,059737157	0,029533373	0,047258979	0,099700897	0,029368576	0,012873326
29.08.2016 15:29:27	2,681666667	16,725	13	0,1	0,198166667	0,1	0,060132291	0,030385901	0,049164208	0,110375276	0,032092426	0,014035088
29.08.2016 15:59:30	2,995	18,96666667	12,01666667	0,1	0,2	0,198333333	0,057012543	0,027746948	0,041736227	0,063211125	0,021308332	0,009404684

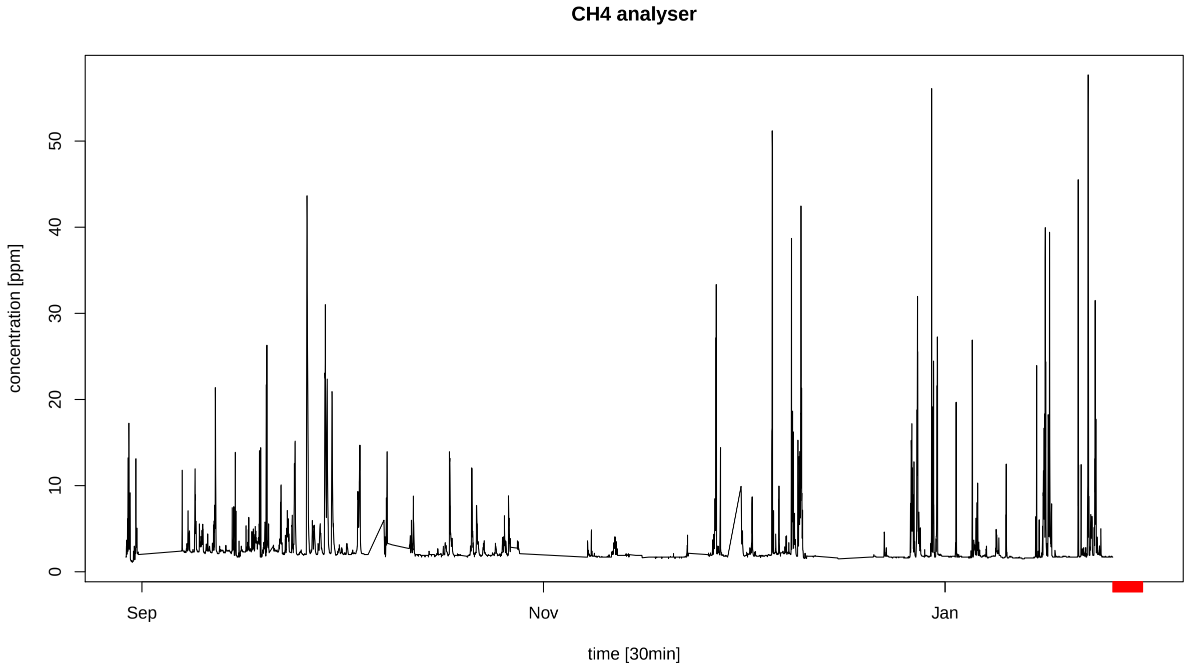
Figure 3.1: First values of the data set

We can see one time column, six analysers and six sensors measurements. Now, the analysis of each of the analysers and sensors can begin.

3.1 Analysers values

We analyse the measurements of every analyser separately to observe missing values, the measuring range, the distribution and atypical values. The missing values are marked by red ticks in the time series. Given that we will work with the analyser measurements taken to the logarithm, the transformed time series is also represented.

3.1.1 CH_4 Analyser



Min	Q_1	Median	Mean	Q_3	Max	Missing values
1.1	1.781	2	2.871	2.429	57.689	216

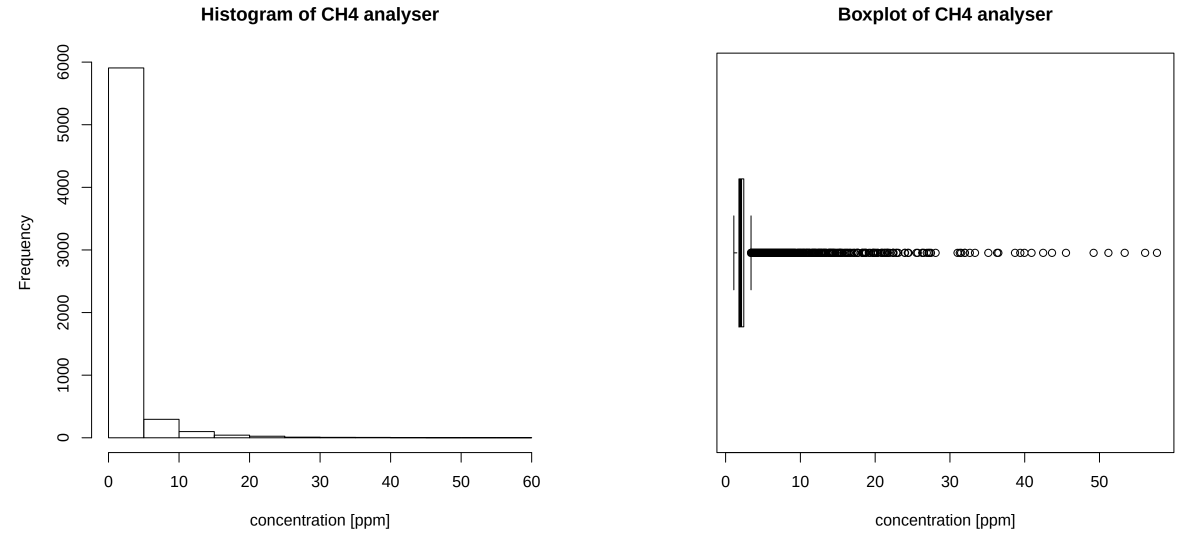


Figure 3.2: Descriptive statistics for CH_4 analyser

Distribution On closer examination of the histogram and boxplot, it is confirmed that the distribution of CH_4 values is very asymmetric, because of the huge amount of small values and the few high measurements of CH_4. Moreover, we can observe several extreme concentrations exceeding the right whisker of the boxplot. This indicates that events of high CH_4 concentration have been quite rare on the measurement site during the observation period.

Range On the time series and the table beneath, we observe the range of the CH_4 measurements going from 1.1 to 57.69 ppm. But there are only a few measurements reaching the high values. The third quartile is equal to 2.429, that is to say that 75% of the observed CH_4 concentrations are less or equal to this value. This predominant small range is confirmed by the histogram and boxplot and also by the mean equal to 2.871.

Missing values The statistic summary indicates 216 missing values occurring in the CH_4 variable. These CH_4 values are missing from "2017-01-26 11:39:11 UTC" to "2017-01-30 23:55:13 UTC" and define the last measurements of this analyser. A power failure is probably the reason for these missing measurements.

Logarithmic time series The following Figure 3.3 represents the CH_4 concentrations after logarithmic transformation.

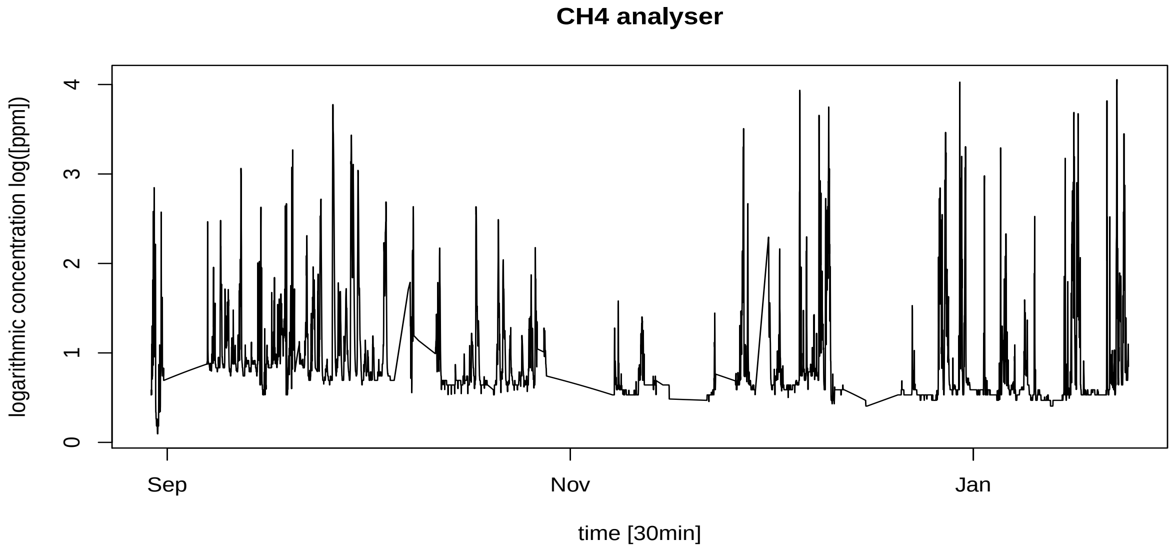
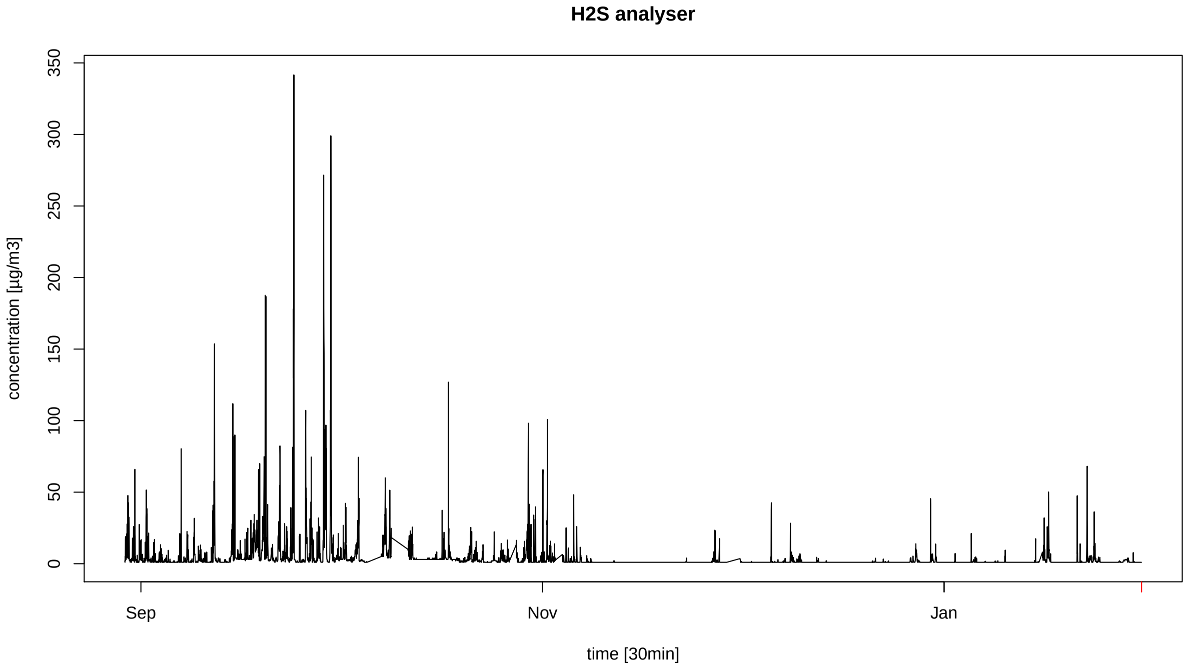
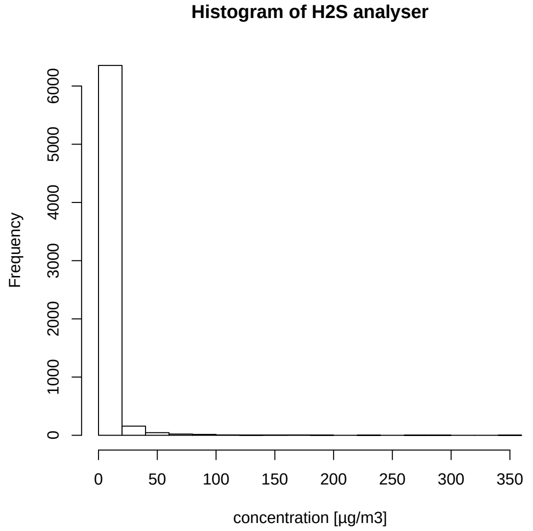


Figure 3.3: Logarithmic CH_4 concentration

3.1.2 H_2S Analyser



Min	Q_1	Median	Mean	Q_3	Max	Missing values
1	1	1	4.347	3	341.653	1



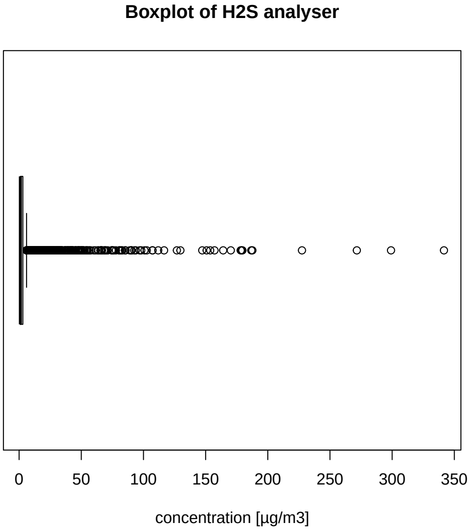


Figure 3.4: Descriptive statistics for H_2S analyser

Distribution We see much more fluctuation of the H_2S measurements from September to November and afterwards nearly no more signal except little elevations. The intense drop out from November has no known explicit reason and leads to a huge amount of small values. Moreover, it leads to a strong right-tailed distribution of the H_2S measurements as seen on the histogram and boxplot.

Range The H_2S analyser provided measurements from 1 to 341.653 $\mu g/m^3$. Like for the CH_4 analyser, most values are very small, three-quarter of them are less or equal to 3 $\mu g/m^3$. This high quantity of measurements around 1-3 $\mu g/m^3$ is also confirmed by the width of the box in the boxplot. Nevertheless, the range of 340.653 $\mu g/m^3$ is the widest over all the analysers.

Missing values The time series indicates only one missing value taking place the "2017-01-30" at 23:55:13 o'clock. At this time, the last measure of the sensors data set has been collected, but the analysers stopped measuring at approximately 23:30:00. Thus there is no value 30 minutes later and a missing value occurs in the interpolation.

Logarithmic time series The logarithm of the H_2S concentrations is shown on Figure 3.5.

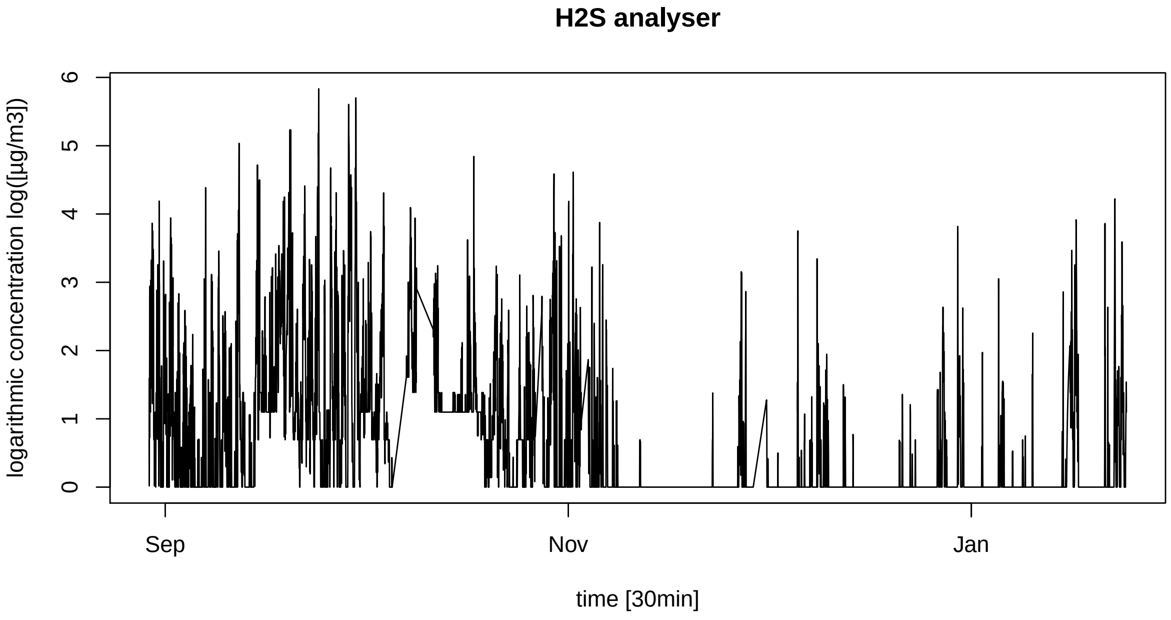
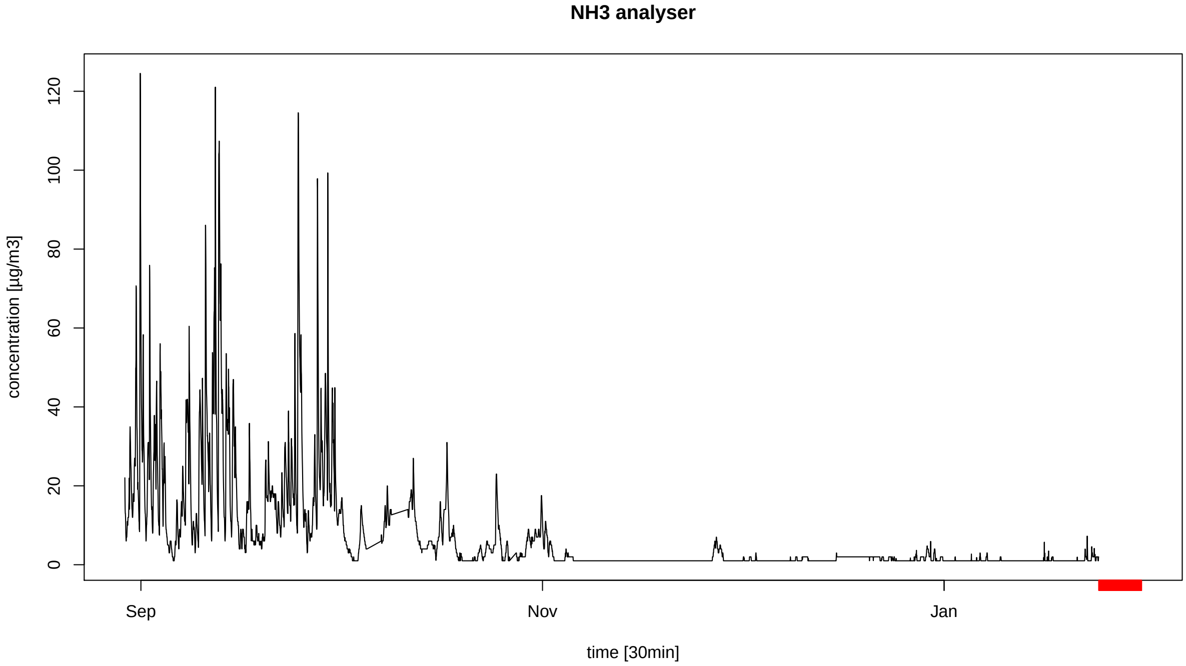


Figure 3.5: Logarithm of the H_2S concentration

3.1.3 NH_3 Analyser



Min	Q_1	Median	Mean	Q_3	Max	Missing values
1	1	2	7.625	8	124.54	311

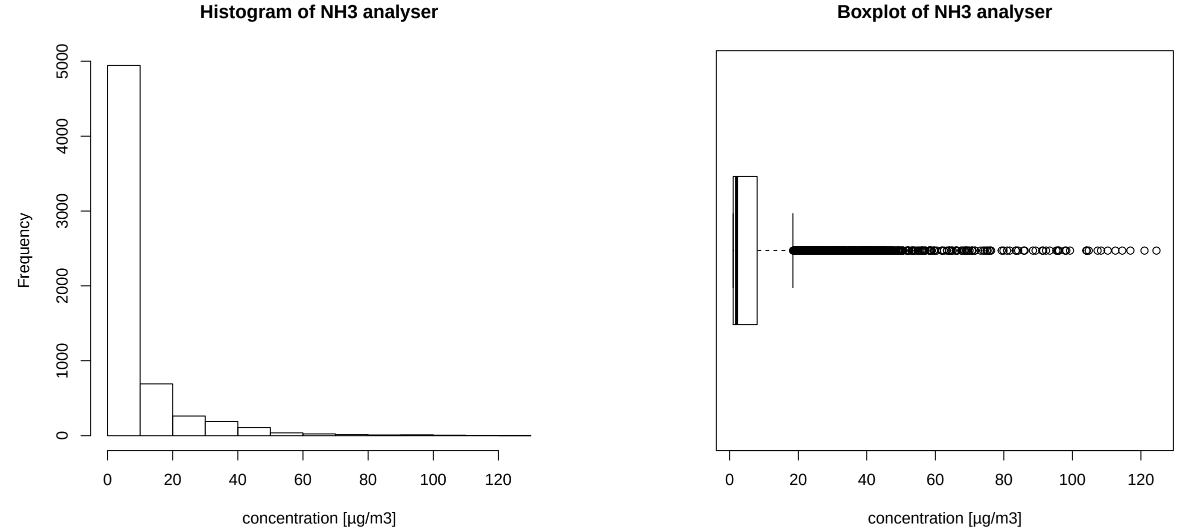


Figure 3.6: Descriptive statistics for NH_3 analyser

Distribution Like for the H_2S analyser, we can observe very oscillating signals in September, then the values decrease in October and from November there is nearly no more ammoniac measured. As before, we do not have any explication for this phenomena, but it provides an important number of weak ammoniac measurements. The histogram and boxplot show up a right-skewed distribution. Many observations reside outside the whiskers on the right side of the boxplot.

Range The NH_3 analyser collected measurements in a range from 1 to 124.54 $\mu g/m^3$. So, we have a smaller range than for H_2S. However, a more important number of observations remains high for this analyser. About 25% of the measurements reside between 2 and 8 $\mu g/m^3$.

Missing values We observe 311 missing values for the NH_3 analyser. These values are missing from "2017-01-24 11:50:02 UTC" to "2017-01-30 23:55:13 UTC", so nearly six days of measurements, most probably caused by a problem of the electric power supply for this analyser.

Logarithmic time series The following Figure 3.7 shows up the logarithmic transformation of the NH_3 concentration.

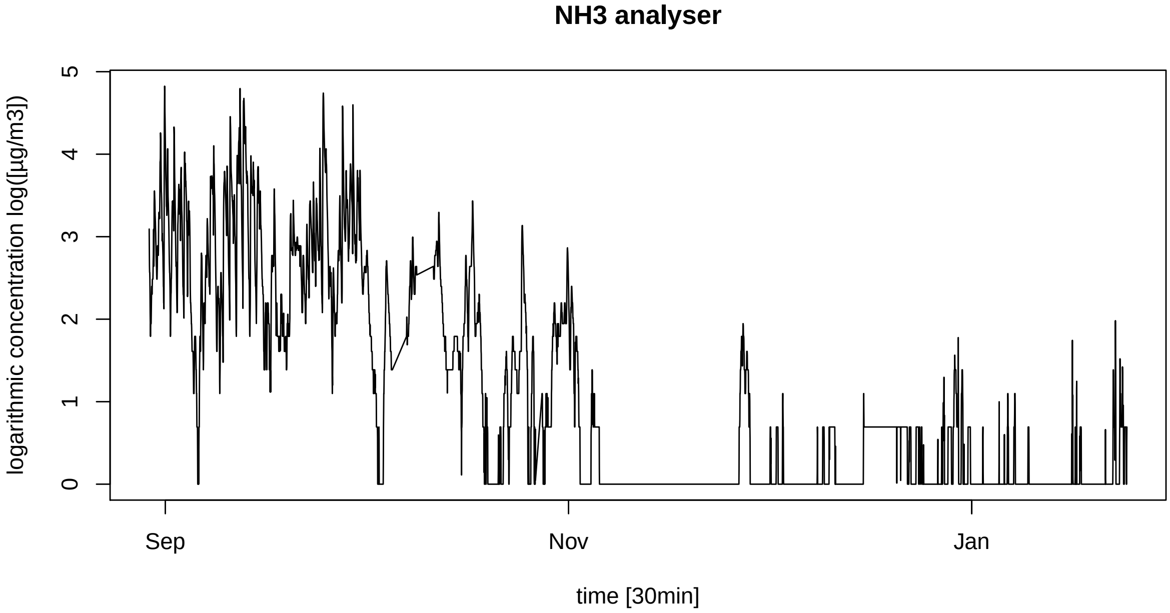
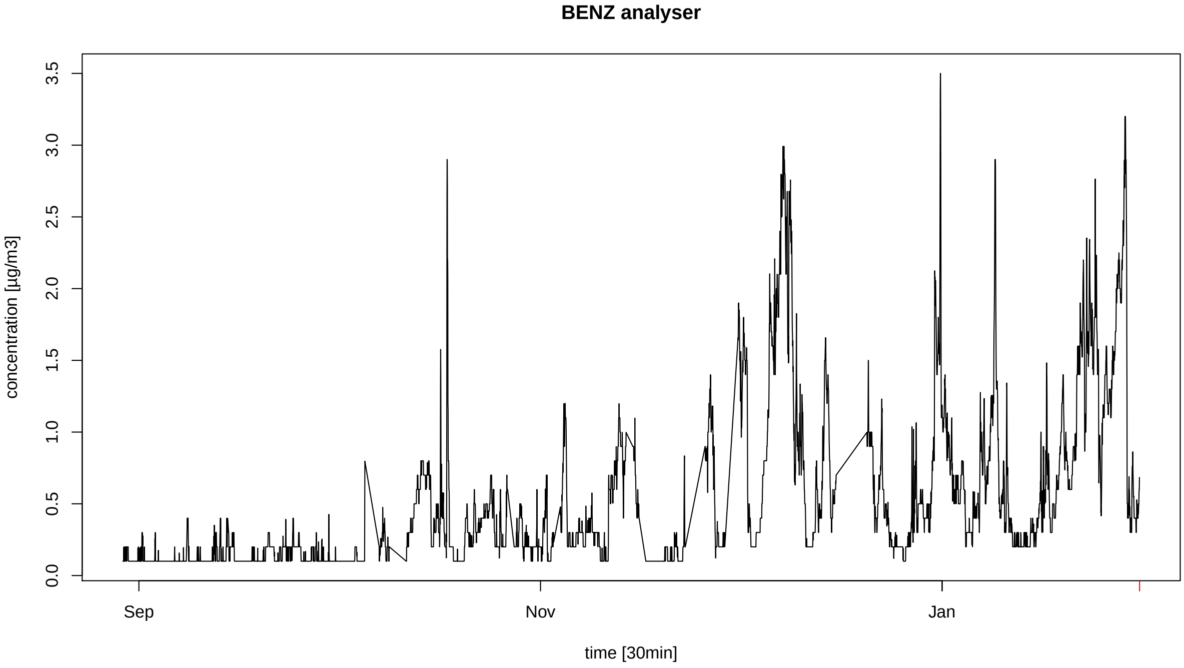


Figure 3.7: Logarithmic NH_3 concentration

3.1.4 BENZ Analyser



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.1	0.1012	0.3	0.5119	0.6882	3.5	1

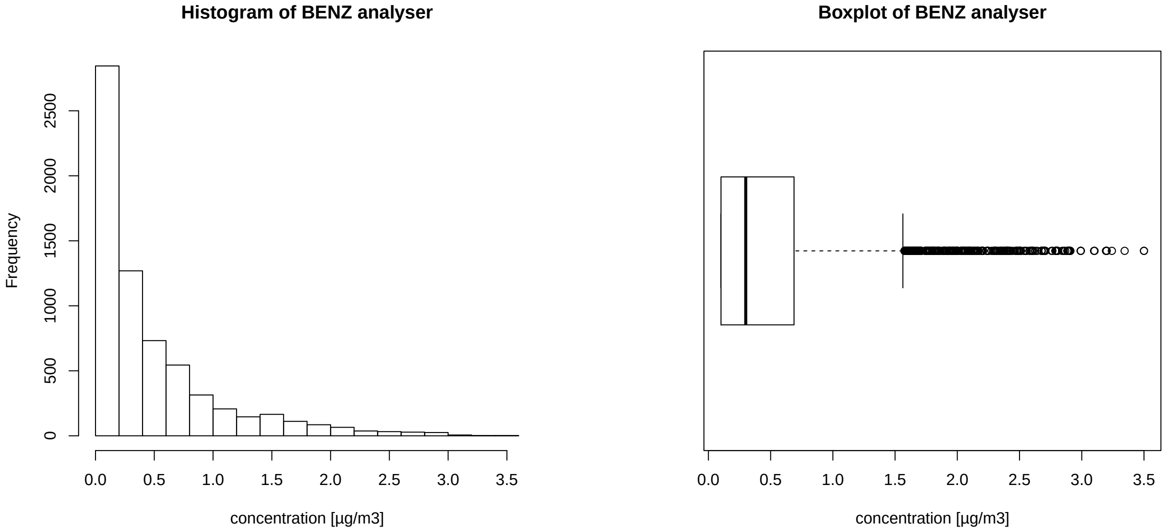


Figure 3.8: Descriptive statistics for BENZ analyser

Distribution The BENZ analyser generates the inverse phenomena of the H_2S and NH_3 analysers. Until mid-October the measurements of benzene remain low, then increase and provide more fluctuations. Here, the distribution is right-tailed as well and there are also several extreme values bigger than the right whisker in the boxplot.

Range These fluctuations of the benzene measurements are not so immense, because the analyser has a very small range from 0.1 to 3.5 $\mu g/m^3$ compared to the precedent analysers.

Missing values There is one missing value for the benzene measurements. Like for the H_2S analyser, it is the last measure at 23:55:13 universal time on 30/1/2017. As before, the missing value occurs when performing the linear interpolation of the analysers measurements between around 23:30:00 and 23:55:13 o'clock. The last measure of the BENZ analyser has been provided at 23:30:00, so the interpolation gives a NA value as output afterwards.

Logarithmic time series Figure 3.9 represents the logarithmic benzene measurements.

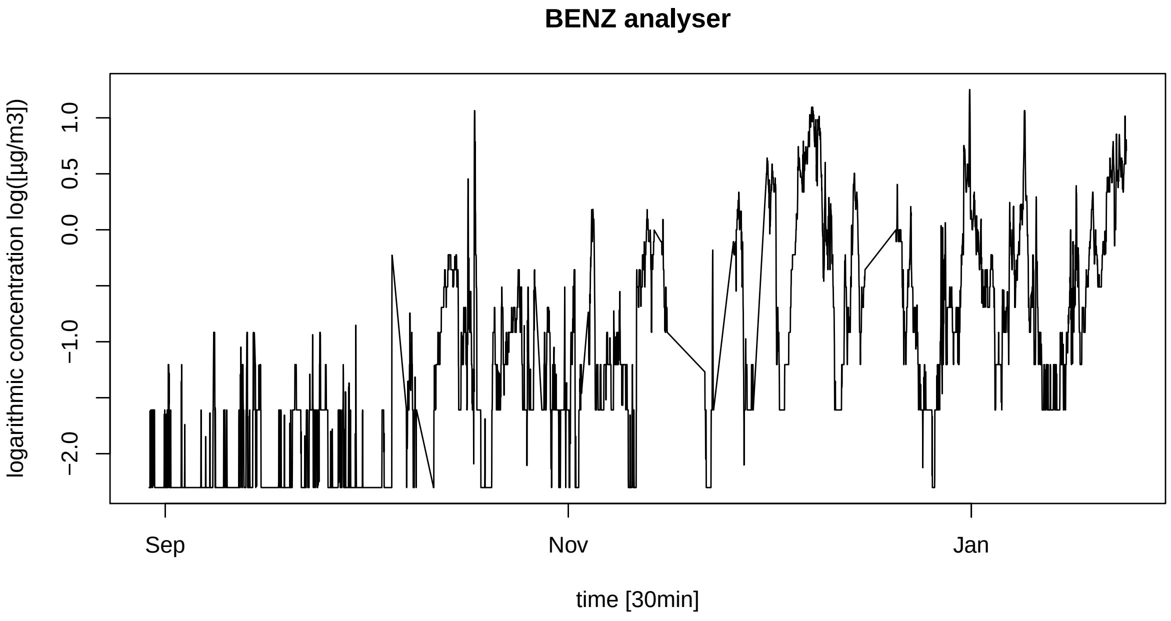
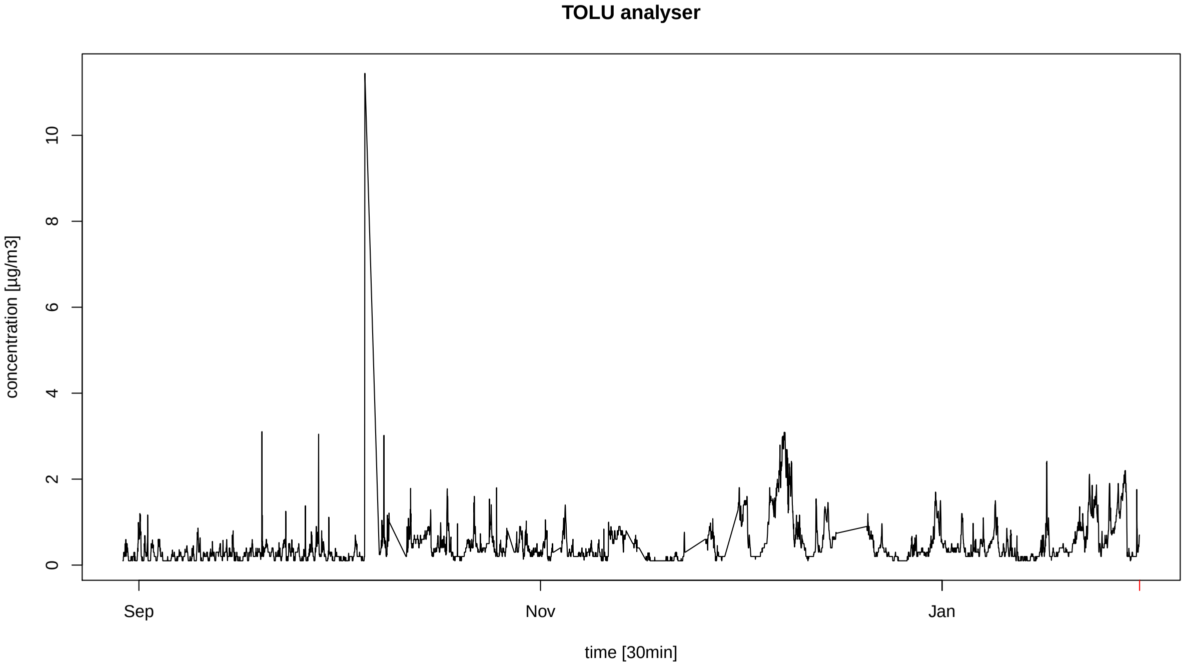


Figure 3.9: Logarithmic BENZ concentration

3.1.5 TOLU Analyser



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.1	0.2	0.3	0.5471	0.6	11.4423	1

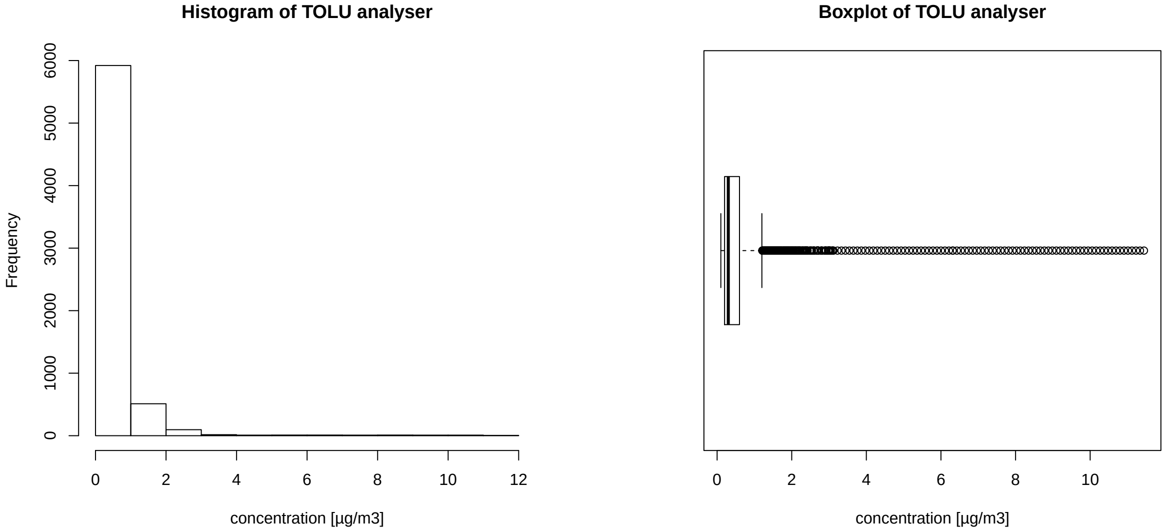


Figure 3.10: Descriptive statistics for TOLU analyser

Distribution Excepted a high peak on 5/10/2016, the values of the TOLU analyser remain stable between 0 and 3 approximately. Again, we can observe a right-skewed distribution of values, as shown on the histogram and boxplot.

Range The analyser dedicated to toluene provides a range from 0.1 to circa 11.44 $\mu g/m^3$. Furthermore, 75% of the TOLU measurements are between 0.1 and 0.6 $\mu g/m^3$.

Missing values There is one missing value present in the measurements of the TOLU analyser. As before, this single value provides from the interpolation between the last value of the analyser and the missing value 30 minutes later.

Logarithmic time series The following time series represents the TOLU measurements taken to the logarithm.

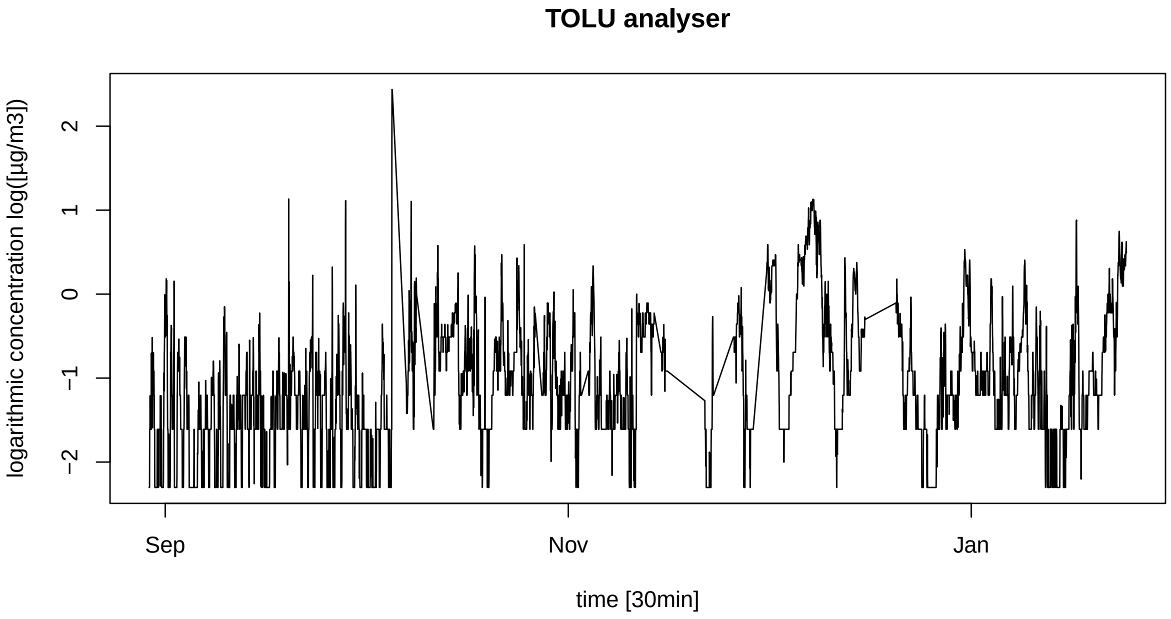
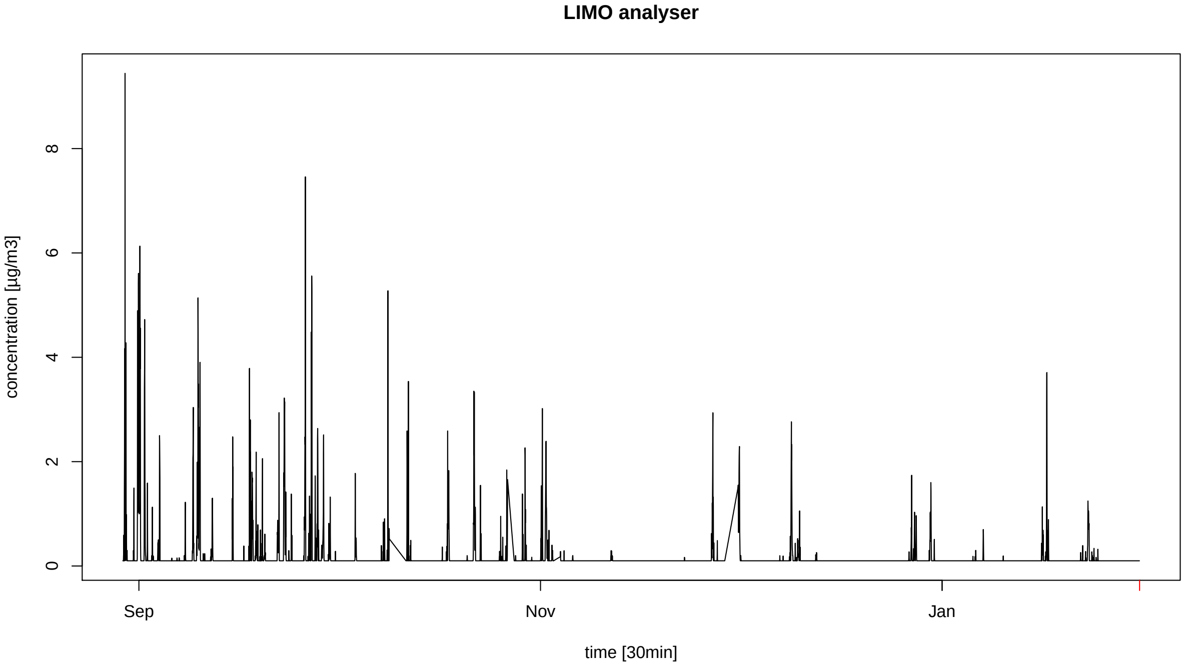


Figure 3.11: Logarithmic TOLU concentration

3.1.6 LIMO Analyser



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.1	0.1	0.1	0.1936	0.1	9.4417	1

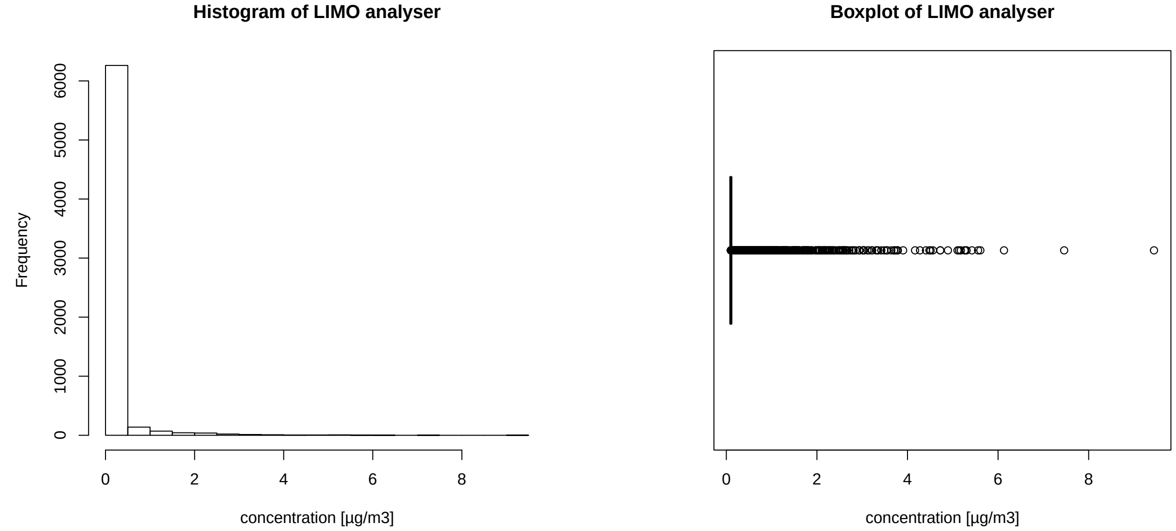


Figure 3.12: Descriptive statistics for LIMO analyser

Distribution We observe important and frequent signals from the two first months of measuring, and then again nearly no more signal. So, the LIMO analyser will perform a high number of very weak concentrations of limonene. Therefore, the distribution of the limonene measurements is strongly right-tailed.

Range The LIMO analyser possesses a measuring range from 0.1 to 9.4417 $\mu g/m^3$. Three quarter of its values are equal to 0.1 $\mu g/m^3$, as confirmed in the histogram and boxplot.

Missing values One missing value occurs in the measurements of limonene at 23:55:13 o'clock on 30/1/2017. This NA value has also its seeds in the interpolation of the analysers measurements.

Logarithmic time series Figure 3.13 shows the logarithmic LIMO values.

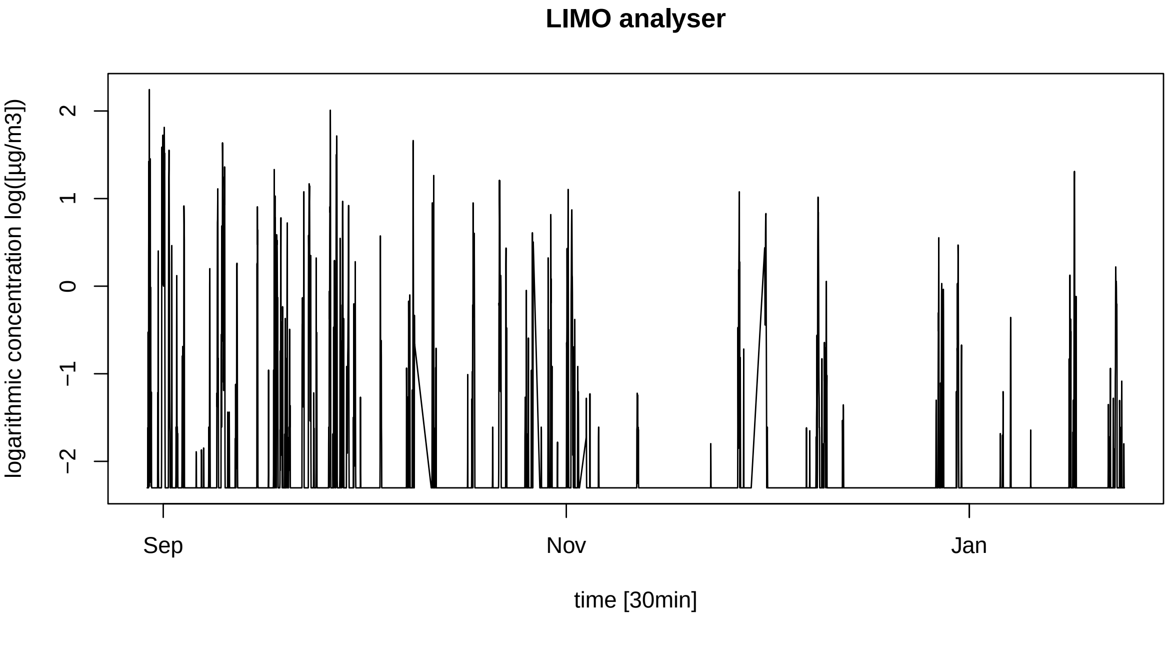


Figure 3.13: Logarithmic LIMO concentration

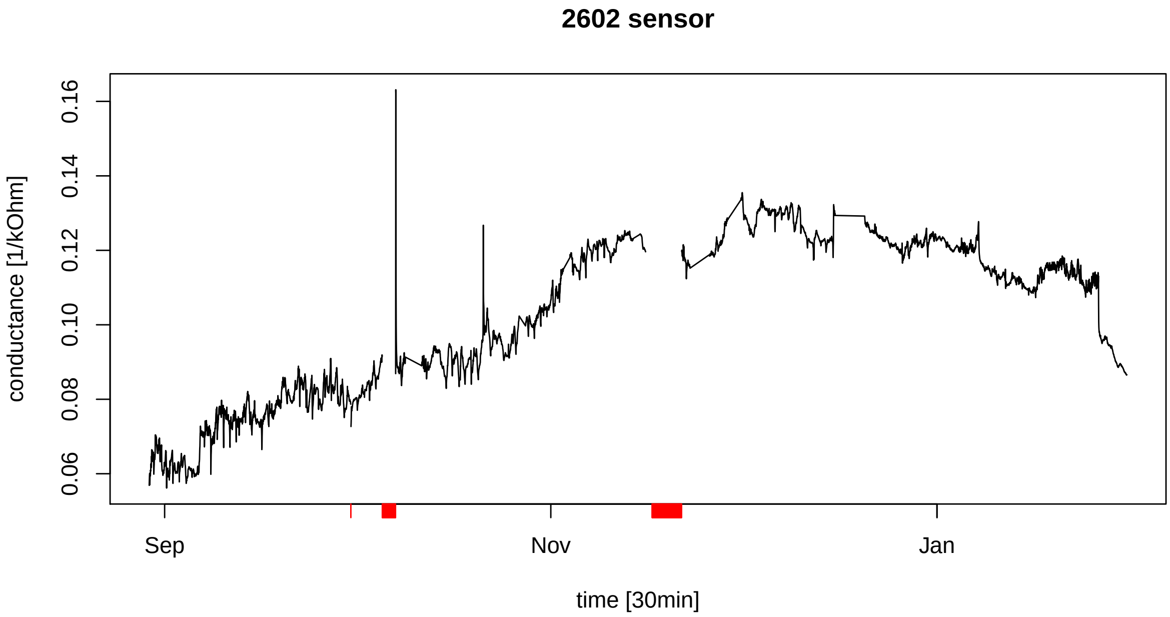
3.2 Sensors values

The same descriptive statistics and graphical representations as for the analysers will be presented for the sensors measurements. Remember that the sensors resistances have been transformed in conductance measurements, so the used unit is $kOhm^{-1}$.

Furthermore, the sensors, situated in the sensors chamber, performed at the same time. Thus, when there is a power breakdown, all sensors stopped measuring simultaneously. The 326 missing values for all sensors take place on three different time periods:

- on "2016-09-30 09:57:00 UTC"
- from "2016-10-05 09:17:00 UTC" to "2016-10-07 11:17:00 UTC"
- from "2016-11-17 00:05:00 UTC" to "2016-11-21 15:35:00 UTC"

3.2.1 Sensor TGS2602



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.0561	0.0844	0.1098	0.1024	0.1212	0.1631	326

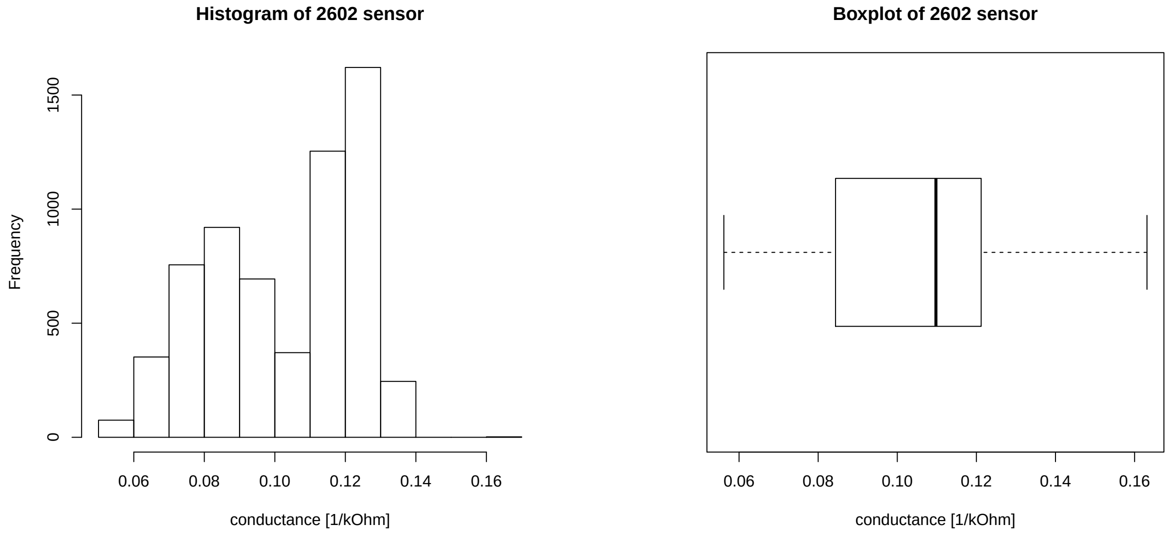
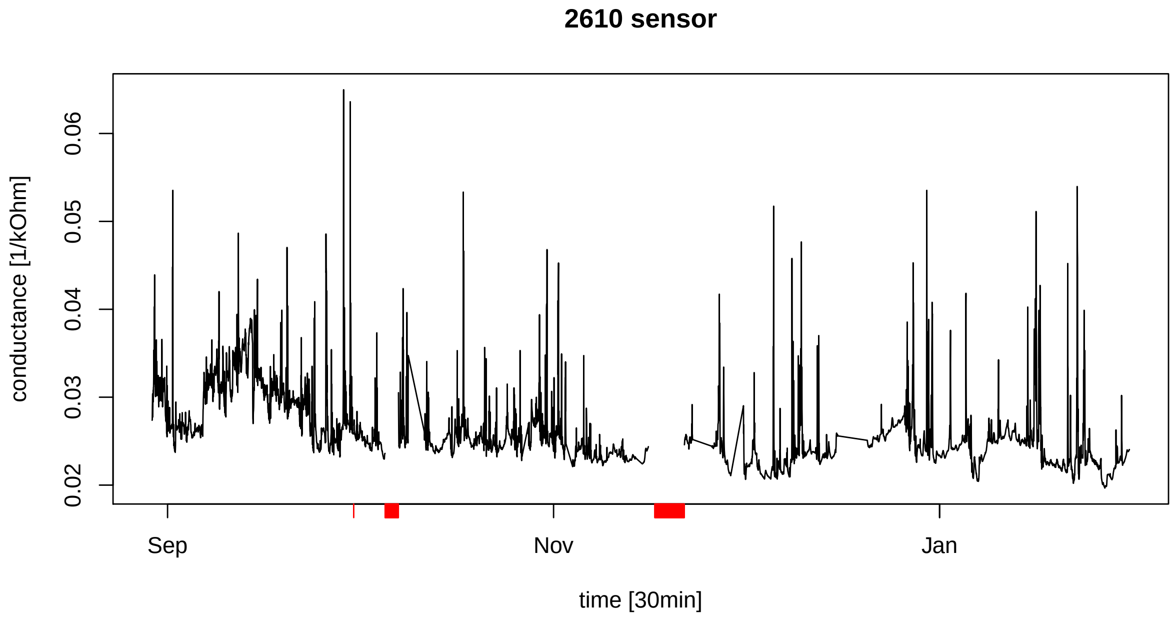


Figure 3.14: Descriptive statistics for the TGS2602 sensor

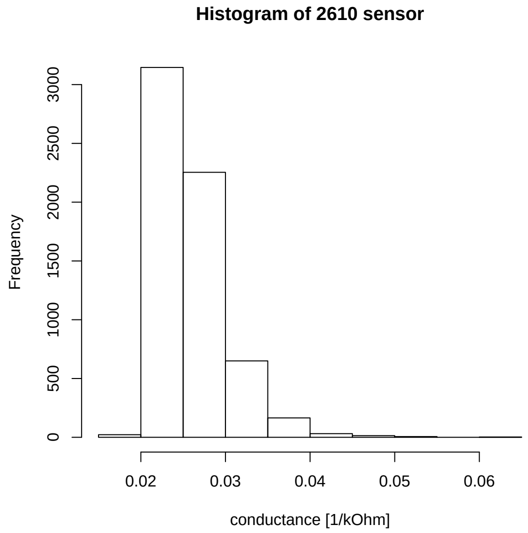
Distribution On Figure 3.14 the measurements of the TGS2602 sensor are illustrated. We observe a global increase from September to December followed by a weak falling down. Remember that the sensors are non specific but very sensitive instruments, which can be influenced by temperature and humidity. Therefore, a reason for this in- and later decrease could be the drop in temperature (winter). A remarkable peak occurs on 7/10/2016 at 12:17:34. Otherwise, the distribution remains very stable, as confirmed in the histogram and the boxplot. Compared to the right-tailed analysers distributions, the distribution of the TGS2602 sensor seems to be more symmetric.

Range The TGS2602 sensor has a measuring range from approximately 0.06 to 0.16 $kOhm^{-1}$. The most frequently occurring measure is between 0.12 and 0.13 $kOhm^{-1}$ by regarding the histogram. On the boxplot, we do not observe extreme values exceeding the whiskers.

3.2.2 Sensor TGS2610



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.0197	0.0235	0.025	0.0259	0.0269	0.065	326



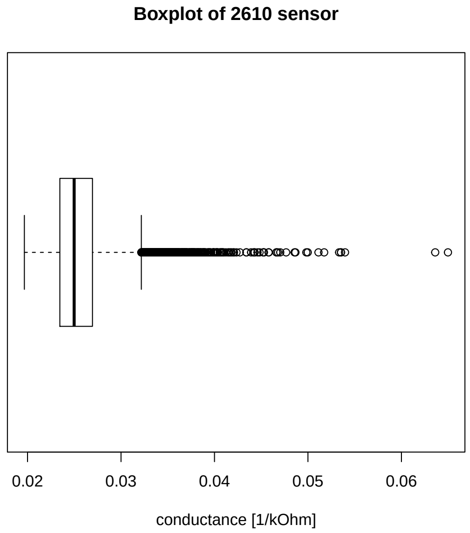
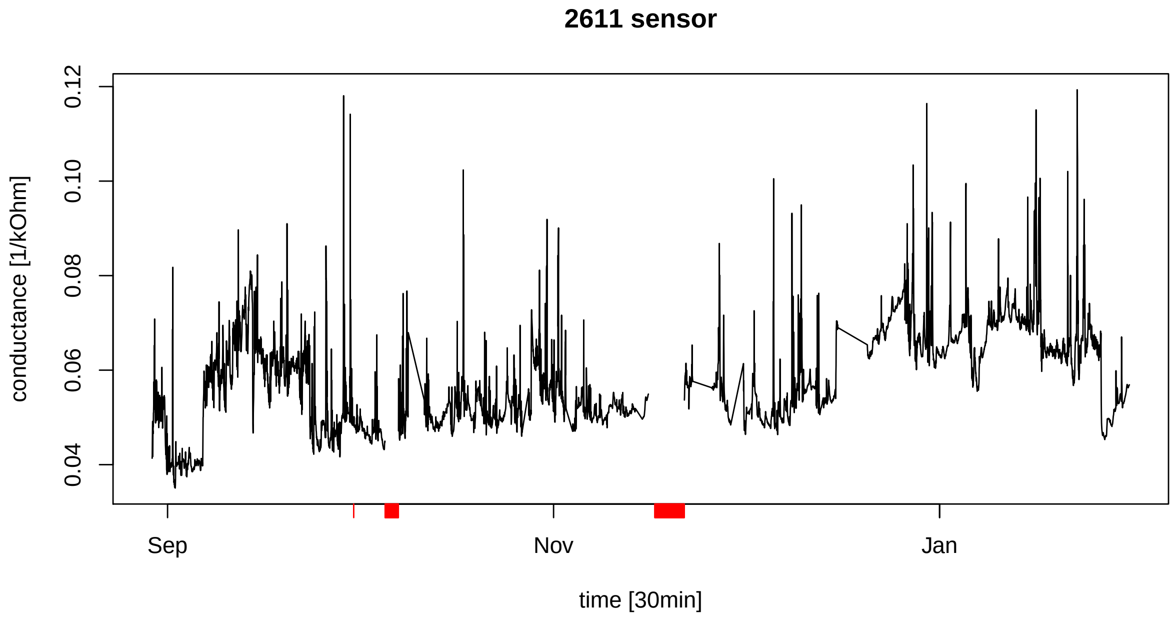


Figure 3.15: Descriptive statistics for TGS2610 sensor

Distribution The TGS2610 sensor presents more fluctuations than the TGS2602 sensor (see Figure 3.15). The measurements of the TGS2610 sensor oscillate around circa 0.025 $kOhm^{-1}$. This can be confirmed by looking on the histogram and boxplot. This figure illustrates also the rather right-tailed character of the distribution. Compared to the TGS2602 sensor just before, we have a more important number of extreme values in the boxplot.

Range The range of the TGS2610 sensor goes from 0.02 to 0.065 $kOhm^{-1}$ approximately. Three quarter of the measurements are less or equal to 0.027. The boxplot and more in detail the small width of the box confirms the huge number of values around 0.025-0.027 $kOhm^{-1}$.

3.2.3 Sensor TGS2611



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.0351	0.0503	0.0559	0.0577	0.0648	0.1193	326

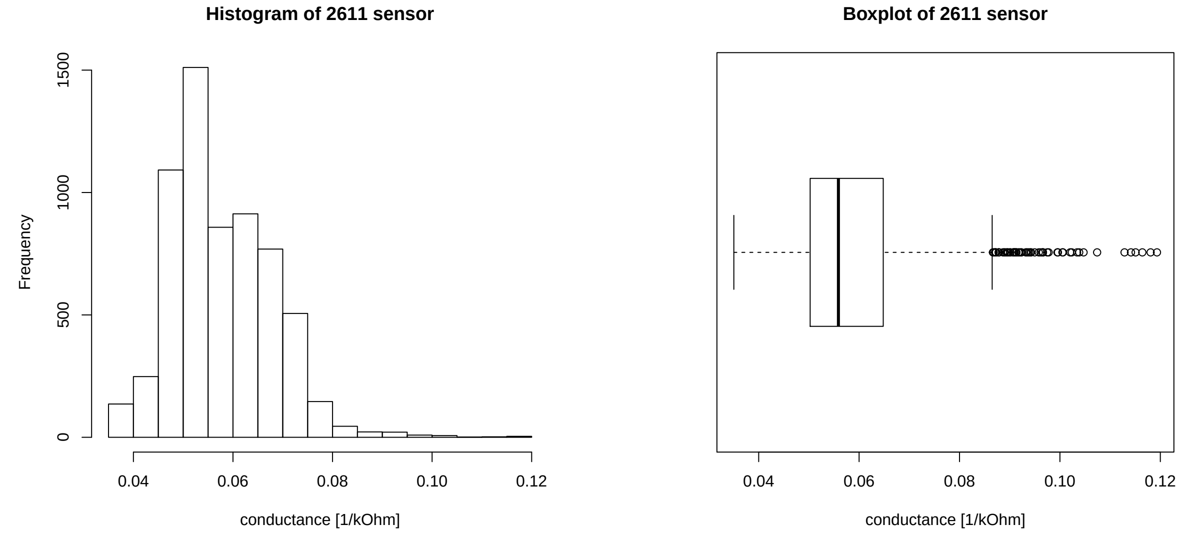
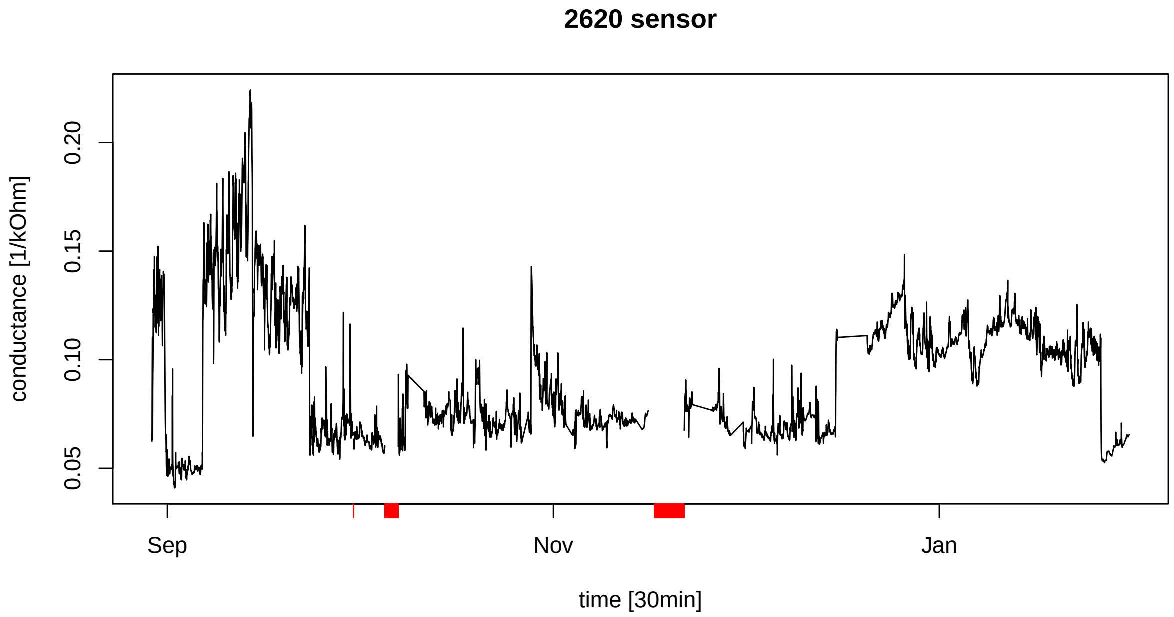


Figure 3.16: Descriptive statistics for TGS2611 sensor

Distribution The TGS2611 sensor produces a very fluctuating signal around 0.05-0.06 $kOhm^{-1}$. The histogram and boxplot represent a light right-tailed distribution with a few extreme values on the right side. The time period from the "2016-09-06 13:46:18 UTC" to "2016-09-23 09:28:36 UTC" shows higher conductance values which precede a little drop down on the first measuring days. During the last five days, the TGS2611 sensor presents smaller measurements as well.

Range We can observe a measuring range from 0.035 to 0.119 $kOhm^{-1}$ for the TGS2611 sensor, which is greater than the range of the precedent TGS2610 sensor.

3.2.4 Sensor TGS2620



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.0409	0.0684	0.0791	0.091	0.1116	0.2242	326

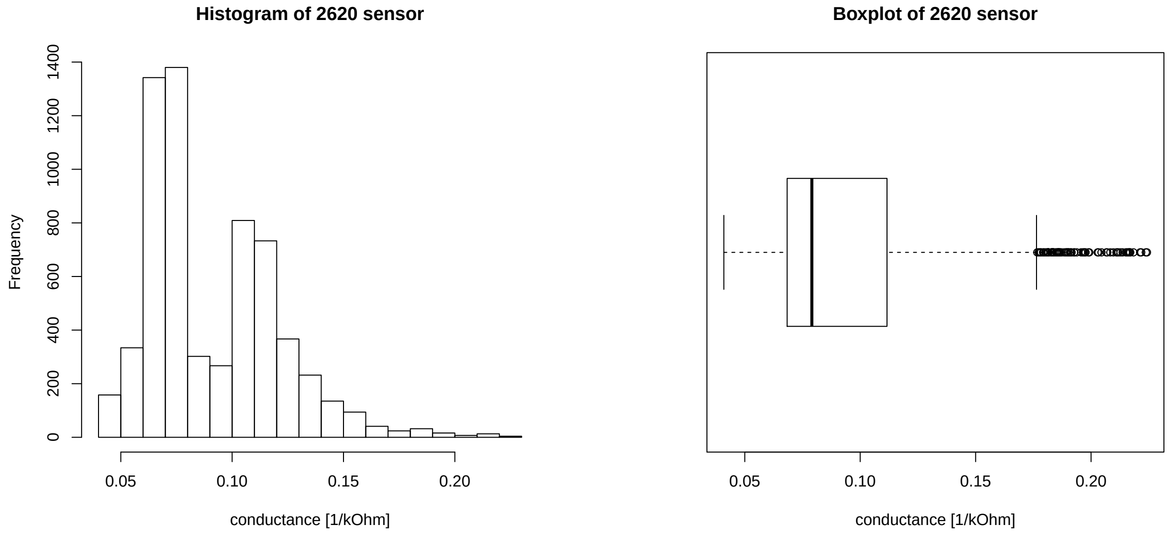
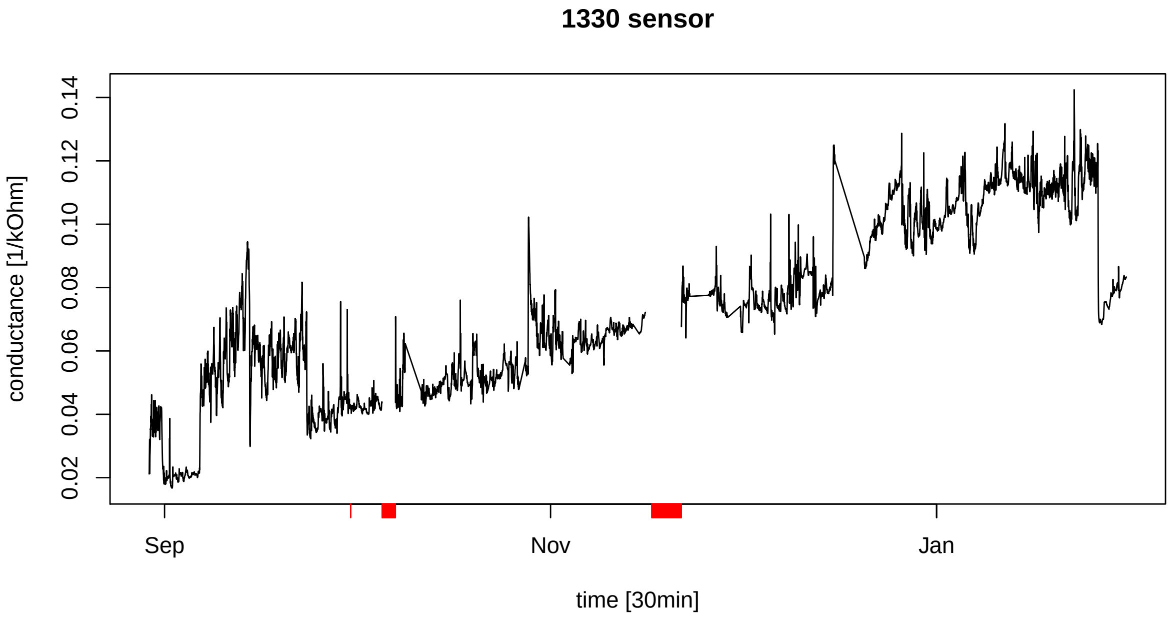


Figure 3.17: Descriptive statistics for TGS2620 sensor

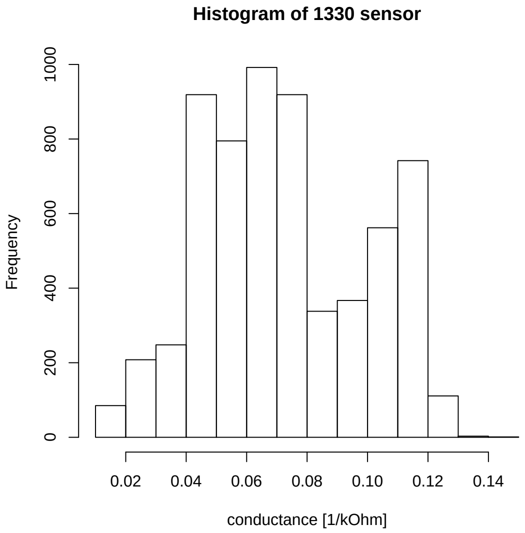
Distribution The values of the TGS2620 sensor represent 3 levels of conductance measurements (see Figure 3.17). First, the values are around 0.125 $kOhm^{-1}$, then decrease up to 0.05 $kOhm^{-1}$ before coming back around 0.125 $kOhm^{-1}$. This period shows the same shifted values as the TGS2611 sensor. After the 23/9/2016, the conductance measurements drop down to 0.05 $kOhm^{-1}$ and increase for the last month around 0.10 $kOhm^{-1}$. We see a weak trend to the left of the distribution. The extreme values in the boxplot present the higher values between the 6/9/2016 and the 23/9/2016.

Range The summary table shows up a measurement range from 0.04 to 0.22 $kOhm^{-1}$ approximately for the TGS2620 sensor. With a value of 0.1833 $kOhm^{-1}$, it is the widest range of the six sensors.

3.2.5 Sensor GGS1330



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.0167	0.0511	0.0685	0.0723	0.0972	0.1425	326



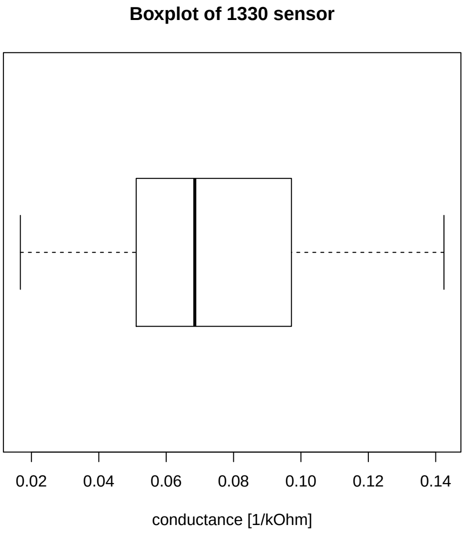
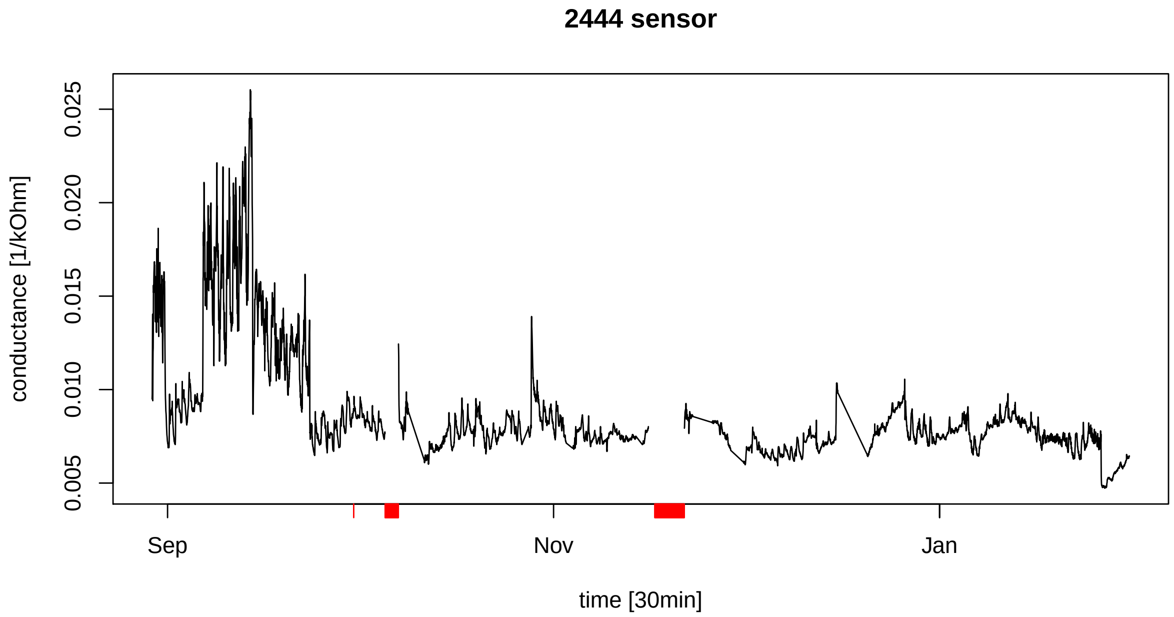


Figure 3.18: Descriptive statistics for GGS1330 sensor

Distribution In Figure 3.18 we can observe a global increasing trend. As for the TGS2611 and TGS2620 sensors, we see a decrease of measurements during the first days followed by higher conductance values for the GGS1330 sensor. Regarding the time series, the distribution shows up a symmetry and no extreme values (see boxplot). The last five days (26/1/2017-30/1/2017), we remark a drop down of the GGS1330 sensor measurements with a difference of circa 0.04 $kOhm^{-1}$.

Range The conductance measurements of the GGS1330 sensor go from 0.0167 to 0.1425 $kOhm^{-1}$, where 50% of the values are between 0.0235 and 0.0972 $kOhm^{-1}$.

3.2.6 Sensor TGS2444



Min	Q_1	Median	Mean	Q_3	Max	Missing values
0.0047	0.0073	0.0078	0.0087	0.0087	0.026	326

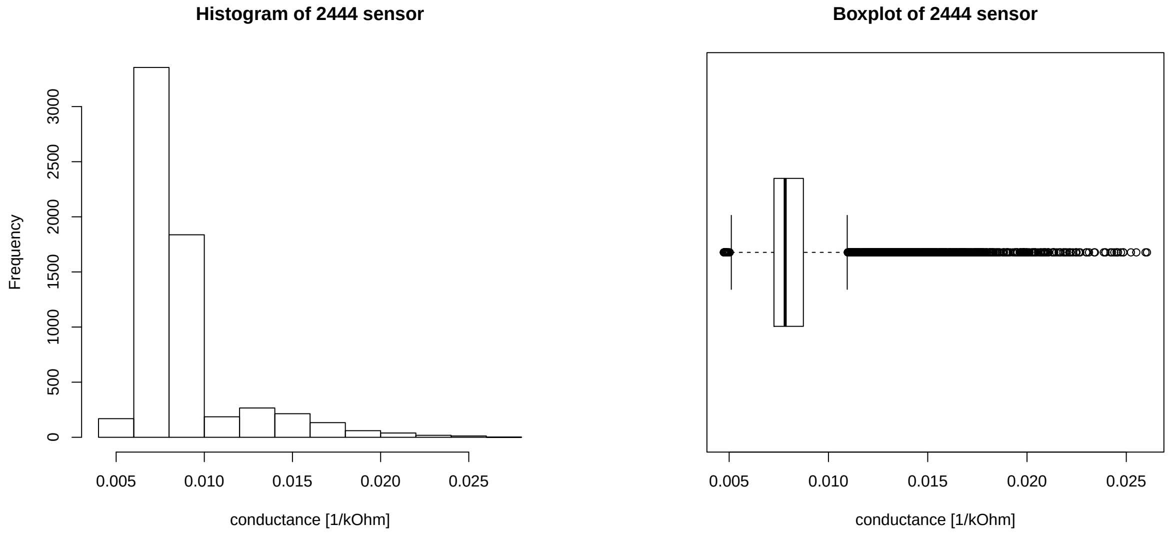


Figure 3.19: Descriptive statistics for TGS2444 sensor

Distribution The TGS2444 sensor presents the most conductance values around 0.006-0.007 $kOhm^{-1}$. More fluctuations occur at the beginning of the TGS2444 sensor measurements. We examine the same signature at the beginning of the curve as for the TGS2611, TGS2620 and GGS1330 sensors: a drop down starting the third day of measuring (31/8/2016) is followed by a strong increase on 6/9/2016 after decreasing again to values oscillating around 0.007-0.008 $kOhm^{-1}$. Like for the other sensors, we can note the little fall down at the end of measurement period. In contradiction to the precedent GGS1330 sensor, we observe a more right-tailed distribution. The outstanding high conductance measurements starting on the 6/9/2016 are probably the reason of the right-skewed form and the extreme values in the boxplot. In contradiction to the other sensors, we also observe measurements at the left side of the left whisker. They represent the measurements at the end of the time series, which are smaller than all other measurements of the TGS2444 sensor.

Range The summary table represents a measurement from 0.0047 to 0.026 $kOhm^{-1}$ for the TGS2444 sensor. Furthermore, we see that an important majority of measurements (75%) are beneath 0.0087 $kOhm^{-1}$. The width of the box in the boxplot confirms the oscillating main values around 0.007 $kOhm^{-1}$.

3.2.7 Joint time evolution of the sensors

Collinearity of the sensors variables

The sensors are sensitive and non specific. Therefore, their measurements often increase simultaneously for all sensors and inversely. Looking at their measurements, we see the high dependence between the six sensors as shown in Figure 3.20. The correlation matrix confirms this dependence (see Figure 3.21).

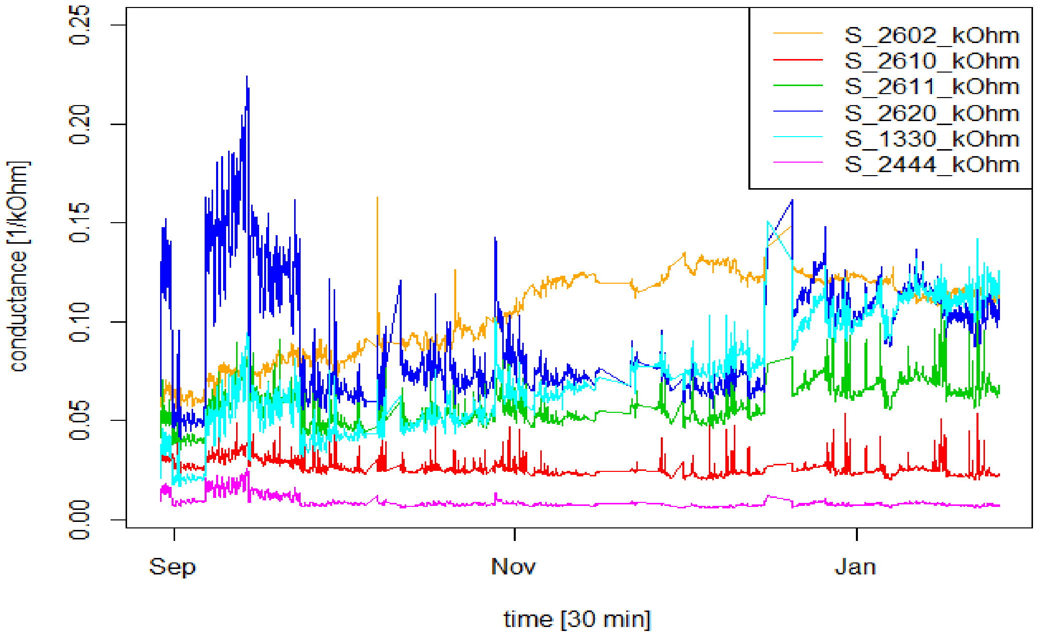


Figure 3.20: Sensors measurements

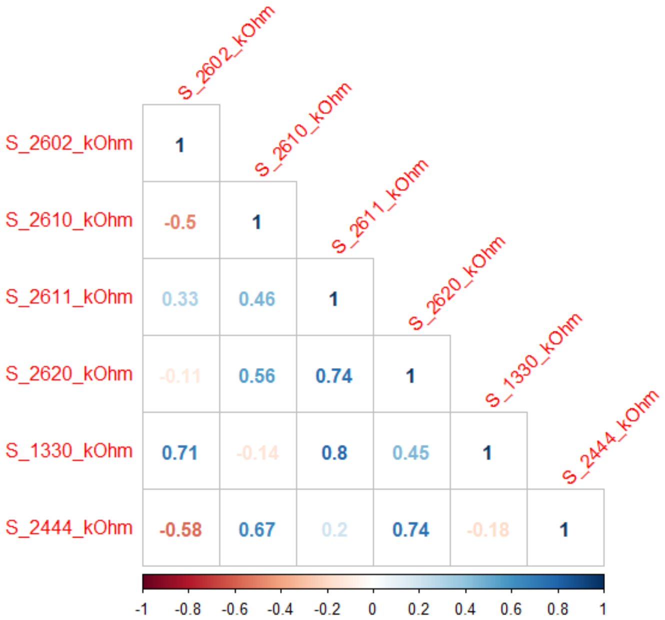


Figure 3.21: Correlation matrix of the six sensors

Interaction of the sensors

Remember that the sensors are non specific instruments returning a conductance value[1] and not the concentration of a particular chemical component. So, because of their low sensitivity[2] and non specificity[3], the signal is expected to be located in the combination of several sensors. Therefore, we add the interactions up to the maximal order of all six sensors to the explanatory variables (see Table 3.1).

Table 3.1: Table of sensor interactions

Order	Number	Example
1	$C_6^1 = 6$	S_{2602}
2	$C_6^2 = 15$	$S_{2602} \cdot S_{2610}$
3	$C_6^3 = 20$	$S_{2602} \cdot S_{2610} \cdot S_{2611}$
4	$C_6^4 = 15$	$S_{2602} \cdot S_{2610} \cdot S_{2611} \cdot S_{2620}$
5	$C_6^5 = 6$	$S_{2602} \cdot S_{2610} \cdot S_{2611} \cdot S_{2620} \cdot S_{1330}$
6	$C_6^6 = 1$	$S_{2602} \cdot S_{2610} \cdot S_{2611} \cdot S_{2620} \cdot S_{1330} \cdot S_{2444}$
Total	63	

[1] We converted the resistance measurements into the conductance of the sensors in the preprocessing (Section 2.2.4).

[2] The intensity of the responses is not very high for any component.

[3] They react to a large range of chemical compounds.

Naturally, the interactions of the sensors are also dependent on the sensors measurements. Therefore, we perform a principal component analysis (PCA) on the interactions of sensors.

Non reliable periods of measurements

On Figure 3.22, the sensors measurements are representing two time periods standing out by higher values. The orange time period represents the measures from "2016-10-09 01:00:00 UTC" to "2016-10-11 13:00:00 UTC" and the green one from "2016-12-15 22:00:00 UTC" to "2016-12-20 14:00:00 UTC". These shifts are not reliable in terms of the real measurements and are most likely a consequence of dirt present in the pipe of air intake.

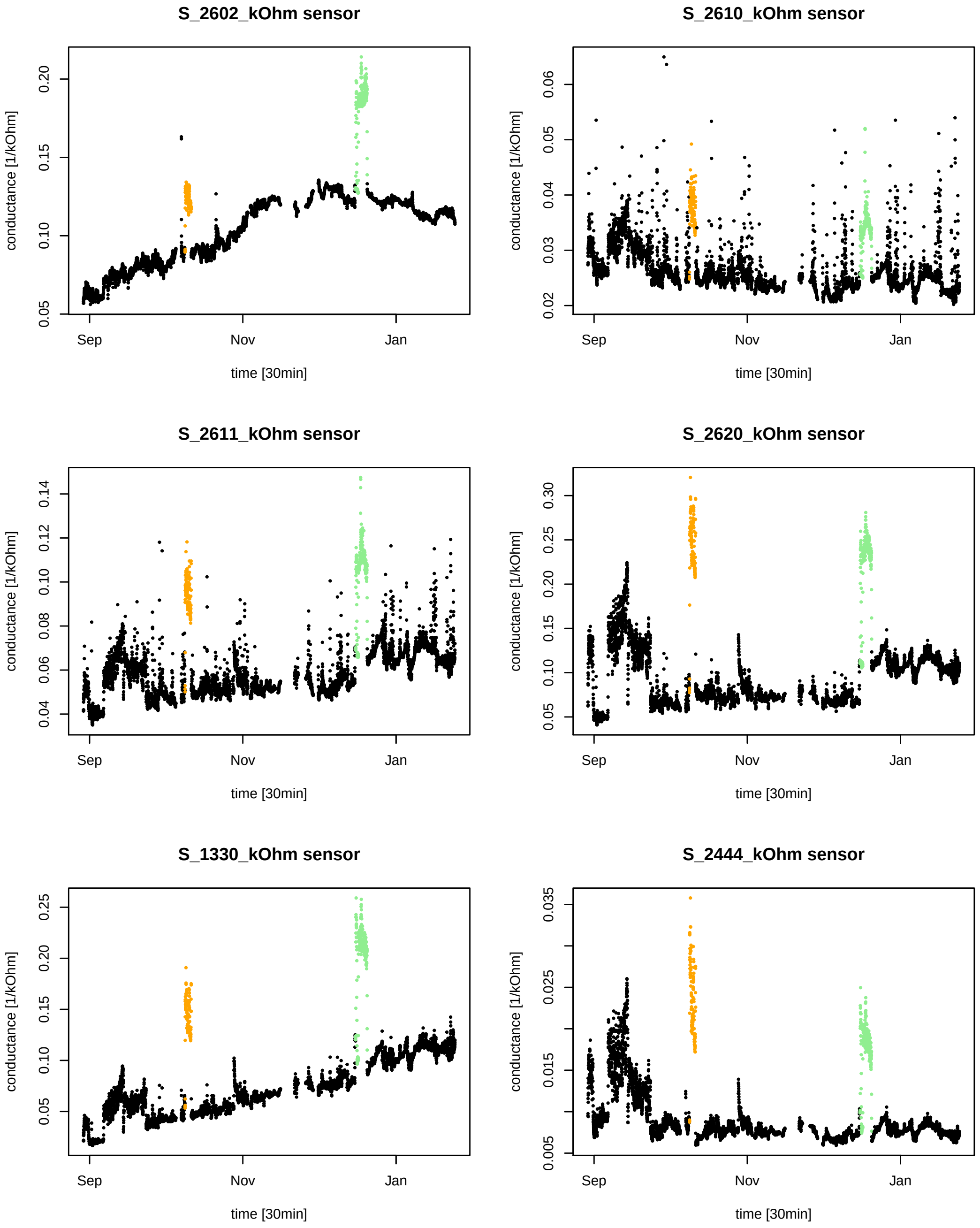


Figure 3.22: Shifts of the sensors

The next Figure 3.23 represents the individual factor map in terms of the first and second principal components of the PCA. Every individual on this figure represents a moment in time. Dim.1 and Dim.2 are the first two principal components of the PCA, which are linear combinations of the interactions. On this figure, we remark that the two specified groups of observations have a different temporary connection than the other individuals. Consequently, they influence strongly the principal components and thus also the prediction model, whose explanatory variables are these principal components of the sensors interactions.

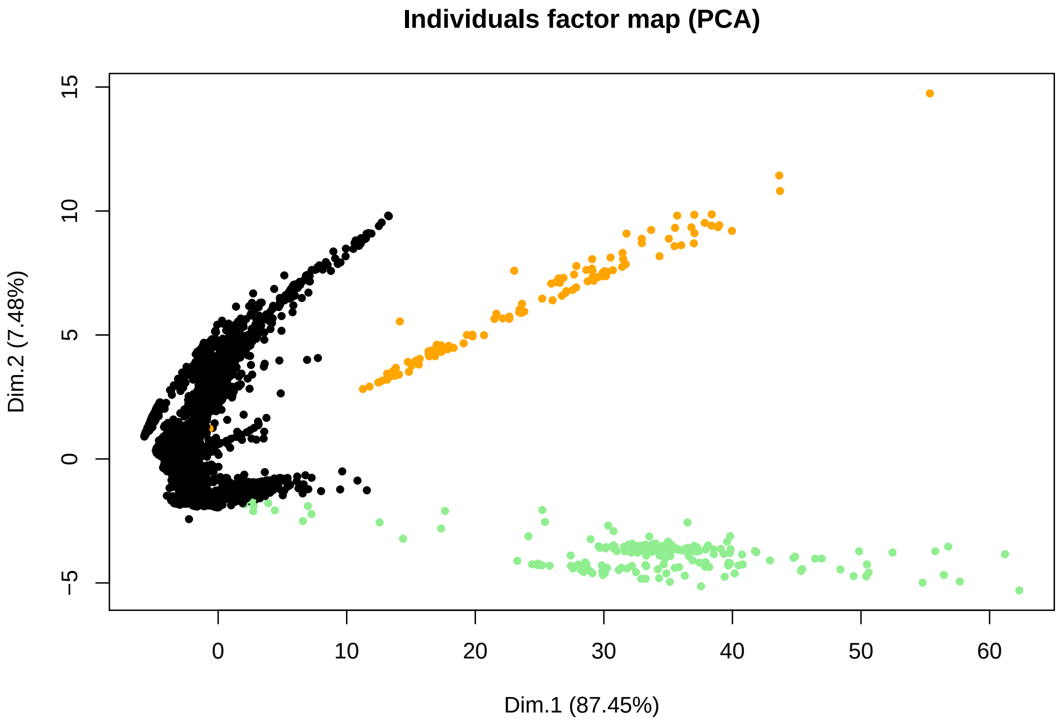


Figure 3.23: Individuals represented in function of the two principal components

Baseline

When we return to Figure 3.20, we observe that all the sensors present a baseline. We take the TGS2620 sensor to illustrate more in detail this baseline (see Figure 3.24). The presence of the baseline can be explained by the environmental sensibility of the sensors. This influence factor causes deviations which are not conditional on concentrations. The analysers in contrary do not report these deviations.

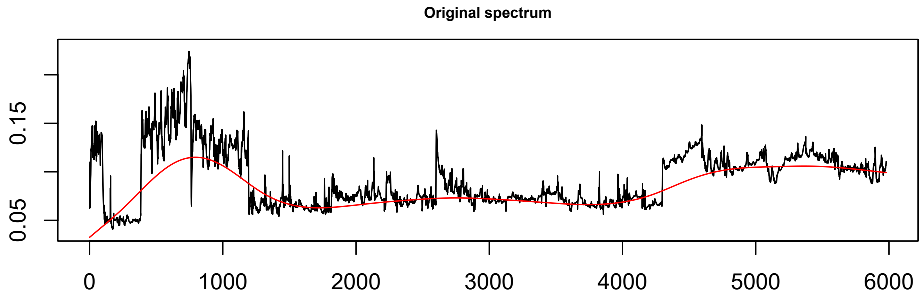


Figure 3.24: Presence of baseline for the TGS2620 sensor

The next Figure 3.25 represents the individual factor map in terms of the first and the second principal components after having removed the observations from "2016-10-09 01:00:00 UTC" to "2016-10-11 13:00:00 UTC" and from "2016-12-15 22:00:00 UTC" to "2016-12-20 14:00:00 UTC".

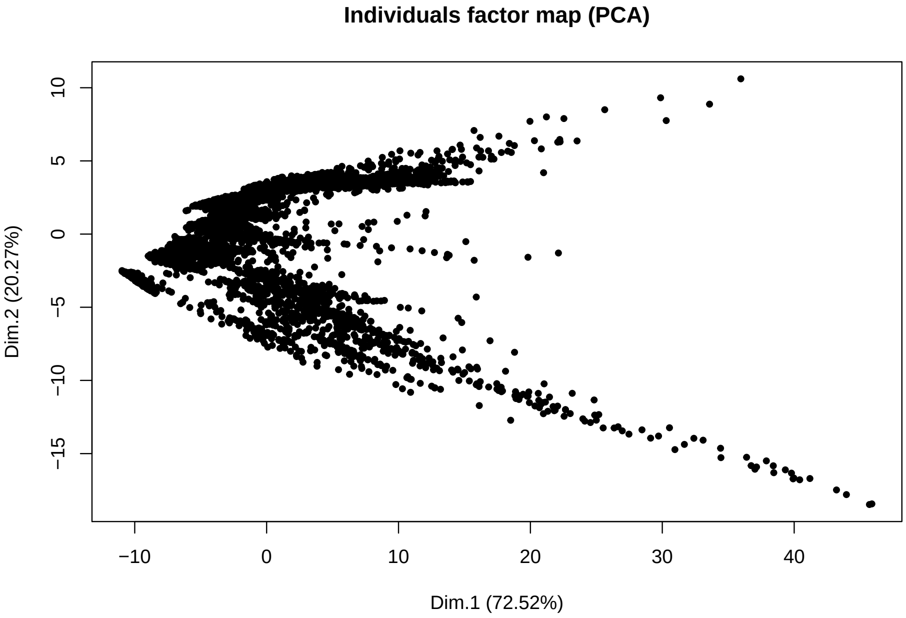


Figure 3.25: Individual factor map of the PCA

We see a certain structure in this representation. Again, we will go back to the representation of the sensors to examine where this structure has its origin (see Figure 3.26). We observe a "jump" for 3 of the six sensors (2620, 1330, 2444). These observations, signalised in light blue, have been measured in the time period from "2016-09-06 13:46:18 UTC" to "2016-09-23 09:28:36 UTC". The same colouration is used in the individual factor map plot (see Figure 3.27). We conclude that the "shift" in the sensors values is the reason for this special structure in the individual plot.

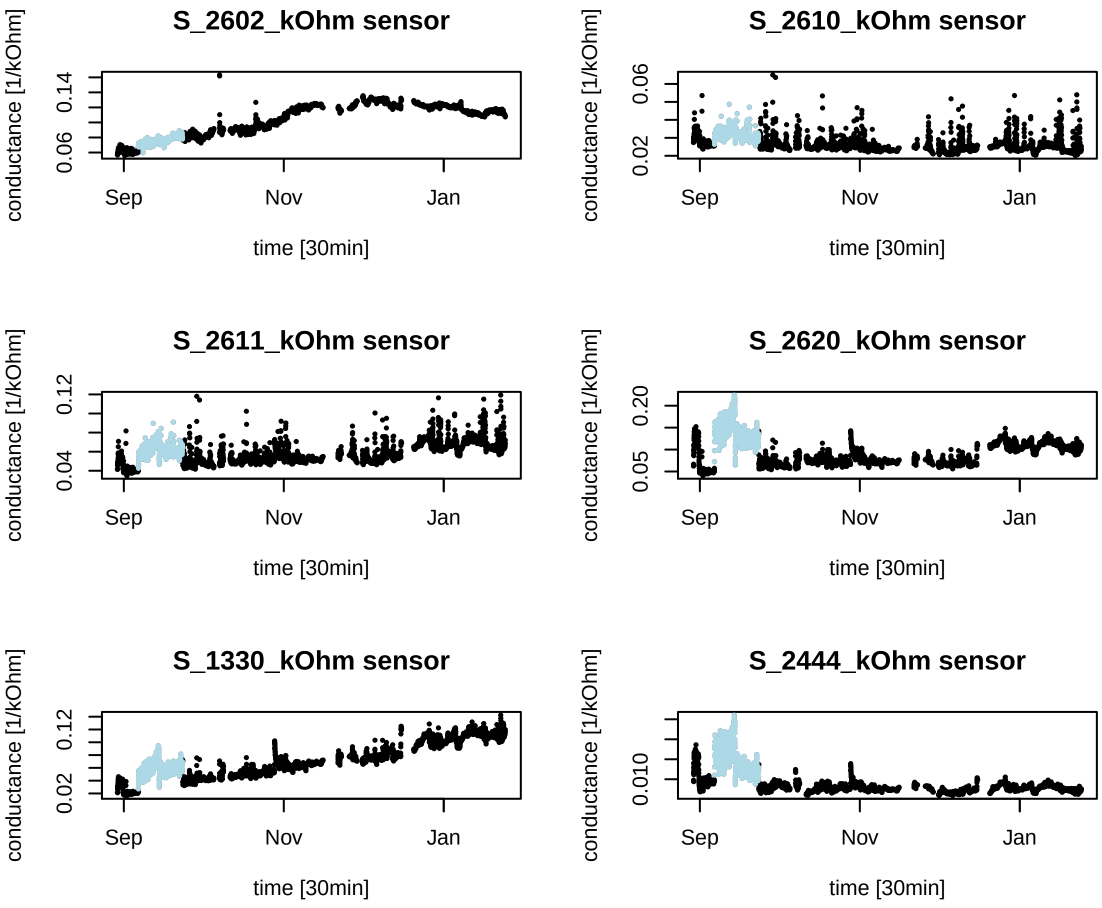


Figure 3.26: Shift of the sensors

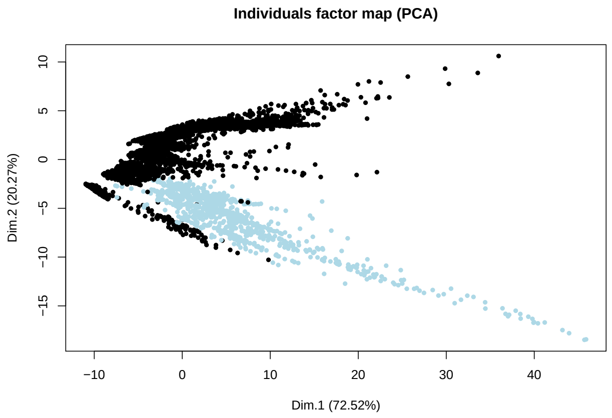


Figure 3.27: Individual factor map of the PCA

3.2.8 Data pretreatment

Exclusion of the non reliable measurements

Given that the shifted values are not representing real concentrations but rather the presence of an external event, we exclude these observations from the data set.

Baseline correction

By the aid of the baseline correction explained beneath, we want to avoid that the principal component analysis is defined for the most part by the deviation of the sensors. We will take the TGS2620 sensor as example, to show how the baseline correction works. The conductance measurements of this sensor are represented in the following Figure 3.28.

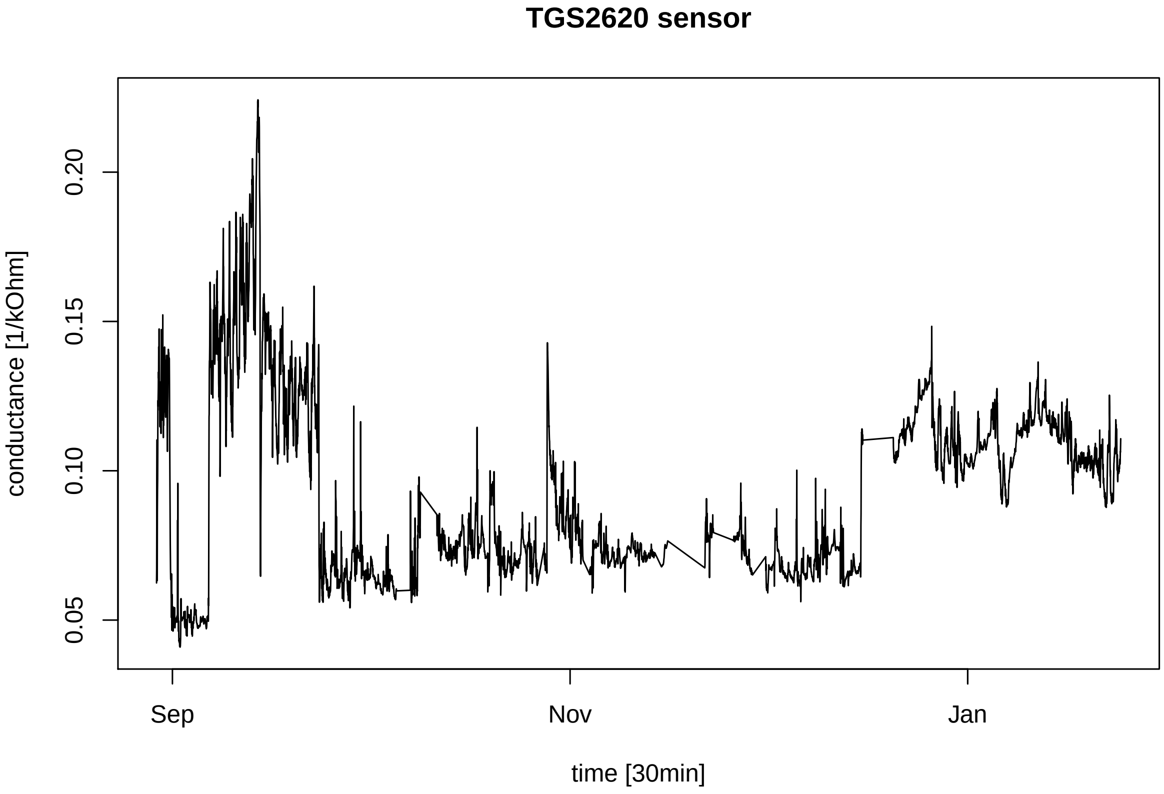
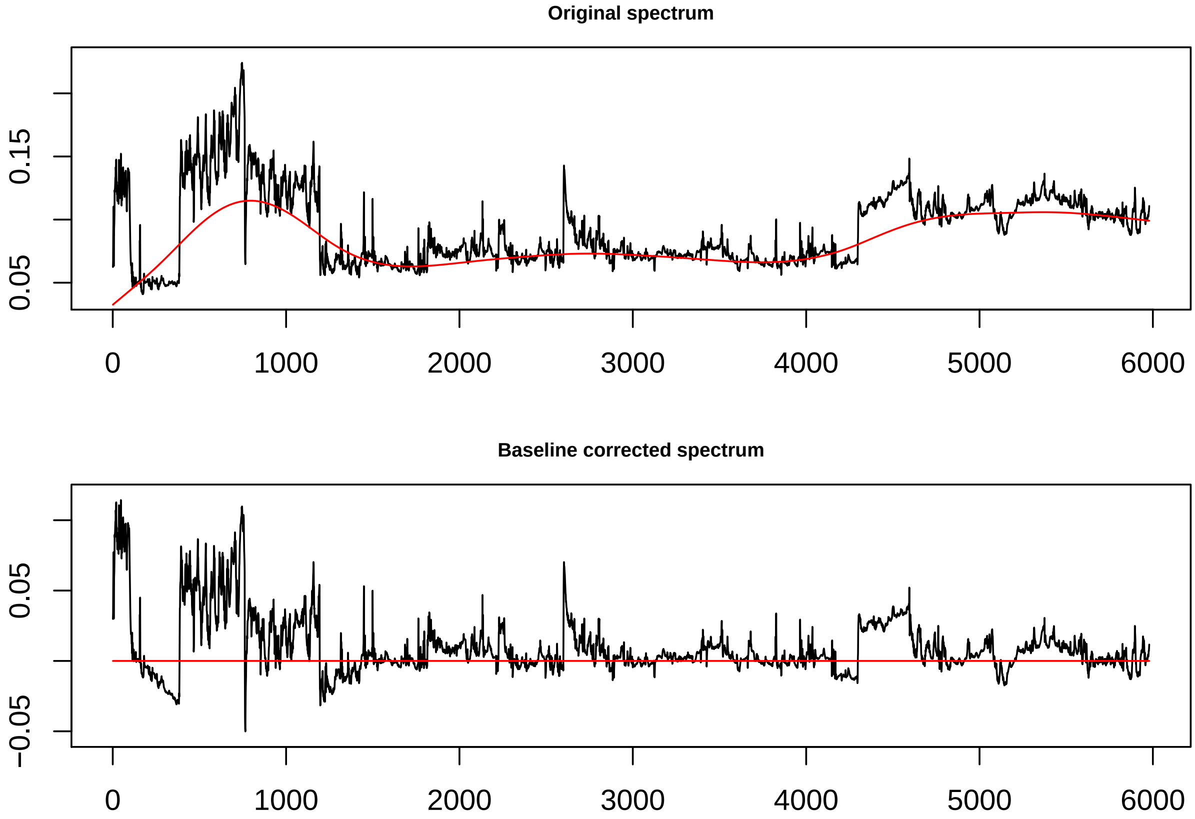


Figure 3.28: TGS2620 sensor measurements

We can observe the shifted period over the time series from the 2016-09-06 at 13:46:18 to the 2016-09-23 at 09:28:36. We will proceed a *baseline correction*, to avoid a too important influence of this shift in our prediction later on.

We employ the baseline function of the baseline package in R [13]. This function uses the default method "IRLS", standing for "Iterative Restricted Least Squares". The description of this method says, that it consists of an algorithm with primary smoothing and repeated baseline suppressions and regressions with second derivative constraint. On Figure 3.29, the red line represents the baseline for the example of the TGS2620 sensor. Above, we see the original measurements of the sensor, underneath the correction of the same sensor by the baseline is illustrated.



Original spectrum

Baseline corrected spectrum

Figure 3.29: TGS2620 sensor measurements before and after baseline correction

A disadvantage of the baseline correction is a limitation of the rows in the data set. If the data set, that has to undergo the correction, has a too large number of rows, the baseline correction can not take place and an error message appears in R. A manual research of the limitation leads to the result that data sets with up to 45000 rows approximatively can be corrected by the baseline algorithm. Beyond this, the error can occur and the correction is not possible. If the limitation number of rows exceeds, the function returns a warning message and the columns will not be corrected. In this study, we do not exceed this limit because of the time reduction to half hour steps.

Chapter 4

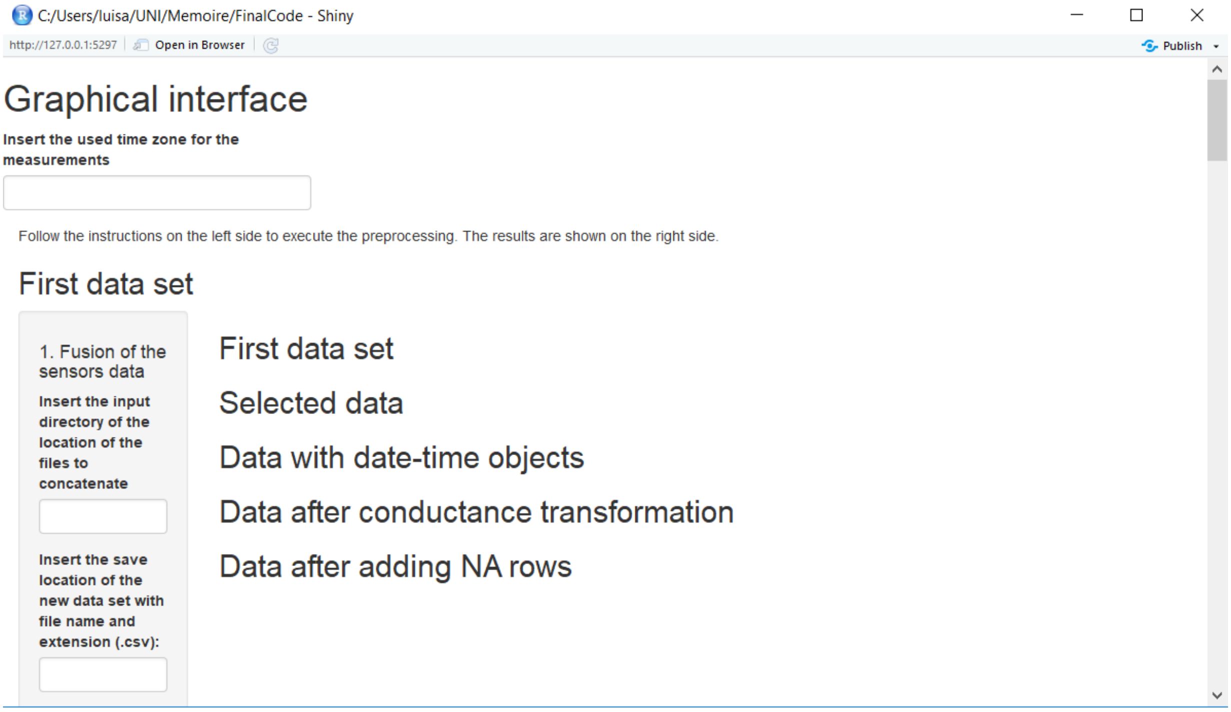
Graphical user interface

In this chapter, we present the graphical user interface created for a simple use of the preprocessing functions described in Chapter 2, which can be applied to any dataset having a similar format as the dataset in this study. The R script of the user interface is available on the MatheO platform.

First, we have to remark that the interface is created in a Shiny application in R-Studio. The R-package *shiny* is used for the creation of interactive interfaces, which has to be installed before executing the application. The second required package is the library *lubridate* for the creation of the date-time objects. Once the necessary libraries are installed, the entire script can be executed and the shiny interface appears in a new window.

On Figure 4.1, the first view after having executed the code is illustrated. The instructions for the user are written on the left side, the consequential results are displayed on the right side. Furthermore, the instructions are in the same order as the functions in the preprocessing.

We start to fill in the questioned inputs and see the first data set. The first input defines the time zone of our data (e.g. UTC), which is not enclosed in quotation marks as in R. Then, we start with the preprocessing of the first data set. We insert the location of the files which have to be concatenated to one file, like the sensors files. Therefore, we must use the forward slash (/) and we do not have to enclose the directory in quotation marks, e.g. C:/Users/User/folder1/sensors_files. Afterwards, the location to save the concatenated data set is demanded with the extension .csv but also without quotation marks, e.g. C:/Users/user/folder2/merged_data.csv. The last step in this part is to insert the name of the column containing the date values, e.g. Date. The entries are validated by clicking the "Ok"-button and the header of the data set appears on the right side as shown in Figure 4.2.



Graphical interface

Insert the used time zone for the measurements

Follow the instructions on the left side to execute the preprocessing. The results are shown on the right side.

First data set

1. Fusion of the sensors data

Insert the input directory of the location of the files to concatenate

Insert the save location of the new data set with file name and extension (.csv):

First data set

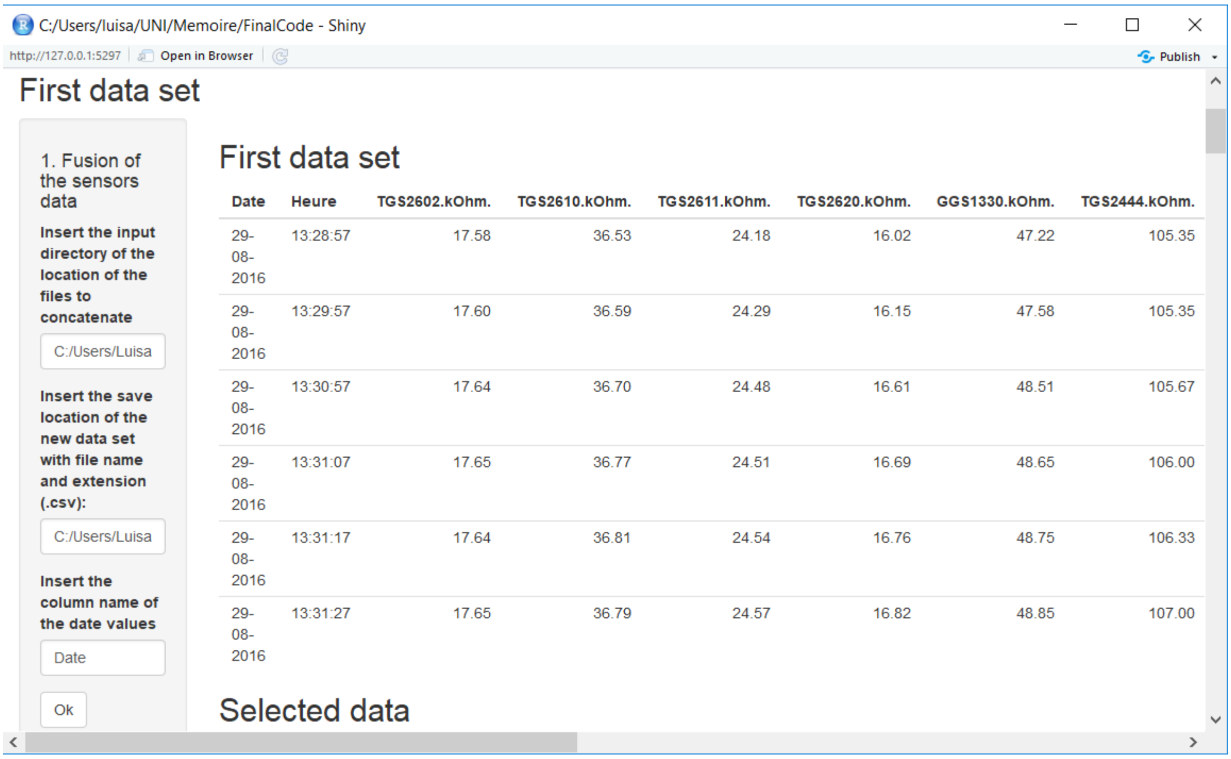
Selected data

Data with date-time objects

Data after conductance transformation

Data after adding NA rows

Figure 4.1: First view of the shiny application



First data set

1. Fusion of the sensors data

Insert the input directory of the location of the files to concatenate

C:/Users/Luisa

Insert the save location of the new data set with file name and extension (.csv):

C:/Users/Luisa

Insert the column name of the date values

Date

Ok

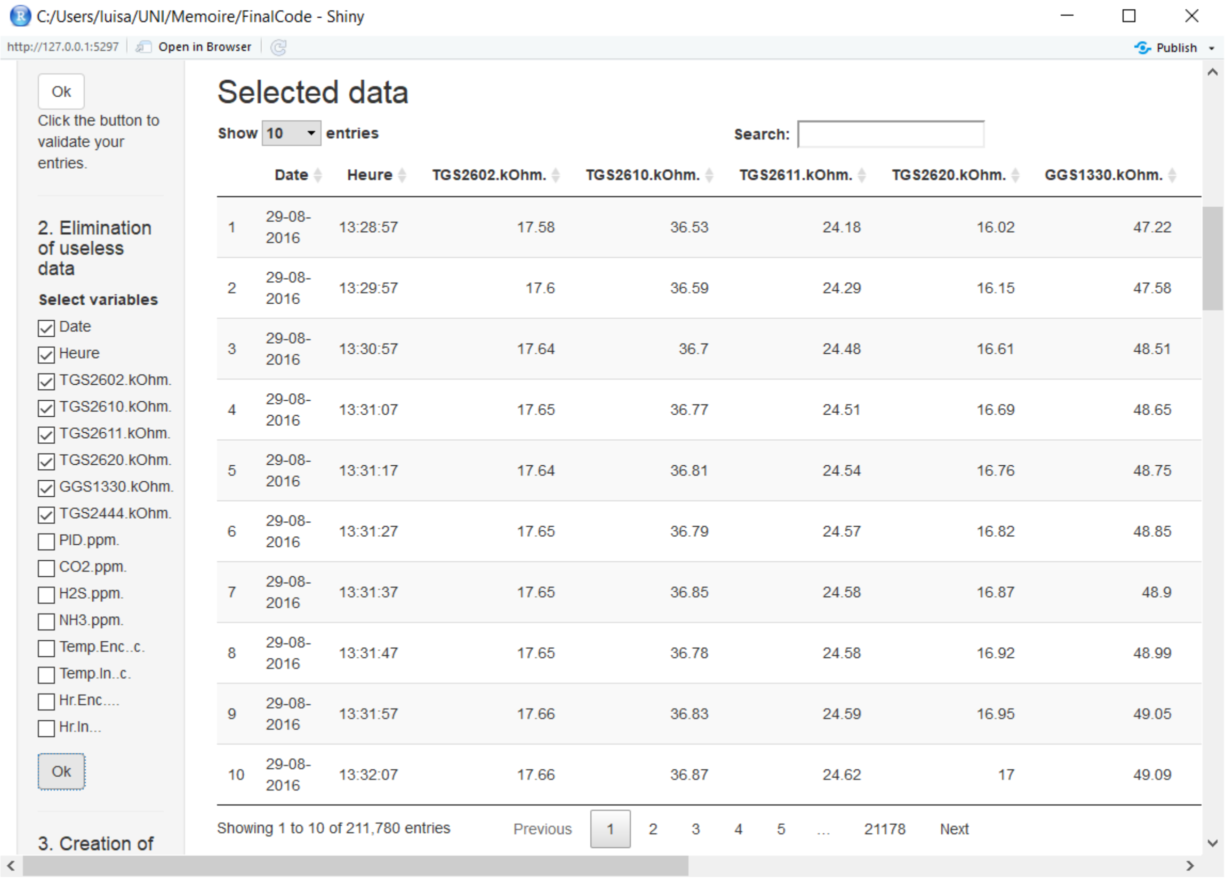
First data set

Date	Heure	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS2444.kOhm.
29-08-2016	13:28:57	17.58	36.53	24.18	16.02	47.22	105.35
29-08-2016	13:29:57	17.60	36.59	24.29	16.15	47.58	105.35
29-08-2016	13:30:57	17.64	36.70	24.48	16.61	48.51	105.67
29-08-2016	13:31:07	17.65	36.77	24.51	16.69	48.65	106.00
29-08-2016	13:31:17	17.64	36.81	24.54	16.76	48.75	106.33
29-08-2016	13:31:27	17.65	36.79	24.57	16.82	48.85	107.00

Selected data

Figure 4.2: Creation of the first data set

The second step is dedicated to the elimination of useless data in the first data set. Therefore, we choose the variables we want to keep in the data set by marking them, and delete the mark for the useless variables. Anew, we click the "Ok"-button and a new data set with the selected variables appears (see Figure 4.3).



Click the button to validate your entries.

2. Elimination of useless data

Select variables

☑ Date
☑ Heure
☑ TGS2602.kOhm.
☑ TGS2610.kOhm.
☑ TGS2611.kOhm.
☑ TGS2620.kOhm.
☑ GGS1330.kOhm.
☑ TGS2444.kOhm.
□ PID.ppm.
□ CO2.ppm.
□ H2S.ppm.
□ NH3.ppm.
□ Temp.Enc..c.
□ Temp.In..c.
□ Hr.Enc....
□ Hr.In...

Selected data

Show 10 entries   Search:

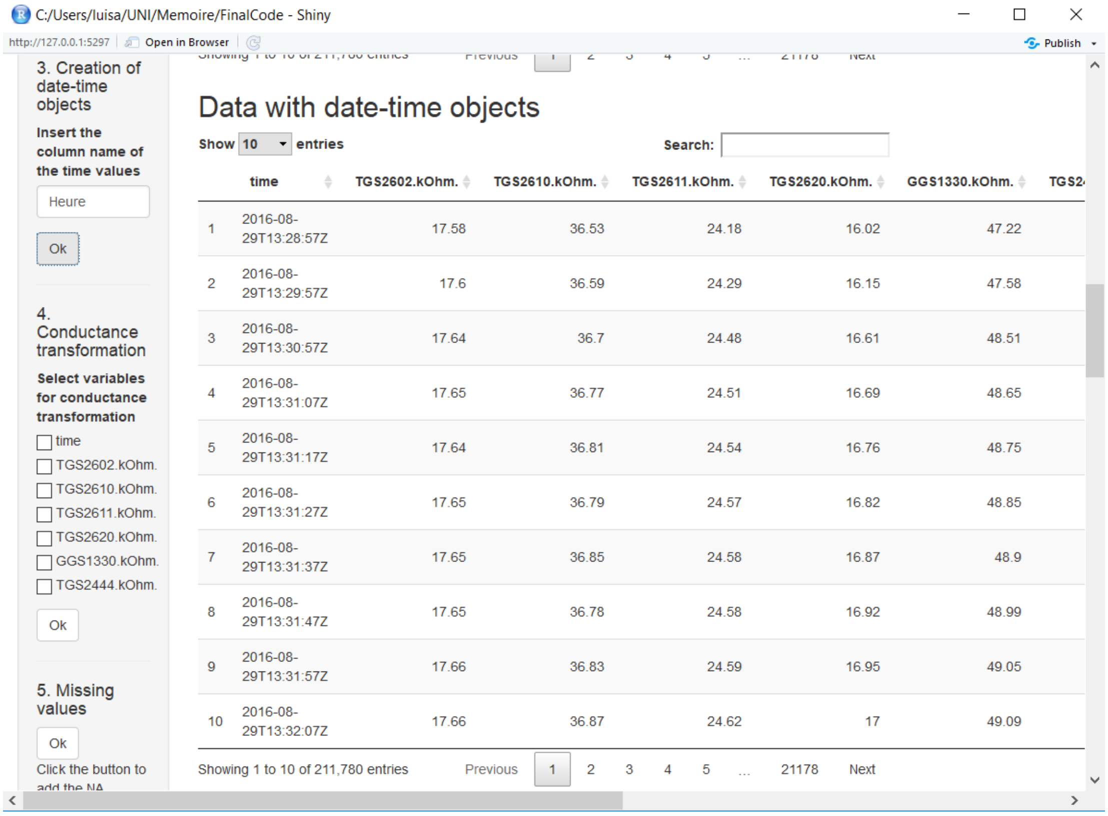
	Date	Heure	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.
1	29-08-2016	13:28:57	17.58	36.53	24.18	16.02	47.22
2	29-08-2016	13:29:57	17.6	36.59	24.29	16.15	47.58
3	29-08-2016	13:30:57	17.64	36.7	24.48	16.61	48.51
4	29-08-2016	13:31:07	17.65	36.77	24.51	16.69	48.65
5	29-08-2016	13:31:17	17.64	36.81	24.54	16.76	48.75
6	29-08-2016	13:31:27	17.65	36.79	24.57	16.82	48.85
7	29-08-2016	13:31:37	17.65	36.85	24.58	16.87	48.9
8	29-08-2016	13:31:47	17.65	36.78	24.58	16.92	48.99
9	29-08-2016	13:31:57	17.66	36.83	24.59	16.95	49.05
10	29-08-2016	13:32:07	17.66	36.87	24.62	17	49.09

Showing 1 to 10 of 211,780 entries

Figure 4.3: Selected data set

The third step of the preprocessing of the sensors data is the parsing into date-time objects. For it, we have to enter the name of the column containing the time values, e.g. Heure. Then, we click the "Ok"-button and the new data set with date-time objects in the first column is shown as seen in Figure 4.4.

Remember that the sensors measurements are resistance measurements, but we want them to be transformed into conductance measurements. Therefore, we choose all variables of the last data set that should be transformed into conductance measurements by marking them on the left side. We validate our choice and the new data set is represented as illustrated in Figure 4.5.



3. Creation of date-time objects

Insert the column name of the time values

Heure

Ok

4. Conductance transformation

Select variables for conductance transformation

□ time
□ TGS2602.kOhm.
□ TGS2610.kOhm.
□ TGS2611.kOhm.
□ TGS2620.kOhm.
□ GGS1330.kOhm.
□ TGS2444.kOhm.

Ok

5. Missing values

Ok

Click the button to add the NA

Showing 1 to 10 of 211,780 entries Previous 1 2 3 4 5 ... 21178 Next

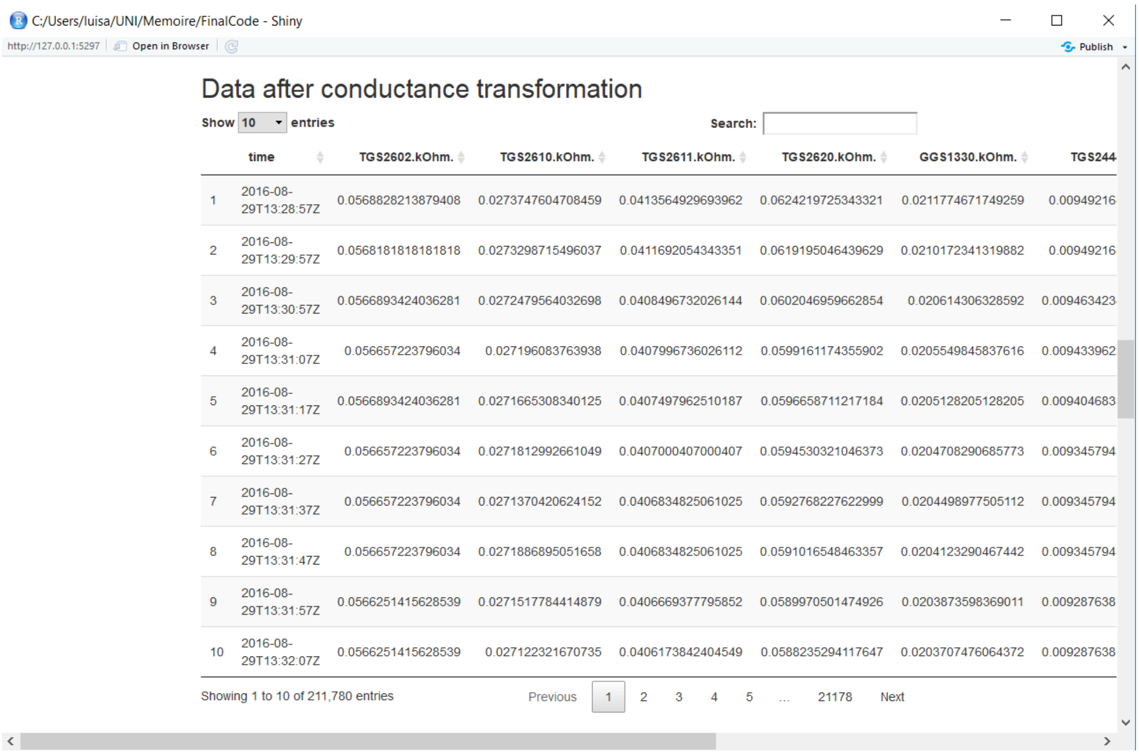
Data with date-time objects

Show 10 entries Search:

	time	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS24
1	2016-08-29T13:28:57Z	17.58	36.53	24.18	16.02	47.22	
2	2016-08-29T13:29:57Z	17.6	36.59	24.29	16.15	47.58	
3	2016-08-29T13:30:57Z	17.64	36.7	24.48	16.61	48.51	
4	2016-08-29T13:31:07Z	17.65	36.77	24.51	16.69	48.65	
5	2016-08-29T13:31:17Z	17.64	36.81	24.54	16.76	48.75	
6	2016-08-29T13:31:27Z	17.65	36.79	24.57	16.82	48.85	
7	2016-08-29T13:31:37Z	17.65	36.85	24.58	16.87	48.9	
8	2016-08-29T13:31:47Z	17.65	36.78	24.58	16.92	48.99	
9	2016-08-29T13:31:57Z	17.66	36.83	24.59	16.95	49.05	
10	2016-08-29T13:32:07Z	17.66	36.87	24.62	17	49.09	

Showing 1 to 10 of 211,780 entries Previous 1 2 3 4 5 ... 21178 Next

Figure 4.4: Data set with date-time objects



Data after conductance transformation

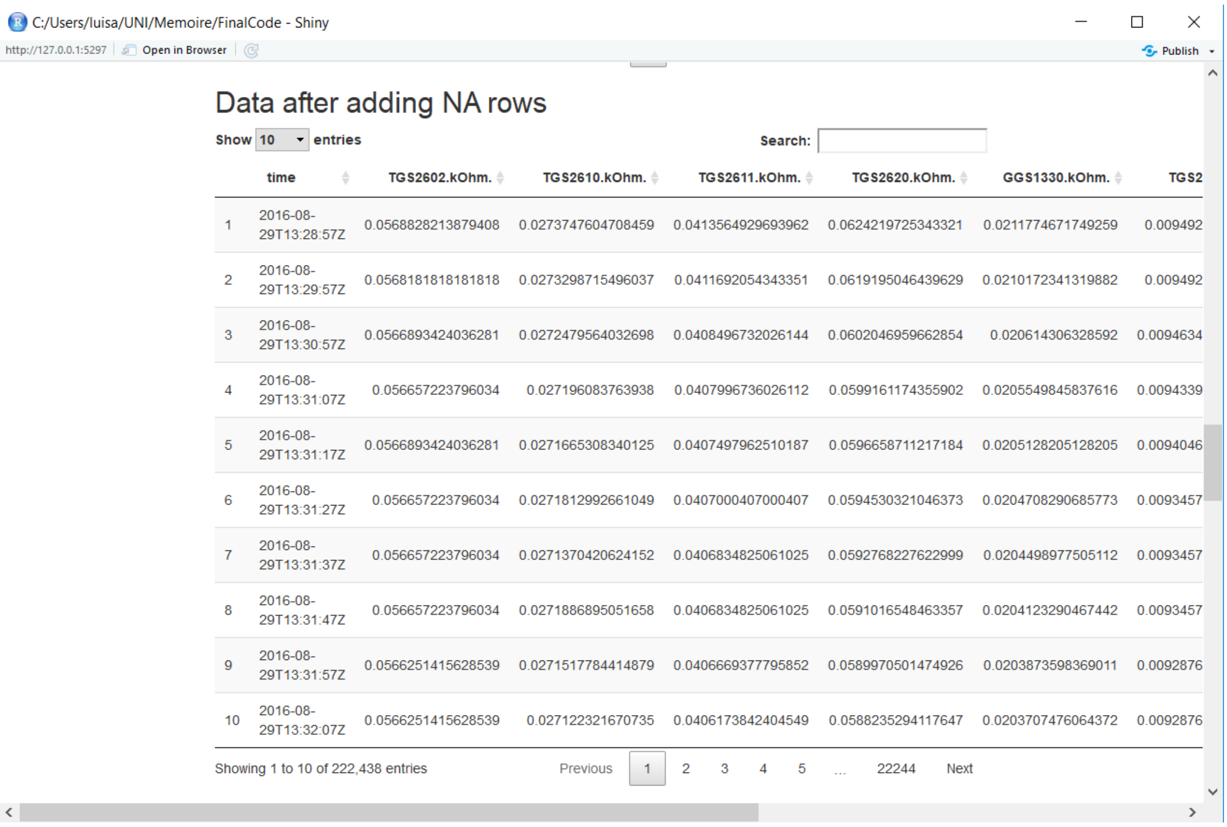
Show 10 entries Search:

	time	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS244
1	2016-08-29T13:28:57Z	0.0568828213879408	0.0273747604708459	0.0413564929693962	0.0624219725343321	0.0211774671749259	0.00949216
2	2016-08-29T13:29:57Z	0.0568181818181818	0.0273298715496037	0.0411692054343351	0.0619195046439629	0.0210172341319882	0.00949216
3	2016-08-29T13:30:57Z	0.0566893424036281	0.0272479564032698	0.0408496732026144	0.0602046959662854	0.020614306328592	0.009463423
4	2016-08-29T13:31:07Z	0.056657223796034	0.027196083763938	0.0407996736026112	0.0599161174355902	0.0205549845837616	0.009433962
5	2016-08-29T13:31:17Z	0.0566893424036281	0.0271665308340125	0.0407497962510187	0.0596658711217184	0.0205128205128205	0.009404683
6	2016-08-29T13:31:27Z	0.056657223796034	0.0271812992661049	0.0407000407000407	0.0594530321046373	0.0204708290685773	0.009345794
7	2016-08-29T13:31:37Z	0.056657223796034	0.0271370420624152	0.0406834825061025	0.0592768227622999	0.0204498977505112	0.009345794
8	2016-08-29T13:31:47Z	0.056657223796034	0.0271886895051658	0.0406834825061025	0.0591016548463357	0.0204123290467442	0.009345794
9	2016-08-29T13:31:57Z	0.0566251415628539	0.0271517784414879	0.0406669377795852	0.0589970501474926	0.0203873598369011	0.009287638
10	2016-08-29T13:32:07Z	0.0566251415628539	0.027122321670735	0.0406173842404549	0.0588235294117647	0.020370747606 4372	0.009287638

Showing 1 to 10 of 211,780 entries Previous 1 2 3 4 5 ... 21178 Next

Figure 4.5: Data with conductance values

The last preprocessing step, which is applied only for the sensors data, is to add the missing values with the corresponding date-time objects in the data set. This is done by clicking on the "Ok"-button and the data set including NA values appears on the right side of the interface (see Figure 4.6).



Data after adding NA rows

Show 10 entries   Search:

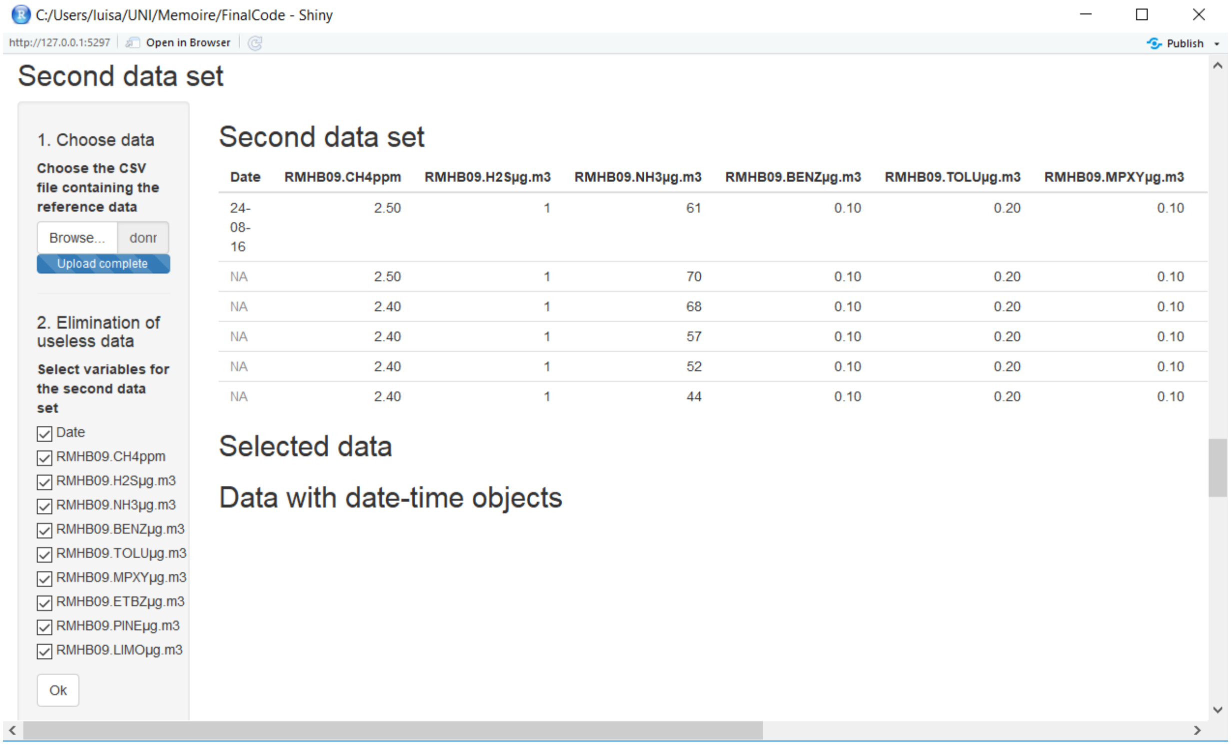
	time	TGS2602.kOhm.	TGS2610.kOhm.	TGS2611.kOhm.	TGS2620.kOhm.	GGS1330.kOhm.	TGS2
1	2016-08-29T13:28:57Z	0.0568828213879408	0.0273747604708459	0.0413564929693962	0.0624219725343321	0.0211774671749259	0.009492
2	2016-08-29T13:29:57Z	0.0568181818181818	0.0273298715496037	0.0411692054343351	0.0619195046439629	0.0210172341319882	0.009492
3	2016-08-29T13:30:57Z	0.0566893424036281	0.0272479564032698	0.0408496732026144	0.0602046959662854	0.020614306328592	0.0094634
4	2016-08-29T13:31:07Z	0.056657223796034	0.027196083763938	0.0407996736026112	0.0599161174355902	0.0205549845837616	0.0094339
5	2016-08-29T13:31:17Z	0.0566893424036281	0.0271665308340125	0.0407497962510187	0.0596658711217184	0.0205128205128205	0.0094046
6	2016-08-29T13:31:27Z	0.056657223796034	0.0271812992661049	0.0407000407000407	0.0594530321046373	0.0204708290685773	0.0093457
7	2016-08-29T13:31:37Z	0.056657223796034	0.0271370420624152	0.0406834825061025	0.0592768227622999	0.0204498977505112	0.0093457
8	2016-08-29T13:31:47Z	0.056657223796034	0.0271886895051658	0.0406834825061025	0.0591016548463357	0.0204123290467442	0.0093457
9	2016-08-29T13:31:57Z	0.0566251415628539	0.0271517784414879	0.0406669377795852	0.0589970501474926	0.0203873598369011	0.0092876
10	2016-08-29T13:32:07Z	0.0566251415628539	0.027122321670735	0.0406173842404549	0.0588235294117647	0.0203707476064372	0.0092876

Showing 1 to 10 of 222,438 entries   Previous 1 2 3 4 5 … 22244 Next

Figure 4.6: Data with missing values

Now, we pass to the preprocessing of the reference analysers data. By clicking on the "Browse"-button, a new window opens and the data set to be read can be chosen on the PC. When the upload of the data is complete, the data set is displayed as shown in Figure 4.7.

Then, the elimination of unused data is performed like for the first data set. We select the variables we want to keep, validate our choice and the selected data set appears (see Figure 4.8).



Second data set

1. Choose data

Choose the CSV file containing the reference data

Browse... donr

Upload complete

2. Elimination of useless data

Select variables for the second data set

☑ Date
☑ RMHB09.CH4ppm
☑ RMHB09.H2Sµg.m3
☑ RMHB09.NH3µg.m3
☑ RMHB09.BENZµg.m3
☑ RMHB09.TOLUµg.m3
☑ RMHB09.MPXYµg.m3
☑ RMHB09.ETBZµg.m3
☑ RMHB09.PINEµg.m3
☑ RMHB09.LIMOµg.m3

Ok

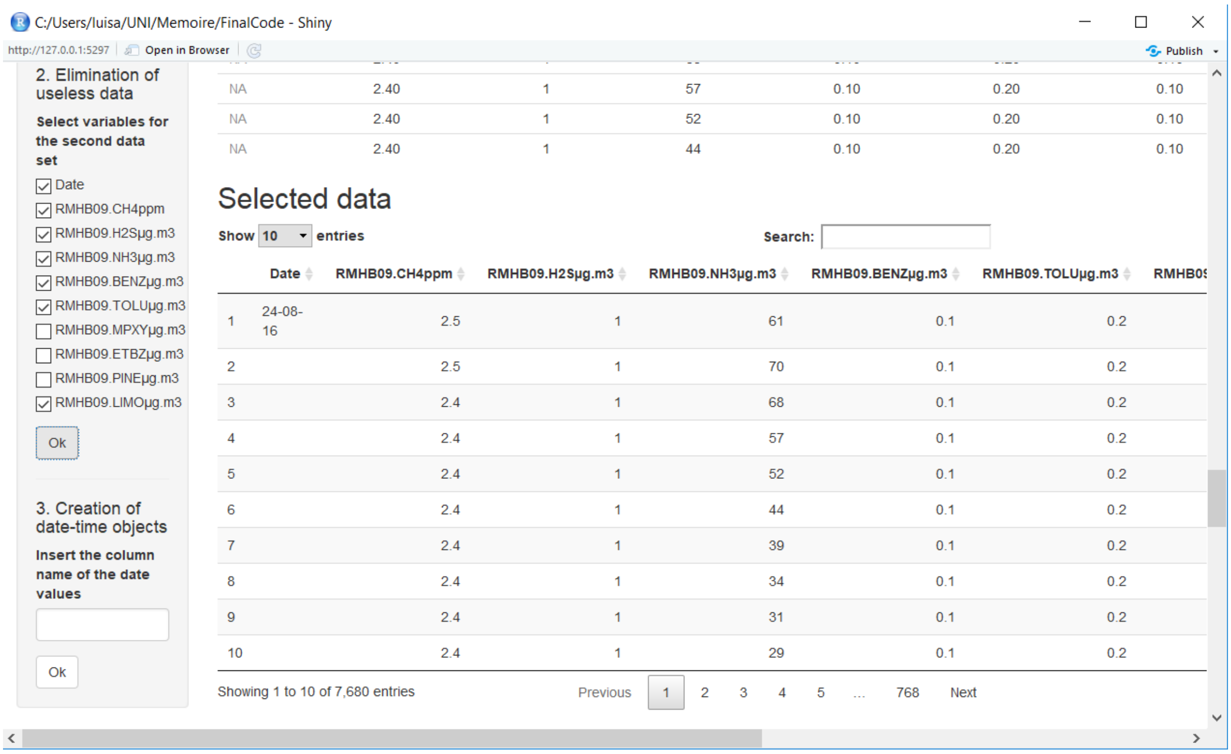
Second data set

Date	RMHB09.CH4ppm	RMHB09.H2Sµg.m3	RMHB09.NH3µg.m3	RMHB09.BENZµg.m3	RMHB09.TOLUµg.m3	RMHB09.MPXYµg.m3
24-08-16	2.50	1	61	0.10	0.20	0.10
NA	2.50	1	70	0.10	0.20	0.10
NA	2.40	1	68	0.10	0.20	0.10
NA	2.40	1	57	0.10	0.20	0.10
NA	2.40	1	52	0.10	0.20	0.10
NA	2.40	1	44	0.10	0.20	0.10

Selected data

Data with date-time objects

Figure 4.7: Upload of the second data set



2. Elimination of useless data

Select variables for the second data set

☑ Date
☑ RMHB09.CH4ppm
☑ RMHB09.H2Sµg.m3
☑ RMHB09.NH3µg.m3
☑ RMHB09.BENZµg.m3
☑ RMHB09.TOLUµg.m3
□ RMHB09.MPXYµg.m3
□ RMHB09.ETBZµg.m3
□ RMHB09.PINEµg.m3
☑ RMHB09.LIMOµg.m3

Ok

3. Creation of date-time objects

Insert the column name of the date values

Ok

NA	2.40	1	57	0.10	0.20	0.10
NA	2.40	1	52	0.10	0.20	0.10
NA	2.40	1	44	0.10	0.20	0.10

Selected data

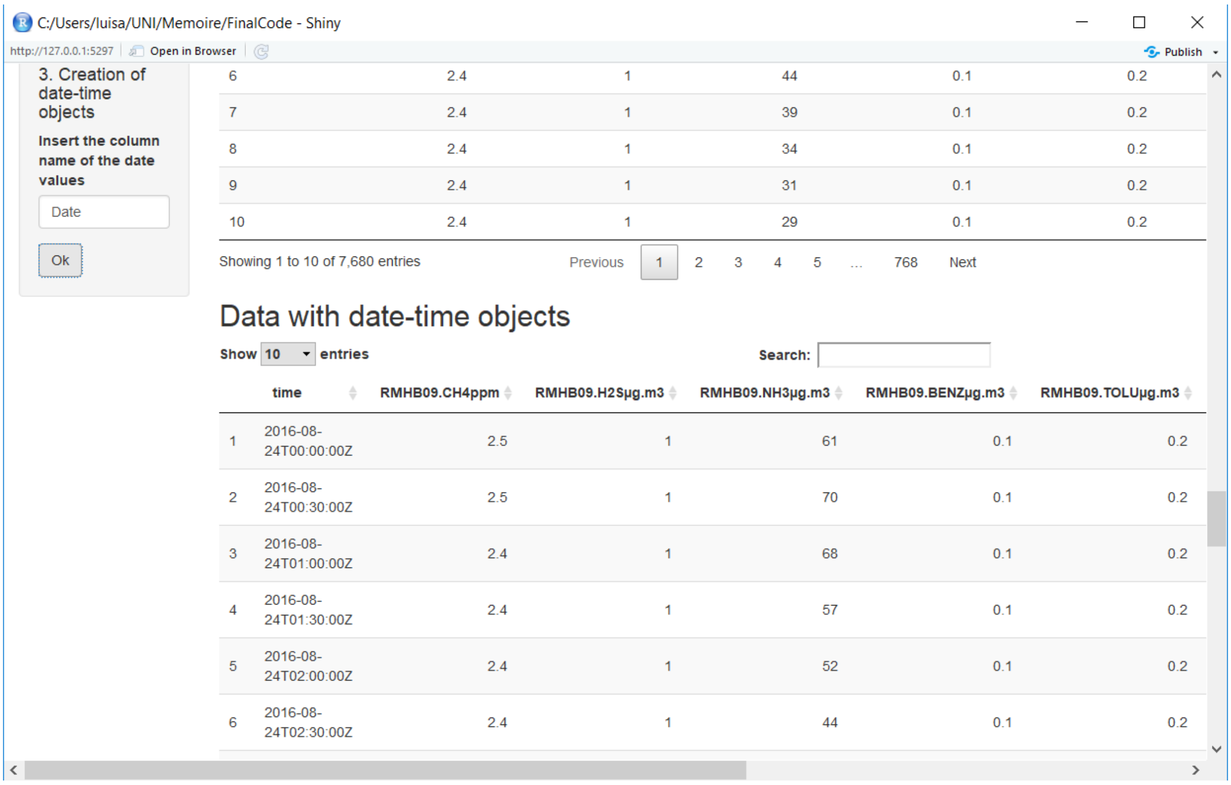
Show 10 entries   Search:

	Date	RMHB09.CH4ppm	RMHB09.H2Sµg.m3	RMHB09.NH3µg.m3	RMHB09.BENZµg.m3	RMHB09.TOLUµg.m3	RMHB09
1	24-08-16	2.5	1	61	0.1	0.2	
2		2.5	1	70	0.1	0.2	
3		2.4	1	68	0.1	0.2	
4		2.4	1	57	0.1	0.2	
5		2.4	1	52	0.1	0.2	
6		2.4	1	44	0.1	0.2	
7		2.4	1	39	0.1	0.2	
8		2.4	1	34	0.1	0.2	
9		2.4	1	31	0.1	0.2	
10		2.4	1	29	0.1	0.2	

Showing 1 to 10 of 7,680 entries

Figure 4.8: Elimination in the second data set

For the creation of the date-time objects in the second data set, we have to insert the name of the column containing the date values, e.g. Date. By clicking the "Ok"-button, the new data set including date-time objects is represented like illustrated in Figure 4.9.



3. Creation of date-time objects

Insert the column name of the date values

Date

Ok

6	2.4	1	44	0.1	0.2
7	2.4	1	39	0.1	0.2
8	2.4	1	34	0.1	0.2
9	2.4	1	31	0.1	0.2
10	2.4	1	29	0.1	0.2

Showing 1 to 10 of 7,680 entries

Previous 1 2 3 4 5 ... 768 Next

Data with date-time objects

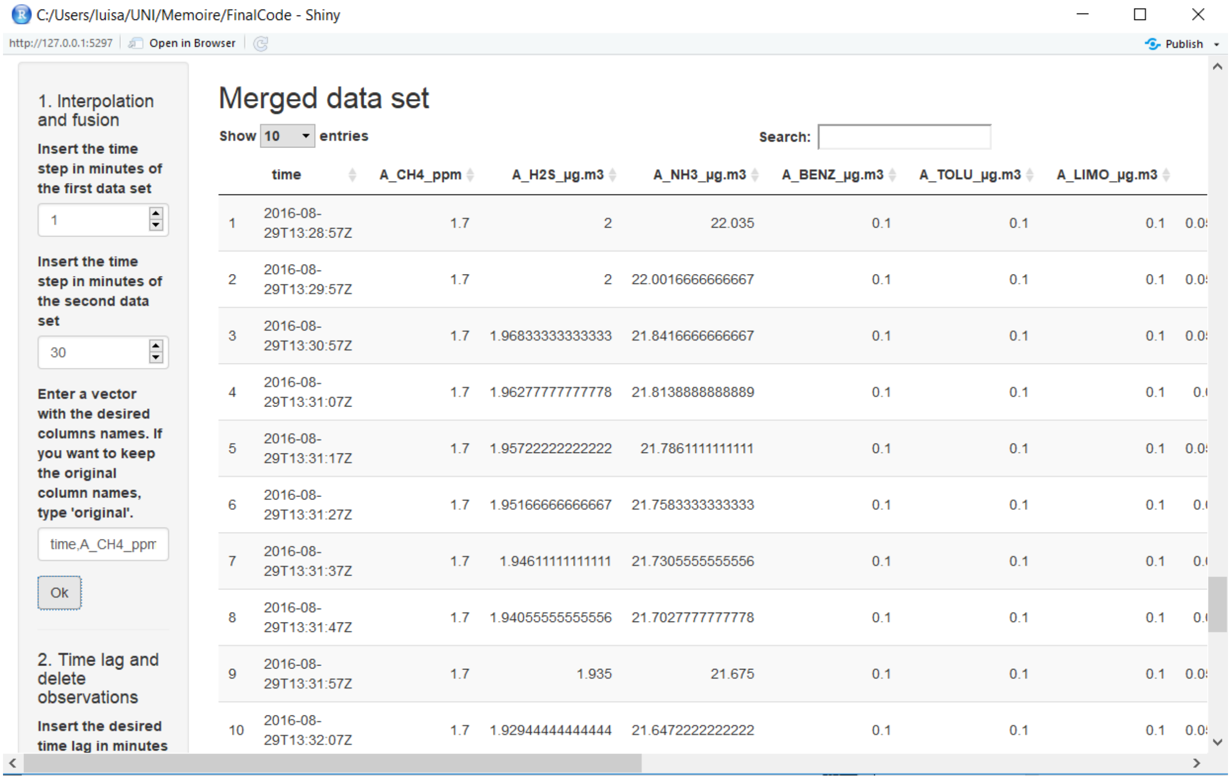
Show 10 entries

Search:

	time	RMHB09.CH4ppm	RMHB09.H2Sµg.m3	RMHB09.NH3µg.m3	RMHB09.BENZµg.m3	RMHB09.TOLUµg.m3
1	2016-08-24T00:00:00Z	2.5	1	61	0.1	0.2
2	2016-08-24T00:30:00Z	2.5	1	70	0.1	0.2
3	2016-08-24T01:00:00Z	2.4	1	68	0.1	0.2
4	2016-08-24T01:30:00Z	2.4	1	57	0.1	0.2
5	2016-08-24T02:00:00Z	2.4	1	52	0.1	0.2
6	2016-08-24T02:30:00Z	2.4	1	44	0.1	0.2

Figure 4.9: Date-time objects in the second data set

The next step is to merge the two data sets. Therefore, the preprocessing executes first a linear interpolation on the data set with the higher time lag, then the approximated and the other unchanged data set are merged. So, we have to input the time lag in minutes for the two data sets. It is also demanded to enter desired column names which have to be separated by commas and without quotation marks. An example for our data is to enter: time, A_CH4_ppm, A_H2S_μg.m3, A_NH3_μg.m3, A_BENZ_μg.m3, A_TOLU_μg.m3, A_LIMO_μg.m3, S_2602_kOhm, S_2610_kOhm, S_2611_kOhm, S_2620_kOhm,S_1330_kOhm, S_2444_kOhm. If, for any reason, we do not want to name the columns different, we type "original" and the original column names are retained. Then, we click on the "Ok"-button to execute the interpolation and fusion (see Figure 4.10).



1. Interpolation and fusion

Insert the time step in minutes of the first data set: 1

Insert the time step in minutes of the second data set: 30

Enter a vector with the desired columns names. If you want to keep the original column names, type 'original'.: time,A_CH4_ppm

2. Time lag and delete observations

Insert the desired time lag in minutes

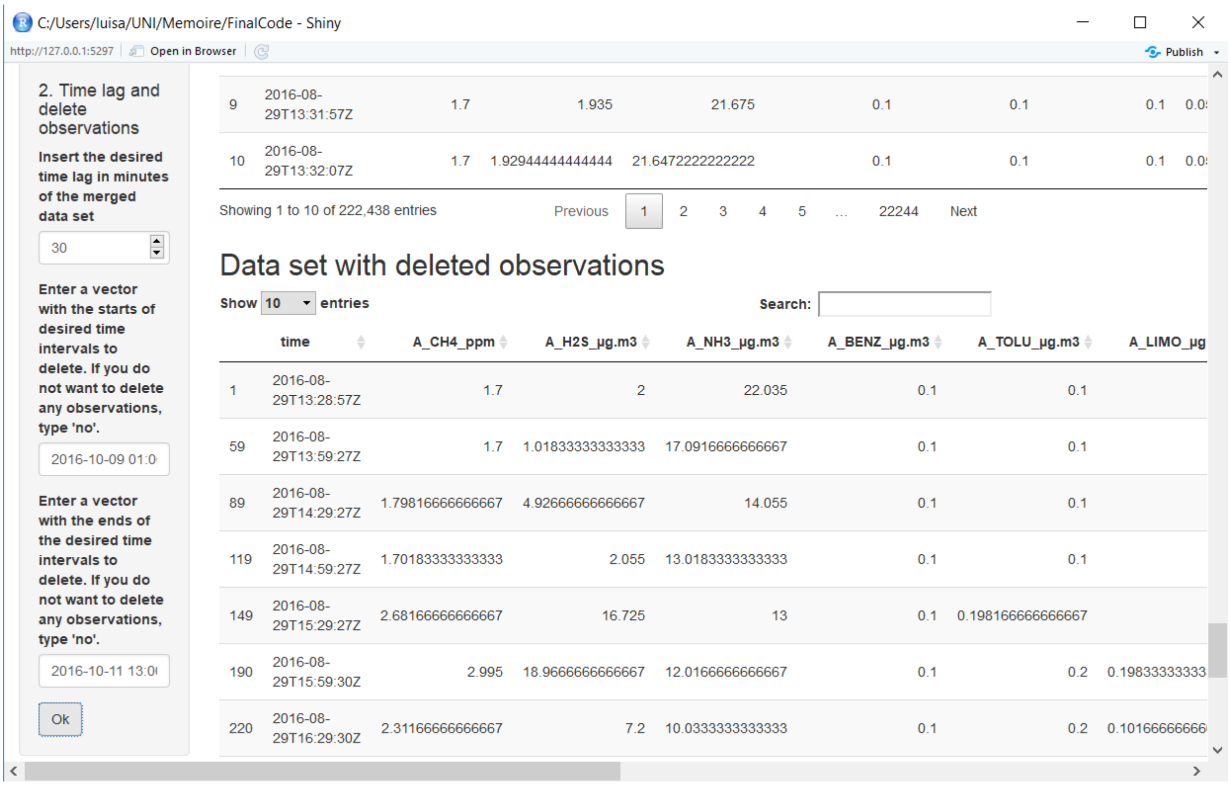
Merged data set

Show 10 entries — Search:

	time	A_CH4_ppm	A_H2S_µg.m3	A_NH3_µg.m3	A_BENZ_µg.m3	A_TOLU_µg.m3	A_LIMO_µg.m3	
1	2016-08-29T13:28:57Z	1.7	2	22.035	0.1	0.1	0.1	0.0
2	2016-08-29T13:29:57Z	1.7	2	22.0016666666667	0.1	0.1	0.1	0.0
3	2016-08-29T13:30:57Z	1.7	1.96833333333333	21.8416666666667	0.1	0.1	0.1	0.0
4	2016-08-29T13:31:07Z	1.7	1.96277777777778	21.8138888888889	0.1	0.1	0.1	0.
5	2016-08-29T13:31:17Z	1.7	1.95722222222222	21.7861111111111	0.1	0.1	0.1	0.0
6	2016-08-29T13:31:27Z	1.7	1.95166666666667	21.7583333333333	0.1	0.1	0.1	0.
7	2016-08-29T13:31:37Z	1.7	1.94611111111111	21.7305555555556	0.1	0.1	0.1	0.
8	2016-08-29T13:31:47Z	1.7	1.94055555555556	21.7027777777778	0.1	0.1	0.1	0.
9	2016-08-29T13:31:57Z	1.7	1.935	21.675	0.1	0.1	0.1	0.0
10	2016-08-29T13:32:07Z	1.7	1.92944444444444	21.6472222222222	0.1	0.1	0.1	0.0

Figure 4.10: Interpolation and merge of the data sets

The last part of the preprocessing consists in the choice of a higher time lag and the deletion of observations. Therefore, we insert the desired time lag in minutes and the start and ending times of the time intervals we want to delete. These vectors must have the same length, being separated by commas and all elements are of the format "year-month-day hour:minute:second". They do not require quotation marks. An example could be 2016-10-09 01:00:00,2016-10-27 00:00:00 as starting vector and 2016-10-11 13:00:00,2016-10-27 23:59:59 as ending vector. If we do not want to delete any observations, enter "no" in the two cases. The resulting data set is shown on Figure 4.11.



2. Time lag and delete observations

Insert the desired time lag in minutes of the merged data set

30

Enter a vector with the starts of desired time intervals to delete. If you do not want to delete any observations, type 'no'.

2016-10-09 01:0

Enter a vector with the ends of the desired time intervals to delete. If you do not want to delete any observations, type 'no'.

2016-10-11 13:0

9	2016-08-29T13:31:57Z	1.7	1.935	21.675	0.1	0.1	0.1	0.0
10	2016-08-29T13:32:07Z	1.7	1.92944444444444	21.6472222222222	0.1	0.1	0.1	0.0

Showing 1 to 10 of 222,438 entries   Previous 1 2 3 4 5 … 22244 Next

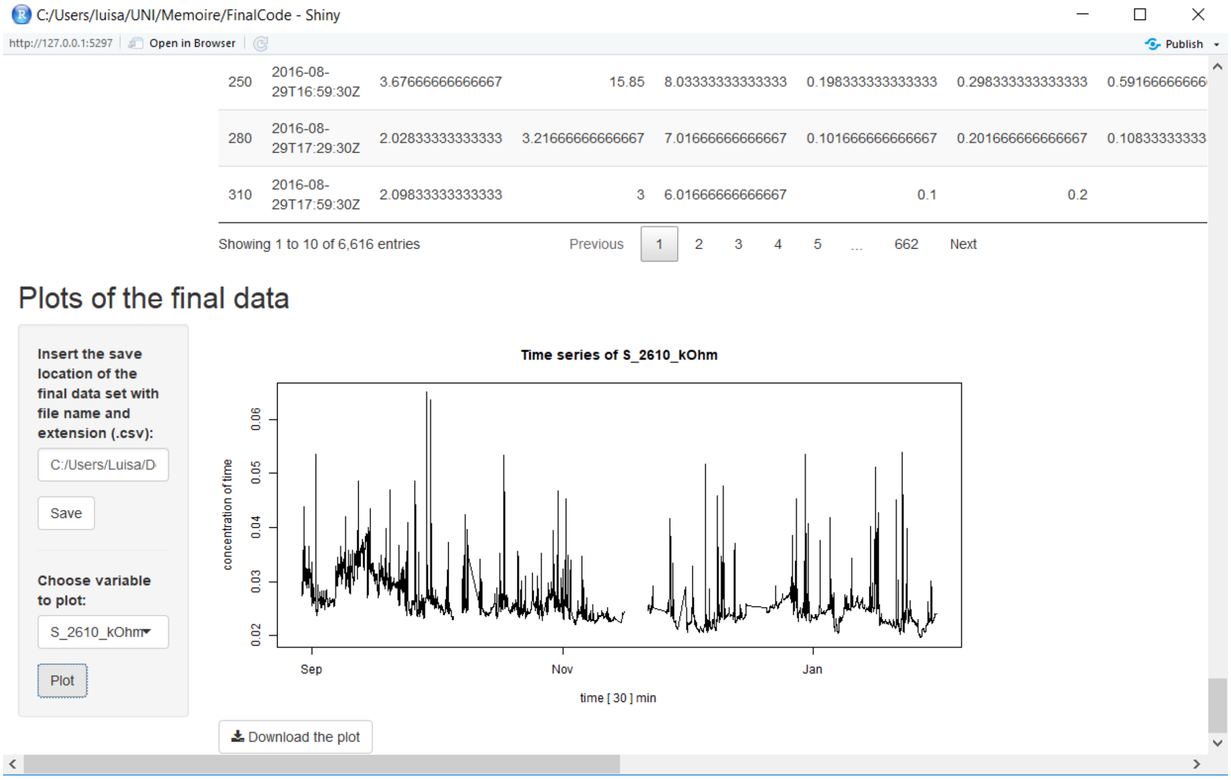
Data set with deleted observations

Show 10 entries   Search:

	time	A_CH4_ppm	A_H2S_µg.m3	A_NH3_µg.m3	A_BENZ_µg.m3	A_TOLU_µg.m3	A_LIMO_µg
1	2016-08-29T13:28:57Z	1.7	2	22.035	0.1	0.1	
59	2016-08-29T13:59:27Z	1.7	1.01833333333333	17.0916666666667	0.1	0.1	
89	2016-08-29T14:29:27Z	1.79816666666667	4.92666666666667	14.055	0.1	0.1	
119	2016-08-29T14:59:27Z	1.70183333333333	2.055	13.0183333333333	0.1	0.1	
149	2016-08-29T15:29:27Z	2.68166666666667	16.725	13	0.1	0.198166666666667	
190	2016-08-29T15:59:30Z	2.995	18.9666666666667	12.0166666666667	0.1	0.2	0.19833333333
220	2016-08-29T16:29:30Z	2.31166666666667	7.2	10.0333333333333	0.1	0.2	0.10166666666

Figure 4.11: Time lag increase and deletion of observations

Now, the preprocessing is finished and we have the possibility to save the final data set and to save plots of the variables (see Figure 4.12). Therefore, we enter the save location with forward slashes, without quotation marks and with the extension .csv, e.g. C:/Users/User/folder/final_data.csv. By clicking on the "Save"-button, the data set is saved on the specified location on the PC. Finally, the user can choose a variable of the data set to plot. By clicking the "Plot"-button, the time series with the selected measurements appears on the right side. The "Download"-button enables to save this plot under PDF-format by choosing the direction and the name of the plot in an opened window. Naturally, another variable can be selected and illustrated before saving the plot.



C:/Users/luisa/UNI/Memoire/FinalCode - Shiny

http://127.0.0.1:5297

250	2016-08-29T16:59:30Z	3.67666666666667	15.85	8.03333333333333	0.198333333333333	0.298333333333333	0.59166666666
280	2016-08-29T17:29:30Z	2.02833333333333	3.21666666666667	7.01666666666667	0.101666666666667	0.201666666666667	0.10833333333
310	2016-08-29T17:59:30Z	2.09833333333333	3	6.01666666666667	0.1	0.2	

Showing 1 to 10 of 6,616 entries

Plots of the final data

Insert the save location of the final data set with file name and extension (.csv): C:/Users/Luisa/D

Choose variable to plot: S_2610_kOhm

Figure 4.12: Plot and download of the time series

This interface is also useful for the preprocessing of the control parameters. The only thing to mind is to not select any variable for the conductance transformation, because it makes no sense to inverse a control parameter.

A second version of this application has been written for the case when a reference data set is not available. Then, only the preprocessing of the sensors data is necessary. And instead of changing the time lag and deleting the observation of the fusion, we execute these functions on the first final data set.

Chapter 5

Predictive model

In this chapter, we present the models created to predict the presence of different chemical compounds in the air. Therefore, an adjusted model is established for every analyser separately. We first present the modelling approach we developed. A model is then fitted for each pollutant separately. A diagnostic analysis of the performances of the model is provided for each targeted pollutant. Several aspects will be discussed, like the robustness and the performances of the models. Finally, we identify the contribution of every sensor in the prediction of the chemical air components. The R script for the application of the predictive models is available on the MatheO platform.

5.1 Statistical linear modelling

The main objective of this study is to find a prediction model that explains the presence of chemical components in the air - as detected by the analysers - as a function of the signals returned by the sensors. Remember that the signal is expected to be located in the combination of several sensors (see Section 3.2.7). The first step is to perform a linear model before passing to a more complex prediction model. Linear models are well known very performing [11]. We examine the contributions of each sensor with the linear model. This analysis is aimed to serve as a basis for a further study, in which more complex modelling approaches could be implemented in order to get better quality predictions.

5.1.1 Response analysers variables

For each chemical component in the study, the measurements returned by the dedicated analyser are used as proxies for the true concentrations of the component in the air. As a reminder, ISSEP's analysers are certified instruments. As a result, every analysers signal serves as a response variable for a distinct model.

Logarithm of analysers values By performing a MLR model, we state a multiplicative error term in the prediction. To avoid this, we choose to work with the logarithm of the analysers values in the subsequent prediction. The following Figure 5.1 represents the measurements of the CH_4 analyser on the left, and its transformation to the logarithm on the right.

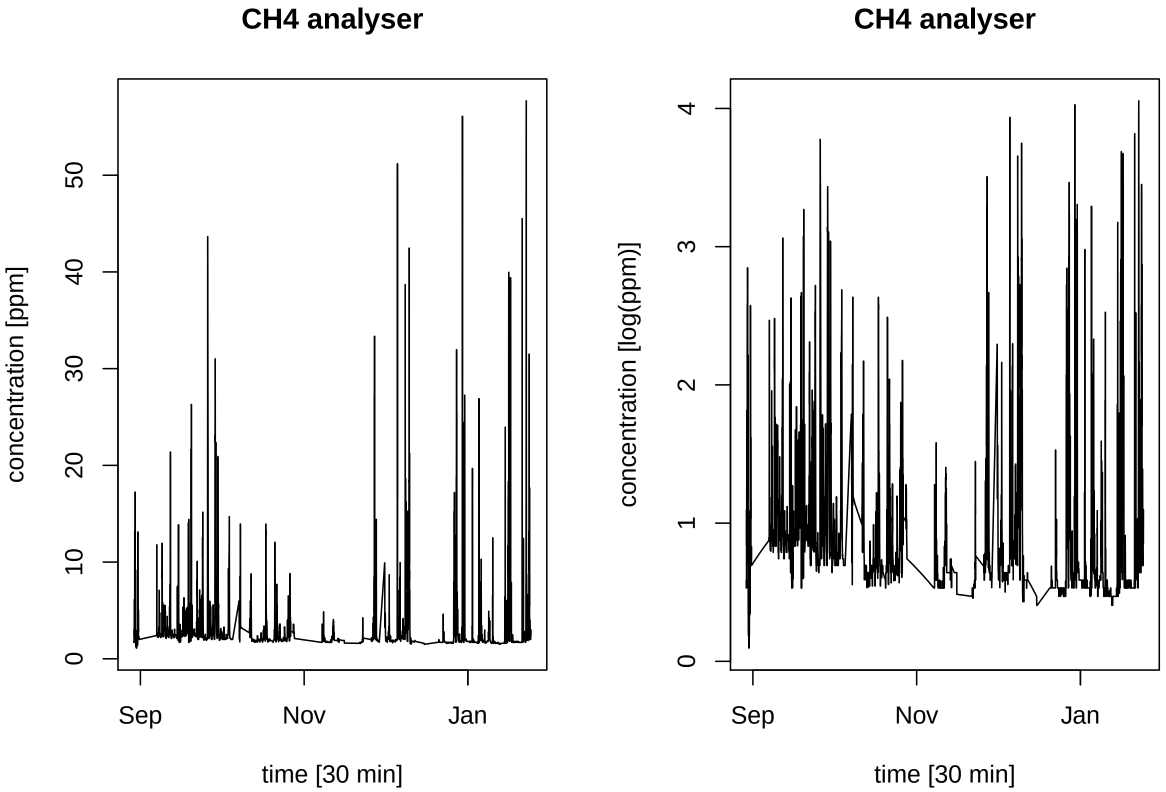


Figure 5.1: Logarithmic transformation of the CH4 analysers measurements

Henceforth, the six response variables will be named as follows: $A_{\mathrm{CH_4}}$, $A_{\mathrm{H_2S}}$, $A_{\mathrm{NH_3}}$, A_{BENZ}, A_{TOLU}, A_{LIMO}.

5.1.2 Explanatory sensors variables

We want the sensors to predict the chemical components in the air. In Chapter 3, we perform a data pretreatment for the further analysis. To handle the collinearity we execute a principal component analysis on all interactions from order 1 to 6. After this, we continue to work with the 63 principal components from this analysis instead of the precedent interaction variables. We name this 63 principal components PC_1,...,PC_{63}.

5.1.3 Time independence

The atmospheric state, and henceforth the concentration of pollutants in the air, evolves continuously in time. This time dependency affects the true concentrations of the chemical compounds in the air, and thus it affects the target values that are estimated by both, the analysers and the sensors. Accounting for this time dependency between the target values would suggest that we could improve the prediction of a concentration at time t+1 by knowing the (estimated) concentrations at times t, t-1, t-2... This is a perspective for a further study.

In the present study, we do not focus on the time evolution of the concentrations of air pollutants, but rather on the link between the measurements provided by the sensors and the analysers. Therefore, we are concerned about the time independence of the measurements errors affecting the devices. For this reason, we decided to restrict the model calibration on data collected every 30 minutes. In this way, we can consider that there is no "instrumental memory" between two successive measurements and that their correlation is only related to the fact that the targeted value is similar. A perspective for a more advanced further study could be the increase of the time frequency and the adaptation of the model to account for the auto-correlation of the measurements.

5.1.4 Multiple linear regression

First, we tried out a multiple linear regression (MLR) for every analyser variable separately by taking all the principal components as explanatory variables. Unfortunately, the error terms do not follow a normal distribution for all analysers. All QQ-plots have approximatively the same form as shown in Figure 5.2 and the Kolmogorov-Smirnov Test with all p-values less than $2.2 \cdot 10^{-16}$ also confirms the non-normal distribution of the error terms.

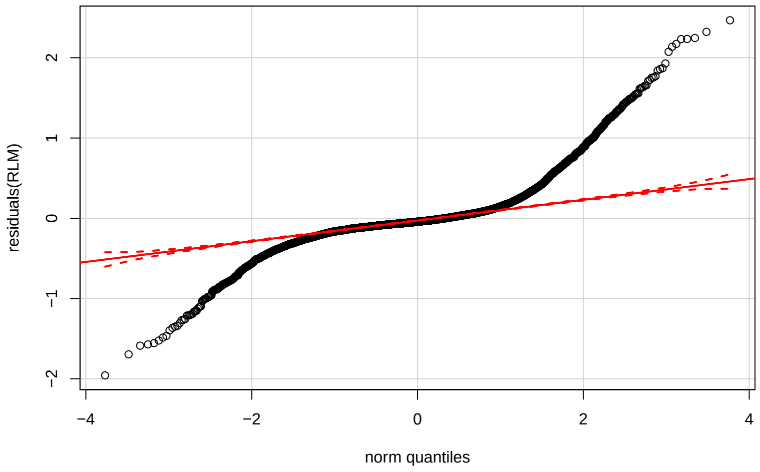


Figure 5.2: QQ-plot of the residuals for the CH_4 analyser

The hypotheses of normality and homoscedasticity of the residual terms not being respected, the usual procedures used for the linear regression inference are not applicable. The quantile regression model, making it possible to overcome normality assumptions and homoscedasticity of the residuals to predict the median of concentrations on terms of explanatory variables would probably provide an adequate setting for regression on these data.

5.1.5 Creation of binary analysers variables

To concentrate only on the signals of the analysers measurements, we dichotomise the values of the response variables separately into `signal` and `noise`. Therefore, an individual threshold for every analyser will be defined. Then, the value "signal" is associated to the values exceeding the threshold, and "noise" to the values beneath.

Let's take the analyser A_{CH_4} to show the dichotomisation in detail. The next figure Figure 5.3 illustrates the real concentration measurements of this analyser. The red horizontal line represents the chosen threshold equal to 5 *ppm* for CH_4. So all measurements greater or equal to this threshold will define a signal, those less than 5 *ppm* define a noise. We obtain a discrete binary variable containing `signal` or `noise` defined by the selected threshold. For this example, the CH_4 analyser with threshold equal to 5 *ppm* corresponds to 493 signals and 5917 noise values.

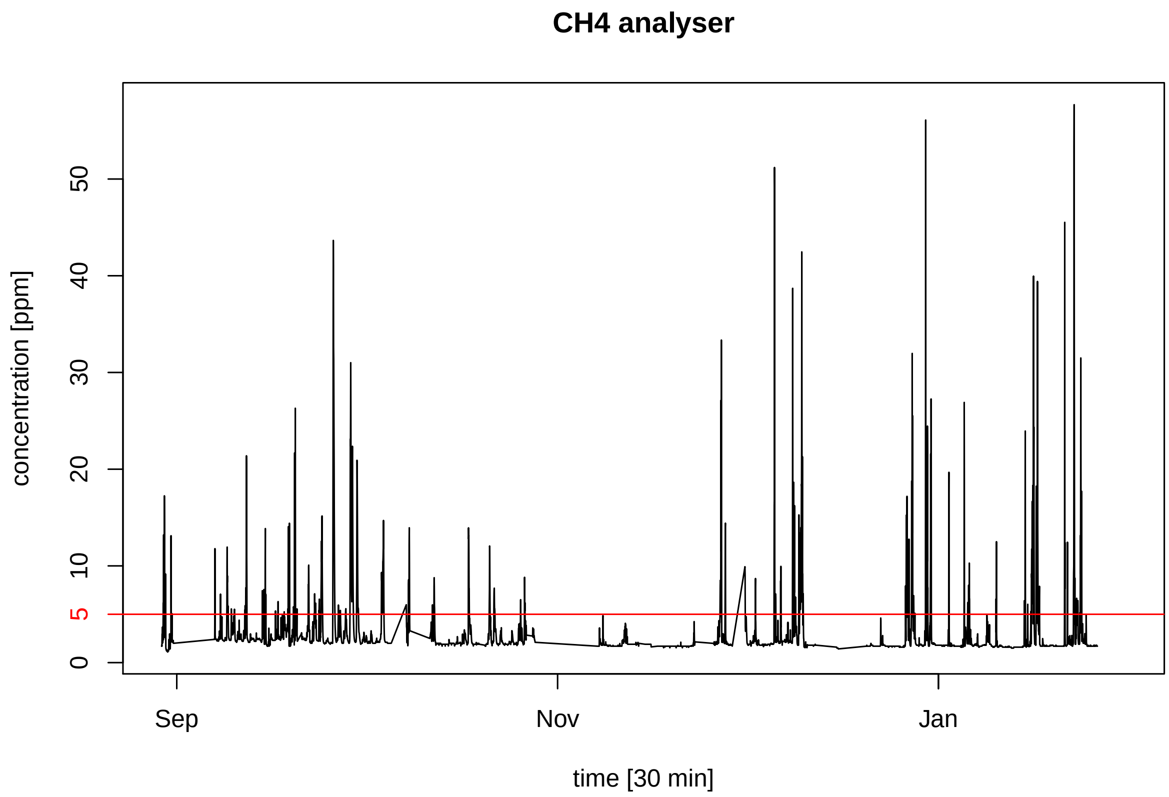


Figure 5.3: Measurements of A_{CH_4}

5.1.6 Linear discriminant analysis

We created binary response variables, so we are in case of a classification problem. One aim of the linear discriminant analysis (LDA) is to determine if the variables make it possible to discriminate groups, here `signal` and `noise`. Then, the explanatory variables are called *discriminant*. In this case, the second aim is to classify new observations in one group.

In general, we dispose of n observations explained by p explanatory variables. Suppose that we have q classes, which are defined by the modalities of the response variables. So, in the case of a binary response like in our study, the response variables define 2 classes. The linear discriminant analysis then, is divided in two steps. The first step consists in a search of linear discriminant functions on a learning or training sample. These functions are linear combinations of the p explanatory variables separating best possibly the classes. Afterwards, the second step proceeds the classification of new observations in the classes given the explanatory variables of these observations.

The choice of the linear combinations is determined by the minimisation of the variance within the groups and the maximisation of the variance between the groups. With this criterion, the difference between the classes is enforced and simultaneously the variation range is delimited. Then, the next linear combination which is not correlated with the first one and which discriminates the best the groups is chosen and so on.

Given that $\mathbf{X}$ is the data set of n observations (rows) and p variables (columns). Suppose that $\mathbf{a}$ defines a linear combination of the p centred explanatory variables and take an individual i:

$$a(i) = \sum_{j=1}^{p} a_j(x_{ij} - \bar{x}_j)$$

where $\bar{x}_j$ defines the j-th coordinate of the center of gravity G of the individuals.
As $a(i)$ is centred, the variance of $a(i)$ is given by:

$$\begin{aligned} var(\mathbf{a}) &= \frac{1}{n}\sum_{i=1}^{n} a^2(i) \\ &= \frac{1}{n}\sum_{i=1}^{n}(\sum_{j=1}^{p} a_j(x_{ij} - \bar{x}_j))^2 \\ &= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p}\sum_{j'=1}^{p} a_j a_{j'}(x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'}) \\ &= \sum_{j=1}^{p}\sum_{j'=1}^{p} a_j a_{j'} cov(x_j, x_{j'}) \\ &= \mathbf{a}'\mathbf{S}\mathbf{a} \end{aligned}$$

The last but one equality is found by inversion of the sums and by the definition of the covariance matrix. For the last equality, we pose $\mathbf{S}$ the covariance matrix of the p variables.

By the decomposition of König-Huygens, we have

$$\mathbf{S} = \mathbf{W} + \mathbf{B}$$

where $\mathbf{W}$ represents the covariance within groups and $\mathbf{B}$ the covariance between groups.

Therefore, the variance of the linear combination $\mathbf{a}$ is decomposed into the sum of the within variance and the between variance:

$$\mathbf{a}'\mathbf{S}\mathbf{a} = \mathbf{a}'\mathbf{W}\mathbf{a} + \mathbf{a}'\mathbf{B}\mathbf{a}. \ (1)$$

Remember that we search $\mathbf{a}$ such that the variance within groups is minimal and the one between the groups is maximal, thus we want to maximise the ratio $\dfrac{\mathbf{a'Ba}}{\mathbf{a'Wa}}$. To search the maximum of this ratio, we search the stationary points of the Lagrangian L with $\|\mathbf{a}\| = 1$ as constraint [1]:

$$L(\mathbf{a}, \lambda) = \frac{\mathbf{a'Ba}}{\mathbf{a'Wa}} - \lambda(\mathbf{a'a} - 1)$$

$$L(\mathbf{a}, \lambda) \text{ maximal} \Leftrightarrow \begin{cases} \dfrac{\partial L}{\partial \mathbf{a}} = 0 \\ \\ \dfrac{\partial L}{\partial \lambda} = 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} \mathbf{Ba} = \lambda \mathbf{Wa} \\ \mathbf{a'a} = 1 \end{cases}$$

$$\Leftrightarrow \mathbf{W}^{-1}\mathbf{Ba} = \lambda \mathbf{a}$$

So we obtain that $\mathbf{a}$ is an eigenvector of $\mathbf{W}^{-1}\mathbf{B}$ associated to the greatest eigenvalue value λ of the matrix $\mathbf{W}^{-1}\mathbf{B}$. The next linear combination is chosen to be the eigenvector of $\mathbf{W}^{-1}\mathbf{B}$ to the second greatest eigenvalue value λ of $\mathbf{W}^{-1}\mathbf{B}$ and so on.

Hypotheses Before discrimination into groups, it is usually suggested to test if the explanatory variables are able to separate in two groups. Therefore, a test of mean comparison is adequate:

$$H_0 : \mu_1 = \mu 2 \longleftrightarrow H_1 : \mu_1 \neq \mu 2$$

If the null hypothesis is not rejected, it does not make much sense to continue the analysis. We have to notice that a normal multivariate distribution of the explanatory variables, that is the 63 principal components and their homoscedasticity, should be verified if we want to test for these hypotheses using a Hotelling statictics.

Discussion

We do not apply this test of mean comparison because of the non-multi normal distribution of the explanatory variables, which is necessary to apply this test. To verify that the

[1] We can suppose that $\|\mathbf{a}\| = 1 \Leftrightarrow \mathbf{a'a} = 1$ because a continuous function reaches its maximum on a compact.

model discriminates well signals and noises, we perform a cross-validation and observe the prediction capacity on validation sets.

5.1.7 Model selection

A LDA prediction model for a binary response variable `signal/noise` has several outputs: a confusion matrix, sensitivity, specificity, accuracy and the ROC curve. A confusion matrix is represented on Table 5.1.

Table 5.1: Definition of the confusion matrix

		Reference	
		noise	signal
Prediction	noise	TN	FN
	signal	FP	TP

The confusion matrix contains the following information:

- TN: number of true negatives
- FN: number of false negatives
- FP: number of false positives
- TP: number of true positives

The sensitivity of a prediction is defined as the true positive rate equal to $\frac{TP}{FN+TP}$, the specificity as the true negative rate equal to $\frac{TN}{TN+FP}$. The accuracy gives the true prediction rate which is $\frac{TN+TP}{TN+TP+FN+FP}$. The ROC curve is a plot representing the sensitivity in terms of 1-specificity. The optimal curve would have the maximal area under the curve, which represents the accuracy of the prediction equal to 1 [7].

In the LDA model, we proceed a model selection by the aid of the `stepclass()` R-function of the `klaR` R-package [15]. This function selects by estimating a classification performance measure. In our study, we choose to select under the criterion "accuracy" with an improvement of 0.001 in the forward direction. It means that a variable is added to the prediction model if the accuracy increases by at least 0.001 when the model is enlarged. If there is no more variable which increases the accuracy by at least 0.001, the algorithm stops and the selection is finished.

5.1.8 Model diagnosis

Analysis of the LDA model In the model diagnosis for every analyser separately, we discuss about the results of the LDA model over the complete data set. The evolution of the accuracy is represented in terms of the model selection and the number of detected events is analysed.

Comparison with the MLR model We execute a MLR model with the selected variables from the LDA model for an a additional comparison of the two models. We also represent the evolution of the adjusted R squares in terms of the selected variables.

Cross-validation Afterwards, a cross-validation with random split is performed. In this section, a model is produced twenty times by splitting randomly the data set into a training and validation set (80% - 20% respectively). The principal component analysis is executed on the training set before applying the LDA model and the selection. It is important to remark that the principal components are reconstructed for every new training set. Then, the minimum, mean and maximum of the sensitivities, specificities and also the adjusted R squares of the selected models are interpreted. The twenty ROC curves are superposed for a supplementary information.

Prediction for the time series When we split the data set according to the sequence and not randomly, we interpret the ability of the prediction model for the near future. However, for three of the six analysers, namely H_2S, NH_3 and LIMO, the variation range decreases drastically in the time period from November to January. Therefore, a prediction in the future for these analysers is not relevant because of the non-existence of signal in the final 20% of the series.

Influence of the control parameters in the prediction The residual terms of the MLR model with the selected variables from the LDA model are represented in terms of the control parameters to show their possible influence on the prediction model. Possible reasons for false predictions can be concluded in this section.

Which interactions of sensors contribute to the prediction? Remember that the main objective of this study is to find a prediction model that explains the presence of chemical components in the air - as detected by the analysers - as a function of the signals returned by the sensors. Therefore, we consider the LDA model over the complete data set without separation in training and test set and we want to analyse which interactions of sensors contribute in the predictions.

5.2 Model diagnosis for the CH_4 analyser

The threshold for methane signals has been chosen to 5 ppm, based on the values in this study.

5.2.1 Fitting linear discriminant analysis model over the complete data set

Selected principal components The selection in the LDA model results in twelve principal components, which are represented on Figure 5.4 with the evolution of the accuracy. We see that with one principal component, an accuracy of 0.77 is already reached. The addition of the next variables increases the accuracy a bit, but then it remains nearly stable around 0.87.

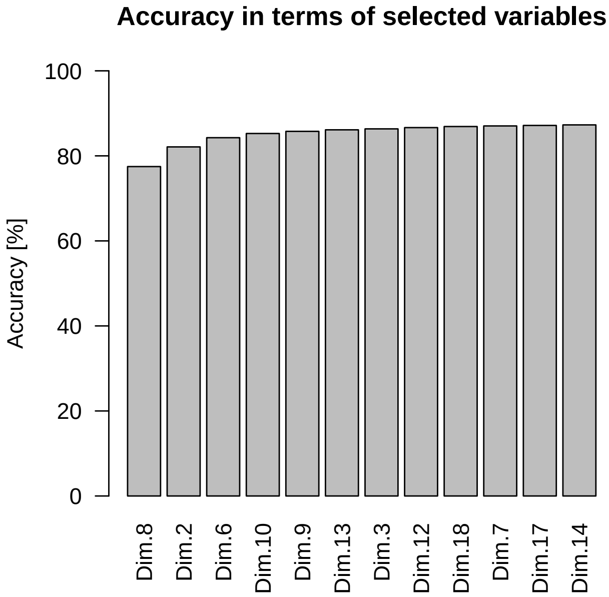


Figure 5.4: Accuracy evolution in the model selection

Prediction of signals and pollution events The predicted signals are represented in Figure 5.5 on the real CH_4 analyser's time series. The confusion matrix of this predictions is shown in Table 5.2.

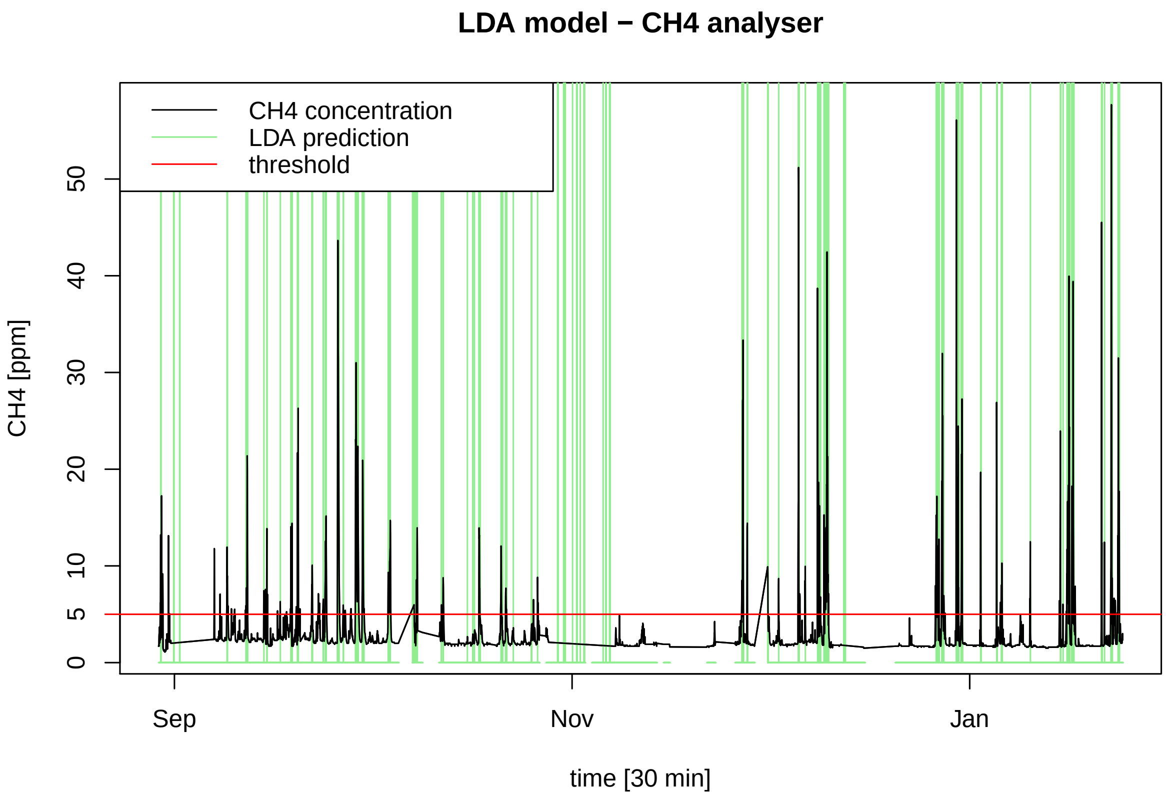


Figure 5.5: CH_4 prediction with LDA model

Table 5.2: Confusion matrix of the LDA prediction for the complete data set

		Reference	
		noise	signal
Prediction	noise	5397	248
	signal	105	229

We see that 248 out of 477 signals and only 105 out of 5502 noise values have been wrong predicted. This results in a sensitivity of 0.48, a specificity of around 0.98, a false positive rate of 0.019 and a false negative rate of 0.52. The false negative rate is relatively high. Nevertheless, when we go back to Figure 5.5, we observe that all the important peaks are nearly always detected at least once. The model does not detect the entirety of exact signals but the pollution events are most of time captured. We count the number of detected and non detected events (Table 5.3) and the number of true and false predicted events (Table 5.4).

Table 5.3: Detection of pollution events

	Detected	Non detected	Total
Pollution events	64	17	81

Table 5.4: Predicted pollution events

	True	False	Total
Predicted events	63	18	81

The 17 non detected events corresponds to CH_4 concentrations beneath 15 ppm and among the 17, there are 11 events with weak concentrations under 10 ppm. Concerning the false predicted events, 13 out of 18 false predicted events occur at moments, when the CH_4 analyser did not work. The prediction takes place on the interpolated values, so it is not excluded that a pollution event actually occurred at that time.

False positive and negative predictions In Figure 5.6 the false positive and negative predictions are represented in red and green respectively. We see as before in the confusion matrix, that there are much more false negative signals than false positive ones. This effect underlies also the high number of noise values in the data set and the more weak signals in the CH_4 values. We can also observe that the most false positive predictions are more concentrated near to the threshold. The figure concerning the false predictions confirms the high false negative rate. Nevertheless, we see that the important peaks in the CH_4 curve are nearly always detected at least once. In the high CH_4 concentrations there are always right positive predictions (in black), like already observed in the precedent tables. Some gaps are visible on this figure in October, November and December and originate from the missing values in the sensors data. Either the observations are missing because the sensors did not work or because of the deletion when odour bags were connected.

ROC curve The ROC curve in Figure 5.7 confirms the results already discussed about the confusion matrix. Although the true positive rate (sensitivity) is not so strong, the curve behaves not so bad. Remember that the optimal curve would have the maximal area under the curve, which represents the accuracy of the prediction equal to 1. This would be the case when the sensitivity equals to 1 and the specificity to 1. For the CH_4 prediction, the area under the curve equals 0.88.

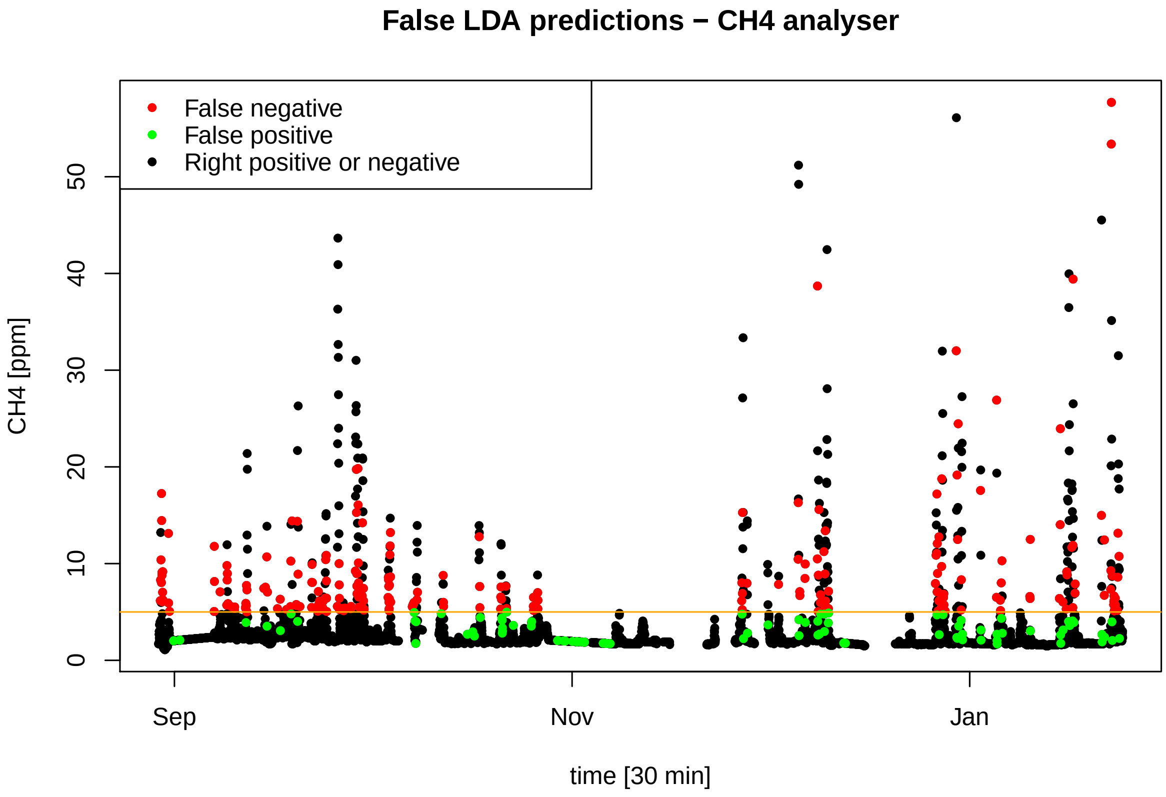


Figure 5.6: False predictions in the LDA model

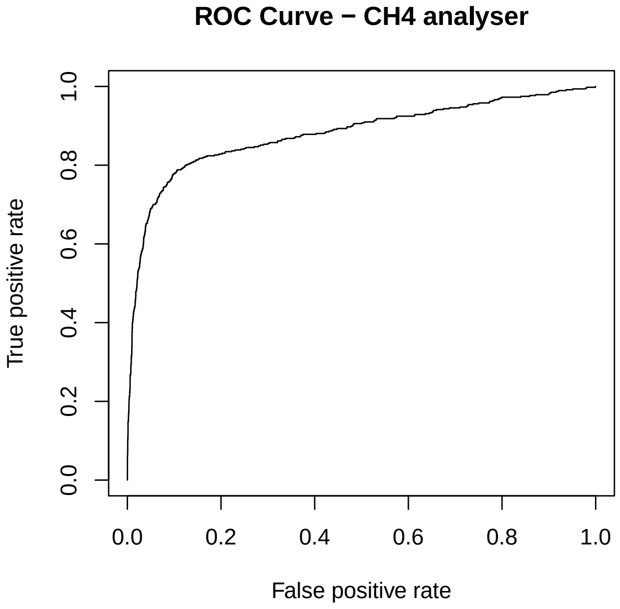


Figure 5.7: ROC curve of the LDA model over the complete data set

5.2.2 Comparison of the LDA and MLR model predictions

In the following Figure 5.8, the predictions of the LDA and MLR model are superposed. We observe that the LDA and MLR model provide very similar predictions. When a signal has not been detected by the LDA model, the MLR predictions are underestimating the true concentration as well.

The evolution of the adjusted R squared is illustrated in Figure 5.9. We observe an increase from circa 0.1 up to 0.39 of the adjusted R squared.

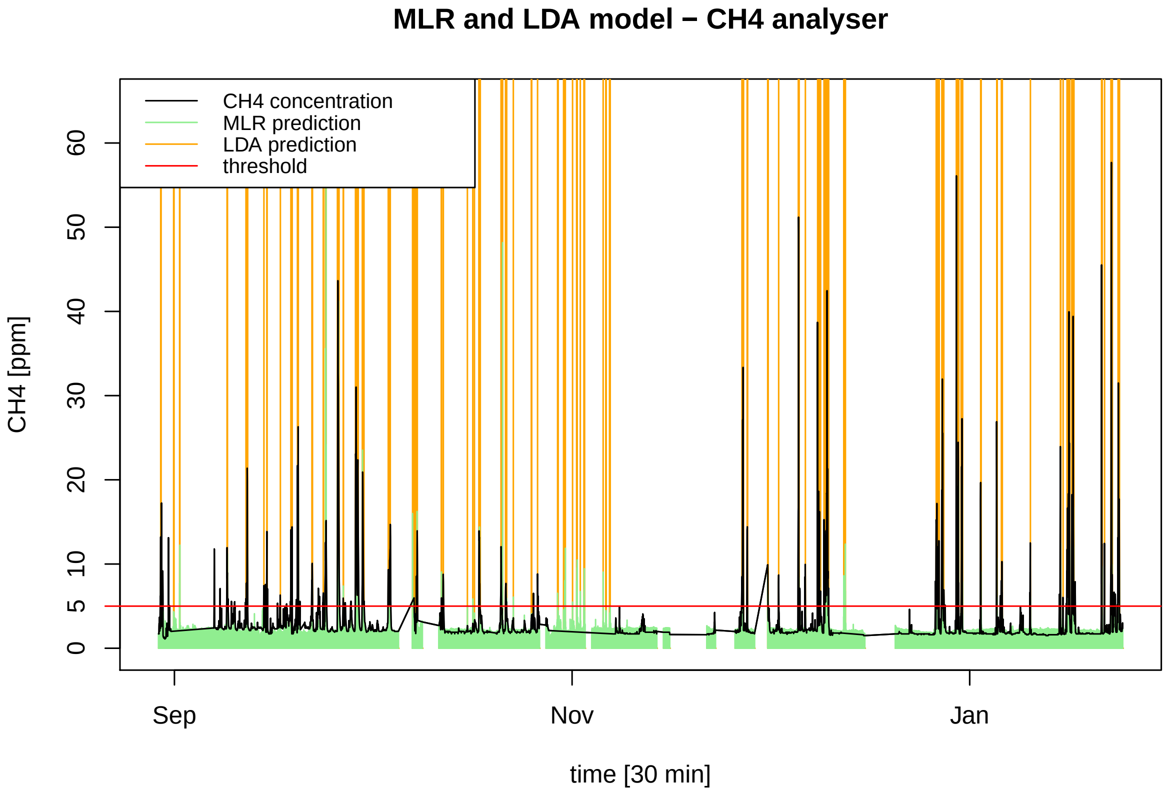


Figure 5.8: LDA and MLR predictions for the CH_4 analyser

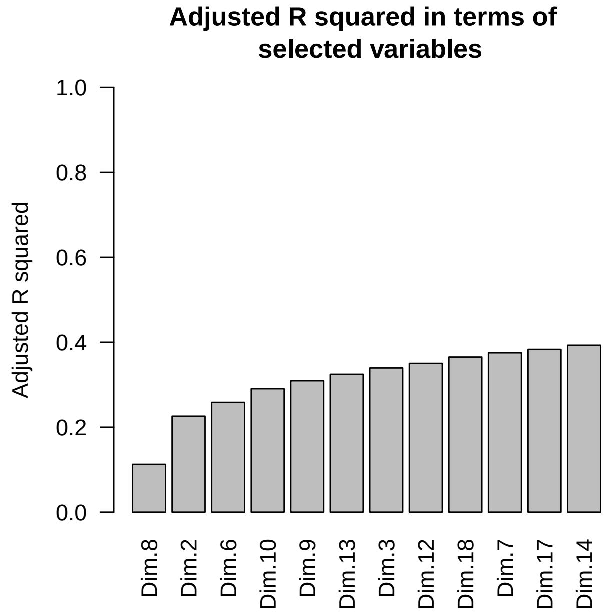


Figure 5.9: Adjusted R squared in terms of the selected variables

5.2.3 Cross-validation with random split

By cross-validation, we examine how robust the model is. Therefore, we extract the sensitivities and specificities of the LDA model, as well as the adjusted R squares from the MLR model over the twenty executions shown in Table 5.5.

Table 5.5: Range and mean over the sensitivities, specificities and adjusted R squares

	Min	**Mean**	**Max**
Sensitivity	0.4	0.45	0.52
Specificity	0.97	0.98	0.99
Adjusted R^2	0.36	0.38	0.42

The sensitivity presents its values between 0.4 and 0.52. The second rate in comparison remains even more stable around 0.98 by a factor of 0.01. The mean of the adjusted R squared equals 0.38. The table shows a good stability in terms of these values.

Figure 5.10 represents the twenty ROC curves in the cross-validation of the LDA model. These curves present the same trend and confirm the good stability of true positive and true negative rate (sensitivity and specificity respectively).

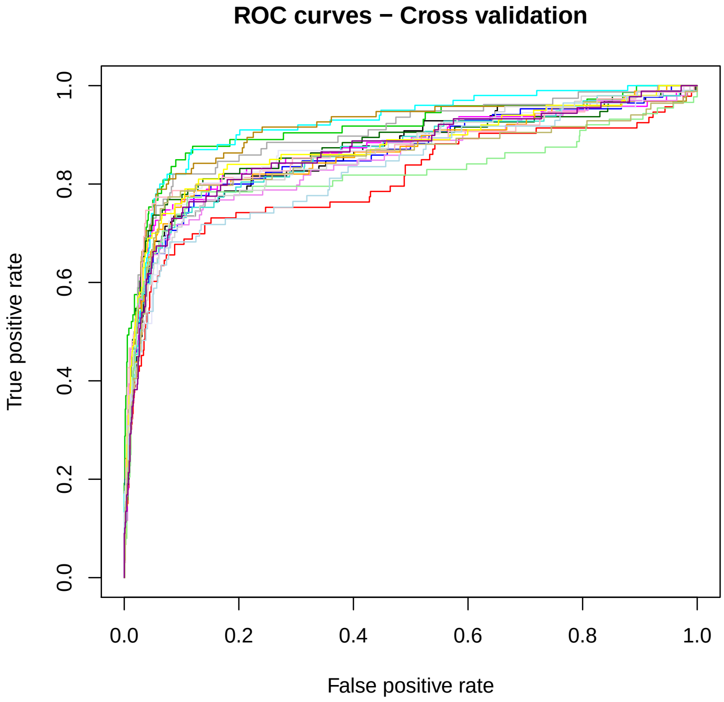


Figure 5.10: ROC curves in the CH_4 prediction

5.2.4 Prediction with split of the time series

On the following Figure 5.11 the prediction of the training set by the LDA model in blue, the one of the test set in green. The prediction of the MLR model with the selected variables from the LDA model is also represented. These prediction values are shown in pink, the test set in orange.

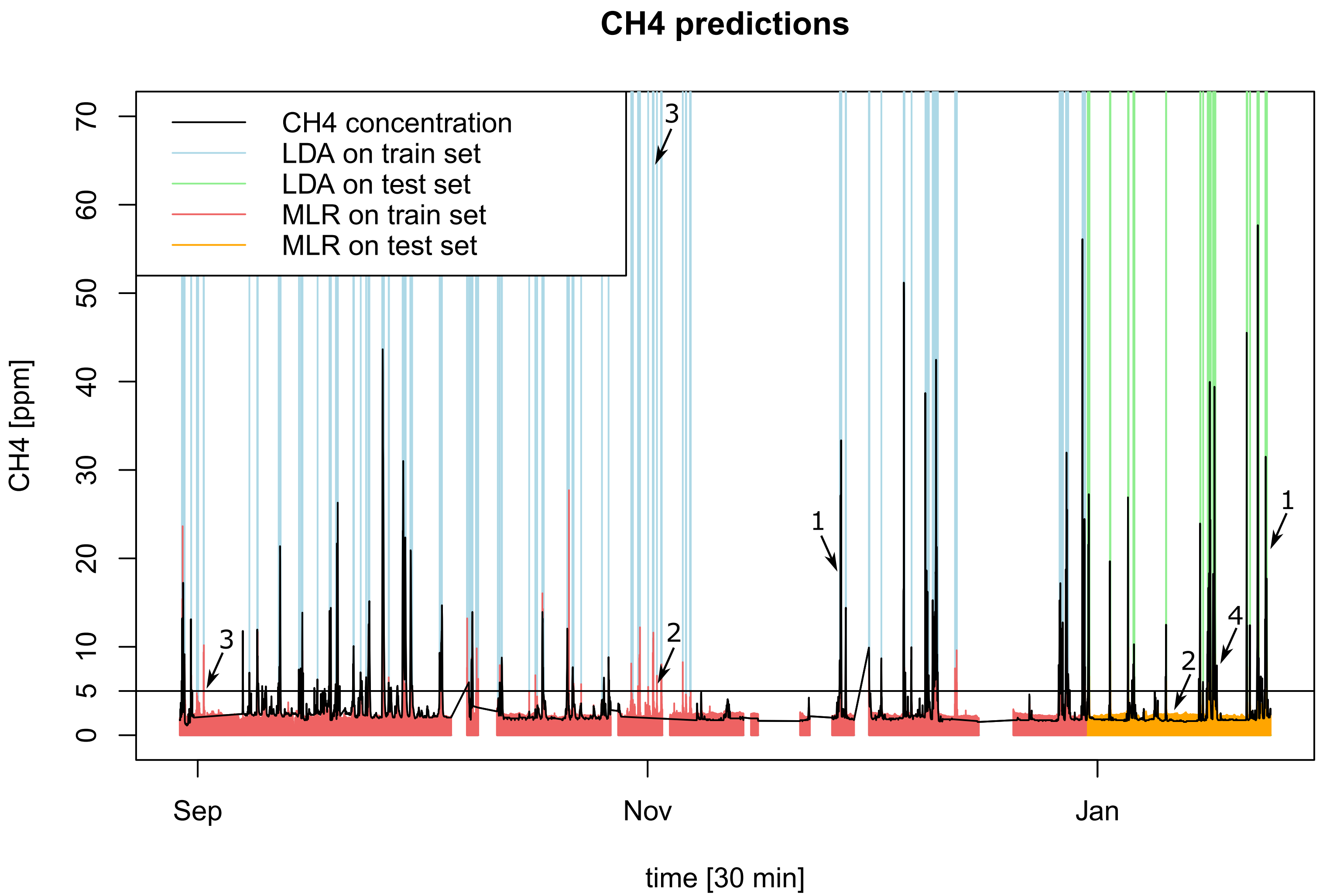


Figure 5.11: Predictions by MLR and LDA methods

Observations of the predictions The following observations are indicated by the respective numbers in Figure 5.11:

1. The signal predictions of the MLR method have too small signals when there are true signals in the training and test set.
2. However, when the analysers measurements are beneath the threshold, the MLR predictions are often higher than the reference values.
3. For the LDA method, there are some predicted false signals, mainly at the beginning of November, but no false signals in the test set.
4. When we look on the prediction of the test set, one smaller peak of the CH_4 has not been detected by neither of the two methods.

5.2.5 Discussion of residuals and false predictions

In terms of the response variable

The following Figure 5.12 represents the residuals in terms of the logarithmic CH_4 values. The red coloured points correspond to false negative predictions, the green ones to false positive predictions provided by the LDA model. Remember that the residuals are defined as the difference between the observed and the predicted values. We see that most of the false negative observations (red) have a positive residual term. Thus, the prediction is smaller than the observation and the real signal is predicted as noise. This is also enforced by the unbalance in the number of signal and noise observations. For the false positive observations (green), the contrary takes place. The predicted values are higher than the real measurements and give false signal predictions.

In terms of the control parameters

Remember that the sensors are very sensitive instruments which are depending on humidity, temperature and so on. Therefore, it is possible that the control parameters contribute to the high number of false negative predictions in the LDA model. We represent the residuals in terms of these control parameters in Figure 5.13.

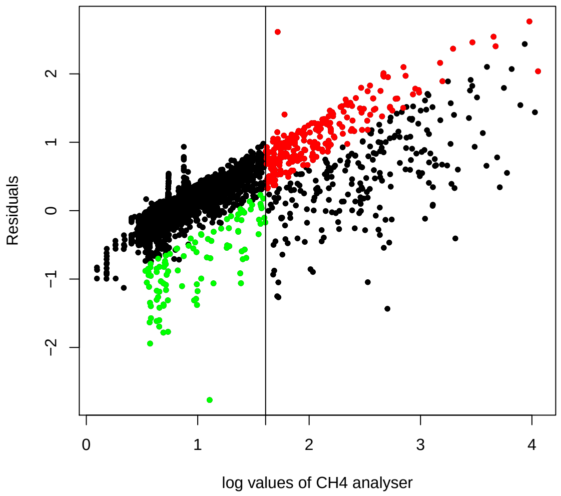


Figure 5.12: Residuals of the MLR model in terms of log(A_CH4_ppm)

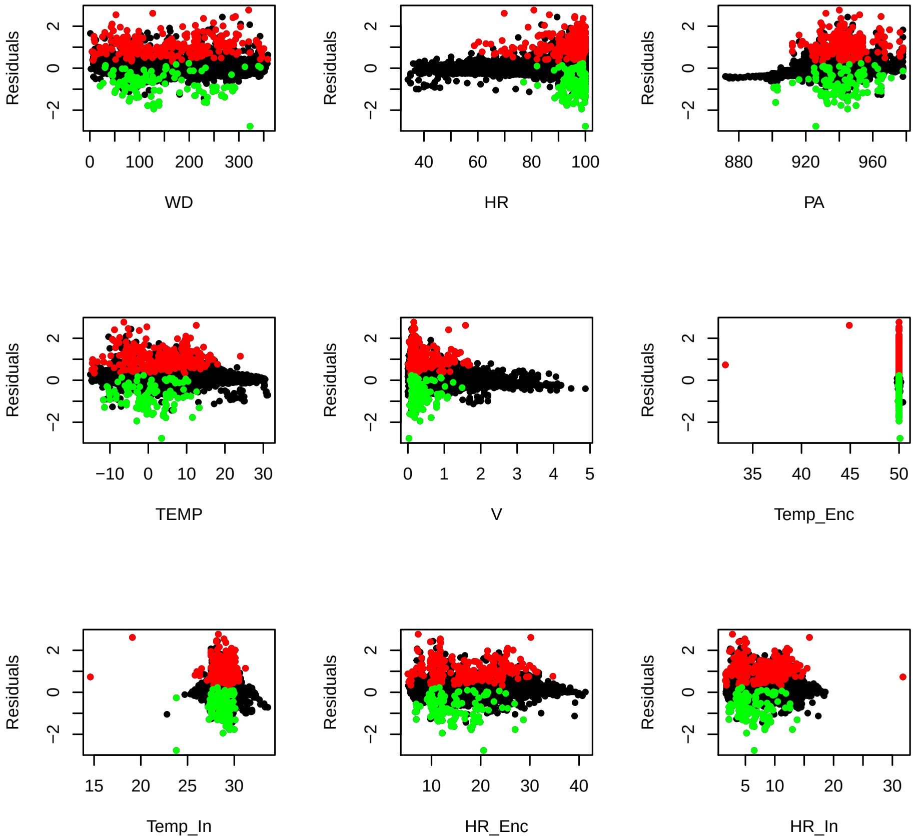


Figure 5.13: Residuals of the MLR model in terms of the control parameters

We observe a dependence between control parameter and residuals for the relative humidity and the wind velocity. When the relative humidity increases, the residual range becomes more important. The contrary effect is the case for the wind velocity, when the velocity increases, the residual error gets smaller. So high humidity and little velocity measurements lead to high residuals, and therefore also to bad predictions. When the humidity has small values (40%-60%), there are nearly no false predictions, as well as for high velocity values. An explication of this effect could be that under these conditions (high humidity and little velocity), the concentration of methane presents its highest values. But the high number of noise values truncates the predictions. Concerning the other control parameters, the false negatives and positives appear under the same conditions.

5.2.6 Which interactions of sensors contribute to the prediction?

Figure 5.14 represents the contributions of every interaction of sensors in the selected principal components in the LDA model. The size of the circles represents the contribution of the interactions as percentages. The colour of the circles enforces this proportion. A very light big circle stands for an important contribution, a very dark small circle for a weak contribution in the concerned principal component. The first selected principal component is Dim.8, which shows up one very important contribution of 30%: the sensor TGS2602. We remark that in the contributions of this principal component, the interactions containing TGS2610 · TGS2611 present often slightly larger circles. In the second selected variable Dim.2, we remark the same. Moreover, many of the interactions in higher orders contribute to the prediction which supports our hypothesis that there is information in a combination of sensor signals. Dim.6 provides two bigger circles for the TGS2610 and TGS2611 sensors (16% and 19% respectively). The TGS2620 sensor reappear in Dim.7 with a contribution of 38%. The sensors TGS2620 and TGS2444 are present in the last two selected principal components with respectively 23% and 16%.

It was expected that the TGS2611 sensor contributes strongly on the prediction of methane, because of its announced selectivity. Indeed, this sensor shows up in the first three selected components, but it is often accompanied by the sensor TGS2610. The TGS2602 sensor however represents also an important contribution in even two selected variables. This could be explained by the higher correlation between the methane and hydrogen sulphide concentrations (see Figure 5.15).

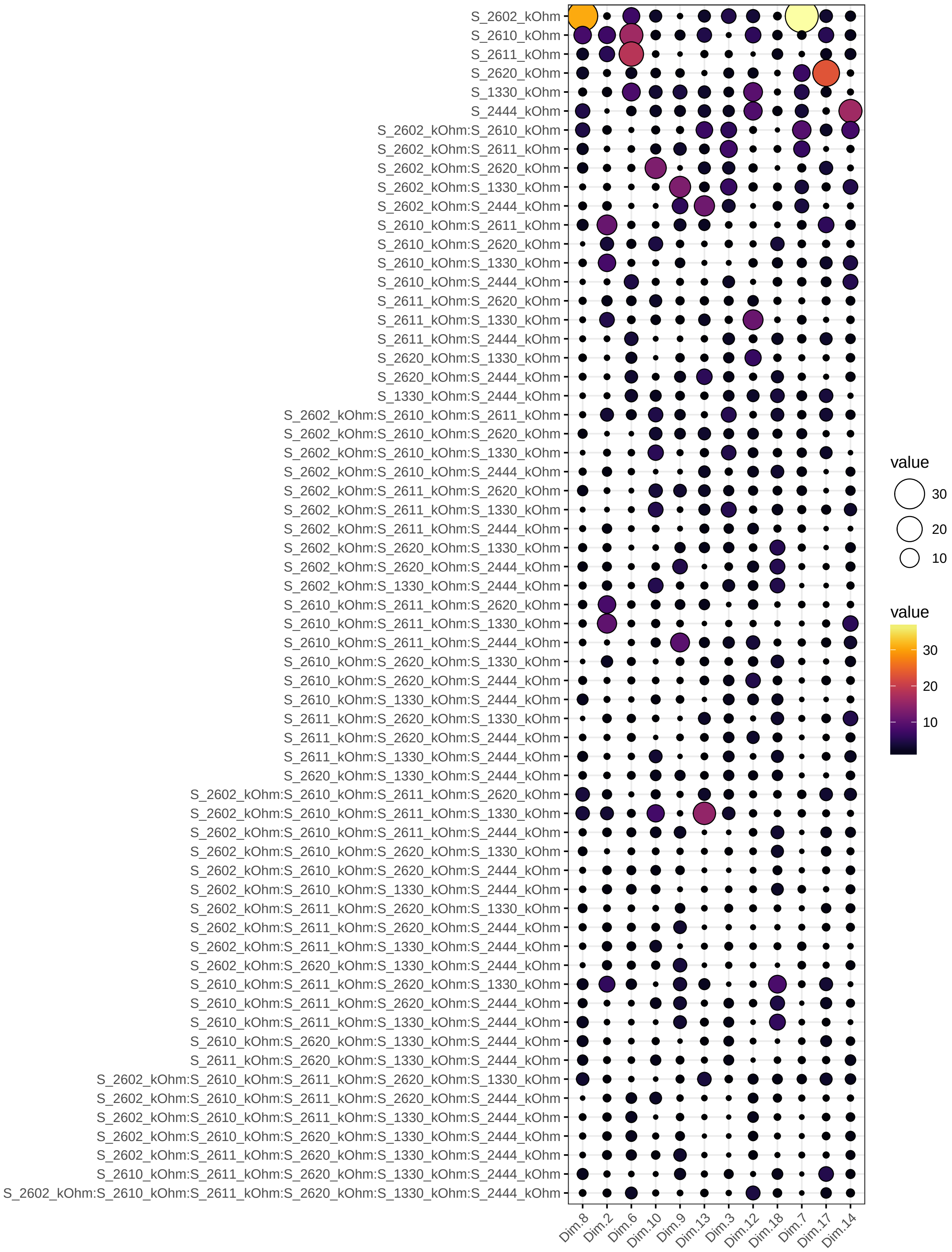


Figure 5.14: Contributions of interactions of sensors in LDA prediction [%]

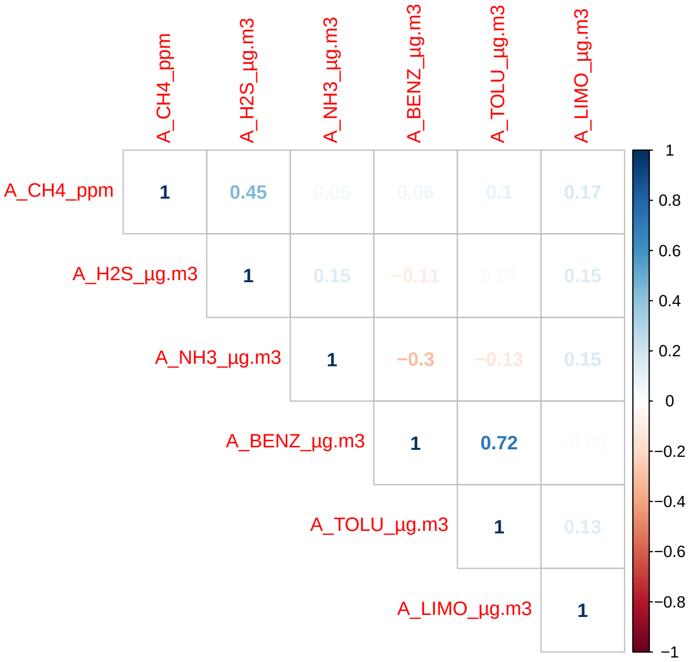


Figure 5.15: Correlation matrix of the analysers concentrations

5.3 Model diagnosis for the H_2S analyser

5.3.1 Fitting linear discriminant analysis model over the complete data set

For the H_2S analyser we chose the threshold of 20 $\mu g/m^3$, which is the standard olfactory threshold for hydrogen sulphide [14].

Selected principal components The model selection of the LDA model chooses 11 out of the 63 principal components. The evolution of the accuracy during the selection is represented in Figure 5.16. As for the CH_4 analyser, the accuracy starts already high (around 0.87) and increases a bit for the further variables until reaching its maximum of 0.895.

Prediction of signals and pollution events Figure 5.17 shows the LDA prediction on the real CH_4 analyser's curve. The confusion matrix of these predictions is shown in Table 5.6.

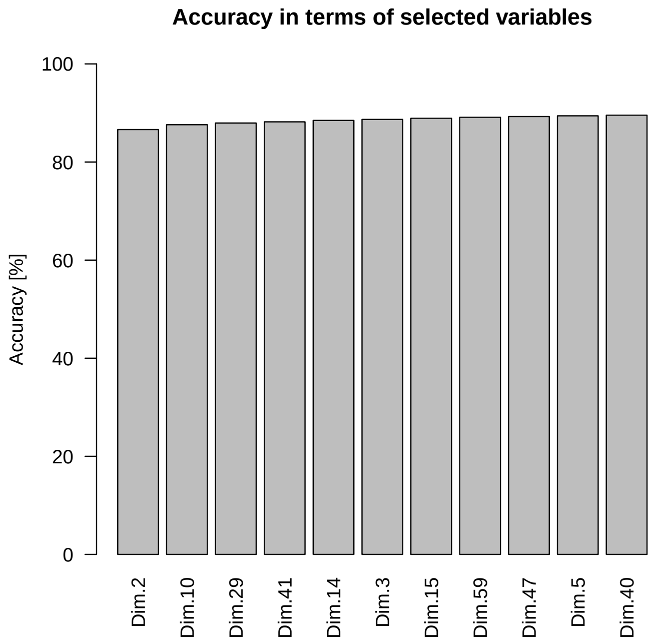


Figure 5.16: Accuracy evolution in the model selection

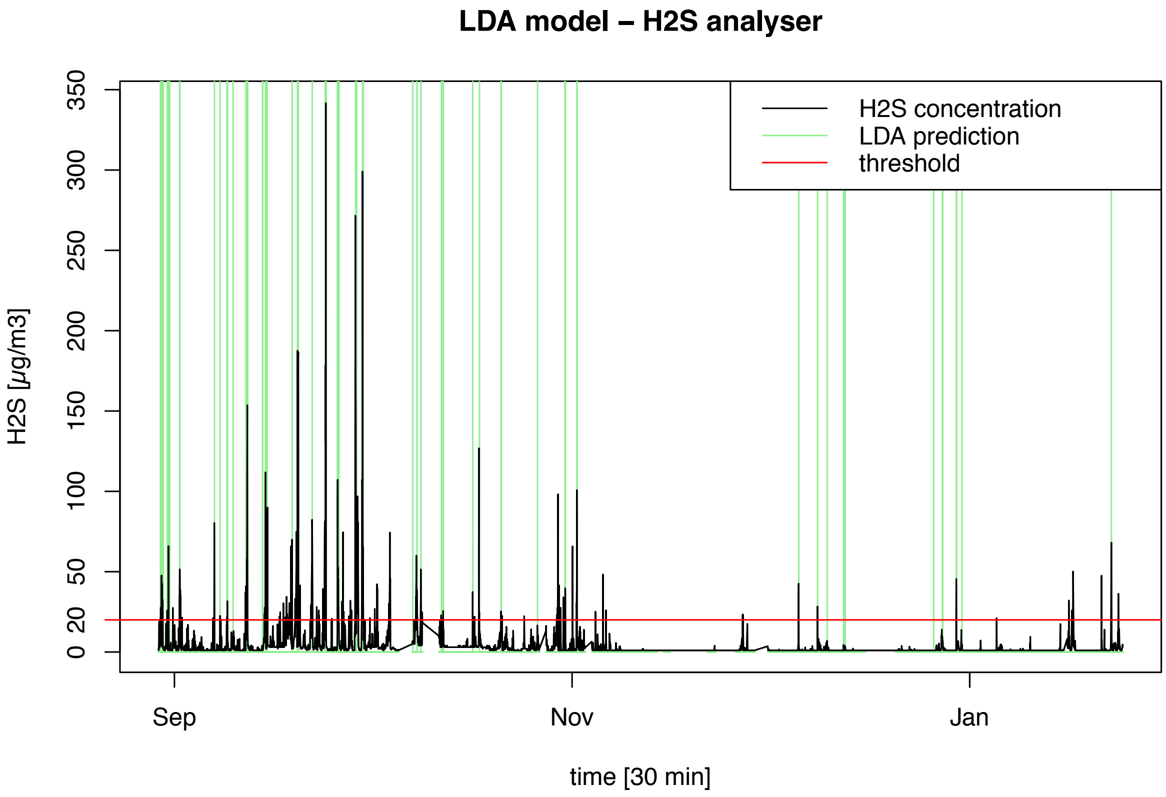


Figure 5.17: H_2S prediction with LDA model

Table 5.6: Confusion matrix of the LDA prediction for the complete data set

		Reference	
		noise	signal
Prediction	noise	5680	208
	signal	38	53

The confusion matrix shows that 208 out of 261 signals and only 38 out of 5718 noise values have been wrong predicted. This results in a weak sensitivity of 0.20, a specificity of around 0.9, a false positive rate of 0.006 and a false negative rate of 0.797. The false negative rate is very high. Nevertheless, when we go back to Figure 5.17, we observe that nearly all the important peaks are detected at least once. Some smaller signals have not been detected, for example at the end of January. We count the number of detected and non detected events (Table 5.7) and the number of true and false predicted events (Table 5.8).

Table 5.7: Detection of pollution events

	Detected	Non detected	Total
Pollution events	37	53	90

Table 5.8: Predicted pollution events

	True	False	Total
Predicted events	36	11	47

Among the 53 non detected events, there are 24 beneath 30 ppm, so very near to the threshold of 20 ppm. Concerning the false predicted events, 3 out of 11 false predicted events occur at moments, when the H_2S analyser not worked. The prediction takes place on the interpolated values, so it is not excluded that a pollution event actually occurred at that time.

False positive and negative predictions On Figure 5.18, the false signal and noise predictions are represented in green and red respectively. This figure confirms the results of the confusion matrix. There are much more false negative signals than false positive ones. Here, the number of noise values is even greater than for the CH_4 analyser. Moreover, the signals of H_2S analyser decrease strongly after November for an unknown reason. But we can see that the important peaks of H_2S are almost all always detected at least once. Thus, when counting the exact predicted signals, the result is very bad. However, in terms of pollution events the important H_2S signals are very often detected. As for the CH_4 analyser, the gaps in this figure are due to missing sensors values due to a non functioning or the connection of an odour bag.

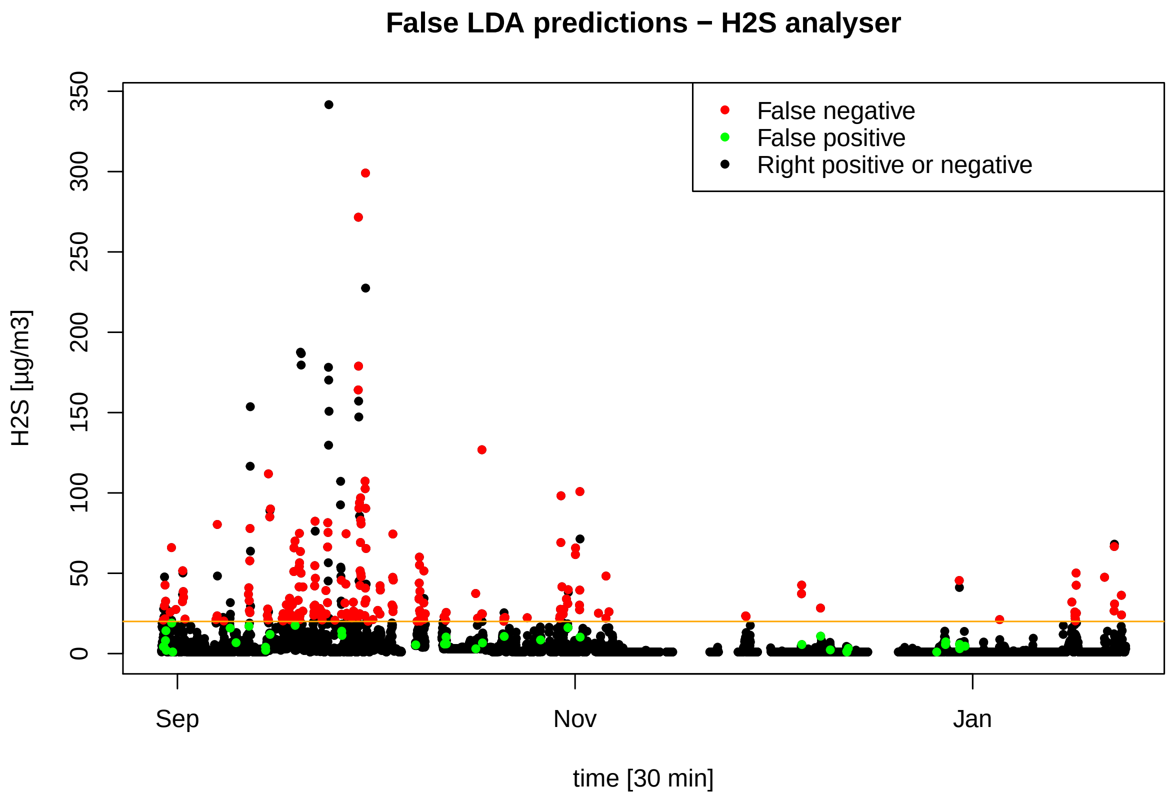


Figure 5.18: False predictions in the LDA model

ROC Curve Figure 5.7 represents the ROC curve for the hydrogen sulphide concentrations. Naturally, the results already discussed before are confirmed in this figure. The curve increases more slowly in terms of the true positive rate (sensitivity), but the false positive rate is small. For the H_2S prediction, the area under the curve equals 0.81.

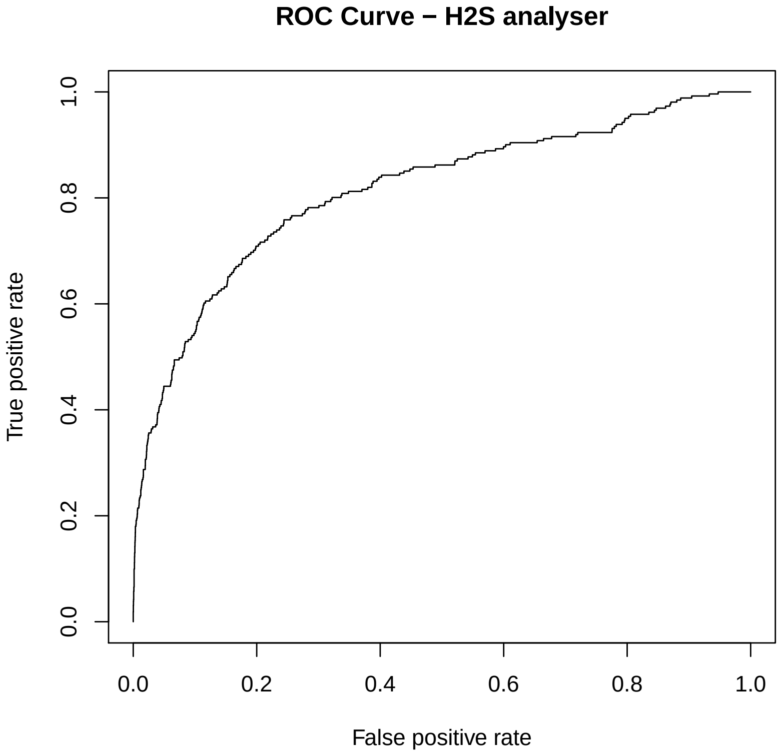


Figure 5.19: ROC curve of the LDA model over the complete data set

5.3.2 Comparison of the LDA and MLR model predictions

The predictions of the LDA and MLR model are illustrated with the real H_2S measurements in Figure 5.20. We can observe that the LDA and MLR model provide very similar predictions. When a signal has not been detected by the LDA model, the MLR predictions are underestimating the true concentration as well.

The evolution of the adjusted R squared is illustrated in Figure 5.21. We observe an increase from circa 0.06 up to 0.15 of the adjusted R squared, which is smaller than the ones for the CH_4 predictions.

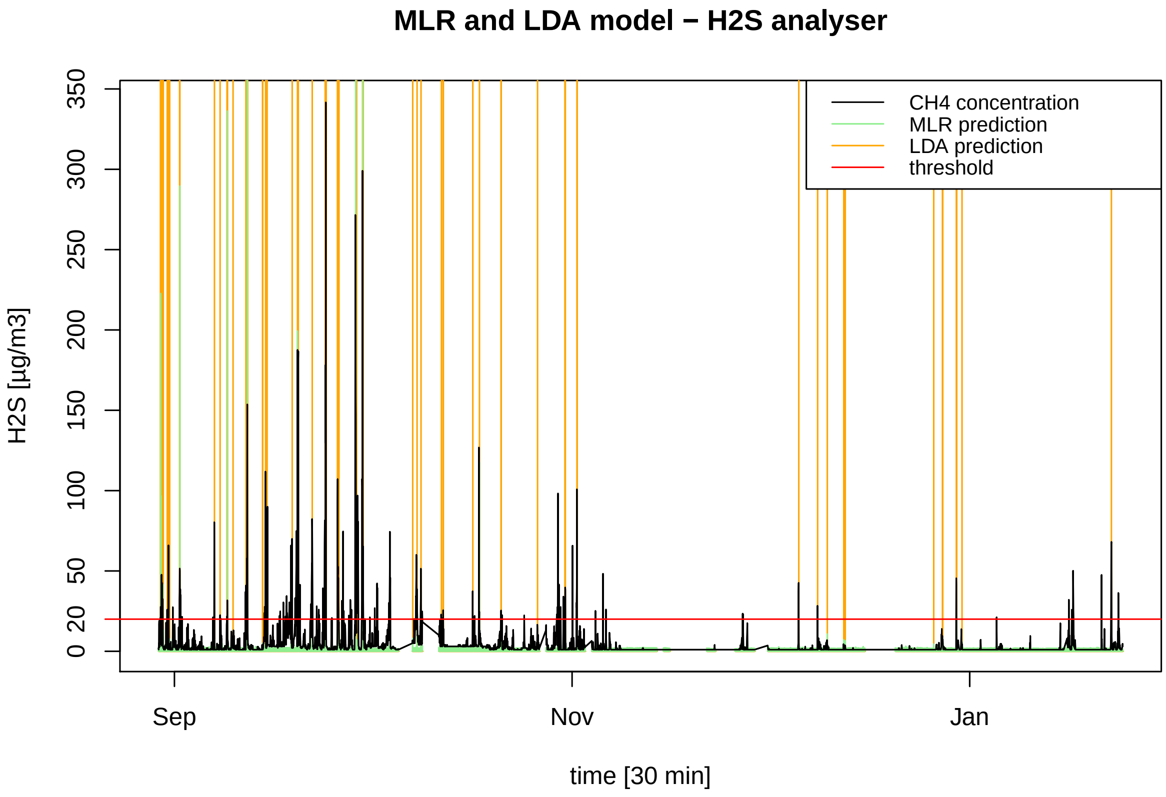


Figure 5.20: LDA and MLR predictions for the H_2S analyser

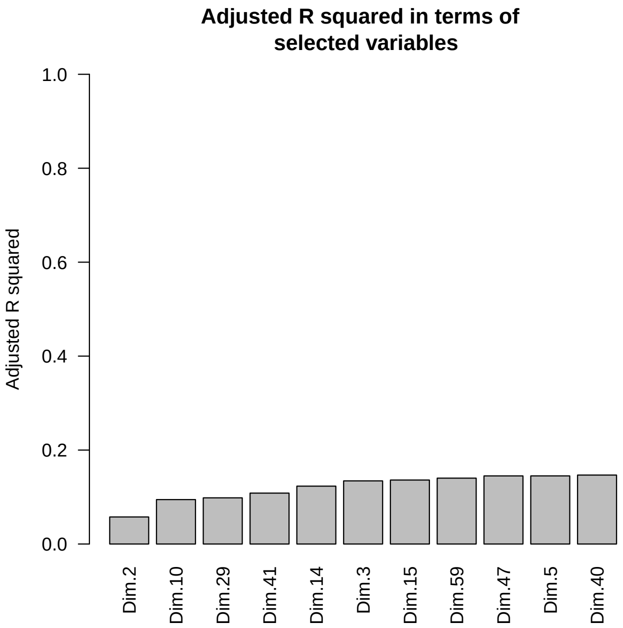


Figure 5.21: Adjusted R squares in terms of the selected variables

5.3.3 Cross-validation with random split

Table 5.9 represents the minimum, mean and maximum of the sensitivities, specificities and adjusted R squares from the cross-validation with 20 executions.

Table 5.9: Range and mean over the sensitivities, specificities and adjusted R squared

	Min	**Mean**	**Max**
Sensitivity	0.12	0.2	0.31
Specificity	0.987	0.991	0.996
Adjusted R^2	0.104	0.151	0.179

The sensitivity presents its values between 0.12 and 0.31. The specificity is very high and stable. The mean of the adjusted R squared equals 0.151 and remains also stable.

Figure 5.10 represents the twenty ROC curves in the cross-validation of the LDA model. These curves present the same trend and confirm the good stability of true positive and true negative rate (sensitivity and specificity respectively).

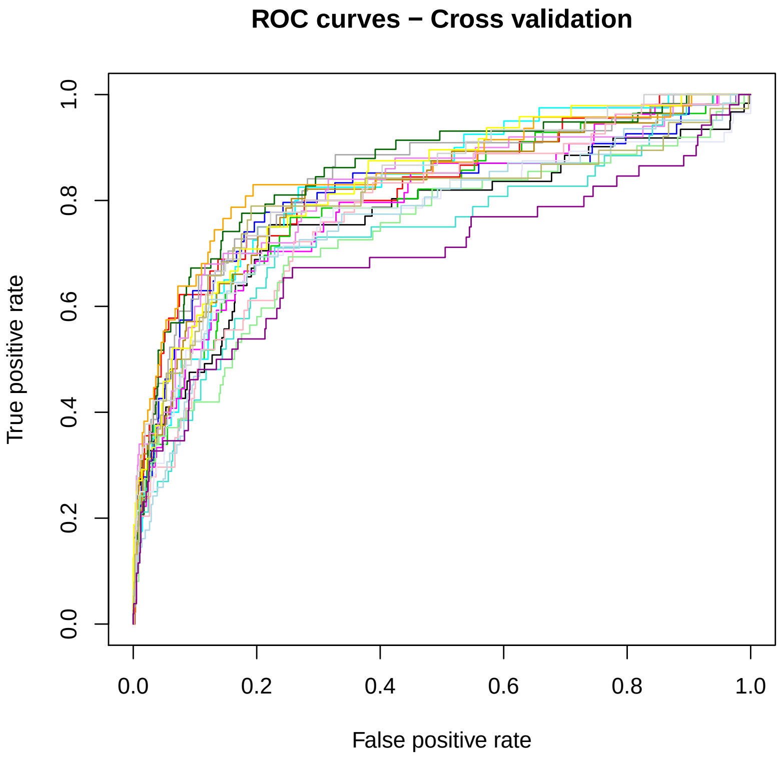


Figure 5.22: ROC curves in the H_2S prediction

5.3.4 Prediction with split of the time series

Because of the decrease in H_2S concentrations for the time period from November to January, it makes no sense to split the time series in 80% and 20% to perform a prediction.

5.3.5 Discussion of residuals and false predictions

In terms of the response variable

The following Figure 5.23 represents the residuals in terms of the logarithmic H_2S values. The red coloured points correspond to false negative predictions, the green ones to false positive predictions provided by the LDA model. We see that all false negative observations (red) have a positive residual term. Thus, the prediction is smaller than the observation and the real signal is predicted as noise. This is also enforced by the strong unbalance in the number of signal and noise observations. For the false positive observations (green), the contrary takes place. Most of the predicted values are higher than the real measurements and give false signal predictions.

In terms of the control parameters

We observe a dependence between control parameter and residuals for the relative humidity and the wind velocity as for the CH_4 analyser before. A high humidity and little velocity measurements lead to high residuals, thus to bad predictions. When the humidity has small values (40%-60%), there are nearly no false predictions, as well as for high velocity values. An explication of this effect could be that under these conditions (high humidity and little velocity), the concentration of hydrogen sulphide presents its highest values. But the high number of noise values truncates the predictions. Concerning the other control parameters, the false negatives and positives appear under the same conditions.

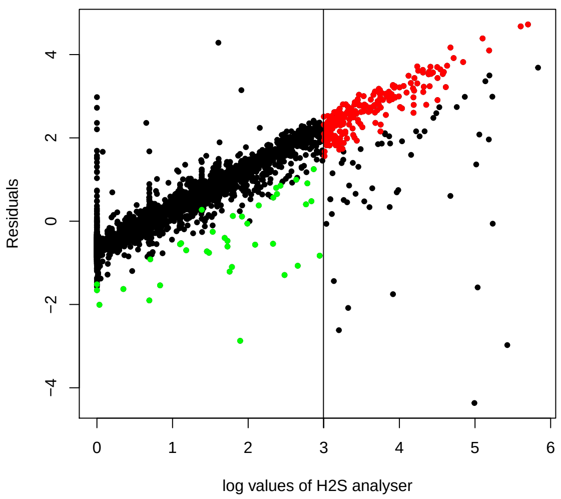


Figure 5.23: Residuals of the MLR model in terms of $\log(A\_H_2S)$

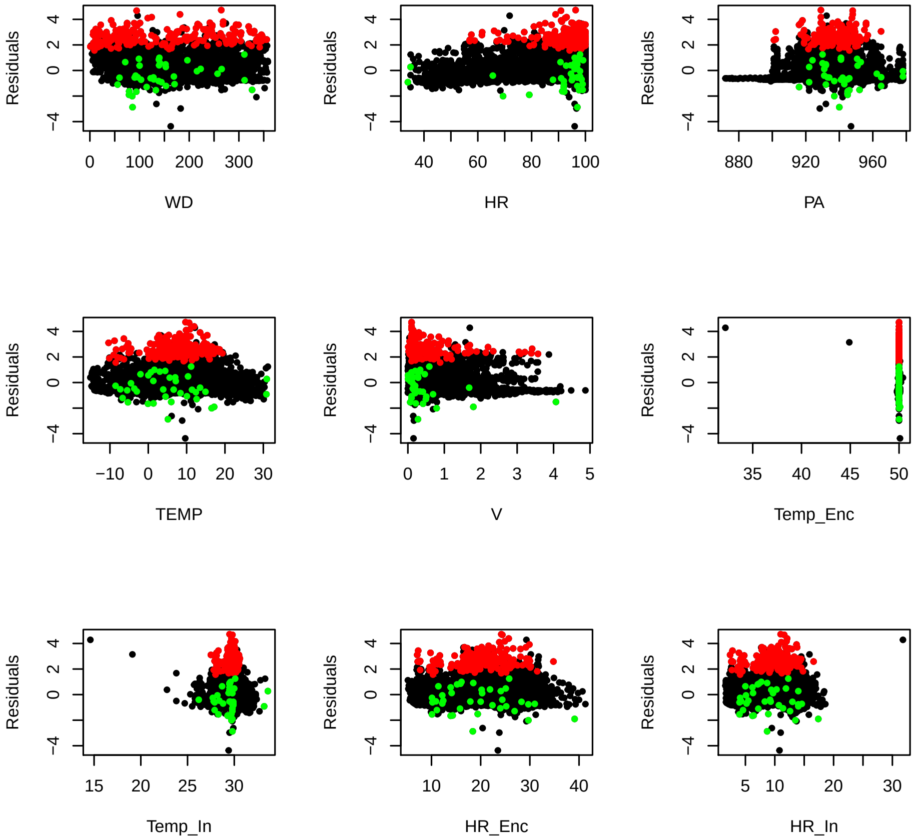


Figure 5.24: Residuals of the MLR model in terms of the control parameters

5.3.6 Which interactions of sensors contribute to the prediction?

The contributions of every interaction of sensors in the selected principal components in the LDA model are represented in Figure 5.25. Remember that the size and the lightness of the circles are proportional to the contribution percentages. The first selected principal component is Dim.2 showing up the most important contribution of 11% for the interaction TGS2610 · TGS2611. The two sensors individually show up a higher contribution as well. In higher orders of interactions, the contributions containing TGS2610 · TGS2611 are always more present, combined either with TGS2620 or TGS1330. The second variable Dim.10 displays a contribution of 13% for TGS2602 · TGS2620. The interaction TGS2602 · TGS2610 appears with a contribution of 17.55% in Dim.5. Finally, the greatest contribution (around 20%) is shown for the last selected variable Dim.40 for the interaction TGS2611 · TGS2444. The presence of important contributions for interactions in higher orders affirms our hypothesis that there is information in a combination of sensor signals. We observe that even the interaction with maximum order presents an important contribution to the prediction in Dim.47.

The sensor TGS2602 announces a selectivity for hydrogen sulphide among others. In the contributions, we observe interactions with this sensor, but the sensor alone seems to be not enough to predict H_2S. Furthermore, it is naturally that the TGS2611 sensor appears often in the important contributions of the H_2S prediction. The announced selectivity of this sensor is methane, and methane and hydrogen sulphide occur often together (like shown in the correlation matrix (see Figure 5.15).

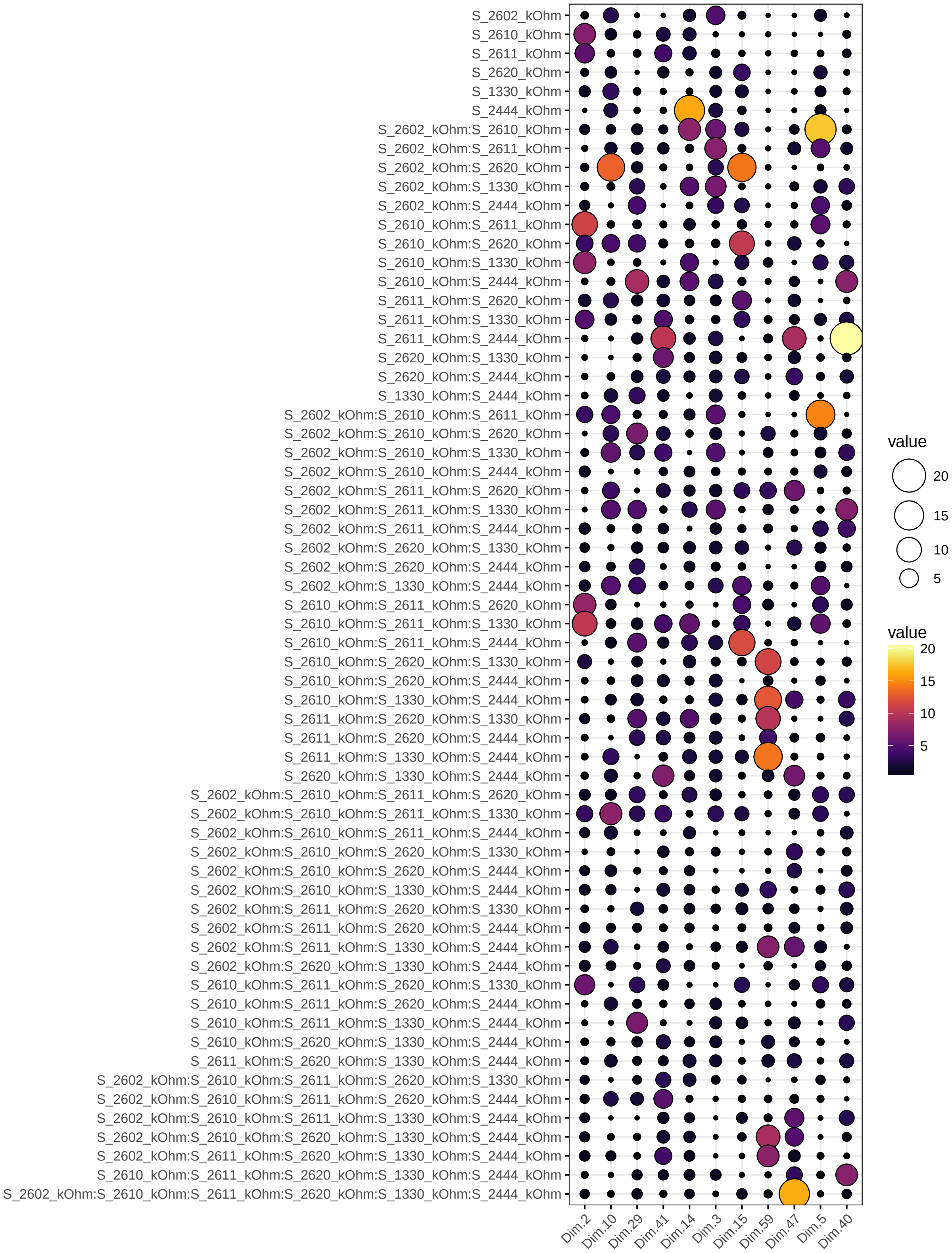


Figure 5.25: Contributions of interactions of sensors in LDA prediction [%]

5.4 Model diagnosis for the NH_3 analyser

The threshold of 25 $\mu g/m^3$ is chosen for the prediction model of ammoniac regarding the range of signals in this data set.

5.4.1 Fitting linear discriminant analysis model over the complete data set

Selected principal components For the ammoniac prediction, 17 variables have been chosen in the model selection. Figure 5.26 represents the evolution of the accuracy when the 17 principal components are added one by one. The addition of the first variable Dim.1 returns an accuracy equal to 0.73. Then, the accuracy increases by 0.1 for each new variable and becomes stable for the last 7 components around 0.81.

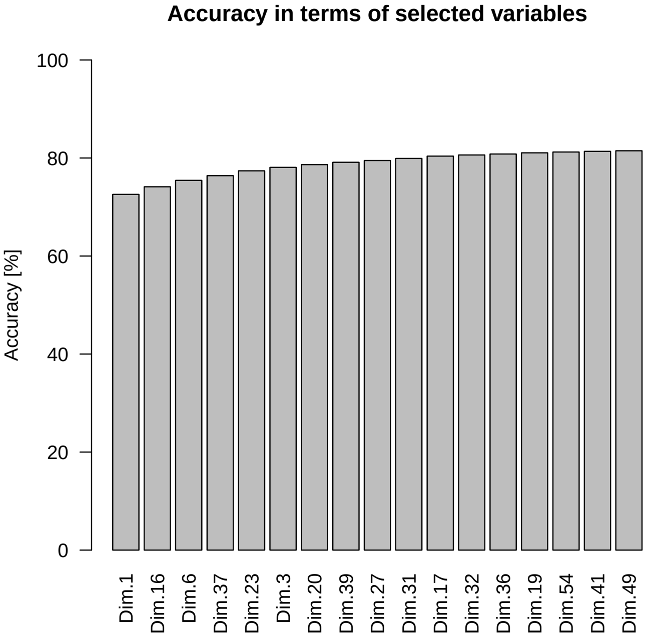


Figure 5.26: Accuracy evolution in the model selection

Prediction of signals and pollution events The LDA prediction with the real NH_3 measurements is illustrated on Figure 5.27. The confusion matrix of these predictions is shown in Table 5.6.

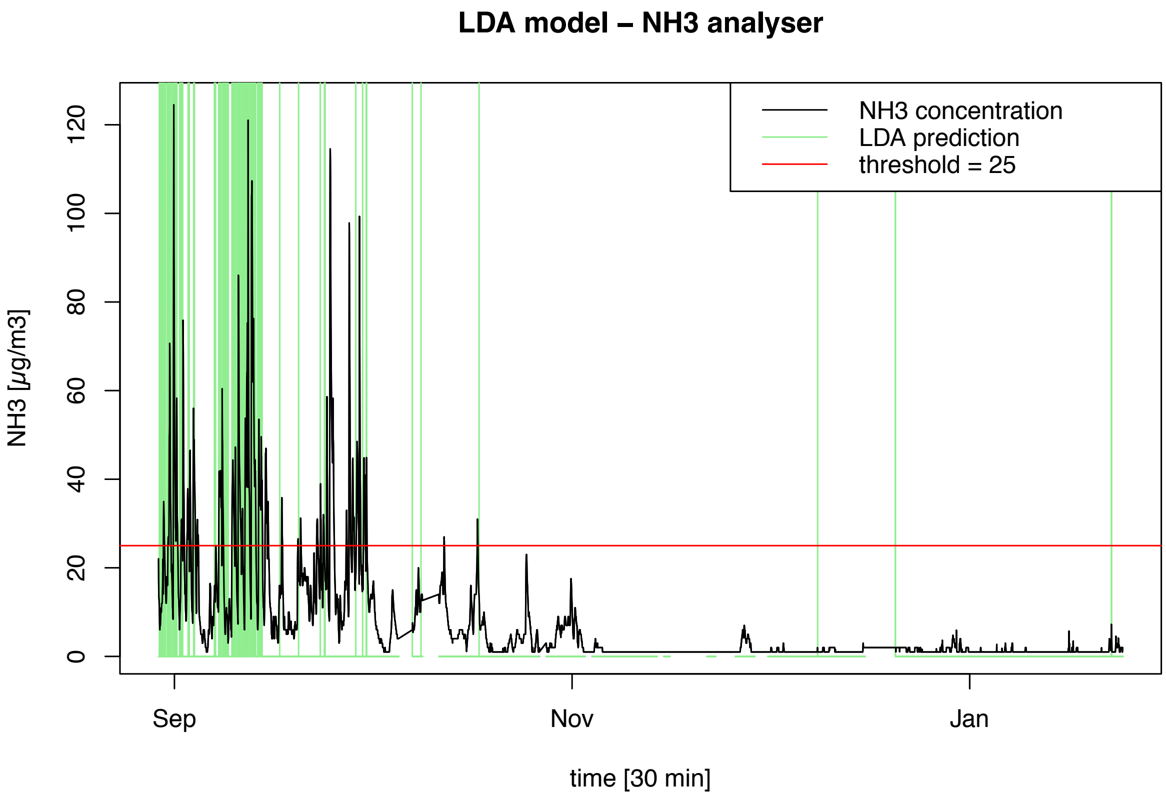


Figure 5.27: NH_3 prediction with LDA model

Table 5.10: Confusion matrix of the LDA prediction for the complete data set

		Reference	
		noise	signal
Prediction	noise	5353	343
	signal	98	185

We observe that 98 out of 5451 noise and 343 out of 528 signal values have been false predicted. In conclusion, the sensitivity equals 0.35, the specificity equals 0.98, the false positive rate has a value of 0.02 and the false negative rate a value of 0.65. The sensitivity is higher than the one for the H_2S prediction, but there are 6 variables more selected in this model. Although the sensitivity is not so high, we see that nearly all strong peaks in Figure 5.27 detected at least once. Some smaller signals have not been detected and between December and January, three false positive signals occur. We count the number of detected and non detected events (Table 5.11) and the number of true and false predicted events (Table 5.12) for a better comprehension of the prediction.

Table 5.11: Detection of pollution events

	Detected	Non detected	Total
Pollution events	28	17	45

Table 5.12: Predicted pollution events

	True	False	Total
Predicted events	29	13	42

Among the 17 non detected events, there are 7 beneath 35 $\frac{\mu g}{m^3}$. There is only one out of the 13 false predicted pollution events occurring when the NH_3 analyser did not work and values are taken from the interpolation. It is possible, that an event occurred at that time.

By taking a zoom on the NH_3 predictions (see Figure 5.28) we observe daily effects: every day, an important peak is shown. This phenomena can be observed almost everywhere from September up to the middle of October. Then, the effect disappears because of the non existence of high signals. The decrease in ammoniac from the end of October has no known reason.

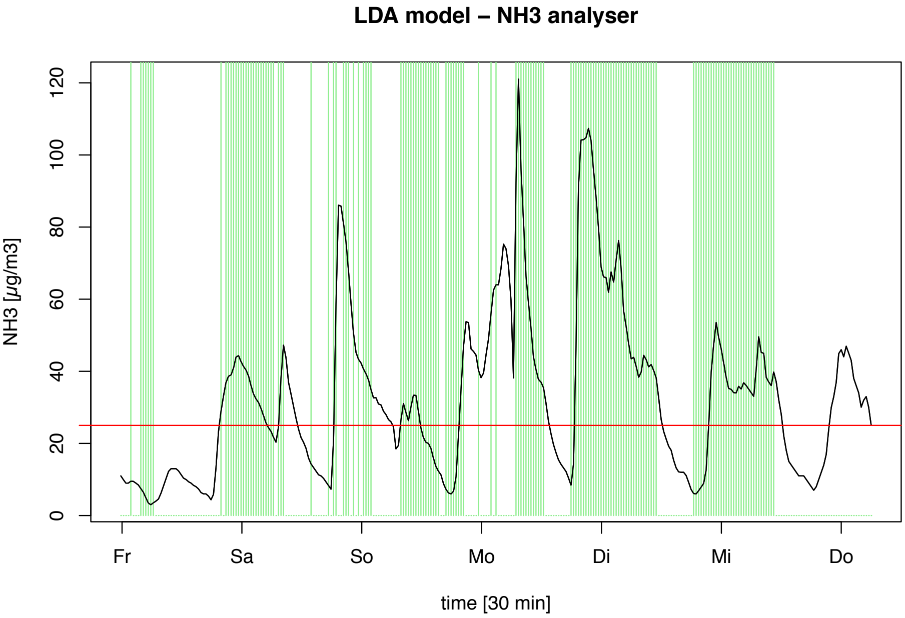


Figure 5.28: Zoom on the NH_3 analyser's signal

False positive and negative predictions In Figure 5.29, the false positive and negative predictions are represented in red and green respectively. We can see, as before in the confusion matrix, that there are much more false negative signals than false positive ones. This effect underlies also the high number of noise values among the NH_3 values. We also observe that the first to groups of peaks are well detected at least once, and afterwards the prediction of signals gets weaker. The prediction is more difficult because there are much more noise values, and no more signals from October up to the end. The missing sensor values present the gaps here as well.

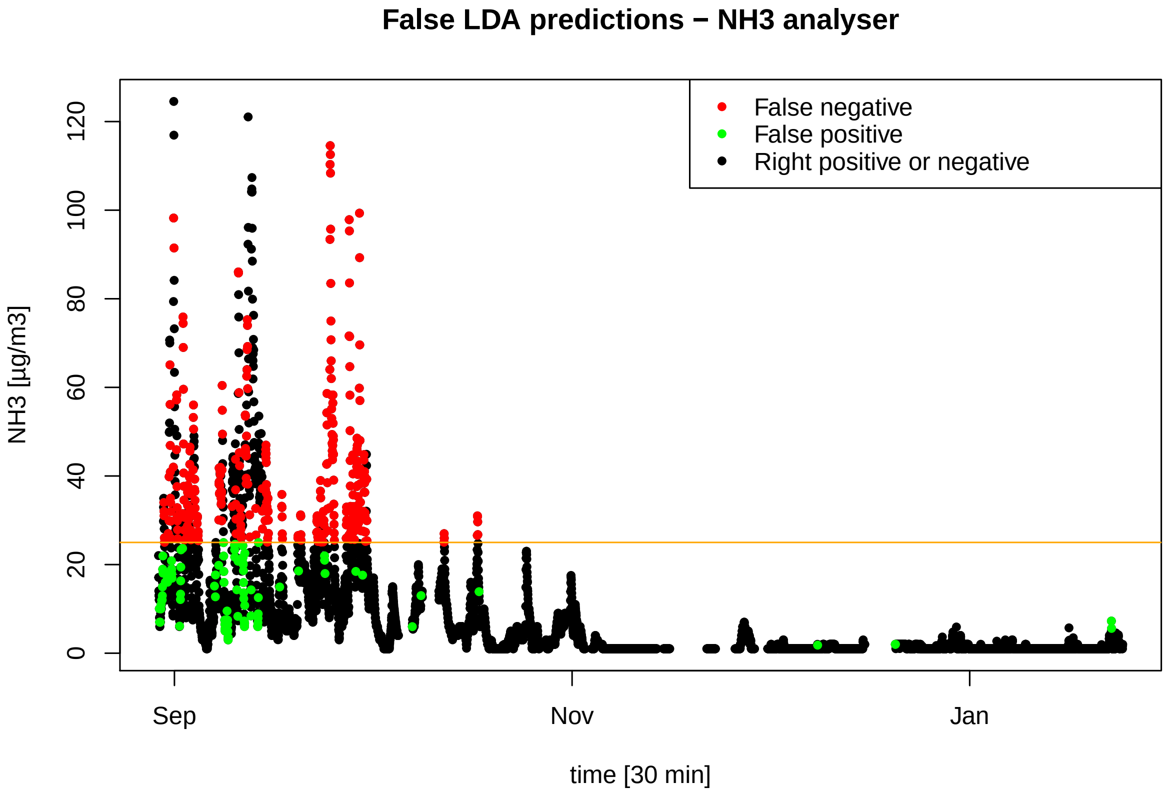


Figure 5.29: False predictions in the LDA model

ROC Curve Figure 5.30 illustrates the ROC curve for the NH_3 prediction. The examination of this curve confirms the results already developed before. The area under the ROC curve is equal to 0.88, which is a very good result.

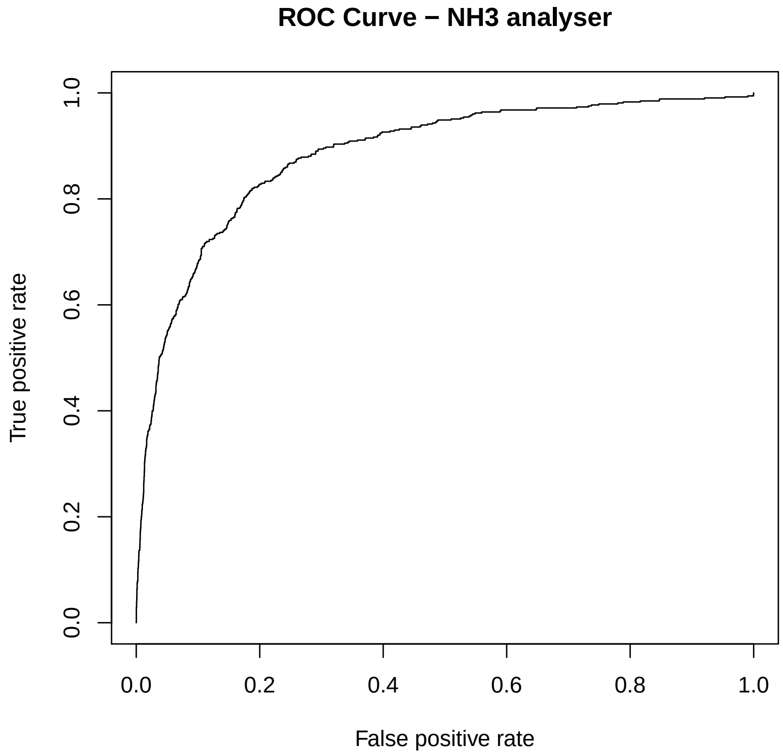


Figure 5.30: ROC curve of the LDA model
over the complete data set

5.4.2 Comparison of the LDA and MLR model predictions

In the following Figure 5.31, the predictions of the LDA and MLR model are superposed. We observe that the LDA and MLR model provide very similar predictions. When a signal has not been detected by the LDA model, the MLR predictions are underestimating the true concentration as well (1). Furthermore, when the LDA model predicts a false positive, the MLR prediction was too high likewise (2). The two events are illustrated in Figure 5.33.

The evolution of the adjusted R squared is illustrated in Figure 5.32. We con observe an increase from circa 0.07 up to 0.3 of the adjusted R squares.

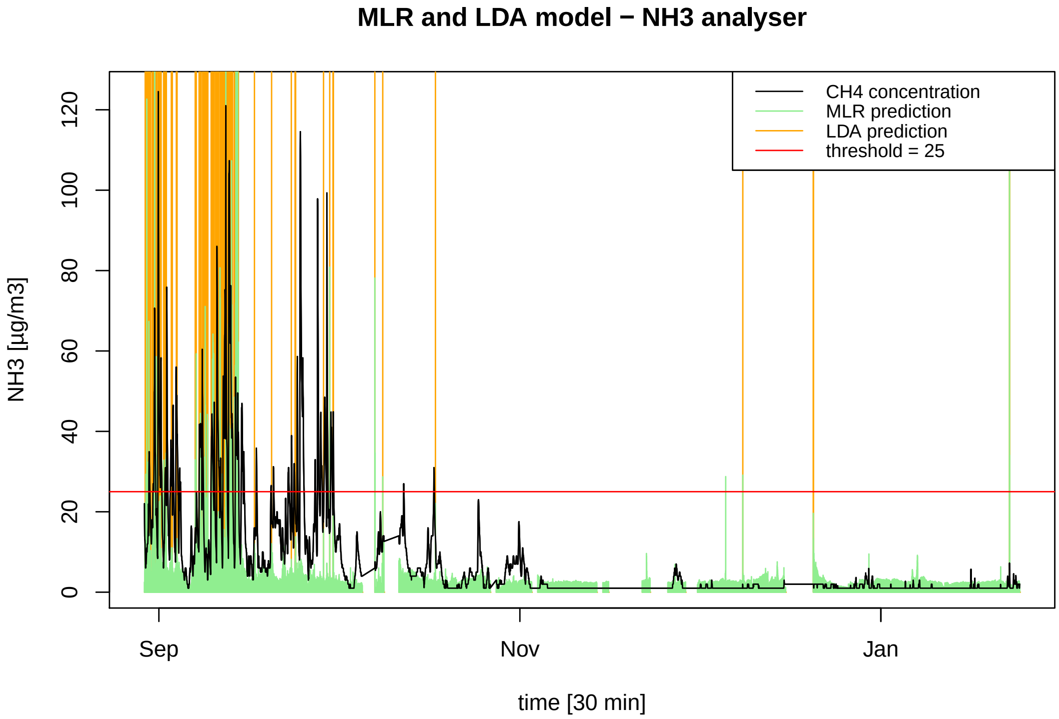


Figure 5.31: LDA and MLR predictions for the NH_3 analyser

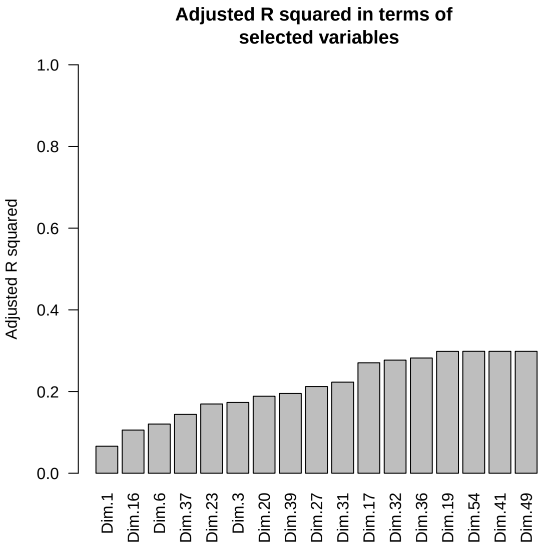


Figure 5.32: Adjusted R squares in terms of the selected variables

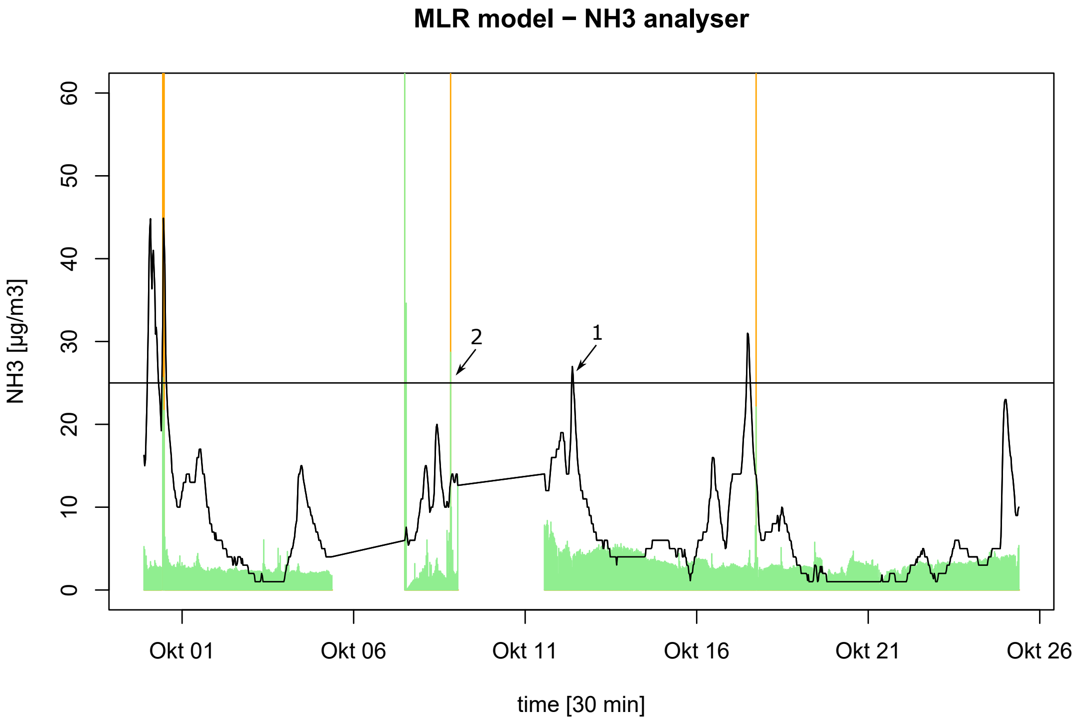


Figure 5.33: Zoom of the LDA and MLR predictions for the NH_3 analyser

5.4.3 Cross-validation with random split

The minimum, mean and maximum of the sensitivities, specificities and adjusted R squares from the cross-validation with 20 executions are represented in Table 5.13.

Table 5.13: Range and mean over the sensitivities, specificities and adjusted R squared

	Min	**Mean**	**Max**
Sensitivity	0.27	0.34	0.42
Specificity	0.97	0.98	0.99
Adjusted R^2	0.28	0.34	0.38

The sensitivity is slightly unstable between 0.27 and 0.42. The specificity is as before very high and stable. The adjusted R squared is almost stable around the mean of 0.34.

Figure 5.34 represents the twenty ROC curves in the cross-validation of the LDA model.

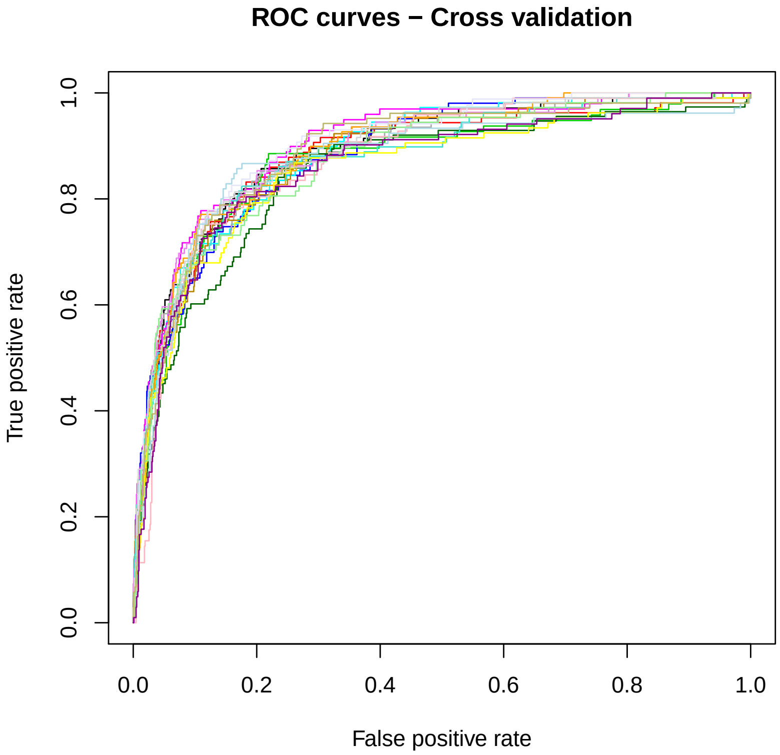


Figure 5.34: ROC curves in the NH_3 prediction

5.4.4 Prediction with split of the time series

Because of the decrease in NH_3 concentrations for the time period from November to January, it makes no sense to split the time series in 80% and 20% to perform a prediction.

5.4.5 Discussion of residuals and false predictions

In terms of the response variable

The following Figure 5.35 represents the residuals in terms of the logarithmic NH_3 values. The red coloured points correspond to false negative predictions, the green ones to false positive predictions provided by the LDA model. We see that all false negative observations (red) have a positive residual term. Thus, the prediction is smaller than the observation and the real signal is predicted as noise. This is also enforced by the unbalance in the number of signal and noise observations. For the false positive observations (green), the contrary takes place. The predicted values are higher than the real measurements and give false signal predictions.

In terms of the control parameters

Figure 5.36 represents the residuals of the MLR model after variable selection in terms of the control parameters. Unlike for the first two analysers, the prediction is not visually affected by humidity and only slightly by the wind velocity. And concerning all the other parameters of control, the false predictions are present under the same conditions as well.

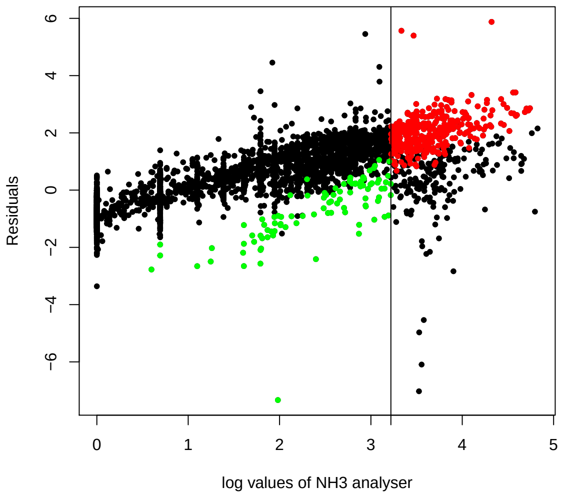


Figure 5.35: Residuals of the MLR model in terms of log(A_NH$_3$)

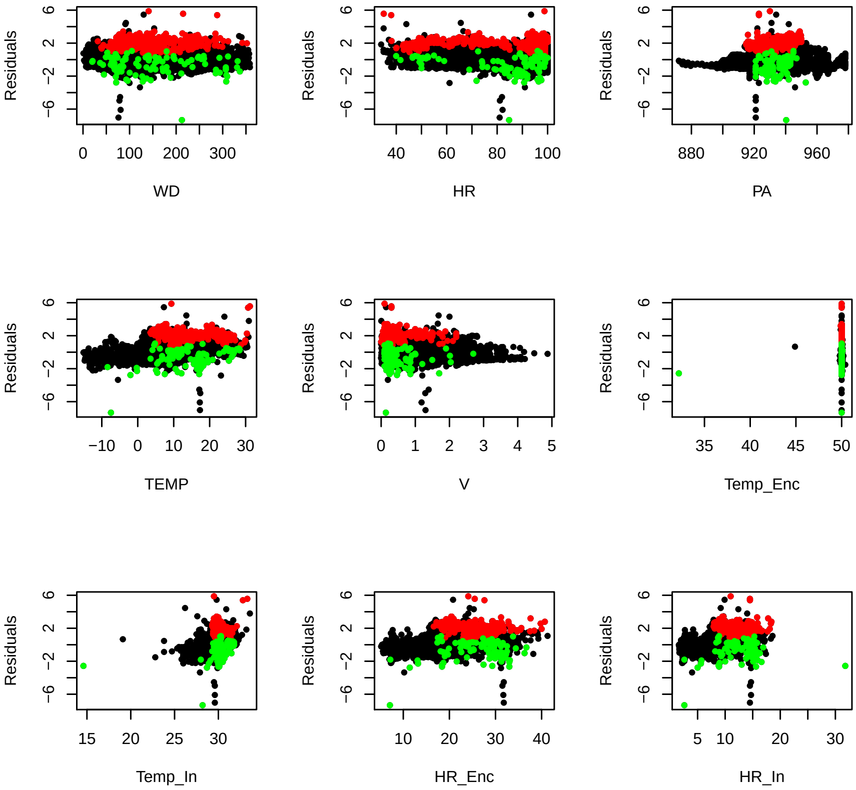


Figure 5.36: Residuals of the MLR model in terms of the control parameters

5.4.6 Which interactions of sensors contribute to the prediction?

Figure 5.37 illustrates the contributions of every sensors interaction with a circle. A big and light circle indicates an important contribution to the prediction of NH_3. Concerning the first selected variable, namely Dim.1, there is no standing out interaction. In the second selected variable Dim.16 however, the highest contribution of around 26% occurs for the TGS2444 sensor. This principal components presents also a higher contribution of 15% for the TGS2620 sensor. The third selected variable Dim.6. shows up higher contributions for the sensors TGS2610 and TGS2611, but also for interactions containing the TGS2444 sensor. Dim.37 presents the interaction TGS2602 · TGS2620 · TGS2444 with 12%. Afterwards, Dim.17 contains a contribution of 23.1 % for the sensor TGS2620. In Dim.19, the interaction TGS2602 · TGS2444 occurs with 15.87%. The appearance of important contributions for interactions in higher orders confirms our hypothesis of information present in a combination of sensor signals.

Regarding the announced compounds selectivities of the sensors, the sensor TGS2444 is expected to detect ammoniac as well as the TGS2602. The TGS2444 sensor is very present, already in the second selected component with the highest contribution. The TGS2602 sensor appears also in higher contributions but often in combination with other sensors.

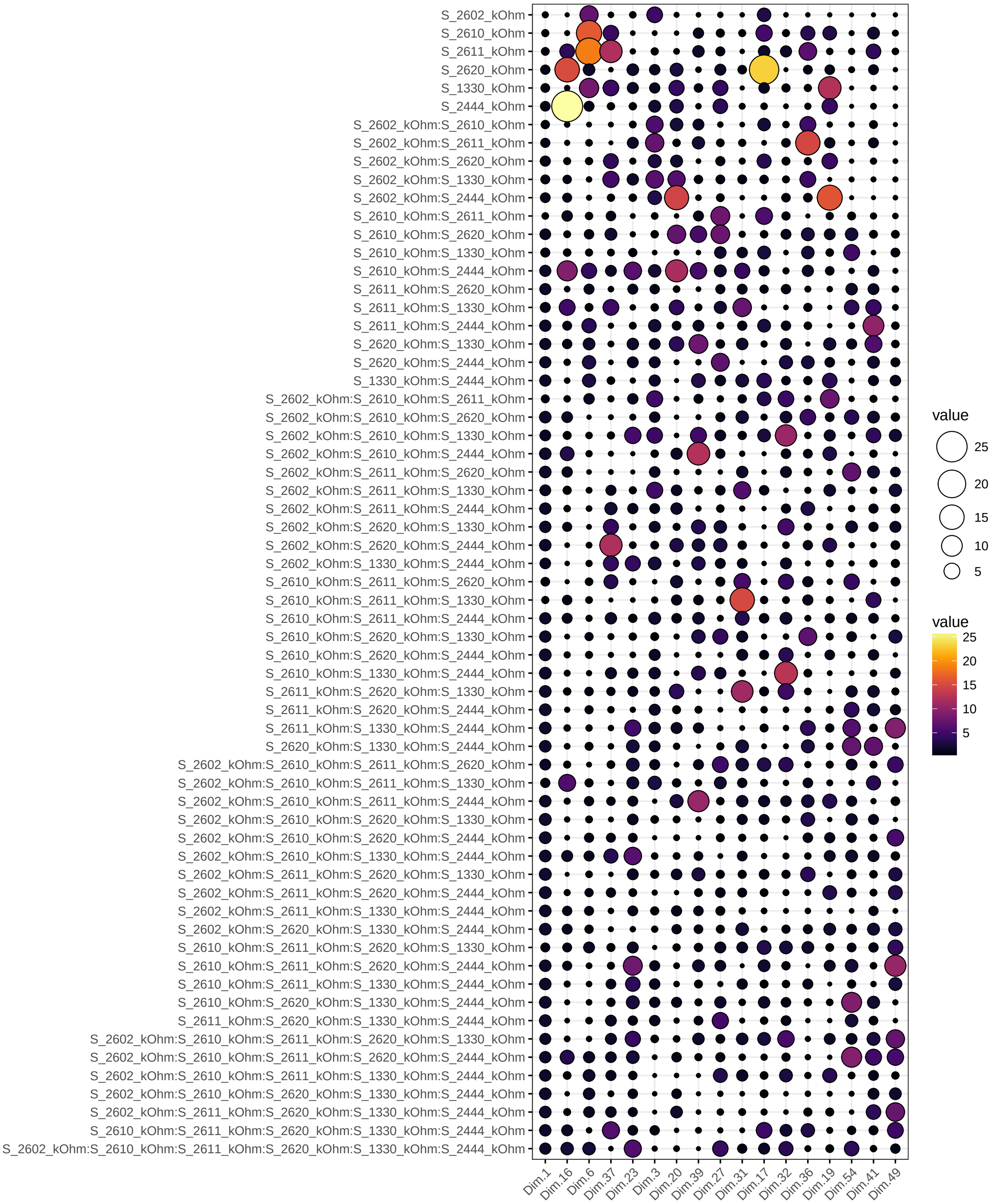


Figure 5.37: Contributions of interactions of sensors in LDA prediction [%]

5.5 Model diagnosis for the BENZ analyser

For the BENZ analyser, we select a threshold equal to 1 $\mu g/m^3$. It has to be remarked that the intoxication limit is equal to 0 $\mu g/m^3$, but the maximal concentrations has to be beneath 5 $\mu g/m^3$ [14].

5.5.1 Fitting linear discriminant analysis model over the complete data set

The model selection of the LDA model chooses 20 out of the 63 principal components. The evolution of the accuracy during the selection is represented in Figure 5.38. The accuracy starts lower compared to the three last model selections with a value of 0.55. Then, it increases constantly by steps for around 0.002 and reaches a final value of 0.6. It seems to be more difficult to predict the benzene concentration because more variables are selected compared to the precedent predictions and the accuracy remains smaller as well.

Prediction of signals and pollution events Figure 5.39 shows the LDA prediction on the real BENZ analyser's curve. The confusion matrix of these predictions is shown in Table 5.14. We see on this figure the benzene curve in levels, so the measurements reach the limit of detection of the BENZ analyser.

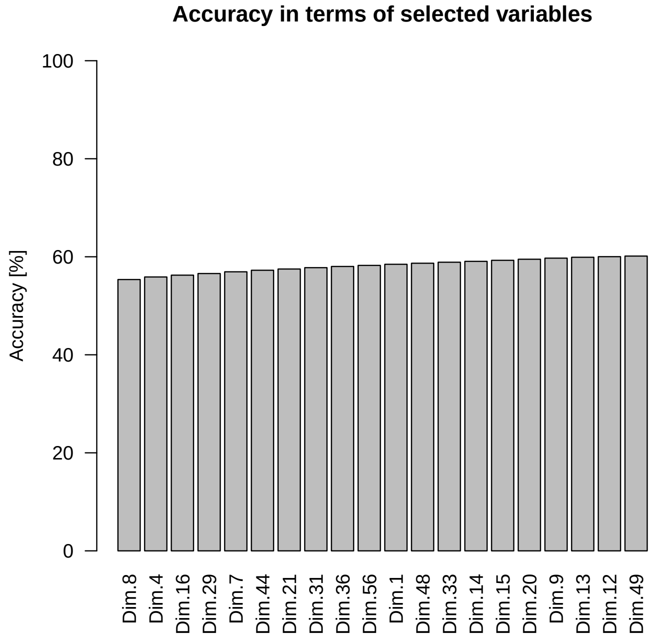


Figure 5.38: Accuracy evolution in the model selection

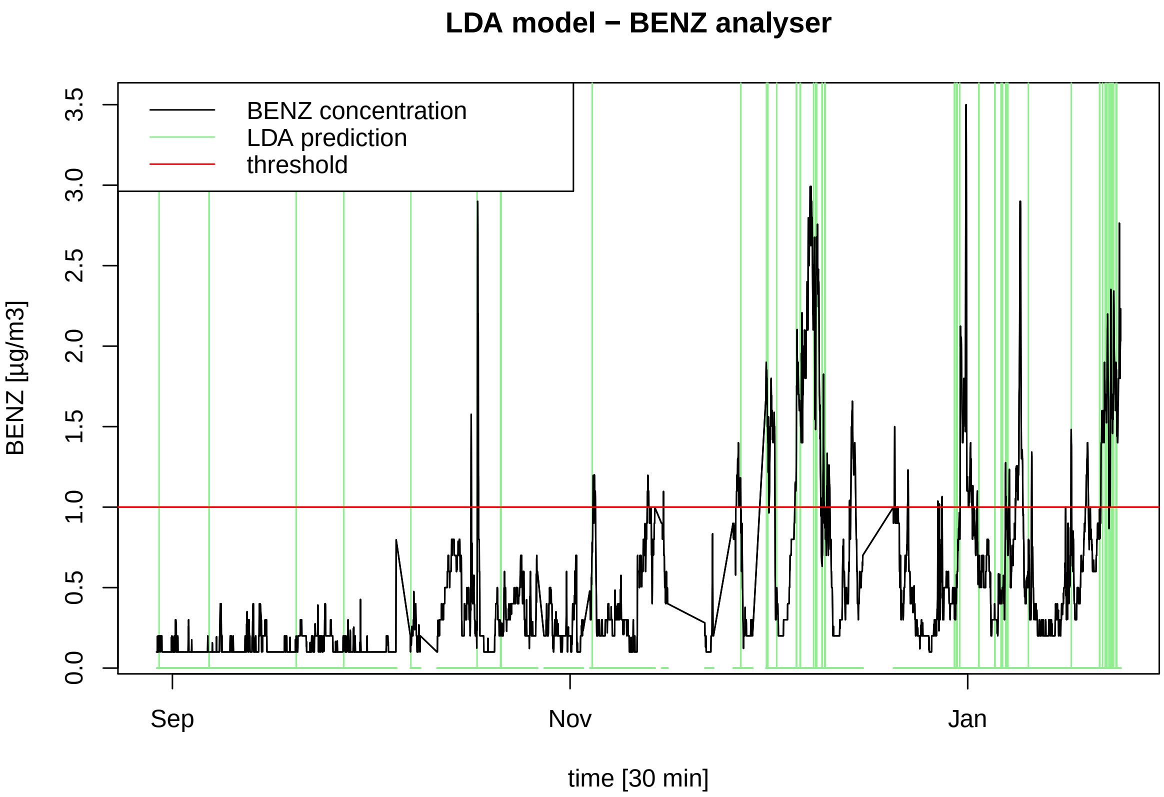


Figure 5.39: BENZ prediction with LDA model

Table 5.14: Confusion matrix of the LDA prediction for the complete data set

		Reference	
		noise	signal
Prediction	noise	5140	715
	signal	60	64

Prediction of signals and pollution events The confusion matrix shows that 715 out of 779 signals and only 60 out of 5200 noise values have been wrong predicted. This results in a very small sensitivity of 0.08, a specificity of around 0.99, a false positive rate of 0.01 and a false negative rate of 0.92. The false negative rate is extremely high. When we go back to Figure 5.39, we observe that many important peaks are detected at least once. The smaller signals with concentrations around 1.5 $\mu g/m^3$ are not always detected. Moreover, we observe false positives in September and October. In terms of pollution events, (Table 5.15) shows the number of detected and non detected events and (Table 5.16) the number of true and false predicted events.

Table 5.15: Detection of pollution events

	Detected	Non detected	Total
Pollution events	26	38	64

Table 5.16: Predicted pollution events

	True	False	Total
Predicted events	21	20	41

Among the 38 non detected events, there are 14 beneath 1.5 $\mu g/m^3$, so very near to the threshold of 1 $\mu g/m^3$. Concerning the false predicted events, 2 out of 20 false predicted events occur at moments, when the BENZ analyser did not work. The prediction takes place on the interpolated values, so it is not excluded that a pollution event actually occurred at that time.

False positive and negative predictions On Figure 5.40, the false signal and noise predictions are represented in green and red respectively. This figure confirms the results of the confusion matrix. There are detected signals, but not in a high amount. In contrary, the prediction of noise values shows only a minimal number of false positives. The unbalance in terms of signal and noise enforces this result. The visible gaps are present because of the missing values in the sensors data. Either the sensors did not work or odour bags were connected at this moments.

ROC Curve Figure 5.41 represents the ROC curve for the benzene concentrations. Naturally, the results already discussed before are confirmed in this figure. The curve increases more slowly in terms of the true positive rate (sensitivity), but the false positive rate is small. We observe that the curve approximates slightly the diagonal and therefore, the area under the curve is a bit less than for the other analysers, namely 0.76.

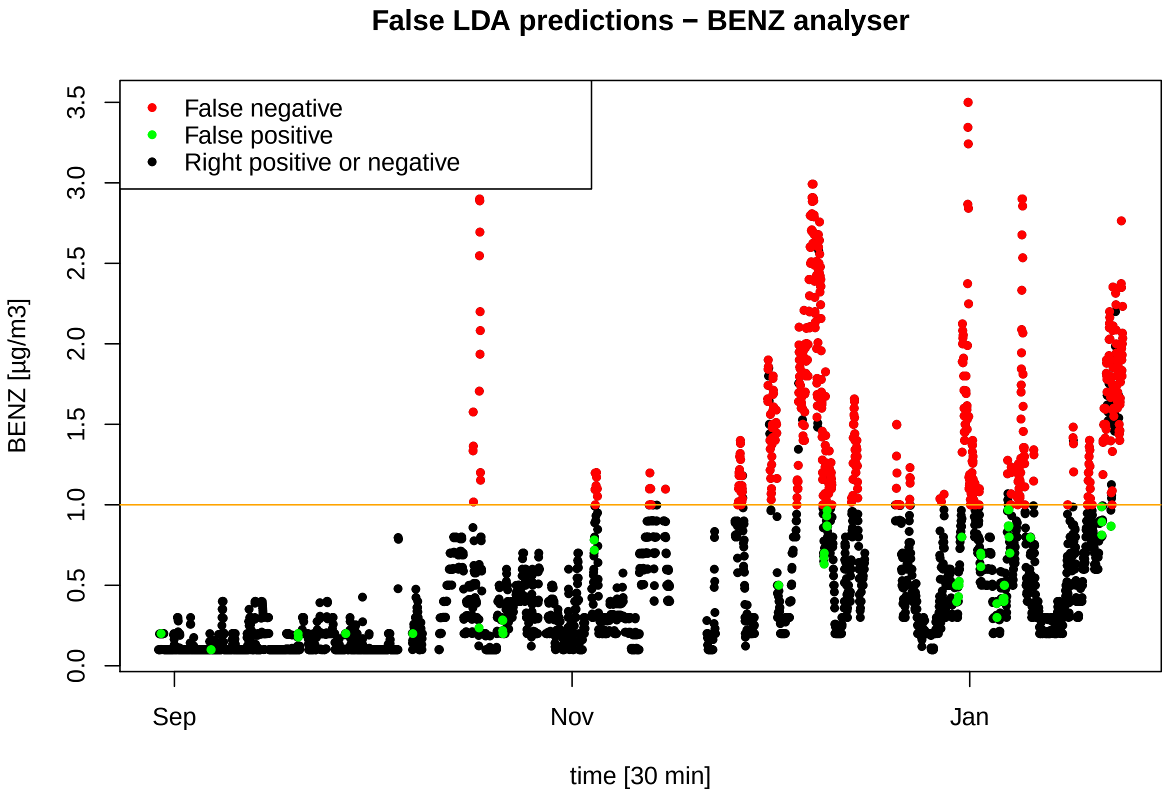


Figure 5.40: False predictions in the LDA model

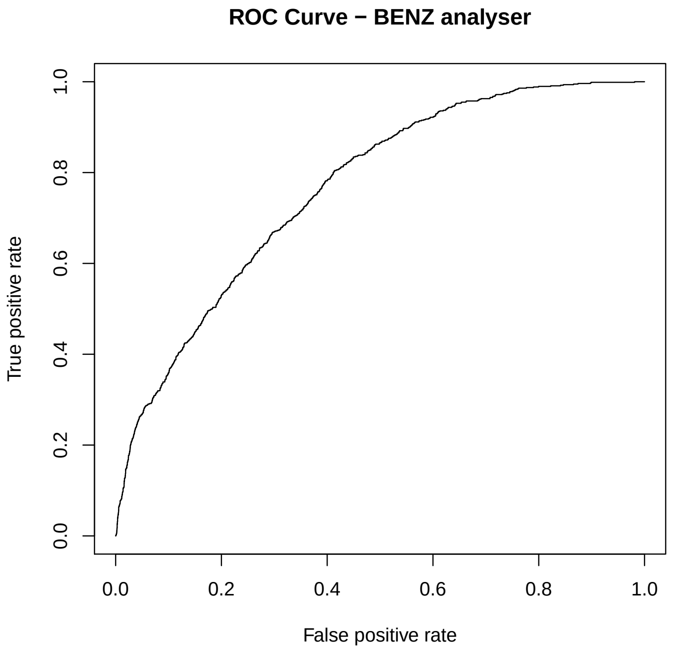


Figure 5.41: ROC curve of the LDA model
over the complete data set

5.5.2 Comparison of the LDA and MLR model predictions

The predictions of the LDA and MLR model are illustrated with the real BENZ concentrations in Figure 5.42. We observe that the LDA and MLR model provide very similar predictions. When the LDA model predicts a false positive, the MLR prediction was too high likewise. Moreover, signals that have not been detected by the LDA model are underestimated by the MLR model as well.

The evolution of the adjusted R squared is illustrated in Figure 5.43. We observe an increase from circa 0.05 up to 0.24.

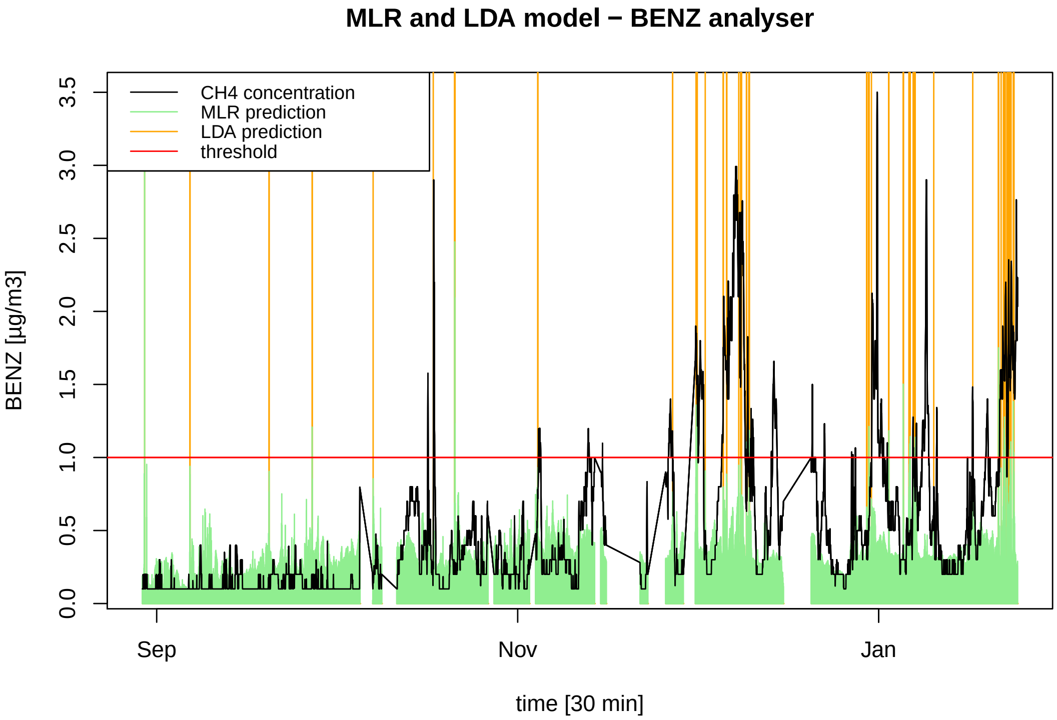


Figure 5.42: LDA and MLR predictions for the BENZ analyser

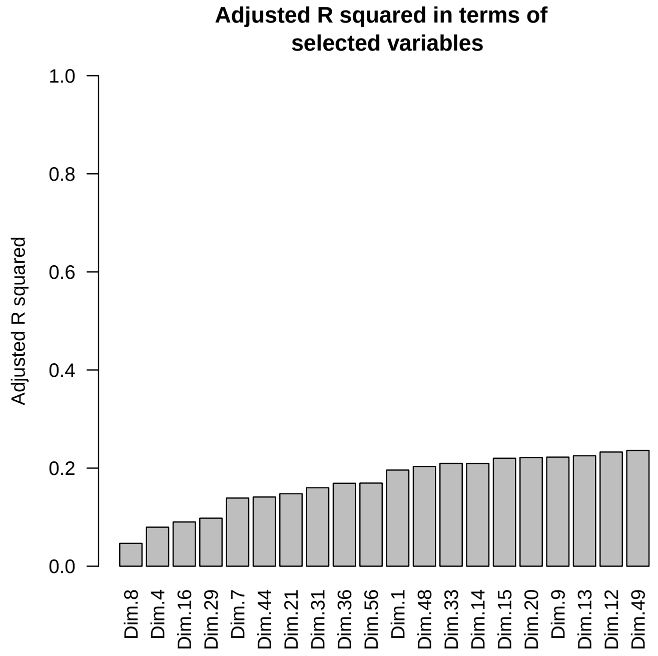


Figure 5.43: Adjusted R squares in terms of the selected variables

5.5.3 Cross-validation with random split

Table 5.17 represents the minimum, mean and maximum of the sensitivities, specificities and adjusted R squares from the cross-validation with 20 executions.

Table 5.17: Range and mean over the sensitivities, specificities and adjusted R squares

	Min	**Mean**	**Max**
Sensitivity	0.03	0.08	0.11
Specificity	0.975	0.987	0.993
Adjusted R^2	0.21	0.25	0.29

The sensitivity presents its very small but stable values between 0.03 and 0.11. The specificity is very high and also stable with a mean of 0.987. The adjusted R squared goes from 0.21 to 0.29 and remains also stable.

Figure 5.10 represents the twenty ROC curves in the cross-validation of the LDA model. These curves present the same trend and confirm the strong stability of sensitivity and specificity.

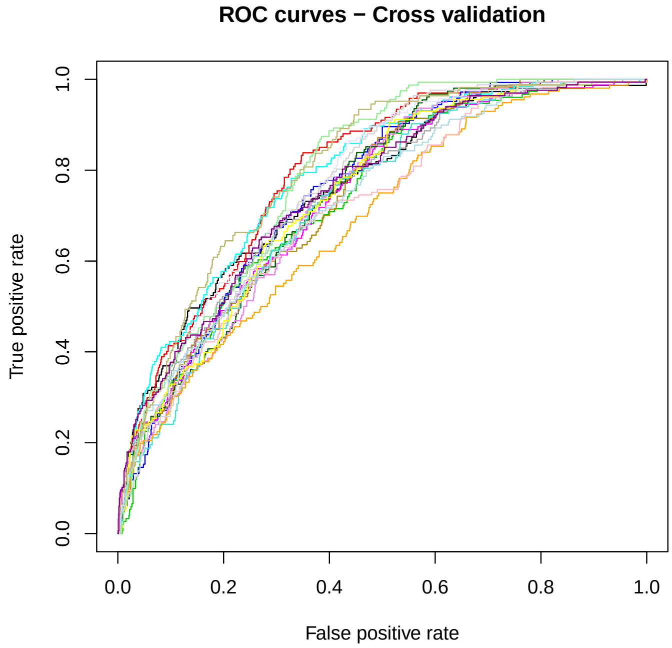


Figure 5.44: ROC curves in the BENZ prediction

5.5.4 Prediction with split of the time series

On the following Figure 5.45 , the prediction of the training set by the LDA model in blue, the one of the test set in green. The prediction of the MLR model with the selected variables from the LDA model is also represented. These prediction values are shown in pink, the test set in orange.

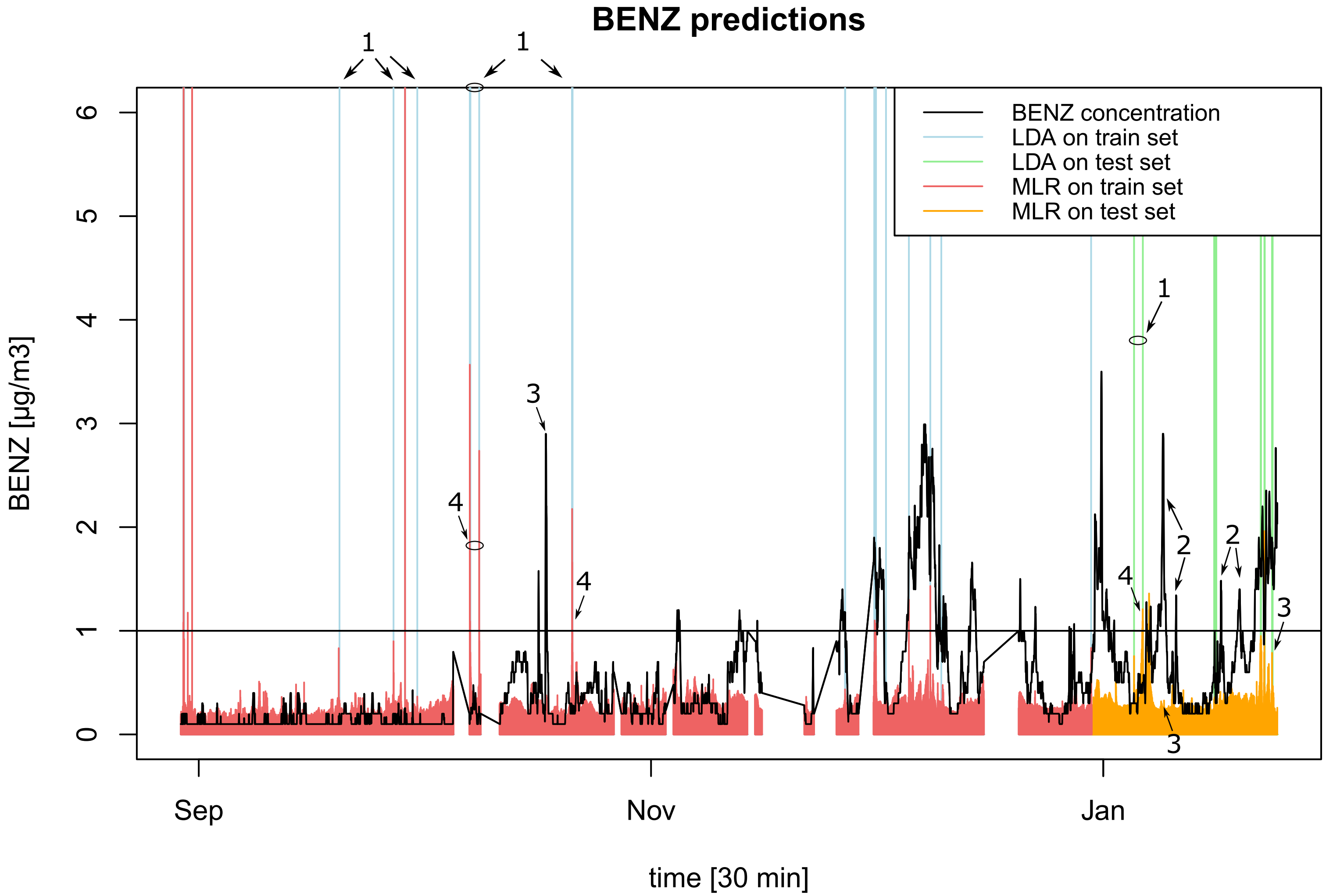


Figure 5.45: Predictions by MLR and LDA methods

Observations of the predictions The following observations are indicated by the respective numbers in Figure 5.45:

1. We observe false signals of the LDA prediction in the training and test set. Among the 6 false predicted events, one is occurring at a moment when the BENZ analyser did not work. It is possible that the sensors have detected a pollution event while the analyser did not work.

2. There are some signals that are not detected in the validation set.

3. The signal predictions of the MLR method have to small signals when there are true signals in the training and test set.

4. In contradiction, when the analysers measurements are beneath the threshold, the MLR predictions are often higher than the reference values.

5.5.5 Discussion of residuals and false predictions

In terms of the response variable

The following Figure 5.46 represents the residuals in terms of the logarithmic benzene values. The red coloured points correspond to false negative predictions, the green ones to false positive predictions provided by the LDA model. We see that all false negative observations (red) have a positive residual term. Thus, the prediction is smaller than the observation and the real signal is predicted as noise. This is also enforced by the strong unbalance in the number of signal and noise observations. For the false positive observations (green), the contrary takes place. Most of the predicted values are higher than the real measurements and give false signal predictions. Moreover, the high number of false negatives and the small number of false positives is visible in this figure.

In terms of the control parameters

We observe a weak dependence between control parameter and residuals for the relative humidity and the wind velocity. A high humidity and little velocity measurements lead to high residuals, thus to bad predictions. This dependence remains however very slight and the other control parameters have not at all a dependence on the false predictions of benzene.

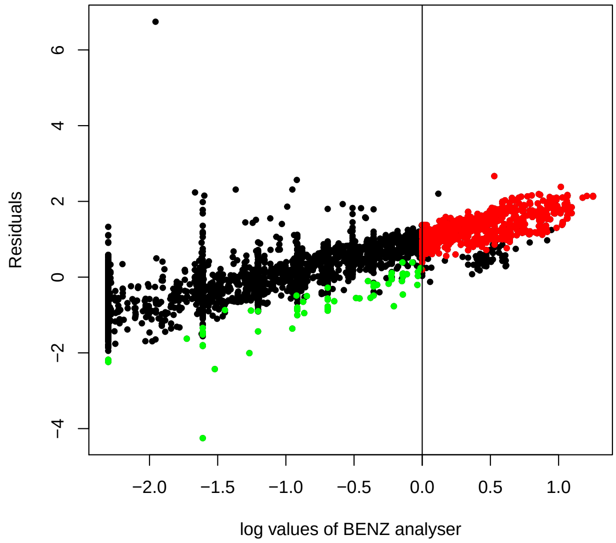


Figure 5.46: Residuals of the MLR model in terms of log(A_BENZ)

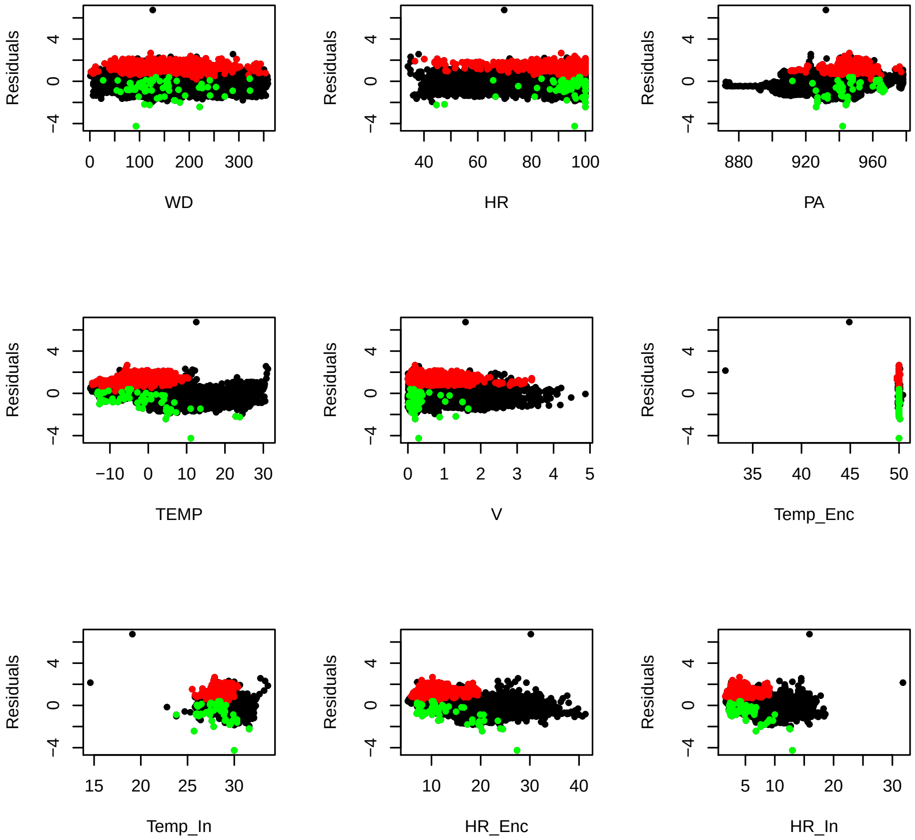


Figure 5.47: Residuals of the MLR model in terms of the control parameters

5.5.6 Which interactions of sensors contribute to the prediction?

The contributions of every interaction of sensors in the selected principal components in the LDA model are represented in Figure 5.48. The size and the lightness of the circles are proportional to the contribution percentages. The first selected principal component is Dim.8 showing up the most important contribution of over 30% for the sensor TGS2602. Moreover, the interactions in higher orders containing the sensor TGS2602 are also more contributing than those without this sensor. This can be observed in the second selected variable too. Thus, our hypothesis that there is information in a combination of sensors signals is confirmed. The principal component Dim.16 presents a contribution of 26% for the sensor TGS2444 and one of 15% for the sensor TGS2620. The greatest contribution with a value of 37.9% is present in the principal component Dim.7 and corresponds also to the sensor TGS2602.

The sensor TGS2602, which is the most present in the contribution figure, announces a selectivity for VOC, hydrogen sulphide and ammoniac and the sensor TGS2444 for ammoniac. These are compounds occurring during the emission of waste and compost.

The benzene compound arises typically from traffic. It is possible that trucks transporting waste implicate that the benzene could be accompanied by VOC, H_2S and NH_3. Another reason for the accompanied compounds of benzene could be the location of the compost and waste between the highway and the measuring station [14].

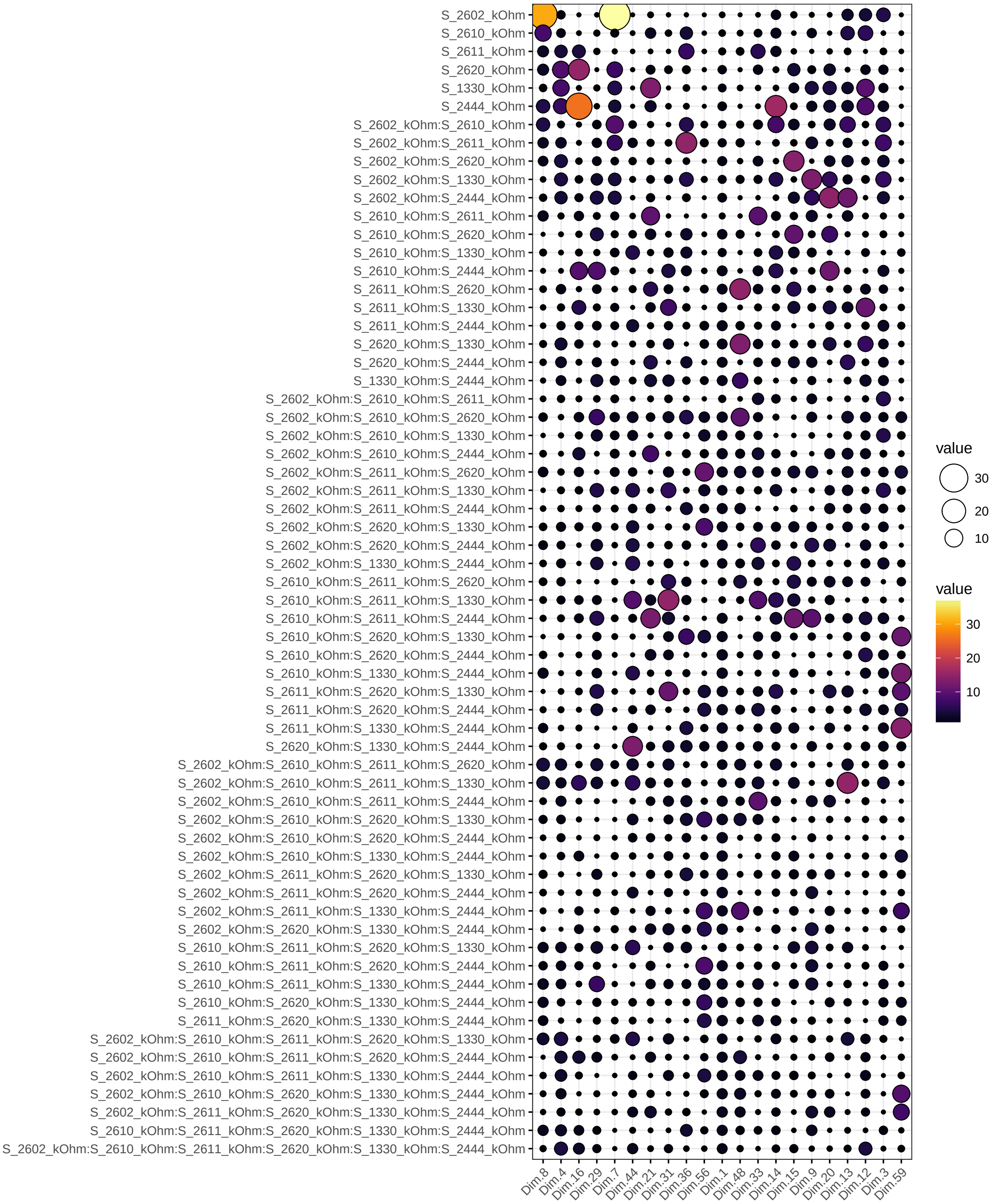


Figure 5.48: Contributions of interactions of sensors in LDA prediction [%]

5.6 Model diagnosis for the TOLU analyser

The threshold of 0.8 $\mu g/m^3$ is chosen for the prediction model of toluene regarding the range of its signals in this data set. With this threshold, we have 720 signal and 5289 noise values.

5.6.1 Fitting linear discriminant analysis model over the complete data set

For the toluene prediction, 17 variables have been chosen in the model selection. Figure 5.49 represents the evolution of the accuracy when the 17 principal components are added one by one. The addition of the first variable Dim.8 returns an accuracy equal to 0.59. Then, the accuracy increases up to 0.63 very slowly.

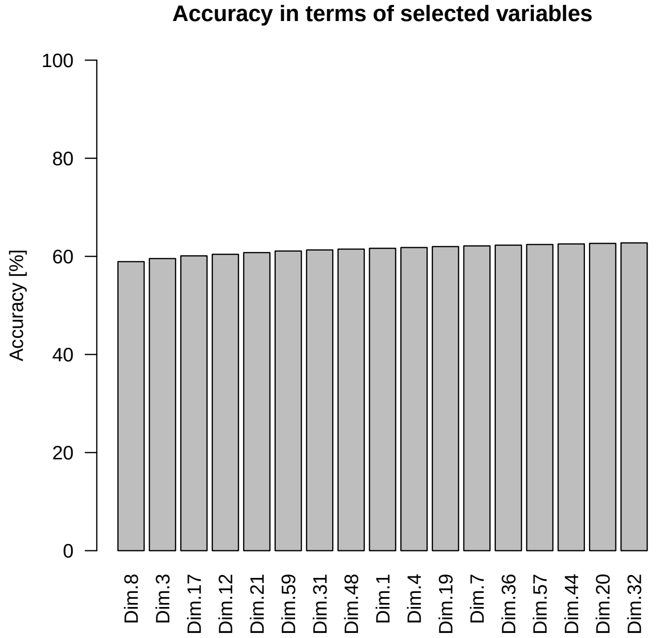


Figure 5.49: Accuracy evolution in the model selection

Selected principal components

Prediction of signals and pollution events The LDA prediction with the real TOLU measurements is illustrated on Figure 5.50. The confusion matrix of these predictions is shown in Table 5.18. As for the BENZ analyser, we observe that the limit of detection for the TOLU analyser is reached.

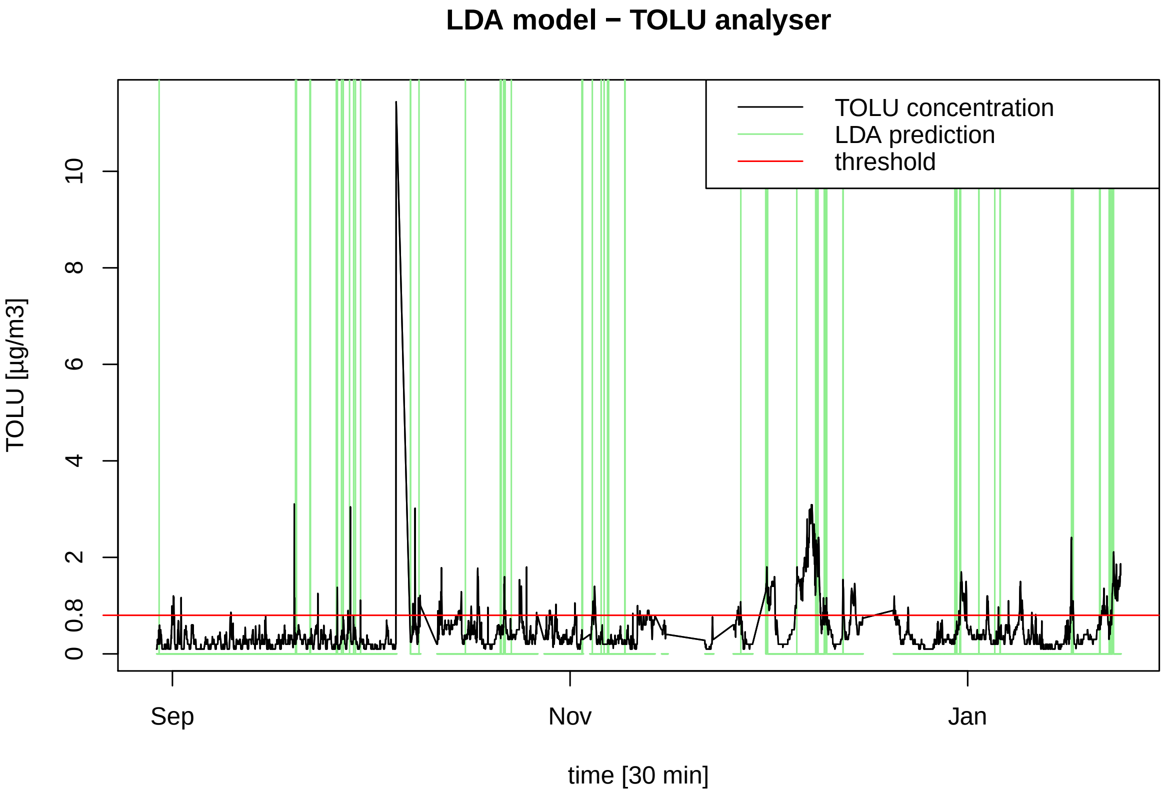


Figure 5.50: TOLU prediction with LDA model

Table 5.18: Confusion matrix of the LDA prediction for the complete data set

		Reference	
		noise	signal
Prediction	noise	5177	648
	signal	82	72

We observe that 82 out of 5177 noise and 72 out of 648 signal values have been false predicted. In conclusion, the sensitivity equals 0.1, the specificity equals 0.98, the false positive rate has a value of 0.02 and the false negative rate a value of 0.9. The sensitivity is very small and the specificity extremely high. Going back to Figure 5.50 we observe that several higher signals are nevertheless detected at least once, but many smaller ones are not detected. We count the number of detected and non detected events (Table 5.19) and the number of true and false predicted events (Table 5.20) for a better comprehension of the prediction.

Table 5.19: Detection of pollution events

	Detected	Non detected	Total
Pollution events	28	56	84

Table 5.20: Predicted pollution events

	True	False	Total
Predicted events	27	24	51

Among the 56 non detected events, there are 35 beneath 1 $\frac{\mu g}{m^3}$. Moreover, there are 2 out of the 24 false predicted pollution events occurring when the TOLU analyser did not work and values are taken from the interpolation. It is possible, that an event occurred at that time.

False positive and negative predictions In Figure 5.51, the false positive and negative predictions are represented in red and green respectively. In each of the higher peaks remain black points. Thus, every pollution event is detected at least once. Moreover, there are only a few false positive predictions. The gaps, most visible at the end of November and in December, are due to missing sensors values by the same reasons: they did not work or odour bags were connected.

ROC Curve Figure 5.52 illustrates the ROC curve for the toluene prediction. The examination of this curve confirms the results already developed before. The area under the ROC curve is equal to 0.77. The curve behaves well because of the extremely high specificity.

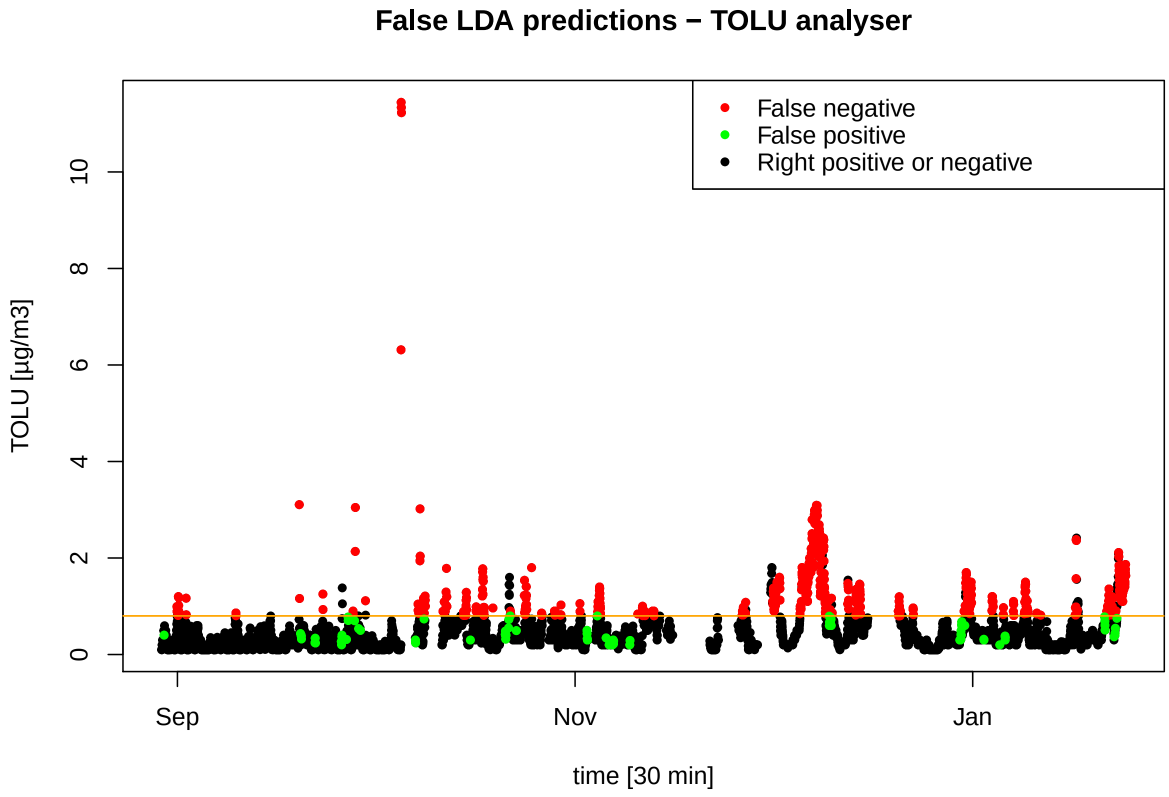


Figure 5.51: False predictions in the LDA model

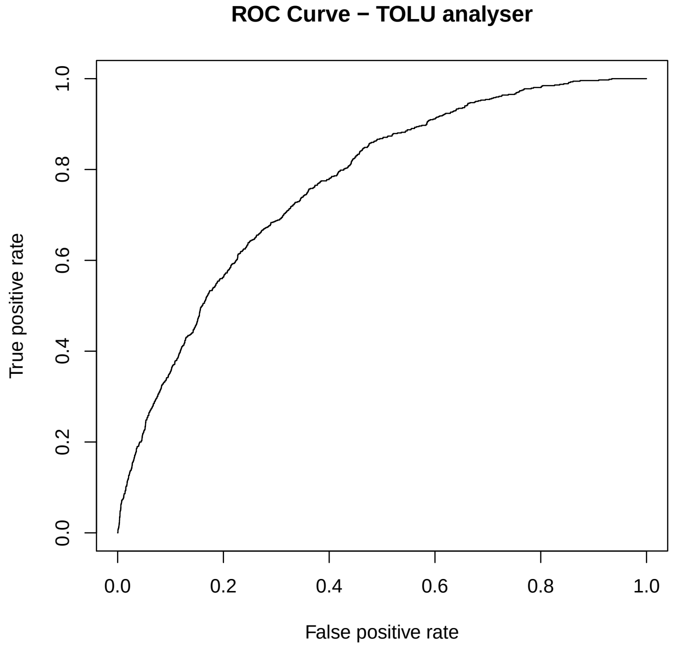


Figure 5.52: ROC curve of the LDA model over the complete data set

5.6.2 Comparison of the LDA and MLR model predictions

In the following Figure 5.53, the predictions of the LDA and MLR model are superposed. We observe that the LDA and MLR model provide very similar predictions. When a signal has not been detected by the LDA model, the MLR predictions are underestimating the true concentration as well. Furthermore, when the LDA model predicts a false positive, the MLR prediction was too high likewise.

The evolution of the adjusted R squared is illustrated in Figure 5.54. We can observe an increase from circa 0.09 up to 0.21.

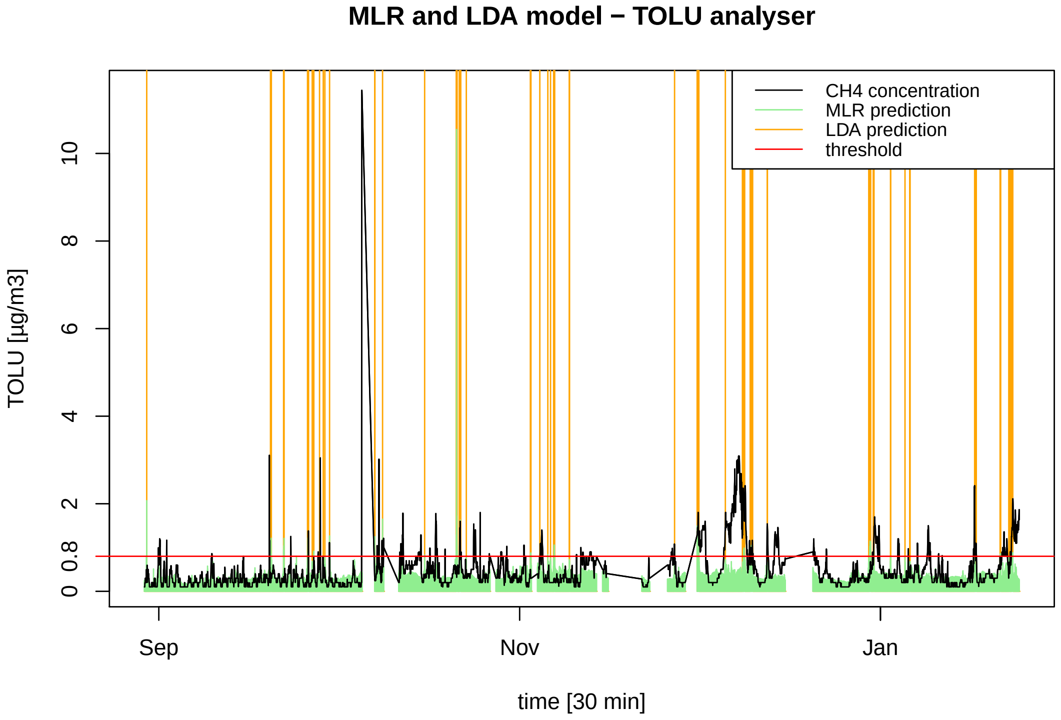


Figure 5.53: LDA and MLR predictions for the TOLU analyser

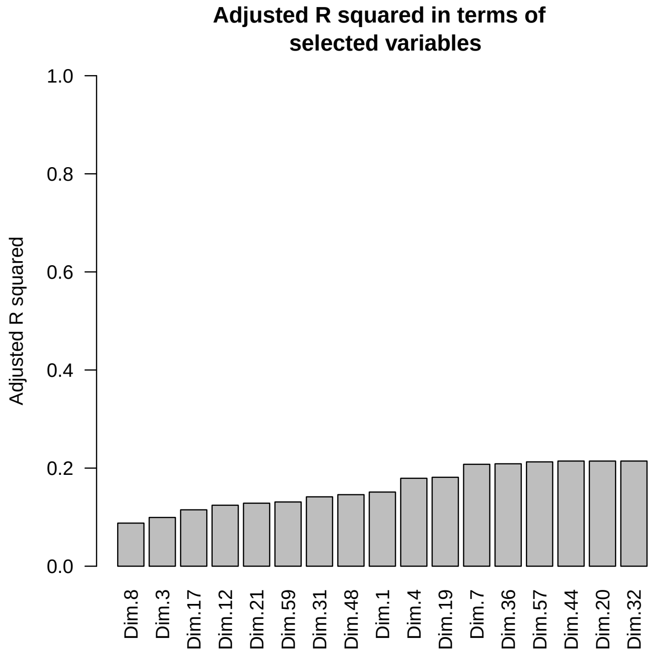


Figure 5.54: Adjusted R squares in terms of the selected variables

5.6.3 Cross-validation with random split

The minimum, mean and maximum of the sensitivities, specificities and adjusted R squares from the cross-validation with 20 executions are represented in Table 5.21.

Table 5.21: Range and mean over the sensitivities, specificities and adjusted R squares

	Min	**Mean**	**Max**
Sensitivity	0.03	0.09	0.15
Specificity	0.97	0.98	0.99
Adjusted R^2	0.18	0.21	0.24

The sensitivity is stable between 0.03 and 0.15. The specificity is as before very high and stable. The adjusted R squared is almost stable around the mean of 0.21.

Figure 5.55 represents the twenty ROC curves in the cross-validation of the LDA model.

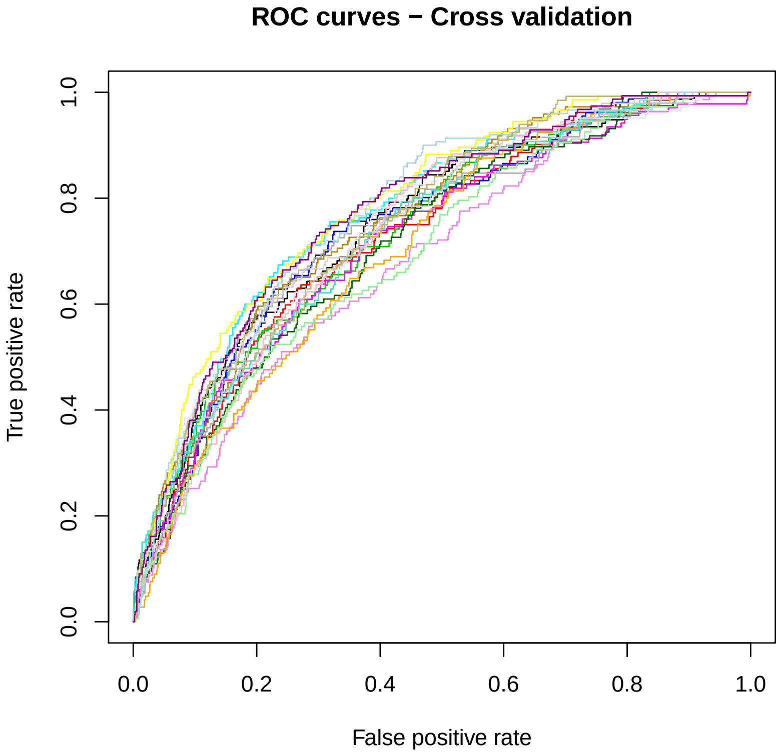


Figure 5.55: ROC curves in the TOLU prediction

5.6.4 Prediction with split of the time series

On the following Figure 5.56 , the prediction of the training set by the LDA model in blue, the one of the test set in green. The prediction of the MLR model with the selected variables from the LDA model is also represented. These prediction values are shown in pink, the test set in orange.

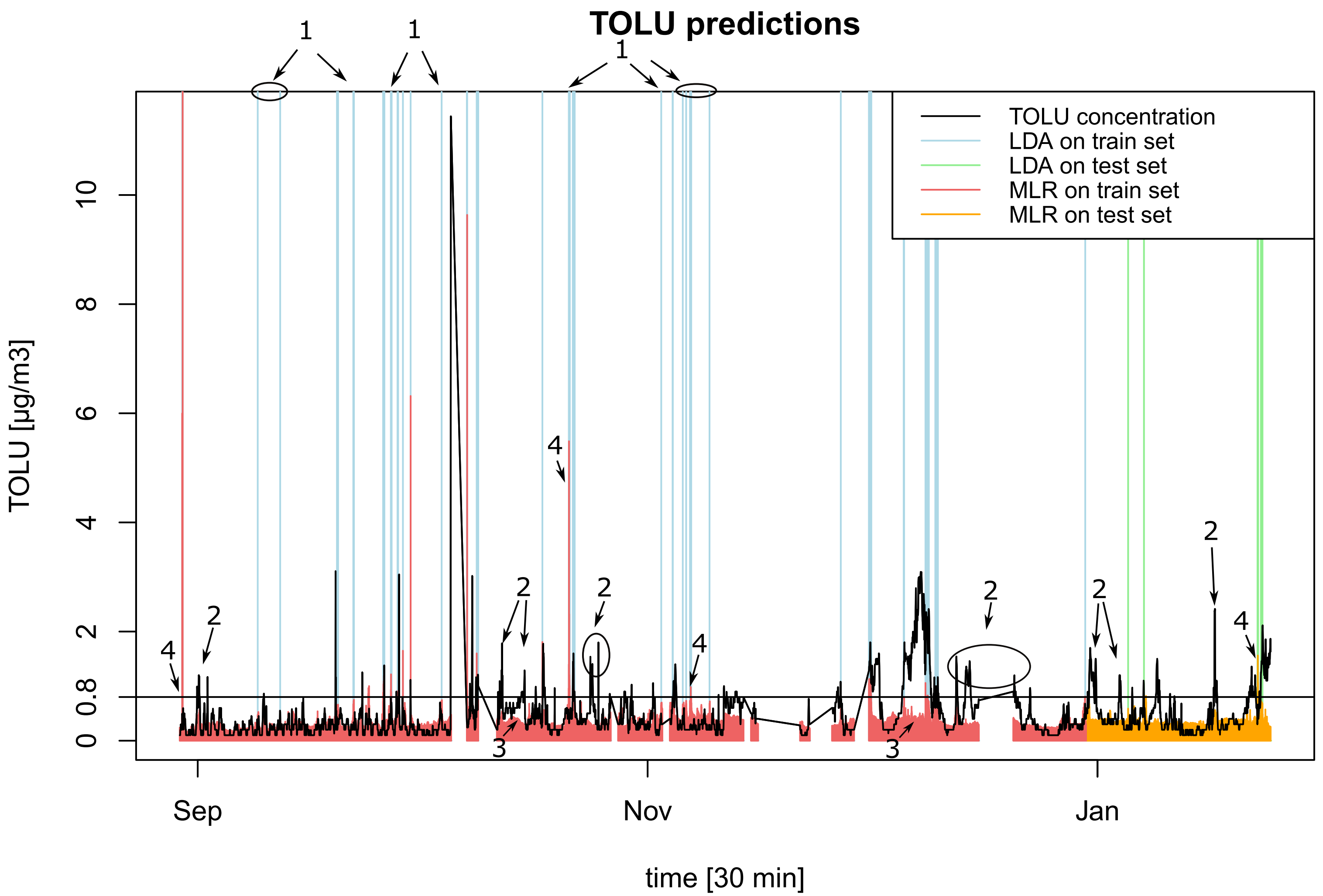


Figure 5.56: Predictions by MLR and LDA methods

Observations of the predictions The following observations are indicated by the respective numbers in Figure 5.56:

1. We observe several false signals of the LDA prediction in the training and test set.
2. There are some signals that are not detected in the training and validation set by the LDA model.
3. The signal predictions of the MLR method have to small signals when there are true signals in the training and test set.
4. When the analysers signal is beneath the threshold, the MLR prediction sometimes overestimates these measures.

5.6.5 Discussion of residuals and false predictions

In terms of the response variable

The following Figure 5.57 represents the residuals in terms of the logarithmic toluene values. The red coloured points correspond to false negative predictions, the green ones to false positive predictions provided by the LDA model. We see that all false negative observations (red) have a positive residual term. Thus, the prediction is smaller than the observation and the real signal is predicted as noise. This is also enforced by the strong unbalance in the number of signal and noise observations. For the false positive observations (green), the contrary takes place. Most of the predicted values are higher than the real measurements and give false signal predictions.

In terms of the control parameters

We represent the residuals in terms of these control parameters in Figure 5.58. We observe a weak dependence between control parameter and residuals for the relative humidity and the wind velocity. A high humidity and little velocity measurements lead to higher residuals, thus to bad predictions. This dependence remains however very slight and the other control parameters have not at all a dependence on the false predictions of toluene.

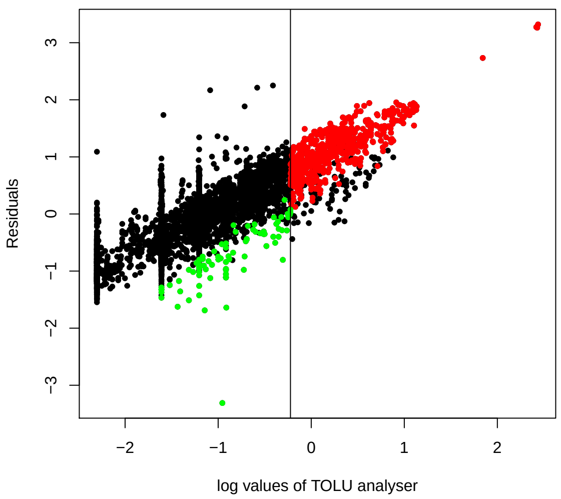


Figure 5.57: Residuals of the MLR model in terms of log(A_TOLU)

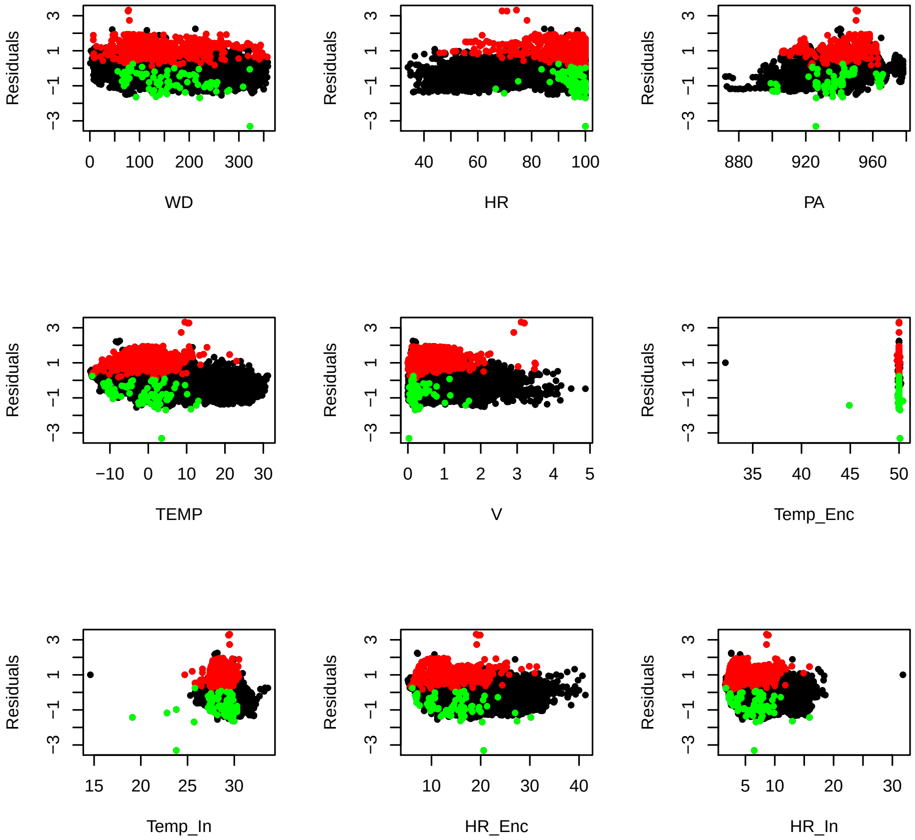


Figure 5.58: Residuals of the MLR model in terms of the control parameters

5.6.6 Which interactions of sensors contribute to the prediction?

Figure 5.59 illustrates the contributions of every sensors interaction with a circle. A big and light circle indicates an important contribution to the prediction of toluene. The first selected principal component Dim.8 contains a contribution of 30.7% for the TGS2602 sensor. The interactions in higher orders containing this sensor present also higher contributions than those without it. In the second selected variable Dim.3, the interactions of order two TGS2602 · TGS2610 and TGS2602 · TGS2611 show up the most important contributions. In the third component Dim.17, the TGS2620 sensor occurs with 23%. The highest contribution of around 37.9% is present for the TGS2602 sensor in Dim.7. Often, the interactions containing the TGS2611 sensor show up higher contributions as well. We observe that many interactions in higher orders contribute to the prediction which supports our hypothesis that there is information in a combination of sensor signals.

Regarding the announced compounds selectivities of the sensors, the sensor TGS2602 is expected to detect VOC, hydrogen sulphide and ammoniac, the sensor TGS2620 for organic solvents, alcohols and carbon monoxide. The sensor TGS2602 sensor shows up the most important contributions in the toluene prediction likewise for the benzene prediction. This could be explained by the high correlation (equal to 0.72) between the toluene and benzene measurements (see Figure 5.15). So, the same reasons as for the benzene analyser persist.

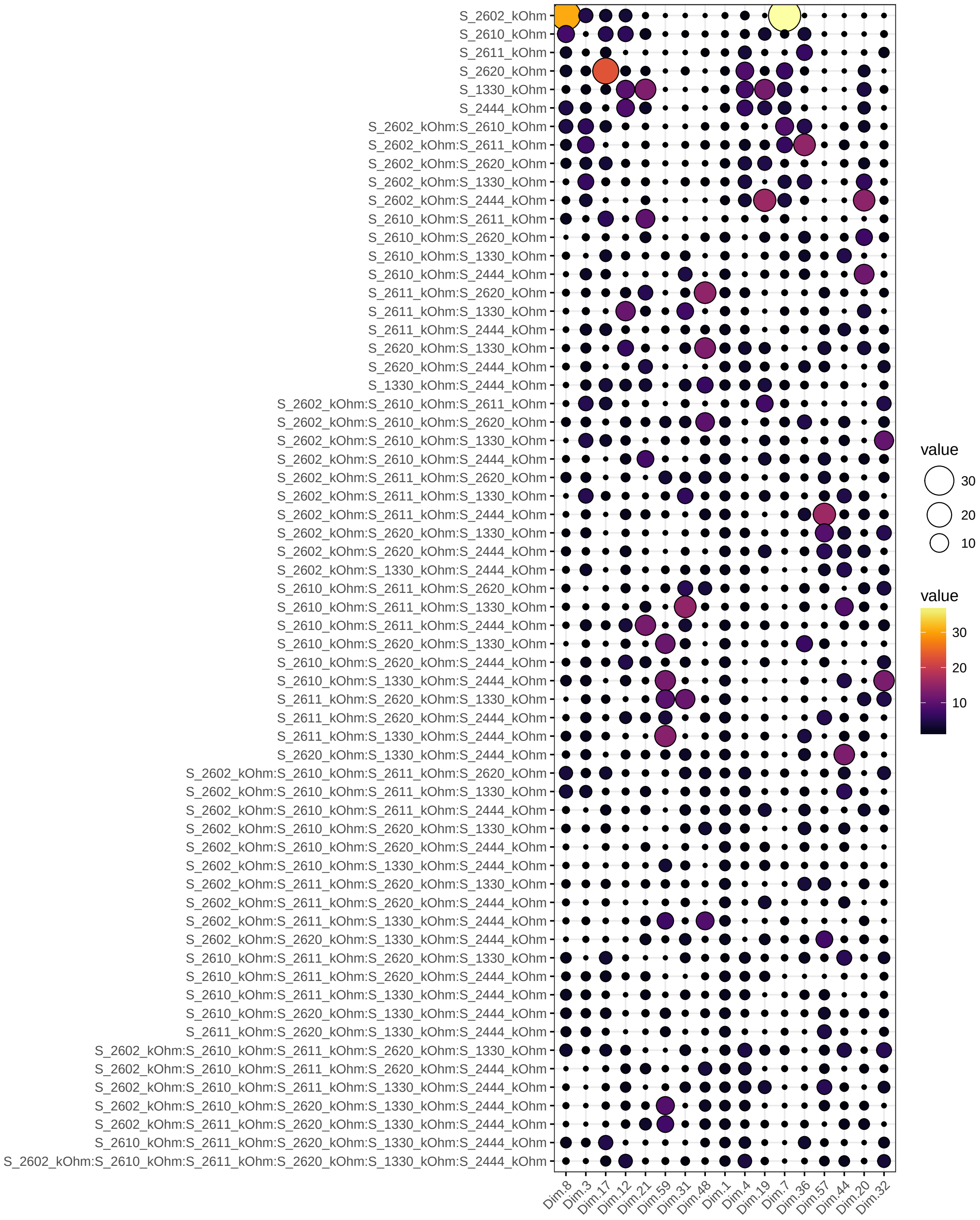


Figure 5.59: Contributions of interactions of sensors in LDA prediction [%]

5.7 Model diagnosis for the LIMO analyser

For the limonene analyser, we select a threshold equal to 1 $\mu g/m^3$ regarding the range of its signals in this data set.

5.7.1 Fitting linear discriminant analysis model over the complete data set

Selected principal components The model selection of the LDA model chooses 11 out of the 63 principal components. The evolution of the accuracy during the selection is represented in Figure 5.60. The accuracy presents a high starting value as the CH_4 analyser. The first accuracy equals 0.87. Then, it increases constantly by slight steps until reaching a final value of 0.9.

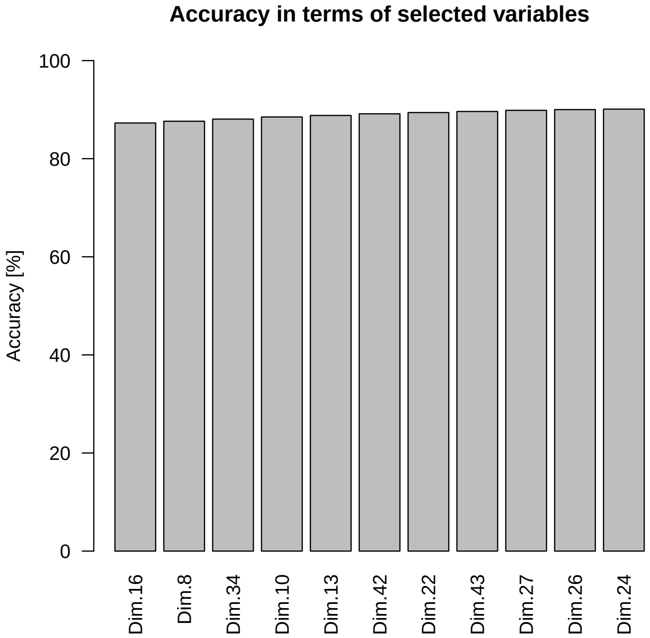


Figure 5.60: Accuracy evolution in the model selection

Prediction of signals and pollution events Figure 5.61 shows the LDA prediction on the real LIMO analyser's curve. The confusion matrix of these predictions is shown in Table 5.22.

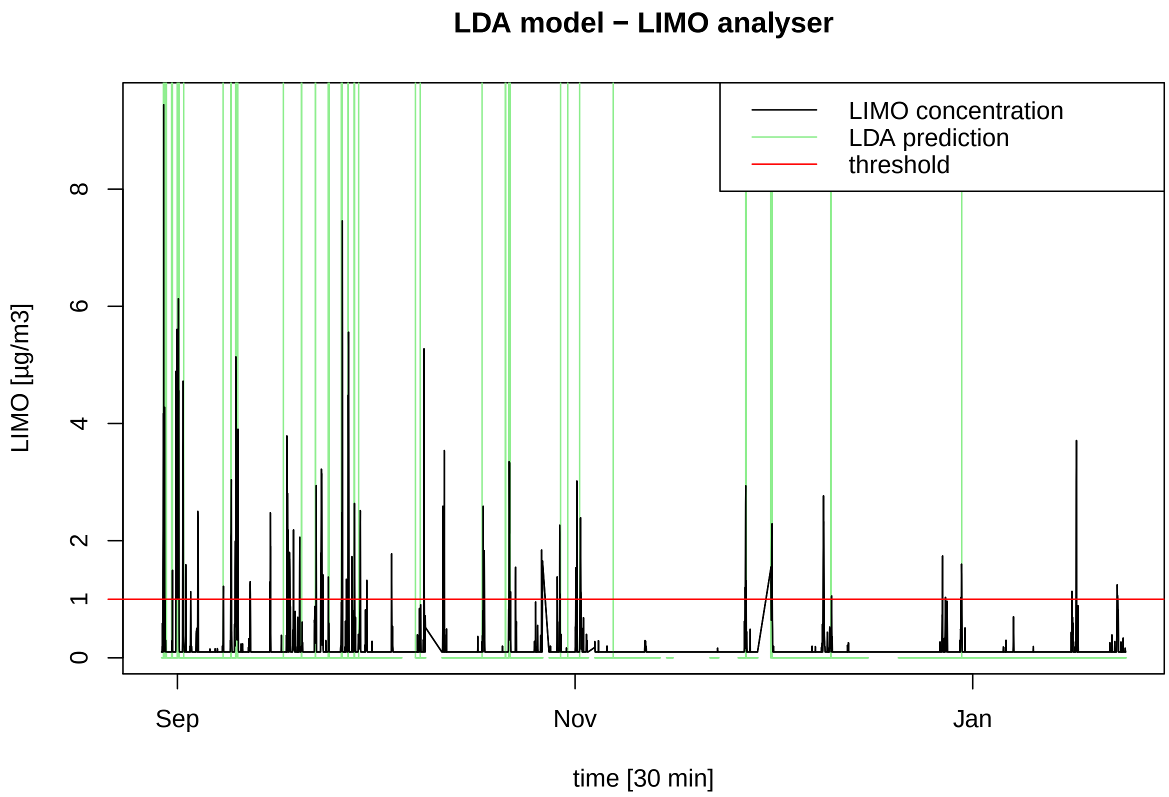


Figure 5.61: LIMO prediction with LDA model

Table 5.22: Confusion matrix of the LDA prediction for the complete data set

		Reference	
		noise	signal
Prediction	noise	5713	156
	signal	50	60

The confusion matrix shows that 156 out of 216 signals and only 50 out of 5713 noise values have been wrong predicted. This results in a sensitivity of 0.28, a specificity of around 0.99, a false positive rate of 0.01 and a false negative rate of 0.72. When we go back to Figure 5.61, we observe that almost all high peaks are detected at least once. The smaller signals with concentrations under 2 $\mu g/m^3$ are not always detected. Moreover, we do not observe as much false positives in the limonene prediction. In terms of pollution events, (Table 5.23) shows the number of detected and non detected events and (Table 5.24) the number of true and false predicted events.

Table 5.23: Detection of pollution events

	Detected	Non detected	Total
Pollution events	26	34	60

Table 5.24: Predicted pollution events

	True	False	Total
Predicted events	27	10	37

Among the 34 non detected events, there are 14 beneath 1.5 $\mu g/m^3$, so very near to the threshold of 1 $\mu g/m^3$. Concerning the false predicted events, 2 out of the 10 false predicted events occur at moments, when the LIMO analyser did not work. The prediction takes place on the interpolated values, so it is not excluded that a pollution event actually occurred at that time.

False positive and negative predictions On Figure 5.62, the false signal and noise predictions are represented in green and red respectively. This figure confirms the results of the confusion matrix. In the important peaks are always black points located, so the pollution event is detected. Like for all the other analysers, we observe only a little amount of false positive predictions. The gaps in this figure are due to missing values in the sensors data due to a non functioning or the connection of odour bags.

ROC Curve Figure 5.63 represents the ROC curve for the limonene concentrations. Naturally, the results already discussed before are confirmed in this figure. Although the true positive rate (sensitivity) is not so strong, the curve behaves not so bad. Remember that the optimal curve would have the maximal area under the curve equal to 1. For the LIMO prediction, the area under the curve equals 0.87, which is a good result.

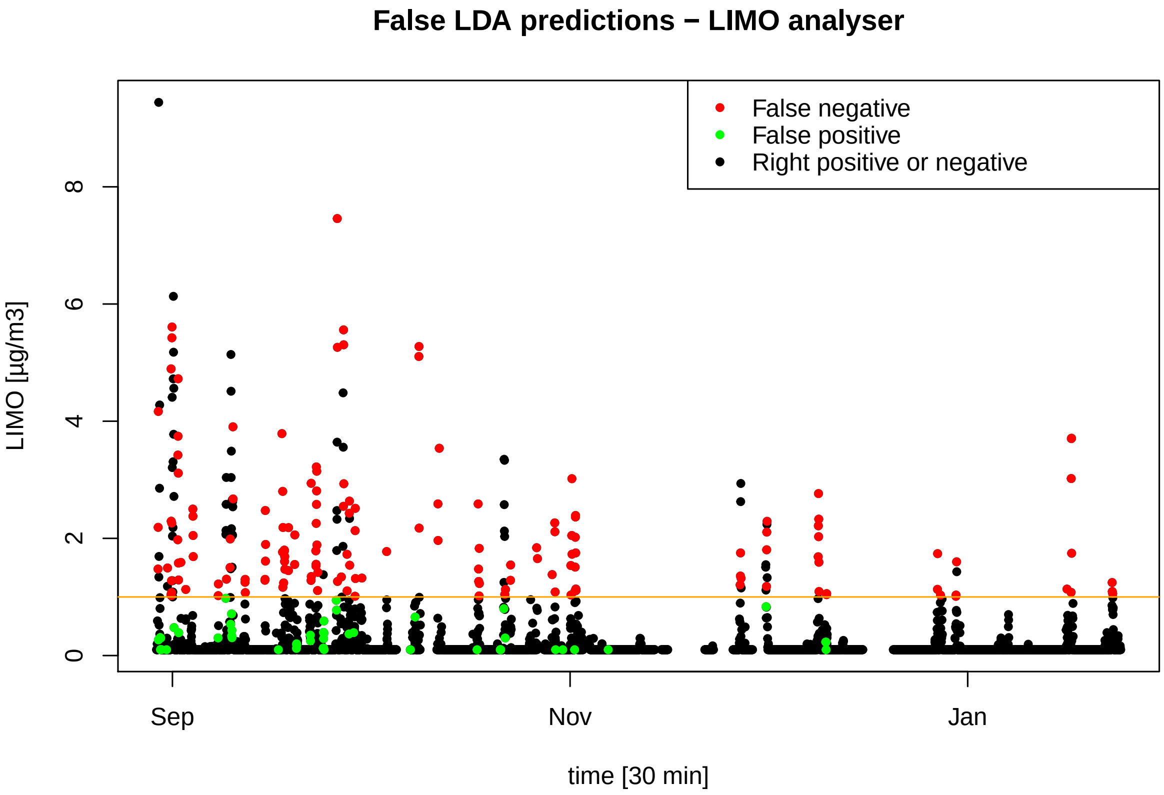


Figure 5.62: False predictions in the LDA model

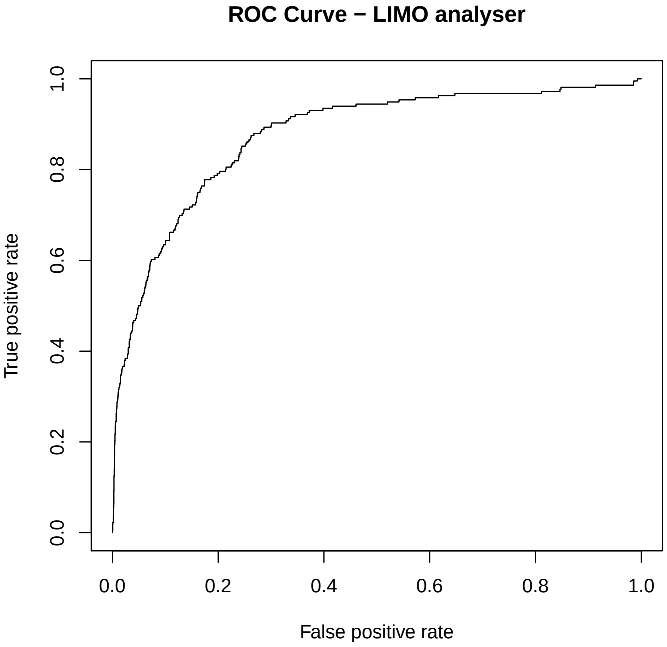


Figure 5.63: ROC curve of the LDA model over the complete data set

5.7.2 Comparison of the LDA and MLR model predictions

The predictions of the LDA and MLR model are illustrated with the real LIMO concentrations in Figure 5.64. We observe that the LDA and MLR model provide very similar predictions. When the LDA model predicts a false positive, the MLR prediction was too high likewise. Moreover, signals that have not been detected by the LDA model are underestimated by the MLR model as well.

The evolution of the adjusted R squares is illustrated in Figure 5.65. We observe an increase from circa 0.04 up to 0.18, which is not very high.

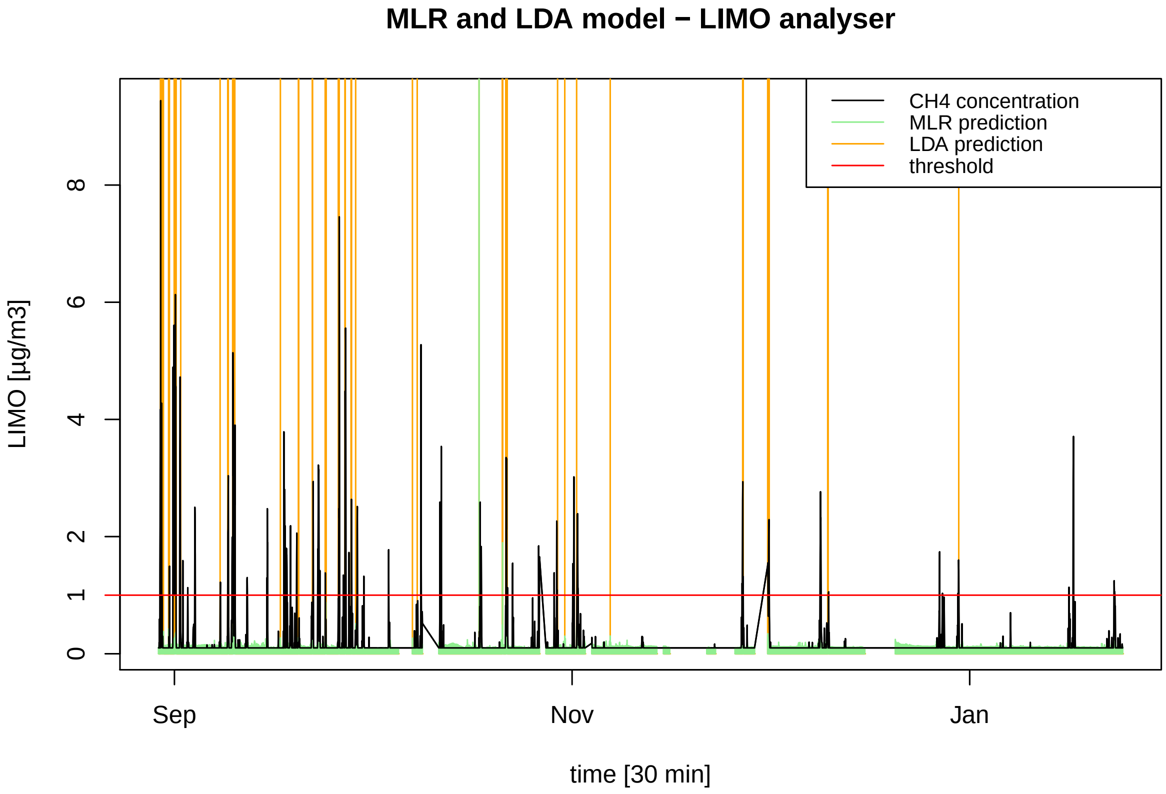


Figure 5.64: LDA and MLR predictions for the LIMO analyser

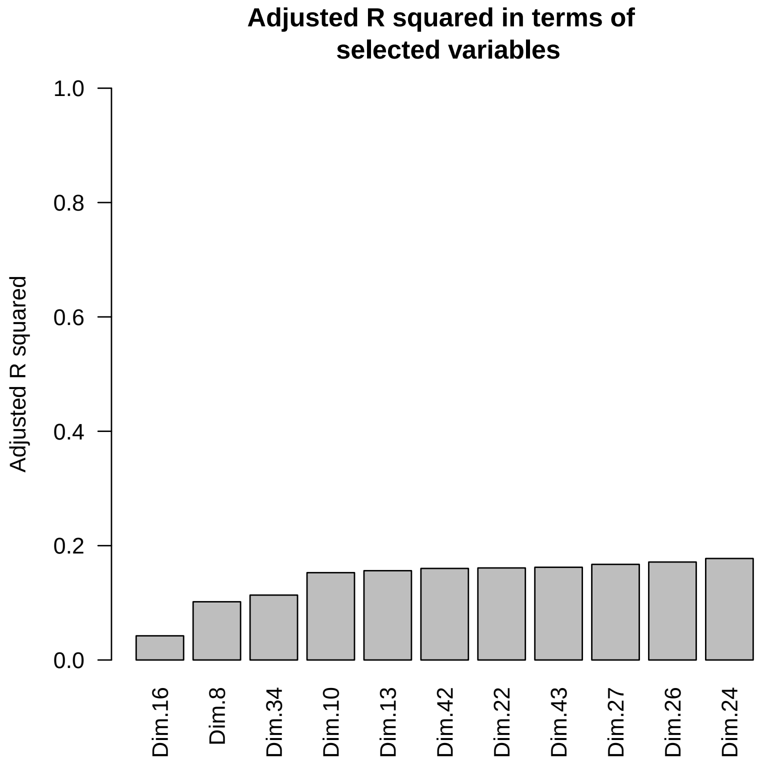


Figure 5.65: Adjusted R squared in terms of the selected variables

5.7.3 Cross-validation with random split

Table 5.25 represents the minimum, mean and maximum of the sensitivities, specificities and adjusted R squares from the cross-validation with 20 executions.

Table 5.25: Range and mean over the sensitivities, specificities and adjusted R squared

	Min	**Mean**	**Max**
Sensitivity	0.14	0.25	0.4
Specificity	0.985	0.991	0.996
Adjusted R^2	0.09	0.16	0.21

The sensitivity presents its values from 0.14 and 0.4, which remains more or less stable. The specificity is extremely high and stable with a mean of 0.991. The adjusted R squares go from 0.09 to 0.21 and remain also stable.

Figure 5.66 represents the twenty ROC curves in the cross-validation of the LDA model. For a small false positive rate and the sensitivity up to 0.3, all curves are very stable. But after 0.4 for the sensitivity and 0.1 for 1 - specificity, the curves are more dispersed. There, the ROC curves are a slightly more unstable.

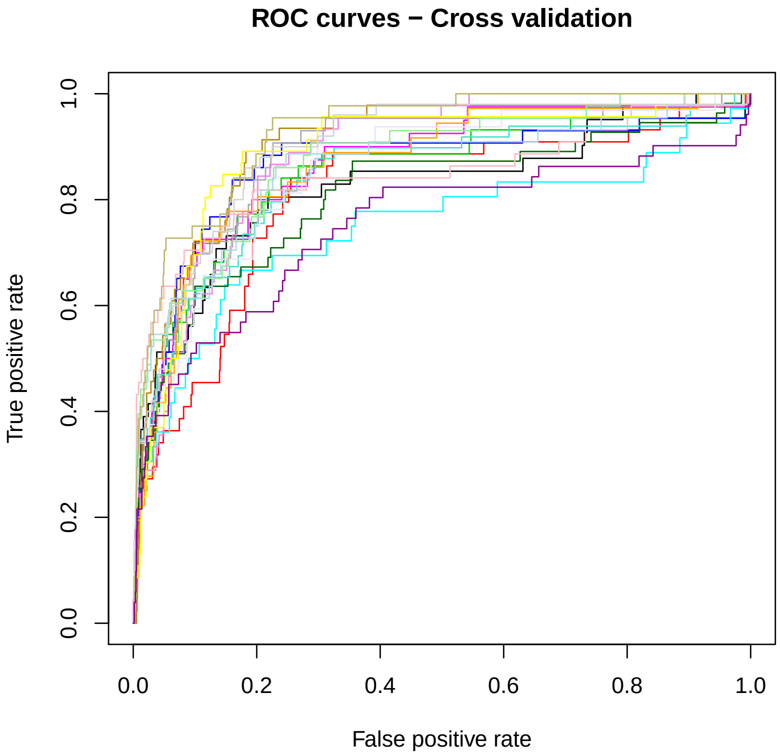


Figure 5.66: ROC curves in the LIMO prediction

5.7.4 Prediction with split of the time series

The number of limonene signals decreases extremely from November. Therefore, it makes no sense to execute a prediction model based on training and test set obtained by splitting the time series in 80% and 20%. The validation set would contain only 10 out of around 1200 signals.

5.7.5 Discussion of residuals and false predictions

In terms of the response variable

The following Figure 5.67 represents the residuals in terms of the logarithmic limonene values. The red coloured points correspond to false negative predictions, the green ones to false positive predictions provided by the LDA model. We see that all false negative observations (red) have a visible positive residual term. Thus, the prediction is smaller than the observation and the real signal is predicted as noise. This is also enforced by the strong unbalance in the number of signal and noise observations. Concerning the false positive predictions, most of the residuals are negative. The predicted values are higher than the real measurements and give false signal predictions.

In terms of the control parameters

Figure 5.68 represents the residual terms of the MLR model in terms of the control parameters. We observe a dependence between control parameter and residuals for the relative humidity and the wind velocity like for the CH_4 analyser. When the relative humidity increases, the residual range becomes more important. The contrary effect is the case for the wind velocity. So high humidity and little velocity measurements lead to high residuals, and therefore to bad predictions. When the humidity has small values, there are nearly no false predictions as well as for high velocity values. Concerning the other control parameters, the false negatives and positives appear under the same conditions.

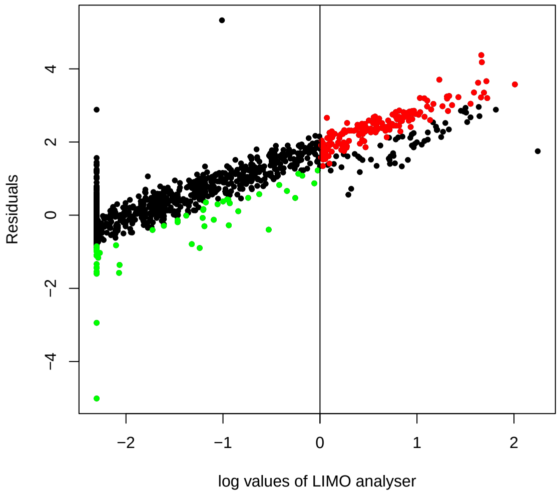


Figure 5.67: Residuals of the MLR model
in terms of log(A_LIMO)

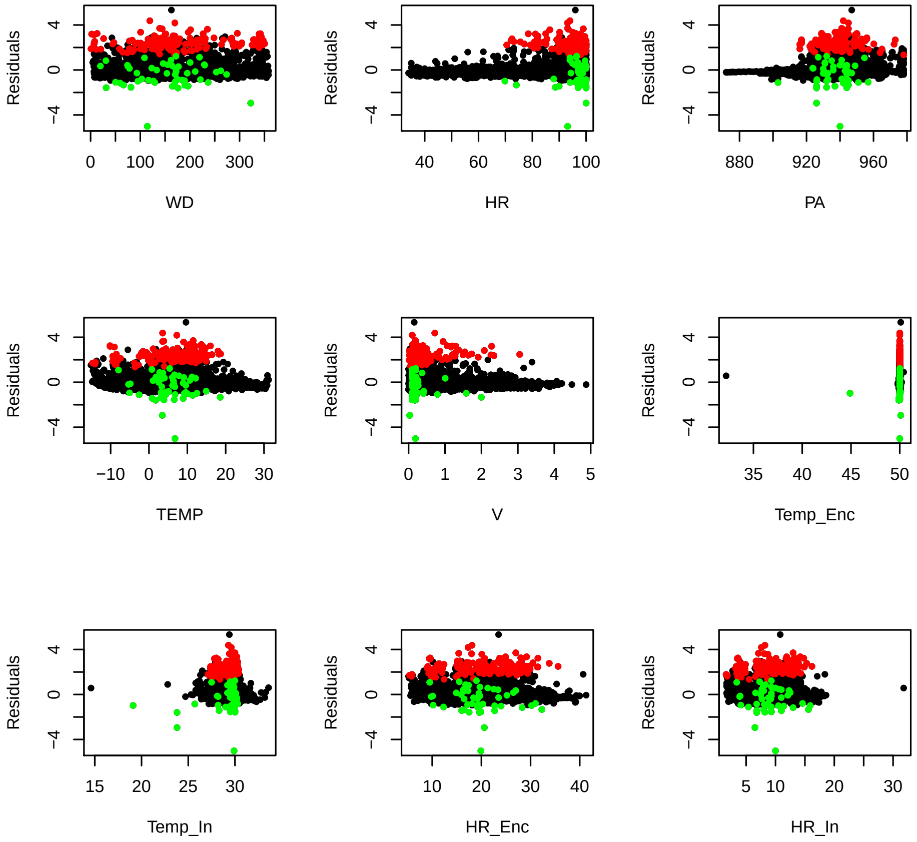


Figure 5.68: Residuals of the MLR model in terms of the control parameters

5.7.6 Which interactions of sensors contribute to the prediction?

The contributions of every interaction of sensors in the selected principal components in the LDA model are represented in Figure 5.69. The size and the lightness of the circles are proportional to the contribution percentages. The first selected principal component is Dim.16 showing up the most important contribution of 26% for the sensor TGS2444. The sensor TGS2620 contributes with 15% in this principal component. Several interactions containing the TGS2611 sensor present also higher contributions. The greatest contribution with a value of more than 30% is present in the second principal component Dim.8 and corresponds to the sensor TGS2602. Afterwards, the interactions containing the TGS2602 sensor seem to be more present in the contributions. For example, the two last selected principal components Dim.24 and Dim.26 show up two interactions of third order: TGS2602 · GGS1330 · TGS2444 with 17% and TGS2602 · TGS2610 · TGS2611 with 18%. The appearance of important contributions for interactions in higher orders confirms our hypothesis of information present in a combination of sensor signals.

Limonene is always present in rural areas. It is a typical compound of plant emissions but it is also characteristic of the odours of waste. However, depending on the direction of the wind it may be accompanied by either waste emissions or the green waste composting center [2]. Emissions of waste and compost may contain NH_3, ammoniac and other VOC. Thus, the high contributions of the sensors TGS2602 and TGS2444 arise from the accompanied compounds.

[2] According to the ISSeP, limonene could come from the composting depending on the wind direction [9].

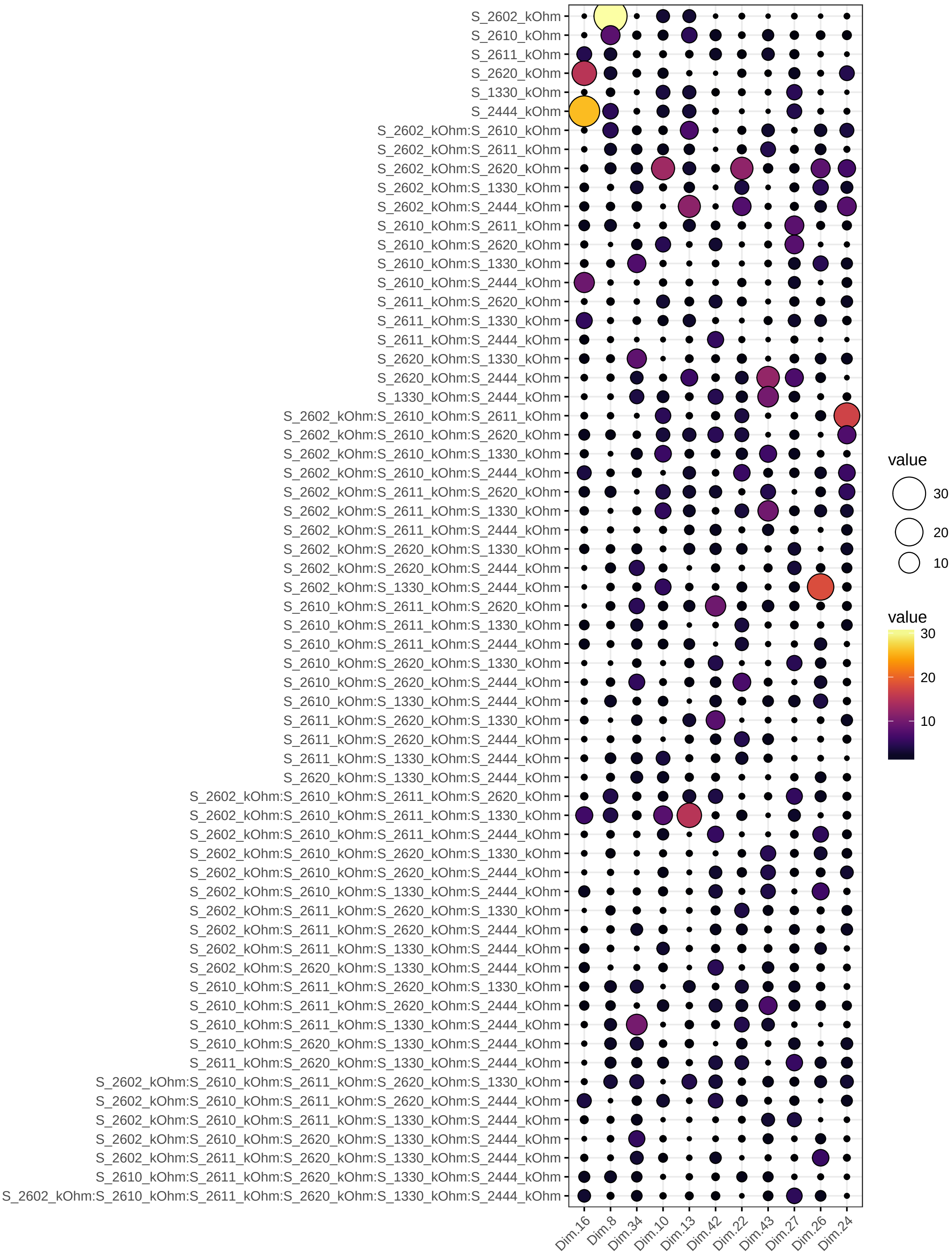


Figure 5.69: Contributions of interactions of sensors in LDA prediction [%]

Chapter 6

Conclusion and perspectives

The first important step of any data analysis is the collection and preprocessing of the available measurements. Several manipulations have to be done before starting the detailed data analysis. Therefore, a package of functions has been written to perform the preprocessing in a semi-automatic process in the software R. These functions are designed to be general enough so as to be applied to a dataset coming from similar studies. Furthermore, an user interface has been created to apply these preprocessing functions in a simple way, even for users that are newcomers in R. Once, the pretreatment process finished, the data analysis starts. In particular, a prediction model for every analyser has been developed based on the interactions of the sensors. The construction of these models has also been implemented by R functions and hence, can be executed rapidly to obtain the statistical results. We have been able to deduce several conclusions by examining the returned interesting findings.

In summary, in terms of signal detection for all analysers and therefore for all chemical compounds, the numeric results are a little deflating. However, on closer inspection we assert that the pollution events emerge being not so poor in prediction quality. Whereas the sensors are sensitive to many environmental factors, the analysers are specific to their chemical compounds. The decrease in concentration for three out of the six analysers (H_2S, NH_3, LIMO) in the second half of measuring phase brings some difficulties about the prediction by the sensors. Additionally, the small ranges of the chemical concentrations complicate the prediction by the sensors. A possible improvement in a new study could be the choice of a measuring location ensuring a more important variation of the chemical compounds. Concerning the contributions of the sensors interactions in the predictions, many of the interactions in higher orders contribute to the prediction which confirms our hypothesis that there is information in the combinations of sensor signals.

Concerning the CH_4 analyser, the pollution events are detected most of time. The sensor expected to detect methane, namely TGS2611, contributes to the prediction of CH_4, but most of time its combination with other sensors presents higher contributions in the methane prediction. The TGS2602 sensor shows up important contributions as well.

The detection of hydrogen sulphide presents somewhat more difficulty because of the decrease in concentration at the end of the measurement phase. The TGS2602 sensor dedicated to measure H_2S contributes to the prediction, but the sensor alone seems to be not enough to predict well. However, in interaction with other sensors like TGS2444 and TGS2611, the presence of higher contributions can be observed.

The problem of decrease in concentration is even more present in the prediction of ammoniac. However, the TGS2444 sensor expected to detect NH_3 shows up the most important contribution in the prediction. Concerning the TGS2602 sensor, which is also dedicated to measure ammoniac among others, accounts only in interaction with other sensors in the NH_3 prediction.

Concerning the last three analysers, namely benzene, toluene and limonene, there is no sensor very sensible explicitly to these compounds. The sensors TGS2602 and TGS2444 are common in the highest contributions in the prediction of the three pollutants. This can be explained by the accompanied compounds of benzene, toluene and limonene.

In subsequent studies, it is recommended to keep the sensors TGS2602 and TGS2444 which proved their contributions in the prediction of methane, hydrogen sulphide and ammoniac. The sensor TGS2611 occurred often in the contributions but mainly in interaction with other sensors. Its share in the prediction of methane could be observed but could probably be improved by combining it with another sensor or to take the TGS2602 sensor for the methane prediction. Concerning the sensors TGS2610, TGS2620 and GGS1330, their contribution in prediction of air pollution could be verified in further studies by concentrating on chemical compounds like those specified in the announced selectivities of these three sensors. The three sensors shared often in the predictions in the interactions in higher orders.

Several perspectives arise from this study either to resolve difficulties encountered in the created model or to improve the prediction quality.

First of all, we can either restrict ourselves to a model that predicts the presence or absence of a signal or a model that predicts the concentration of chemical compounds, depending on the interests in a study.

Because of the non-normality present in the data, the hypotheses of the linear multiple regression and the linear discriminant analysis could not be verified. The passage to a quantile regression model permits to handle the non-normality by estimating the conditional median (or other quantiles) instead of the conditional mean by the least mean squares method [2],[10].

In terms of the time dependence, two perspectives are possible fur subsequent studies. First, the prediction quality could be improved by taking into account the past. Therefore, the predictors are defined as the multivariate observations in the past [1]. Secondly, we could reduce the time lag between the measurements to improve the prediction model.

The prediction model could also be improved by choosing another method for the interpolation of the analysers and for the baseline algorithm [13].

In this study, we performed a principal component analysis on the sensors independently of the concerned response variable. A partial least squares discriminant analysis (PLS-DA) enables a principal component analysis in terms of the response variable which maximises the separability [5].

Finally, the first approach in the model research being to create a linear model can be followed by the establishment of a more complex model. There are many possibilities as for example the addition of quadratic or non linear terms or non parametric modelling. Furthermore, several approaches in machine learning could be good candidates to improve the prediction, for example a neural network [7]. Nevertheless, the inconvenient of these processes is that they are very complex and can not always return information based on the original data. In matters of prediction however, there are many possibilities for improvement.

Appendix A

List of abbreviations

EEA	European Environment Agency
WHO	World Health Organisation
SAM	Sensing of Atmosphere and Monitoring Laboratory
ISSeP	Official Wallonia public scientific institute
VOC	Volatile organic compound
PID	Photo ionization detector
CO_2	Carbon dioxide
H_2S	Hydrogen sulphide
NH_3	Ammonia
ppm	Parts per million
O_3	Ozone
CH_4	methane
BENZ	benzene
TOLU	toluene
MPXY	xylene
ETBZ	ethylbenzene
PINE	pinene
LIMO	limonene
NO	nitric oxide
Temp. Enc.	temperature in the enclosure
Temp. In.	temperature around the enclosure

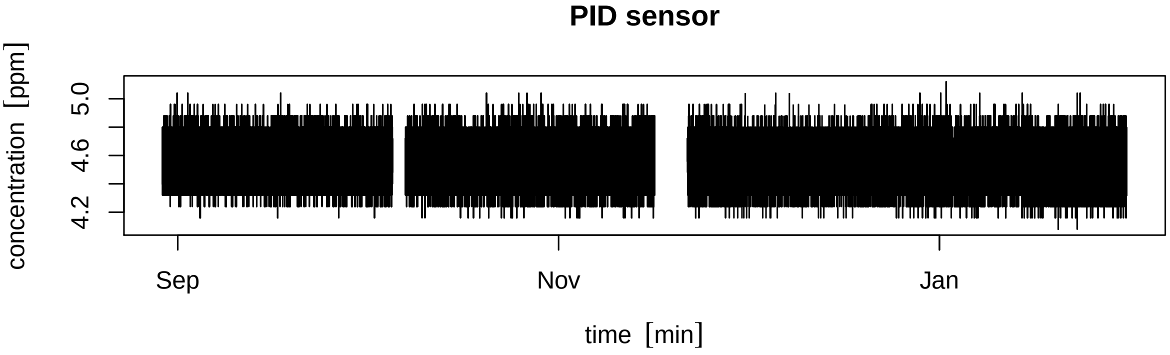
Hr. Enc.	relative humidity in the enclosure
Hr. In.	relative humidity around the enclosure
DV	wind direction
HR	relative humidity
TT	temperature
VV	wind velocity
UTC	Coordinated Universal Time
MET	Middle European Time
CET	Central European Time
IRLS	Iterative Restricted Least Squares
MLR	Multiple linear regression
LDA	Linear discriminant analysis
PC	Principal components
ROC	Receiver operating characteristic
PLS-DA	Partial least squares discriminant analysis

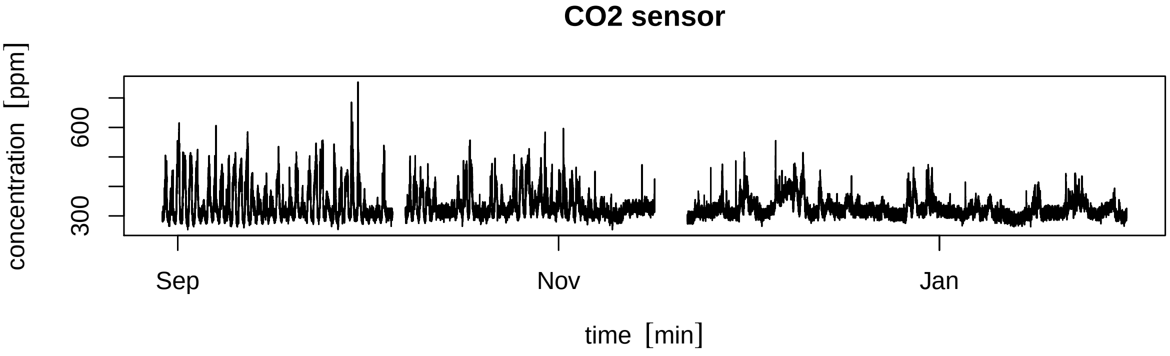
Appendix B

Descriptive analysis of the unused data

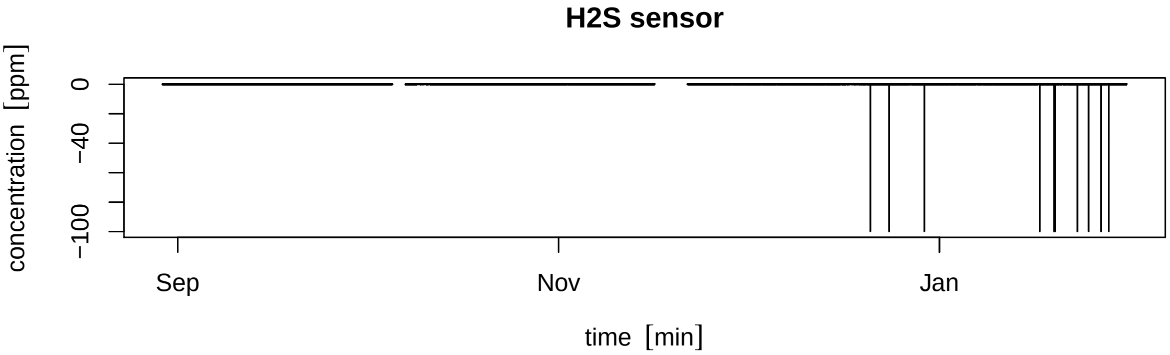
B.1 Specific sensors

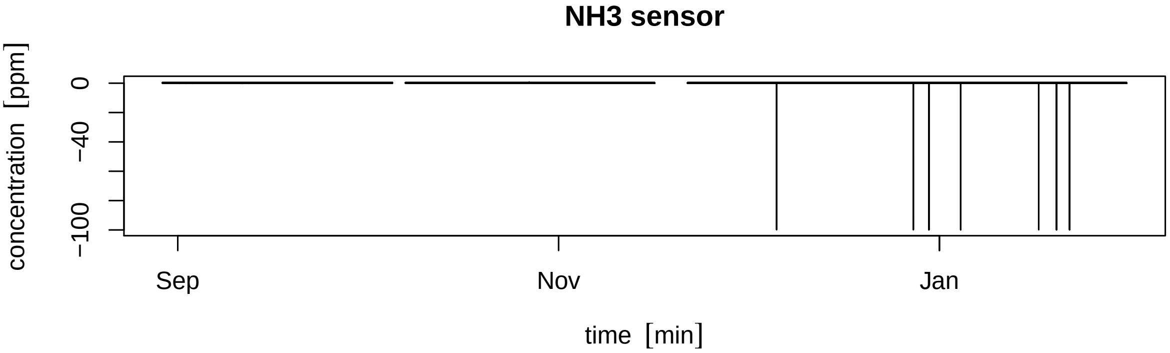
The measurements of the four specific sensors PID [1], CO_2, H_2S and NH_3 are illustrated below:





[1] PID stands for photo ionization detector, which is used to detect VOC.





The concentrations of PID, H_2S and NH_3 were too low during the measuring, as we can see in the next Table B.1. Furthermore, the sensors report negative values for ammoniac and hydrogen sulphide, which is absurd for concentrations.

Table B.1: Statistics for the specific sensors

Sensors	Minimum	Moyenne	Maximum
PID	4.08	4.54	5.12
CO2	253.8	331.3	753.8
H2S	-99.9	0.06	0.35
NH3	-99.9	0.1884	0.72

Concerning the carbon dioxide measurements are concerned, it is expected that they remain stable around 300-400 ppm. Hence, this sensor will provide little useful information too. As additional information, we can have a look on the histograms on these four sensors on Figure B.1.

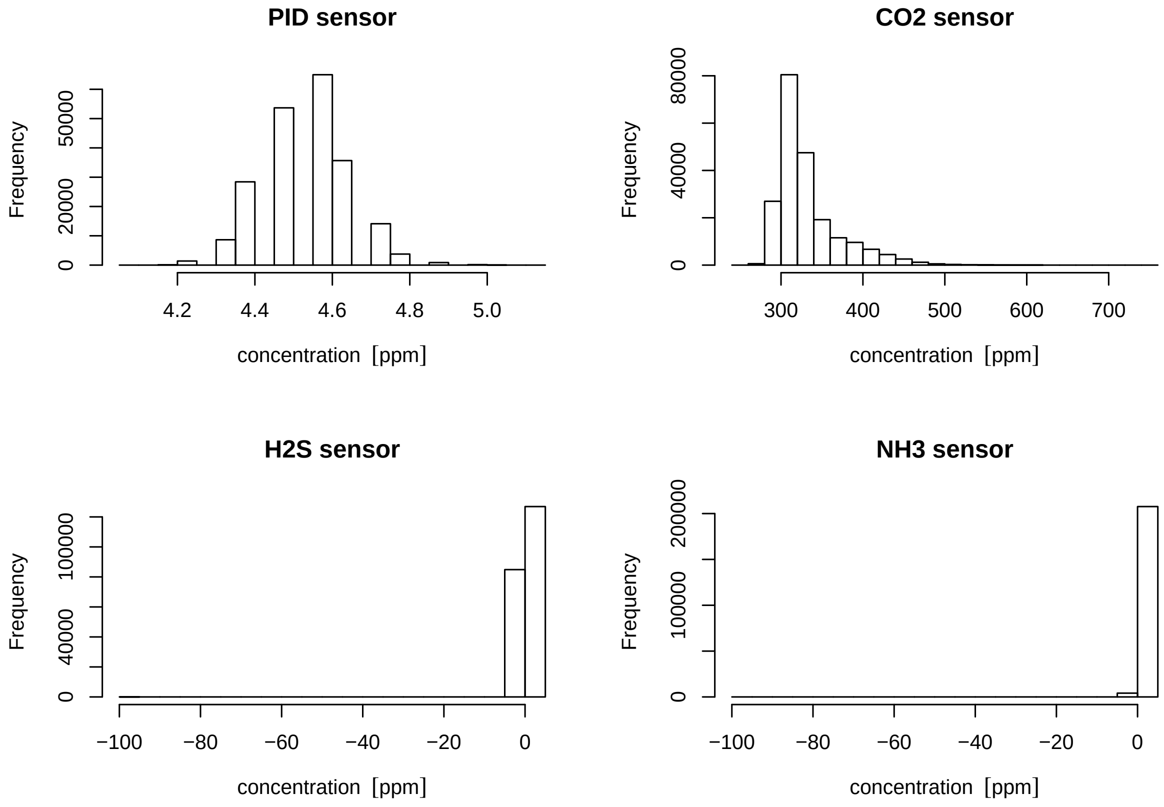


Figure B.1: Histograms of the sensors PID, CO_2, H_2S and NH_3

B.2 Analysers MPXY, ETBZ and PINE

First, we will observe the measurements of the three analysers inFigure B.2.

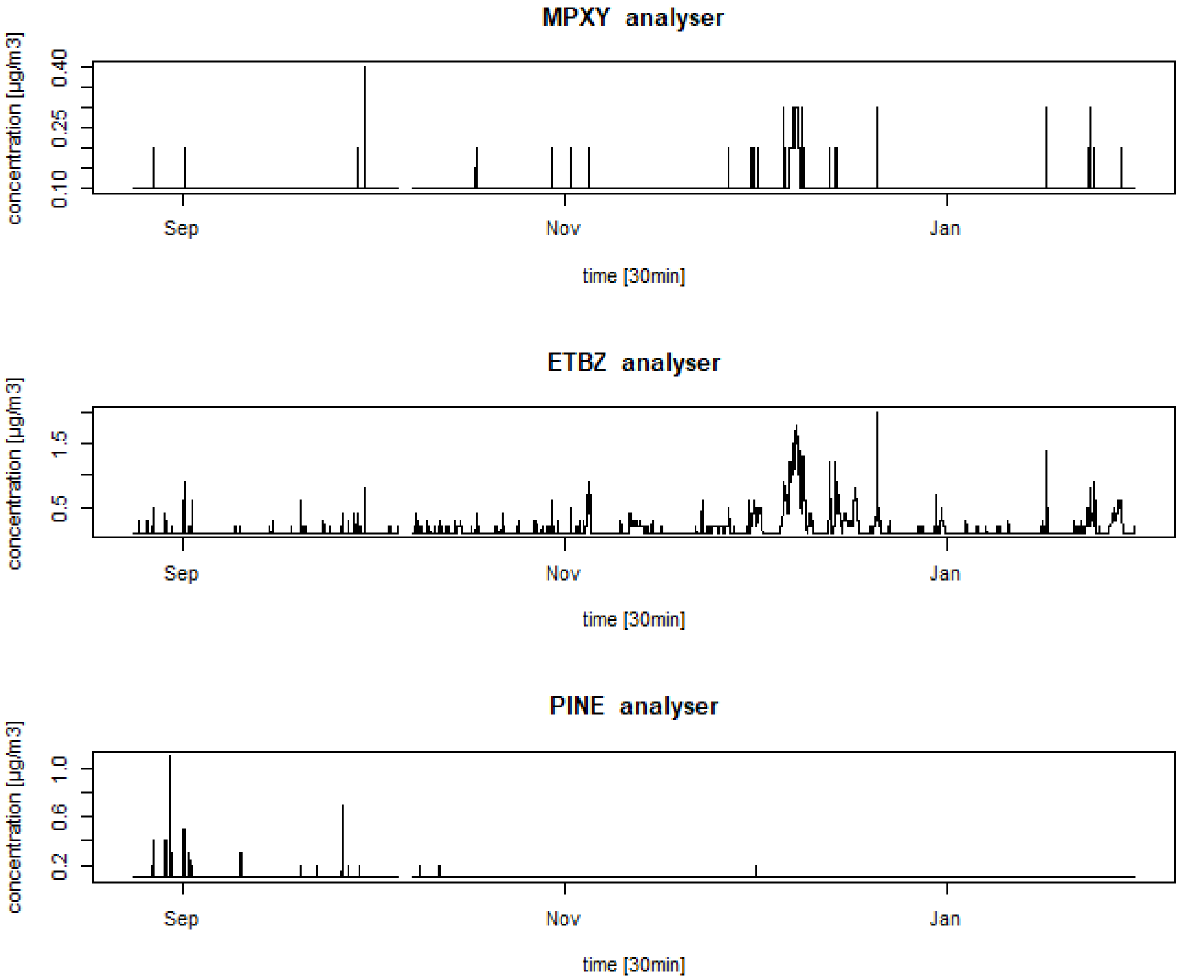


Figure B.2: Measurements of MPXY,ETBZ and PINE analysers

We can see that the measuring range of these three analysers is very small. The table containing minimum, mean and maximum values of these measurements confirms the weak concentrations of MPXY, ETBZ and PINE (see Table B.2).

Table B.2: Statistics for the unused analysers

Analyser	Minimum	Moyenne	Maximum
MPXY	0.1	0.1025	0.4
ETBZ	0.1	0.1598	2.0
PINE	0.1	0.1013	1.1

Furthermore, we can have a look on the histograms on Figure B.3.

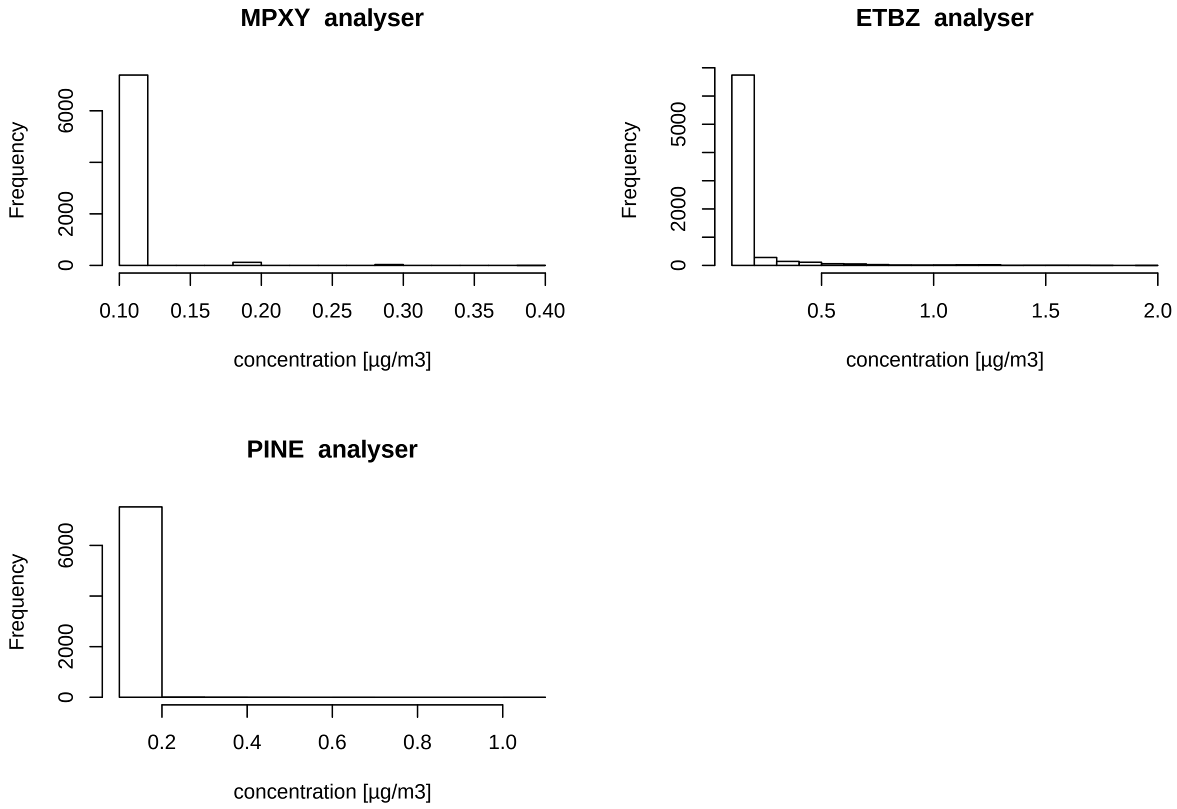


Figure B.3: Histograms of MPXY, ETBZ and PINE analysers

Bibliography

[1] Peter J. Brockwell and Richard A. Davis, *Time series: Theory and methods, 2nd edition*, Springer, 1991.

[2] Brian S. Cade and Barry R. Noon, *A gentle introduction to quantile regression for ecologists*, Frontiers in Ecology and the Environment **1** (2003), 412–420.

[3] Winston Chang, Joe Cheng, JJ Allaire, Yihui Xie, and Jonathan McPherson, *shiny: Web application framework for r*, 2017, available via the URL `<https://CRAN.R-project.org/package=shiny>`, r package version 1.0.5.

[4] European environment agency, *Air pollution*, `https://www.eea.europa.eu/themes/air/intro`, 2017.

[5] Paul H. Garthwaite, *An interpretation of partial least squares*, Journal of the American Statistical Association **89** (1994), No. 425, 122–127.

[6] Garrett Grolemund and Hadley Wickham, *Dates and times made easy with lubridate*, Journal of Statistical Software **40** (2011), No. 3, 1–25, available via the URL `<http://www.jstatsoft.org/v40/i03/>`.

[7] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The elements of statistical learning: Data mining, inference, and prediction, second edition (springer series in statistics)*, Springer-Verlag, 2016.

[8] World health organization, *Air pollution*, `http://www.who.int/airpollution/en/`, 2018.

[9] ISSeP, *Réseau de contrôle des c.e.t. en wallonie*, `http://environnement.wallonie.be/data/dechets/cet/10hab/pdf/10_RapCMP_HAB_2017_air_(4165-2017).pdf`, 2017.

[10] Roger Koenker, *Quantile regression*, Econometric Society Monograph Series, Cambridge University Press (2005), 1–11.

[11] Michael H. Kutner, John Neter, Christopher J. Nachtsheim, and William Li, *Applied linear statistical models*, McGraw-Hill Education, 2004.

[12] Ludovic Lebart, *Statistique exploratoire multidimensionnelle*, Dunod, 1995.

[13] Kristian Hovde Liland, *Baseline correction of spectra*, 2015, available via the URL <`https://cran.r-project.org/web/packages/baseline/baseline.pdf`>.

[14] Anne-Claude Romain, *Personal communication*, 2017-2018.

[15] Claus Weihs, Uwe Ligges, Karsten Luebke, and Nils Raabe, *klar analyzing german business cycles*, Data analysis and decision support (Berlin) (D. Baier, R. Decker, and L. Schmidt-Thieme, eds), Springer-Verlag, 2005, p. 335–343.