rijksuniversiteit groningen / faculty of mathematics and natural Sciences

wln water | onderzoek | advies

The effect of historic rainfall on the quality of sewage

Internship Applied Mathematics

November 2016

Student: M.M. Bronts, s1992740

Company: WLN, Glimmen, department of technology, water quality and advice

Internal RUG supervisor: prof. dr. H.L. Trentelman

External supervisor: Dr. ir. P. van der Maas

Abstract

In 2015 WLN started monitoring the quality of sewage and surface water on several locations in Groningen, Assen and Leeuwarden using EC sensors (measuring conductivity) and turbidity sensors. In periods of long term rain, the sewage is diluted and can cause unwanted overflow on surface water which might become contaminated. It is expected that there is a relation between historic rainfall and the sewage quality. In this report we tried to answer the research question whether there is a relation between historic rainfall and the sewage quality (EC sensor) for the location Leeuwarden, Bilgaard. This is a relevant question, because if such a relation can be found, it is not longer necessary to use the sensors to measure the water quality. Instead, historic rainfall can be used as an important steering parameter for optimized urban wastewater management.

We used data that was measured by an EC sensor in the sewer system in Leeuwarden, Bilgaard and we used historic rainfall data that was collected from a local weather station in Leeuwarden. We expect a non-linear relation between historic rainfall and the sewage quality, because we have time series data which means that there is autocorrelation between the EC measurements over time. To answer the research question we used Generalized Additive Models (GAMs) in R. The model we found gives an R-squared value of 0.61. This model gave the best approximation of the EC.

Contents

1 Introduction

1.1 About WLN

WLN is an independent water laboratory, founded in 1976, where the quality of water is monitored, controlled and treated when necessary. They have experts in biology, chemistry and technology which results in a variety of services offered and thus a broad span of customers. WLN advises the water companies in Groningen and Drenthe, but also governments, knowledge institutes and other companies such as energy companies, chemical industries and recreational institutes. Around 70 people work at WLN, divided over three departments: laboratories for biological and chemical research (28), the department for technology, water quality and advice (12) and the department for business, finance and support (23). This internship was at the department for technology, water quality and advice and took place from 05-09-2016 until 11-11-2016.

In 2015 WLN started a project where the quality of sewage and surface water is monitored using sensors on several locations in the sewer system in Groningen, Assen and Leeuwarden. The sewer system on these locations is a combined system in which rain water and urban wastewater are drained via the same pipe to the sewage treatment plant. In periods of long term rain the sewage becomes diluted. If there is too much diluted sewage, there can be an unwanted overflow at a location which cannot cope with the diluted sewage. In principle this can be minimized by controlling overflow at chosen locations which can cope with this diluted sewage.

The goal of this project by WLN is to determine the potency of sensor technology for overflow control, to optimize the capacity of the existing infrastructure for the drainage and treatment of urban wastewater. This internship is part of this project.

1.2 Problem description

WLN has monitored the sewage quality using sensors on five locations in Groningen, Assen and Leeuwarden over the past year (from September 2015 until October 2016).

1. Groningen: Damsterdiep (population equivalent (p.e.) of 81.838)
2. Groningen: industrial area Euvelgunne (p.e. 8300)
3. Assen: Pittelo Zuid (two locations) (p.e. 2000)
4. Leeuwarden: Bilgaard (p.e. 10.000)

At these locations there are EC sensors measuring the electrical conductivity in the sewage (according to the amount of salt in the water) in μS/cm and sensors that measure turbidity in FTU. We will only look at the EC sensor data, because this sensor gives a good representation of the quality of the water. Furthermore, local rainfall data

in mm per 30 minutes has been collected over the past year.

In periods of long term rain, the sewage is diluted, which theoretically leads to a decrease in the value of the EC sensor (since rain water does not contain much salt). The latter fact seems to be validated by the data. Linear modelling techniques have been applied, with the EC measurements as the response variable, and rainfall measurements as a predictor variable. Unfortunately, the application of these techniques was not successful. This gave a low R-squared value of 0.01 which means that the regression does not give a good approximation of the real data. Therefore it was not possible to conclude anything immediately.

The challenge is to find a relation between: historic rainfall and sewage quality. If we can find such a relation, it is not longer necessary to use the sensors to measure the water quality. Instead, we can use historic rainfall data as important steering parameter for optimal urban wastewater management.

At the beginning of this internship the objective was to analyse all these data and try to find a relation between historic rainfall on location X and the sewage quality on location Y. This appeared to be a lot more work than we thought. Since, the data from Leeuwarden, Bilgaard seemed to be the best data to work with, we reformulated the objective to only analyzing the data from Leeuwarden, Bilgaard and trying to find a relation between historic rainfall and the sewage quality for that specific location. The research question therefore is as follows: 'is there a relation between historic rainfall and the sewage quality (EC sensor) for the location Leeuwarden, Bilgaard?'

1.3 Outline of this report

During this internship I worked with the programming language R which is introduced in Section 2.2. Within R, I used Generalized Additive Models to analyse the EC data from Leeuwarden. In Section 2.1 it is explained what Generalized Additive Models are. In Section 3, some exploratory and formal data analysis is presented. Then I will do a model analysis in Section 4 and discuss which model fits the data best. I will end with a summary and some advice for future research at WLN in Section 6. In the appendices one can find a script with all the models and some illustrative figures of EC and rainfall per month.

2 Preliminaries

Before analyzing the problem I will explain the tools that are used during the statistical analysis in Section 3. I will explain what Generalized Additive Models are, and I will introduce the language R.

2.1 Generalized additive models

Since linear regression with the EC sensor as the response variable and rainfall as a predictor variable gave a low R-squared value of 0.01 we know that linear regression does not give a good approximation of the real data. Our data is more complex, because we have measurements that are taken over time which can be seen as only one experiment. We do not have multiple experiments that can be compared with each other. Since we have time series data, we have to deal with correlation between the EC measurements over time. Because of this we expect that the relation between historic rainfall data and EC data will be non-linear. So, we need a different type of models to analyse the EC data. A good type of models that can be used, are Generalized Additive Models (GAMs). GAMs are introduced by Hastie and Tibshirani in [8].

We will not work with the linear predictor of the form

$$\hat{Y} = \beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n + \epsilon, \tag{1}$$

where $\beta_1, \ldots, \beta_n$ are the parameters, and the predictor variables $X_1, \ldots, X_n$ give a linear estimation of the dependent variable Y. We also have an error term ϵ. Instead we will work with Generalized Additive Models. A GAM can be formulated as

$$g(E[Y|X_1, \ldots, X_n]) = \beta_0 + s_1(X_1) + \cdots + s_n(X_n) + \epsilon. \tag{2}$$

Here Y is the dependent variable, $E[Y|X_1, \ldots, X_n]$ is the conditional expectation of Y, and g is a known function that links the mean of the distribution of Y with the predictor variables $X_1, \ldots, X_n$ in a linear way (the s_j's do not have to be linear). One could for example let g be an inverse function ($g(x) = 1/x$) or a log-function ($g(x) = ln(x)$). In this report we will use the identity function, $g(x) = x$, because we have no reason to change this link. Furthermore, β_0 is a constant parameter and the functions $s_1, \ldots, s_n$ are unknown non-parametric smooth functions of the predictor variables $X_1, \ldots, X_n$. Each of these functions s_j, $j = 1, \ldots, n$, depends on the data and shows what the contribution is of that specific predictor variable X_j in the model. The s_j's are defined as

$$s_j(x) = \sum_{k=1}^{N_j} \beta_{jk} b_{jk}(x), \tag{3}$$

where the β_{jk}'s are the parameters to be estimated and the $b_k(x)$'s are known functions $b_1(x) = 1$, $b_2(x) = x$, $b_3(x) = x^2$, $\ldots$, $b_{N_j}(x) = x^{N_j - 1}$ that form a basis for each s_j.

Thus every s_j is a linear combination of these functions.

The functions s_j are estimated by a non-parametric 'scatter plot smoother'. If nothing is known about the distributions of the variables and the number of parameters that has to be estimated, we use non-parametric estimation methods. There are different ways to estimate the functions s_j. The running mean is a simple example of a scatter plot smoother. Another way to estimate the s_j's is via the minimization of a penalized least squares score given by

$$\sum_{i=1}^{m}((Y_i - \sum_{j=1}^{n} s_j(x_{ij}))^2) + \sum_{j=1}^{n} \lambda_j \int_a^b (s_j''(t))^2 dt. \tag{4}$$

Here $\lambda_j \geq 0$ are smoothing parameters, m is the number of measurements with $a \leq x_1 \leq \cdots \leq x_m \leq b$ ordered, Y_i is the i-th measurement of the dependent variable Y, x_{ij} is the i-th measurement of the j-th predictor variable and s_j'' is the second derivative of the function $s_j(x)$. The first term in equation (4) makes sure that s_j fits the data well enough and the second term imposes a penalty if s_j is too rough. λ_j is a smoothing parameter which determines the level of smoothness by penalizing the curvature of s_j. If λ_j is large, the curve is smoother and if λ_j is smaller, the curve is more wiggly. So, we want equation (4) to fit the data, but we do not want it to be too rough.

The s_j that minimizes equation (4) is found via a backfitting algorithm. Given an estimate $\sum_{k\neq j} \hat{s}_k(X_k)$ (for example the mean of Y), we find an estimate of $s_j(X_j)$ by smoothing the residual $[Y - \sum_{k\neq j} \hat{s}_k(X_k) \mid X_j]$. In this way we can find estimates of all s_j's. With these estimates we can improve the estimate of $\sum_{k\neq j} \hat{s}_k(X_k)$. This cyclic process can be repeated until the smooths of the residuals converge to $\hat{s}_j(X_j)$, for each $j = 1, \ldots, n$. The convergence is checked by equation (4).

The s_j that minimizes equation (4) is called a cubic smoothing spline, which is one way of estimating the s_j's. A smoothing spline is a function that piecewise connects two points (knots) by a polynomial function. A cubic smoothing spline means that we have a cubic polynomial function. In the knots there is a high degree of smoothness. We want to keep the number of knots as low as possible, but still capture important patterns. If the number of knots is too high, there is a risk for oversmoothing. Oversmoothing is a problem, because this means that the model is trying to fit the outliers and the noise instead of reflecting the overall data. It can happen that the R-squared value is misleading. If we would have a different sample set of the data, we want our model to fit that sample too.

2.2 Programming language R

When trying to solve this problem we will make use of the programming language R. This is an open-source computer language for statistical computing, developed by Ross

Ihaka and Robert Gentleman in 1990. 'R Studio' is a program in which one can work with R. [10] gives a good introduction to working with R. R is a language for beginners in statistics and in this report it is used for analyzing the data.

R has many packages for different types of functions. Before one can use these functions one has to install the right packages by typing **install.packages("packagename")** and then call them via **library(packagename)**. If one needs help with a package or function one can type **?name**.

2.2.1 Modelling GAMs in R

A GAM is a type of model that predicts the response variable by using a sum of smooth functions as in equation (2). For working with GAMs in R one needs the packages **mgcv**, **itsadug**, **ggplot2** and **visreg** and the functions **gam** or **bam**. These functions fit a GAM to the data. When the data set is large it is best to use **bam**. If we want to predict **Y** non linearly from, say, two variables **X1** and **X2**, we write

```
> mod ← gam(Y ∼ s(X1) + s(X2), family = ..., data = name,
+ method = "REML")
```

where the '+' on the second line in the script means that this line is still part of the previous line. With the functions **gam** and **bam** one chooses a method for smoothing the parameter estimation and one also chooses a family which specifies the distribution and the link function (default is Gaussian with an identity link, so from now on it is left out). The method we will use is "REML", which stands for restricted maximum likelihood. The likelihood function is defined as

$$\mathcal{L}(\theta|X) = P(X|\theta),$$

where θ is a set of parameter values and X is the outcome. With the method "REML" the likelihood function is calculated from a transformed set of data in such a way that nuisance parameters do not have any effect. Nuisance parameters are not meaningful for the goal of the analysis, but have to be accounted for in the model.

Within each function **s()** in **gam** we can specify the number of knots we want to use by **k**. This **k** equals the basis dimension N_j in equation (3).

```
> mod ← gam(Y ∼ s(X1, k=4) + s(X2, k=7), data = name,
+ method = "REML")
```

Furthermore we can specify which scatter plot smoother we want to use for the estimation of each **s()**. Typing **?smooth.terms** gives a list of all smoothers. An example

of a scatter plot smoother is a cyclic cubic spline, which estimates **s(X1)** correct if **X1** shows a cyclic effect in Y. If **X2** is for example a categorical variable that contains categories (day, month...) we can use a smoother that gives an estimation via a random effect (per day, month...). Adding a variable as a random effect to the model means that it will be part of the error term of the model and therefore it is not necessary to specify the basis dimension **k**.

```
> mod ← gam(Y ∼ s(X1, bs = "cc", k=4) + s(X2, bs = "re"),
+ data = name, method = "REML")
```

If one wants to fit the model only for a subset of the data, one can write the following, where the **subset** command must be a logical expression.

```
> mod ← gam(Y ∼ s(X1) + s(X2), data = name, method= "REML",
+ subset = Y>0 & Y<100)
```

When the model has autocorrelation in the residuals it means that the residuals are correlated with each other. Residuals are the differences between the observed and predicted responses. Autocorrelation is a value that lies between -1 and 1 and is zero if the residuals are completely random. We want the residuals to be random, or at most only correlated with neighboring residuals. To do something about autocorrelation in the model we can write the following:

```
> mod ← bam(Y ∼ s(X1) + s(X2), data = name, method= "REML")
> AC ← start_value_rho(mod, plot = T)
> modAC ← bam(Y ∼ s(X1) + s(X2), data = name,
+ method = "REML", rho = AC, AR.start = AR)
```

Here **AC** is the value of the autocorrelation starting after the first lag. If this value is high, it means that the residual of the measurement is highly correlated with the residual of two measurements further. In this case we account for autocorrelation in the model 'modAC'. We set rho equal to the value **AC**. In the model 'modAC' **AR** is a variable which is set TRUE if we want to check for autocorrelation and FALSE if not. In this way one can specify a time step for which one wants to account for autocorrelation. This only works with the function **bam**.

2.2.2 GAM diagnostics and summary

If we want to know how well the GAM fits the data, we look at the GAM diagnostics and the summary via **summary(model)** and **diagnostics(model)**. The summary tells us

the significance of each variable and gives the R-squared value. From the diagnostics we obtain several plots that give information about the following things.

- A general trend in the residuals. The idea is that the residuals are randomly spread in the plot. If one can get information from this plot (a decrease or an increase of the residuals) it means that there is a predictor variable missing in the model that should capture this information. One should not be able to predict the error term.
- The residuals for every predictor variable which can be used to see if the model captures the trend of the residuals for each predictor.
- The distribution of the residuals plotted against the residuals of a normal distribution for comparison. When the residuals look normally distributed the model fits the data better and the results will be more reliable. The residuals should lie close to the line $y = x$.
- Autocorrelation in the residuals (ACF).
- Averages of random effect smooths (if added).
- Distributions of predictor variables that are numeric.

3 Statistical analysis

When trying to find a relation between historical rainfall and the sewage quality we have to find a statistical model that gives as output a good estimation of the EC data with historic rainfall data as input. We have to keep in mind that we can only give an approximation of the EC data. We will try to find a model that fits the data best and captures important information. We will do this via a black box approach, which means that the computer tries to find relations between EC data and rainfall data using pre-programmed models. With GAMs in R we have the ability to add more predictor variables that might influence the EC data.

First we prepare the data, so that we can work with it. As explained later on, we extend the data set with some new predictor variables that we might need when modelling GAMs. Second we do some exploratory analysis. We compute the mean, standard deviation and median of some of the variables and we look at several plots of the EC and rainfall data.

3.1 Data preparation

During this internship we have only looked at the data measured in Leeuwarden, Bilgaard. From now on, if we talk about the data, we mean the data measured in Leeuwarden, Bilgaard.

When looking at the data, we notice that there is a certain delay between the rainfall data and the EC data. When it rains, the value of the EC sensor does not immediately decrease (or does not decrease at all, because it's not raining hard or long enough). This can be explained by the fact that the rain water is not immediately present at the location of the sensor. It has to travel through the sewer system before being measured by the sensor. If after a dry period it starts to rain, the sensor first measures the pollution that was stored in the sewer system during dry periods, called the first flush, before decreasing. This also influences the delay in the EC data.

There are a few things we notice from the data:

- The EC sensor has measurements from 22-9-2015 13:30 to 2-9-2016 12:30. The rainfall data is collected from a local weather station during the same period (see Figure 1).
- The EC measurements before 18-11-2015 00:00 canot be used, because the sensor was placed too high in the sewer system which lead to the sensor running dry.
- The EC sensor once measured 0 μS/cm, which probably was due to something covering the sensor. This measurement was deleted from the data set.

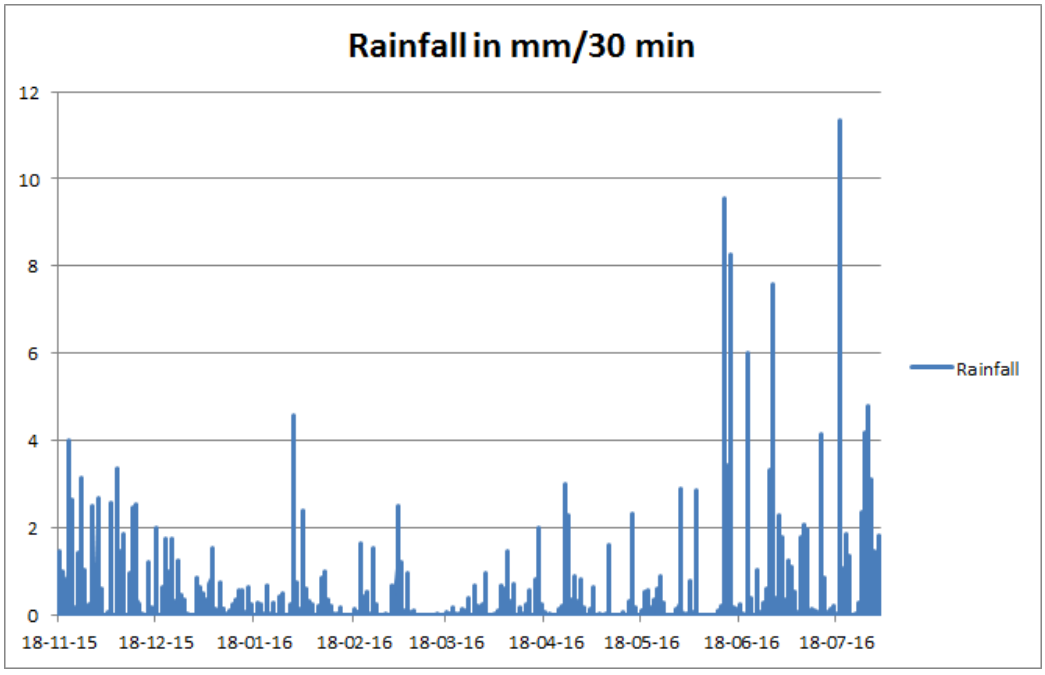

Figure 1: Rainfall in mm/30 min for Leeuwarden, Bilgaard

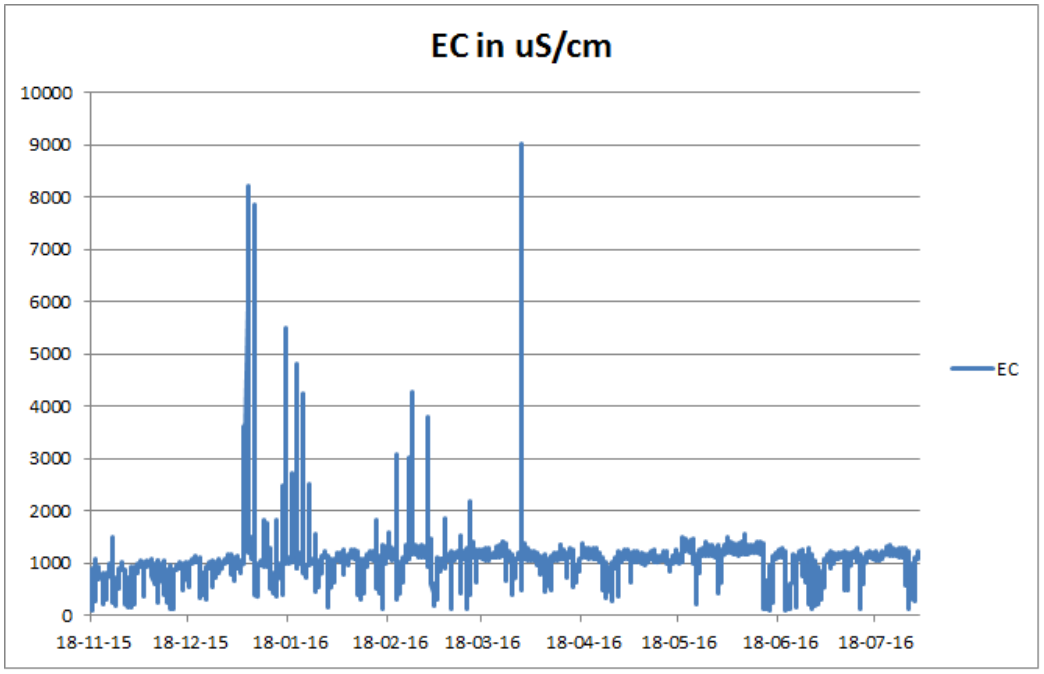

Figure 2: EC in μS/cm for Leeuwarden, Bilgaard

- There are very high peaks in the EC data around 5-1-2016, 17-1-2016 until 20-1-2016 and 20-2-2016 until 1-3-2016. See Figure 2. These peaks are due to the prevention of slippery roads by spreading salt. The high peak on 30-3-2016 is probably a wrong measurement.

- The rainfall data after 1-8-2016 is not correct, because of wrong measurement of the local weather station. So, we have left this data out of the analysis.

- In total we have 12329 EC and rainfall measurements over the period 18-11-2015 00:00 until 31-7-2016 23:30 which can be used for the analysis.

Next to the original EC data and the collected rainfall data we have constructed some additional variables that can be of interest when working with GAMs. For example, we added temperature, because this might link low temperatures (below zero) to high peaks in the EC measurements (prevention of slippery roads). Furthermore we added some time variables to check whether there is a cyclic effect per day or it might be the case that each day or month has a specific effect. To capture historic rainfall of the last 12 hours we added another variable as given below. We added all of them to the data set.

- We have collected temperature data in degrees Celsius per 30 minutes from the KNMI website [3]. The variable is called **Temp**. We made this variable smooth and called it **avg.temp**, because it is a running average of the temperature.

- We have constructed a variable **hours_nm** that represents a cyclic effect within a day. The variable is a vector in which the subvector $(1, 2, \ldots, 48)$ is repeated for every day. Since the data is measured for every 30 minutes, this subvector runs from 1 up to 48 in one day.

- We have constructed a variable **days_factor** in which we count every day in the data set. This is a categorical variable of the form $(1, \ldots, 1, 2, \ldots, 2, \ldots, 257, \ldots, 257)$, where 257 is the total number of days in the data set. This variable contains 257 categories.

- In the same way as the variable **days_factor** we have constructed **month_factor**. This categorical variable is of the form $(1, \ldots, 1, 2, \ldots, 2, \ldots, 9, \ldots, 9)$ and contains 9 categories.

- We have constructed the variable **tm** which counts from 1 till the number of measurements (12329) to see what happens over time.

- We have constructed the variable **Rainfall_sum24** in which each element is equal to the sum of the previous 24 elements (12 hours) plus itself. We put zeros on the first 24 elements of this vector. This is a variable that captures historic rainfall.

3.2 Exploring the data

Before going to the model analysis, we do some exploratory data analysis. We can compute the mean, standard deviation and median of some of the variables in the data set (see Table 1).

Variable	Min	Max	Mean	Std. dev.	Median
EC (μS/cm)	101.4	9006	946.5	250.5	1035.2
Rainfall (mm/30 min)	0	11.35	0.053	0.305	0
Temp (°Celsius)	-4.5	31.5	9.3	6.1	8.7
Rainfall_sum24 (mm/30 min)	0	29.54	1.315	2.924	0.15

Table 1: Minimum, maximum, mean, standard deviation and median of some variables

Furthermore, we see in the histograms in Figure 3 that the data is highly skewed, which means that we have heavy tailed data. The EC data without salt peaks has a heavy tail on the left and the rainfall data has a heavy tail on the right. The percentage of the rainfall measurements that are less than 1 mm/30 minutes is 99%. In Figure 4 we left out the rainfall measurements that are less than 1 mm/30 minutes to zoom in. In Appendix B the histograms of EC per month and Rainfall per month are given in the Figures 10 and 11. From the skewness of the data we conclude that our data is not normally distributed.

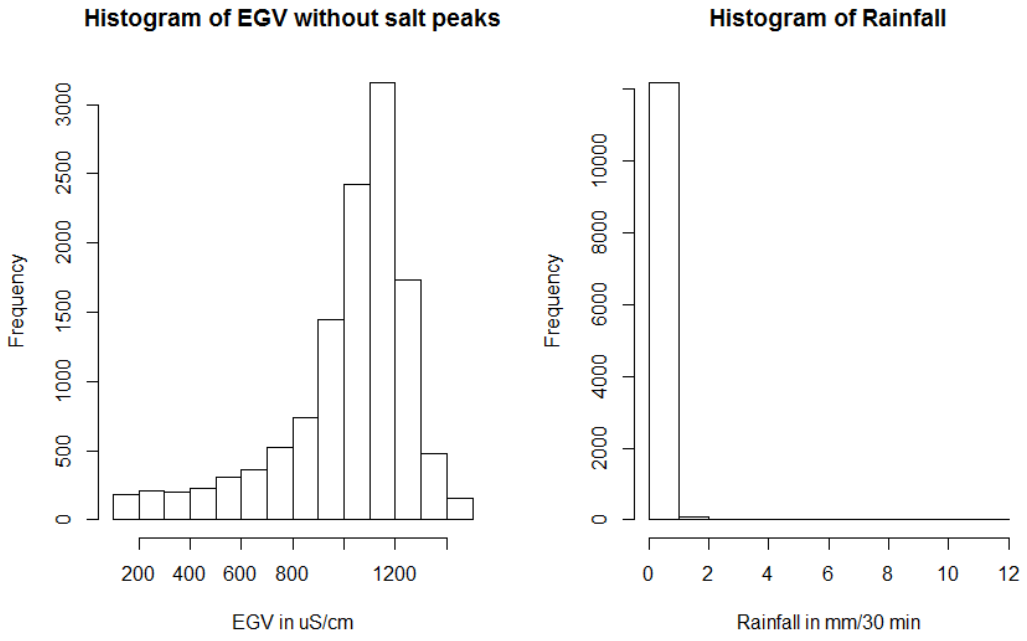

Figure 3: Left: histogram of EC<1500. Right: histogram of Rainfall

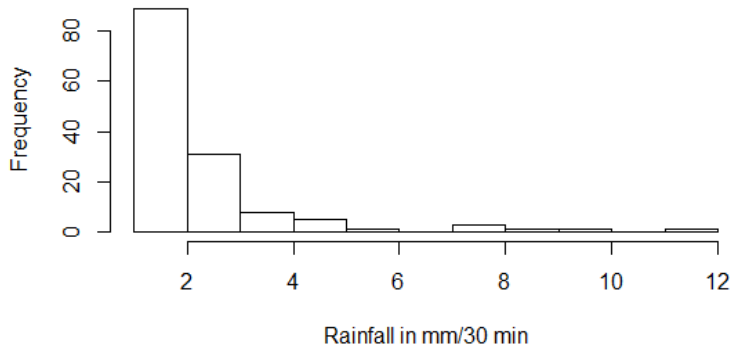

Figure 4: Histogram of rainfall measurements $\geq$1 mm/30 minutes

In Figure 5 we see the Q-Q plots (quantile-quantile plots) of both **EC** and **Rainfall**. In a Q-Q plot the distribution of the data is plotted against a normal distribution. If the data was normally distributed, it had to be the case that all points were on the line $y = x$, which is also plotted. This is certainly not the case for our data (both EC and rainfall). In this way we can also see that our data is not normally distributed.

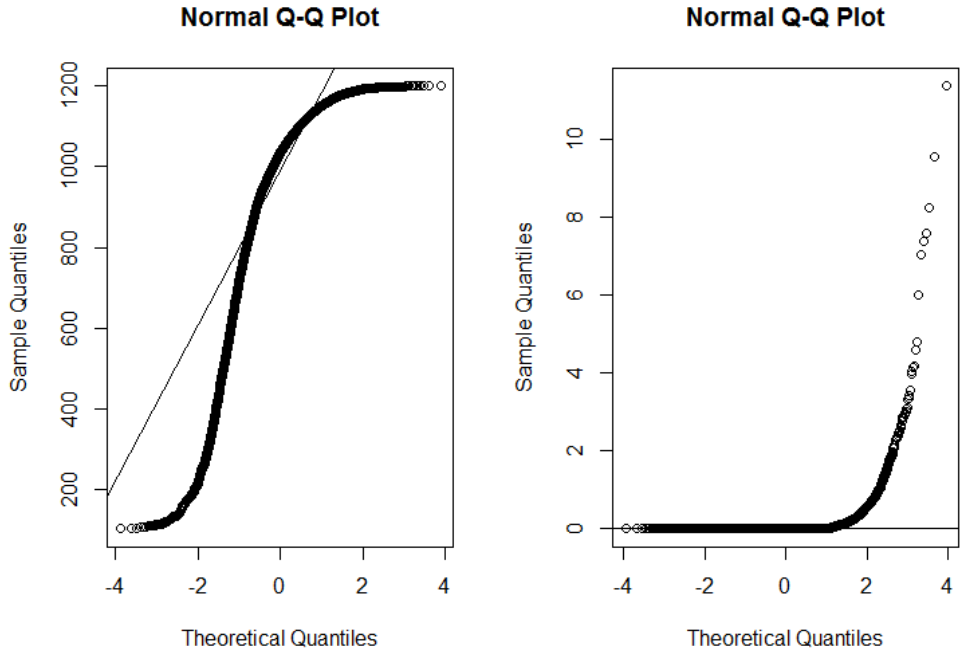

Figure 5: Left: Q-Q plot of EC. Right: Q-Q plot of Rainfall

4 Results

To find a model that fits the data best, we work with GAMs. In this section we state multiple models to see which model gives the best approximation of the EC data. We discuss whether several variables that we stated in Section 3 should or should not be part of the final model.

mod1
If we look at the results of the following model, we get an R-squared value of 0.11 which is better than the 0.01 we had with a linear regression model. The difference is that this is a non-linear approach to find a relation between EC and rainfall.

```
> mod1 ← bam(EC ∼ s(Rainfall, bs = "cr", k=10),
+ data= dataleeuwarden, method = "REML", subset = EC<1500)
```

In this model we only look at **Rainfall** as a predictor variable for **EC**. For this the cubic regression (smoothing) spline is used with a basis dimension of **k=10**. This means that a cubic polynomial is used to estimate a smooth function through the knots. We work with the function **bam** instead of the regular function **gam**, because we have a large data set. We only look at the subset of EC measurements less than 1500. The method we use is "REML" with the default family Gaussian.

The diagnostics of 'mod1' show high autocorrelation in the residuals (close to 1), as can be seen in Figure 6. We have tried to solve this problem of autocorrelation in the way we saw in Section 2.2.1 for an hourly time step, but this gave a much lower R-squared value of 0.03.

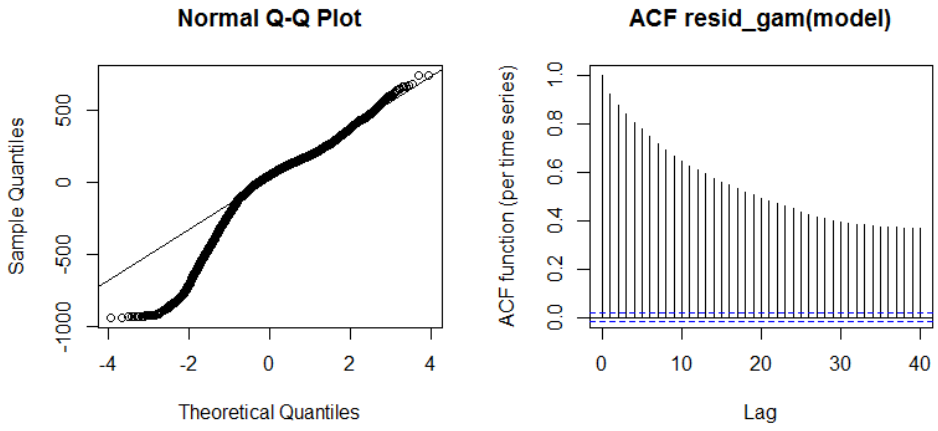

Figure 6: Left: normal Q-Q plot of mod1. Right: ACF of residuals mod1

We also see in Figure 6 a normal Q-Q plot where the distribution of the residuals of the data is plotted against those of a normal distribution. We see that there is a heavy tail, but most of the residuals lie on the line $y = x$.

mod2

When looking at the data we noticed a delay between rainfall and EC. We tried to capture this delay with the variable **Rainfall_sum24**. By making the variable in which each element is the sum of the previous 24 elements plus itself, we included historic rainfall in the next model. If, for example, the value of an element of **Rainfall_sum24** is low, it means that there has been a dry period before the current rainfall measurement. The opposite holds for a long period of rain. We have also looked at other variables where we captured for example 6 or 24 hours of historic rainfall, but it appeared that the variable **Rainfall_sum24** was the best predictor variable. The model below is a good basis to start with and gives an R-squared value of 0.59.

```
> mod2 ← bam(sqrt(EC) ∼ s(Rainfall_sum24, k=10),
+ data= dataleeuwarden, method = "REML", subset = EC<1500,
+ rho = AC, AR.start = AR)
```

Here we have **Rainfall_sum24** as a predictor variable (estimated via the default spline with a basis dimension of **k=10**) and we transformed the EC data by taking the square root. Transforming data means that one performs the same mathematical operation on every data element. A transformation is meant to change the shape of the distribution and therefore must be non-linear. By taking the square root of the EC data the correlation between **EC** and **Rainfall_sum24** increases and we get better results in the diagnostics. We accounted for autocorrelation in this model with an hourly time step. The vector **AR** looks as follows: (TRUE, FALSE, . . . , TRUE, FALSE).

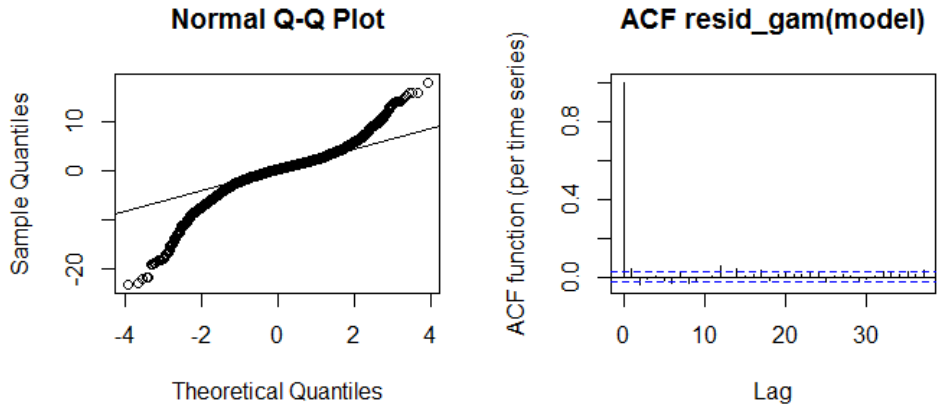

Figure 7: Left: normal Q-Q plot of mod2. Right: ACF of residuals mod2

The diagnostics of 'mod2' look good. As one can see in Figure 7 we solved the problem of autocorrelation in the residuals. The correlation stays most of the time between the blue dotted lines. We have tails in the normal Q-Q plot of the residuals, so it is not as good as the previous model, but still acceptable.

mod3
Another variable we can think of as being important is **Temp**, but adding it to the model does not help a lot. R-squared goes to 0.6 and the diagnostics remain similar. Adding the variable **avg.temp** and performing a different transformation on the EC (namely taking the natural logarithm) gives an R-squared values of 0.61 which is an improvement of 0.02 of 'mod2'. The model is given as 'mod3'.

```
> mod3 ← bam(log(EC) ~ s(Rainfall_sum24, k=10) +
+ s(avg.temp, k=10), data= dataleeuwarden, method = "REML",
+ subset = EC<1500, rho = AC, AR.start = AR)
```

The influence of temperature can be due to the fact that with higher temperatures the water starts evaporating, which implies that the concentration of the salt in the sewage increases. This might lead to a higher value of the EC sensor. The influence of temperature is not very big, and therefore one can choose to leave it out if better predictor variables are found.

mod4
If we add the variable **days_factor** to the model as a random effect we see a big increase in the R-squared value: 0.8. Adding this as a random effect, means that every day has an individual effect in the residuals. Think of the days as different people doing the same experiment, but having their own random effect. We do not know why this leads to such a high R-squared value, because there is no logical explanation why adding this variable should help. Since this variable is estimated as a random effect, it is part of the error term in the model. This means that we can predict **EC** partly from the residuals which means that our model is not correct. Therefore, there is probably a variable missing in the model which should capture this information. The model as discussed is given below as 'mod4'.

```
> mod4 ← bam(sqrt(EC) ~ s(Rainfall_sum24, k=10) +
+ s(days_factor, bs="re"), data= dataleeuwarden,
+ method = "REML", subset = EC<1500, rho = AC,
+ AR.start = AR)
```

mod5

If we leave out the variable **days_factor** and add the time variable **tm** instead we get an R-squared value of 0.69. This is an increase of 0,11 in comparison to 'mod2'. It means that **tm** also captures some important information, but time itself is not a logic variable to add to a model. It is not an explanation for what is happening in the model. The model is given as 'mod5'.

```
> mod5 ← bam(sqrt(EC) ∼ s(Rainfall_sum24, k=10) +
+ s(tm, k=50), data= dataleeuwarden, method = "REML",
+ subset = EC<1500, rho = AC, AR.start = AR)
```

For this model we can predict the EC variable over time (**tm**) for a constant value of **Rainfall_sum24**. If **Rainfall_sum24** becomes constant, it means that we are not looking at a sum of historic rainfall anymore, but just at a constant vector of rainfall in mm/30 minutes. We take the variable **Rainfall_sum24** equal to a constant variable of 1 mm/ 30 minutes, i.e. the variable $(1, \ldots, 1)$. In Figure 8 we plotted the estimation of the EC for this model against the number of measurements for a constant value of **Rainfall_sum24**. Here we see something that looks like a two-weekly cycle. This might have something to do with the fact that the EC sensor is being cleaned every two weeks, but it is not something we can say with certainty. One can check this by looking at the moments when there was maintenance and compare them to Figure 8. When this does not lead to anything, it can be the noise in the EC data and in that case it means that we are oversmoothing. Adding this variable **tm** is not a good idea, but it can be used to find out what is happening in the background. It is hard to capture this information with another variable and we did not manage to do so.

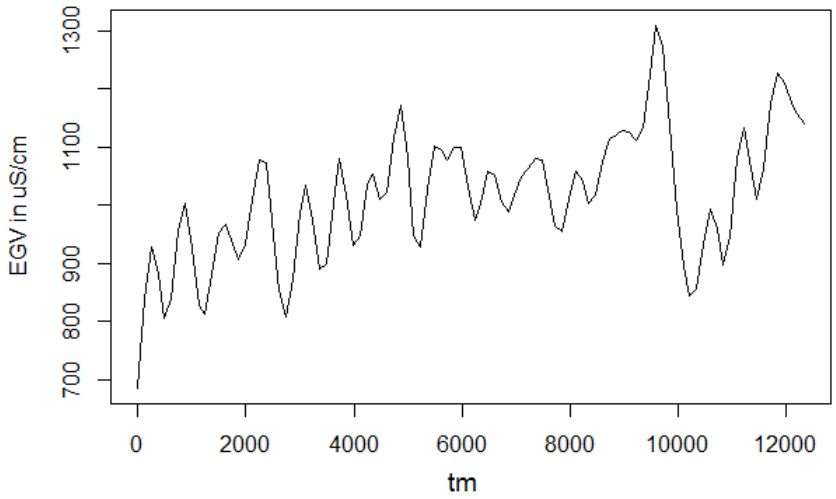

Figure 8: **EC** over time for a constant value of rainfall of 1 mm/30 min

To conclude this section, the "best" model we have seen so far is 'mod4' with an R-squared value of 0.8, but we were not able to find a logical explanation why adding the variable **days_factor** should help. This probably means that this model is not correct and that there is another variable missing.

A correct model that fits the data best is 'mod3' with an R-squared value of 0.61. This is a big improvement of the R-squared value of 0.01 we had in the beginning of this project. The formula that belongs to this model is of the form

$$\log(\text{EC}) = \beta_0 + f(\text{Rainfall\_sum24}) + g(\text{avg.temp}) + \text{error}.$$

So, the logarithm of the EC is a constant β_0, plus a function of historic rainfall, plus a function of temperature, plus an error term in which we accounted for autocorrelation in the residuals.

In Figure 9 we see a scatter plot of **EC** against **Rainfall_sum24** together with the estimation of **EC** for this model for a constant temperature of 10 degrees Celsius. For other temperatures the plot looks similar which is due to the fact that temperature does not have a big impact on the value of **EC**. We see that the relation between **EC** and **Rainfall_sum24** is non-linear as expected. In the figure we see that after 18 mm/12 hours of rainfall the value of **EC** starts to increase, which is probably because of the fact that we do not have a lot of measurements with a high amount of rainfall. It can also be due to the fact that if a shower stopped 11 hours before the current EC measurement, the EC has stabilized but the variable **Rainfall_sum24** still uses these rainfall measurements to form the sum of the previous 24 measurements.

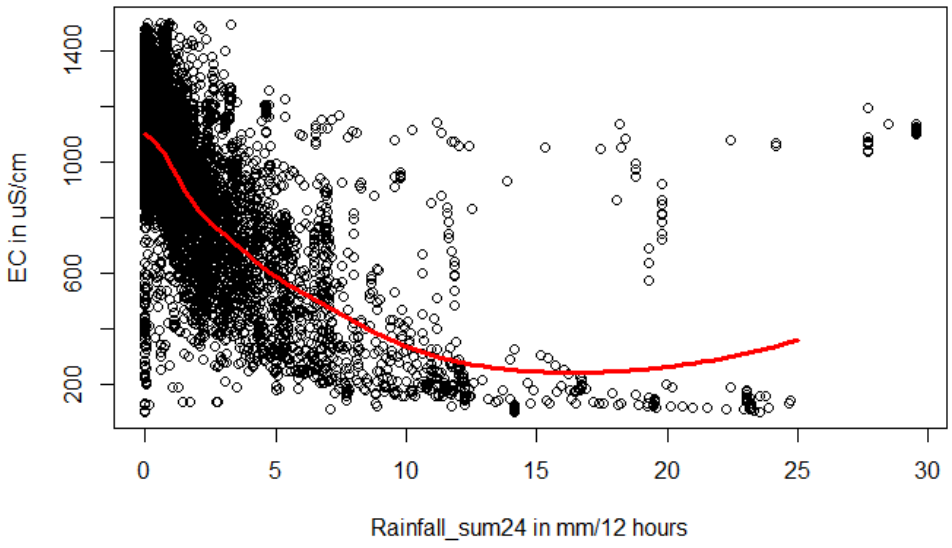

Figure 9: Scatterplot of **EC** against **Rainfall_sum24** and the red line is the estimation of **EC** based on 'mod3' for a constant temperature of 10 degrees.

5 Discussion

In the beginning of this project there did not seem to be a relation between historic rainfall and the sewage quality. This was based on a linear regression model which gave an R-squared value of 0.01. We ended this project with a Generalized Additive Model that gave an R-squared value of 0.61, which is a big improvement of the linear regression model. It is important to notice that this model is not optimal, so one should not use this model right away. There are still some things that might improve the model.

One thing is that the variable **Rainfall_sum24** does not take into account that every shower is different. It can be the case that it is raining the whole day and that we measure the same amount of rainfall in 12 hours as we would have measured when there was a very heavy shower for only one hour. Both type of showers have a different impact on the sewage quality, while the model does not see this difference. It might be possible to solve this by looking at the ratio of the amount of rainfall during a shower and the length of the shower, but this is not very easy to add as a predictor variable to the model.

Furthermore, we could have taken the water level in the sewer system as a predictor variable in the model, but these measurements were not in the right format to add them to the data set. This might have helped improving the model, because there is a relation expected between rainfall and the water level in the sewer system. Then the model might have found a relation between the water level in the sewer system and the sewage quality.

6 Conclusion

6.1 Summary

In this report we have tried to find a relation between historic rainfall and sewage quality for the location Leeuwarden, Bilgaard. We have used the data measured by an EC sensor which was placed in the sewer system. The EC sensor measured conductivity in the sewage according to the amount of salt in the water. In periods of long term rain the sewage is diluted by the rain water and therefore contains less salt which leads to a decrease in the EC measurements. It was expected that there is a relation between historic rainfall and the sewage quality. In this report we have tried to find such a relation using Generalized Additive Models (GAMs) in R. This is a non-linear statistical black box approach in which we have tried to find a model that gives the best approximation of the EC measurements.

We have started the project with some data preparation in which we have analysed the data and have added some new variables that might be helpful for finding a relation between historic rainfall and sewage quality. For example, we have added the variable **Rainfall_sum24** that included historic rainfall in the model. With this variable we have captured some of the delay between rainfall and EC. This variable appeared to be very significant and gave us an R-squared value of 0.59. We also added temperature to the model and took the log of the EC measurements. This gave us an R-squared value of 0.61. This is the model ('mod3') that gave the best approximation of the EC.

We have also tried adding a random effect per day via the variable **days_factor** and this gave an R-squared value of 0.8. We do not know why this model gave such a high R-squared value, because there is no theoretical explanation. This means that the EC can be predicted partly from the residuals and that means that this model was not correct. We were not able to find another variable that might have captured this information that is hidden in the residuals.

In the discussion in Section 5 we have said that 'mod3' was not optimal. It is not a model that can be used directly in practice. It does not take into account that each shower is different. Furthermore, it can be the case that there is another variable missing that captures some of the delay in the data.

The research question was 'is there a relation between historic rainfall and sewage quality for the location Leeuwarden, Bilgaard?' We can answer this question with 'yes there is a relation of 61%, but there is room for optimization in the model we found.' Since every shower has its own characteristics, one can expect that every response on the EC sensor is different.

6.2 Advice

For future data analysis, it would be important to collect more data. The more measurements one has, the more accurate the variable **EC** can be predicted. For this project we had 9 months of useful measurements, but if we would have had at least two years of data we could have tried to compare seasons or months with each other. This might have helped in finding a better relation between historic rainfall and the sewage quality.

Furthermore, it would be a good idea to collect water level data of the sewer system and get it in a good format to work with. It is expected that historic rainfall and the water level in the sewer system are closely related. If this relation can be shown to be one-to-one via a good model, then one can try to find a model that shows a relation between the water level in the sewer system and the sewage quality, instead of a relation between historic rainfall and the sewage quality.

The water level in the sewer system could also be added as a predictor variable to the model 'mod3'. It is expected that there is a relation between historic rainfall and the water level in the sewer system, so this might explain some information that is not captured by rainfall.

Another question that can be explored in the future is whether there is a relation between the turbidity sensor and the quality of the sewage. This might give a higher R-squared value then with the EC sensor.

Finally, it is expected that the relation between historic rainfall and the sewage quality is location dependent. That is, for each sewer system we probably get a different relation. It would be a good test to measure with an EC sensor at several locations and apply 'mod3' to the data obtained. If it appears to be the case that the R-squared value remains high for these other locations, one can conclude that we have found a good model, that can be applied on other locations besides Leeuwarden, Bilgaard.

References

[1] Overfitting. http://blog.minitab.com/blog/adventures-in-statistics/the-danger-of-overfitting-regression-models. Accessed: 13-10-2016.

[2] Residuals. http://blog.minitab.com/blog/adventures-in-statistics/why-you-need-to-check-your-residual-plots-for-regression-analysis. Accessed: 26-10-2016.

[3] Temperature data Leeuwarden (KNMI). http://knmi.nl/nederland-nu/klimatologie/uurgegevens. Accessed: 30-09-2016.

[4] P. Breheny and W. Burchett. *visreg: Visualization of Regression Models*, 2016. R package version 2.3-0.

[5] G.E.P. Box & D.R. Cox. An analysis of transformations. *Journal of the Royal Statistical Society. Series B*, 26:211–252, 1964.

[6] C. Gu. *Smoothing Spline ANOVA Models*. Springer-Verlag, 2002.

[7] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013.

[8] T.J. Hastie & R.J. Tibshirani. *Generalized Additive Models*. Chapman and Hall, 1990.

[9] J. van Rij, M. Wieling, R. H. Baayen, and H. van Rijn. itsadug: Interpreting time series and autocorrelated data using gamms, 2016. R package version 2.2.

[10] J. Verzani. *Using R for Introductory Statistics*. Chapman & Hall/CRC Press, 2005.

[11] H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2009.

[12] S. N. Wood. Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society (B)*, 73(1):3–36, 2011.

Appendices

A Script with overview of models

```
#Install these packages first and then run them each time
   you work in R
library(mgcv);library(itsadug);library(MASS);library(stats)
   ;library(visreg)

#---------- Model 1 ---------#

mod1 <- bam(EC ~ s(Rainfall, bs = "cr", k=10),
            data = dataleeuwarden, method = "REML",
            subset = EC<1500)

# Summary of model
summary(mod1)
# Check if basis dimensions (k) are good
gam.check(mod1)
# Model diagnostics
diagnostics(mod1)
# Model plot for each variable
plot(mod1,shade = T)
# Save the model
save(mod1,file = "mod1.rda")

#---------- Model 2 ---------#

mod2a <- bam(sqrt(EC) ~ s(Rainfall_sum24, bs = "cr", k=10),
            data = dataleeuwarden, method = "REML",
            subset = EGV<1500)

# First value of autocorrelation
AC <- start_value_rho(mod2a,plot = T)

# The same model as mod2a, but with autocorrelation added
mod2b <- bam(sqrt(EC) ~ s(Rainfall_sum24, k=10),
             data = dataleeuwarden, method = "REML",
             subset = EGV<1500, rho=AC,
             AR.start =dataleeuwarden$AR)

summary(mod2b)
gam.check(mod2b)
```

```
diagnostics(mod2b)
plot(mod2b,shade = T)
save(mod2b,file = "mod2.rda")

#---------- Model 3 ---------#

mod3a ← bam(log(EC) ~ s(Rainfall_sum24, k=10)
            + s(avg.temp, k=10),
            data = dataleeuwarden, method = "REML",
            subset = EC<1500)

AC1 ← start_value_rho(mod3a,plot = T)

mod3b ← bam(log(EC) ~ s(Rainfall_sum24, k=10)
            + s(avg.temp, k=10),
            data = dataleeuwarden, method = "REML",
            subset = EGV<1500, rho=AC1,
            AR.start =dataleeuwarden$AR)

summary(mod3b)
gam.check(mod3b)
diagnostics(mod3b)
plot(mod3b,shade = T)
save(mod3b,file = "mod3.rda")

#---------- Model 4 ---------#

mod4a ← bam(sqrt(EC) ~ s(Rainfall_sum24, k=10)
            + s(days_factor, bs = "re"),
            data = dataleeuwarden, method = "REML",
            subset = EC<1500)

AC2 ← start_value_rho(mod4a,plot = T)

mod4b ← bam(log(EC) ~ s(Rainfall_sum24, k=10)
            + s(days_factor, bs = "re"),
            data = dataleeuwarden, method = "REML",
            subset = EGV<1500, rho=AC2,
            AR.start =dataleeuwarden$AR)

summary(mod4b)
gam.check(mod4b)
diagnostics(mod4b)
```

```
plot(mod4b,shade = T)
save(mod4b,file = "mod4.rda")

#---------- Model 5 ---------#

mod5a ← bam(sqrt(EC) ~ s(Rainfall_sum24, k=10)
             + s(tm, k=50),
             data = dataleeuwarden, method = "REML",
             subset = EC<1500)

AC3 ← start_value_rho(mod5a,plot = T)

mod5b ← bam(log(EC) ~ s(Rainfall_sum24, k=10)
             + s(tm, k=50),
             data= dataleeuwarden, method = "REML",
             subset = EGV<1500, rho=AC3,
             AR.start =dataleeuwarden$AR)

summary(mod5b)
gam.check(mod5b)
diagnostics(mod5b)
plot(mod5b,shade = T)
save(mod5b,file = "mod5.rda")

#---------- Model 6 ---------#
# Extra model where we added several Rainfall_sums. The
   R-squared value is high, but interpretation becomes vague

mod6a ← bam(log(EC)~ s(avg.temp, k=10)
             + s(Rainfall_sum24, k=10)
             + s(Rainfall_sum12, k=10)
             + s(Rainfall_sum48, k=10)
             + s(Rainfall_sum36, k=10)
             + s(Rainfall_sum96, k=10)
             + s(Rainfall_sum144, k=10)
             + s(Rainfall_sum192, k=10)
             + s(Rainfall_sum240, k=10)
             + s(Rainfall_sum288, k=10)
             + s(Rainfall_sum336, k=10),
             data = dataleeuwarden, method = "REML",
             subset = (EGV<1500))

AC4 ← start_value_rho(mod6a,plot = T)
```

```
mod6b ← bam(log(EC) ~ s(avg.temp, k=10)
              + s(Rainfall_sum24, k=10)
              + s(Rainfall_sum12, k=10)
              + s(Rainfall_sum48, k=10)
              + s(Rainfall_sum36, k=10)
              + s(Rainfall_sum96, k=10)
              + s(Rainfall_sum144, k=10)
              + s(Rainfall_sum192, k=10)
              + s(Rainfall_sum240, k=10)
              + s(Rainfall_sum288, k=10)
              + s(Rainfall_sum336, k=10),
              data = dataleeuwarden, method = "REML",
              subset = (EGV<1500), rho=AC4,
              AR.start =dataleeuwarden$AR)

summary(mod6b)
gam.check(mod6b)
diagnostics(mod6b)
plot(mod6b, shade=T)
save(mod6b,file = "mod6.rda")
```

models.R

B Histograms of EC and Rainfall

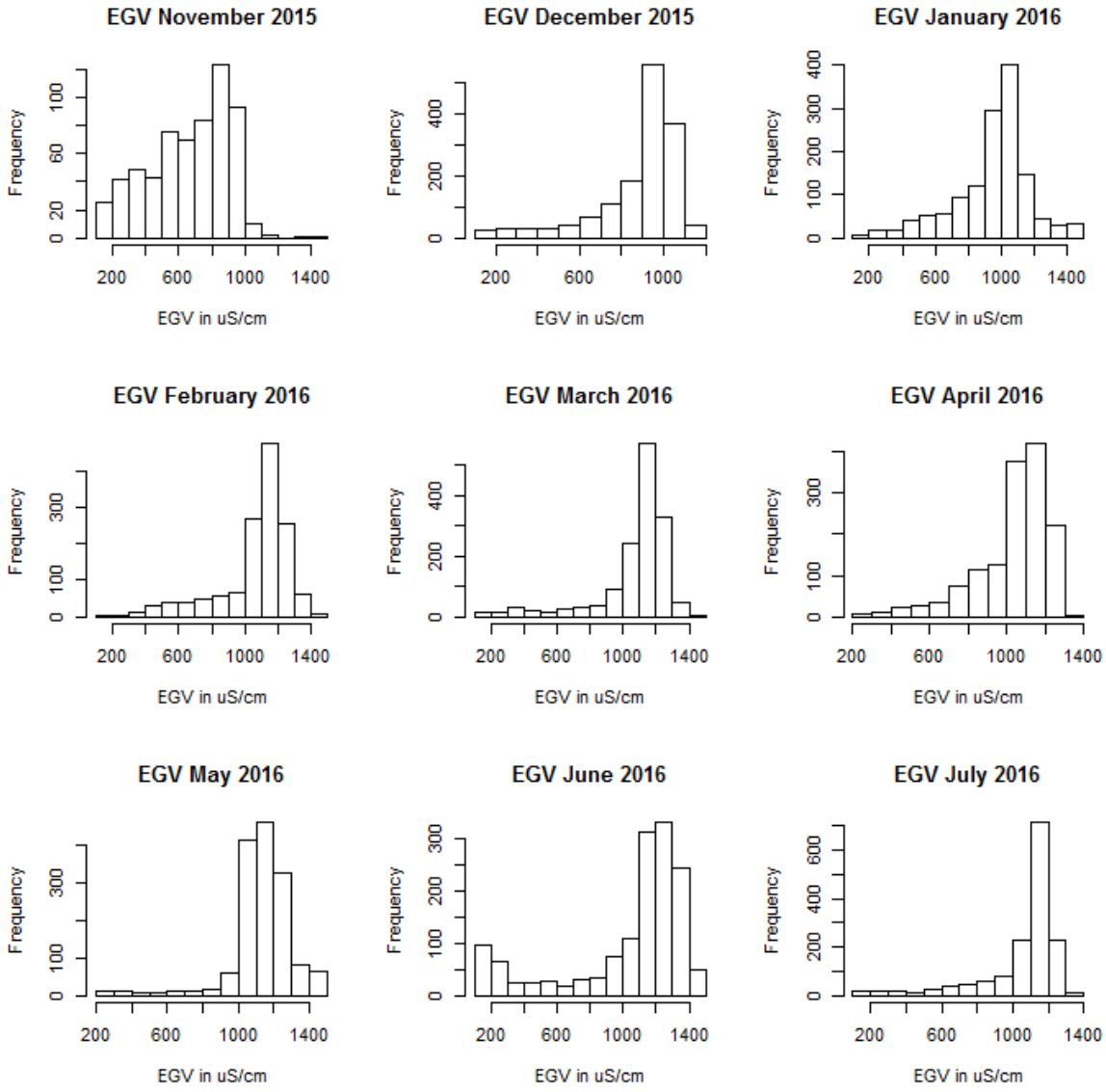

Figure 10: Histograms of EC per month

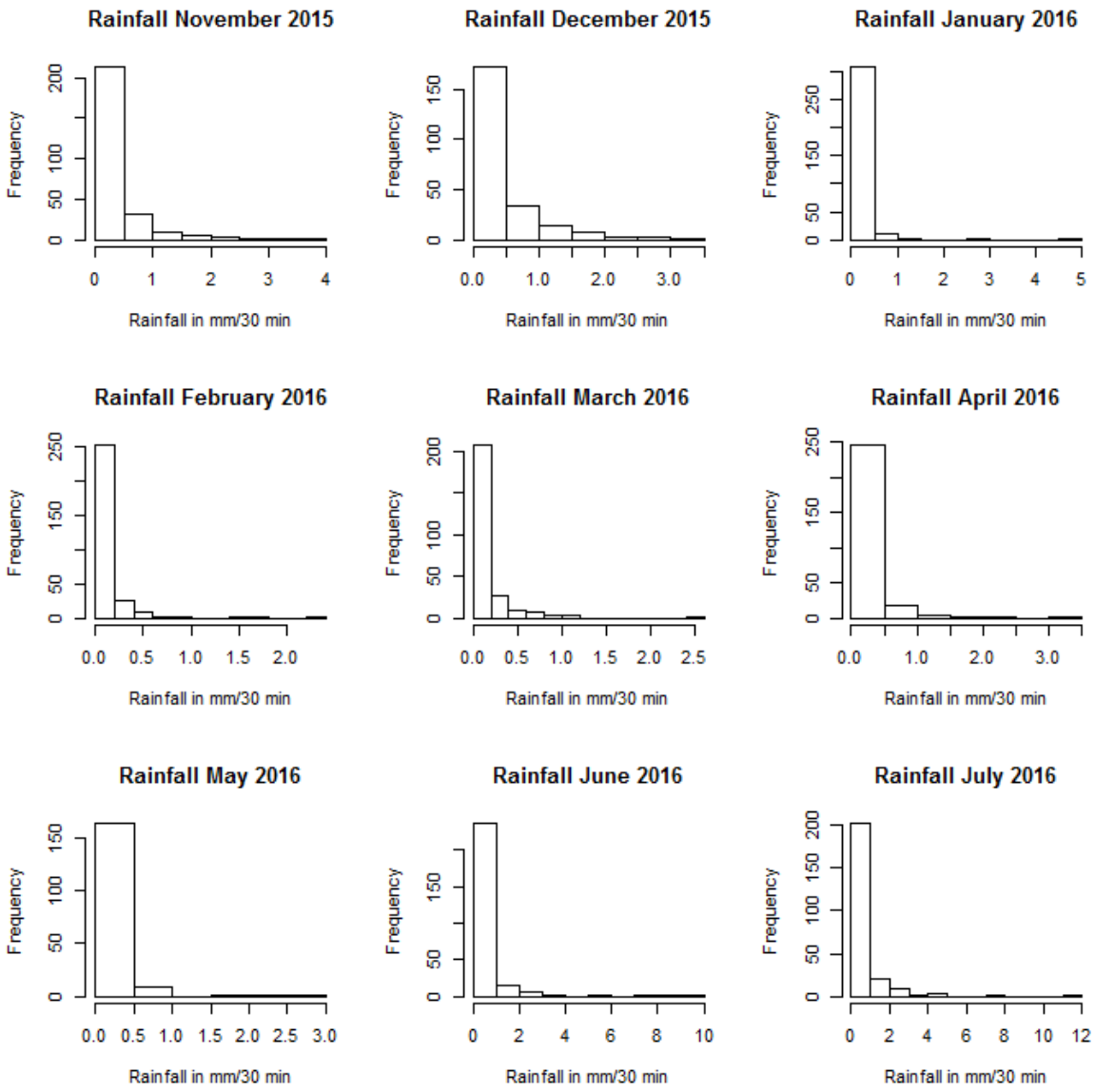

Figure 11: Histograms of Rainfall per month