Trace metal pollution in the Scheldt estuary - a statistical approach to estimate the metal partitioning coefficient for a suite of metals

View of the Scheldt and the marshes of the Drowned Land of Saeftinghe [13]

Elwin van der Auweraert

Delft University of Technology

Faculty of Electrical Engineering, Mathematics and Computer Science

Department of Applied Mathematics

September 30, 2018

Abstract

Trace metals appear in estuarine systems in two forms: dissolved and particulate. Describing the partitioning between these two forms is done by a coefficient, K_d, which relies on a number of environmental parameters, such as the salinity of the water and the seasonally dependent biological activity [2]. Although this coefficient is known to follow a log-normal distribution, models describing estuarine metal dynamics usually simply use average values.

In this study an attempt has been made to create a statistical model of K_d in the Scheldt estuary for a number of trace metals, based on data of some environmental parameters. This model should be able to cover the whole spectrum of K_d values. The used parameters are salinity, the suspended particulate matter concentration, the total metal concentration and the year of measurement. A comparison is made between two linear regression based models, principal components regression and partial least squares regression, and two decision tree based models, random forest and gradient boosting machine.

Although cross-validation performance on the training set is promising, with the decision tree based models clearly outperforming the linear regression based ones, predictions on an independent test set are very poor for all metals except for cadmium. Cadmium is an exception, because its estuarine dynamics are mainly governed by a specific process, that is driven by changes in salinity.

The dynamics of all other metals depend on both the biochemical and the physical characteristics of the estuary. In the proposed model the parameter 'salinity' has to account for both of these characteristics simultaneously, which is inherently impossible.

Inclusion of pH and or dissolved oxygen content seems promising to create an adequate model. The first, because it is correlated to salinity, and the second because it is representative for the seasonal variation.

Introduction

Throughout history the Scheldt has been a river characterised by trade, dispute and conquest. From ancient Roman times onwards it has carried the blood of many a soldier. Later, when the Dutch revolted against their Spanish rulers, gunpowder and exotic spices found their way into the frequently sailed waterway. Fortunately, soldiers no longer have to fear their blood being spilled into the river. However, other threats are lurking below its seemingly tranquil surface.

As a result of constant, high anthropogenic pressure, the Scheldt currently transports all kinds of chemical pollutants that are harmful both to those inhabiting the surrounding areas and to the riverine wildlife.

The best-known ones might well be trace metals -more commonly, but falsely, known as heavy metals- [1]. These metals arrive in the river system through different sources, wastewater treatment plants being the biggest contributor. Detailed modelling of the behaviour of these metals is essential to be able to make predictions and thus to be able to assess the imposed risks.

As a large part of the Scheldt is an estuary, this is all but trivial. In estuarine waters metals undergo different kinds of chemical reactions, are in a constant flux between adsorption and desorption to particles, and are subject to both the freshwater river current and the saline tidal flows. An essential part of models describing the behaviour of trace metals is the exchange between the particulate and dissolved phase. This is not only important for the modelling of the dynamics of the metals, but also to be able to asses the involved risks, as only the metals in the dissolved form are bio-available, which means that they might be toxic both to humans and wildlife.

A common way to model the exchange between the two phases is the use of a partitioning coefficient K_d. The K_d describes the partitioning of metals between the two phases, assuming they are in local equilibrium. This coefficient is affected by several environmental factors such as the seasonally dependent biological activity and the pH-value, which is in turn influenced by the salinity of the water [2].

As so many interrelated processes contribute to the K_d, a mechanistic description remains too complicated. Therefore, most metal dynamics modelling approaches choose to represent the K_d of each of the modelled metals by a constant, average value. In this study the possibilities to model the K_d of different metals as a function of certain environmental variables will be investigated.

Such a function will be an empirical one, resulting from a statistical model on environmental data. More specifically a comparison will be made between functions based on more classical, linear regression based methods and more modern, decision tree based ones.

The goal of this study can be summarised in the research question:

Is it possible to model the K_d of different metals as a function of a number of environmental variables, using linear regression or decision tree based methods, and if so, which of these methods performs better?

Personal motivation

I could start explaining my personal motivation by summing up the dry facts why I was attracted to the project of trace metal pollution in the Scheldt estuary.

I could for instance start by saying that I was motivated by the serious issue that metal pollution poses, threatening humans and wildlife alike. Or I could argue that this project was a logical choice given my background in civil engineering, the field of studies in which I hold my bachelor's degree. Or perhaps I could mention that awareness of environmental issues is a remainder from my childhood, being raised by a father who works as a chemical-environmental engineer.

But I will refrain from these dry facts. Enough of them can be found in the rest of this thesis already. And as we can see more clearly now than ever, in these tumultuous times that we are living in, where irrational politics are slowly turning into normality, people are generally not driven by logical arguments, but rather by their inner feelings and emotions. Let me, being as sentimental a being as any of us, therefore give a more emotional, maybe even slightly poetic motivation.

On the title page, a beautiful picture of the Scheldt displays the river as a tranquil, even serene entity. However, underneath the calm water surface highly chaotic, hydraulic forces are at play. The water particles are subject to a constant struggle between saline tidal currents and freshwater ones, driving them from Belgium to the Netherlands and back again.

Perhaps it were these estuarine dynamics that appealed so strongly to me, as a similar internal turbulence seems to dictate my path. Being of Dutch-Belgian origin, I am also in constant disequilibrium between the two conflicting cultures. To write this thesis I moved to Belgium for the first time in my live. Not only to be closer to the source of the problem studied in this thesis, but also to my own.

Acknowledgements

I would like to use this section to thank all those, who have been involved in my thesis. Be it by contributing to the content, by helping me with the administrative work and logistics or by encouraging me at the moments I needed it most.

A number of people deserve to be mentioned separately. First of all there is Arnold Heemink, the chair of the ’Mathematical Physics’ research group at TU Delft and hence the professor responsible for my graduation. Apart from the guidance, that he gave me throughout the research, he fulfilled a special role, as it was in his office that this journey began. Almost a year ago, I came to professor Heemink to discuss possible thesis topics. Upon mentioning the idea of writing the thesis in Belgium, he referred me to Eric Deleersnijder, who is a part time professor in both Delft and Louvain-la-Neuve. From there it all went pretty quickly.

Eric also deserves a special word of thanks, as he was the one that offered me the opportunity to do my research at the University of Louvain-la-Neuve and guided me there as my daily supervisor and in his function as a professor. Next to that I should thank Eric for the paperwork he had to do in order to enable me to be registered at Louvain-la-Neuve as an exchange student.

Next to Eric, I was supervised by Marc Elskens. Both Eric and Marc helped me by giving feedback at monthly meetings, which I greatly appreciate. I should also thank Marc for the introduction to the realm of applied statistics and more specifically to machine learning models.

For the administrative work, required to secure my position as an exchange student at Louvain-la-Neuve, I thank Jitske van der Laan and Emmanuelle Brun, the exchange coordinators from both universities.

Bert Maetens receives an individual word of gratitude for the stunningly beautiful picture of the Scheldt, which can be found on the title page, that he provided me with.

Last I would like to thank my Belgian grandparents for being able to stay at their place both at the beginning and at the end of my stay in Belgium. This way I had a stable and loving haven, during the busiest periods of my sojourn.

Finally I thank all those, who supported me throughout the research, who kindly listened to the tales of my struggles and who encouraged me at the moments when my motivation was at a low. I could not have done it without all of you.

Contents

List of Tables

List of Figures

List of used abbreviations

In this paper a number of abbreviations will be used, to avoid endless repetition of long terms. For the physical quantities their unit is given in square brackets.

Physical quantities

- K_d - Metal partitioning coefficient
- MeA[1] [µg/mg] - Metal content associated with suspended particles
- MeD[1] [µg/L] - Dissolved metal concentration
- MeP[1] [µg/L] - Suspended metal concentration
- MeT[1] [µg/L] - Total metal concentration
- SPM [mg /L] - Suspended particulate matter concentration
- SAL [psu] - Salinity
- YEAR [-] - year of measurement

Metals

- Cd - Cadmium
- Pb - Lead
- Co - Cobalt
- Ni - Nickel
- Cu - Copper
- Zn - Zinc
- As - Arsenic

[1] The quantities MeA, MeD, MeP and MeT are generic for all metals. The specific quantities for a single metal will be denoted by replacing Me by the abbreviation of that metal. For instance for Cadmium (Cd) the four quantities read: CdA, CdD, CdP and CdT.

Statistical terms

- PCA - Principal component analysis
- PVE - Proportion of variance explained
- PCR - Principal Components Regression
- PLS - Partial Least Squares regression

1 Biochemical background

Metals arrive in the estuarine system from numerous sources, such as wastewater treatment plants, erosion and metallurgy. The first being the biggest contributor for most trace metals [17]. Once the metals are in the estuarine system, they occur in two phases the particulate and dissolved one. The goal of this research is to be able to accurately predict the partitioning between the metal content in both of these phases.

An alternative way of representing this partitioning is by a partitioning coefficient. Given this coefficient and the total metal content, one can compute the metal content in both phases. In this study a model will be constructed to estimate the metal partitioning coefficient, based on a number of environmental parameters, the total metal content and the year of measurement.

Before going into more detail on the modelling approach, the underlying biochemical background will be presented in order to gain better understanding of the processes in question.

1.1 Biochemical processes in the estuarine system

As previously mentioned metals in the riverine system occur in two phases: the particulate and the dissolved one. The first group consists of metals adsorbed to suspended particles in the water column and those on the river bed. The latter contains dissolved metals in the water column and in the pore water of the river bed. Interaction between the two groups is possible through absorption, adsorption and desorption processes. The dissolved metals can diffuse from the water column to the pore water in the river bed and reversely. The particulate metals move from the water column to the river bed by sedimentation. The inverse process is called resuspension. Figure 1 gives a schematic overview of all these biochemical processes.

The sorption processes can be further classified into chemical reactions, such as redox and complexation reactions. However, as this study will be performed from a mathematical point of view, we will initially stick to a crude representation of the biochemical processes. However, at points in this research, where biochemical details become essential to understand the functioning or dysfunctioning of the model, they will be addressed.

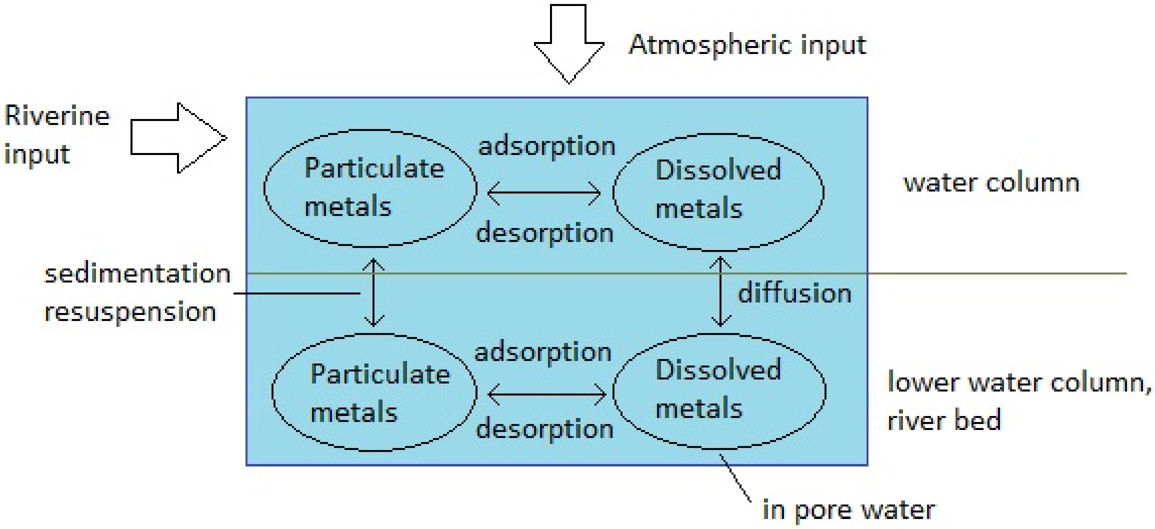


Figure 1: Schematic overview of the biochemical processes that metals are subject to in an open estuarine system. The lower part of the image represents the river bed and the lower layer of the water column, in which interaction with the rest of the water column takes place. The rest of the water column is represented by the upper part of the image.

1.2 Characteristics of the Scheldt estuary

The Scheldt estuary is one of the youngest estuaries in western Europe, hosting a unique flora and fauna [3]. It is under high anthropogenic pressure, because its surrounding area is densely populated. Moreover, those parts that are not used as residential areas are used for agricultural or industrial purposes. Many of the industrial activities are linked to the port of Antwerp, the second biggest port in Europe [3].

The Scheldt estuary is divided into three parts: the Western Scheldt [Westerschelde], the lower Scheldt [Beneden-Schelde] and the upper Scheldt [Boven-Schelde]. The entire Scheldt river is bigger than the Scheldt estuary, its source being in France. The estuarine part is defined as the part of the river that is affected by tidal flows. Figure 2 displays a map of its watershed.

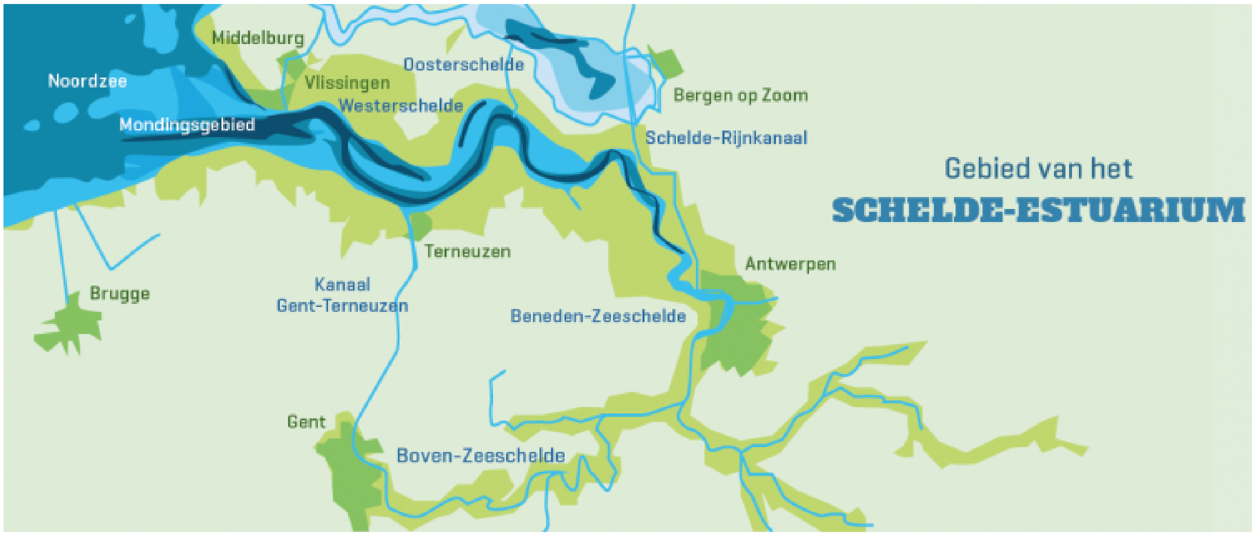


Figure 2: Watershed of the Scheldt estuary [3].

The estuary is well-mixed, except during peak discharges [15]. An estuary is well-mixed, when there is strong mixing of tidal water and one can find salt water even at the top of the water column. Furthermore, many of the biochemical processes that take place in the river are influenced by the season. Both biochemical activity and the salinity of the water, for instance, depend on the water temperature [18].

1.3 The metal partitioning coefficient

1.3.1 Underlying assumptions

The metal partitioning coefficient, $\mathrm{K_d}$, is a measure to describe the partitioning between metals in particulate and dissolved phase. This constant is only meaningful under the assumption that local equilibrium between these two phases is attained. In turn this assumption implies that the rate of the sorption reactions (adsorption, absorption and desorption) is high relative to the advective-diffusive transport of the metals [4]. This is important to realise, as in practice it means that at moments of high flow velocity the aforementioned assumption is violated, possibly resulting in worse K_d predictions.

1.3.2 Formal definition

The metal partitioning coefficient is formally defined as

$$K_d = \frac{MeA}{MeD}, \tag{1}$$

where MeA is the metal content associated with suspended particles [µg/ g] and MeD the dissolved metal concentration [µg/ L]. The commonly used unit of K_d is [L/kg]. MeA is the fraction of the total suspended particle weight that is contributed by metals. MeD is simply the dissolved metal weight per volume of water. An equivalent measure exist for the particulate metals and is defined by

$$MeP = MeA \cdot SPM, \tag{2}$$

where MeP is the particulate metal concentration [µg/L] and SPM is the concentration of suspended particles [mg/L]. The abbreviation SPM stands for Suspended Particulate Matter concentration. Another measure that will be frequently used in this thesis is the total metal concentration, MeT [µg/L] defined as

$$MeT = MeD + MeP. \tag{3}$$

2 Modelling approach

The model for K_d that will be developed in this research will be based on environmental parameters that can be measured easily and at relatively low cost. Two of such parameters are the salinity of the water, SAL, and the suspended particulate matter concentration, SPM.

Another important parameter to include in the modelling of the K_d is the total metal content, MeT, in the water. As we are ultimately trying to model the dissolved and particulate metal concentrations, the K_d is only practically useful in combination with the total metal concentration. In certain scenarios the MeT is known from measurements, but in others this might not be the case. For our modelling domain it is fortunately possible to predict the MeT quite accurately based on SAL and SPM, as was also shown in previous research by Elskens [2].

Furthermore, it is necessary to include the year of measurement into the model, as over the years the quantity of certain metals in the Scheldt estuary might have changed, for instance because of a change of European norms reducing the industrial output of a certain metal. There is also a big difference in the accuracy of the measurements, as major improvements in sampling procedures and analytical techniques were made at the end of the eighties [15].

Thus, two models have to be built. A first one to estimate the total metal content based on the variables SAL, SPM and $YEAR$

$$MeT = f(SAL, SPM, YEAR), \tag{4}$$

and a second one to estimate the metal partitioning coefficient for a certain metal based on the variables SAL, SPM, MeT and $YEAR$

$$Kd = f(SAL, SPM, MeT, YEAR). \tag{5}$$

3 Overview of the used data bases

We are working with two data bases: one collected by the VUB (Vrije Universiteit Brussel) and another one from Rijkswaterstaat. The Rijkswaterstaat data base will be referred to as the MWTL data base. The acronym refers to Monitoring Waterbouwkundige Toestand des Lands.

3.1 VUB data base

The VUB data base contains measurements from the years 2011 up to 2018, with the exception of the years 2013 and 2017. For the year 2018 there is a slight difference in the metals that have been measured and thus it will be addressed separately.

All of the measurements recorded in the VUB data base were performed between February and the beginning of March of the respective years.

3.1.1 Years 2011 to 2016

This data base contains measurements from the years 2011 up to 2016, not including the year 2013. To get a better idea of what the data base looks like, a typical cut-out is presented below.

$SAL\,[psu]$	$SPM\,[mg/L]$	$YEAR$	$Cd_D\,[\mu g/L]$	$Pb_D\,[\mu g/L]$	$Cd_A\,[\mu g/g]$	$Pb_A\,[\mu g/g]$	station
27	35.1	2011	0.11	0.12	0.3	14	S01
7	61.4	2016	0.12	0.16	2.5	83.1	S07

Table 1: Typical cut-out of the VUB data base for the years 2011-2016

In the above cut-out of the data base the following variables are included:

- $SAL\,[psu]$ = salinity
- $SPM\,[mg/L]$ = suspended particulate matter concentration
- $YEAR\,[-]$ = year of measurement
- $Cd_D\,[\mu g/L]$ = concentration of cadmium in the dissolved phase
- $Pb_D\,[\mu g/L]$ = concentration of lead in the dissolved phase
- $Cd_A\,[\mu g/g]$ = cadmium content associated with suspended particles
- $Pb_A\,[\mu g/g]$ = lead content associated with suspended particles

- station = the measurement station[2]

The actual data base consists of 72 data points. Also, there are 14 more variables, which are the concentrations of the other metals in the dissolved phase and their content associated with suspended particles respectively. The following metals are included in the data base.

- Cd (Cadmium)
- Pb (Lead)
- Co (Cobalt)
- Ni (Nickel)
- Cu (Copper)
- Zn (Zinc)
- As (Arsenic)

Recall that we want to build a model for K_d as a function of the variables SAL, SPM, $YEAR$ and MeT. The variables SAL, SPM and $YEAR$ are directly available from the VUB data base. However, the K_d and MeT values have to be computed from the MeD and MeA values. First, the MeT values are computed using relationships 2 and 3. Then, K_d values are be calculated through equation 1. A typical cut-out of the resulting data base, that will be used to construct the K_d model, is shown in table 2 below.

For every data point of the initial VUB data base, 7 data points exist, one for each metal, exist in this modified model data base. Thus, the 2011-2016 part of the VUB data base used for the modelling of the K_d contains 504 data points. In the cut-out a column has been added to specify the metal type.

$K_d\,[L/kg]$	$SAL\,[psu]$	$SPM\,[mg/L]$	$YEAR$	$MeT\,[\mu g/L]$	metal
$2.7 \cdot 10^3$	27	35.1	2011	0.12	Cd
$5.6 \cdot 10^4$	7	61.4	2016	0.38	Ni

Table 2: Typical cut-out of the VUB data base for the years 2011-2016 used to model the K_d

3.1.2 Year 2018

The data base for the year 2018 is highly similar to that of the years 2011-2016. The only difference is that for the year 2018 there are no measurements for arsenic (As).

This data base contains 15 data points per metal.

[2] Information on the locations of the VUB measurement stations can be found in paragraph 3.1.3.

3.1.3 Measurement stations

Figure 3 shows the measurement stations, where the measurements were performed that are recorded in the VUB data base. These are the stations S01 to S22, numbered increasingly in the upstream direction [14]. They are located between Vlissingen and the city centre of Antwerp approximately.

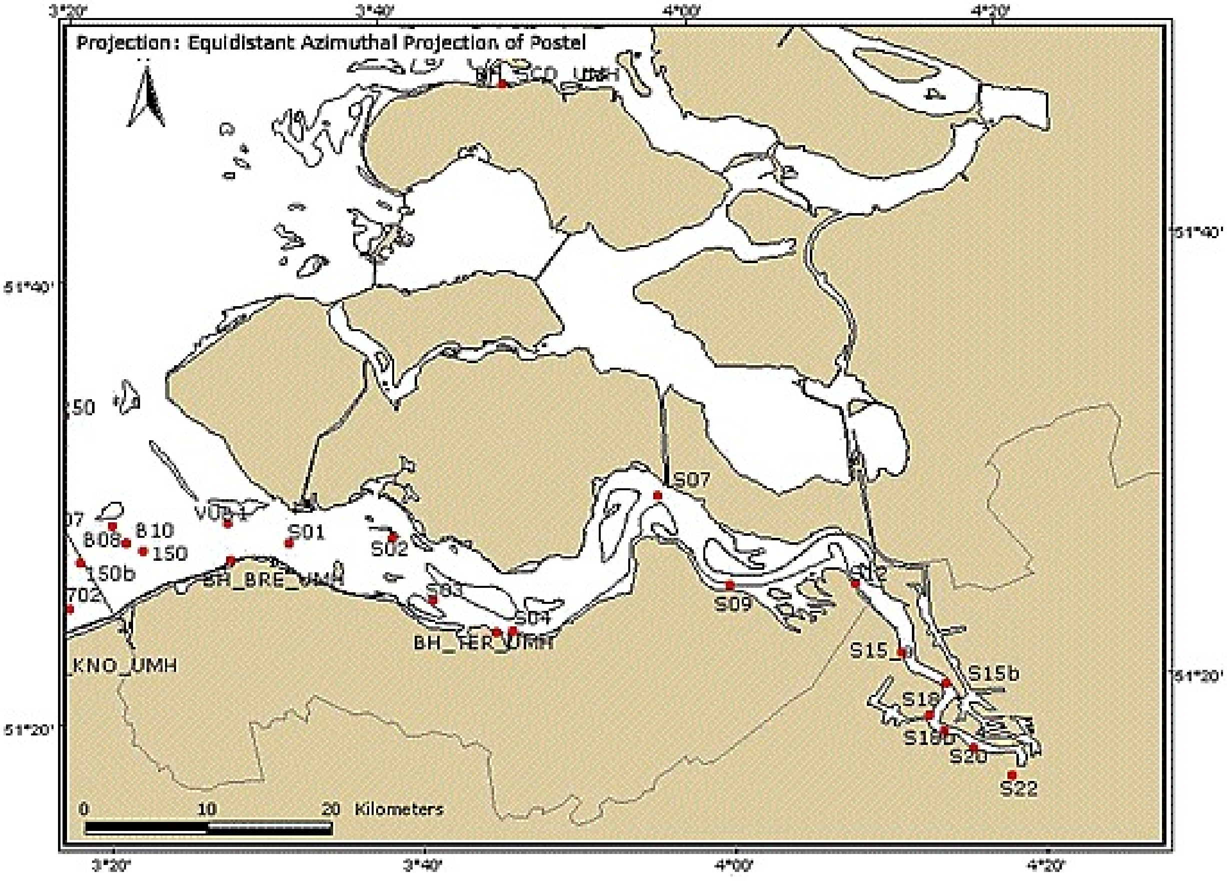


Figure 3: Overview of the VUB measurement stations [14].

3.2 MWTL data base

The MWTL data base is gathered by the Dutch Ministry of Infrastructure and the Environment (Ministerie van Infrastructuur en Milieu) and Rijkswaterstaat [12]. It includes data from the years 1982 to 1995 and 2006 to 2017. In contrast to the VUB data base these measurements are performed over the whole year and not only in the months February and March. These measurements are all performed in the Western Scheldt, i.e. the Dutch part of the Scheldt.

This is a very large data base, containing measurements on some biochemical variables and metal concentrations for a large range of metals. We will only be using data from the metals that are also included in the VUB data base. These are: Cd, Pb, Co, Ni, Cu, Zn and As

From the original data base the following variables are used: SAL, SPM, MeT, MeD, date and time of the measurement and the measurement station. K_d values are computed using equations 1 up to 3 in reversed order.

A typical cut-out of the data base that we will be working with is presented in table 3 below. In total the data base contains 3967 data points. The number of data points per metal can be found in table 4 below.

$K_d\,[L/mg]$	$SAL\,[psu]$	$SPM\,[mg/L]$	$YEAR$	$MeT\,[\mu g/L]$	metal	datetime	station
$1.5 \cdot 10^4$	12.0	120	2016	7.0	Cd	2016-12-29T10:23:00	Schaar van Ouden Doel
$3.2 \cdot 10^4$	11.6	63	1982	46	Zn	1982-02-15T14:35:00	Hansweert Geul

Table 3: Typical cut-out of the MWTL data base used to model the K_d

metal	Cu	Ni	Co	Cd	As	Pb	Zn
n	658	578	307	641	381	701	701

Table 4: Number of data points per metal in the MWTL data base.

3.2.1 Measurement stations

A schematic overview of the locations of the five MWTL measurement stations: Terneuzen, Hoedekenskerke, Hansweert, Bath and Schaar van Ouden Doel, can be found in figure 4.

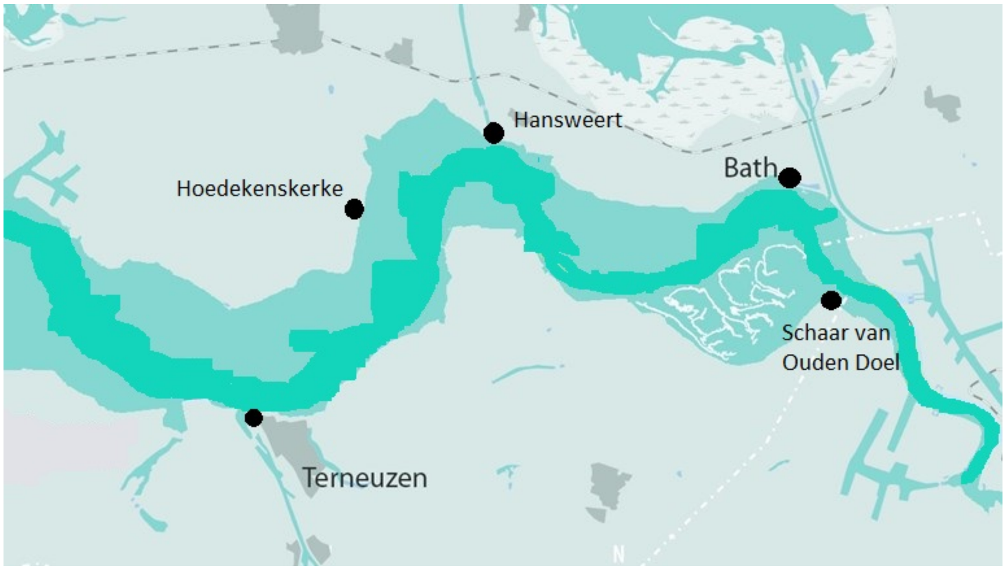


Figure 4: Overview of the MWTL measurement stations

4 Data exploration

Before starting the actual modelling, the data bases will be explored both to get a better idea of the distributions of the variables and to gain insight into the patterns and correlations present in the data.

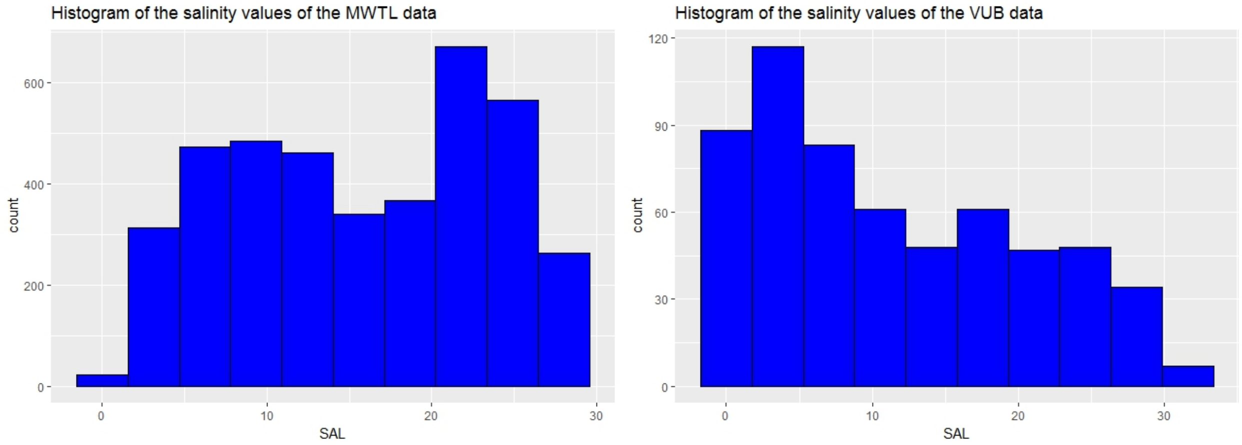
4.1 Distributions of the variables in the data bases

It is important to know the distribution of the variables, that will be used for the modelling. This can influence the performance of the models and hence sometimes transformation of certain variables is desirable. The variables that will be used to build the models are: SPM, SAL, MeT, K_d and $YEAR$. The distribution of $YEAR$ is not that interesting -it is more or less uniformly distributed, as approximately the same amount of measurements is made every year- and will thus be disregarded.

4.1.1 Salinity

The salinity distribution is mainly determined by the geographical location of the measurement stations. Stations further downstream are closer to the sea, where the water is more saline, due to a greater tidal influence. The measurement stations of both data bases are situated in approximately the same region of the Scheldt, as can be seen in figures 3 and 4.

The salinity distributions for the MWTL and the VUB data base are given in figure 5. Although the range of both data bases is comparable: values lie between 0 and 35 *psu*, the MWTL data contains relatively



(a)

(b)

Figure 5: The left panel and right panels show the distributions of the log(SAL) values of the MWTL and VUB data bases respectively.

more high salinity values than the VUB data base. This discrepancy can be explained by the fact that all measurements of the VUB data base were performed in the months February and March exclusively, whereas the MWTL measurements were performed in all months. The explanation is twofold. First, the salinity is positively correlated to the water temperature [18], explaining lower salinity values in during the cold winter months. Moreover, the river discharge is higher in winter than in summer, resulting in a relatively high contribution of fresh water in the estuarine system [15]. This corresponds to a lower salinity. This pattern can clearly be seen in figure 6, which shows, by means of a boxplot, the range of the salinity values for each month of the year for the station 'Schaar van Ouden Doel' of the MWTL data base.

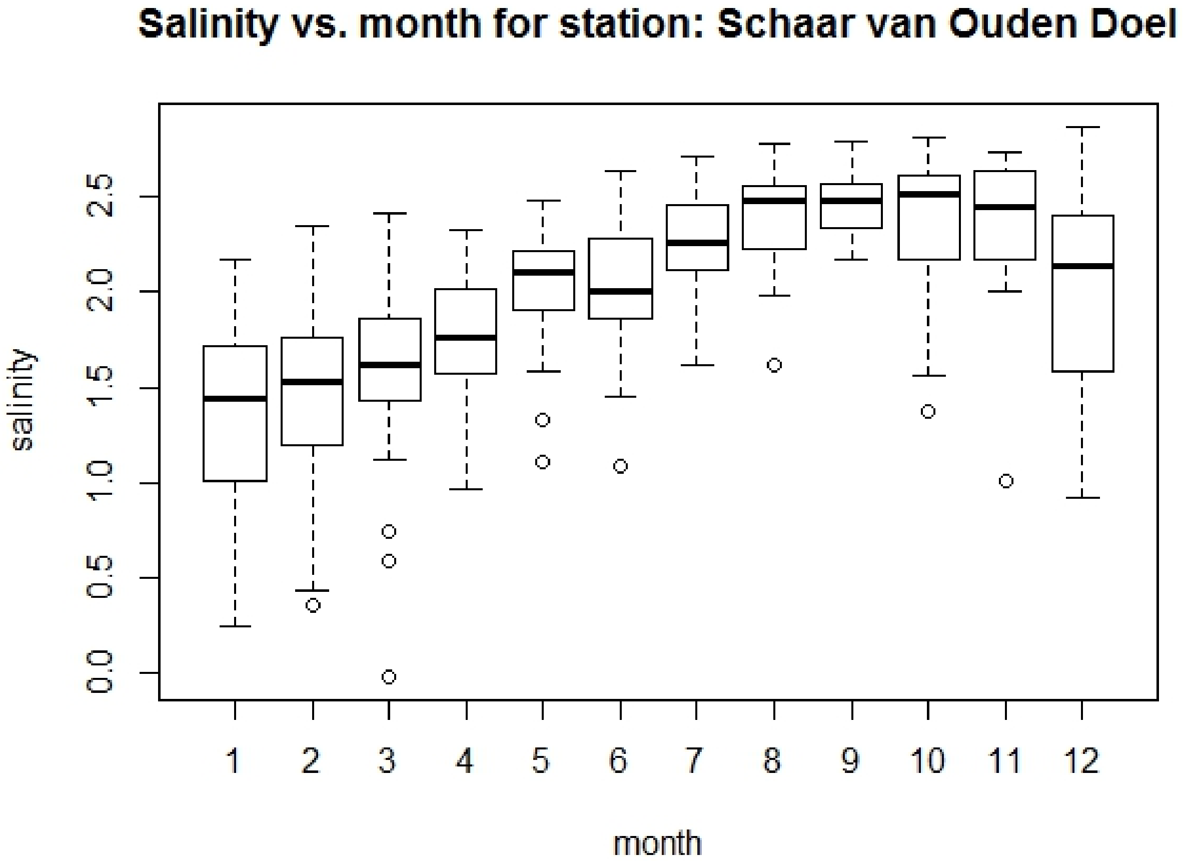
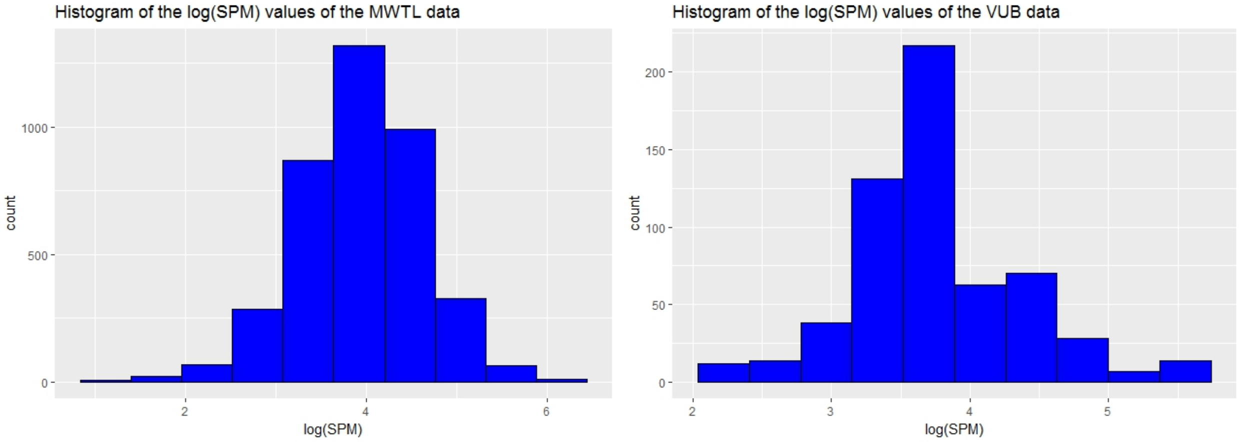


Figure 6: Salinity vs. month plot at measurement station 'Schaar van Ouden Doel' of the MWTL data base. Month 1 corresponds to January, 2 to February, etc.

4.1.2 Suspended particulate matter

All other variables follow more or less a log-normal distribution, i.e. a distribution that is normal after performing a logarithmic transformation. This is the case because environmental data tend to have a distribution with a lot of small values and a small number of very high values [8].

The SPM distributions for both data bases are shown in the two panels of figure 7. They are reasonably similar, nicely following a log-normal distribution.



(a)

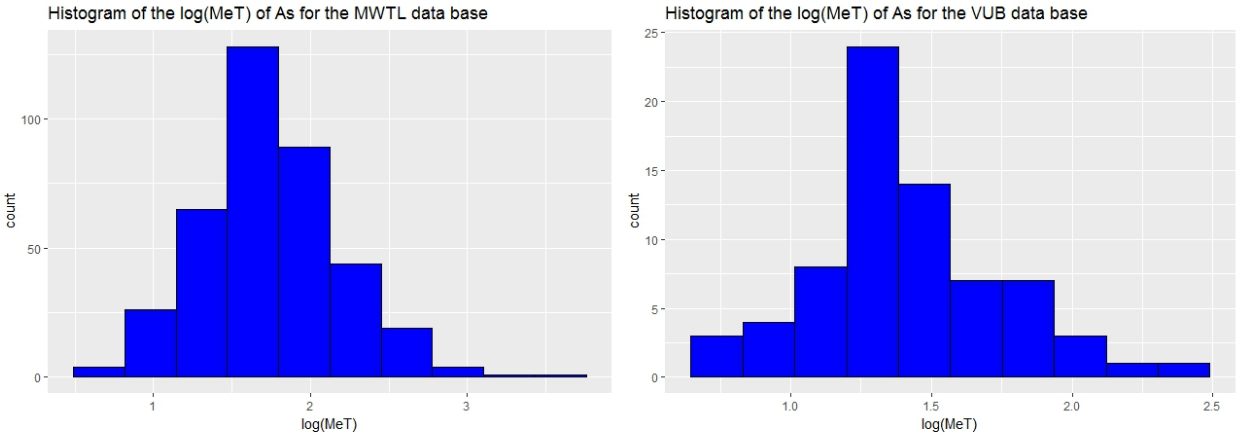
(b)

Figure 7: The left panel and right panels show the distributions of the log(SPM) values of the MWTL and VUB data bases respectively.

4.2 MeT

Little is to be learned by looking at the MeT distribution for the entire data bases. Rather, the distributions per metal type will be presented.

Figure 8 shows the logarithmic MeT distributions of arsenic for the MWTL and the VUB data base. They are both close to a normal distribution. The MeT distributions for the other metals can be found in appendix E.1.



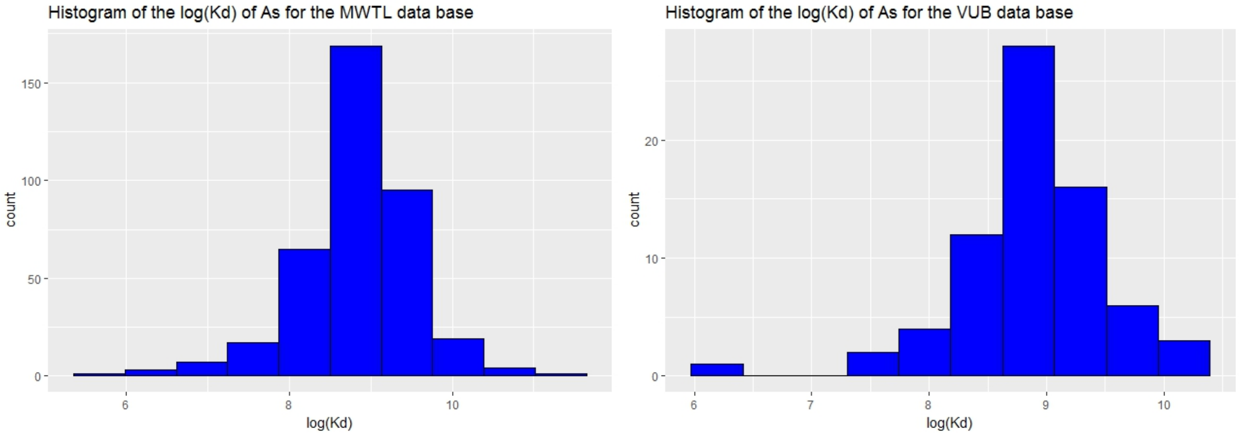
(a) (b)

Figure 8: The left panel and right panels show the distributions of the $\log(MeT)$ values of arsenic in the MWTL and the VUB data bases respectively.

4.3 K_d

4.3.1 Distributions for the VUB and MWTL data base

The logarithmic K_d distributions of arsenic in the MWTL and VUB data base are shown in figure 9 below. They are both more or less normally distributed. The K_d distributions for all other metals of both data bases can be found in appendix E.2.



(a) (b)

Figure 9: The left panel and right panels show the distributions of the $\log(K_d)$ values of arsenic in the MWTL and the VUB data bases respectively.

4.3.2 Comparison of the K_d distributions of the data bases with literature values

The K_d distributions of metals in a specific environment are well-known from literature and do not depend much on time or location [4]. To assure that the data at hand is representative, the K_d distributions of all metals in the VUB and MWTL data base will be compared with literature K_d values of metals in an environment consisting of water and suspended matter. An overview can be found in table 5.

	literature	literature	VUB	VUB	VUB	MWTL	MWTL	MWTL
metal	median	standard deviation	median	standard deviation	n	median	standard deviation	n
Cu	10.8	0.9	10	0.4	87	10	0.9	658
Ni	9.9	0.9	9.3	0.5	87	9.1	0.8	578
Co	10.8	1.8	10.5	0.6	87	11.1	0.9	307
Cd	11.5	1.4	9.9	1	87	9.7	1.4	641
As	9.2	1.2	8.9	0.6	72	8.9	0.7	381
Pb	13.1	0.9	13.2	0.9	87	13.3	0.8	701
Zn	11.7	1.2	10.3	0.4	87	10.8	0.8	701

Table 5: In this table characteristic values of the $\log(K_d)$ distribution for metals in an environment consisting of water and suspended matter can be found. The median and the standard deviation are given for literature values, the VUB data base and the MWTL data base. For the VUB and the MWTL data base the number of measurements is also included.

Most values correspond quite well with the literature ones. However, a remarkable difference is that the standard deviations of the VUB data base are a lot smaller than the literature ones. A possible explanation is that the measurements for the VUB data base were only performed in the months February and March, thus resulting in less variation, as seasonally varying biological activity influences the K_d [15]. Furthermore, the median of the K_d distribution for literature values and the one from the data bases differs a lot for cadmium and zinc.

4.4 Exploratory analysis

In this subsection several analyses are performed to identify possible correlations and patterns in the data. The 2011-2016 VUB data will be used to perform these analyses, as data is available for all metals corresponding to unique salinity and SPM values.

4.4.1 Principal component analysis

A principal component analysis (PCA) is performed on the data to identify possible patterns and correlations in the data base. Background information on PCA can be found in appendix A. The PCA will be based on mean-centered and scaled data and furthermore the Spearman distance will be used. The use of the

Spearman distance means that one uses the ranked version of the data, instead of their actual values. The advantage is that all types of correlations will become clear, rather than just linear correlations.
To ensure good visibility, two separate PCA's will be made. The first one includes the variables SAL, SPM, YEAR and all total metal concentrations. The second one includes SAL, SPM, YEAR and the K_d values of all metals.

PCA with the *MeT*'s

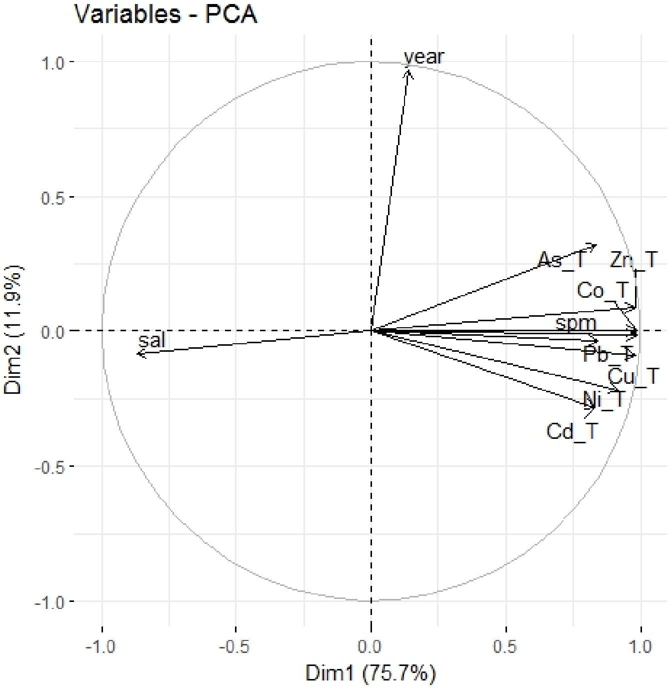


Figure 10: Loadings plot of the PCA with scaled, mean-centered data, based on the Spearman distance. The variables SAL, SPM, YEAR and all MeT's are included. The arrows correspond to the loadings of these variables for the first two principal components represented as a vector. The first and second principal component explain 76 and 12 percent of the total variance respectively.

Figure 10 shows the loadings plot of the PCA with the total metal concentrations. Together the first two PCs contain about 88 percent of the total variance in the data base. This is very high amount, so the two-dimensional figures are very reliable. For the visual interpretation of the PCA, one has to take into account that a way bigger part of the total variance is explained by the first than by the second principal component: 76 in contrast to 12 percent. Consequently, the vertical distances are exaggerated. The angle between the arrows of the variables $YEAR$ and AsT is, for instance, as big as that between AsT and CdT.

However, the associations between AsT and CdT and between $YEAR$ and AsT are of a different order.

Several things can be learned from this image. First one can see that all MeT's are positively correlated to one another and to SPM. Furthermore, all MeT's and SPM are negatively correlated to SAL. Last, one observes that $YEAR$ is not correlated to any of the other variables. All of these correlations make sense from a biochemical point of view and will be discussed separately.

Explanation of the correlations

The positive correlation between the MeT's and SPM is easily explained by the fact that metal particles are sorbed to suspended particles. A higher concentration of suspended particles thus corresponds to a higher concentration of metal particles.

The positive correlation of the MeT's to one another makes sense, as wastewater treatment plants are the main source of most metals. Therefore, when there is a high metal content of a certain metal at a certain place, this implies that it is close to the effluent of a wastewater treatment plant and thus it makes sense that for the other metal types the metal content is also high there.

Next, we look at the negative correlation between SAL and SPM, which can be explained by a mutual relation of both variables to the season. SAL is low in winter and high in summer, as was already stated in paragraph 4.1.1. Mud concentrations are highest in winter and lowest in summer, which directly implies high SPM in winter and low SPM in summer [18]. Combining the seasonal patterns for SAL and SPM explains the negative correlation between the two.

The last correlation, the negative one between SAL and the MeT's, follows directly from the negative correlation between SAL and SPM and the positive correlation between SPM and the MeT's.

PCA with the K_d's

Figure 11 displays the loadings plot of the PCA with the $\mathrm{K_d}$'s. Because the first two components only account for 64 percent of the total variance in the data base, the two-dimensional representation is not that accurate for all variables.

For variables, corresponding to arrows with length close to one, i.e. with their head close to the unit circle, almost all of their variance is contained in the first two principal components. This implies that they can be compared in the two-dimensional plot to identify correlations. Smaller arrows correspond to variables, for which less of their variance is contained in the first two principal components. Therefore they cannot be compared based on this figure.

Unlike for the MeT's, for the Kd's there is no correlation between all metals. However, several groups of metals that are correlated to one another can be identified. One example is the group As, Cd and Cu, which are all correlated to one another. The correlation between such metals that can be grouped together exists, because these metals share similar chemical properties.

Furthermore, $YEAR$ is correlated to the K_d of nickel and lead. Why this is the case is not immediately clear. It can however not be explained by a change in the total metal content, because no correlation between $YEAR$ and NiT or PbT exists, as we saw in figure 10.

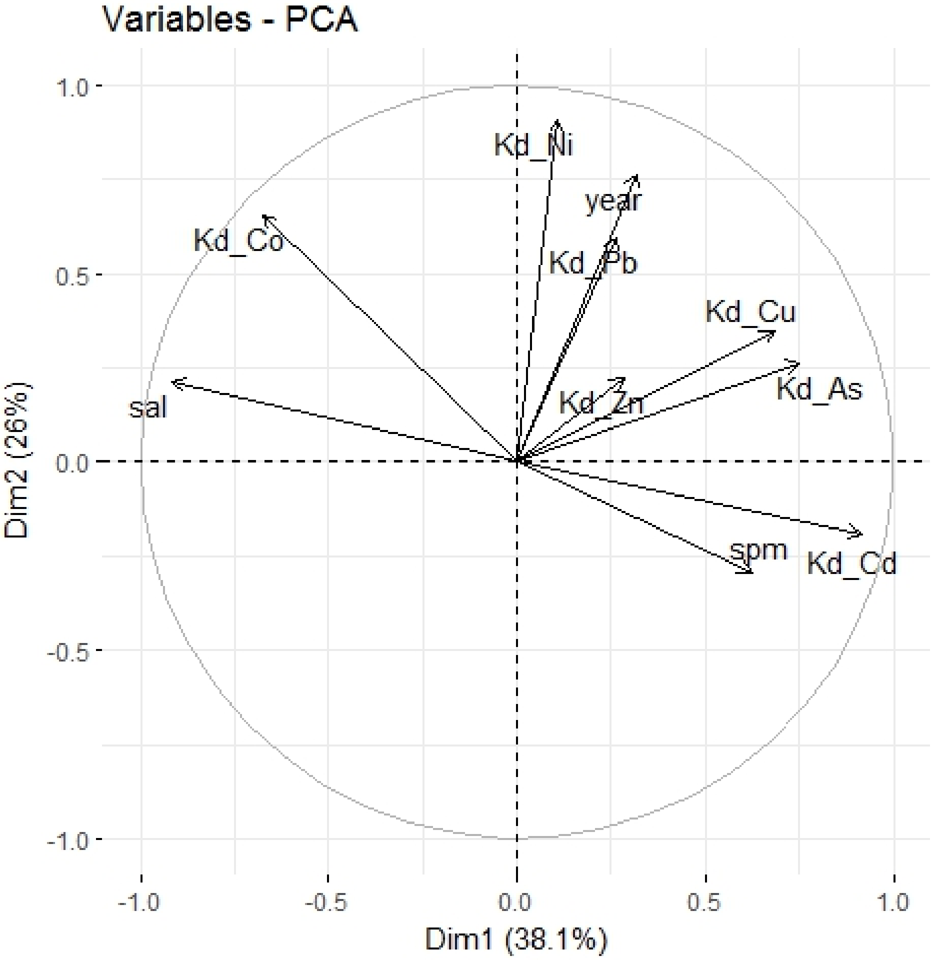


Figure 11: Loadings plot of the PCA with scaled, mean-centred data, based on the Spearman distance. The variables SAL, SPM, YEAR and the K_d for all metals are included. The arrows correspond to the loadings of these variables for the first two principal components represented as a vector. The first and second PC explain 38 and 26 percent of the total variance respectively.

5 Statistical learning methods

Statistical learning methods are used to obtain information from data bases. Usually the goal is to make predictions about a certain variable based on a number of other variables. The variable on which predictions are made is called the response variable and is denoted as Y. The variables which are used to make this prediction are called explanatory variables and denoted as $X = X_1, ..., X_J$.

In practice one usually has a data base containing data on both the response and the explanatory variables. Methods which use all of this data are called *supervised* methods. The best-known example of a supervised statistical learning method is linear regression. The counterparts of supervised methods are *unsupervised* methods. These methods are characterised by the absence of response variable Y. Usually the goal of such methods is not to make predictions, but rather to understand the relationships between a number of variables [5].

PCA is the only unsupervised statistical learning method that will be used in this research. A detailed description of PCA can be found in Appendix A. All other methods are supervised ones, because they require data on the response variable. The will be used to build predictive statistical models. The following four supervised methods will be used.

- Principal components regression
- Partial least squares regression
- Random forest
- Gradient boosting machine

The first two methods are based on linear regression, whilst the latter ones are assemblies of decision trees. Before going into detail about these models, first some general notes on predictive statistical models will be given.

5.1 Predictive statistical models

We will discuss predictive statistical models that try to approximate response variable Y, based on explanatory variables $X_1, ..., X_J$. To this end one assumes that a certain explicit relationship exists between the explanatory and the predictor variables. Say

$$Y = f(X, \beta_1, ..., \beta_n), \tag{6}$$

where $X = \big(X_1, ..., X_J\big)$ is the matrix of explanatory variables and $\beta_1, ..., \beta_n$ are the model coefficients. The form of $f(X)$ is fixed by the choice of statistical learning method. Given available data one determines the set of parameters $\hat{\beta}_1, ..., \hat{\beta}_n$, so that this function best fits the data. The predictive statistical model thus has the following form

$$\hat{Y} = f(X, \hat{\beta}_1, ..., \hat{\beta}_n) = \hat{f}(X). \tag{7}$$

To asses the performance of such models, and hence to be able to compare them, a metric is required that describes the model performance.

5.1.1 Model evaluation

The usual metric to describe the performance of a certain statistical model is the coefficient of determination, denoted as R^2. It describes the ratio between the variance that is explained by the model and the total variance in the data base. Although multiple definitions have been used in statistical literature, the preferable one, as given by Kvålseth [6], is

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}, \tag{8}$$

where $y = y_1, ..., y_n$ is a sample of the response variable, $\hat{y} = \hat{y}_1, ..., \hat{y}_n$ is the approximation of that sample generated by the model and $\bar{y}$ is the sample average of y.

R^2, as defined in equation 8, takes values on the interval $(-\infty, 1]$. The model which predicts the sample average of the response variable, $\bar{y}$, for each observation y_i has an R^2 of zero. Thus for a reasonable model, which predicts better than the sample average, the R^2 will be a value on the interval $[0, 1]$.

A problem with the R^2 coefficient is that it fails to account for the number of explanatory variables. If a number of useless, in terms of explained variance, variables is added to a model, then the R^2 score would remain the same. This issue can be resolved by taking into account the degrees of freedom of the estimates in both the numerator and the denominator of equation 8. This adjusted version of the R^2 is defined as

$$R^2_{adj} = 1 - (1 - R^2)\frac{n-1}{n-p-1}, \tag{9}$$

where n is the sample size and p is the number of explanatory variables in the model. For models where the number of explanatory variables is way smaller than the sample size of the response variable, $R^2_{adj} \approx R^2$.

The adjusted R^2 is the metric that will be used in this research to evaluate model performance. In the following the notation R^2 will be used to denote the adjusted R^2 metric, unless stated differently.

5.1.2 Parameter estimation

Typically, the performance of such a model depends on a number manually controllable internal parameters, also called tunable features, as they should be tuned to result in a well-functioning model [7]. These are not to be confused with the model coefficients, introduced at the beginning of this section. To understand how one chooses them properly, the concept of a training and a test set has to be introduced.

A training set is used to train the model, i.e. to compute the explicit formula $\hat{Y} = \hat{f}(X)$. The test set on the other hand, is used to evaluate the model performance, and hence to be able to choose the optimal parameter settings.

One could of course also simply set the model parameters in such a way that they optimise performance on the training set. However, by doing that, the model becomes highly specific to the training data and will predict worse on new data, this phenomenon is called overfitting. To avoid overfitting a test set is used to validate the model. Depending on the available data, different validation techniques are available.

5.1.3 Validation techniques

The validation techniques used for this research are discussed here.

Separate test and training set

If one has two independent data bases at hand, then one can simply use one data base as the training set and the other as the test set.

However, if one only has one data base, then one can split that data base into two parts, using one part as a training data base and the other one as a test data base. The disadvantage of this technique is that not all of the data can be used to train the model. A clever way to be able to use all data is by k-fold cross-validation.

k-fold cross-validation

The idea of k-fold cross-validation is the following. First, the data base is split into k groups, or folds. Then, one of these folds is chosen to be the test set and the other $k-1$ folds are used as the training set. After that, another fold is chosen to be the test set, leaving the other folds as the training set. This process is repeated until every fold is used exactly once as a test set.

However, the use of an independent validation data set, if available, is preferable, as possible problems of a model, resulting from intrinsic properties of a certain data base, cannot be traced by k-fold cross-validation.

5.2 Overview of the used statistical learning methods

This section gives an overview of the different statistical learning methods that will be used in this research to build predictive models.

5.2.1 Principal Components Regression

In principal components regression (PCR) a linear regression model is built using the principal components of a data base rather than the original variables. Background information on principal components can be found in Appendix A. Only a number of the principal components, containing most of the variability in the data base, is used to build the regression model. Consider the case of a data base with response variable Y, explanatory variables $X_1, ..., X_J$ and corresponding PCs $Z_1, ..., Z_J$. The PCR model can be written as

$$\hat{Y} = \sum_{i=1}^{M} \beta_i Z_i \, , \tag{10}$$

where $M \leq J$ is the number of principal components that is used for the model, and $\beta_1, ..., \beta_M$ are the regression coefficients. If $M = J$, then the PCR model is just a simple linear regression model.

The idea behind this method is that the principal components, which contain most of the variance of the data base, are most strongly associated with the response variable. Using too many principal components can result in overfitting. It is highly advisable to transform all explanatory variables so that they are mean-centred and have standard deviation one, as discussed in appendix A.

PCR relies on a number of assumptions to assure that the computed regression parameters are the optimal ones and that it functions well as a predictive model. An extensive overview of these assumptions can be found in appendix F.

5.2.2 Partial least squares regression

Partial least squares regression (PLS) is similar to PCR. Here too, first a number of components, directions in the explanatory data space, is determined and afterwards linear regression is performed on them. The difference between both methods lies in the way these components are constructed. As PCA is an unsupervised method, the principal components used for PCR are constructed in an unsupervised way. In PLS however, the components are constructed in a supervised way. The process by which the PLS components are constructed can be found in Appendix B.

PLS relies on the same assumptions as PCR, which can be found in appendix F.

5.2.3 Random Forest

A random forest is a type of statistical learning method, based on the concept of decision trees. A single decision tree can be used to build easily interpretable statistical models. For theoretical details on decision trees see Appendix C. Their main drawback however, is that they have high variance. That is to say, they depend strongly on the training set. If one would split the training set in two halves and train a decision tree model on both halves, the resulting models can be quite different. This is obviously an unwanted property of decision trees. Multiple concepts have been developed to deal with this problem. A random forest is one of those solutions.

The trees that constitute the forest

The idea behind a random forest is that a number of different predictive decision tree models is generated on different training sets, say $\hat{f}^1(x), \hat{f}^2(x), ..., \hat{f}^B(x)$. These models are averaged to obtain a predictive model given by

$$\hat{f}_{avg}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^b(x). \tag{11}$$

Although the single decision trees are easily interpretable, a random forest as a whole is not.

The question that immediately arises in a practical setting is how one obtains these different training sets. This is done by taking bootstrap samples from one original training set. Details on bootstrap sampling can be found in appendix D. The bootstrapping brings the randomness into the random forest. However, when a large number of trees is used the effect of this randomness is minimised.

One problem remains however: the trees that are created using the bootstrapped sample tend to be quite similar, which is why they are highly correlated. The problem with this is that the average of the correlated assembly of trees will remain quite variable.

Decorrelation of the trees

The trees in a random forest are decorrelated by restricting the number of explanatory variables that are considered at each cut. By doing this weaker predictors get a chance to be chosen as well. The result is that the trees will differ more, thus reducing the variance of the entire assembly of trees.

Tunable features

We just encountered the first tunable feature of a random forest model, the number of explanatory variables that is considered at each cut. It is typically denoted as m.

Another feature is required to prevent overfitting of the individual trees. This will happen if they are allowed to grow too deeply, that is to say if a large number of cuts is allowed. The straightforward way to avoid this is by restricting the number of cuts, and thus the number of terminal nodes. Another way to achieve the same goal is to set a minimum to the size of the terminal nodes, i.e. the number of data points in a terminal node.

Last, the number of decision trees, which constitute the random forest has to be chosen. This should be sufficiently large, to reduce the variance of the forest. Fortunately, one does not have to worry about choosing a too high number of trees, as overfitting on account of the number of trees is not possible [19]. Summarising, there are three tunable features:

- the number of explanatory variables considered at each cut,
- the maximum number of terminal nodes, and
- the minimum size of the terminal nodes.

5.2.4 Gradient boosting machine

A gradient boosting machine, just as a random forest, is created by growing a large number of decision trees. The difference between both methods is that for a gradient boosting machine the trees are not grown sequentially, rather than independently. This means that when growing a new tree, the information from the previous ones can be used.

Construction of a gradient boosting machine

Gradient boosting machines are constructed in the following way. At each step a decision tree is grown, based not solely on the response variable Y, but also on the previously grown trees. This is done by fitting the decision trees to the residuals of the model at a given step.

A brief description of the construction progress will be given. One starts with an empty model, i.e. $\hat{f}(X) = 0$ and full residuals, i.e. $r = Y$. Then a shallow tree, i.e. a tree with a small number of cuts, is fitted to the data (X, r). Afterwards, a shrunken version of this tree is added to the model and the residuals are updated by subtracting the shrunken prediction of the current tree. This procedure can be described schematically in an algorithmic form.

1. Set $\hat{f}(X) = 0$ and $r = Y$.

2. For b = 1, 2, ..., B repeat:

(a) Fit a tree $\hat{f}^b$ with d cuts to the training data (X, r).

(b) Update $\hat{f}$ by adding a shrunken version of the new tree: $\hat{f}(X) \leftarrow \hat{f}(X) + \lambda \hat{f}^b(X)$

(c) Update the residuals: $r \leftarrow r - \lambda \hat{f}^b(X)$

3. Output the gradient boosted model: $\hat{f}(X) = \sum_{b=1}^{B} \lambda \hat{f}^b(X)$.

In the above algorithm X is the matrix of explanatory variables $\big(X_1 ... X_J\big)$, r is the vector of residuals, B is the total number of decision trees, d is the number of cuts in a single decision tree and λ is the shrinkage parameter.

In practice the gradient boosting algorithm is usually used to compute the gradient boosting model.

Tunable features

A gradient boosting machine has three tunable features, which we already encountered in the previous paragraph. First there is the number of decision trees B. Unlike for a random forest, it is possible to overfit the model if B is too large. However, this overfitting will go slowly, thanks to λ.

The shrinkage parameter λ is the second tunable feature. It controls the rate at which the gradient boosting machine learns. The choices of λ and B are inherently related to one another. The smaller λ one chooses, the bigger B is required to assure good model performance.

The last tunable feature is d, the number of cuts in a single decision tree. It is more commonly referred to as the interaction depth, as it controls the interaction order of the model. A higher d leads to a more complex model, but choosing a too high d will result in overfitting.

5.3 Used R packages

The programming for this research is done in R. The following packages are used to execute the aforementioned statistical learning methods.

- Package 'pls' [20] (PCR and PLS)
- Package 'randomForest' [21] (Random Forest)
- Package 'gbm' [22] (Gradient Boosting Machine)

6 Theory

For this research some theory on statistical transformations, and more specifically their effect on decision trees, has been investigated. The lemmas and theorems, that are presented in this chapter, are devised in the context of this research. They lead up to the useful proof that decision tree based statistical models are not affected by a data transformation, given that it preserves the order of the data points.

The framework that will be constructed here deals with the statistical estimation of a certain so-called response variable, which in the following will be denoted as Y. This response variable is predicted using a number of explanatory variables, denoted as $X = X_1, ..., X_J$. In practice one deals with a data base containing m data points, which means that both the response and the explanatory variables are vectors of length m.

6.1 The data space

Each data point $d \in \Delta$ consists of a coordinate in the explanatory space $x = (x_1, x_2, ..., x_J) \in \Omega$ and a value in the response space $y \in \Psi$. The formal definitions of all of these spaces are given below.

Definition 6.1 (Explanatory space)**.** *The explanatory space is defined as* $\Omega = \Omega^1 \times ... \times \Omega^J$*, where* Ω^j *is the set of possible values for* $X_j, \quad 1 \leq j \leq J$*.*

Definition 6.2 (Response space)**.** *The response space,* Ψ*, is the set of possible values for* Y*.*

Definition 6.3 (Data space)**.** *The data space,* Δ*, is the union of the explanatory and the response space. Formally,* $\Delta = \Omega \cup \Psi$*.*

6.2 The effects of data transformations on decision tree based models

For linear regression models it is well-known that data transformations can improve their performance [9]. Unfortunately, the effects of data transformations on decision tree based methods is not as well documented. Therefore, they will be studied in this section.

Basic concepts regarding decision trees are discussed in appendix D. For the investigations in this section some more definitions have to be introduced.

Definition 6.4 (Transformation)**.** *The transformation of explanatory variable* X_j *is a function* $t : A \rightarrow B$*, where* $\Omega^j \subseteq A$ *and* $\underset{t}{\Omega^j} \subseteq B$*. Here* $\underset{t}{\Omega^j} = \{t(X_j) | X_j \in \Omega^j\}$*.*

Definition 6.5 (Transformed box)**.** *The transformed version of a box R_k^n is defined as*

$$\underset{t}{R}{}_k^n = \{t(r)|r \in R_k^n\}. \tag{12}$$

Definition 6.6 (Cut of a transformed box)**.** *A cut of the transformed box $\underset{t}{R}{}_k^n$ is defined by the triplet $(j, s, \underset{t}{R}{}_k^n)$. The cut is such that box $\underset{t}{R}{}_k^n$ is split into the two half-boxes $\underset{t}{R}{}_{k,1}^n$ and $\underset{t}{R}{}_{k,2}^n$. Formally,*
$\underset{t}{R}{}_{k,1}^n(j, s) = \{t(X)|t(X_j) < s \text{ and } t(X) \in \underset{t}{R}{}_k^n\}$
$\underset{t}{R}{}_{k,2}^n(j, s) = \{t(X)|t(X_j) \geq s \text{ and } t(X) \in \underset{t}{R}{}_k^n\}$.

Definition 6.7 (Transformed residual sum of squares)**.** *The residual sum of squares of box $\underset{t}{R}{}_k^n$ is defined by*

$$\underset{t}{RSS}(\underset{t}{R}{}_k^n) = \sum_{i:t(x_i)\in \underset{t}{R}{}_k^n} (y_i - \hat{y}_{\underset{t}{R}{}_k^n})^2. \tag{13}$$

We now consider the scenario where the explanatory variables are all transformed by the same transformation and the response variable remains untransformed. An important question to answer is whether the decision tree based models trained on the transformed and untransformed data will be the same. In the following it will be proven that this question is true if the transformation retains the ordering of the data points.

Intuitively this makes sense, as the distance between the data points in the explanatory space is not important for the outcome of the decision tree. All that matters is that the data points can be separated from one another in a similar fashion. If the ordering between the data points were to be disturbed this would no longer be possible. Note that a transformation retains the ordering of the data points, if it is a strictly monotonic function. The ideas mentioned here will be formalised and the intuitive answer will be proven.

Definition 6.8 (Strictly increasing function)**.** *A function $t : A \to B$ is called strictly increasing if for any $a, a' \in A$ such that $a < a'$, then $t(a) < t(a')$.*

Definition 6.9 (Strictly decreasing function)**.** *A function $t : A \to B$ is called strictly decreasing if for any $a, a' \in A$ such that $a < a'$, then $t(a) > t(a')$.*

Definition 6.10. *A function $t : A \to B$ is called strictly monotonic if it is either strictly increasing or strictly decreasing.*

Lemma 6.1. *The order of the data points is preserved (or reversed) upon transformation if that transformation is a strictly monotonic function.*

Proof. Given a monotonically increasing transformation $t : A \to B$ and explanatory variable $X_j \subseteq A$. Let $x^j_1 \leq x^j_2 ... \leq x^j_m$ the elements of X_j. Then $t(x^j_1) \leq t(x^j_2) \leq ... \leq t(x^j_m)$ or $t(x^j_1) \geq t(x^j_2) \geq ... \geq t(x^j_m)$, due to the strict monotonicity of t. □

Before generalising for decision tree models, first it is proven in theorem 6.3 that for an optimal cut through the explanatory space given by (j, s, R^n_k), there exists also an optimal cut though the transformed explanatory given by $(j, t(s), \underset{t}{R}^n_k)$.

Lemma 6.2. *Given a transformation $t : A \to B$, explanatory variables $X_1, ..., X_J$ and response variable Y. If t is strictly monotonic, then*

$$RSS\big(R^n_{k,1}(j,s)\big) + RSS\big(R^n_{k,2}(j,s)\big) = \underset{t}{RSS}\big(\underset{t}{R}^n_{k,1}(j,t(s))\big) + \underset{t}{RSS}\big(\underset{t}{R}^n_{k,2}(j,t(s))\big) \tag{14}$$

Proof. The left side of the equation reads

$$\sum_{i:x_i \in R^n_{k,1}(j,s)} (y_i - \hat{y}_{R^n_{k,1}(j,s)})^2 + \sum_{i:x_i \in R^n_{k,2}(j,s)} (y_i - \hat{y}_{R^n_{k,2}(j,s)})^2,$$

and the right side reads

$$\sum_{i:t(x_i) \in \underset{t}{R}^n_{k,1}(j,t(s))} (y_i - \hat{y}_{\underset{t}{R}^n_{k,1}(j,t(s))})^2 + \sum_{i:t(x_i) \in \underset{t}{R}^n_{k,2}(j,t(s))} (y_i - \hat{y}_{\underset{t}{R}^n_{k,2}(j,t(s))})^2.$$

Equality holds if the indices under the sums over both half-boxes are equal. If this is the case for either one of the half-boxes, then it is trivially also the case for the other half-box. It thus suffices to consider only the indices of one half-box.

$$\begin{aligned}
&i : x_i \in R^n_{k,1}(j,s) = \\
&i : x_i \in \{X | X_j < s \text{ and } X \in R^n_k\} = (\star) \\
&i : t(x_i) \in \{t(X) | t(X_j) < t(s) \text{ and } t(X) \in \underset{t}{R}^n_k\} = \\
&i : t(x_i) \in \underset{t}{R}^n_{k,1}(j,t(s))
\end{aligned}$$

Equality $(\star)$ holds, because of the strict monotonicity of t. If t were not monotonic, then data points could fall into a different box after the transformation. The strict monotonicity is required, as otherwise data points could be transformed onto the boundary of another box. □

Theorem 6.3. *Given explanatory variables $X_1, ..., X_J$, response variable Y and strictly monotonic transformation $t : A \to B$.*
Let (σ, γ, R_k^n) the optimal n'th cut of the explanatory space, then $(\sigma, t(\gamma), \underset{t}{R}{}_k^n)$ is an optimal cut of the space of transformed explanatory variables.

Proof.

$$\underset{t}{RSS}\big(\underset{t}{R}{}_{k,1}^n(\sigma, t(\gamma)\big) + \underset{t}{RSS}\big(\underset{t}{R}{}_{k,2}^n(\sigma, t(\gamma)\big) = (\star)$$

$$RSS\big(R_{k,1}^n(\sigma, \gamma)\big) + RSS\big(R_{k,2}^n(\sigma, \gamma)\big) =$$

$$\min\big\{RSS\big(R_{k,1}^n(j, s)\big) + RSS\big(R_{k,2}^n(j, s)\big)\big\} = (\star\star)$$

$$\min\big\{\underset{t}{RSS}\big(\underset{t}{R}{}_{k,1}^n(j, s')\big) + \underset{t}{RSS}\big(\underset{t}{R}{}_{k,2}^n(j, s')\big)\big\}$$

Equality $(\star)$ follows directly from Lemma 6.2. To see why equality $(\star\star)$ holds, one should realise that for each s one can choose s' such that $s' = t(s)$. Thus, it follows from Lemma 6.2 that the minima are equal. □

Consider two decision tree models: one trained on untransformed explanatory data, constructed by the cuts $(\sigma, \gamma, R_k^n)_1, ..., (\sigma, \gamma, R_k^n)_n$, and another one trained on transformed explanatory data, constructed by the cuts $(\sigma, t(\gamma), \underset{t}{R}{}_k^n)_1, ..., (\sigma, t(\gamma), \underset{t}{R}{}_k^n)_n$. The predictions of these models depend only on the final partitioning of the explanatory respectively transformed explanatory space, and thus on the final partitioning of the data points. It is clear that both decision tree models have the same final partitioning of the data points, and therefore also yield identical predictions.
However, theorem 6.3 does not state that either the optimal cut through the explanatory space or the one through the transformed explanatory space is a unique one. There could thus be other cuts that are also optimal. This can lead to the hypothetical scenario, where the decision tree model for the untransformed and transformed data is different.
However, in practice we are never dealing with a single decision tree, but rather with a big number of decision trees over which one averages in one way or another. So, even though hypothetically the transformation of data can result in minor differences for a single decision tree, the effect on an assembly of decision trees, such as a random forest or a gradient boosting machine, is negligible.
This insight leads to the conclusion of this chapter, which will be formulated as a theorem.

Theorem 6.4. *Given explanatory variables $X_1, ..., X_J$, response variable Y and strictly monotonic transformation $t : A \to B$.*

Then, predictions of decision tree based models, such as a random forest and a gradient boosting machine, trained on either transformed explanatory variables $t(X_1), ..., t(X_J)$ and untransformed Y, or untransformed variables $X_1, ..., X_J$ and untransformed Y are identical.

Proof. This follows from theorem 6.3 and the above explanation. □

6.3 Order-preserving transformations

Order-preservingness (recall that this is equivalent to strict monotonicity) is an essential property for transformations, as became clear from the analysis in the previous section. Most of the environmental data that we are dealing with in this research, such as SAL and SPM are more or less log-normally distributed, as we saw in section 4.1. Thus, the logarithmic transformation is the one that will mostly be used for this research. For data that is not as close to a log-normal distribution, the Box-Cox transformation is a common choice. This is a polynomial transformation of the following form

$$t_{bc}^{\lambda}(x) = \begin{cases} \frac{x^{\lambda-1}}{\lambda}, & \text{for } \lambda \in \mathbb{R} \setminus \{0\} \\ \log(x), & \text{for } \lambda = 0 \end{cases} . \tag{15}$$

So actually the logarithmic transformation is a special case of the Box-Cox transformation. Note that its domain is just the positive real numbers. The environmental variables considered in this research are all strictly positive. But even if this were not the case, a simple linear transformation (which is obviously an order-preserving one) can make any type of data positive. It will now be proven that the Box-Cox transformation is order-preserving for all λ.

Theorem 6.5. *The Box-Cox transformation, as defined in equation 15, is strictly monotonic (and thus order-preserving) for all $\lambda \in \mathbb{R}$.*

Proof. Three cases will be considered: (i) $\lambda = 0$, (ii) $\lambda < 0$ and (iii) $\lambda > 0$.

Let $a, b \in \mathbb{R}^+$ such that $a < b$. We will now prove for all three cases that $t_{bc}^{\lambda}(a) < t_{bc}^{\lambda}(b)$.

Case (i):$t_{bc}^{0}(a) = log(a) < log(b) = t_{bc}^{0}(b)$.

Case (ii): $a^{\lambda} > b^{\lambda} \Rightarrow a^{\lambda} - 1 > b^{\lambda} - 1 \Rightarrow \frac{a^{\lambda}-1}{\lambda} < \frac{b^{\lambda}-1}{\lambda}$ (as $\frac{1}{\lambda} < 0$).

Thus, $t_{bc}^{\lambda}(a) = \frac{a^{\lambda-1}}{\lambda} < \frac{b^{\lambda}-1}{\lambda} = t_{bc}^{\lambda}(b)$.

Case (iii): $a^{\lambda} < b^{\lambda} \Rightarrow a^{\lambda} - 1 < b^{\lambda} - 1 \Rightarrow \frac{a^{\lambda}-1}{\lambda} < \frac{b^{\lambda}-1}{\lambda}$ (as $\frac{1}{\lambda} > 0$).

Thus, $t^{\lambda}_{bc}(a) = \frac{a^{\lambda-1}}{\lambda} < \frac{b^{\lambda}-1}{\lambda} = t^{\lambda}_{bc}(b)$. □

7 Methods: Models for MeT

In a practical setting, data on the total metal content might not be present. Or one could be in the scenario where values for SAL, SPM and $YEAR$ are computed by a computer model. In both cases a model is required to compute values for the total metal content of different metals, MeT. Such a model will be of the following form

$$\hat{MeT} = \hat{f}(SPM, SAL, YEAR). \tag{16}$$

The MWTL data base will be used as the training set on which the model is built, and the VUB data base will be used as the test set, to validate the model. This is the logical choice, as the MWTL data base is bigger than the VUB data base: it contains data from the years 1982 to 2017, whereas the VUB data base only contains data from the years 2011 to 2018. Moreover, the measurements in the MWTL data base are performed in all months of the year, whereas the VUB ones were only performed in February and March. Different models will be constructed based on three of the statistical learning methods described in the previous chapter: PCR, PLS and random forest.

Using these methods, different approaches can be used to build a model.

7.1 Overview of different modelling approaches

Ultimately, we wish to have a model that predicts the MeT for each of the different metal types included in the two data bases at hand.

The simplest approach is to construct a different model for each of the metal types, using only the data of that specific metal in the MWTL data base to train the model. This will yield a separate formula in the form of equation 16 for each of the metal types.

A second approach is to use all MWTL data to train a model. To differentiate between the individual metal types it is then necessary to include a variable that indicates the metal type as a number. This model has the following generic form

$$\hat{MeT}_{NR} = \hat{f}(SPM, SAL, YEAR, NR), \tag{17}$$

where NR is the number assigned to a specific metal type and MeT_{NR} is the total metal content belonging to the metal with metal number NR. For instance if the metal to which number 2 is assigned is nickel, then $MeT_2 = NiT$, the total nickel content.

Note that this approach only makes sense for decision tree based models, as they are able to partition the data space into different sections, separating the different metal types to a certain extent. This is necessary for good model performance, as there are big differences in MeT distribution for the different metals. Because linear regression based models lack such a separating property, this approach is unfit for them. PLS and PCR models will thus not be considered for this approach.

One could imagine a third approach, in which the metals are manually separated into different groups, based on certain criteria. The problem with this approach is that it is uncertain on which criteria to determine the grouping of the metals.

Evaluating the grouping problem from a mathematical point of view gives the following insight: a grouping only makes sense if there are different metal types, that have comparable MeT values for similar or even identical SPM, SAL and $YEAR$ values. Ideally, one would like to put these metal types together.

This is, however, exactly what a decision tree algorithm does. It is built to perform such a partitioning of the data. And moreover this partitioning is more refined than a manual one, inferred by a grouping, would be, as it can partition subsets of the data points of certain metals, rather than all of them.

Thus, it is clear that this last hypothetical approach does not make sense. The results of the random forest MeT models will be discussed per approach. They are briefly recapped below.

- Approach 1: An individual model is trained for each metal type.
- Approach 2: A single model, including the metal number as a variable, is trained for all metals.

For approach 1, one trains and test each individual model on a subset of the MWTL and VUB data base respectively. This subset contains the data points belonging to the metal corresponding to that model.

For approach 2 the entire MWTL and VUB data base are used as train and test data base respectively.

7.2 Data transformation

Linear regression based models depend on a number of assumptions, which should be met to assure properly functioning models. An overview of them can be found in appendix F. To assure these are met, the data has to be transformed. In general transforming the data to closely resemble normality is desirable. In our case a logarithmic transformation is performed on the variables SAL, SPM and MeT.

For decision tree based models these transformations are not necessary to assure good model performance. A proof of this is given in chapter 6. However, to assure comparable R^2 scores, the models should all be trained on the same data. Therefore, the decision tree based models will also use data with transformed SAL, SPM and MeT.

7.3 MeT distributions of the MWTL and VUB data bases

It is important to compare the MeT distributions of both data bases, as they influence the quality of the predictions of our models. An overview of the median, standard deviation and range of the MeT distributions for both databases can be found in table 6 below.

metal	MWTL median	MWTL standard deviation	MWTL range	VUB median	VUB standard deviation	VUB range
Cu	1.61	0.4	[0.1 ; 4.33]	1.25	0.44	[-0.40 ; 2.74]
Ni	1.61	0.37	[0.19 ; 3.43]	1.32	0.28	[-0.25 ; 2.34]
Co	0	0.54	[-1.59 ; 2.56]	-0.18	0.57	[-1.94 ; 1.47]
Cd	-0.94	0.73	[-3.22 ; 3.80]	-1.73	0.25	[-2.69 ; 0.10]
As	1.72	0.2	[0.64 ; 3.58]	1.36	0.11	[0.73 ; 2.39]
Pb	1.46	0.86	[-1.22 ; 4.11]	0.95	0.68	[-0.49 ; 3]
Zn	3.09	0.69	[0.33; 5.77]	3.09	0.66	[1.09 ; 4.74]

Table 6: Overview of the median, standard deviation and range of the log(MeT) values for the MWTL and the VUB data base.

In general the distributions for both data bases correspond quite well. There are some clear differences however. The most striking one is the cadmium distribution. The values of the VUB data base do fall within the range of the MWTL one, but the median and also the standard deviation are much smaller. For copper, arsenic and lead the median differs quite a bit as well.

Furthermore, it should be noted that for copper, nickel and cobalt the range of the VUB distributions is larger on the left side, than that of the MWTL ones. Thus smaller MeT values are observed for the VUB data base than for the MWTL one. The values that fall outside of the MWTL range will be difficult to predict for the models.

8 Methods: Models for K_d

The main goal of this research is to see whether it is possible to create a statistical model for the K_d of different metals, which predicts better than the sample average. This model will be built on the following variables SPM, SAL, MeT and $YEAR$. Such a model will be of the following form

$$\hat{K}_d = \hat{f}(SPM, SAL, MeT, YEAR). \tag{18}$$

Following the same reasoning as for the MeT models, the MWTL data base will be used as the training set on which the model is built, and the VUB data base will be used as the test set, to validate the model. Different models will be constructed based on the statistical learning methods described in the previous chapter: PCR, PLS, random forest and gradient boosting machine.

Using these methods, two different approaches can be used to build a model, just as for MeT.

8.1 Overview of the different modelling approaches

The first approach is again an individual model for each different metal type, whereas the second one is a single model for all metals.

Models based on the first approach are of the form of equation 18. Those based on the second approach are of the form

$$\hat{K}_{d,NR} = \hat{f}(SPM, SAL, MeT, YEAR, NR), \tag{19}$$

where NR is the metal number assigned to a specific metal type, just as for the MeT models and $K_{d,NR}$ is the K_d belonging to the metal with metal number NR.

As for the MeT models, subsets, per metal, of the data bases are used for the first approach. For the second approach the entire data bases are used.

8.2 Data transformation

For the same reasoning as was given for the MeT models in the previous chapter, all models will be trained on data with the following transformed variables: SAL, SPM, MeT and K_d.

9 Results: Models for MeT

An explanation of how the models in this chapter are constructed can be found in chapter 7. Detailed information on the used predictive statistical models can be found in chapter 5.

For the random forest models R^2 scores on the training set are based on the predictions for the out-of-bag data points. For all other models R^2 scores on training sets are obtained by 10-fold cross-validation.

9.1 Random Forest

9.1.1 Approach 1: Individual models for each metal type

metal	train	test	m	mn
Cu	0.61	0.35	2	140
Ni	0.86	0.56	1	175
Co	0.77	0.79	2	150
Cd	0.74	-0.13	2	100
As	0.61	0.54	2	160
Pb	0.79	0.74	2	150
Zn	0.76	0.82	1	130
average	0.74	0.51		

Table 7: R^2 values of the predictions of the random forest model for MeT. Individual models were trained for each metal type. The training set is the MWTL data for a single metal and the test set is the VUB data for a single metal. m is the number of explanatory variables considered at each cut and mn is the maximum number of terminal nodes. In the last row the weighted average of the R^2's is given.

Table 7 gives an overview of the R^2 scores of the models for the individual metals at optimal parameter settings. In general the random forest model predicts the MeT's for the different metals fairly well. On the training set the R^2 ranges from 0.61 to 0.86.

For Co, As, Pb and Zn the R^2 scores on the test set are approximately just as high as those on the training set. For the other three metals, however, there is a big discrepancy. The predictions for cadmium are even so bad that the R^2 is slightly negative, implying that they are worse than simply predicting the sample average.

To understand why the predictions on the VUB test set for these metals are worse than on the MWTL training set, we should have a look at the MeT distributions for the individual metals, given in table 6 in section 7.3.

For Cu and Ni, some values of the MeT of the VUB data fall outside the range of the MWTL data. It is inherently impossible for a random forest model to predict values outside of the range of the training set, which explains why for these metals the model performs worse on the test set than on the training set.

For Cd the distribution of MeT for the VUB data does fall within the range of the MWTL data, but they are shaped very differently, as one can see by the median and the standard deviations. These are -0.94 and 0.73 for the MWTL data and -1.73 and 0.25 for the VUB data. Histograms, visualising the MeT distributions of all metals for both data bases, can be found in section 4.1 and appendix E.

Possibly, this difference in the MeT distribution for both data bases explains why the model for cadmium does not predict well on the test set. Other explanations for the malfunctioning will be discussed in the discussion.

9.1.2 Approach 2: Single model for all metals

metal	train	test
Cu	0.67	0.47
Ni	0.86	0.50
Co	0.79	0.79
Cd	0.74	-0.10
As	0.61	0.52
Pb	0.80	0.75
Zn	0.75	0.82
average	0.75	0.54

Table 8: R^2 values of the predictions of the random forest model for MeT. One model, including the metal number as a variable, is trained for all metals. The training set is the entire MWTL data base and the test set the entire VUB data base. The model is trained at $m = 3$ and $mn = 575$. In the last row the weighted average of the R^2's is given.

The results for the single random forest model for all metals are given in table 8. The R^2 scores are computed per metal type, comparing the predictions for that metal type with the sample average for that metal type.

For the training set, the results for all metals are equally well or slightly better than for the models for the individual metals. For the test set, a big improvement, compared to the individual model, can be observed for Cu, from 0.35 to 0.47. For Ni there is a slight decrease in R^2, from 0.56 to 0.50. The R^2 scores on the test set for the other metal types stay more or less equal. The weighted average of all R^2's increases slightly from 0.51 to 0.54, compared to the individual models.

The improvement for Cu can probably be contributed to the measurements of the VUB data for Cu that fall outside of the range of the MWTL data for Cu. The individual model has problems predicting these values, whereas the single model does better, as all MWTL data is used as training data. Thus, a bigger spectrum of MeT values can be predicted for Cu.

Figure 12 shows the measured and predicted $\log(MeT)$ values for the VUB test set.

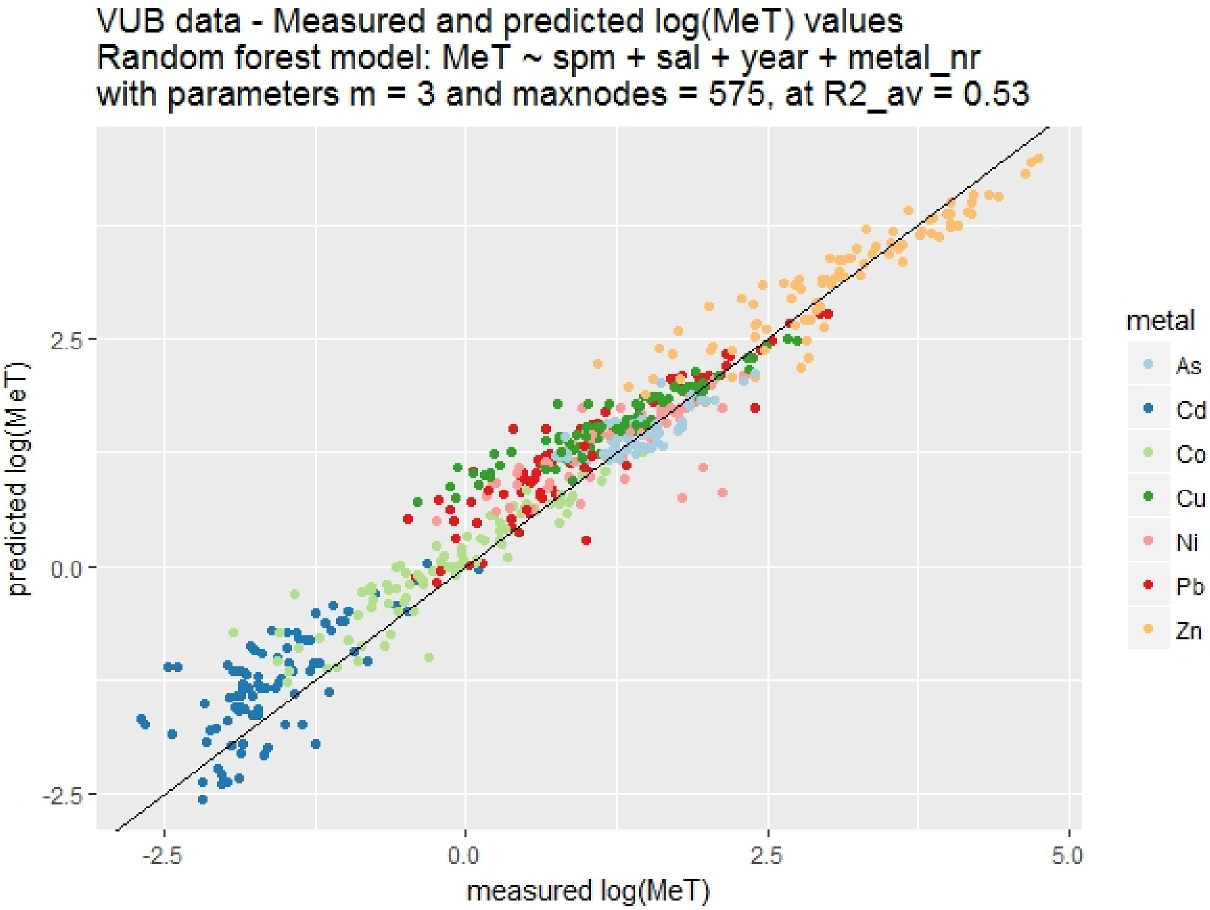


Figure 12: Plot with the measured and predicted $\log(MeT)$ values for the VUB test data set.

9.2 PLS

The R^2 scores of the PLS models for the individual metals can be found in table 9. Both the results for the models with $M = 2$ and $M = 3$ are presented. M is the number of PLS components used to train the model. The models using all PLS components, i.e. the ones with $M = 3$, correspond to simple multiple linear regression.

The results for the models with $M = 2$ and $M = 3$ are highly similar, both for the test and the training set. In general the results on the test data are comparable to those for the random forest model, but all their R^2 scores are approximately 0.10 lower.

For each of the metals the performance of the PLS models on the test set is a lot worse than the random forest models. This can also be seen by looking at the weighted average, which lies at 0.24 for the first, compared to 0.51 for the latter.

data metal	train $M = 2$	test $M = 2$	train $M = 3$	test $M = 3$
Cu	0.55	0.07	0.55	0.08
Ni	0.75	0.19	0.75	0.23
Co	0.75	0.60	0.75	0.60
Cd	0.53	-1.08	0.53	-0.93
As	0.50	0.25	0.51	0.31
Pb	0.76	0.66	0.76	0.67
Zn	0.69	0.68	0.69	0.70
average	0.64	0.24	0.65	0.24

Table 9: R^2 values of the predictions of the PLS model for MeT. Individual models were trained for each metal type. The training set is the MWTL data for a single metal and the test set is the VUB data for a single metal. M is the number of PLS components used to perform the linear regression. In the last row the weighted average of the R^2's is given.

9.3 PCR

The PCR results can be found in table 10. The results for $M = 3$ are identical for PCR and PLS, as this corresponds in both cases to simple multiple linear regression. They can be found in table 9.

The results for the PCR models are similar to those of the PLS models. Both performing a lot worse than the random forest models.

data metal	train $M = 2$	test $M = 2$
Cu	0.55	-0.04
Ni	0.75	0.05
Co	0.76	0.61
Cd	0.52	-1.98
As	0.48	-0.29
Pb	0.73	0.01
Zn	0.69	0.61
average	0.64	0.24

Table 10: R^2 values of the predictions of the PCR model for MeT. Individual models were trained for each metal type. The training set is the MWTL data for a single metal and the test set is the VUB data for a single metal. M is the number of principal components used to perform the linear regression. In the last row the weighted average of the R^2's is given.

9.4 Conclusion

It has become clear in this section that it is possible to develop a decent predictive model for MeT based on the variables SAL, SPM and $YEAR$.

The random forest model outperforms the PCR and PLS models by a big margin. On the training set the R^2 scores of the different metals vary between 0.61 and 0.86, with a weighted average of 0.75, both for the single model for all metals and the individual models for each of the metals. For PLS and PCR these values lie between 0.51 and 0.76, with a weighted average of 0.65.

The difference between the quality of the predictions on the test set for the random forest model and the PLS and PCR ones is even bigger. The single random forest model for all metals has a weighted average of 0.54, compared to 0.24 for the PCR and PLS models.

The quality of the predictions on the test set depends on the MeT distributions of the training and the test set. If the range of the MeT distribution for the training set lies within that of the test set, and the median is not similar, then the predictions are almost equally good on the test set as on the training set. This is the case for arsenic, cobalt, lead and zinc.

If however, the range of the test set lies outside of that of the training set, on either side, then this can cause reduced accuracy of the predictions on the test set. For copper and nickel this effect can be observed. For cobalt the range of the test set also lies slightly outside of that of the training set, but the predictions on the test set are not affected by it.

Cadmium is a last, exceptional case. On the training set the random forest models function well, with an R^2 score of 0.74. However, on the test set the models function badly, resulting in slightly negative R^2 values. Perhaps this is caused by the difference in MeT distributions for both data bases, which can be attributed to the fact that the VUB data base only contains winter measurements. But there could also be other causes, which will be mentioned in the discussion.

10 Results: Models for K_d - First attempt for all methods

In this chapter the results for the K_d models are presented. First PCR, PLS and random forest models are compared on the test and training data.

For the random forest models R^2 scores on the training set are based on the predictions for the out-of-bag data points. For all other models R^2 scores on training sets are obtained by 10-fold cross-validation.

10.1 PCR

data	train	test	train	test	train	test
metal	$M = 3$	$M = 3$	$M = 4$	$M = 4$	$M = 1$	$M = 1$
Cu	0.32	-1.99	0.36	-0.72		
Ni	0.34	-0.24	0.50	-2.25		
Co	0.38	-2.02	0.58	-0.02		
Cd	0.39	-0.5	0.42	-0.48	0.18	0.52
As	0.04	0.15	0.13	-0.24		
Pb	0.08	-0.45	0.42	-0.09		
Zn	0.11	-3.81	0.13	-2.29		
average	0.23	-1.30	0.35	-0.89		

Table 11: R^2 values of the predictions of the PCR model for Kd, given by equation 18. Individual models were trained for each metal type. The training set is the MWTL data for a single metal and the test set is the VUB data for a single metal. M is the number of principal components used to perform the linear regression. In the last row the weighted average of the R^2's is given.

For all metals, except for cadmium, the training and test results are best at either $M = 3$ or $M = 4$. Therefore, for cadmium two columns are added, containing the optimal performance of the model on the test set.

The linear regression assumptions, which can be found in appendix F, are checked for the models with three principal components, as these models are the ones performing best for most metals. Almost all of the assumptions are met for each of the metals. Only for the models for nickel and cadmium the assumption of independent errors is violated. However, in the context of this research, this violation is not that relevant. An overview of the verification of all assumptions can be found in appendix G.

10.2 PLS

The results for PLS with three components are presented in table 12. The results for PLS with four components are equal to those of PCR with four components and can be found in table 11.

For Cd and As the best results on the test set are for the model with one PLS component. Therefore, these

data metal	train $M = 3$	test $M = 3$	train $M = 1$	test $M = 1$
Cu	0.35	-1.77		
Ni	0.48	-0.37		
Co	0.56	-0.97		
Cd	0.41	-0.12	0.31	0.76
As	0.12	0.15	0.06	0.27
Pb	0.42	-0.05		
Zn	0.13	-2.75		
average	0.35	-0.79		

Table 12: R^2 values of the predictions of the PLS model for Kd, given by equation 18. Individual models were trained for each metal type. The training set is the MWTL data for a single metal and the test set is the VUB data for a single metal. M is the number of PLS components used to perform the linear regression. In the last row the weighted average of the R^2's is given.

results are also included in the table.

The regression assumptions are checked for the models with three principal components, as these models are the ones performing best for most metals. Almost all of the assumptions for PCR, which can be found in appendix F, are met for each of the metals. The assumption of independent errors is violated for the nickel, cadmium and arsenic models. However, in the context of this research, this violation is not that relevant. An overview of the verification of all assumptions can be found in appendix G.

10.3 Random forest

10.3.1 Individual models for each metal type

metal	train	test	mn
Cu	0.39	-0.73	90
Ni	0.66	-0.53	140
Co	0.71	0	50
Cd	0.63	0.67	90
	0.61	0.72	35
As	0.09	0.13	10
	0.07	0.15	20
Pb	0.37	0.11	125
Zn	0.18	-2.32	200
average	0.42	-0.38	

Table 13: R^2 values of the predictions of the random forest model for Kd, given by equation 18. Individual models were trained for each metal type. The training set is the MWTL data for a single metal and the test set is the VUB data for a single metal. mn is the maximum number of terminal nodes. All models are trained at $m = 3$. In the last row the weighted average of the R^2's is given.

Table 13 displays the results for the random forest models for the individual metals. Test and training scores are given at optimal parameter setting for the training set. For cadmium and arsenic the R^2 scores at optimal parameter settings for the test set are also displayed.

For the metals with a negative score on the test set it does not make sense to include the optimal parameter settings for the test set. These will always be at an mn close to zero, as then approximately the sample average is predicted for all data points, thus resulting in an R^2 score close to zero, which is indeed the optimum, but not insightful at all.

10.3.2 Single model for all metals

metal	train	test
Cu	0.42	-0.23
Ni	0.66	-0.43
Co	0.73	-0.25
Cd	0.63	0.54
As	0.07	0.08
Pb	0.41	0.12
Zn	0.19	-2.03
average	0.44	-0.32

Table 14: R^2 values of the predictions of the random forest model for Kd, given by equation 19. A single model was trained for all metal types. The training set is the whole MWTL data base and the test set is the whole VUB data base. The results are given for parameter settings $mn = 1000$ and $m = 3$. In the last row the weighted average of the R^2's is given.

The results of the single random forest model are displayed in table 14. The R^2 scores on the training set are comparable to those of the random forest model for the individual metals.

The scores on the test set improve for copper, nickel, but decrease for cobalt and cadmium, leaving the averaged test score quite similar.

Figures 13 and 14 visualise the measured and predicted $\log(K_d)$ values for all data points for the single random forest model at parameter settings $mn = 1000$ and $m = 3$.

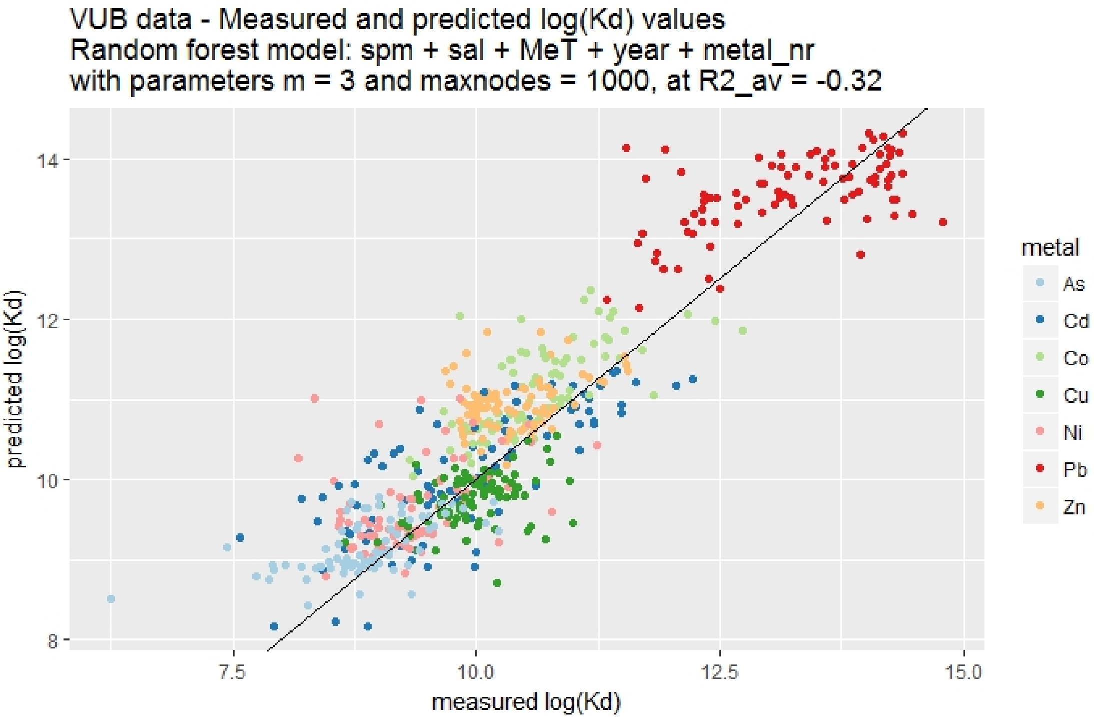


Figure 13: Plot of the measured and predicted $\log(K_d)$ values for the VUB test data.

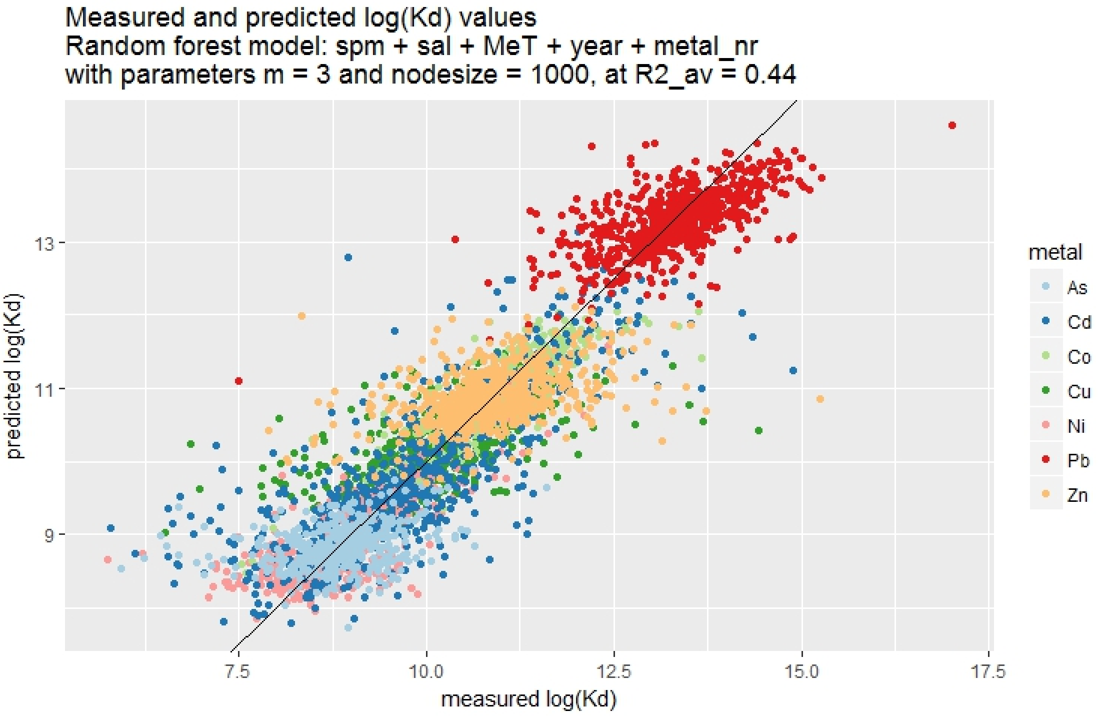


Figure 14: Plot of the measured and predicted $\log(K_d)$ values for the MWTL training data.

10.4 Gradient Boosting Machine

A gradient boosting machine model is trained to see if it performs better than a random forest model. Given the right parameter settings, gradient boosting machine models tend to perform slightly better than random forest models, as they learn more slowly [7]. Their downside compared to random forests is that they are more sensitive to the parameter settings.

As the results here are just presented in order to compare between a random forest and a gradient boosting machine model, only the results for a single model for all metals are presented. This model is thus of the form of equation 19. An overview of the R^2 scores on the training and test set for this model can be found in table 15. The R^2 scores for the training set are computed by 10-fold cross-validation.

On the training set the gradient boosting machine model performs better for all metals than the random forest model (see table 14). The average R^2 increases quite a bit: from 0.44 to 0.47.

On the test set the results are comparable for to those for the random forest model. The average R^2 score decreases from -0.32 to -0.49.

metal	train	test
Cu	0.49	-0.34
Ni	0.70	-0.38
Co	0.72	-0.40
Cd	0.64	0.55
As	0.17	0.00
Pb	0.45	0.00
Zn	0.20	-2.77
average	0.47	-0.49

Table 15: R^2 values of the predictions of the gradient boosting model for Kd, given by equation 19. A single model was trained for all metal types. The training set is the whole MWTL data base and the test set is the whole VUB data base. The results are given for parameter settings $B = 1000$, $\lambda = 3$ and $d = 49$. In the last row the weighted average of the R^2's is given.

10.5 Conclusion

On the training set a model can be constructed, that predicts quite well for most metals. Predictions for arsenic and zinc appear to be difficult. For the other metals the single random forest model has R^2 scores between 0.40 and 0.73. The PCR and PLS models have R^2 scores between 0.36 and 0.58 for these metals.

It is not immediately clear why the predictions for arsenic and zinc are worse than for the other metals. In any case it is clear that the random forest model performs better than the PCR and PLS models.

On the test set all models, except the one for cadmium, perform badly. For copper, nickel, cobalt and zinc the R^2 scores are even negative, implying that the models performs worse than simply predicting the sample average.

The gradient boosting model slightly improves over the random forest model, but is more to the parameter settings. On the training set the average R^2 increases from 0.44 to 0.47. However, the problems with the predictions on the test set remain.

To investigate what is happening for the predictions on the test set, cross-validation R^2 scores will be computed for a random forest model on the VUB data base in the next paragraph. If the results are very poor, then that could indicate that the VUB data is not compatible with a random forest model.

10.6 Random forest model trained on the VUB data base

metal	single model	individual models	mn
Cu	0.64	0.66	30
Ni	0.59	0.59	40
Co	0.68	0.70	40
Cd	0.84	0.87	50
As	0.40	0.43	40
Pb	0.50	0.54	10
Zn	0.45	0.52	55
average	0.59	0.62	

Table 16: Optimal out-of-bag R^2 values of the random forest models for Kd on the VUB data base. The model equations for the individual models and the single model respectively are given by equations 18 and 19. For the single model for all metals the parameter settings are $m = 3$ and $mn = 200$. For the individual models the optimal parameter are $m = 3$ and the mn value given in the last column. In the last row the weighted average of the R^2's is given.

Table 16 shows the R^2 scores, based on the predictions for the out-of-bag data points, of the random forest models on the VUB data base. Both the scores for the single model for all metals and the individual model for each metal can be found in the table. For the second model the scores lie between 0.43 and 0.87 with a weighted average of 0.62.

For arsenic and zinc the random forest models trained on the MWTL data base performed quite poorly, with R^2 scores of 0.09 and 0.18 respectively for the individual models. For the models trained on the VUB data base, these results are 0.43 and 0.52 respectively. Although, they are the worst performing metals on the VUB data, these scores are quite decent.

The logical conclusion to draw from these observations is that seasonal processes are especially important for the dynamics of arsenic and zinc. As the VUB data base only contains measurements from February and March, this causes only minor problems for the out-of-bag predictions. Therefore, although the R^2 scores for arsenic and zinc are the lowest on the VUB data, they are still quite reasonable.

10.7 Conclusion (continued)

The R^2 scores of the random forest models on the VUB data base, based on out-of-bag predictions, are within the same range as those of the MWTL data base, except for arsenic and zinc. These two metals performed poorly on the MWTL data base, but perform quite decently on the VUB data, although their R^2 scores are still the lowest ones. This is likely to be explained by the fact that seasonal processes are especially important for the dynamics of arsenic and zinc.

The fact that the results for all metals on the VUB data base are within a decent range, shows that it is compatible with a random forest model. This leaves the question why the model trained on MWTL data predicts so badly on the VUB data unanswered.

Two other possible causes will be examined in the next chapter. The first is the influence of the season on the K_d and the second is the influence of the location, within the Scheldt estuary.

11 Results: Models for Kd - further investigation of the decision tree based models

There is still a massive discrepancy between the cross-validation results of the model trained on the MWTL data and the results of this model on the VUB data, used as an independent test set. This indicates that the model, based on SAL, SPM, MeT and $YEAR$ misses certain factors to successfully model the K_d in the Scheldt estuary.

In this chapter we will attempt to extend the model by introducing new variables: 'season' and 'location'. We will only extend the random forest model, as it became clear in the previous chapter that the decision tree based models clearly outperform the linear regression based models.

11.1 Extension of the model by the variable 'season'

Upon inclusion of $SEASON$ as a variable in the model it has the following form:

$$\hat{K}_d = \hat{f}(SPM, SAL, MeT, YEAR, NR, SEASON), \tag{20}$$

where the variable $SEASON$ is defined as 1 for January to March, 2 for April to June, 3 for July to September and 4 for October to December. All the VUB measurements took place in the months February and March and thus all fall within the first season.

The single model for all metals is trained on the entire MWTL data base and tested on the VUB data base. The results are shown below in table 17. The R^2 scores on the training set are based on predictions for the out-of-bag data points.

metal	train	test
Cu	0.45	-0.21
Ni	0.66	-0.58
Co	0.73	-0.33
Cd	0.63	0.45
As	0.12	-0.09
Pb	0.46	0.15
Zn	0.22	-2.07
average	0.46	-0.39

Table 17: R^2 values of the single random forest model for K_d, given by equation 20. The training data is the full MWTL data base and the test data is the full VUB data base. R^2 scores are given for the model with parameter settings $m = 3$ and $mn = 1400$. In the last row the weighted average of the R^2's is given.

On the training set the R^2 scores increase slightly for all metals. On the test set this is not the case, however. The fact that the cross-validation performance on the training set increases a bit, the average R^2 increases from 0.44 to 0.46, implies that explicitly including $SEASON$ as a variable slightly improves the model.

It is quite difficult however, to draw any conclusions regarding the influence of seasonal processes on the K_d, by analysing these results. First, one should realise that the $SEASON$ variable, as it is defined here, is a very crude one. In reality seasonal processes take place on a gradual scale, which cannot be accounted for by the crude variable $SEASON$.

Also one should keep in mind that the model without $SEASON$ can already differentiate between seasons implicitly through other variables. This is clear for salinity, as we know from subsection 4.1.1 that it is correlated to the water temperature, and thus to the season.

11.2 Extension of the model by the variable 'location'

To analyse the effects of the 'location' variable on the model, the original model, i.e. the one without $SEASON$ will be extended. The extended model has the form

$$\hat{K}_d = \hat{f}(SPM, SAL, MeT, YEAR, NR, x_{up}), \tag{21}$$

where x_{up} is the upstream location. It would be quite difficult to retrieve and specify the exact upstream location of all measurements, but fortunately this does not matter much for the random forest model. Only the order of the measurement stations is important. Table 18 gives an overview of the order of the MWTL and VUB measurement stations and the x_{up} value attributed to them. An overview of the locations of the VUB and MWTL measurement stations can be found in paragraph 3.1.3 and 3.2.1 respectively. The single model for all metals is trained on the entire MWTL data base and tested on the VUB data base. The results are shown below in table 17. The R^2 scores on the training set are based on predictions for the out-of-bag data points. There is a minor increase in R^2 on the training set for most metals. The average R^2 increases from 0.44 to 0.45. On the test set there is no noteworthy change of the results either.

Just as was mentioned for the $SEASON$ variable, it is also difficult to draw conclusions regarding the importance of the location for the K_d model, based on this analysis.

stations	x_{up}
S01	1
S02	2
S03	3
S04	4
Terneuzen	5
Hoedekenskerke	6
S07	7
Hansweert	7
S09	8
S10	9
Bath	10
S12	11
Schaar van Ouden doel	11
S15	12
S22	13

Table 18: x_{up} values for the MWTL and VUB measurement stations.

metal	train	test
Cu	0.44	-0.46
Ni	0.66	-0.17
Co	0.73	-0.53
Cd	0.64	0.63
As	0.10	-0.07
Pb	0.42	0.11
Zn	0.20	-5.32
average	0.45	-0.83

Table 19: R^2 values of the single random forest model for K_d, given by equation 21. The training data is the full MWTL data base and the test data is the full VUB data base. R^2 scores are given for the model with parameter settings $m = 3$ and $mn = 1000$. In the last row the weighted average of the R^2's is given.

11.3 Conclusion

Neither the inclusion of the variable $SEASON$, nor that of x_{up}, solves the problem of the bad predictions on the VUB test data. The R^2 scores on the training set do increase slightly upon inclusion of the extra variables. From 0.44 for the original model to 0.46 and 0.45 upon inclusion of $SEASON$ and x_{up} respectively. It is difficult to draw conclusions on the importance of 'season' or 'location' for the K_d model based on these results.

That is the case, because the variables $SEASON$ and x_{up}, are defined in a very crude manner here, hence only allowing for crude differentiation. Such crude differentiation could, to a certain extent, already be achieved implicitly by the original model through differentiation in the other variables, especially salinity, which is correlated to season and x_{up}, as will be shown in the next chapter.

It is still puzzling why the random forest models have decent performance on the training set, be it MWTL or VUB, but perform poorly on the other data base, when used as test data. It seems as if there must be some intrinsic difference between the data bases that causes this bad performance. As the location of the measurement stations is the only fundamental difference between both data bases, this will be further investigated. To this end the relation between salinity and x_{up} for both databases will be compared in the following chapter.

12 The relationship between salinity and x_{up}

In general the salinity profile of the Scheldt estuary corresponds well to its longitudinal profile. The further downstream one goes, the more saline the water gets. x_{up} increases in the upstream direction and thus one would expect the salinity to decrease as x_{up} increases. Figure 15 shows the x_{up} values versus the salinity ones for both data bases.

The negative correlation between salinity and x_{up} is, as expected, clearly present in both data bases. The slope of these lines is approximately equal, which also makes sense. However, the intercepts of both regression lines differ by about one half. This implies that at the same location the average $\log(SAL)$ is $1/2$ lower for the measurements of the VUB data base compared to the MWTL data base. This difference exists, because the VUB measurements were performed exclusively in winter. A more detailed explanation of this

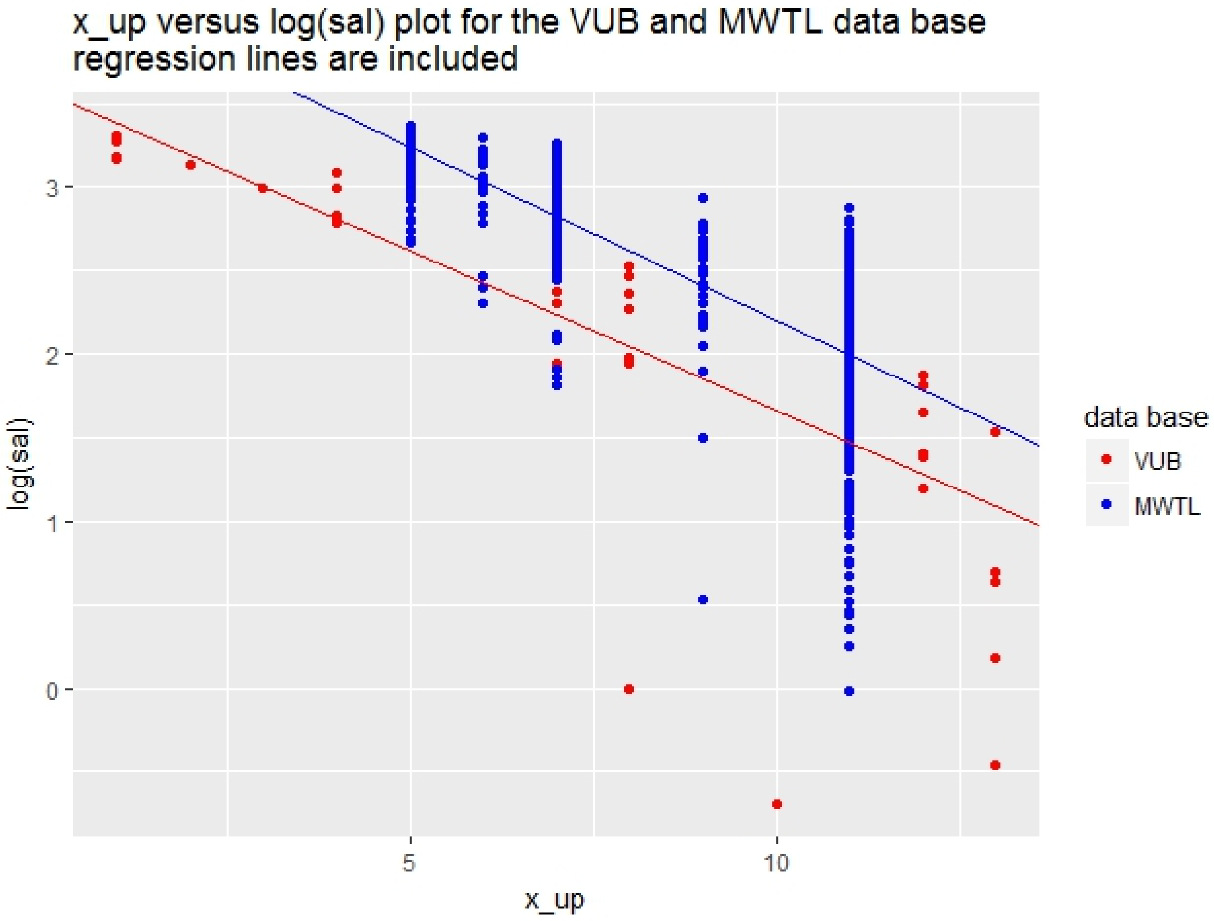


Figure 15: Salinity vs. x_{up} plot for both the VUB and the MWTL data base. The linear regression lines for the data points of both data bases are included.

phenomenon can be found in paragraph 4.1.1.

The fact that the average salinity values of both data bases differ for similar locations causes the model trained on the one data base to predict badly on the other, as will now be explained.

Together, salinity and SPM have to describe the environmental conditions in the model. Based on these environmental conditions, the total metal content and the year of measurement, the model predicts the K_d. However, salinity also implicitly describes the measurement location. Thus, assuming that the measurement location is important to accurately model the K_d, then salinity is overloaded as a variable in this model. It has to describe two things at the same time.

When cross-validating on one data set, this problem does not become clear, as measurements are performed at fixed locations. The model implicitly adjusts to these locations, based on salinity. Predicting on one of these locations is therefore not a problem. However, predicting on an independent test set with other locations and another salinity distribution, such as is the case for the VUB data base, is problematic.

Looking at figure 15 it is very clear why this is problematic. For example at a $\log(SAL)$ value of 2 the x_{up} coordinate for the MWTL data base is approximately 11, whereas that for the VUB data base is approximately 8. The model will thus predict a K_d value, thinking that it is on a completely different location.

This insight explains the discrepancy between the model results on the test and training set. The only metal for which the predictions are equally good on the test and training data is cadmium. This could be explained if the behaviour of cadmium in the Scheldt estuary is different from the other metals, and if it can be accurately described by the parameters in the model.

From literature one learns that this is indeed the case. The evidence will be presented in the following chapter.

13 The estuarine dynamics of cadmium

In the previous chapter it became clear that salinity is overloaded as a variable in the K_d model, as it has to explain both the location and the environmental conditions at the same time. The only metal, which does not suffer from this problem is cadmium.

This is the case, because cadmium has different estuarine dynamics than all other trace metals included in this research [15]. Cadmium bound to suspended matter, is desorbed when river water mixes with seawater. In literature it is suggested that this is caused by the formation of Cd-chloro-complexes, during estuarine water mixing. Thus, as the salinity of the water increases, the suspended cadmium concentration will decrease, whereas the dissolved cadmium concentration will increase. This corresponds with an increase in K_d.

These particular estuarine dynamics for cadmium can be described well by a model based on SAL, SPM and MeT, as they are more or less independent of the physical characteristics of a specific estuary. This insight explains why the predictions for cadmium on the test set are just as good as on the training set.

For none of the other trace metals considered in this study a general estuarine distribution pattern of the suspended and dissolved metal concentrations exists. In different studies removal, addition, as well as conservative dissolved metal distributions were found in different estuaries. These conflicting distributions reflect the physical and biochemical characteristics of a specific estuary [15].

In order to create a well-functioning predictive model for K_d for the other metals, the explanatory variables should therefore be able to specify these characteristics. In the next chapter two possible variables that could fulfil this purpose will be discussed.

14 Proposed variables to extend the model

The K_d models presented in this research are inadequate for all metals, except for cadmium. This is the case because the dynamics of the other metals depend highly on not only the biochemical, but also the physical characteristics of the Scheldt estuary. The variables in the model, as presented in chapter 10, do not suffice to accurately describe these characteristics. Extension of the models by the variables $SEASON$ or x_{up}, as proposed in chapter 11, does not resolve this problem either. Including other promising variables could perhaps result in an adequate model. Two variables will be proposed to add to the model: pH and the dissolved oxygen content. They could, for different reasons, help to better describe the K_d dynamics. An explanation why these two variables are promising will be given below. Also it is worth mentioning that both of these variables are available in the MWTL data base [12].

14.1 pH

First of all the pH of the water is an important environmental variable affecting the K_d [16]. Therefore, its inclusion is likely to improve the model performance.

However, there is another highly useful property of pH in the Scheldt estuary. It is namely correlated to salinity [15]. As salinity is overloaded as a variable in the model, having to explain two things at the same time, the inclusion of a correlated variable could remedy this.

14.2 Dissolved oxygen content

K_d dynamics in the Scheldt estuary are highly seasonally dependent. Introducing the crude $SEASON$ variable, as tried in chapter 11, does not suffice to capture this seasonal variability.

In the Scheldt estuary, the dissolved oxygen content is directly related to the season [15]. In summer, large parts of the estuary are fully or partially anoxic due to a combination of high biochemical activity, because of the relatively high water temperatures, and high amounts of available organic load, because of domestic, industrial and agricultural pollution. Fully anoxic zones up to a length of 30 km can be encountered in summer. The presence of such a large anoxic zone is an extraordinary feature for a well-mixed estuary, such as the Scheldt estuary. In winter the anoxic zone is much smaller, or fully absent, but still dissolved oxygen is highly under-saturated at low salinities.

Furthermore, the dissolved oxygen content is directly related the redox potential and thus to the K_d [16].

15 Conclusion

From the results in chapter 10 it became clear that it is possible to create a K_d model, that performs well on the training data, based on out-of-bag error prediction for the random forest model and cross-validation for the other ones. This is true when using both the MWTL and the VUB data as training data. Out of these models the decision tree based models outperform the linear regression based ones by a big margin. Also, as was proven in chapter 6, decision tree based models do not require transformed data, which is advantageous over linear regression based models. Comparing the two decision tree based models shows that the gradient boosting machine model performs slightly better than the random forest one.

However, predictions for a model trained on one data base and tested on the other one are very poor for all metals except for cadmium. To see whether these predictions could be improved upon by adding another parameter to the model, the variables $SEASON$ and x_{up} were introduced in chapter 11.

The inclusion of neither one of the variables increased the predictions on the test data and only very slightly on the training data. However, it is difficult to relate these results to the importance of location or seasonal variability for the K_d model, because the variables $SEASON$ and x_{up} are defined very crudely.

In chapter 12 it became clear that the discrepancy between the predictions for the training and the test set is caused by the 'overloading' of the variable salinity in the model. It functions as an indicator for both the location and the environmental conditions. The model will adapt to the measurement locations of the training data base. Therefore, the cross-validation predictions on the training set do not cause any problems. However, for a test set with different measurement locations and a different salinity profile, this problem becomes clear.

The only metal, which does not suffer from this problem, is cadmium. An investigation of the estuarine dynamics of cadmium in chapter 13 shows that this is the case, because these dynamics are mainly driven by the desorption of cadmium when river water mixes with seawater. This specific process can be described well by the variables of the original model, as it does not depend much on the physical characteristics of the estuary.

For the other metals both the physical and biochemical characteristics of the Scheldt estuary should be better represented to create a functioning model.

The pH and the dissolved oxygen content are proposed as variables to add to the model in order to better describe the above-mentioned characteristics. pH, next to affecting the K_d directly, is correlated to salinity and could thus help solve the problem of salinity being overloaded. The dissolved oxygen content is directly related to the season, but provides a gradual measure rather than the crude $SEASON$ variable that was used in chapter 11.

This brings us to the final conclusion of this research, answering the research question, which was formulated in the introduction.

Yes, it is possible to model the K_d of different metals as a function of a number of environmental variables. For cadmium a model based on SAL, SPM, MeT and $YEAR$ functions well, as the estuarine dynamics of cadmium are mainly driven by a specific process, which can be described well by the four original model variables. However, for the other metals the physical and biochemical estuarine characteristics should be better specified by the model variables. To this end the inclusion of either one or both of the variables pH and dissolved oxygen content is proposed.

Out of the statistical methods considered in this research the decision tree based ones clearly outperform the linear regression based ones. An optimally parametrised gradient boosting machine model performs slightly better than a random forest model, but its performance is more sensitive to the parameter settings.

16 Discussion

Doing research sometimes feels like performing a herculean task. Whenever a laborious question is answered by sheer heroic perseverance, three new ones are ready to take its place. As this research is no exception, the remaining heads of the hydra will be discussed. They can serve as the offset for future research.

16.1 MeT models

The MeT model for cadmium is the only one, which fails to predict decently on the test set, as was demonstrated in chapter 9. On the training set however, cross-validation predictions are fine. This discrepancy is probably related to the difference in CdT distribution for the MWTL and the VUB data base. The VUB one has a way lower median and standard deviation, as can be seen in table 6.
The major difference between the two data bases is the fact that the VUB data base only contains measurements performed in winter. Therefore, a seasonal phenomenon is likely to cause the different CdT distributions. Identifying this phenomenon will help understanding the problems in the CdT model. One thing to look into is the sources of cadmium in the Scheldt estuary, out of which metallurgy is the biggest contributor [17]. It could be that this industry is more active in summer than in winter, which would explain the difference in CdT distributions.
If some seasonal phenomenon is important for the CdT model, then it makes sense to add the dissolved oxygen content as a variable to explain this.

Either way it would be interesting for all MeT models to investigate whether extending the models by pH and dissolved oxygen content would improve their performance. This should be combined with a check of the variance importance, to see which variables are most important for the model. Probably dissolved oxygen content will be less important as an explanatory variable than pH, as the seasonal influence is not of big importance for most metals, except perhaps for cadmium.

One thing that has been omitted is to check the regression assumptions for the MeT models based on PCR and PLS. It is however likely that the assumptions that were met for the K_d models based on PCR and PLS, are also met for the MeT models. This makes sense, as the MeT models use a part of the variables of the K_d models, with the same distributions.

16.2 K_d models

For arsenic and zinc it became apparent that seasonal processes have an essential influence on their dynamics within the Scheldt estuary. Literature should be checked on this.
Just as for the MeT models it would also be interesting to look into the variable importance, of the K_d models extended with pH and dissolved oxygen content. This can provide deeper insight into the processes driving the estuarine dynamics of the metals. For arsenic and zinc, for example, one would expect a relatively high importance for the dissolved oxygen content, as this variable is directly related to seasonal variation.

Another thing that should be investigated more deeply is the prediction domain of the model. The MWTL data base, for instance, only contains measurements performed at stations in the Western Scheldt. For the current model, based on SAL, SPM, MeT and $YEAR$, it is clear that predictions outside of the measurement domain will not be accurate. However, after extension of the model with pH and dissolved oxygen content, it could be that a model trained on the MWTL data will also predict reasonably well in the part of the Scheldt estuary that lies outside of the measurement domain of the MWTL data.

Last, there remains a mathematical question, regarding the random forest models, that include the metal number. Partitioning of the explanatory data space will depend to a big extend on the metal number and most data points of a certain metal type will be grouped together. To understand the question that will be raised here, try to imagine the data space being divided into seven distinct boxes, one for each metal type. Although most data points of a certain metal type are contained in the box corresponding to its own metal type, there might be some data points that fall into the box of another metal type. For the prediction of these data points the K_d distribution of the neighbouring metal type is clearly important. It thus makes sense to order the boxes in such a way that the metal types with a similar K_d distribution neighbour one another.
The order of the boxes, corresponding to the metal types, depends on the definition of the metal number. If, for instance, the assignment of the numbers to the metals were to be done alphabetically, then cobalt would be in between cadmium and copper. Because of the aforementioned argument it would make more sense to order the metals based on their K_d distribution. One could for instance assign the metal number in such a way that it corresponds to the median of the K_d distribution.
The question if and how the ordering of the metals influences the model performance remains to be investigated.

References

[1] J.H. Duffus *Heavy Metal-A Meaningless Term?* Pure and Applied Chemistry, Vol. 74, No. 5, pp. 793-807, 2002.

[2] M. Elskens e.a. *Modeling metal speciation in the Scheldt Estuary: Combining a flexible-resolution transport model with empirical functions* Science of the Total Environment, Vol. 476-477, pp. 346-358, April 2014.

[3] *Het Schelde-estuarium,*
`https://www.vnsc.eu/de-schelde/het-schelde-estuarium/`, 28 August 2018

[4] J.D. Allison and T.L. Allison. *Partition Coefficient for Metals in Surface Waters, Soil and Waste* July 2005.

[5] G. James, D. Witten, T. Hastie, R. Tibshirani. *An Introduction to Statistical Learning.* First edition, 2013. ISBN: 978-1-4614-7137-0.

[6] T. Kvålseth. *Cautionary Note about R^2.* The American Statistician, Vol. 39, No. 4, pp. 279-285, November 1985.

[7] A. C. Müller and S. Guido. *An Introduction to Machine Learning with Python.* First edition, 2016. ISBN: 978-1-4493-6941-5.

[8] L.G. Blackwood. *The lognormal distribution, environmental data and radiological monitoring.* Environmental Monitoring and Assessment, Vol. 21, No. 3, pp. 193-210, June 1992.

[9] M. Williams and C. A. Gomez Grajales. *Assumptions of Multiple Regression: Correction Two Misconceptions.* Practical Assesment, Research and Evaluation, Vol. 18, No. 11, September 2013.

[10] N. E. Savin and K. J. White. *The Durbin-Watson test for Serial Correlation with Extreme Sample Sizes or Many Regressors.* Econometrica, Vol. 45, No. 8, pp. 1989-1996, November 1977.

[11] J. Kim. *How to Choose the Level of Significance: a Pedagogical Note.* Munich Personal RePEc Archive, Paper No. 66373, August 2015.

[12] *MWTL chemisch monitoring netwerk Westerschelde [MWTL chemical monitoring network Westerschelde],*
`http://www.scheldemonitor.be/nl/imis?module=dataset&dasid=135`, 17 August 2018

[13] *Verdronken Land van Saeftinghe*, photo by Bert Maetens, `https://bkites.wordpress.com/2013/12/26/verdronken-land-van-saeftinghe/`, August 2018

[14] *Stations in the Western Scheldt estuary area*, `http://www.mumm.ac.be/datacentre/Documentation/viewstations.php?map=stations_estuary`, August 2018

[15] J.J.G. Zwolsman, B.T.M. Van Eck, C.H. Van der Weijden. *Geochemistry of dissolved trace metals (cadmium, copper, zinc) in the Scheldt estuary, southwestern Netherlands: Impact of seasonal variability.* Geochimica et Cosmochimica Acta, Vol. 61, No. 8, pp 1635 - 1652, January 1997

[16] F. F. Pérez, et al. *Adsorption and desorption kinetics of* ^{60}Co *and* ^{137}Cs *in fresh water rivers.* Journal of Environmental Radioactivity, Vol. 149, pp 81-89, November 2015

[17] M. Elskens. *Measuring and modeling trace metals in aquatic systems (lecture notes).* 2018

[18] M. Fettweis, M. Sas and J. Monbaliu. *Seasonal, Neap-spring and Tidal Variation of Cohesive Sediment Concentration in the Scheldt Estuary, Belgium.* Estuarine, Coastal and Shelf Science, Vol. 47, pp 21-36, 1998

[19] L. Breiman. *Random Forests.* Machine learning, Vol. 45, pp 5 - 32, 2001

[20] B.H. Mevik, R. Wehrens, K. H. Liland. *R package 'pls'.* December 2016

[21] L. Breiman, A. Cutler, A. Liaw and M. Wiener *R package 'randomForest'.* March 2018

[22] G. Ridgeway *Package 'gbm'.* March 2017

Appendix A Principal component analysis

Principal component analysis (PCA) is an explanatory statistical method, which can be used to achieve several ends. It can be used to visualise data in a lower dimension than the number of variables, which allows one to detect patterns in the data and to identify correlations between the considered variables. Another use of PCA is the derivation of variables that can be used in supervised learning methods [5]. This is discussed in the subsection on principal components regression, 5.2.1.

Given variables $X_1, ..., X_J$, that span $\mathbb{R}^J$, PCA produces new variables $Z_1, ..., Z_J$, which are linear combinations of $X_1, ..., X_J$ and also span $\mathbb{R}^J$. The new variables are called principal components. They have the following form:

$$Z_i = \sum_{j=1}^{J} \phi_{ij} X_j. \tag{22}$$

The coefficients ϕ_{ij} are called the loadings of the principal components. They indicate the relative importance of the different parameters. The principal components are constructed hierarchically, which means that first Z_1 is constructed and then Z_2, given certain restrictions implied by Z_1, etc. The principal components are constructed in such a way that the first one covers most of the variability contained in the variables, the second one contains covers most of the variability in the data, that is not yet captured by the first principle component, etc.

Let $x_{j1}, ..., x_{jn}$ the elements of X_j. The corresponding elements of the PCs are called their scores and are denoted as

$$z_{ij} = \sum_{k=1}^{J} \phi_{ik} x_{jk}. \tag{23}$$

The ϕ_{ij}'s are chosen such that they maximise

$$\max_{\phi_{ij}, \ldots, \phi_{iJ}} \{\frac{1}{n} \sum_{i=1}^{n} z_{ij}^2\} \quad \text{for } 1 \leq i \leq J. \tag{24}$$

Two restrictions on the ϕ_{ij}'s are necessary: normalisation and orthogonality with respect to one another. The first one prevents them from getting infinitely big and the second one assures that the PCs are uncorrelated. For the construction of Z_i these restrictions can be formally written as

$$\sum_{j=1}^{J} \phi_{ij}^2 = 1 \text{ and } \sum_{k=1}^{J} \phi_{ik} \phi_{lk} = 0, \quad 1 \leq l < i. \tag{25}$$

Transformation of data before PCA

In practice one, virtually always, transforms the data before performing PCA such that each variable is mean-centred and has standard deviation one. The motivation behind this will be explained in the following.

Note that if the data is mean-centred, i.e. $\frac{1}{n}\sum_{i=1}^{n} x_{ij} = 0$, then the scores are mean-centred as well. Thus, in equation 24 the sample variance of $z_{1j}, ..., z_{nj}$ is maximised. To see this, realise that $\sum_{j=1}^{n}(z_{ij} - \bar{z}_{ij})^2 = \sum_{j=1}^{n} z_{ij}^2 - 2z_{ij}\bar{z}_{ij} + \bar{z}_{ij}^2 = \sum_{j=1}^{n} z_{ij}^2$, as $\bar{z}_{ij} = \frac{1}{n}\sum_{i=1}^{n} z_{ij} = 0$.

Furthermore, the scaling of the data implies that the results of a PCA remain the same for variables of different orders of magnitude. If this is not done, then the variable with the biggest order of magnitude will also tend to have the biggest variance. Thus, this variable would be strongly represented in the first PC, although the variability of this variable is not necessarily biggest.

Proportion of variance explained

The proportion of variance explained (PVE) is used as a measure to determine how much of the total variance in a data base is contained in a single principal component. The PVE of the m-th principle component is defined in the following way

$$\text{PVE}_m = \frac{\text{Var}(Z_m)}{\sum_{j=1}^{J}\text{Var}(Z_j)}, \quad \text{where } \text{Var}(Z_j) = \frac{1}{n}\sum_{i=1}^{n} z_{ij}^2. \tag{26}$$

$\text{PVE}_1 \geq ... \geq \text{PVE}_J$ is valid, due to the hierarchical construction of the principal components.

Uniqueness of the principal components

The loading vectors of the principal components are unique up to their sign. This is the case, because the results of a PCA only depend on the direction of the loading vectors, not on its sign. Similarly, the scores of the principal components are also unique up to their sign.

Appendix B Partial least squares components

This part of the appendix describes the construction of the PLS components. We consider the case where one has a data base with response variable Y and explanatory variables $X_1, ..., X_J$.

Just as for the construction of the PCs, the construction of the PLS components is a hierarchical process. The PLS components that are computed will be denoted as $\zeta_1, ..., \zeta_J$ and are of the form

$$\zeta_i = \sum_{j=1}^{J} \phi_{ij} X_j, \tag{27}$$

where ϕ_{ij} are the loadings of the i-th PLS component.

Construction of the first PLS component

The first PLS component is constructed by performing a simple linear regression of Y onto $X_1, ..., X_J$. The coefficients of this simple linear regression model are set to be the coefficients of the first PLS component. Let

$$\hat{Y} = \sum_{j=1}^{J} \beta_{1j} X_j + \alpha_1 \tag{28}$$

the simple linear regression model of Y onto $X_1, ..., X_J$.

Now $\phi_{11}, ..., \phi_{1J} = \beta_{11}, ..., \beta_{1J}$.

Construction of the other PLS components

The explanatory variables are now corrected by regressing ζ_1 onto each X_j and taking the residuals. Let the regression model of ζ_1 onto each X_j be given by

$$\hat{\zeta}_{1j} = \beta X_j + \alpha, \tag{29}$$

then the vector of residuals of X_j is given by

$$e_{1j} = \zeta_1 - \hat{\zeta}_{1j}. \tag{30}$$

Then Y is regressed onto $e_{11}, ..., e_{1J}$ and the regression coefficients are set to be the loadings of the second PLS component. Let

$$\hat{Y} = \sum_{j=1}^{J} \beta_{2j} e_{1j} \tag{31}$$

the simple linear regression model of Y onto $e_{11}, ..., e_{1J}$.

Now $\phi_{21}, ..., \phi_{2J} = \beta_{21}, ..., \beta_{2J}$.

This process is repeated $J - 2$ times to obtain all J PLS components. By constructing the components in such a way, orthogonality is assured. Standardisation of the data is not explicitly required for the computation of PLS components, in contrast to principal components.

Appendix C Decision trees

A decision tree is a supervised statistical learning method, which can be used to make predictions about a response variable Y, based on a number of explanatory variables $X_1, ..., X_J$.

The principle behind a decision tree is a simple, but elegant one. The way it is constructed can formulated in two steps [5].

1. Several cuts are made in the explanatory space to divide it into a number of distinct, non-overlapping, rectangular boxes. If N unique cuts are made, then the data space is partitioned into the boxes $R_1, R_2, ..., R_{N+1}$.

2. For every data point in box R_i the same prediction is made, namely the mean of the values of the response variable corresponding to the data points in that box.

The partitioning in step one is made in such a way that it minimises the residual sum of squares of the response variable over all the boxes.

Finally the decision tree model will have the following form:

$$\hat{f}(X) = \sum_{n=1}^{N+1} \hat{y}_{R_n} \cdot \mathbb{1}_{(X \in R_n)}, \tag{32}$$

where $\hat{y}_{R_n}$ is the average of the y-values corresponding to the data points in box R_n.

C.1 The greedy algorithm

In practice the number of cuts N is smaller than the number of data points m, as otherwise all of the data points would be assigned their own box. This would result in a perfect prediction on the training data, but would reduce the predictive ability of the tree.

Theoretically, it is possible to find the optimal partitioning for a fixed number of cuts $N < m$. However, as the computational costs of finding this partitioning are too high, a different kind of algorithm is used in practice. This algorithm is known as the greedy algorithm. The name greedy refers to the fact that, every time that a cut is made in the explanatory space, it chooses the one that is optimal at that moment, rather than looking for one which would result in a better partitioning in the end.

To understand the algorithm the concept of an optimal cut has to be formalised.

Definition C.1 (Temporary partitioning of the explanatory space)**.** *Before the n'th cut the explanatory space is divided into the temporary partitioning $R_1^n, ..., R_n^n$. With $R_1^1 = \Omega$, the entire explanatory data space.*

Definition C.2 (Cut)**.** *A cut is defined by the triplet (j, s, R_k^n). This means that a cut, given by the line $X_j = s$, is made in box R_k^n. The cut is such that box R_k^n is split into the two half-boxes $R_{k,1}^n$ and $R_{k,2}^n$. Formally,*

$R_{k,1}^n(j,s) = \{X|X_j < s \text{ and } X \in R_k^n\}$

$R_{k,2}^n(j,s) = \{X|X_j \geq s \text{ and } X \in R_k^n\}$.

Definition C.3 (Residual sum of squares)**.** *The residual sum of squares of box R_k^n is defined by*

$$RSS(R_k^n) = \sum_{i:x_i \in R_k^n} (y_i - \hat{y}_{R_k^n})^2. \tag{33}$$

Definition C.4 (Optimal cut)**.** *An optimal cut is a cut (j, s, R_k^n) that minimises*

$$RSS(R_{k,1}^n(j,s)) + RSS(R_{k,2}^n(j,s)). \tag{34}$$

So, the greedy algorithm can be summarised as follows:

1. At step n of the algorithm the explanatory space is divided into the temporary partitioning $R_1^n, ..., R_n^n$ ($R_1^1 = \Omega$). This temporary partitioning is created by $n-1$ previous optimal cuts of the explanatory space.

2. An optimal cut is made through the temporary partitioning for each step $n \leq N$.

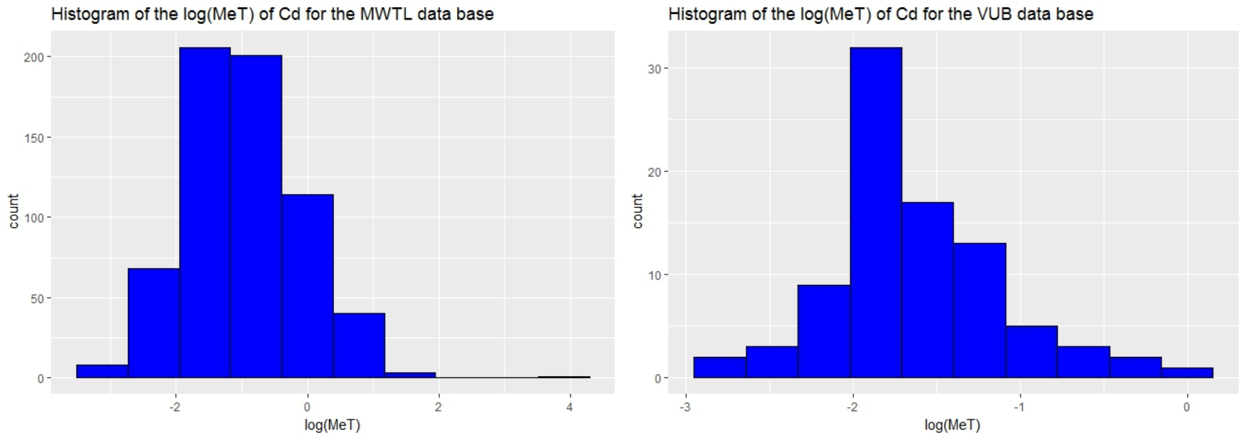
Appendix D Bootstrap sampling

Bootstrap sampling is a technique to create new samples from a given data base, which are similar to it. The idea is that these new samples have more or less the same properties as the initial data base [5]. Assume one has a data base D of dimension $(m \times n)$. Let $d_1, ..., d_n$ the columns of the data base. Then a new data base is created by taking a permutation with replacement of each d_i. This means that the elements of d_i can be used multiple times.
For the hypothetical data base A $= \left(\begin{smallmatrix} a & b \\ c & d \\ e & f \end{smallmatrix}\right)$, A’ $= \left(\begin{smallmatrix} c & f \\ c & d \\ a & b \end{smallmatrix}\right)$ and A” $= \left(\begin{smallmatrix} e & d \\ c & f \\ e & f \end{smallmatrix}\right)$ are possible bootstrap samples.

Appendix E Distribution of the data

E.1 *MeT*

E.1.1 Cadmium

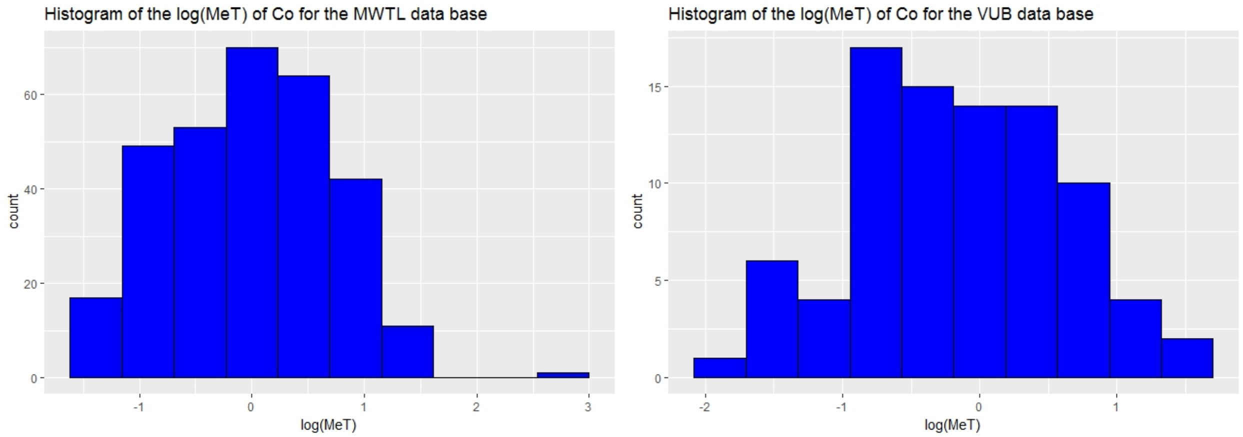


(a)

(b)

Figure 16: The left panel and right panels show the distributions of the log(MeT) values of cadmium in the MWTL and the VUB data bases respectively.

E.1.2 Cobalt

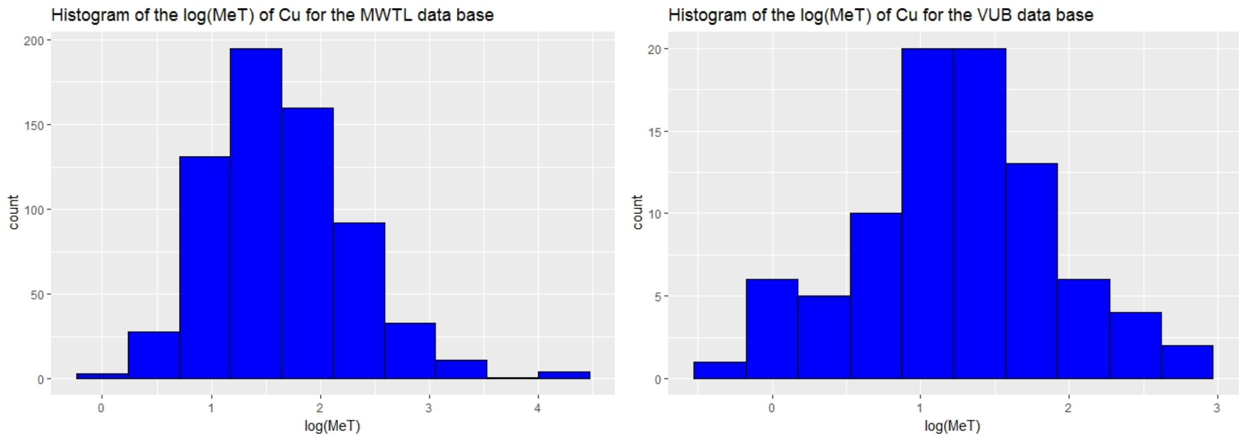


(a)

(b)

Figure 17: The left panel and right panels show the distributions of the log(MeT) values of cobalt in the MWTL and the VUB data bases respectively.

E.1.3 Copper

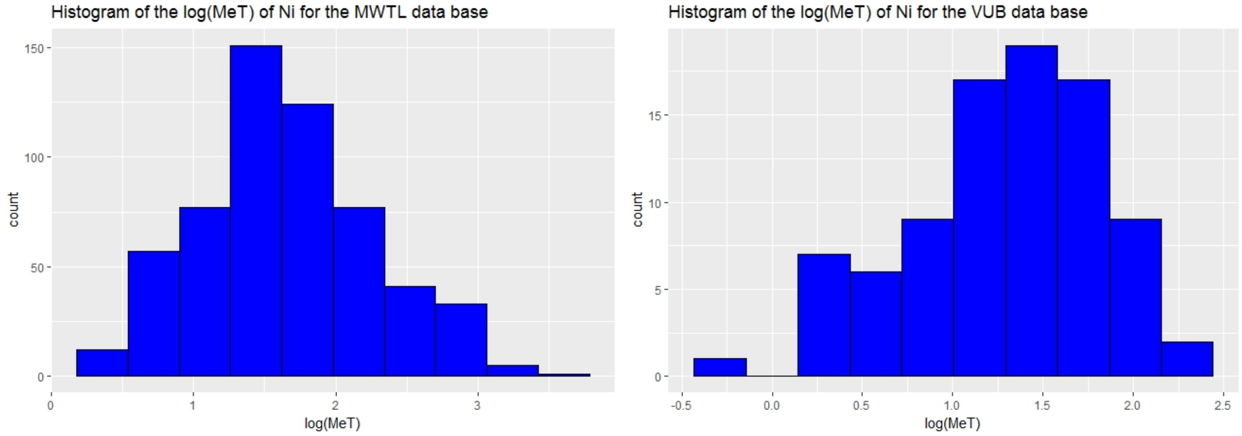


(a)

(b)

Figure 18: The left panel and right panels show the distributions of the log(MeT) values of copper in the MWTL and the VUB data bases respectively.

E.1.4 Nickel

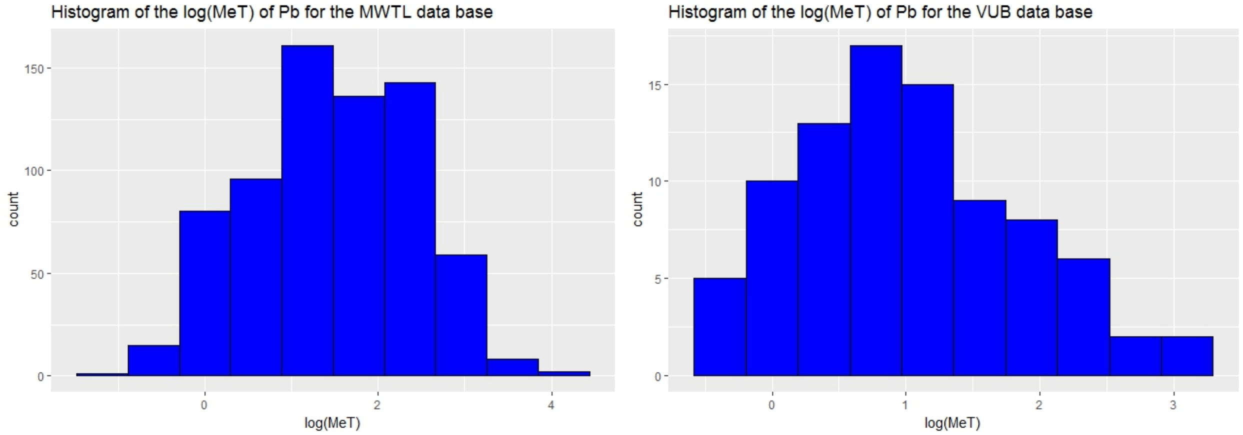


(a)

(b)

Figure 19: The left panel and right panels show the distributions of the log(MeT) values of nickel in the MWTL and the VUB data bases respectively.

E.1.5 Lead

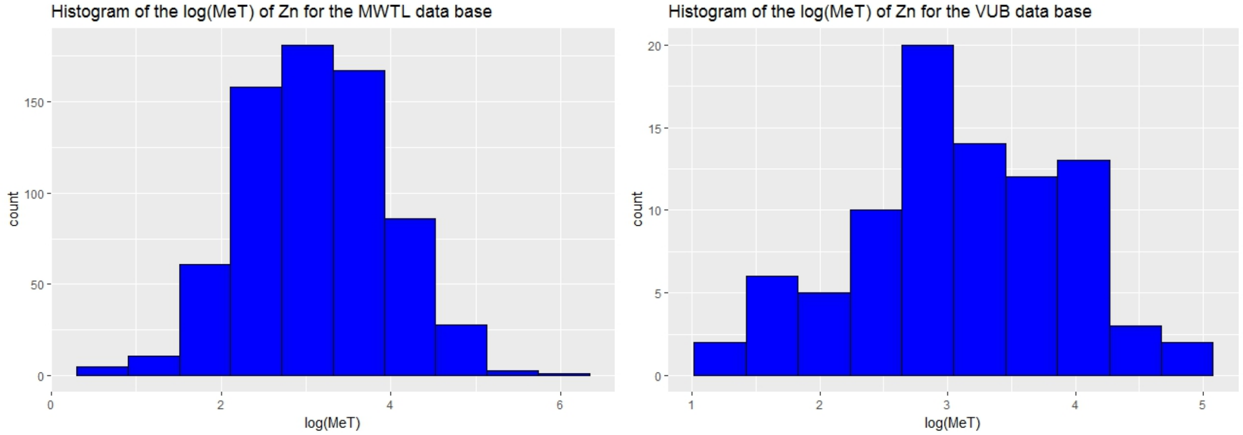


(a)

(b)

Figure 20: The left panel and right panels show the distributions of the log(MeT) values of lead in the MWTL and the VUB data bases respectively.

E.1.6 Zinc



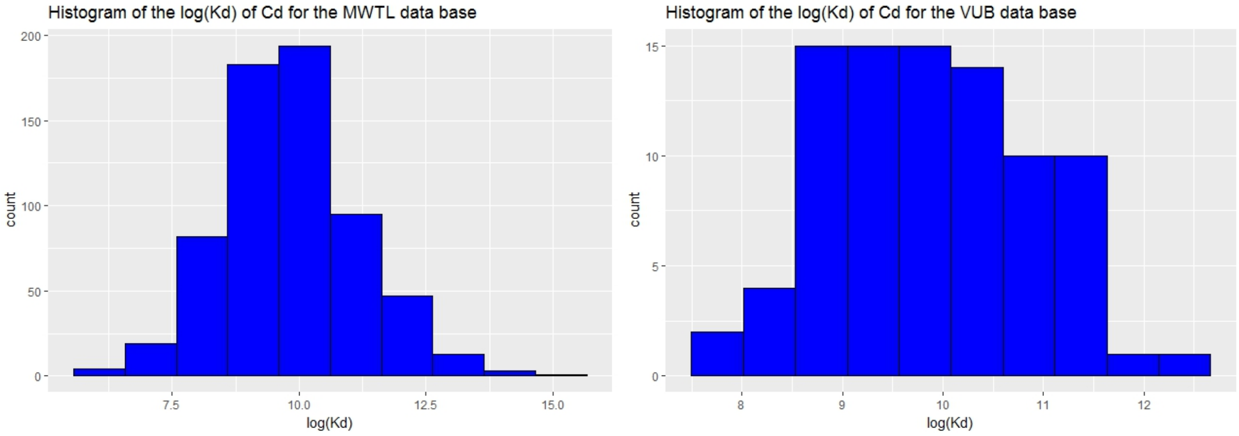
(a)

(b)

Figure 21: The left panel and right panels show the distributions of the log(MeT) values of zinc in the MWTL and the VUB data bases respectively.

E.2 K_d

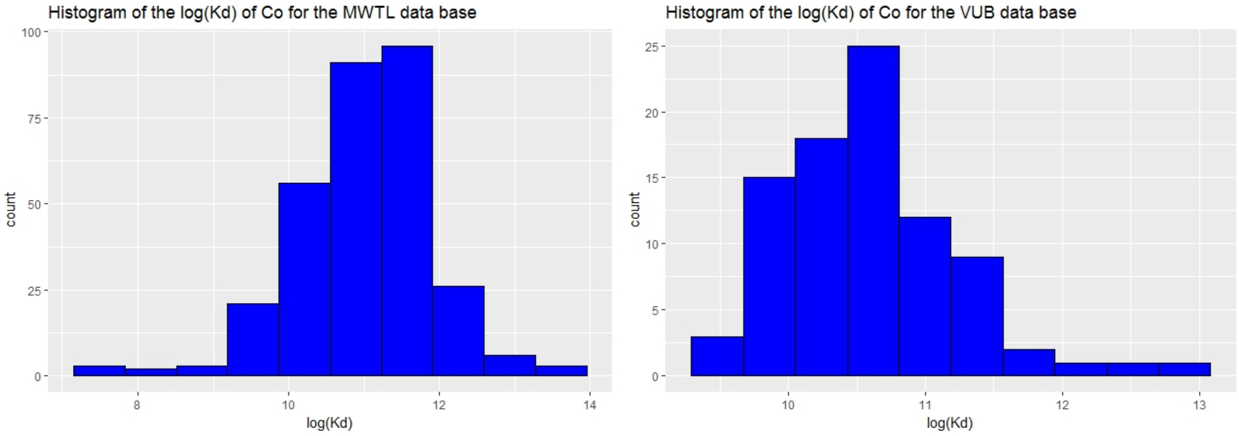
E.2.1 Cadmium



(a) (b)

Figure 22: The left panel and right panels show the distributions of the $\log(K_d)$ values of cadmium in the MWTL and the VUB data bases respectively.

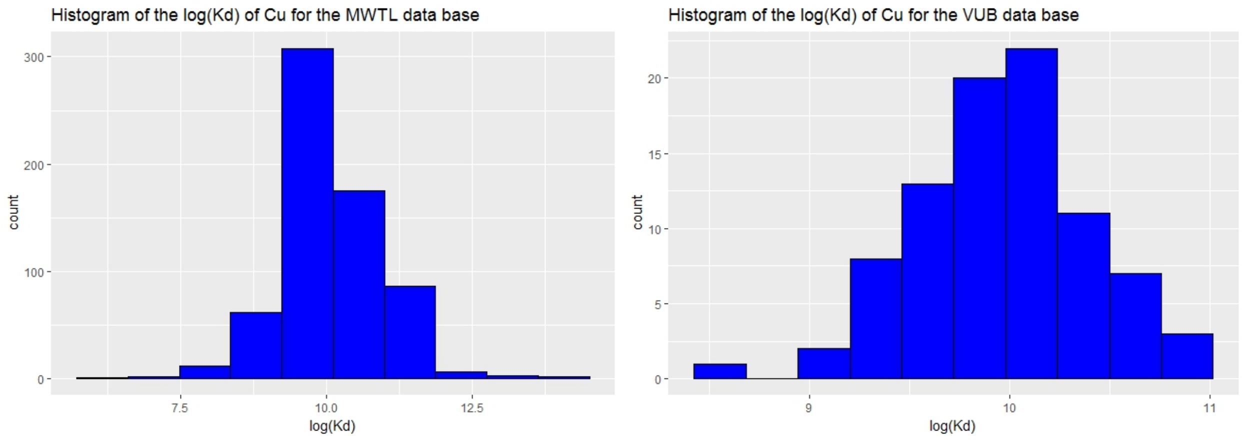
E.2.2 Cobalt



(a) (b)

Figure 23: The left panel and right panels show the distributions of the $\log(K_d)$ values of cobalt in the MWTL and the VUB data bases respectively.

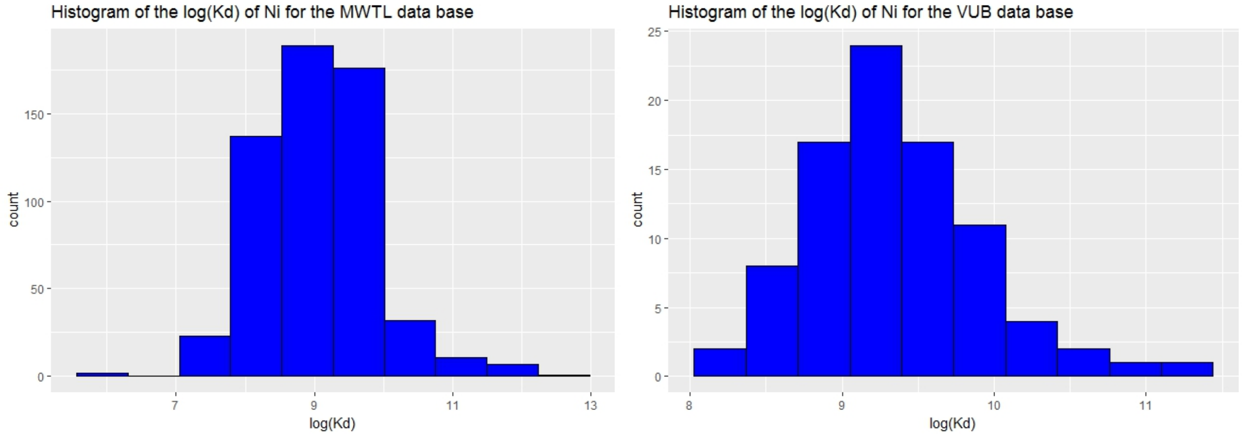
E.2.3 Copper



(a) (b)

Figure 24: The left panel and right panels show the distributions of the $\log(K_d)$ values of copper in the MWTL and the VUB data bases respectively.

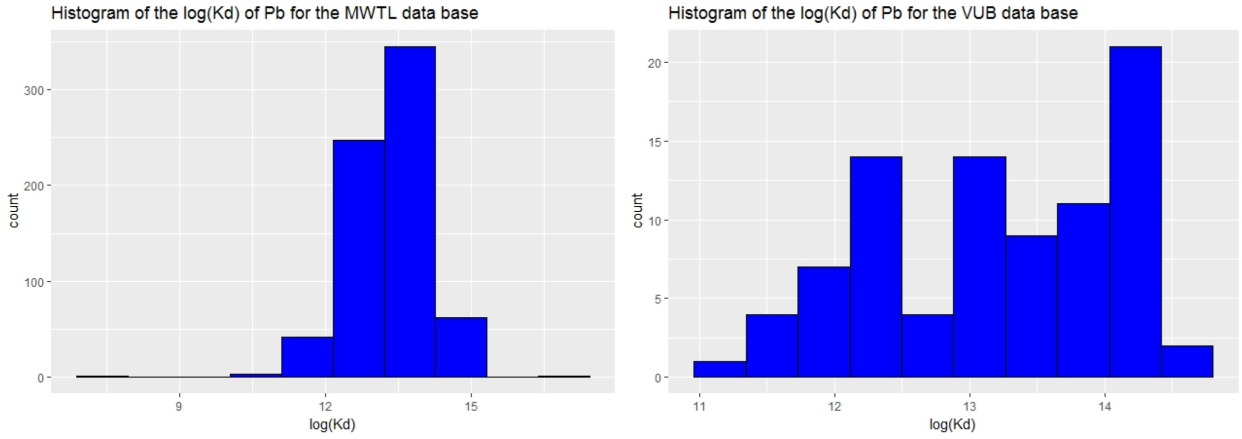
E.2.4 Nickel



(a) (b)

Figure 25: The left panel and right panels show the distributions of the $\log(K_d)$ values of nickel in the MWTL and the VUB data bases respectively.

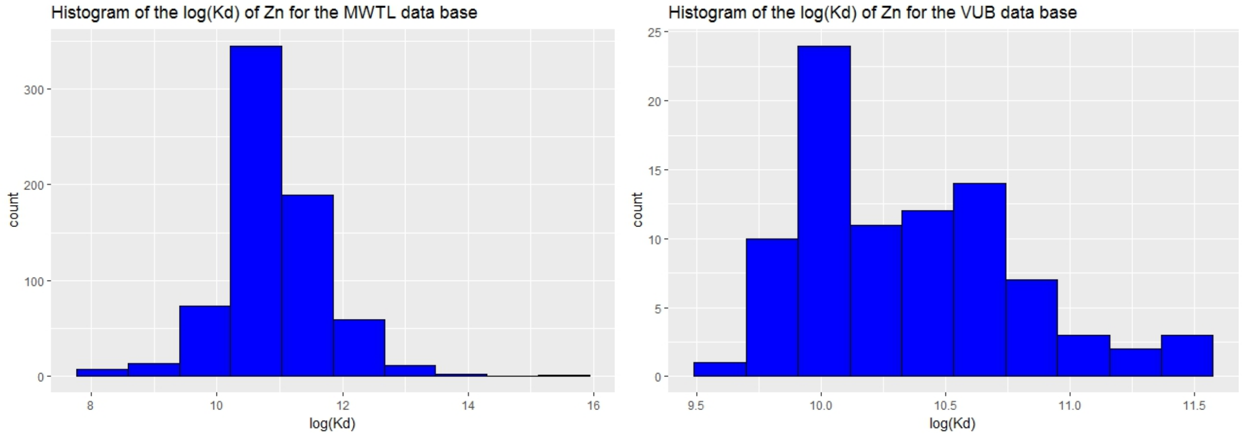
E.2.5 Lead



(a) (b)

Figure 26: The left panel and right panels show the distributions of the $\log(K_d)$ values of lead in the MWTL and the VUB data bases respectively.

E.2.6 Zinc



(a) (b)

Figure 27: The left panel and right panels show the distributions of the $\log(K_d)$ values of zinc in the MWTL and the VUB data bases respectively.

Appendix F Multiple linear regression assumptions

The theory of multiple linear regression relies on a number of underlying assumptions. To assure properly functioning models based on it, these assumptions should be met. In this section the assumptions will be introduced and subsequently the standard ways to test them in practice will be discussed [9].

F.1 Assumptions

F.1.1 (1) Linearity in the parameters

The model relating the response variable Y to the explanatory variables $X_1, ..., X_J$ should be linear in the regression parameters, $\beta_1, ..., \beta_J$. This means that Y is assumed to be a linear function of $\beta_1, ..., \beta_J$. Note that this does not mean that Y is assumed to be a linear function of $X_1, ..., X_J$. For example the function

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon, \tag{35}$$

where ϵ is the error term, is linear in the regression parameters, but not in the explanatory variable and thus meets this assumption.

F.1.2 (2) Zero conditional mean of errors

The errors are assumed to have a mean of zero, given the explanatory variables. Possible reasons for a violation of this assumption could be a wrongly chosen model relation, e.g. a linear relation between the response and explanatory variables is chosen, but in reality there is a quadratic relation, or a correlated measurement error.

F.1.3 (3) Independence of errors

The errors are supposed to be independent. Violation of this assumption leads to biased estimates of the standard error and significance of the regression parameters.

F.1.4 (4) Homoscedasticity of errors

The errors are assumed to have a constant and finite variance, across all levels of the explanatory variables. This property is called homoscedasticity.

F.1.5 (5) Normality of errors

The errors are assumed to follow a normal distribution. If the assumption is met, then the regression parameters will also follow a normal distribution and thus inference about them, based on methods requiring a normal distribution, will be trustworthy.

For bigger samples this assumption is not that important, as the distribution of the regression parameters will approach normality as the sample size increases.

F.2 How to check the assumptions

F.2.1 (1) Linearity in the parameters

Nothing has to be checked here. All that is required, is that the model relationship between the response and the explanatory variables is linear in the regression parameters, such as in equation 35.

F.2.2 (2) Zero conditional mean of errors

To check this assumption one checks if the mean of the sample residuals is significantly different from zero or not.

F.2.3 (3) Independence of errors

Visually this assumption can be checked by plotting the values of the residuals against the number of the corresponding data point. If any specific patterns are observed, this can indicate that the errors are not independent.

A quantitative test exists in the form of the Durbin-Watson statistic

$$d = \frac{\sum_{i=2}^{n}(e_i - e_{i-1})^2}{\sum_{i=1}^{n} e_i^2}. \tag{36}$$

This statistic has values between 0 and 4. A value around 2 indicates that the errors are independent, whereas values towards 0 or 4 indicate the contrary [10].

Hypothesis testing if d differs significantly from 2 is done by comparing d with critical values from a table. Consider the case that d is smaller than 2. Then, given the sample size and the number of explanatory variables of the regression model, one can find critical upper and lower values dU and dL respectively from a table. In this research the table from Savin and White [10] is used, which gives critical bounds for a significance level of 1%. If d is smaller than dL, the null hypothesis is rejected and thus the errors are not independent. If d is bigger than dU, there is no significant evidence to reject the null hypothesis and thus one infers that the errors are independent. For d between dL and dU the test is inconclusive.

In this research we will stick to a visual verification, rather than performing the Durbin-Watson test.

F.2.4 (4) Homoscedasticity of errors

Several quantitative test exist to check whether the regression errors have equal variance, which all come with their advantages and disadvantages.

For this research a visual verification will be relied upon, rather than a test statistic. The predicted values of the response variable are plotted against the studentised residuals to detect if there are any clear patterns. If this is not the case, then homoscedasticity will be concluded [9].

F.2.5 (5) Normality of errors

To check if the errors are normally distributed both a histogram and a QQ-normality plot of the residuals will be presented.

Appendix G Results K_d models - PCR

In this part of the appendix the residuals, their distributions and plots of the studentised residuals versus predicted $\log(K_d)$ values are presented for the PCR models with three principal components for each of the metals. This is done to check the linear regression assumptions discussed in appendix F. Only the errors for the models with three components are shown, as these models perform best for most metals.

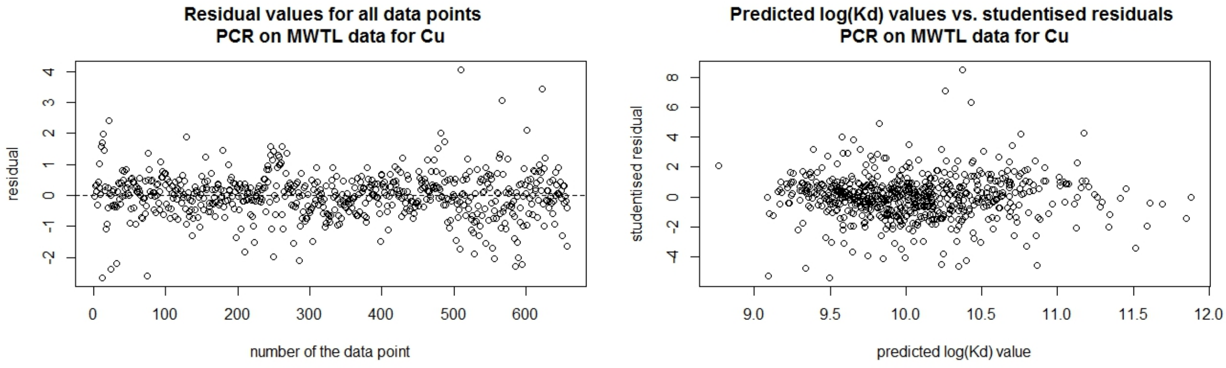
Both for nickel and cadmium there is a spike of the residual values around data point 250, which implies that the errors are not independent for the models for these metals. For the other metals no patterns are detected in the errors, thus implying that they are independent.

The violation of the assumption of independence for nickel and cadmium can cause biased estimates of the standard error and the significance of the regression parameters. In the content of this research, this is not that important.

The other two assumptions: homoscedasticity and normality of the errors, are met for all metals.

Furthermore, the regression coefficients are presented for each of the metals for the models with 3 and 4 principal components. For cadmium they are also given for the model based on a single principal component.

G.1 Copper



(a) (b)

Figure 28: The left panel shows the values of the residuals of the PCR model trained on the MWTL data for copper for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

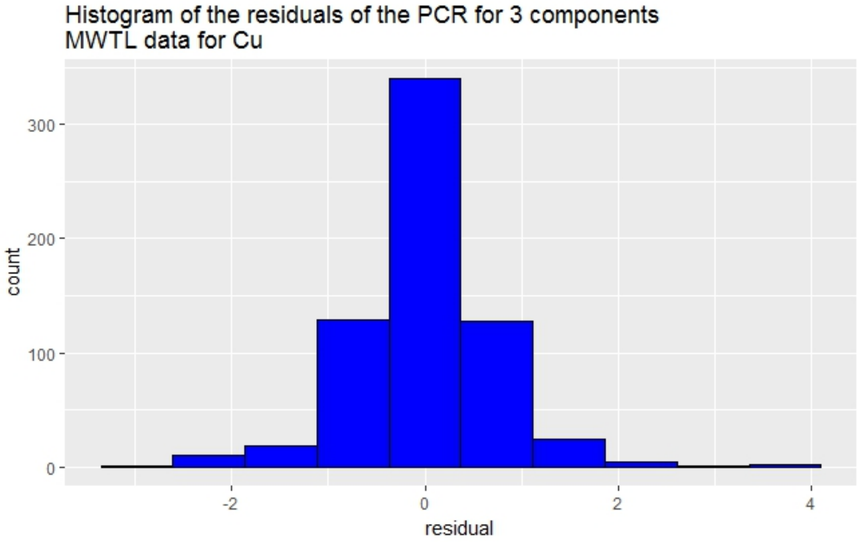
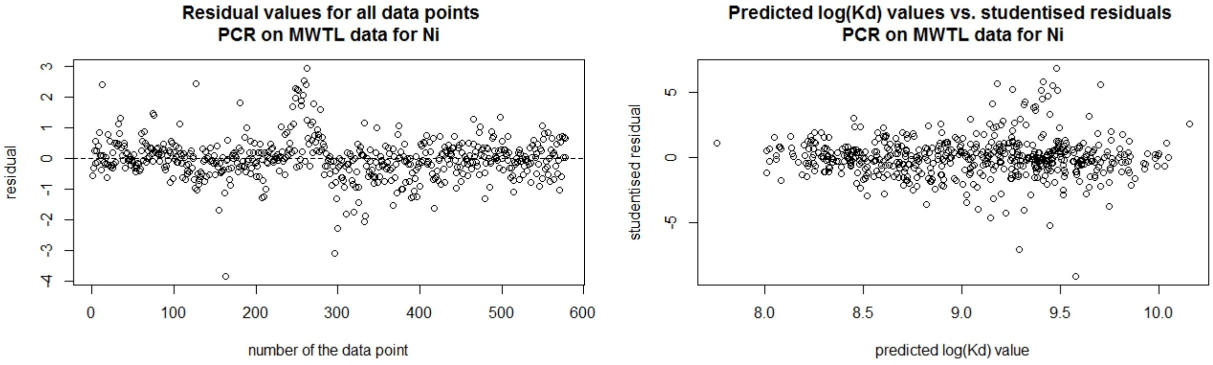


Figure 29: Histogram of the residuals of the PCR model with three principal components for the MWTL data for copper.

G.2 Nickel



(a)

(b)

Figure 30: The left panel shows the values of the residuals of the PCR model trained on the MWTL data for nickel for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

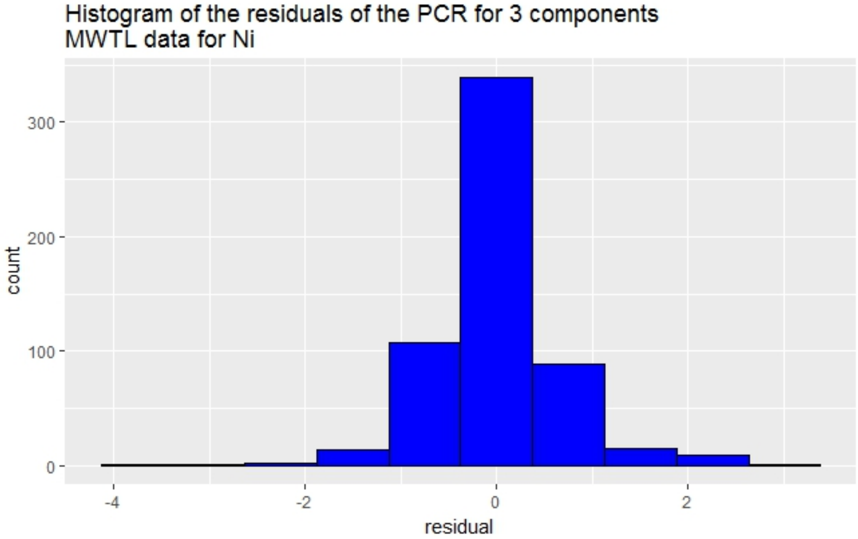
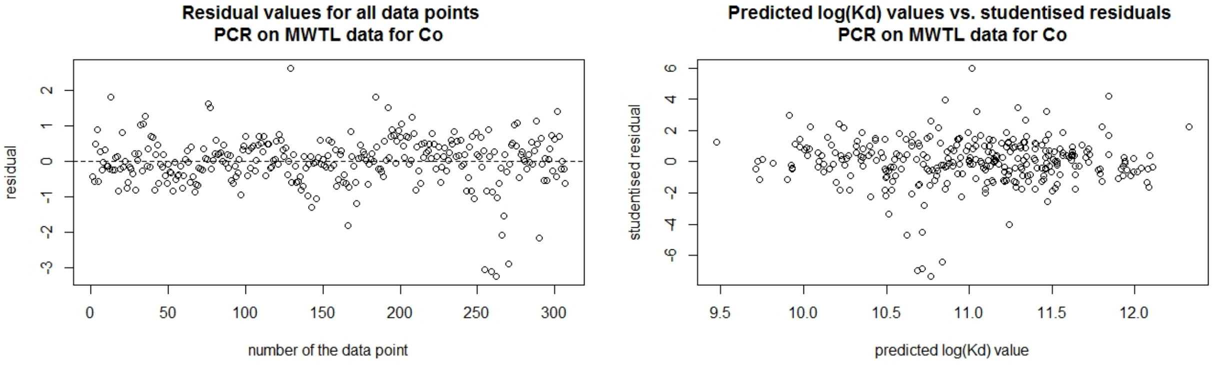


Figure 31: Histogram of the residuals of the PCR model with three principal components for the MWTL data for nickel.

G.3 Cobalt



(a) (b)

Figure 32: The left panel shows the values of the residuals of the PCR model trained on the MWTL data for cobalt for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

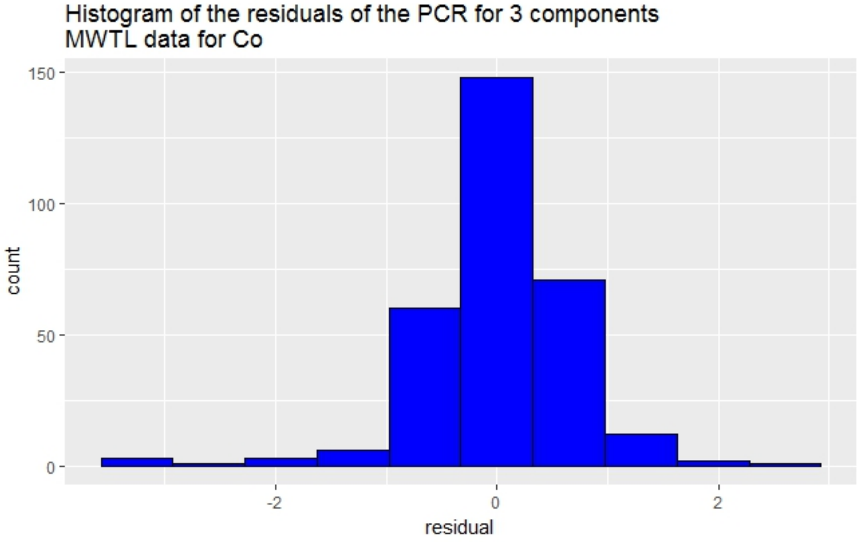
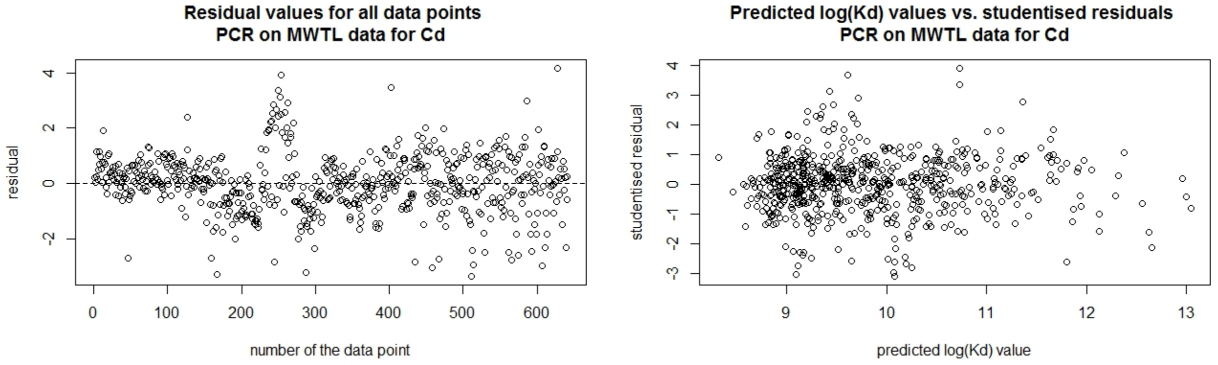


Figure 33: Histogram of the residuals of the PCR model with three principal components for the MWTL data for cobalt.

G.4 Cadmium



(a)

(b)

Figure 34: The left panel shows the values of the residuals of the PCR model trained on the MWTL data for cadmium for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

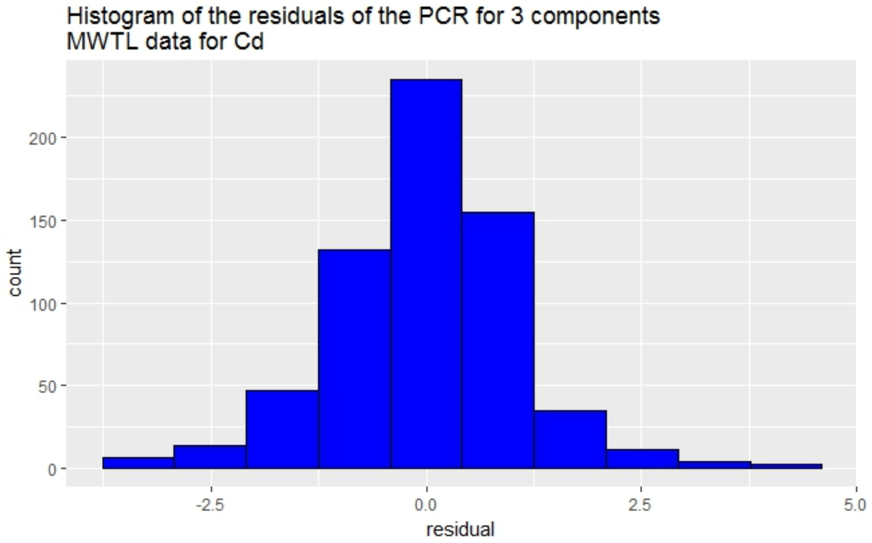
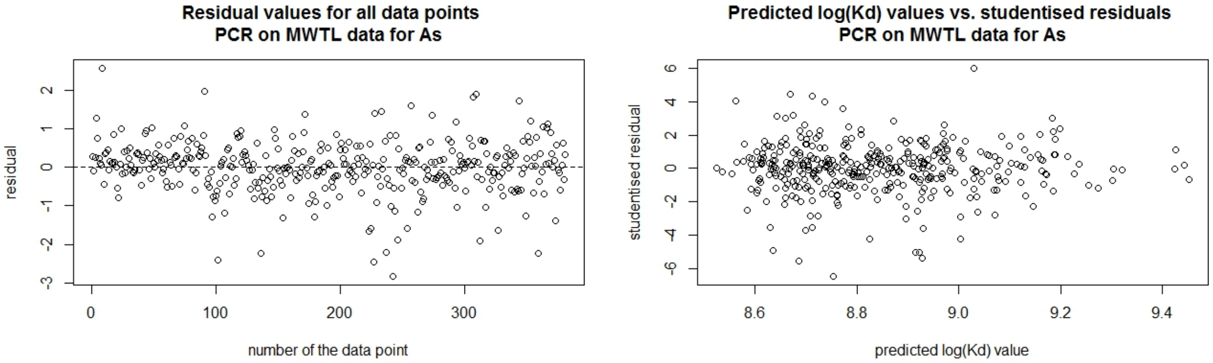


Figure 35: Histogram of the residuals of the PCR model with three principal components for the MWTL data for cadmium.

G.5 Arsenic



(a) (b)

Figure 36: The left panel shows the values of the residuals of the PCR model trained on the MWTL data for arsenic for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

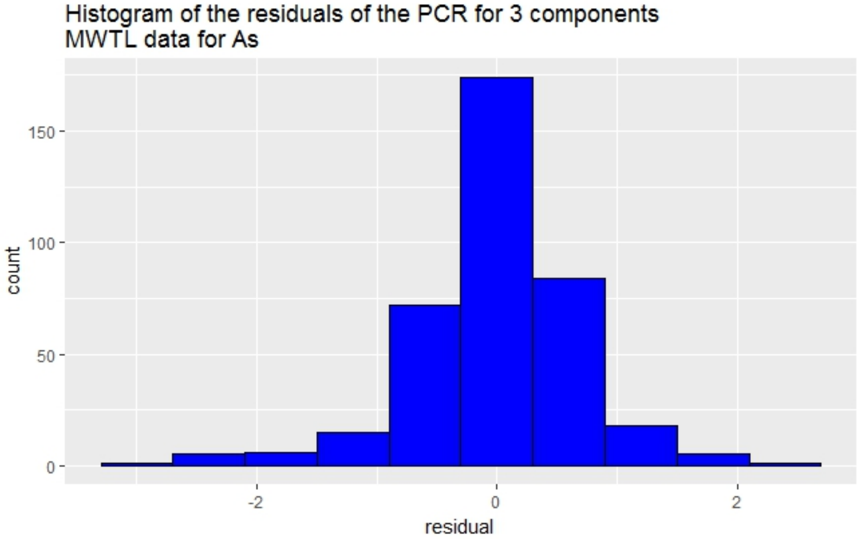
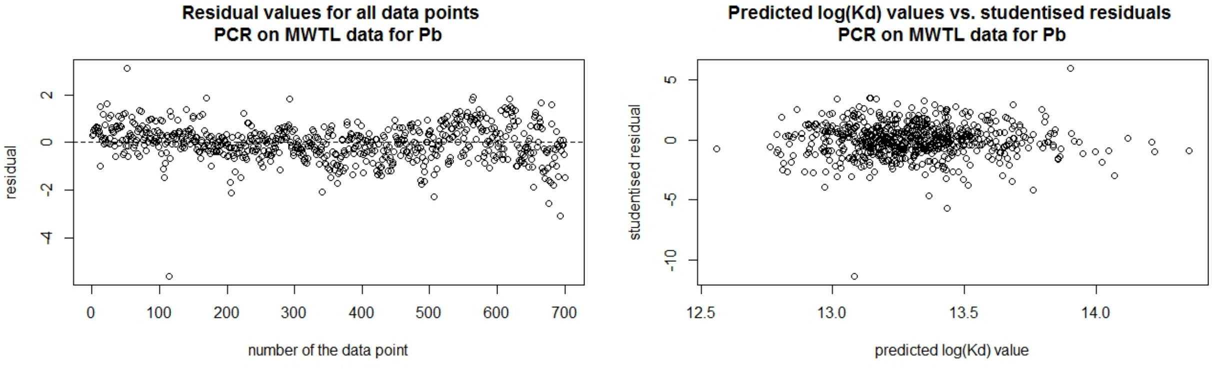


Figure 37: Histogram of the residuals of the PCR model with three principal components for the MWTL data for arsenic.

G.6 Lead



(a)

(b)

Figure 38: The left panel shows the values of the residuals of the PCR model trained on the MWTL data for lead for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

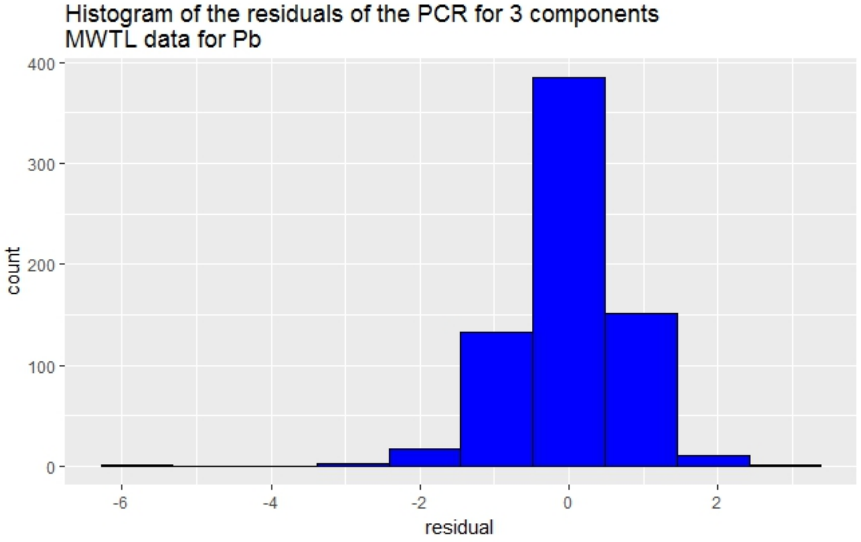


Figure 39: Histogram of the residuals of the PCR model with three principal components for the MWTL data for lead.

G.7 Zinc

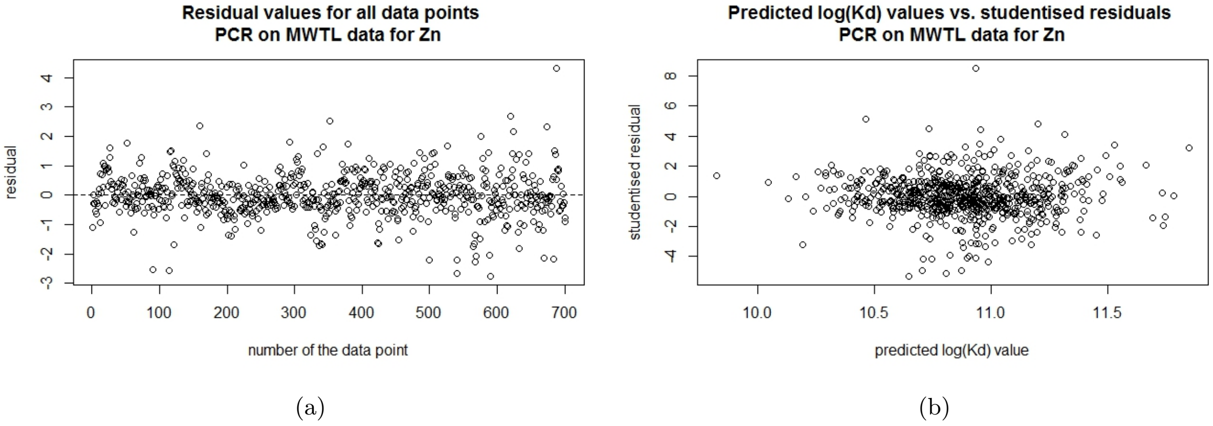


Figure 40: The left panel shows the values of the residuals of the PCR model trained on the MWTL data for zinc for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

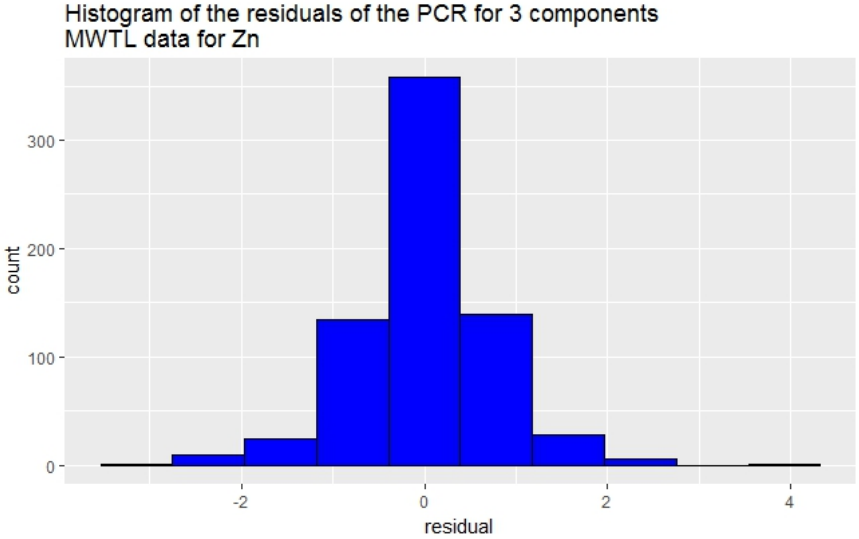


Figure 41: Histogram of the residuals of the PCR model with three principal components for the MWTL data for zinc.

G.8 Regression coefficients

The PCR models are of the following form:

$$\hat{K}_d = \hat{\beta}_1 \cdot SPM + \hat{\beta}_2 \cdot SAL + \hat{\beta}_3 \cdot MeT + \hat{\beta}_4 \cdot YEAR. \tag{37}$$

The regression coefficients $\hat{\beta}_1$ to $\hat{\beta}_4$ for the models with three and four principal components are given in table 20 et 21 respectively.

metal	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$
Cu	-0.23	-0.4	0.11	-0.19
Ni	-0.13	-0.04	-0.23	0.30
Co	-0.16	0.07	-0.16	0.40
Cd	-0.24	-0.86	0.16	0.10
As	-0.04	-0.19	0	0.04
Pb	-0.17	-0.26	0.03	0.04
Zn	-0.24	-0.09	-0.10	0.01

Table 20: Regression coefficients of the PCR models, using three principal components, for the different metals.

metal	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$
Cu	-0.38	-0.26	0.37	-0.12
Ni	-0.33	0.26	0.44	0.76
Co	-0.67	0.47	0.62	0.44
Cd	-0.40	-0.74	0.46	0.27
As	-0.20	-0.12	0.28	0.24
Pb	-0.77	0.06	0.87	0.41
Zn	-0.36	0.02	0.11	0.04

Table 21: Regression coefficients of the PCR models, using four principal components, for the different metals.

The model for Cd with one principal component has the following coefficients: $\hat{\beta}_1 = 0.19$, $\hat{\beta}_2 = -0.22$, $\hat{\beta}_3 = 0.26$ and $\hat{\beta}_4 = -0.13$.

Appendix H Results K_d models - PLS

In this part of the appendix the residuals, their distributions and plots of the studentised residuals versus the predicted $\log(K_d)$ values are presented for the PLS models with three PLS components for each of the metals. This is done to check the linear regression assumptions discussed in appendix F. Only the errors for the models with three components are shown, as these models perform best for most metals.

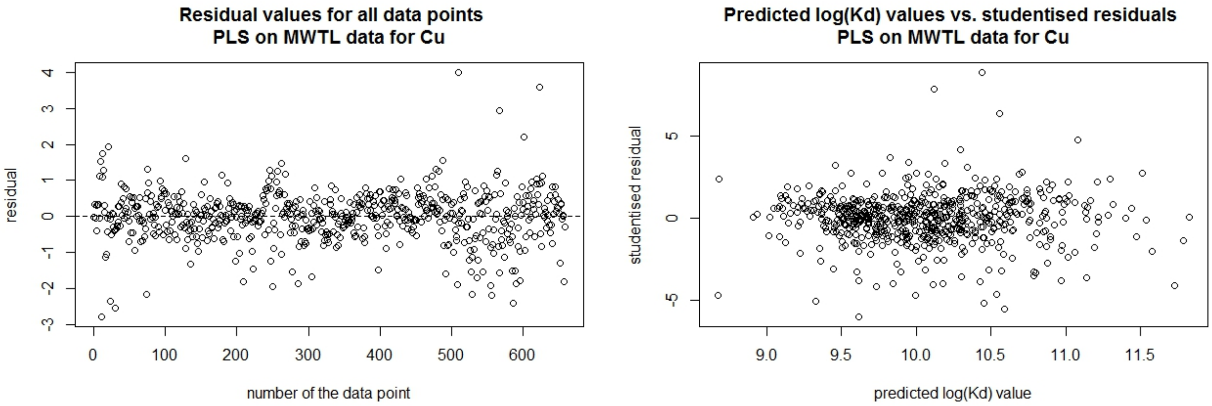
Both for nickel and cadmium there is a spike of the residual values around data point 250, which implies that the errors are not independent for the models for these metals. Also for lead it is clear that the errors are not independent, as a fan-like pattern can be seen, where the absolute values of the errors increase approximately linearly from data point 300 to 700. For the other metals no patterns are detected in the errors, thus implying that they are independent.

The violation of the assumption of independence for nickel, cadmium and lead can cause biased estimates of the standard error and the significance of the regression parameters. In the content of this research, this is not that important.

The other two assumptions: homoscedasticity and normality of the errors, are met for all metals.

Furthermore, the regression coefficients are presented for each of the metals for the models with 3 principal components. For cadmium and arsenic they are also given for the model based on a single principal component.

H.1 Copper



(a) (b)

Figure 42: The left panel shows the values of the residuals of the PLS model trained on the MWTL data for copper for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

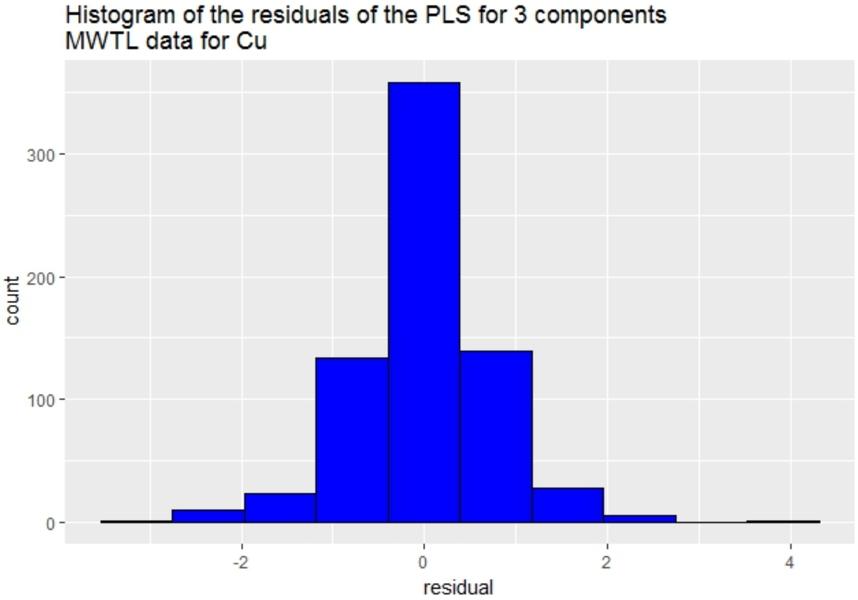
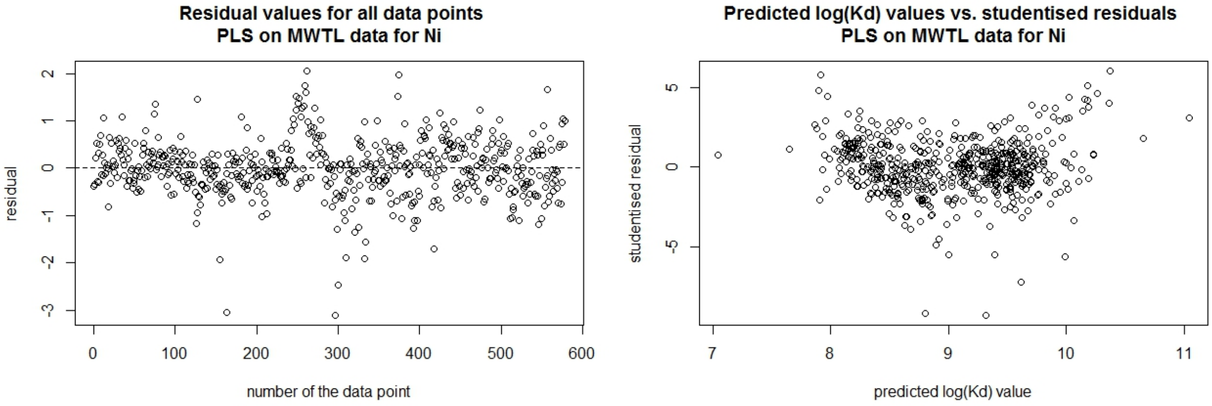


Figure 43: Histogram of the residuals of the PLS model with three principal components for the MWTL data for copper.

H.2 Nickel



(a)

(b)

Figure 44: The left panel shows the values of the residuals of the PLS model trained on the MWTL data for nickel for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

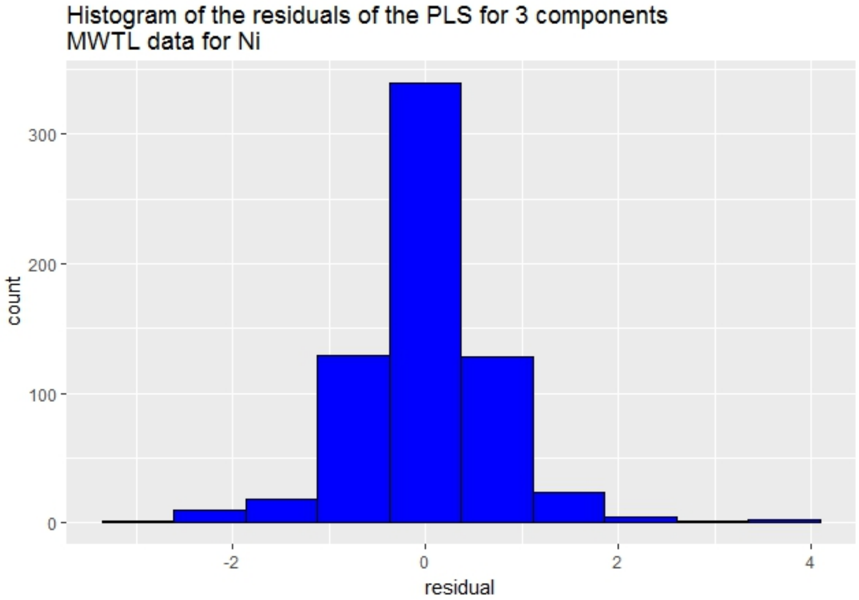
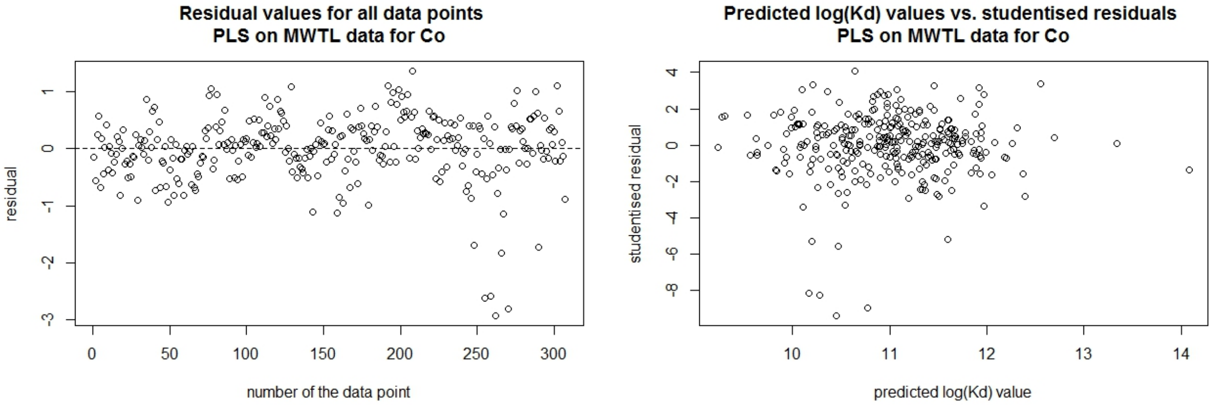


Figure 45: Histogram of the residuals of the PLS model with three principal components for the MWTL data for nickel.

H.3 Cobalt



(a)

(b)

Figure 46: The left panel shows the values of the residuals of the PLS model trained on the MWTL data for cobalt for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

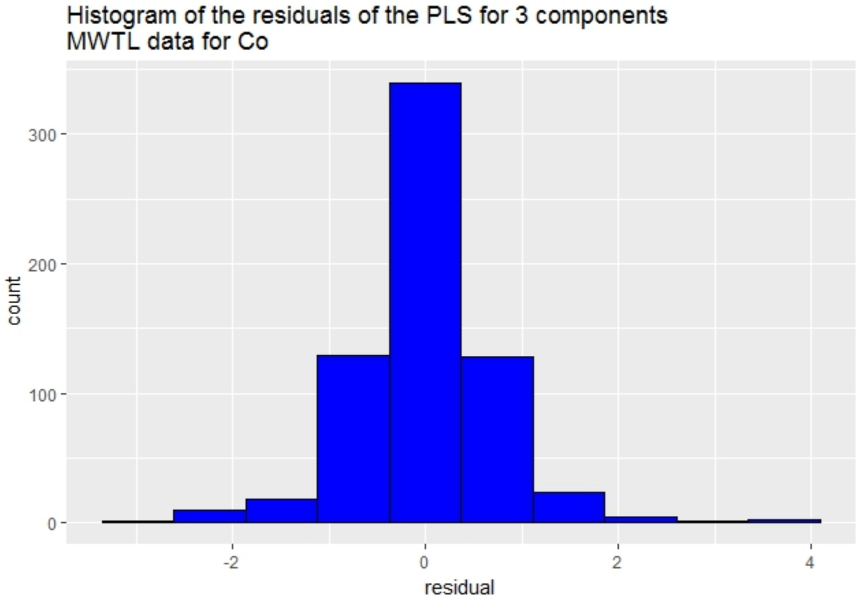
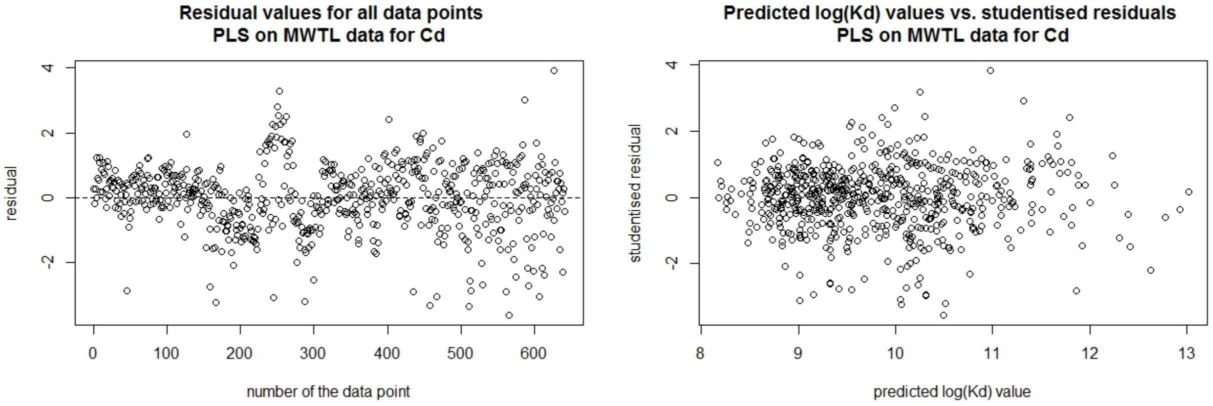


Figure 47: Histogram of the residuals of the PLS model with three principal components for the MWTL data for cobalt.

H.4 Cadmium



(a) (b)

Figure 48: The left panel shows the values of the residuals of the PLS model trained on the MWTL data for cadmium for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

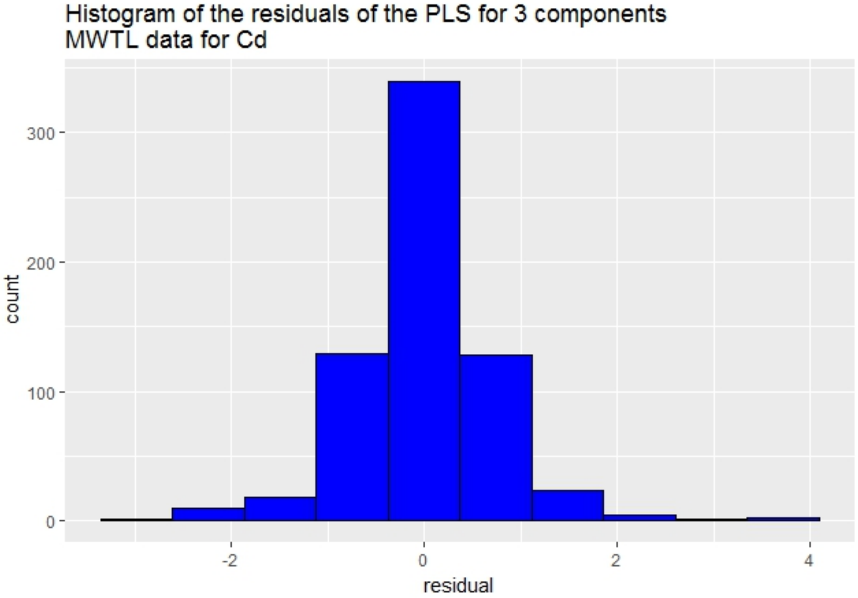
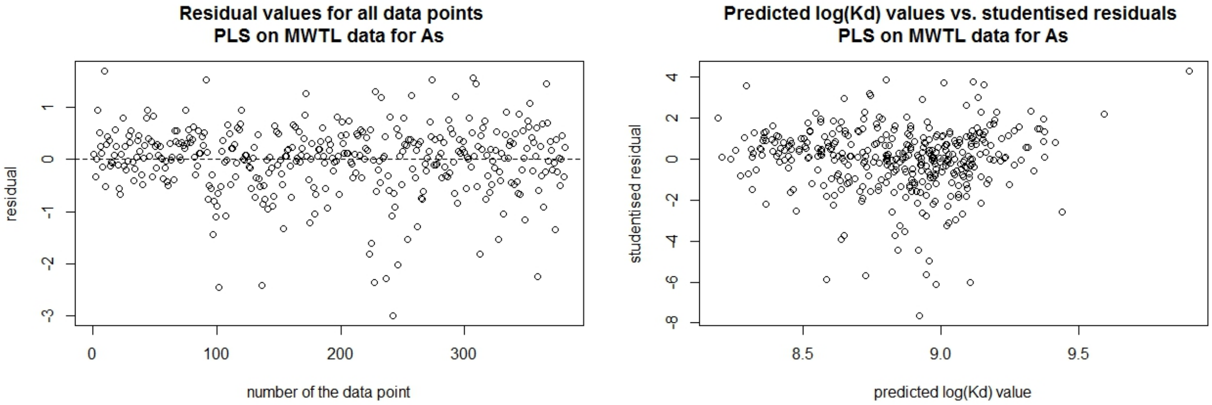


Figure 49: Histogram of the residuals of the PLS model with three principal components for the MWTL data for cadmium.

H.5 Arsenic



(a)

(b)

Figure 50: The left panel shows the values of the residuals of the PLS model trained on the MWTL data for arsenic for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

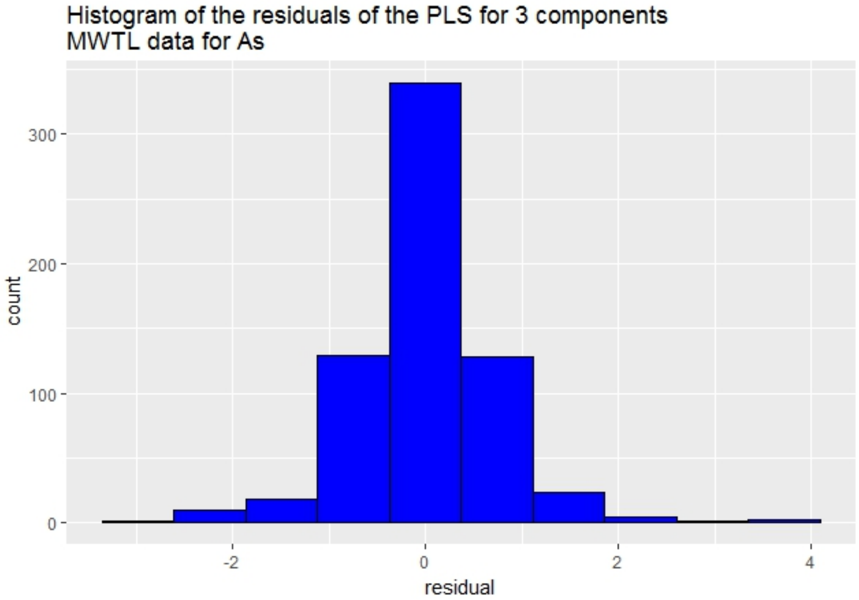
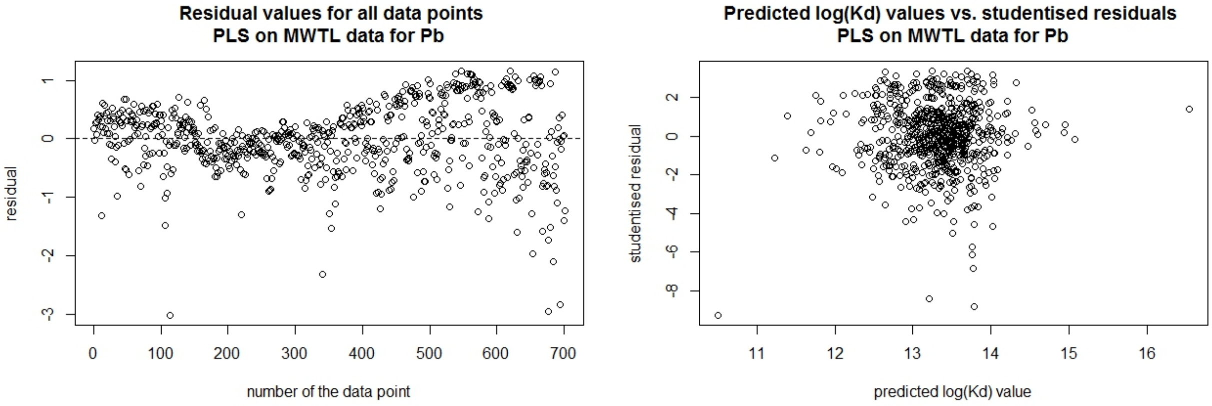


Figure 51: Histogram of the residuals of the PLS model with three principal components for the MWTL data for arsenic.

H.6 Lead



(a)

(b)

Figure 52: The left panel shows the values of the residuals of the PLS model trained on the MWTL data for lead for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

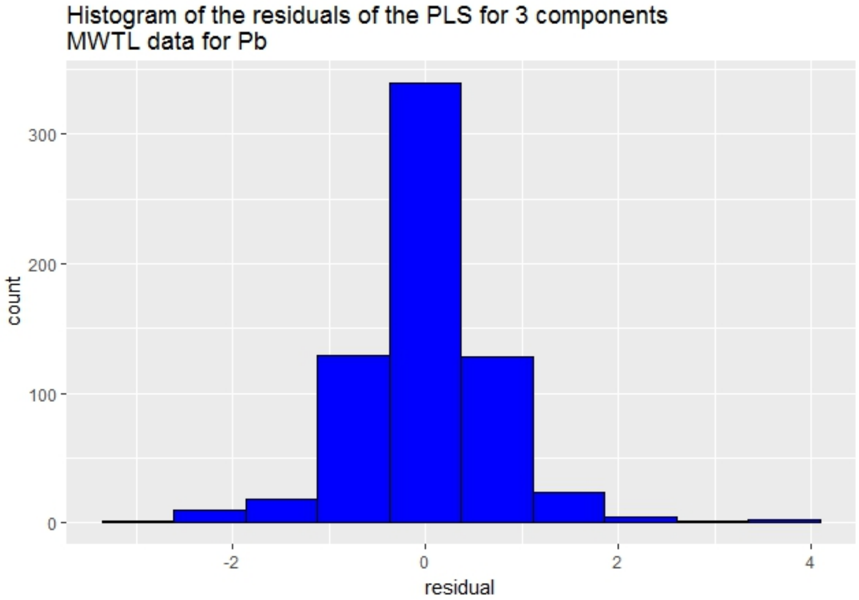
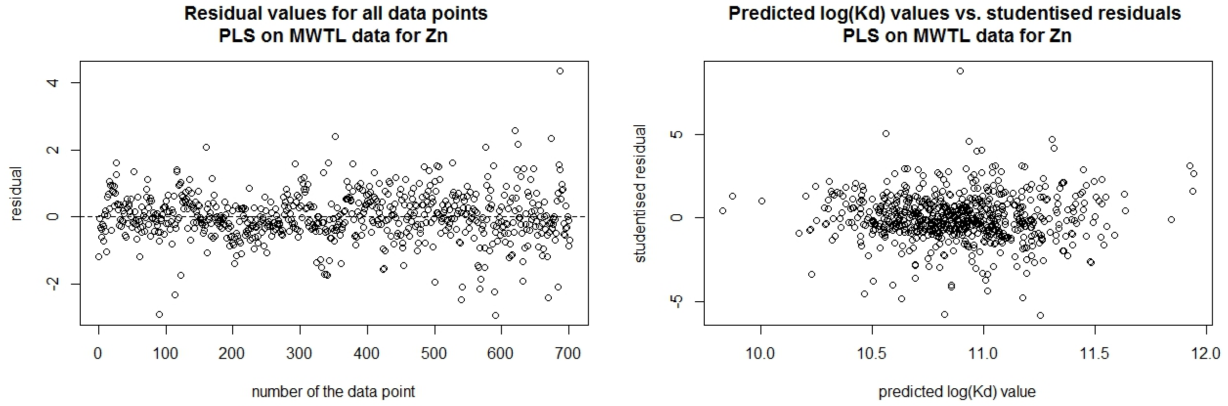


Figure 53: Histogram of the residuals of the PLS model with three principal components for the MWTL data for lead.

H.7 Zinc



(a)

(b)

Figure 54: The left panel shows the values of the residuals of the PLS model trained on the MWTL data for zinc for each data point. The right panel shows the studentised residuals for the same model versus the predicted $\log(K_d)$ values.

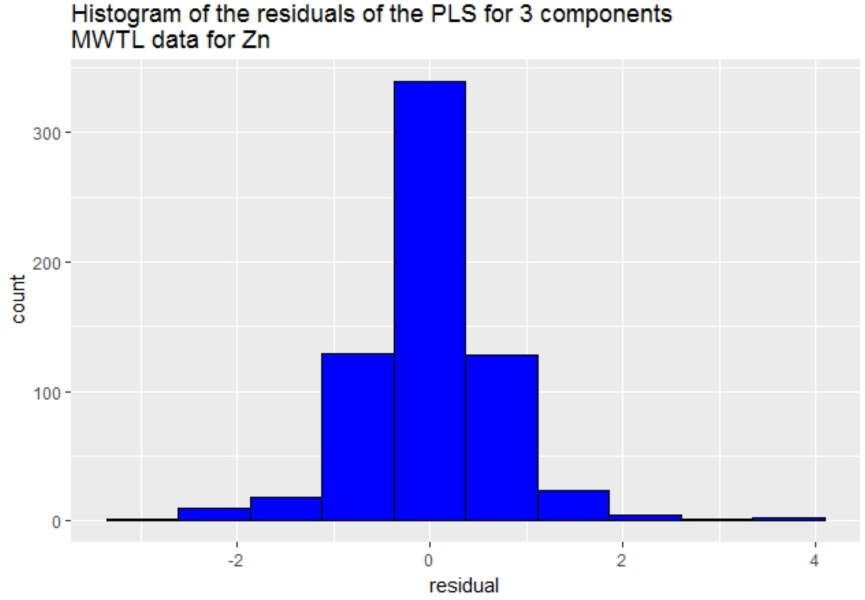


Figure 55: Histogram of the residuals of the PLS model with three principal components for the MWTL data for zinc.

H.8 Regression coefficients

The PCR models are of the following form:

$$\hat{K}_d = \hat{\beta}_1 \cdot SPM + \hat{\beta}_2 \cdot SAL + \hat{\beta}_3 \cdot MeT + \hat{\beta}_4 \cdot YEAR. \tag{38}$$

The regression coefficients $\hat{\beta}_1$ to $\hat{\beta}_4$ for the models with three and four principal components are given in table 20 et 21 respectively.

metal	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$
Cu	-0.36	-0.32	0.31	-0.12
Ni	-0.38	0.11	0.29	0.67
Co	-0.66	0.29	0.48	0.48
Cd	-0.40	-0.76	0.43	0.22
As	-0.21	-0.15	0.26	0.21
Pb	-0.73	0.02	0.83	0.45
Zn	-0.33	-0.04	0.03	0.05

Table 22: Regression coefficients of the PLS models, using three principal components, for the different metals.

The model for Cd with one principal component has the following coefficients: $\hat{\beta}_1 = 0.06$, $\hat{\beta}_2 = -0.49$, $\hat{\beta}_3 = 0.34$ and $\hat{\beta}_4 = -0.04$.

The model for As with one principal component has the following coefficients: $\hat{\beta}_1 = -0.01$, $\hat{\beta}_2 = -0.15$, $\hat{\beta}_3 = 0.12$ and $\hat{\beta}_4 = 0.08$.