Lab 4

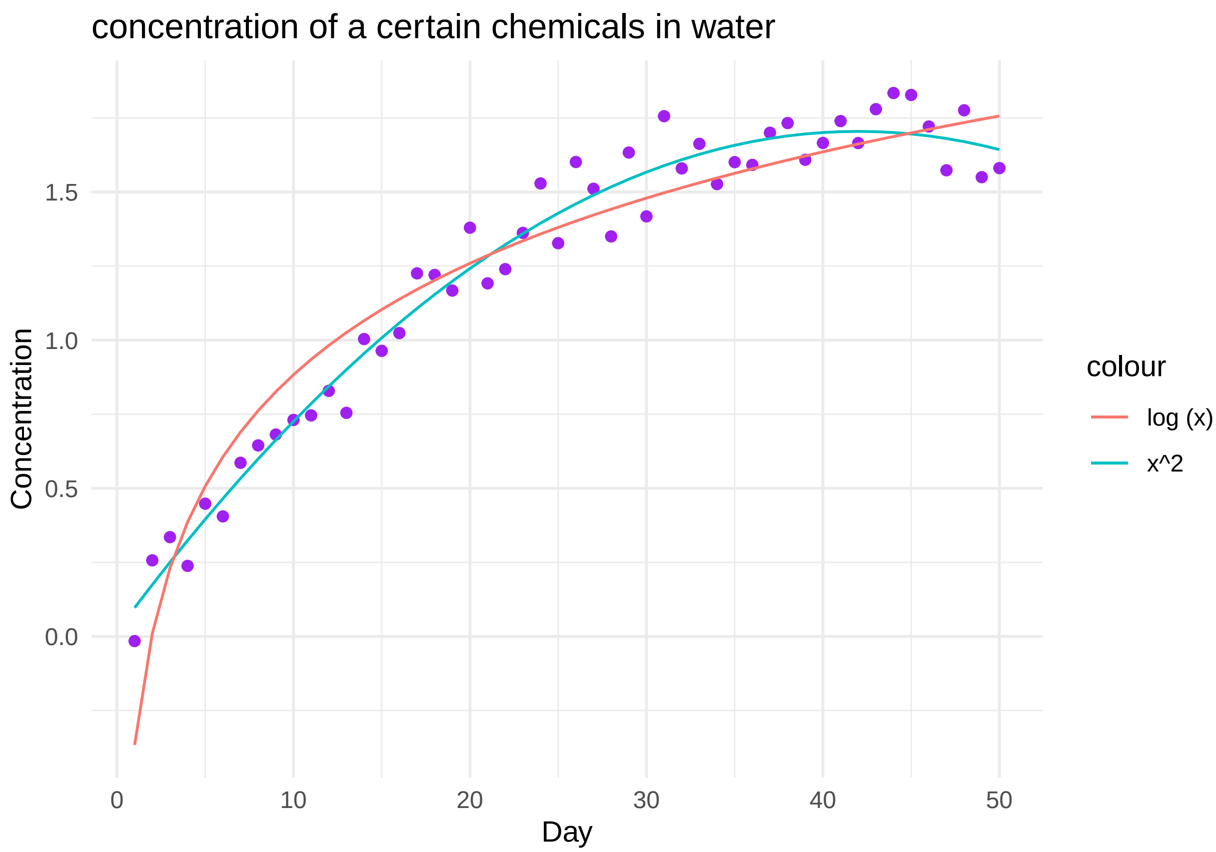
Group 6

11/30/2020

Question 2: Gibbs sampling

1. Import the data to R and plot the dependence of Y on X.

What kind of model is reasonable to use here?



The model is noisy so a simple linear regression can not be used to describe the model. In the plot we have shown a second degree polynomial regression and log(x). Best on the plot, the second degree polynomial is the better choice.

2. A researcher has decided to use the following (random-walk) Bayesian model (n=number of observations, $\vec{\mu} = (\mu_1, ..., \mu_n)$ are unknown parameters):

$$Y_i = \mathcal{N}(\mu, \sigma = 0.2), i = 1, ..., n$$

where the prior is

$$p(\mu_1) = 1$$
$$p(\mu_{i+1} \mid \mu_i) = \mathcal{N}(\mu_i, 0.2) i = 1, ..., n1$$

Present the formulae showing the likelihood $p(Y\ \mu)$ *and the prior* $p(\mu)$. *Hint: a chain rule can be used here* $p(\vec{\mu}) = p(\mu_1)p(\mu_2|\mu_1)...p(\mu_n|\mu_{n-1})$. the PDF of Gaussian Distribution is :

$$\mathcal{N}(\mu,\, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\vec{y_i} - \vec{\mu_i})^2}{2\sigma^2}}$$

and $\sigma^2 = 0.2$

2.2 Likelihood:

to obtain a formulae for likehood we should take product of this PDF:

$$P(\vec{Y}|\vec{\mu}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi 0.2^2}} exp(-\frac{(y_i - \mu_i)^2}{2 * 0.2^2}) = \Big(\frac{1}{\sqrt{0.08\pi}}\Big)^n exp\Big(-\frac{\sum_{i=1}^{n}(y_i - \mu_i)^2}{0.08}\Big)$$

Prior formulae:

$$p(\vec{\mu}) = \prod_{i=1}^{n-1} \frac{1}{\sqrt{2\pi * 0.2^2}} exp(-\frac{(\mu_{i+1} - \mu_i)^2}{2 * 0.2^2})$$
$$P(\vec{\mu}) = \Bigg(\frac{1}{\sqrt{2\pi * 0.2^2}}\Bigg)^n \exp\Bigg[-\frac{1}{2 * 0.2^2}\sum_{i=1}^{n-1}(\vec{\mu_{i+1}} - \vec{\mu_i})^2\Bigg]$$

2.3:

Use Bayes' Theorem to get the posterior up to a constant proportionality, and then find out the distributions of $(\mu_i|\mu_{-i}, Y)$, where μ_{-i}i is a vector containing all μ values except of μ_i.

From Bayes' Theorem we know that:

$$Posterior \propto likelihood * prior$$

hence,

$$P(\vec{\mu}|\vec{Y}) \propto \exp\Bigg[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mu_i)^2\Bigg]\exp\Bigg[-\frac{1}{2\sigma^2}\sum_{i=1}^{n-1}(\mu_{i+1} - \mu_i)^2\Bigg]$$

if $n = 1$ from Hint A:

$$P(\mu_1|\vec{\mu}_{-1}, \vec{Y}) \propto \exp\Bigg[-\frac{1}{2 * 0.2^2}\Bigg[\mu_1 - \frac{y_1 + \mu_2}{2}\Bigg]^2\Bigg]$$

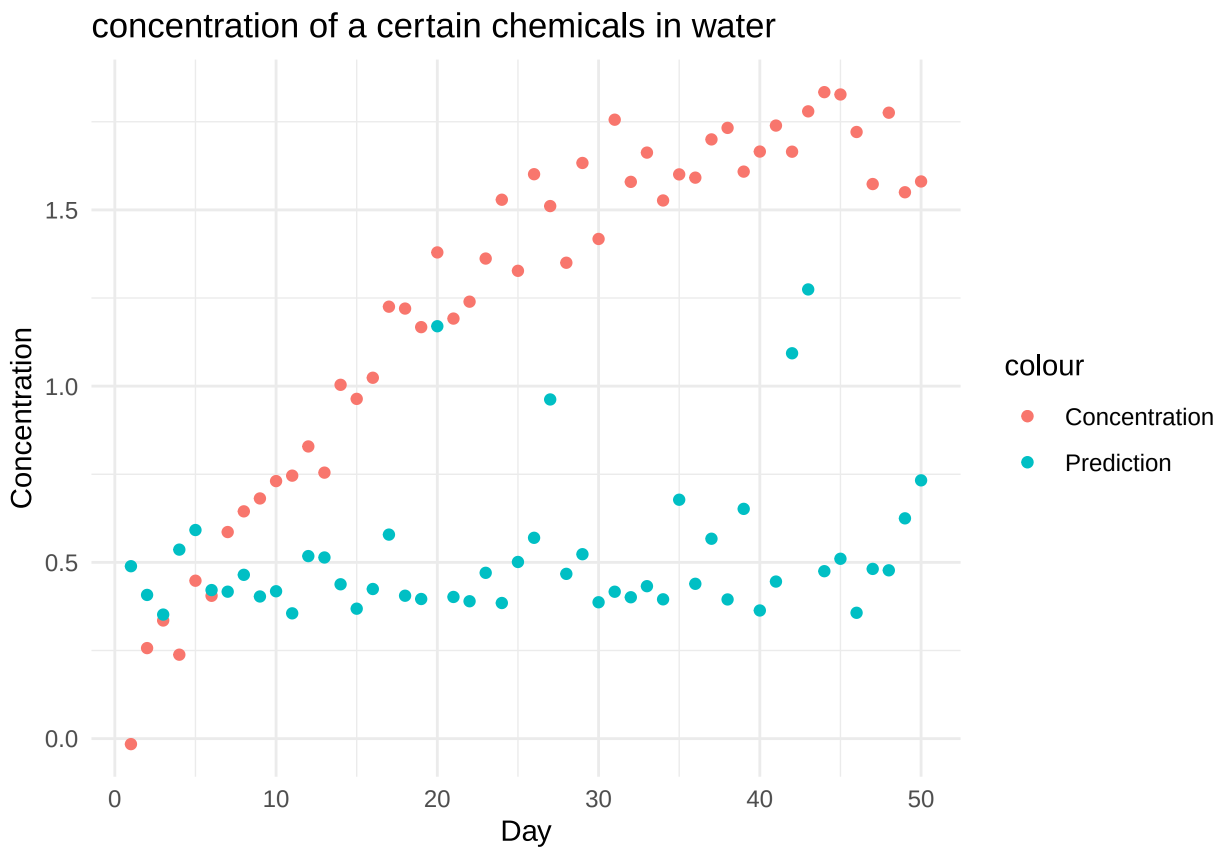
if $i \neq 1, n$ from Hint c:

$$P(\mu_i|\vec{\mu}_{-i}, \vec{Y}) \propto \exp\Bigg[-\frac{3}{0.08}\Bigg[\mu_i - \frac{y_i + \mu_{i-1} + \mu_{i+1}}{3}\Bigg]^2\Bigg]$$

if $i = n$ from Hint B:

$$P(\mu_n|\vec{\mu}_{-n}, \vec{Y}) \propto \exp\Bigg[-\frac{1}{0.04}\Bigg[\mu_n - \frac{y_n + \mu_{n-1}}{2}\Bigg]^2\Bigg])$$

2.4

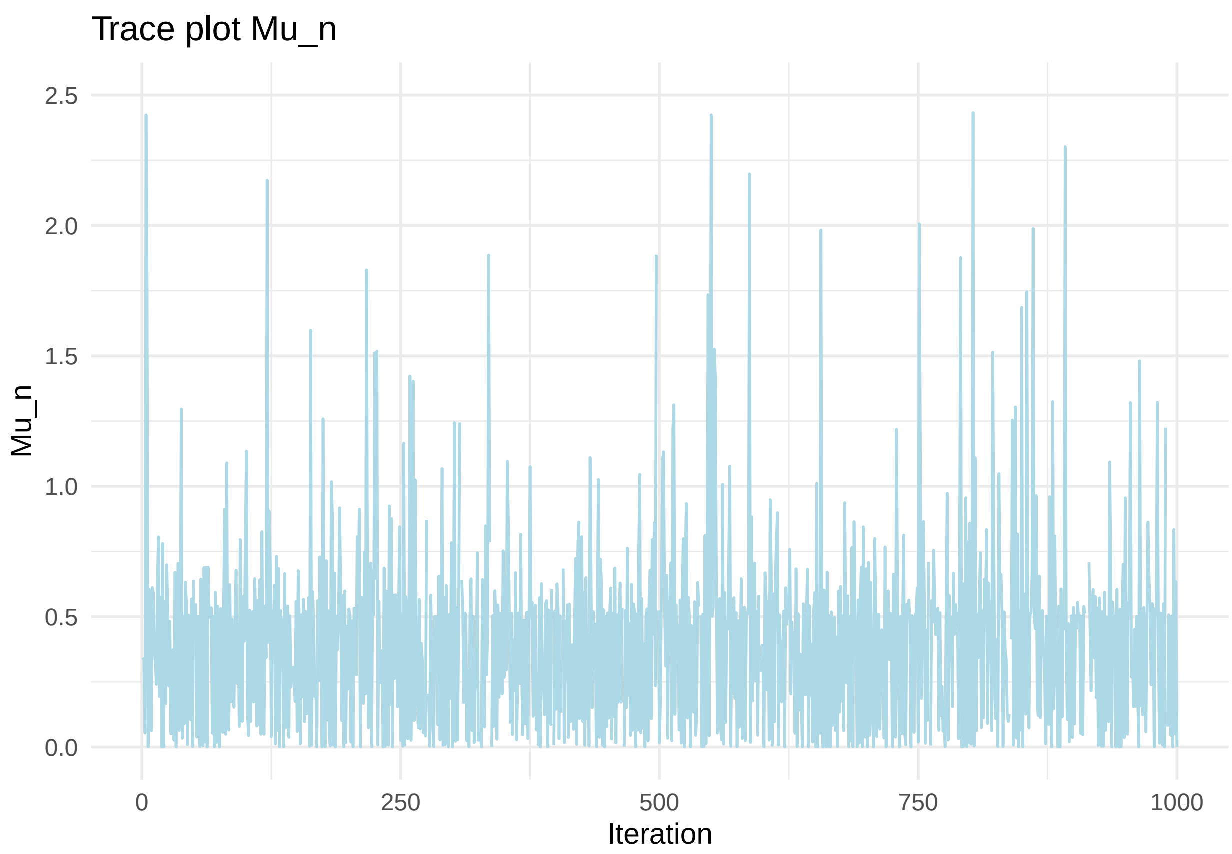


By using the formulae we calculated in 2.3 , we implemented a Gibbs Sampler. The algorithm ran for 1000 times. In 1000 time the Gibbs Sampler has not been able to capture the original data.

2.5:

trace plot of μ_n over its 1000 iterations.

```
mu_samples <- as.data.frame(mu)
gg3 <- ggplot(mu_samples, aes(x = 1:nrow(mu_samples), y = mu_samples[,50])) + geom_line(color="lightblu
  ggtitle("Trace plot Mu_n")+ ylim(0,2.5)+theme_minimal()+xlab("Iteration") +ylab("Mu_n")
gg3
```


Trace plot Mu_n
Mu_n
Iteration
2.5
2.0
1.5
1.0
0.5
0.0
0
250
500
750
1000

Appendix

```
knitr::opts_chunk$set(echo = TRUE)
#-------------------------------------------------------------
#A2
#-------------------------------------------------------------
load("chemical.RData")
require(ggplot2)
data = data.frame("X" = X,
                  "Y" = Y)
x.2 <- lm(Y ~  poly(poly(X, 2)) ,
          data = data)
lg.x = lm(Y ~ log(X),
          data = data)

gg <- ggplot(data, aes(x = X, y = Y)) +
  geom_point(color="purple")+
  ggtitle("concentration of a certain chemicals in water")+
  geom_line(aes(x = X, y = predict(x.2), colour = "x^2"))  +
  geom_line(aes(x = X, y = predict(lg.x), colour = "log (x)"))+  theme_minimal() +
  xlab("Day") + ylab("Concentration")

gg
## the model is noisy

f.MCMC.Gibbs <- function (nsteps , mu0) {
   d <- length ( mu0 )
  mu <- matrix (0, nrow =nsteps , ncol = d)
  mu [1, ] <- mu0
  Y <- matrix (0, nrow =nsteps , ncol = d)
  Y[1, ] <- sapply (X=mu0 , FUN = function (mu){
    y <- rnorm (1, mean =mu , sd =0.2)
    return (y)
  })
  k = 1


  k = 1
  while (k < nsteps ) {

    #i=1
    mu[k+1, 1] <- exp (-1/ 0.04 *
                      (mu[k, 1] -(Y[k, 1]+ mu[k, 2]) /2) ^2) /( dnorm (Y[k, 1], mean =mu[k, 1],
                                                                          sd =0.2) )
    # for i's between 1 to 50
    for (i in 2: (d -1) ) {
      mu[k+1, i] <- exp (-1/ ((2*0.04 )/3)*
                        (mu[k, i]-(Y[k, i]+ mu[k+1, i -1]+ mu[k, i +1]) /3) ^2) /( dnorm (Y[
        k, i], mean =mu[k, i], sd =0.2) )
    }
    #i=50
    mu[k, d] <- exp (-1/ 0.04 *(mu[k, d]-(Y[k, d]+ mu[k+1, d -1])
                              /2) ^2) /( dnorm (Y[k, d], mean =mu[k, d
```

```
    ], sd =0.2) )

    # update Y
    Y[k+1, ] <- sapply (X=mu[k+1, ], FUN = function (x){
      y <- rnorm (1, mean =x , sd =0.2)
      return (y)
    })

    k <- k+1
  }
  return (mu)
}
set.seed(147)
n <- length(Y)
x0 <- rep (0, n)

nsteps <- 1000
mu <- f.MCMC.Gibbs ( nsteps=nsteps , mu0 = x0 )
predicted_mu <- colMeans (mu)
df <- as.data.frame(data)
df$Prediction <- predicted_mu
gg2 <- ggplot(df, aes(x = X, y = Y, col = "Concentration")) + geom_point() +
  xlab("Day") + ylab("Concentration") +
  ggtitle("concentration of a certain chemicals in water") +
  geom_point(aes(x = X, y = Prediction, col = "Prediction")) + theme_minimal()
gg2
mu_samples <- as.data.frame(mu)
gg3 <- ggplot(mu_samples, aes(x = 1:nrow(mu_samples), y = mu_samples[,50])) + geom_line(color="lightblu
  ggtitle("Trace plot Mu_n")+ ylim(0,2.5)+theme_minimal()+xlab("Iteration") +ylab("Mu_n")
gg3
```