A comparably robust approach to estimate the left-censored data of trace elements in Swedish groundwater

Author: Cong Li
Supervisor: Prof. Rolf Larsson
A comparably robust approach to estimate the left-censored data of trace elements in Swedish groundwater

Author: Cong Li
Supervisor: Prof. Rolf Larsson

Abstract

Groundwater data in this thesis, which is taken from the database of Sveriges Geologiska Undersökning, characterizes chemical and quantitative status of groundwater in Sweden. The data usually is recorded with only quantification limits when it is below certain values. Accordingly, this thesis is aiming at handling such kind of data. The thesis considers this topic by using the EM algorithm to get the results from maximum likelihood estimation. Consequently, estimations of distributions on censored data of trace elements are expounded on. Related simulations show that the estimation is acceptable.

Keywords: groundwater, left-censored data, the EM algorithm, maximum likelihood estimation

Acknowledgements

I owe my deepest gratitude to my supervisor Prof. Rolf Larsson. His patient supervising helps me a lot on writing this thesis.

I give great thanks to Dr. Bo Thunholm and other hydrogeologists in Sveriges geologiska undersökning. Their careful explanations on the data help me to start the thesis.

I also thank for everyone I have met in Sweden. I have learned a lot about knowledge and life here.

1Email: Cong.Li.7752@student.uu.se
Contents

1 Introduction 1

2 Methodologies 2
   2.1 The EM algorithm ........................................ 2
      2.1.1 The general form .................................... 2
      2.1.2 Applying the EM algorithm in this thesis ............ 2
   2.2 Essential tests ........................................... 5
      2.2.1 Mann-Kendall test ................................... 5
      2.2.2 Shapiro-Wilk test .................................... 5
      2.2.3 Ljung-Box test ....................................... 6

3 Data 7

4 Models 8
   4.1 Trend analysis ............................................ 10
   4.2 The AR(0) model .......................................... 11
   4.3 The AR(1) model, including how to extend to the ARMA(p,q)
      model .......................................................... 11

5 Results 13

6 Conclusions 17

Appendix A Plots 21

Appendix B Part of codes 22
1 Introduction

In laboratories, scientists cannot acquire exact measurements, when the real values are beyond the detection limits of certain methods or present instruments. At that time, the detection limits are registered instead. It is common that experimental stations only record the lowest detection levels of trace elements in groundwater tests, which are called left-censored data in statistics.

This thesis is aiming at developing a more robust approach to estimate such kind of data. In practice, if we use quantifications of the censored data instead of real values, the sample values will be overestimated. Or on the condition that we treat all the censored data as zero, the sample values will be underestimated. Thus, the descriptive statistics and inference statistics made from either one can not be dependable. We need to find more information in the censored data therefore.

For the moment, on the case to analyse censored data, there are three main solutions. Intuitively, we can substitute censored data with other values. One of the most popular ways is to replace them with half the values of the limit of quantifications (LOQ). It is widely used as a routine without any strict proof. The standard statistical way to settle censored data is to use statistical methods on fitted distributions, like maximum likelihood estimation (MLE) used in this thesis. In purpose to get the results of MLE, we have to choose an algorithm. In this thesis, we select the expectation-maximization algorithm (EM), which makes MLE to be restricted MLE (RMLE) and actually a more robust method. In fact, the newest way for this topic is to use robust methods, which means the method relies more on the observed data than the fitted distribution. To check more on the statistical application on groundwater data, we refer to papers like Helsel(1990) and Helsel(2005). Also, there is a paper written by Aastrup and Thunholm(2001) which can be referred to.

To discuss on the topic, we firstly concentrate on part2, which is focused on introducing important methods used in this thesis. Then we describe the data
in part 3. To demonstrate how to cope with the left-censored data, we take a complete record from year 1968 to year 2010 of the element Fe from a station. In part 4, we show how to set models at the first place, then apply the EM algorithm to estimate the parameters of the distribution of the left-censored data. For the detailed codes, see part B in the appendix. In the last part 6, we summarize the whole thesis and point out what we can improve on the topic.

2 Methodologies

2.1 The EM algorithm

For MLE in incomplete data, there is an algorithm, the EM algorithm, which is often plied. In this thesis, we mainly refer to the paper by Larsson (1990). By contrast with the Newton-Raphson method, the EM algorithm does not demand the calculations of second derivatives. So for many instances, it may become much simpler in computations. Accordingly, the EM algorithm is widely used in MLE using censored data.

2.1.1 The general form

We suppose there are two series of data which are X and Y. To be specific, X contains \((x_1, x_2, ..., x_n)\) and Y contains \((y_1, y_2, ..., y_n)\). Moreover, the distribution of the element \(x_i\) is \(F_\alpha(x)\), while the data set Y is observed. Once we have the relationship \(y_i = y_i(x_i)\), the EM algorithm can be employed for estimating the parameter \(\alpha\). It is usually divided into two steps: the E step to build the likelihood function, and the M step to get the MLEs of the parameters of E step.

2.1.2 Applying the EM algorithm in this thesis

Observing figure 1, the observed data and censored data can be temporally related. We use the autoregressive model with a Gaussian white noise to apply
the EM algorithm.

**The E step** We have \( E(\log L(x_1, ..., x_n; \varphi, \sigma)|y_0, ..., y_{n-1}; \varphi^{(m)}, \sigma^{(m)}) \) as the general function, where \( \log L \) can be expressed as follows.

\[
\log L = -(n \log(\sqrt{2\pi}\sigma^{(m)})) + \sum_{j=1}^{n} \frac{(x_j - \varphi^{(m)}(y_{j-1}))^2}{2(\sigma^{(m)})^2}
\]

(1)

The above equation only depends on \( \sum x_j^2, \sum x_jy_{j-1} \). So the expectation of the log likelihood function only depends on \( E(\sum x_j^2) \) and \( E(\sum x_jy_{j-1}) \). Since the two expectations can be transformed into \( \sum E(x_j^2) \) and \( \sum E(x_j) \), the general function only depends on \( E(x_j^2), E(x_j) \).

To get the expectation of the log-likelihood function, we have to get the \( E(x_j^2) \) and \( E(x_j) \) in advance. Considering censored data, they have limits of quantifications. So the expectation of the \( x_j \) is the next equation.

\[
E(x_j|x_j \in (a, b)) = \frac{E(x_j, x_j \in (a, b))}{P(x_j \in (a, b))} = \frac{\int_a^b xf_j(x)dx}{\int_a^b f_j(x)dx}
\]

(2)

Here, the range of \((a, b)\) should be \((0, b)\), and \((-\infty, \log b)\) after they are transformed, where \(b\) is the corresponding limit of quantification of the censored data. In part3, we do log-transformation on all the data, so the range is not the original one.

We have the next equation of the expectation of the \( x_j \) when it belong to the censored part.

\[
E_1 = E(x_j|x_j \in (-\infty, b)) = \frac{E(x_j, x_j \in (-\infty, b))}{P(x_j \in (-\infty, b))} = \frac{\int_{-\infty}^b xf_j(x)dx}{\int_{-\infty}^b f_j(x)dx}
\]

(3)

Let \( Z_x = \frac{x_j-\mu^{(n)}}{\sigma^{(n)}} \), which follows a standard normal distribution \( N(0,1) \), whose density function is notated as \( f \). So when \( Z_b = \frac{b-\mu}{\sigma^{(n)}} \), we have the density function of \( x_j \) as the following equation.

\[
f_j(x) = f(Z_x)\left|\frac{\partial Z_x}{\partial x}\right| = \frac{f(Z_x)}{\sigma^{(n)}},
\]

(4)

Then equation 3 can be transformed into the next equation.

\[
E_1 = \mu^{(n)} - \frac{\int_{-\infty}^b f'(Z_x)dz_x}{\Phi(Z_b)} = \mu_i^{(n)} - \sigma^{(n)} \frac{\phi(Z_i)}{\Phi(Z_b)},
\]

(5)
where $\Phi(Z_b) = F_x(x)$ and $\phi(Z_b) = f_x(x)$.

For the expectation of $x_j^2$, we can get similar results as equation 3 in the following equation.

$$E_2 = E(x_j^2|x_j \in (-\infty, b)) = \frac{E(x_j^2, x_j \in (-\infty, b))}{P(x_j \in (-\infty, b))} = \frac{\int_{-\infty}^{b} x_j^2 f_j(x) dx}{\int_{-\infty}^{b} f_j(x) dx} \quad (6)$$

Based on $Z_x$ and $Z_b$ as before, we can get the new expression of $E_2$ as follows.

$$E_2 = (\mu^{(n)})^2 + (\sigma^{(n)})^2 - 2\mu^{(n)}\sigma^{(n)} \frac{\phi(Z_b)}{\Phi(Z_b)} - (\sigma^{(n)})^2 \frac{Z_b \phi(Z_b)}{\Phi(Z_b)} \quad (7)$$

Since $y_0, ..., y_{n-1}; \varphi^{(m)}, \sigma^{(m)}$ are already known, $E_1$ and $E_2$ can be got directly.

**The M Step**  
Firstly, we introduce the AR(0) and AR(1) models. The AR(0) model can be expressed as the following one.

$$y_t = \mu + \varepsilon_t. \quad (8)$$

where $\varepsilon_t$ is a Gaussian noise with expectation 0 and standard error $\sigma$.

And the AR(1) model can be expressed as follows.

$$y_{t+1} = \phi y_t + \varepsilon_{t+1}, \quad (9)$$

where $\varepsilon_t$ is a Gaussian noise.

Then, the expectation of log likelihood function for the AR(0) model is given in the next equation.

$$E(\log L) = -(n \log(\sqrt{2\pi\sigma}) + \frac{(\sum E_2 - 2\mu \sum E_1 + \mu^2)}{2(\sigma)^2}. \quad (10)$$

The expectation of log-likelihood function for model AR(1) is given in the next equation.

$$E(\log L) = -(n \log(\sqrt{2\pi\sigma}) + \frac{(\sum E_2 - 2\varphi \sum E_1 y_{j-1} + \varphi^2 \sum y_{j-1}^2)}{2(\sigma)^2}. \quad (11)$$
To find the maximum likelihood estimates of $\varphi, \sigma$, the results are recorded as $\varphi^{(m+1)}, \sigma^{(m+1)}$. Next, $\varphi^{(m+1)}, \sigma^{(m+1)}$ are taken into E step again.

In this paper, when the log-likelihood function is calculated, the expectation of the log-likelihood can only be taken when we just know the censored data. Once the data is observed, we just put the ordinary log-likelihood function into the calculations.

### 2.2 Essential tests

#### 2.2.1 Mann-Kendall test

The Mann-Kendall test is a non-parametric test on trends for temporal data. It is firstly introduced by Mann (1945) and Kendall (1975). When there is little correlation between the observations, we can form the null hypothesis as, for a univariate time series, there is no trend existing.

The corresponding statistic is usually defined as $T = \sum_{i<j} \text{sgn}(Z_j - Z_i)$, where we have the next function.

$$
\text{sgn}(x) = \begin{cases} 
1 & \text{when } x > 0 \\
0 & \text{when } x = 0 \\
-1 & \text{when } x < 0
\end{cases} 
$$

(12)

As for this statistic $T$, it is asymptotically normal distributed, which satisfies $E(T) = 0$ and $\text{Var}(T) = n(n-1)(2n+5)/18$.

For this case, when we choose two-sided Mann-Kendall test, it means that we test whether the statistic is covered by the interval $(-\Phi^{-1}(1-\frac{\alpha}{2}), \Phi^{-1}(1-\frac{\alpha}{2}))$, where $\Phi(x)$ is the distribution function of a standard normal distribution.

#### 2.2.2 Shapiro-Wilk test

The Shapiro-Wilk test is a test to check whether a univariate sample is from a normally distributed population. It was released by Shapiro and Wilk (1965). Suppose we have a sample which is $x_1, x_2, ..., x_n$, then the null hypothesis is that the population for this sample follows the normal distribution.
The test statistic in this test is defined as the next equation.

\[ W = \frac{\left( \sum_{i=1}^{n} a_i x_{(i)} \right)^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \]  

(13)

where \( x_{(i)} \) is the i-th order statistic, \( \bar{x} \) is the mean of this sample, and \((a_1, \ldots, a_n)\) equals \( \frac{m^T V^{-1} V^{-1} m}{m^T V^{-1} V^{-1} m} \), in which \( m = (m_1, m_2, \ldots, m_n)^T \). As for \( m_i \), it stands for the expectation of the \( z_{(i)} \), where \( z_{(i)} \) is the i-th order statistics from a sample of independent and identical random variables \( z_1, z_2, \ldots, z_n \) that follow the standard normal distribution. Similarly, \( V \) stands for the covariance matrix of \( z_{(1)}, z_{(2)}, \ldots, z_{(n)} \).

When \( W \) is very small, the null hypothesis will be rejected. The critical value of \( W \) is usually got from simulations when \( n < 500 \).

### 2.2.3 Ljung-Box test

The Ljung-Box test, which is given by *Ljung and Box (1979)*, is a test for time series data to check whether there is any autocorrelation between different items. When we have a series of data \( x_1, x_2, \ldots, x_n \), the null hypothesis is that we suppose each item is distributed independently. In other words, the hypothesis means that we assume there is no observed correlation in the sample data. We have the statistic given below.

\[ Q = n (n + 2) \sum_{k=1}^{h} \frac{\hat{\rho}_k^2}{n - k} \]  

(14)

where \( \hat{\rho}_k \) is the sample autocorrelation when lag is \( k \), and \( h \) is the amount of lags which are chosen to be tested.

As for the statistic \( Q \), it follows a chi-squared distribution whose degrees of freedom is \( h \). When \( Q > \chi^2_{1-\alpha, h} \), the test result reject the null hypothesis, where the significance level is \( \alpha \).

According to the simulation studies, the Ljung-Box test is better than Box-Pierce test in different sample sizes. That is why we choose it in this thesis.
3 Data

Groundwater data of Sweden in the thesis is coming from hundreds of stations of Sveriges Geologiska Undersökning (SGU), which records chemical and quantitative status of the groundwater bodies. Generally speaking, records in different stations only differ in the numbers and frequencies of the record years, but remain the same in detailed recorded entries. This thesis concentrates on the data of trace elements from station 1 in area 2, whose data starts at the earliest year 1968 and ends at the latest year 2010. As for the others, part of them do not have enough records for statistical inference. Or they can be dealt with just like what will be done here.

Observing figure 7 in appendix and figure 1, among all the recorded chemical elements, Fe is recorded most completely. Moreover, it has the three characters which are going to be discussed: different frequencies in different years, missing data and censored data that is the center of this thesis. The plot of the element Fe is given in figure 1.

![Figure 1: Trend of Fe](image)

Observing figure 1, there are semiyearly records from year 1973 to 1999 and seasonal records from year 1968 to 1972 and year 2000 until now. The data of year 1976 to 1977 and so on are completely missing. The data of year 1988 to 1992 and so on has only a qualification limit. This thesis takes it as
an example to show how to treat censored data.

The log transformation of the data is $y_t = \log(z_t)$, where $z_t$ is the value of year $t$. Then we get figure 2.

![Figure 2: Log transformation of Fe](image)

Observing figure 2, the level of element Fe varies largely during different periods. According to the illumination of SGU, the data of element Fe that is recorded by different experimental methods can be divided into two successive periods. It is shown in figure 3. We will analyse different periods separately. For this example, before the year 1992, all the data are got from SGU. After this year, they are got from Sveriges lantbruksuniversitet (SLU).

4 Models

To handle the censored data, which only has limits of quantification, there are two problems to be solved firstly, the difference in frequencies of sampling and missing data.

For keeping the consistency of the frequency, the seasonal records are transformed into semiyearly records when modeling. However, to keep different frequencies as they are is applicable by using the approach in this thesis, but the process will become much more complex, unless we use the AR(0) model.

In regard to missing data, autoregressive models are utilized. To get the
parameters of the model, the MLE are employed. In fact, for the ARMA(p,q) models, it is not necessary to estimate the missing data if we just want to get the estimation of the censored data. Here, we will estimate the missing data because we want to show a more complete process of estimation.

Before modelling, we have to decide the lag operators for the autoregressive model, which usually are found by observing the trend of the autocorrelation function (ACF) and partial autocorrelation function (PACF).

The ACF can be expressed as the following equation.

$$\rho(k) = \frac{E[(X_t - \mu_t)(X_{t+k} - \mu_{t+k})]}{\sigma_t \sigma_{t+k}}$$

For a certain sample, it can be estimated as the next one.

$$\hat{\rho}(k) = \frac{1}{(n-k) \sigma_t \sigma_{t+k}} \sum_{t=1}^{n-k} (X_t - \mu_t)(X_{t+k} - \mu_{t+k})$$

Based on the above two equations, we can get the PACF. The sample estimation of the PACF is derived from the sample estimation of the ACF.
The sample PACF is highly relying on the sample ACF. To check more on the relation between ACF and PACF, we can refer to the paper by Durbin J. (1960) and the book by Box, Jenkins and Reinsel (1976).

Since there are missing data and censored data, different values in the missing data and censored data will influence the ACFs and PACFs much, which is obvious from equation 15 and 16. So it is not reliable to estimate the correlations in this data set.

One possible way to solve this question is to try various values from 0 to \( n \). After getting the model based on a certain value \( i \), we regard \( i \) is improper for the approach when the EM algorithm can not converge well. It may be because \( i \) does not fit for the model.

Firstly, the lag operator for autoregressive models is set as zero without considering the moving average lag operator. To show how to extend the model, we also try the autoregressive lag operator as one.

4.1 Trend analysis

Before applying certain models, we have to know whether there is any trend inside the series.

Observing figure 3, the transformed data seem to have no obvious trends. To prove the verdict, we take some trend analyses on the transformed data.

When there is no correlation in the data, the nonparametric Mann-Kendall test can be employed, which is just the condition for the AR(0) model. Using the two-sided Mann-Kendall test, the p value for the first period is 0.07, and the p value for the second period is 0.17. Using the significance level 0.05, we conclude there are no trends in the data.

Or if there are some correlations in the data, it is better to use regression with \( y_t \) on \( t \). In this example, observing figure 3, \( y_t \) shows weak linear relation with \( t \). So the Mann-Kendall test is more suitable for this instance.
4.2 The AR(0) model

After the first two steps, the data that we have is recorded in the same frequency without missing data. To handle the censored data, based on semiyearly records from 1968 until now, when lag is set as 0, we have the AR(0) model, which is expressed as equation 8.

Before applying the EM algorithm, we should find the starting values. Considering the whole observed data as a sample, we can select the sample mean as the starting value for $\mu$ and the sample variance for $\sigma^2$. In this dataset, the starting values of $\mu$ and $\sigma$ in the first period are 2.37 and 1.14. Those for the second period are -4.11 and 0.73.

Observing the data, the likelihood function $L$ should satisfy the next equation.

$$-\log L = n \log \sqrt{2\pi}\sigma + \frac{1}{2(\sigma)^2} \sum_i (x_i - \mu)^2 + \sum_j (E_2 - 2\mu E_1 + \mu_x^2), \quad (17)$$

where $x_i$ belongs to the observed data and $x_j$ belongs to the censored data.

4.3 The AR(1) model, including how to extend to the ARMA(p,q) model

Towards the censored data, when the lag is set as 1, we have the AR(1) model, that is expressed as below.

Considering the conditional distribution of $y_t$, we have the conditional equation as equation 9.

$$x_{t+1} = y_{t+1} | y_t = \phi y_t | y_t + \varepsilon_{t+1}. \quad (18)$$

We have that $y_t, ..., y_{t+p}$ is observed and $x_t, ..., x_{t+p}$ is a sample from a normal distribution with mean as $\varphi_1 y_t$ and variable as $\mu^2$. Suppose that the density function of $x_j$ is $f_j(x)$, we have the density function as below.

$$f_j(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x - \varphi y_{j-1})^2/(2\sigma^2)} \quad (19)$$
After the transformation, the EM algorithm can be plied here. The detailed process of applying the EM algorithm in this thesis is referred to part 2.1.2.

But when the previous item is in the censored part, the second mean cannot be calculated for the first item is unknown. So we need to find a more general expression for the mean.

By considering a series which contains \( y_j \) which belongs to the censored data, we have to do the following deduction.

\[
\begin{align*}
y_j &= \phi y_{j-1} + \varepsilon_j \\
y_j &= \phi(\phi y_{j-2} + \varepsilon_{j-1}) + \varepsilon_j \\
... &= ...
\end{align*}
\]

\[
y_j = \phi^{j-k} y_1 + (\phi^{j-k-1} \varepsilon_{k+1} + \phi^{j-k-2} \varepsilon_{k+2} + ... + \varepsilon_j)
\]

where \( k \) satisfies that \( y_k \) is \( y_{(1)} \), \( y_{(1)} \) is the first item in the observed data, and \( j > k \).

In addition, we notice that \( y_{(1)} \) in this data set is \( y_1 \). Hence, \( y_j | y_1 \) that is a new \( x_j \) follows a normal distribution whose mean is \( \phi^j y_1 \) and whose variance is \( \frac{(1-\phi^{j-1})}{1-\phi} \sigma^2 \). The likelihood function is as in the following formula.

\[
- \log L = \sum_i ( n \log \sqrt{2\pi \sigma_i} + \frac{(E_2 - 2\mu_i E_1 + \mu_i^2)}{2(\sigma_i^2)} )
\]

\[
+ \sum_j ( n \log \sqrt{2\pi \sigma_j} + \frac{(E_2 - 2\mu_j E_1 + \mu_j^2)}{2(\sigma_j^2)} ) + \sum_k ( n \log \sqrt{2\pi \sigma} + \frac{(x_k - \mu)^2}{2(\sigma^2)} )
\]

where \( x_i \) belongs to censored data under level 0.01mg/l, \( x_j \) belongs to censored data under level 0.05mg/l and \( x_k \) belongs to the ordinary observed data. It is important to notice that for these three kinds of data, they each have their own \( E1, E2, \mu \) and \( \sigma \). In all the models, we choose \( y_{(1)} \) which is the first item in the observed data as the condition. By dividing the censored data into different levels and calculating together, this thesis answers the question how to analyse censored data with different censored levels together.
The extension of this process to the ARMA(p,q) model  

If there is an ARMA(p,q) model fit the time series data well, we can have $x_i$ which equals $y_i | y_{(1)}$. It follows a normal distribution which has the mean that can be expressed with $y_{(1)}$ and parameters of the model. Also, the variance can be expressed with parameters of the model and $\sigma^2$. Then we can apply the same process as in the AR(1) model above.

5 Results

After applying the EM algorithm, whose codes are in part B in the appendices, we get the values of the parameters, which are given in the next table.

<table>
<thead>
<tr>
<th>Method</th>
<th>Stand error ($\sigma$)</th>
<th>Mean ($\phi$ or $\mu$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The EM algorithm in AR(0) in period 1</td>
<td>1.46</td>
<td>-3.01</td>
</tr>
<tr>
<td>The EM algorithm in AR(0) in period 2</td>
<td>0.17</td>
<td>-0.24</td>
</tr>
<tr>
<td>The EM algorithm in AR(1) in period 1</td>
<td>5.13</td>
<td>-0.80</td>
</tr>
<tr>
<td>The EM algorithm in AR(1) in period 2</td>
<td>1.14</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Referring to the content in methodologies, we can infer that the density function of the log-transformed censored data $x_j$ is as follows.

\[
 f_j(x) = f_j(x|x < b) = \frac{f_j(x, x < b)}{p(x < b)} = \frac{\phi(Z_x)}{\Phi(Z_b)} 
\]  

where $Z_x$ equals $\frac{x-\mu}{\sigma}$, $Z_b$ is $\frac{b-\mu}{\sigma}$, $\phi$ and $\Phi$ stands for density and distribution functions for the standard normal distribution, and $x$ equals $\log z$, where $z$ stands for the original values of the censored data. For the AR(0) model, $\mu$ and $\sigma$ can be observed directly from the table 5. For the AR(1) model, $\sigma$ could be found directly in the table 5, while $\mu$ is $\phi^1 y_1$, which can be referred to part 4.3.
In the same way, we can get estimates of the expectations and variances according to equation 5 and 6. In figure 4, we compare estimates of the expectations of the censored data with other estimates from other methods.

Figure 4: Comparisons of different methods with original data: In the figure, there are four different coloured lines. Among them, the red one is the observed data with LOQs, while the yellow one is the observed data with half the LOQs. The results from the AR(0) model are the green ones, and the results from the AR(1) model are the blue ones.
Observing figure 4, when the EM algorithm is used to calculate a series of adjacent items, the results remain stable after several iterations.

For the AR(1) model, since the precondition of using the EM algorithm in this thesis is that $\varepsilon_{t+1}$ follows $N(0, \sigma^2)$ where $\varepsilon_{t+1} = x_{t+1} - \phi y_t | y_t$, every time the EM algorithm is used, it is necessary to test whether the residuals follow the normal distribution. In the appendix, there is figure 8 to test whether they follow normal distributions. For the AR(0) model, the process is similar.

We give the next table to show the results of Shapiro-Wilk tests and Ljung-Box tests for the AR(0) and AR(1) models.

<table>
<thead>
<tr>
<th>Table 2: Shapiro-Wilk tests results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
</tr>
<tr>
<td>The EM algorithm in AR(0) in period 1</td>
</tr>
<tr>
<td>The EM algorithm in AR(0) in period 2</td>
</tr>
<tr>
<td>The EM algorithm in AR(1) in period 1</td>
</tr>
<tr>
<td>The EM algorithm in AR(1) in period 2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 3: Ljung-Box tests results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
</tr>
<tr>
<td>The EM algorithm in AR(0) in period 1</td>
</tr>
<tr>
<td>The EM algorithm in AR(0) in period 2</td>
</tr>
<tr>
<td>The EM algorithm in AR(1) in period 1</td>
</tr>
<tr>
<td>The EM algorithm in AR(1) in period 2</td>
</tr>
</tbody>
</table>

Observing table 5, it seems that the AR(0) model is not applicable in period 1. But considering that we test the residuals on observed data, while the model is built on observed data and censored data, we can not be sure that we really reject the null hypothesis. It is because we do not know the performances of the residuals on the censored part. Similarly, the rejections of null hypotheses
on Ljung-Box tests can not be really trustable. In addition, we can check the iterations in the next figure.

Figure 5: 100 times iterations of the two models in period 1

Observing the above figure, only the AR(0) model can be used in this approach. To compare which method is the best, we can assume that the results in table 5 are true values of corresponding parameters. Then we use the AR(0) model as an example to show how we know the algorithm is good or not.

From the table 5, we can get the values of parameters in AR(0) model. We make simulations of AR(0) models with random generated $\varepsilon_t$. In this example, we do the simulations 5000 times, and every time we make simulations for 100 items. For example, for one simulation of 100 items, we get parameter values $\sigma = 1.46$ and $\mu = -3.01$. Then by the formula 8 and the generated $\varepsilon_t$, we can get 100 item values of this model. For this simulation, we use certain limits of
quantifications to censor the whole data, like 0.05 mg/l. Since we actually have
known the observed values of the censored data, we can compare the MSEs of
different methods on the censored part. Repeating the process 5000 times, we
give the next table to show the properties of the EM algorithm.

### Table 4: MSE, in this instance, all the data are censored by level 0.05 mg/l

<table>
<thead>
<tr>
<th>Method</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimation with the whole data</td>
<td>2.11</td>
</tr>
<tr>
<td>Estimation with half the LOQ</td>
<td>8.79</td>
</tr>
<tr>
<td>Estimation with the EM algorithm</td>
<td>0.12</td>
</tr>
</tbody>
</table>

Observing table 5, estimation using the EM algorithm obviously has an ad-
vantage over the other methods. In fact, we can have the MSE of the observed
data as $MSE = \frac{1}{mn} \sum \sum (x_{ij} - E(x_{ij}))^2$, and the MSE of the data with predic-
tions from the EM algorithm is $MSE = \frac{1}{mn} \sum \sum (x_{ij} - E(x_{ij}|x_{ij} \in (-\infty, b)))^2$,
where $m$ and $n$ are the number of simulations and items, $i$ and $j$ are the rank-
ing in the simulations and items. It is obvious that by giving a constraint to
$E(x_{ij})$, the MSE will become smaller, which corresponds with the values in
the table. To see more about the simulations, we give the next distribution
plots.

### 6 Conclusions

By concentrating on groundwater data with the censored part, we are mainly
trying to solve these questions: finding the estimation of the distribution of
the left-censored data, dealing with left-censored data in different levels si-
multaneously and programming applicable codes which can deal with different
left-censored data without big changes. In the appendix, we give a solution
to cope with data from different stations at the same time by using the EM
algorithm in the AR(0) model, which works well in most situations.
During solving the above questions, we have to consider how to treat time series data recorded in different frequencies. Also, when facing data recorded by different methods, we simply divided it into different segments, and did the statistical analysis separately. Before modeling the data, we discussed two methods to check whether a time series has a trend or not, whether to use the linear regression method or Mann-Kendall test.

After all, there are several parts on this topic which still can be developed more. At the first place, other methods on dealing with censored data can be considered in the future, like Koul(1996) introducing some robust statistical methods in this area.
References


Appendices

Appendix A   Plots

Figure 7: Plots of other elements with censored data in Station 1 in Area 2: the green points stand for the censored data, while the red points stand for the observed data.
(a) QQ plot for the EM algorithm in period 1 using AR(0)  
(b) QQ plot for the EM algorithm in period 2 using AR(0)  
(c) QQ plot for the EM algorithm in period 1 using AR(1)  
(d) QQ plot for the EM algorithm in period 2 using AR(1)  

Figure 8: QQ plots of AR models  

Appendix B   Part of codes

```r
library(tseries); library(stats4); library(Kendall)
DenFe=read.table("fe.txt",header=F)
denFe=ts(DenFe,start=c(1968,2),end=c(2010,2),frequency=2)
fe=ts(denFe)
```
fe=abs(fe)
nfe=as.numeric(fe)
logfe=log(fe)

# Updating 'logfe' with predictions from AR models#

prepois=c('Filling serial numbers of missing data')
n0=length(prepois)
for(i in 1:n0){
ts0=AR(logfe[1:prepois[i]],method="MLE")
logfe[prepois[i]]=predict(ts0,1)
}

# The EM algorithm (for all the levels in period 1 of the AR(1))#

n="length of the first period"
prenum1="Filling serial numbers of censored data under level 1"
prenum2="Filling serial numbers of censored data under level 2"

E1=rep(0,n);E2=rep(0,n);Mu=rep(0,n)
Y=as.numeric(logfe[1:n])
A=log(0.01);B=log(0.05)
Phi=0;Sigma=0;Phi3=-0.0146;Sigma3=sqrt(1.59)

while(abs(Sigma3-Sigma)>1e-3 & abs(Phi3-Phi)>1e-3){
Sigma=Sigma3
Phi=Phi3
for(j in 1:n){
    if(j%in%prenum1){
        Mu[j]=Phi^j*Y[1]
        Sigma01=sqrt((1-Phi^2)/(1-Phi^2))*Sigma
        StA=(A-Mu[j])/Sigma01
        N2=dnorm(StA)
        N4=pnorm(StA)
    }
}
\[ E1[j] = \mu[j] - \sigma01 \ast \frac{N2}{N4} \]
\[ E2[j] = \mu[j]^2 + \sigma01^2 - 2 \ast \mu[j] \ast \sigma01 \ast \frac{N2}{N4} - \sigma01^2 \ast \text{StA} \ast \frac{N2}{N4} \]

if \((j \%in\% \text{prenum2})\) {
\[ \mu[j] = \phi^j \ast \text{Y}[1] \]
\[ \text{Sigma02} = \sqrt{\left(1 - \phi^2 \right) / \left(1 - \phi^2 \right)} \ast \text{Sigma} \]
\[ \text{StB} = (B - \mu[j]) / \text{Sigma02} \]
\[ \text{N2} = \text{dnorm} (\text{StB}) \]
\[ \text{N4} = \text{pnorm} (\text{StB}) \]
\[ E1[j] = \mu[j] - \sigma02 \ast \frac{N2}{N4} \]
\[ E2[j] = \mu[j]^2 + \sigma02^2 - 2 \ast \mu[j] \ast \sigma02 \ast \frac{N2}{N4} - \sigma02^2 \ast \text{StB} \ast \frac{N2}{N4} \]
}

if \((! j \%in\% \text{prenum1} \& \! j \%in\% \text{prenum2})\) {
\[ E1[j] = \text{Y}[j] \]
\[ E2[j] = \text{Y}[j]^2 \]
}
}

\[ \text{LLD} = \text{function} (\text{Sigma2}, \phi2) \{
\text{if} (\text{Sigma2} > 0) \{
\text{for} (j \text{ in } 2:n) \{
\text{if} (j \%in\% \text{prenum1} \mid j \%in\% \text{prenum2}) \{
\text{Mu}[j] = \phi2^j \ast \text{Y}[1] 
\text{newSigma} = \sqrt{\left(1 - \phi2^2 \right) / \left(1 - \phi2^2 \right)} \ast \text{Sigma2} 
\text{LD}[j-1] = \log \left(\sqrt{2 \ast \pi} \ast \text{newSigma} \right) + \left(E2[j] - 2 \ast E1[j] \ast \text{Mu}[j] + \text{Mu}[j]^2 \right) / \left(2 \ast \text{newSigma}^2 \right)
\}
\text{else} \{
\text{LD}[j-1] = \log \left(\sqrt{2 \ast \pi} \ast \text{Sigma2} \right) + \left(\text{Y}[j] - \phi2 \ast \text{Y}[j-1] \right)^2 / \left(2 \ast \text{Sigma2}^2 \right)
\}
\}
\text{return} \left(\text{sum} (\text{LD}) \right) \}
\]
```r
fit01=mle(LLD, start=list(Sigma2=1.59, Phi2=-0.01))
Cof=as.numeric(coef(fit01))
Sigma3=Cof[1]
Phi3=Cof[2]
```

# To predict the censored data under level 0.01 mg/l and 0.05 mg/l #

```r
E1[prenum1]
E1[prenum2]
```

# To get the EM algorithm for level 0.05 mg/l in period 2 of the AR(1), #
# the data need to be upended firstly. Then it can be predicted as above. #

```r
Mu=0; Sigma=0; Mu3=-2.37; Sigma3=sqrt(1.59)
while(abs(Sigma3-Sigma) >= 1e-3 | abs(Mu3-Mu) >= 1e-3){
    Sigma=Sigma3
    Mu=Mu3
    for(j in 1:n){
        if(j%in%prenum1){
            StA=(A-Mu)/Sigma
            N2=dnorm(StA)
            N4=pnorm(StA)
        }
    }
}
```

```r
```

25
E1[ j ]=\text{Mu} - \text{Sigma} \ast \text{N2} / \text{N4}

E2[ j ]=\text{Mu}^2 + \text{Sigma}^2 - 2 \ast \text{Mu} \ast \text{Sigma} \ast \text{N2} / \text{N4} - \text{Sigma} \ast \text{StA} \ast \text{N2} / \text{N4}

} i f (j \notin \text{prenum2}) {

StB=(B-\text{Mu}) / \text{Sigma}

\text{N2}=\text{dnorm}(\text{StB})

\text{N4}=\text{pnorm}(\text{StB})

E1[ j ]=\text{Mu} - \text{Sigma} \ast \text{N2} / \text{N4}

E2[ j ]=\text{Mu}^2 + \text{Sigma}^2 - 2 \ast \text{Mu} \ast \text{Sigma} \ast \text{N2} / \text{N4} - \text{Sigma}^2 \ast \text{StB} \ast \text{N2} / \text{N4}

} i f ( \text{!j} \notin \text{prenum1} \& \text{!j} \notin \text{prenum2}) {

E1[ j ]=Y[ j ]

E2[ j ]=Y[ j ]^2

}

\text{LLD}=\text{function}(\text{Sigma2},\text{Mu2}){

i f (\text{Sigma2}>0)

\text{for (j in 1:n)}{

i f (j \notin \text{prenum1} | j \notin \text{prenum2}){

LD[j-1]=\log (\sqrt(2 \ast \pi) \ast \text{Sigma2})

+ (E2[j]-2 \ast E1[j] \ast \text{Mu2}+\text{Mu2}^2)/(2 \ast \text{Sigma2}^2)

}

e l s e 

LD[j-1]=\log (\sqrt(2 \ast \pi) \ast \text{Sigma2})+(Y[j]-\text{Mu2})^2/(2 \ast \text{Sigma2}^2)

}

r e t u r n (s u m(LD))

}

e l s e 

{\text{NA}}

}

\text{fit01}=\text{mle}(\text{LLD}, \text{start}=\text{list}(\text{Sigma2}=1.59,\text{Mu2}=-2.37))

\text{Cof}=\text{as.numeric}(\text{coef(fit01)})

\text{Sigma3}=\text{Cof}[1]
Mu3=Cof[2]

# To calculate the starting values, we give it here.

prenum=c('Filling serial numbers of censored data')
Mu3=mean(logfe[-c(prenum)])
Sigma3=var(logfe[-c(prenum)]-mean(logfe[-c(prenum)]))

# Checking the EM algorithm is good or not.

A=log(0.05)

cen=function(dat){
    for(i in 1:length(dat)){
        if(max(A,dat[i])>dat[i]){
            cDat[i]=A
            prenum=c(prenum,i)
        }
        else{
            cDat[i]=dat[i]
        }
    }
    return(prenum)
}

m=100;N=5000;prenum=0;cDat=NA
nMu=-3.01;MSE1=rep(0,N);MSE2=rep(0,N);MSE3=rep(0,N)

for(i in 1:N){
    rd=rnorm(m,0,1.46)
    nDat=rep(nMu,m)+rd
    nPrenum=cen(nDat)[-1]
    le=length(nPrenum)
```r
mts1 = ar(nDat, method = "yule-walker", se.fit = T)
MSE1[i] = 1/le * (mts1$resid[nPrenum])^2 * mts1$resid[nPrenum]

cDat = nDat
cDat[nPrenum] = 1/2 * A
mts2 = ar(cDat, method = "mle")
MSE2[i] = 1/le * (mts2$resid[nPrenum] - nDat[nPrenum] + 1/2 * A)^2 *
(t(mts2$resid[nPrenum] - nDat[nPrenum]))

Y = as.numeric(nDat)
Mu = 0; Sigma = 0; Mu3 = -3; Sigma3 = 1.4

# To insert the EM algorithm for all the levels in period 1 of the AR(0).
MSE3[i] = 1/le * (E1[nPrenum] - nDat[nPrenum])^2 *
(t(E1[nPrenum] - nDat[nPrenum]))

# Here we give codes for calculating data of large scale in all the stations.
Grw = read.csv("data.csv")

index = Grw$Index
Num = length(index)

divid = 1
for (i in 2:Num) {
    if (index[i] != index[i - 1]) {
        divid = c(divid, i)
    }
}
nocensor = NULL; noobserve = NULL; nodata = NULL; censor = NULL
```
arsig=NULL; armu=NULL; Mor=NULL; smallsamp=NULL

N=length(divid)−1

for(i in 1:N){
    I=i
    s1=divid[i]
    s2=divid[i+1]−1
    Y=as.numeric(Grw$Pb[s1:s2])
    Y=Y[!is.na(Y)]
    Y=Y[Y!=0]
    LY1=length(Y[Y<0])
    LY2=length(Y[Y>0])
    if(LY1>0 & LY2>10){
        Mu=0
        Sigma=0
        Mu0=mean(log(abs(Y[!is.na(Y)])))
        Sigma0=var(log(abs(Y[!is.na(Y)])))
        Mu3=Mu0
        Sigma3=Sigma0
        lnY=log(abs(Y))
        Iterat=0

        while(abs(Sigma3−Sigma)>=1e−3 & abs(Mu3−Mu)>=1e−3){
            Sigma=Sigma3
            Mu=Mu3

            for(j1 in 1:length(Y)){
                if(Y[j1]<0){
                    StB=(lnY[j1]−Mu)/Sigma
                    N2=dnorm(StB)
                    N4=pnorm(StB)
                    E1[j1]=Mu−Sigma*N2/N4
                    E2[j1]=Mu^2+Sigma^2−2*Mu*Sigma*N2/N4−Sigma^2*StB*N2/N4
                } else{
                    E1[j1]=lnY[j1]
                }
            }
        }
    }
}

29
E2 \[ j1 \] = \ln Y \[ j1 \]^2 

LLD = function (Sigma2, Mu2) {
  LD = rep(0, length(Y))
  if (Sigma2 > 0) {
    for (j2 in 1:length(Y)) {
      if (Y[j2] < 0) {
        LD[j2-1] = log(sqrt(2*pi)*Sigma2) + 
                    (E2[j2] - 2*E1[j2]*Mu2 + Mu2^2)/(2*Sigma2^2)
      }
      else {
        LD[j2-1] = log(sqrt(2*pi)*Sigma2) + (lnY[j2] - Mu2)^2/(2*Sigma2^2)
      }
    }
    return(sum(LD))
  } else {
    NA
  }
}

fit01 = mle(LLD, start = list(Sigma2 = Sigma0, Mu2 = Mu0))
Cof = as.numeric(coef(fit01))
Sigma3 = Cof[1]
Mu3 = Cof[2]

Iterat = Iterat + 1
if (Iterat == 20) {
  Mor = c(Mor, 1)
  Sigma3 = 0
  Mu3 = 0
  break
}
arsig = c(arsig, Sigma3)
armu = c(armu, Mu3)
if (LY2 == 0) {
    noobserve = c(noobserve, i)
}

if (LY1 == 0) {
    nocensor = c(nocensor, i)
}

if (LY1 > 0 & LY2 <= 10) {
    smallsamp = c(smallsamp, i)
}

...