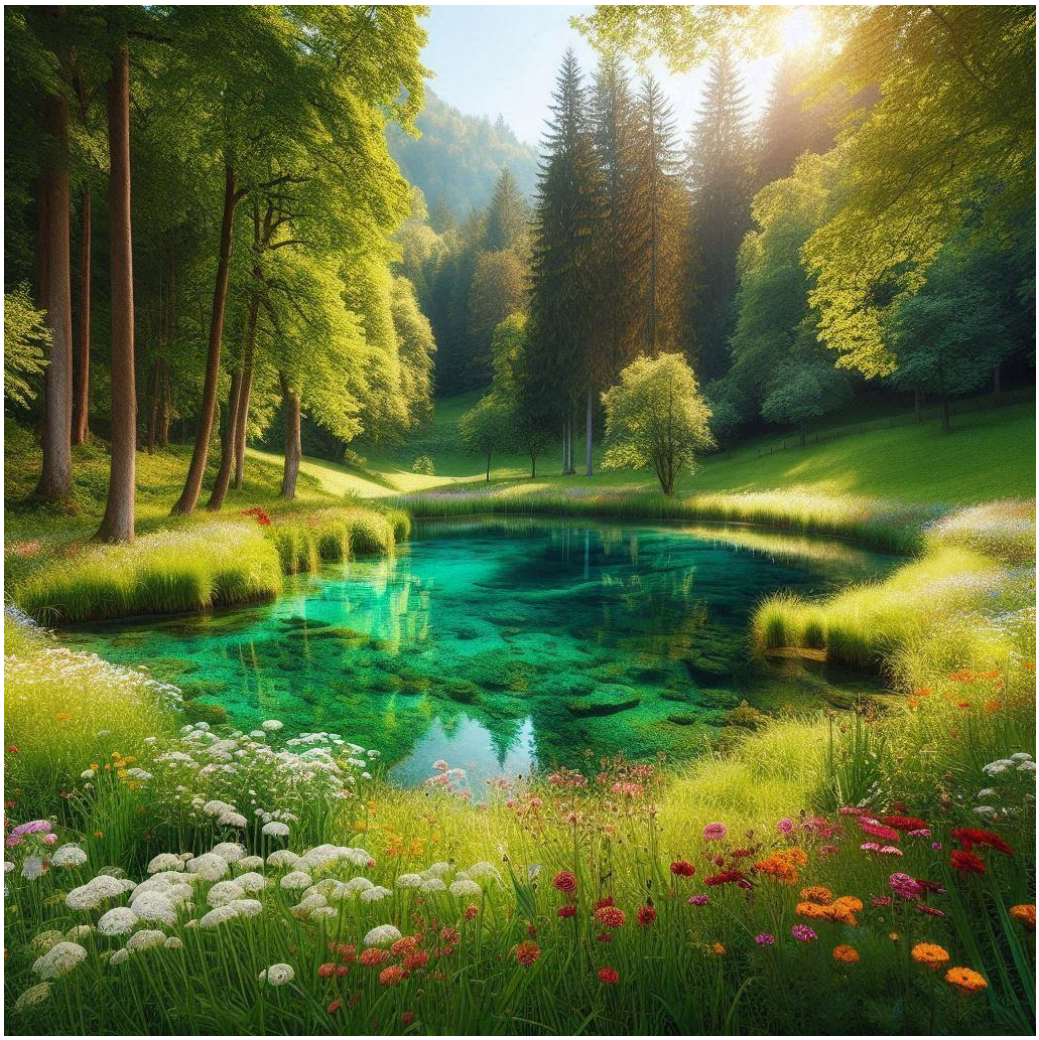
Analyzing temporal trends in data series variables with values below reporting limits – a case study of groundwater Pb concentrations



Claudia von Brömssen and Erik Olsson

SLU, Department of Energy and Technology

Report / Department of Energy and Technology, SLU
Report nr: 129
ISSN: 1654-9406

Department of Energy and Technology, SLU
Box 7032
750 07 Uppsala

https://www.slu.se/en/departments/energy-technology/

Cover picture: The front cover is created by AI using Bing Image Creator, https://www.bing.com/images/create?FORM=GDPGLP, Prompt: a ground water body

Uppsala, January 2025

Preface

This report is partly financed by a project of the Swedish agency of marine and water management Nr 2024-000836, ”Statistikstöd under revision av programområde Sötvatten”.

Contents

1. Introduction

Values below reporting limit[1] (RL) are common when levels of the observed variable are small. In order to allow handling of such values it is often chosen to substitute values with a fixed value, typically half of the used limit. If the goal is to use the data further in statistical models, this procedure is satisfactory, as long as the number of values below a limit is small. There are, however, a few situations where it would be advantageous to retain more of the available information in the data. This is true especially if the RL changes, which is common in environmental monitoring when chemical analysis instruments or procedures are improved over time. An improved strategy for analysing such data is also valuable if a large percentage of values lie below the limit.

Different types of censored regression has been suggested to handle data under a RL. In these models, observations below the RL are not substituted with a constant, but instead the information of the limit itself and the proportion of observations below this limit are used in the calculation of the model estimates.

In this report we describe different ways to handle values below a reporting limit, which are readily available in R (R Core Team, 2024), give examples of R code and describe the procedure of analysing time series of Pb in groundwater. Values under the RL will also be called censored values.

2. Handling data with reporting limit

The reporting limit is the lowest concentration that can be reported with a satisfactory level of accuracy. Observations that lie below such reporting limits are handled in varying ways in databases. A common way is to denote them with a “less than” sign, e.g <0.01, giving information both about the RL and that the value observed is below that. Other codings that are used are to indicate the value as a negative, when negative values are not possible, e.g. -0.01, or to use a secondary data column, where the first one contains either the observation or the RL and the second column indicates if the value is censored or not.

2.1 Substitution

A much-used way to statistically handle a smaller amount of values below a RL is to substitute them with a fixed numeric value. The most common choice is to use half of the RL as a substitute, but other option like the RL itself or zero are also sometimes used. After substitution data is handled as if there are no censored values present (Section 3.2). Substitution can introduce bias in the produced estimates, especially if the proportion of values below the RL is high.

2.2 Incorporating reporting limits in statistical methods

Several possibilities exist to use censored values directly in the statistical analysis. Nonparametric methods are commonly used to estimate means or medians (Wood et al., 2011) and trends (section 3.1). Since the methods are rank based the choice of substitution does not matter for the outcome as long as the RL is constant over time.

Another approach available is to use methods based on maximum-likelihood estimations, where the likelihood is composite, including the information from uncensored data and the proportion and limit of censored data (Helsel, 2011; Section 3.3).

[1] Here, we assume a limit of quantification (LOQ) and denote it reporting limit (RL). Statistically, limits of detection (LOD) can be handled in the same way as can other types of left-censored data.

2.3 Multiple reporting limits

Data with multiple RL needs to be handled especially careful. When data points need to be substituted a common RL must be determined for the entire data set. For this, typically, the highest RL is chosen. Multiple RLs are especially important to handle in trend analysis since improvement in chemical analysis methods can lead to lower RL over time and, thus, can introduce an artificial trend if not handled correctly.

As an example, we simulated a series with no trend and introduced multiple RL (concentration 4 until time point 14 and 3 after that, Figure 1). Using the RL (or half of RL) values as substitute would easily lead to a significant result of a trend test (e.g. Mann-Kendall test gives a p-value of 0.02). Instead all values below 4 should be censored for the entire series. In that case statistical trend test would give more reliable results (Mann-Kendall p-value of 0.14), but also leaves us with less information as a large part of the data is now censored (17 of 30 observations) and the improved accuracy of observed data after time point 14 cannot be taken advantage of.

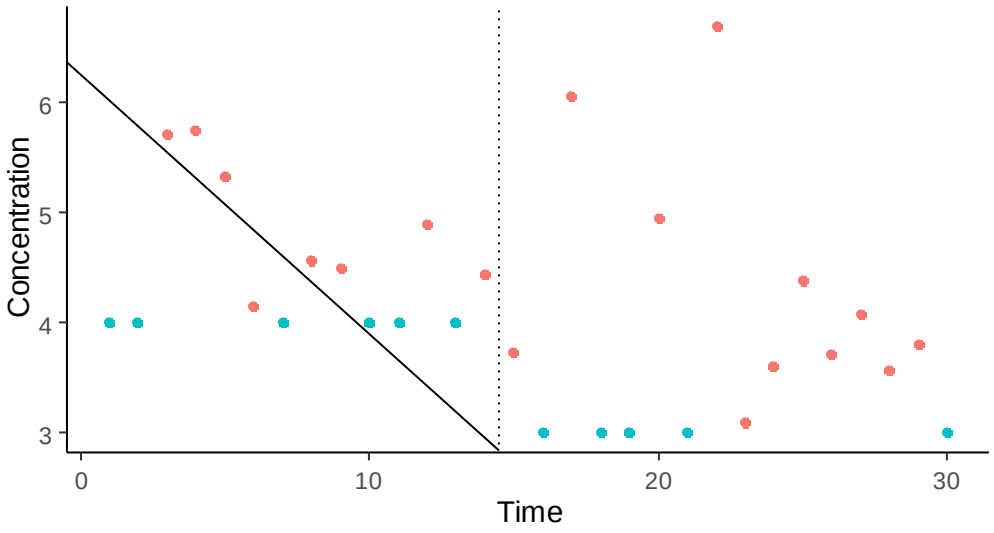


Figure 1: A simulated time series. The reporting limit is 4 for the first 14 observations and then changes to 3. The series contains no trend, but the changing RL can introduce an artificial trend if not handled correctly. Data points in blue are censored at the RL, while data points in red are not censored.

3. Trend analysis for data that include values below a reporting limit

3.1 Nonparametric trend tests

A common trend detection method in environmental data are Mann-Kendall tests (Hirsch et al., 1982; Kendall and Gibbons, 1990; Mann, 1945). These tests are based on ranks of data, which allows the inclusion of values under a single RL by attributing them the lowest rank. This will result in a number of tied observations (observations with the same value), for which the variance of the Mann-Kendall test routinely is adjusted for. If datasets contain multiple RL data needs to be adjusted to

contain only a single RL. There are several packages in R that compute Mann-Kendall tests (Appendix B1).

The size of any observed trend can be computed using the Theil-Sen slope. This slope computation is not directly related to the Mann-Kendall test, but the methods are often used together as both are based on ranks. The Theil-Sen slope represents the median change per time unit.

3.2 Regression and GAM on substituted values

Regression methods can be used to analyze trends when values under a reporting limit are substituted. If series are relatively short or have sparse data it is often chosen to let the trend be linear or exponential (by log-transforming the response). For this, simple linear regression can be used by including time as an explanatory variable. For longer series it is usually not recommended to assume linearity for the temporal trend. Instead, a data-driven trend fitted by a smooth function can be used to describe the development in time. This is usually implemented in a general additive model (GAM). Examples of linear regression and GAMs for trend analysis after substitution are given in Appendix B.2.

3.3 Censored regression and GAM

Specialized models for censored data can account for different types of censoring direction. Typically we distinguish between left-censored, right-censored and interval-censored data. In environmental monitoring, measurements are typically censored due to limitations of chemical analysis instruments, i.e. how low concentrations can be determined with satisfying accuracy. Such measurements are considered either left-censored (less than RL) or interval censored (between zero and RL). We will not discuss right-censored values (higher than a limit) further.

Censored data can be included into the statistical analysis by specialized estimation based on maximum-likelihood methods (Appendix A). For each observation, information is provided if the observation is censored or not, and what the level of RL is. Alternatively, for interval censored data the information is given in two variables containing lower and upper limit, while non-censored data are presented as the actual observed value in both columns. The maximum likelihood approach includes this given information in the estimation of the model parameters. Several functions allow the implementation of censored data in regression and GAMs, of which some only allow a single censoring level (Appendix B.3) and some allow multiple levels (Appendix B.4). Not all methods allow interval-censored data.

4. Case study of trend analysis of metals in groundwater

Data from trend stations in the national environmental groundwater monitoring program were obtained by SGU. For this analysis we chose to study the parameter Pb, with data from 1996-2023. While the coding of the values below RL was ambiguous in the database, it is deemed very likely that a RL of 0.02 was used until 2012 and a RL of 0.01 after that. Observations below RL are common between 2003 and 2010, but also in the end of the series, which in itself is a sign of downward trend (Figure 2). Data below the RL are considered interval-censored, as concentrations of Pb cannot be lower than zero.

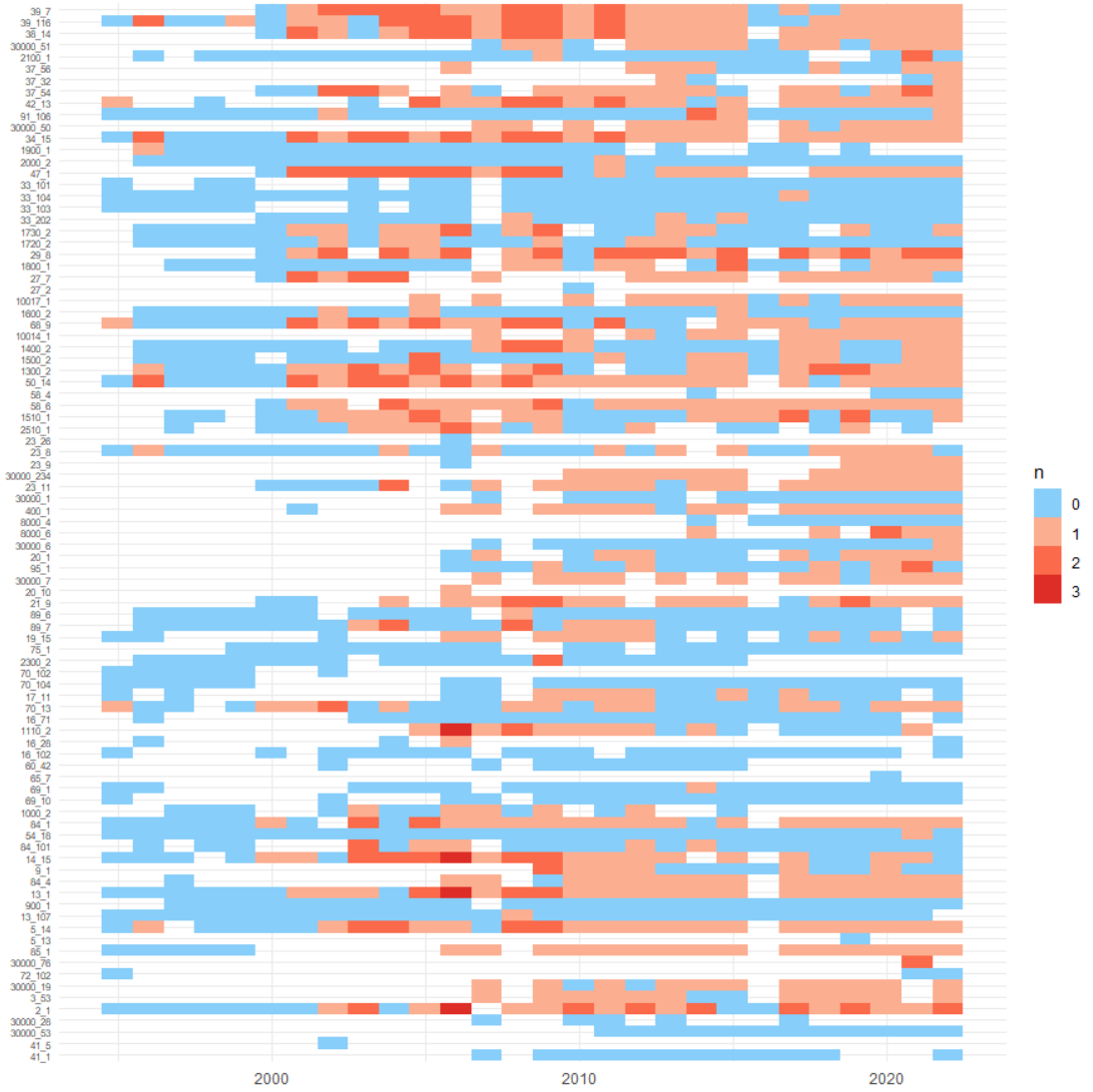


Figure 2: Number of values below reporting limit for Pb at stations within the trend program. Stations are sorted north to south.

4.1 Analyzing a single station

To show the analysis for a single station object 84_1 is chosen. A large number of observations, 20 of 37, are censored (Figure 3), especially during the second half of the series. Additionally, since values approach very low concentrations at the end of the series also the variation in the data must be assumed to be smaller than in the beginning of the series, i.e. the assumption of equal variances that we usually rely on in regression models is violated. A common way to handle this is to log-transform the response variable.

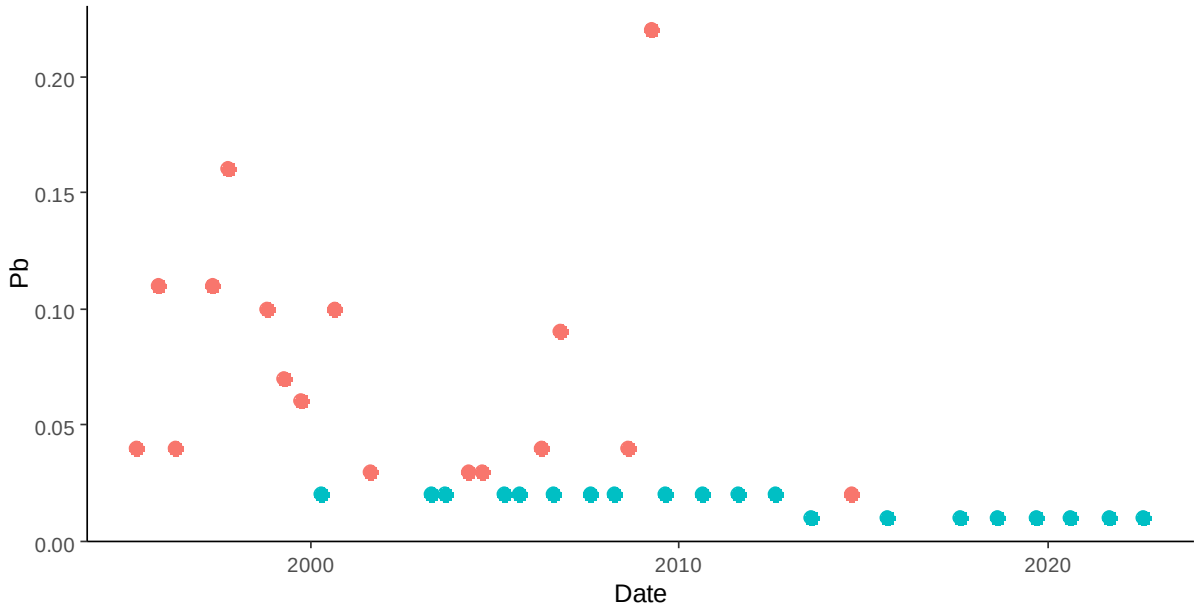


Figure 3: Pb at station 84_1. Red dots indicate quantified values, blue dots values below the reporting limit (with the dot at the reporting limit), which changed from 0.02 to 0.01 in 2013.

For environmental time series it is also common to assume that observed trends are not linear over time, especially if the observed time period is fairly long. The combination of non-linear trends and interval-censored data with multiple reporting limits makes GAMs with a “cnorm”-distribution most appropriate and will be compared to a GAM based on substituted values.

When substituting values below RL we use a common RL of 0.02 for the entire series which means that all data are assumed to have a RL of 0.02, even data points after 2013 that in fact have a lower RL. This is important to not introduce an artificial shift in the series. In this case, since the values are generally lower later in the series we can substitute with the value of RL in order to not overestimate the prevailing trend or use the more common approach of substituting with half of the RL. A regular GAM is then fitted to the data, using a thin plate smooth for the time trend, and using a normal (Gaussian) distribution with a log-link to log-transform the expected value of the response variable in the model. The log-transform of the response both improves data distribution by decreasing the skewness of the model residuals and preserves the data property that no observations can be lower than zero.

For the GAM model including interval-censored data the response variable is first prepared as two entities, one containing the lower and one the upper limit. For this approach we do not need to assume a common RL, but we can accommodate the two different RL in the series by specifying the corresponding intervals. If both values are the same the observation is considered uncensored. If the values are different they determine the lower and upper limit of the censored value and are then chosen as zero for the lower limit and the value of RL at that time point as the upper limit. The time trend component in the model is again a thin plate spline based on the date variable.

As comparison we also code data as left-censored to investigate if this would lead to a different fit, i.e. no lower limit is given. For this the first column contains either the observed value or the RL and the second column entry is the observed value for uncensored data. For a censored value the second

column is given as `-Inf`, meaning that the lower limit is negative and infinite, since no lower limit can be specified.

The two models with substitution and the model using interval-censored data provide quite similar results (Figure 4). As expected the substitution with the value of the highest reporting limit (blue) leads to a less steep trend compared to the series where observations are substituted by half of the RL, as the values at the end of the series probably are not well represented. The model using the interval-censored data leads to a slightly steeper trend curve (black). The model using left-censoring as coding for the censored observations exhibits more curvature (green). From about 2005 the predicted values are very close to zero, which must be interpreted as an underestimation.

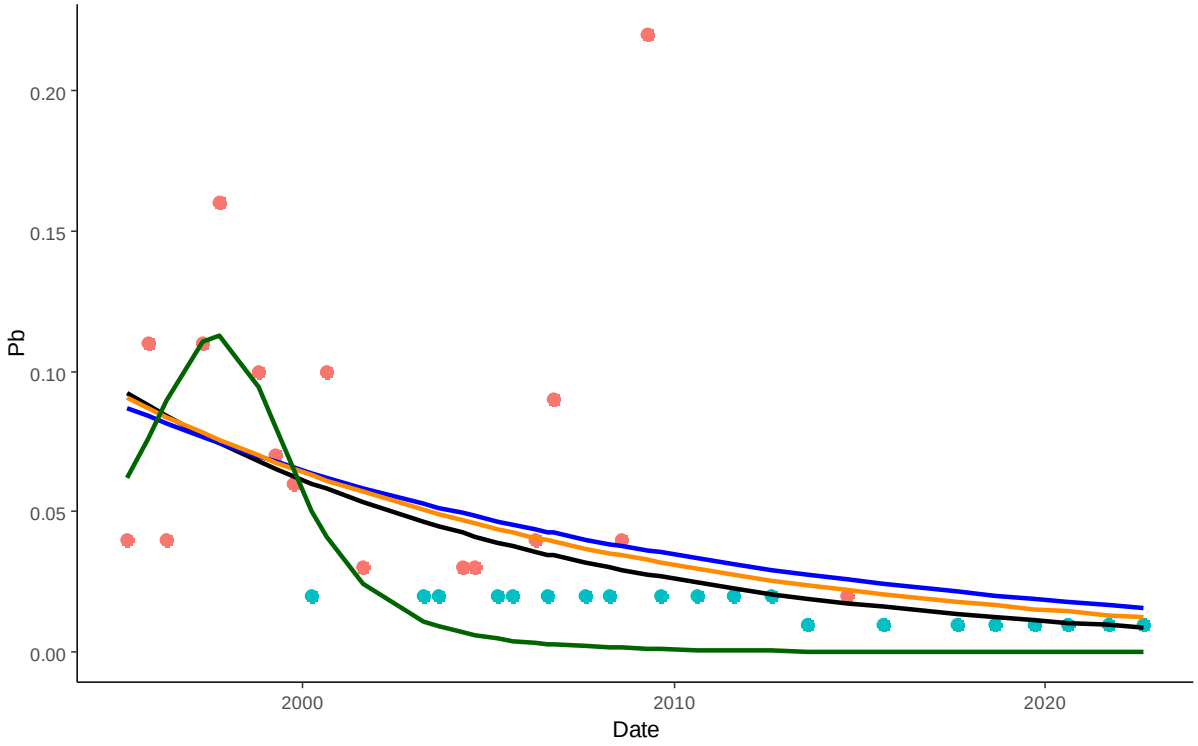


Figure 4: Back-transformed fitted trend curves for four models: Substitution with reporting limit (blue), half of the reporting limit (orange), using interval-censored data (black) and left-censored data (green).

4.2 Analyzing several stations

When analyzing several stations we need to filter out series that carry little information and could make model fitting slow and even lead to convergence problems. Here we select only stations that have no more than 70% of the available data censored and that have at least 15 observations. Usually there are 1-2 observations per year, which means that the series selected have observations during at least 7 years. This leads to 45 stations to be analyzed using substituted data and 47 stations using interval-censored data. The discrepancy stems from that observations at 0.02 are considered censored over the entire time period for substituted data, but only up until 2013 for interval-censored data. For substitution we choose the RL as representing value. Two of the fitted models including interval-censored data did not converge, leading to 45 series analyzed with both methods.

Principles described by von Brömssen et al. (2021) are adapted to be able to handle interval-censored data (Appendix C). Generally, for time series models an autocorrelated error term should

be included in the model to adjust for the fact that observations are not independent. However, using a “cnorm”- distribution to account for the censored data does not allow such an autocorrelation estimate.

In the automated analysis for several stations some general strategies are used to allow a standardized analysis. One of these is that the maximum complexity of the models were set to a maximum. In GAM the maximum complexity can be controlled using the model parameter k, which is here set to the number of observations divided by eight. Additionally, using restricted maximium likelihood (REML) estimation in the original models was replaced by the default optimization method implemented in the mgcv package (GCV.Cp).

The obtained plots indicating periods of increasing and decreasing levels for each station and are sorted from north to south. The results for substituted data (Figure 5) and interval-censored data (Figure 6) are quite similar and the list of analyzed stations overlap to great extent. However, station 5_14 lead to a fitted model only with the substitution method and station 19_15 was only analyzed using interval-censored data due to a large amount of censored data. For both these stations the fit was not satisfactory (not shown), which indicates that the models do not work well with series in which a high percentage of observations is censored. Therefore, the inclusion criteria should be more restrictive.

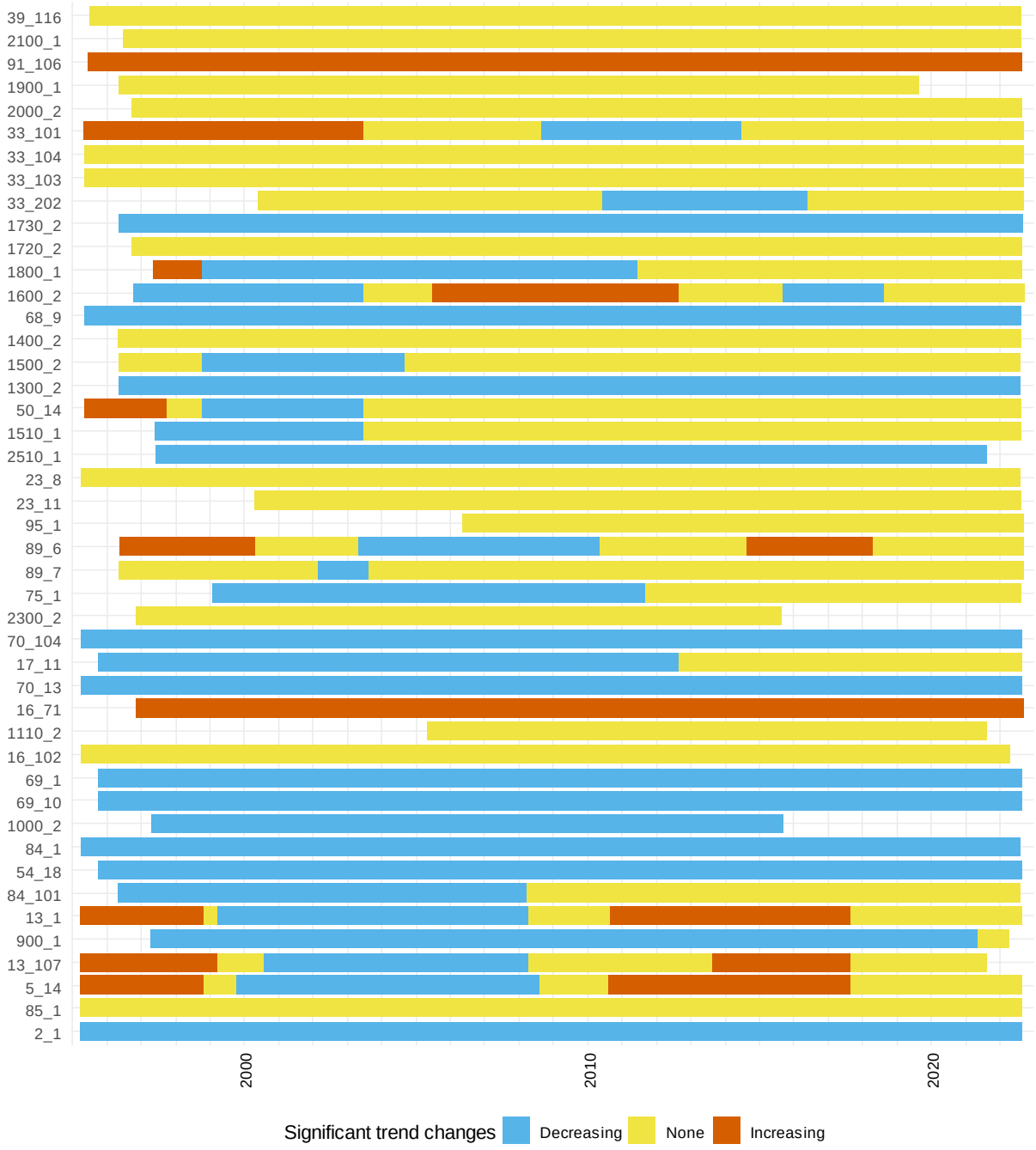


Figure 5: Trends in Pb at groundwater stations observed between 1995 and 2022. Values below the reporting limit are substituted by the highest reporting limit. Blue indicates significant downward trends, red significant upward trends and yellow no significant trends.

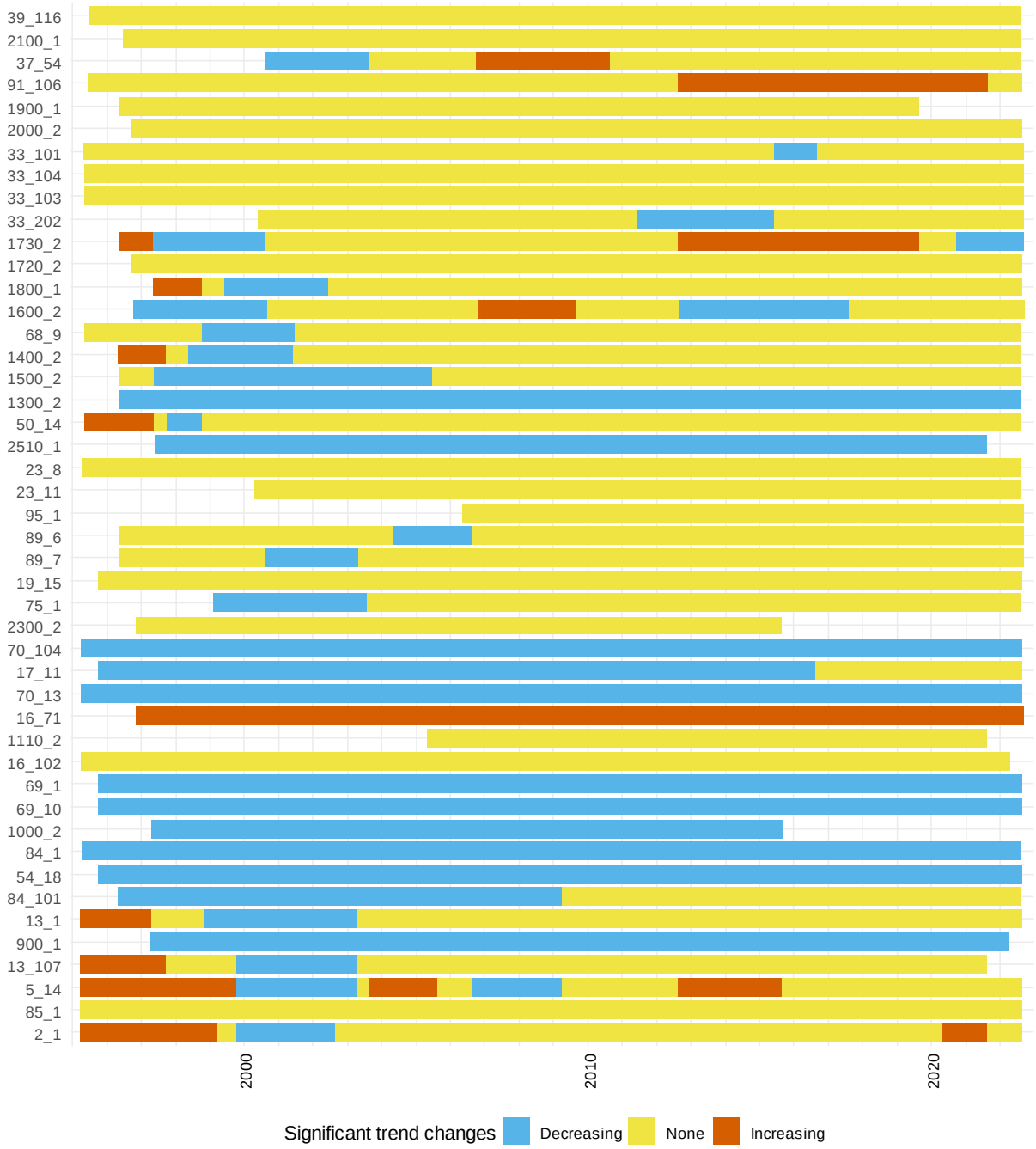


Figure 6: Trends in Pb at groundwater stations observed between 1995 and 2022. Values below the reporting limit are included as interval-censored data. Blue indicates significant downward trends, red significant upward trends and yellow no significant trends.

In this type of analysis it is important to check the individual model fits, as the modelling is very dependent on dataset characteristics, such as outliers, missing data and sudden large changes in concentration. Several of the stations exhibit a nonsensical model fit and mostly both methods fail for these cases. There are especially two types of series that exhibit problems: (i) series that have many values below the RL, especially if they arise early or late in the series, e.g. station 13_1 (Figure 7, Figure D.1) and (ii) series that have single extreme outliers, which are often identified by a red period directly followed by a blue one. An example is station 1800_1 (Figure 7). Increasing the demand on non-censored observation to be at least 50% decreases the number of problematic

model fits, but problems for series with outliers can be observed even with stronger inclusion criteria.

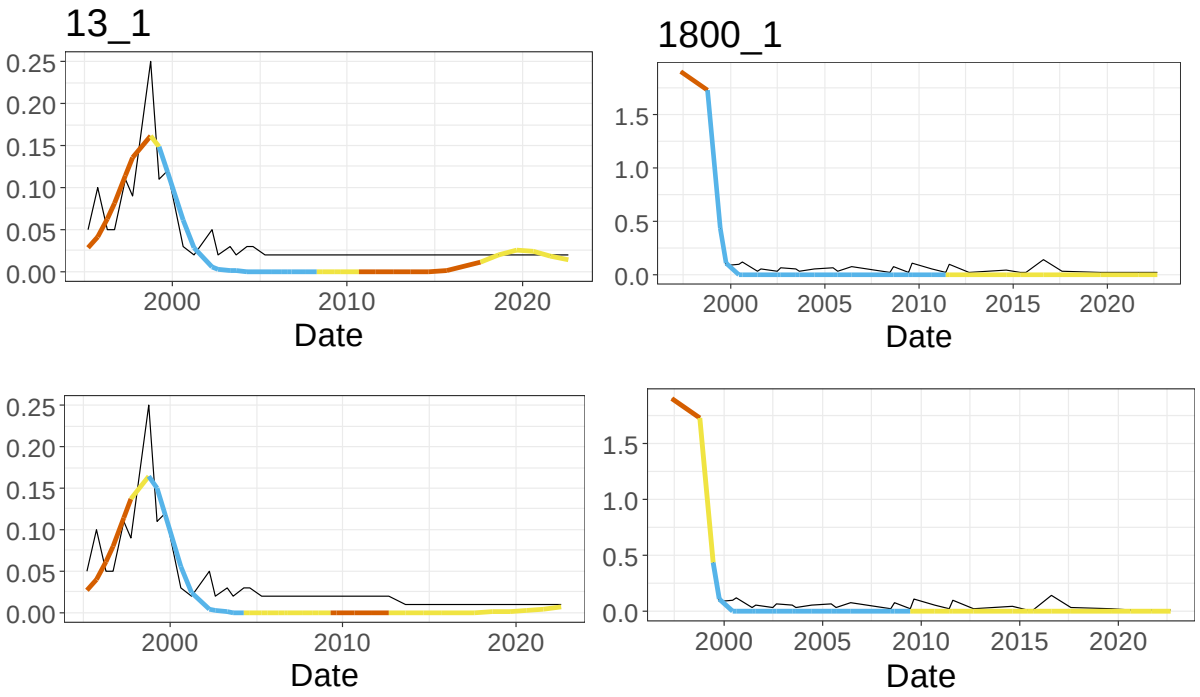


Figure 7: Pb at station 13_1 and 1800_1 together with the fit of the series implementing a GAM with substitution (top) and a GAM based on interval-censored data (bottom).

5. Conclusion

Several different approaches are possible to handle values below a RL. A multitude of R packages provide solutions for specific cases. For the analysis of environmental monitoring data, especially concentration data are usually assumed to be left-censored or interval-censored. For our dataset we found that left-censoring did not work well to describe the trend in Pb when data in fact was interval-censored with many values below the reporting limit. Therefore, we did not follow-up on the left-censoring case.

In our approach, we choose GAM implemented in the mgcv package as we want to include interval-censored data, while allowing the temporal trend to be data-driven. Substitution of data both with the highest RL and half of the RL gave very similar results. For both model types it is, however, necessary to study the obtained model fits in detail, since models were very sensitive to (i) the number of values below RL, (ii) when these values are observed, especially if they are clustered in one end of the series, which is common if trends are present, and (iii) the presence of outliers or very fast drops in level that make trend modelling more difficult in general.

General recommendations from this small case study is (i) to apply trend models only to sites that have at least 50% non-censored data and (ii) to always plot individual model results to verify that the obtained model is meaningful.

For Pb in groundwater in Sweden we found that negative trends prevail in Southern Sweden.

6. References:

Helsel, D.R., 2011. Regression and Trends, in: Statistics for Censored Environmental Data Using Minitab® and R. John Wiley & Sons, Ltd, pp. 236–267. https://doi.org/10.1002/9781118162729.ch12

Hirsch, R.M., Slack, J.R., Smith, R.A., 1982. Techniques of trend analysis for monthly water quality data. Water Resour. Res. 18, 107–121. https://doi.org/10.1029/WR018i001p00107

Kendall, M., Gibbons, J.D., 1990. Rank Correlation Methods, 5 edition. ed. Oxford University Press, London : New York, NY.

Mann, H.B., 1945. Nonparametric Tests Against Trend. Econometrica 13, 245–259. https://doi.org/10.2307/1907187

R Core Team, 2024. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.

von Brömssen, C., Betnér, S., Fölster, J., Eklöf, K., 2021. A toolbox for visualizing trends in large-scale environmental data. Environ. Model. Softw. 136, 104949. https://doi.org/10.1016/j.envsoft.2020.104949

Wood, M.D., Beresford, N.A., Copplestone, D., 2011. Limit of detection values in data analysis: Do they matter? Radioprotection 46, S85–S90. https://doi.org/10.1051/radiopro/20116728s

Appendix

Appendix A: Censored Regression

The censored normal regression model, first considered by Tobin (1958), is:

$$y_i^* = \beta x_i + u_i \qquad u_i \sim N(0, \sigma^2)$$

The observed y_i are related to y_i^* by:

$$y_i = y_i^* \quad if \;\; y_i^* > y_0$$
$$y_i = y_0 \quad otherwise$$

Where y_0 is a predetermined constant, in the context of this paper y_0 is the reporting limit.

The intercept and the slope for each explanatory variable are fit by maximum likelihood estimation instead of least squares estimation which is commonly used in ordinary linear regression.

This change in estimator is necessary for consistency.

Such a likelihood function can be written as:

$$L(x_i, \delta_i) = \prod p(x_i)^{\delta_i} \cdot F(x_i)^{1-\delta_i}$$

Where x_i is the value of the measurement or the reporting limit and δ_i is an indicator designating whether x_i is censored (0) or uncensored (1). Further $p(x)$ and $F(x)$ is the probability function and cumulative distribution function respectively.

Appendix A: Bibliography

Tobin, 1958. Estimation of Relationships for Limited Dependent Variables. *Econometrica*, vol. 26, no. 1, pp. 24–36. https://doi.org/10.2307/1907382

Helsel, D.R., 2011. Regression and Trends, in: Statistics for Censored Environmental Data Using Minitab® and R. John Wiley & Sons, Ltd, pp. 236–267. https://doi.org/10.1002/9781118162729.ch12

Appendix B: R code for trend analysis when data contain values below a reporting limit

In the following sections a number of approaches using the statistical software R are described. The functions and results are illustrated with the data set also given in the main text. It is simulated according to the following code and shown in Figure B.1:

```
library(tidyverse)

set.seed(23367)
data_original<-data.frame(conc=rnorm(30, 4, 1.2), date=seq(1:30))


data_original%>%
  mutate(conc_1=case_when(date<15 & conc<4 ~4,
                          date>=15 & conc<3~3,
                          TRUE~conc))->data_censored_multiple

data_censored_multiple%>%
  ggplot(aes(y=conc_1, x=date))+
  geom_point()+
  xlab("Time")+
  ylab("Concentration")+
  geom_vline(xintercept=14.5, lwd=0.5, lty=3)+
  theme_classic()
```

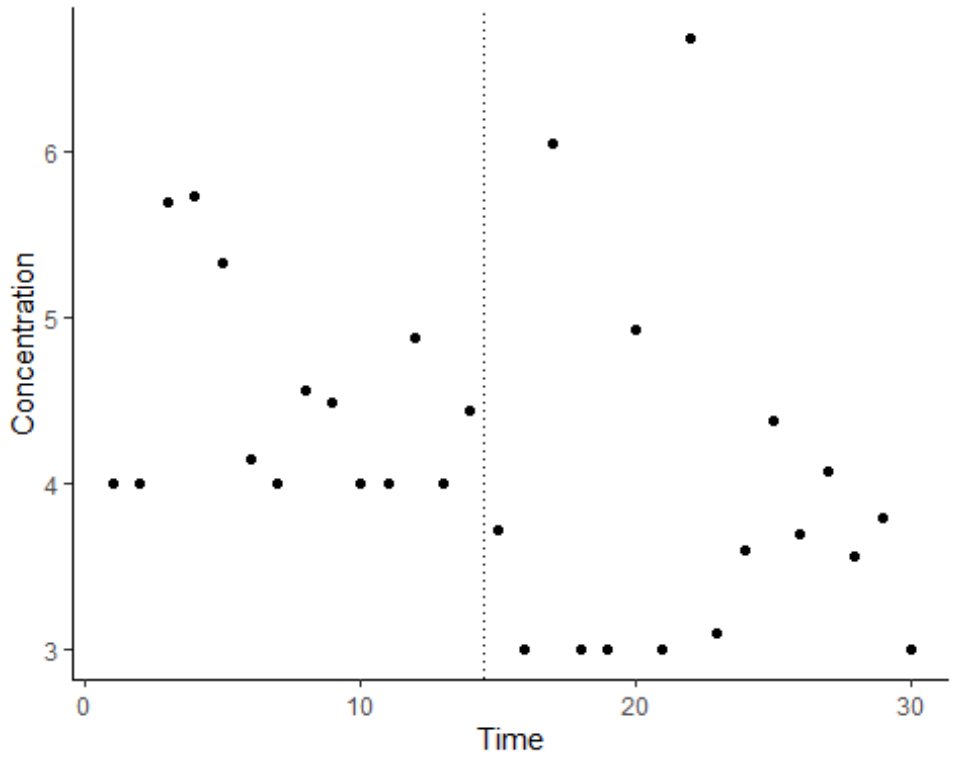


Figure B.1 A series with multiple censoring levels: at 4 up to time point 14 and at 3 afterwards.

B.1 Mann-Kendall

Mann-Kendall are available in a variety of R packages. All methods handle ties in the same way as described by the original publications (Hirsch et al., 1982; Hirsch and Slack, 1984). For Mann-Kendall test only one censoring level is allowed. For the simulated data we adjust the censoring level to the higher of the two prevailing levels and substitute with half of the censoring level (Figure B.2). Observe that failing to adjust the two censoring levels to a single one in a meaningful way will lead to artificial trends shown in analyses.

```
data_censored_multiple%>%
  mutate(conc_2=case_when(date<15 & conc<4 ~4/2,
                          date>=15 & conc<4~4/2,
                          TRUE~conc))->data_censored_single

data_censored_single%>%
  ggplot(aes(y=conc_2, x=date))+
  geom_point()+
  xlab("Time")+
  ylab("Concentration")+
  geom_vline(xintercept=14.5, lwd=0.5, lty=3)+
  theme_classic()
```

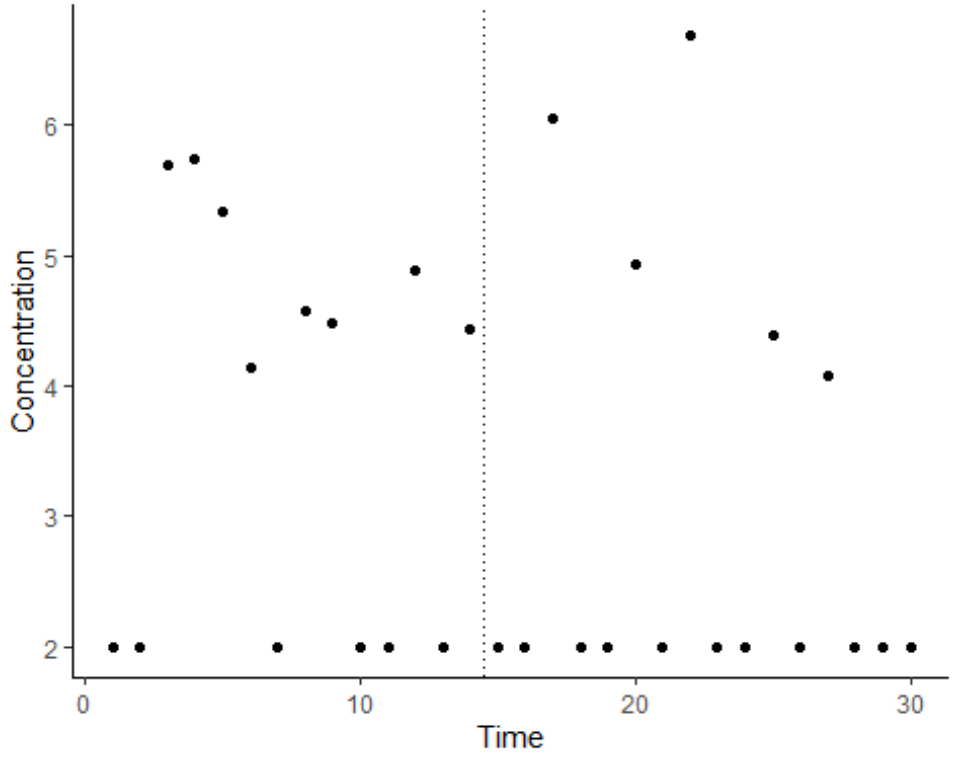


Figure B.2 A series with single censoring levels at 4, with values substituted at half of the censoring level.

B.1.1 Mann-Kendall tests with the package `rkt`

`rkt()` computes Mann-Kendall tests either for individual series or separately for blocks with subsequent combination into one trend test statistics. Blocks can, for example, be seasons or sites. This function also allows the inclusion of covariates.

```
library(rkt)

rkt(data_censored_single$date, data_censored_single$conc_2)


Standard model
Tau = -0.1724138
Score =  -75
var(Score) =  2552.333
2-sided p-value =  0.1429896
Theil-Sen's (MK) or seasonal/regional Kendall (SKT/RKT) slope=  0
```

B.1.2 Mann-Kendall tests with the package `trend`

In this package the input data need to be specified as a dataset of type `ts`, i.e. a time series object.

`mk.test()` computes Mann-Kendall tests on single series.
`smk.test()` computes seasonal Mann-Kendall tests, i.e. it determines individual Mann-Kendall test statistics for each season and then combines them into a single statistics.

```
library(trend)

mk.test(data_censored_single$conc_2)


    Mann-Kendall trend test

data:  data_censored_single$conc_2
z = -1.4647, n = 30, p-value = 0.143
alternative hypothesis: true S is not equal to 0
sample estimates:
           S         varS          tau
 -75.0000000 2552.3333333   -0.2079606
```

B.1.3 Mann-Kendall tests with the package `Kendall`

`MannKendall()` computes a simple Mann-Kendall test on a series, no time variable is provided, i.e. it is assumed that the data is ordered in time when passed to the function.

`SeasonalMannKendall()` computes Mann-Kendall tests for monthly series and combines them to a common trend test.For this a time series object in matrix form (one column per season) can be passed.

```
library(Kendall)

MannKendall(data_censored_single$conc_2)

tau = -0.208, 2-sided pvalue =0.14299
```

B.2 Regression and GAM with substitution

Regression models are a common choice for trend analysis. They are more reliant on distributional assumptions compared to Mann-Kendall tests, but also have the advantage that they are more flexible and more information can be extracted from the model results. For example, there is the possibility to specify a form for the assumed trend, such as linear or exponential, or let the trend be data-driven. Explanatory variables other than time can be

added to the models to decrease residual variation. When fitting trend models to data with substituted values we need again use a single RL at which data is censored.

B.2.1. Regression in base R

`lm()` allows the fit of a linear trend by using `date` as explanatory variable.

```
model_lm_subst<-lm(conc_2~date, data=data_censored_single)

summary(model_lm_subst)


Call:
lm(formula = conc_2 ~ date, data = data_censored_single)

Residuals:
    Min      1Q  Median      3Q     Max
-1.9415 -1.2698 -0.7497  1.3778  3.6544

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.98486    0.59742   6.670 3.08e-07 ***
date        -0.04334    0.03365  -1.288    0.208
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.595 on 28 degrees of freedom
Multiple R-squared:  0.05592,   Adjusted R-squared:  0.0222
F-statistic: 1.658 on 1 and 28 DF,  p-value: 0.2083
```

B.2.2 Regression and GAM with package `mgcv`

The package `mgcv` can run models with linear trends, but is mainly used if the trend is chosen to be data-driven, i.e. the trend curve is fitted by a smooth function.

`gam` allows the inclusion of a thin plate spline or other types of spline functions to obtain a data-driven trend by using the function `s()`, where s indicates the type of spline used. Using `date` without `s()` gives a linear trend.

```
library(mgcv)

model_gam_subst<-gam(conc_2~s(date), data=data_censored_single)

summary(model_gam_subst)


Family: gaussian
Link function: identity

Formula:
conc_2 ~ s(date)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.3131     0.2913   11.38 5.21e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Approximate significance of smooth terms:
        edf Ref.df     F p-value
s(date)   1      1 1.658   0.208

R-sq.(adj) =  0.0222   Deviance explained = 5.59%
GCV =  2.727  Scale est. = 2.5452    n = 30
```

B.3 Censored regression and GAM with constant censoring level

To include a single constant censoring levels directly in the analysis regression and GAM models can be extended to include the censoring level and whether the direction of censoring is left or right. The following packages can be used.

B.3.1 Censored regression with package `censReg`

`censReg()` allows both left and right censoring. The statement `left=` or `right=` allows the specification of the constant censoring level and its direction, i.e. it gives the upper limit for left-censored data and the lower limit for right censored data. The data provided does not need to be adjusted to the censoring level. Any observation that is below the limit given for left-censored data is regarded at censored and adjusted to that level. Only linear trends can be estimated.

```
library(censReg)

model_censReg<-censReg(conc_1 ~ date,left=4, data = data_censored_multiple)

summary(model_censReg)


Call:
censReg(formula = conc_1 ~ date, left = 4, data = data_censored_multiple)

Observations:
         Total  Left-censored     Uncensored Right-censored
            30             17             13              0

Coefficients:
            Estimate Std. error t value  Pr(> t)
(Intercept)  4.43589    0.55913   7.934 2.13e-15 ***
date        -0.04300    0.03346  -1.285    0.199
logSigma     0.28697    0.22060   1.301    0.193
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Newton-Raphson maximisation, 5 iterations
Return code 1: gradient close to zero (gradtol)
Log-likelihood: -32.42829 on 3 Df
```

B.3.2 Censored regression and GAM with the package `VGAM`

The package VGAM also allows a constant censoring level.

`vglm()` is used if the trend is estimated to be linear. The censoring level is set in the `familiy` statement, where `tobit`is chosen. The user specifies the direction of censoring (`Upper` or `Lower`) and the level of censoring.

```
library(VGAM)

model_VGLM<-vglm(conc_1~ date, family=tobit(Lower = 4), data = data_censore
d_multiple)

summary(model_VGLM)

Call:
vglm(formula = conc_1 ~ date, family = tobit(Lower = 4), data = data_censor
ed_multiple)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1  4.43589    0.54980   8.068 7.14e-16 ***
(Intercept):2  0.28708    0.18580   1.545    0.122
date          -0.04300    0.03252  -1.322    0.186
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Names of linear predictors: mu, loglink(sd)

Log-likelihood: -32.4283 on 57 degrees of freedom

Number of Fisher scoring iterations: 13

No Hauck-Donner effect found in any of the estimates
```

`vgam()` extends the `vglm` model by also allowing smooth terms, i.e. here we can fit a data-driven trend. For this `s()` is used around the time variable. Defining the censoring levels works in the same way as for `vglm`.

```
library(VGAM)

model_VGAM<-vgam(conc_1~ s(date), family=tobit(Lower = 4), data = data_cens
ored_multiple)

summary(model_VGAM)


Call:
vgam(formula = conc_1 ~ s(date), family = tobit(Lower = 4), data = data_cen
sored_multiple)

Names of additive predictors: mu, loglink(sd)

Dispersion Parameter for tobit family:   1

Log-likelihood: -30.72037 on 54.092 degrees of freedom

Number of Fisher scoring iterations:  20

DF for Terms and Approximate Chi-squares for Nonparametric Effects
```

```
              Df Npar Df Npar Chisq   P(Chi)
(Intercept):1  1
(Intercept):2  1
s(date)        1     2.9     3.7153 0.280268
```

B.4. Censored regression and GAM with multiple censoring levels

If several censoring levels are present it can be advantageous to use models that can incorporate that to retain as much information as possible from the data. Again functions that allow to estimate linear trends only and functions that allow smooth trend curves are available.

*B.4.1 Censored regression with the package **NADA***

`cenreg()` fits a regression model on data with multiple censoring levels. For this the response variable needs to be given in two parts: The first column should contain the observed concentration or the value of the censoring limit. The second column contains information if this observation is censored or not as a logical variable.

This function does not allow a `data=` statement, therefore it is called using `with` and the data set name before specifying the model.

```
library(NADA)

data_original%>%
  mutate(conc_3=case_when(date<15 & conc<4 ~4,
                          date>=15 & conc<3~3,
                          TRUE~conc),
         cens=case_when(date<15 & conc<4~TRUE,
                        date>=15 & conc<3~TRUE,
                        TRUE~FALSE)) -> data_censored_multipe_for_cenreg

with(data_censored_multipe_for_cenreg, cenreg(Cen(conc_3, cens)~date))->mod
el_cenreg

summary(model_cenreg)

               Value Std. Error     z        p
(Intercept)  1.44647    0.11815 12.24 1.84e-34
date        -0.00708    0.00641 -1.11 2.69e-01
Log(scale)  -1.25349    0.17616 -7.12 1.11e-12

Scale = 0.286

Log Normal distribution
Loglik(model)= -40.7   Loglik(intercept only)= -41.3
Loglik-r:  0.192659

Chisq= 1.13 on 1 degrees of freedom, p= 0.29
Number of Newton-Raphson Iterations: 3
n = 30
```

B.4.2 Censored regression and GAM with the package `mgcv`

`gam()` allows the fit of both a linear trend and a smooth trend. Censored data are modeled by the family called `cnorm`, which demands a combined response variable that contains both an upper and lower limit for every observation. In our example we have interval-censored observations as the concentrations cannot be lower than 0. Two variables are created: `conc_upper` and `conc_lower`. If the observation is not censored, both these columns contain the observed value. If the observation is interval-censored, `conc_lower`is set to zero, while `conc_upper`contains the censoring limit. The two variable are combined into one called `conc_comb`to be further used in the model.

```
library(mgcv)

data_original%>%
  mutate(conc_upper=case_when(date<15 & conc<4 ~4,
                              date>=15 & conc<3~3,
                              TRUE~conc),
                       conc_lower=case_when(date<15 & conc<4 ~0,
                              date>=15 & conc<3~0,
                              TRUE~conc))%>%
  mutate(conc_comb=cbind(conc_lower, conc_upper)) -> data_censored_for_gam_
cnorm

model_gam_cnorm<-gam(conc_comb~date, family=cnorm, data=data_censored_for_g
am_cnorm)

summary(model_gam_cnorm)


Family: cnorm(1.325)
Link function: identity

Formula:
conc_comb ~ date

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.26611    0.53407   7.988 1.37e-15 ***
date        -0.02658    0.02955  -0.900    0.368
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R-sq.(adj) =  -0.056   Deviance explained = 2.57%
-REML = 8.5604  Scale est. = 1         n = 30
```

Including a smooth term for the trend we use `s(date)`.

```
library(mgcv)

model_gam<-gam(conc_comb~s(date), family=cnorm, data=data_censored_for_gam_
cnorm)
summary(model_gam)
```

```
Family: cnorm(1.325)
Link function: identity

Formula:
conc_comb ~ s(date)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   3.8541     0.2561   15.05   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
        edf Ref.df Chi.sq p-value
s(date)   1      1  0.809   0.368

R-sq.(adj) =  -0.056   Deviance explained = 2.57%
-REML = 6.4023  Scale est. = 1         n = 30
```

*B.4.3 Censored regression and GAM with the package **brms***

Similarly, the models can be specified in a Bayesian model. Input is given as the concentration value or the censoring level (`conc_3`) and as a variable indicating if and how the data point is censored (`cens`).

```
library(brms)

data_original%>%
  mutate(conc_3=case_when(date<15 & conc<4 ~4,
                          date>=15 & conc<3~3,
                          TRUE~conc),
         cens=case_when(date<15 & conc<4~"left",
                        date>=15 & conc<3~"left",
                        TRUE~"none"))->data_censored_multipe_for_brms

model_brm_linear <-
  brm(data = data_censored_multipe_for_brms,
      family = gaussian,
      conc_3 | cens(cens) ~ date,
      prior = c(prior(normal(0, 1), class=b),
                prior(normal(0, 1), class = sigma)),
      chains = 4, cores = 4)

print(model_brm_linear)

 Family: gaussian
  Links: mu = identity; sigma = identity
Formula: conc_3 | cens(cens) ~ date
   Data: data_censored_multipe_for_brms (Number of observations: 30)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Regression Coefficients:
          Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
```

```
Intercept     4.23      0.56     3.07     5.31 1.00     3341     2652
date         -0.03      0.03    -0.09     0.03 1.00     3352     2691

Further Distributional Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma     1.36      0.24     0.98     1.91 1.00     2478     2729

Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
and Tail_ESS are effective sample size measures, and Rhat is the potential
scale reduction factor on split chains (at convergence, Rhat = 1).
```

To include a smooth trend term we run the same model but specifying the date variable as a spline (`s(date)`).

```
model_brm_smooth <-
  brm(data = data_censored_multipe_for_brms,
      family = gaussian,
      conc_3 | cens(cens) ~ s(date),
      prior = c(prior(normal(0, 1), class=b),
                prior(normal(0, 2), class = sigma),
                prior(normal(1,5), class=sds)),
      chains = 4, cores = 4)

print(model_brm_smooth)

 Family: gaussian
  Links: mu = identity; sigma = identity
Formula: conc_3 | cens(cens) ~ s(date)
   Data: data_censored_multipe_for_brms (Number of observations: 30)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Smoothing Spline Hyperparameters:
             Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sds(sdate_1)     1.39      1.42     0.03     5.41 1.00     1206     1725

Regression Coefficients:
          Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept     3.81      0.30     3.16     4.35 1.00     3053     2279
sdate_1      -0.18      0.92    -1.93     1.67 1.00     3531     2203

Further Distributional Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma     1.45      0.28     1.03     2.09 1.00     3209     2422

Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
and Tail_ESS are effective sample size measures, and Rhat is the potential
scale reduction factor on split chains (at convergence, Rhat = 1).
```

Appendix C: Updated code for trend screening in several series

C.1: Main function

The function to run trend analysis for a number of data series that contain censored data follow the same structure as the earlier published code for non-censored data (von Brömssen et al., 2021; https://github.com/claudiavonbromssen/Trend-screening). Compared to that code, some controls are removed from the source code due to a difficulty to combine them with the current structure of the response variable. Therefore it is necessary to verify that there are only single observations for each time point and to remove lines that contain missing values in the response before running the models.

```
screeningmodeling_cnorm <- function(.data,
                                    datevar, #variabel med datum (i datumformat!)
                                    values, # variabel med värden
                                    link = "identity",
                                    autocor = FALSE,
                                    conf.type = "confidence",
                                    conf.level=0.95,
                                    tdist = FALSE, # only works with autocor = FALSE
                                    beep = FALSE,
                                    ...){ # Variablerna att nesta under (stationsid, etc,
ibland variabelnamn om gather är kört)

  nestvars <- enquos(...)
  datevar <- enquo(datevar)
  variable <- enquo(values)
  plan(multisession)
  tictoc::tic()
  .data %>%
    mutate(variable = !!variable,
           date = !!datevar) %>%
    select(date, variable, !!!nestvars) %>%
    # group_by(!!!nestvars, date) %>%
    # summarise_at("variable", mean) %>%
    # ungroup() %>%
    #drop_na(variable) %>%
    group_by(!!!nestvars) %>%
    mutate(decimaldate = decimal_date(date),
           month = month(date)) %>%
    nest() %>%
    ungroup() %>%
    mutate(
      fit = future_map(data, possibly(~ modeling_cnorm(.x,
                                                       link = link,
                                                       autocor = autocor,
                                                       tdist = tdist),

                                       otherwise = NA_integer_,
                                       quiet = F),
                       .progress = T, seed=T),
     fderiv = map2(fit, data, possibly(~ derivatives(object=.x,
type="forward",select = "s(decimaldate)", interval=conf.type, level=conf.level,
n=NROW(.y)), otherwise = NA_integer_)),

      predict = map2(fit, data, possibly(~ predict(.x, newdata = .y, type =
"terms") %>% as_tibble(), otherwise = NA_integer_)),
      fitted = map2(fit, data, possibly(~ predict(.x, newdata = .y, type = "link"),
otherwise = NA_integer_)),
      autocor = map_lgl(fit, possibly(~.x$autocor, otherwise = NA_integer_)),
      intercept = map_dbl(fit, possibly(~ coef(.x) %>% .[1], otherwise =
NA_integer_))
```

```
    ) %>%
    group_by(!!!nestvars) %>%
    dplyr::select(!!!nestvars, autocor, everything())->
    output
  tictoc::toc()
  if(beep){beepr::beep()}
  return(output)
}
```

C.2 Calling the GAM model

```
modeling_cnorm <- function(x, link = "identity",autocor=FALSE,  tdist = tdist) {

 formula <- variable ~ s(decimaldate, k = round(nrow(x)/2))

  x <- drop_na(x, variable)
  if(autocor == TRUE)
  {out <- try(model(x, link, formula))
  if ("try-error" %in% class(out)) {
    out <- try(model(x, link, formula, opt = "optim"))
  }
  if ("try-error" %in% class(out)) {
    out <- model_gam_cnorm(x, link, formula)
  }}else{if(tdist == T){out <- model_gam_t(x, link, formula) }
    else{out <- model_gam_cnorm(x, link, formula)}}

  return(out)
}


model_gam_cnorm <- function(x, link = "identity", formula) {
  y <- gam(
    data = x,
    formula = formula,
    family = cnorm(link = link),

    method = "REML"
  )
  y$autocor <- FALSE
  return(y)
}
```

C.3: Running from main script

Lower and upper limits are defined for each observation. If the data point is not censored both the upper and lower limit is the observed value, otherwise the lower limit is set to zero. This is necessary to define interval-censored data. A variable "ocens" is created to count the number of uncensored variables in order to filter out series with few uncensored data points.

```
data1%>%mutate(year=year(provtagningsdatum),
  conc_upper=case_when(matvardetal<=0.02 & year<2013 ~0.02,
                       matvardetal<0.01 & year>=2013 ~0.01,
                       TRUE~matvardetal),
            conc_lower=case_when(matvardetal<=0.02 & year<2013~0,
                       matvardetal<0.01 & year>=2013~0,
                       TRUE~matvardetal),
            ocens=case_when(matvardetal<=0.02 & year<2013~0,
                       matvardetal<0.01 & year>=2013~0,
                       TRUE~1))->data3
```

To run the model a variable "pb_comb" is created using the lower and upper limits of all observations. Only data after 1994 and only series with at least 30% non-sensored data are used.

Rows with missing values in the response variable are removed using the `drop_na()` function. `Autocor` and `tdist` needs to specified as FALSE.

```
data3%>%
  mutate(month=month(provtagningsdatum), pb_comb=cbind(conc_upper, conc_lower))%>%
  filter(year>1994)%>%
  group_by(station)%>%
  filter(n()>15, sum(ocens)>=n()*0.3)%>%
  drop_na(matvardetal)%>%
   select(station,
         provtagningsdatum,
         pb_comb)%>%

  mutate(SiteID=as.factor(station))%>%

    screeningmodeling_cnorm(values=pb_comb,
                    datevar = provtagningsdatum,
                    link = "log",
                    conf.type = "conf",
                    conf.level=0.95,
                    beep = TRUE,
                    tdist = F,
                    autocor = FALSE,
                    station) ->
  trendplotdata_grund_cnorm
```

C.4: Plotting single series

To validate that the model fit is acceptable, single series with their predicted temporal trend can be visualized. The source function from the original script is adjusted for that.

```
plot_individual_trend_cnorm <- function(x, y=NULL, title=NULL){
  if(nrow(x) != 1){stop("Filter out the variable (and/or station) you are
interested in.")}
  annualterm <- predict(x$fit[[1]], newdata=x$data[[1]], type="response")
  #intercept <- x$fit[[1]]$coef["(Intercept)"]
  x$fderiv[[1]] %>%
    transmute(deriv = .derivative,
              deriv_se = .se,
              lower=.lower_ci,
              upper=.upper_ci) %>%
    as_tibble %>%
    rowwise %>%
    mutate(signif = !between(0, lower, upper),
           sign=sign(deriv),
           signif_sign = signif*sign) %>%
    ungroup %>% bind_cols(x$data[[1]],.) %>%
    mutate(trend = annualterm) %>%
    drop_na(variable) %>%
    ggplot(aes(x=date))+geom_line(aes(y=variable[,1]))+
    geom_line(aes(y=trend, color=as_factor(signif_sign), group=c(0)), lwd=1.5)+
    scale_color_manual(values=c("-1" = "#56B4E9",
                                "0" = "#F0E442",
                                "1" = "#D55E00"))+
    theme_bw()+
    theme(text= element_text(size = 20))+
    labs(x="Date", y=y,title=title)+
    #ylim(0,20)+
    #xlim(as.Date("2020-01-01"), as.Date("2021-01-01"))+
    theme(legend.position = "none") -> p
  return(p)
}
```

Appendix D: Additional figures

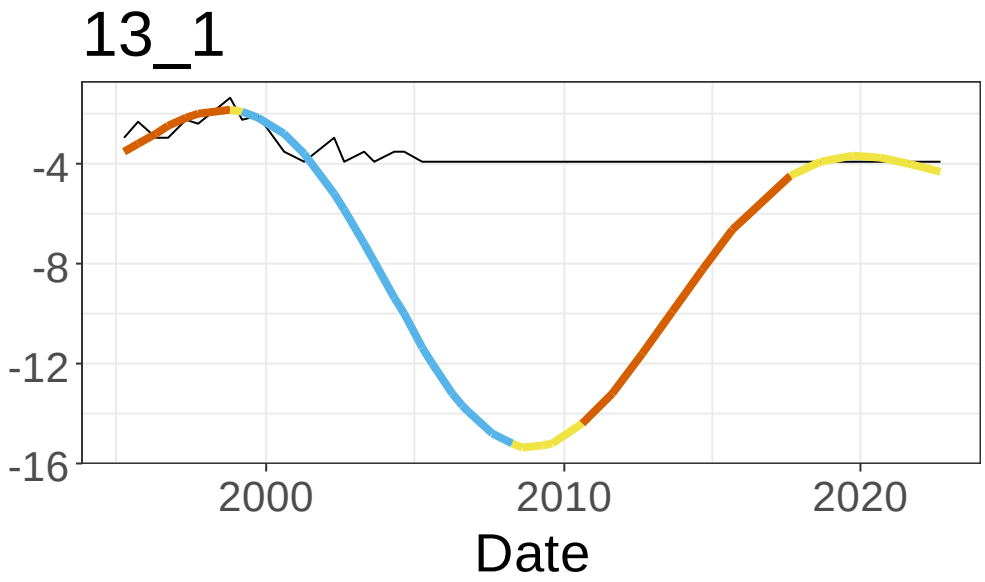


Figure D.1 Model for Pb at station 13_1 using substitution. The model with interval-censored data leads to similar results.