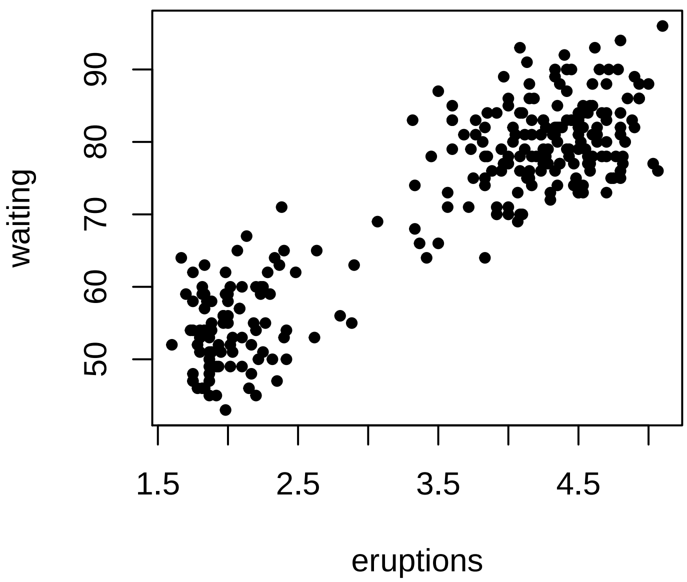
Assignment on MCMC

Sourish

Due Date: 26-March-2021

Consider the `Old Faithful Geyser Data` from the `datasets` package.

```
library(datasets)
plot(faithful,pch=20)
```



In class we modeled the `waiting` with the mixtures of two Gaussian distributions using the Metropolis-Hastings algorithms.

Problem 1

Model the the `waiting` with the mixtures of two Gamma distributions, i.e.,

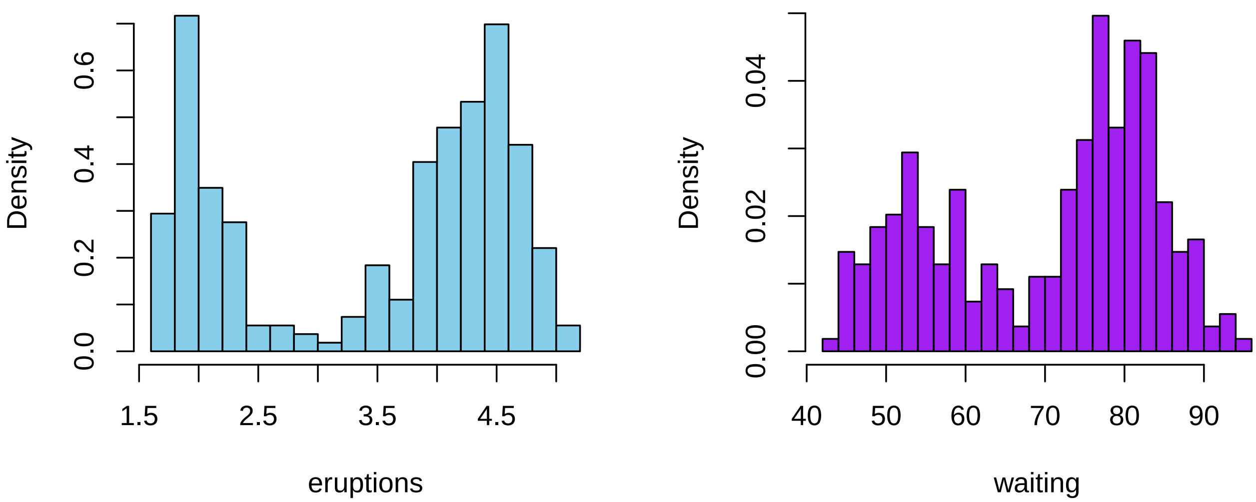
$$f(x) = p.Gamma(\alpha_1, \beta_1) + (1-p).Gamma(\alpha_2, \beta_2),$$

where $0 < p < 1$, $0 < \alpha_i < \infty$, $0 < \beta_i < \infty$, $i = 1, 2$.

a) Simulate `10000` MCMC samples after `1000` burn-in.
b) Implement the Bayesian density estimate
c) Using `10000` simulations of the Bayesian density estimate the following probability

$$\mathbb{P}(72 < \texttt{waiting} < 88 \mid \text{Data})$$

```r
par(mfrow=c(1,2))
eruptions=faithful$eruptions
waiting = faithful$waiting
hist(eruptions,probability = T,xlab = 'eruptions',col='skyblue',main='',nclass=20)
hist(waiting,probability = T,xlab = 'waiting',col='purple',main='',nclass=20)
```



Problem 2

Model the the `eruptions` as the mixtures of one Gamma distributions and one Gaussian distributions i.e.,

$$f(x) = p.Gamma(\alpha, \beta) + (1-p).Normal(\mu, \sigma^2),$$

a) Simulate `10000` MCMC samples after `1000` burn-in.
b) Implement the Bayesian density estimate
c) Using `10000` simulations of the Bayesian density estimate the following probability:

$$\mathbb{P}(0 < \texttt{eruptions} < 3 \mid \text{Data}).$$

d) Obtain the `posterior mode` using the optimization routine, you can use the `optim` routine in `R`.

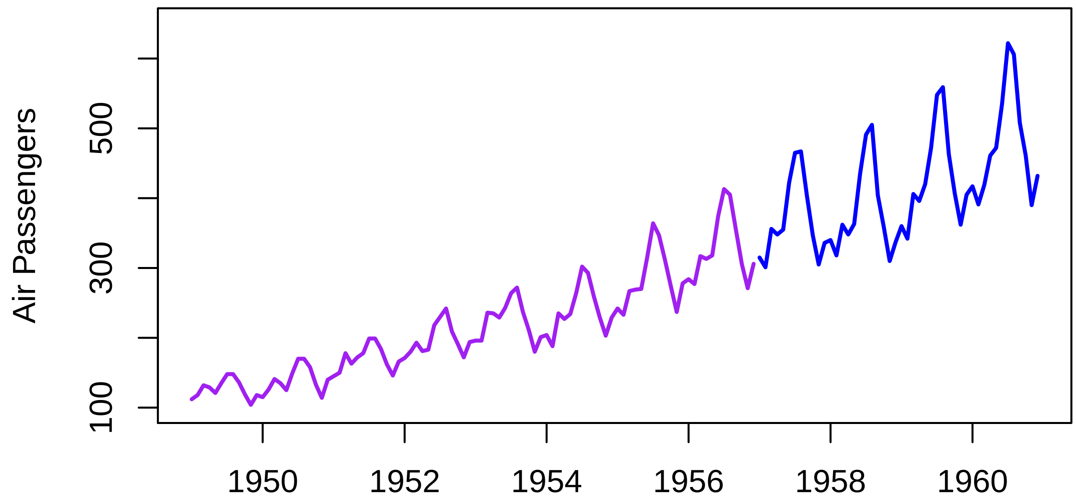
Problem 3

Consider `AirPassengers` data set and use the data up to Dec of 1956 as training data and fit the following model:

$$y(t) = \exp\{\alpha_0 + \alpha_1 t + \beta_1 \sin(\omega t) + \beta_2 \cos(\omega t) + \epsilon_t\},$$

where $\omega = 2\pi$ and $t = \texttt{time} - 1949$, and $\epsilon_t \sim N(0, \sigma^2)$.

```r
data_train=AirPassengers[1:96]
data_test = AirPassengers[97:144]
tme = time(AirPassengers)
tme_min = min(tme)
tme_max =max(tme)
plot(NULL,xlim=c(tme_min,tme_max),ylim=c(100,650)
     ,xlab='',ylab='Air Passengers')
lines(tme[1:96],data_train,lwd=2,col='purple')
lines(tme[97:144],data_test,lwd=2,col='blue')
```



Reppresent the data in `data.frame` as

```
data.train = cbind.data.frame(time=tme[1:96],AirPassengers=data_train)
data.test = cbind.data.frame(time=tme[97:144],AirPassengers=data_test)
data.train$t = data.train$time -1949
data.test$t = data.test$t - 1949
head(data.train)
```

```
##       time AirPassengers          t
## 1 1949.000           112 0.00000000
## 2 1949.083           118 0.08333333
## 3 1949.167           132 0.16666667
## 4 1949.250           129 0.25000000
## 5 1949.333           121 0.33333333
## 6 1949.417           135 0.41666667
```

```
tail(data.train)
```

```
##        time AirPassengers        t
## 91 1956.500           413 7.500000
## 92 1956.583           405 7.583333
## 93 1956.667           355 7.666667
## 94 1956.750           306 7.750000
## 95 1956.833           271 7.833333
## 96 1956.917           306 7.916667
```

```
head(data.test)
```

```
##       time AirPassengers        t
## 1 1957.000           315 8.000000
## 2 1957.083           301 8.083333
## 3 1957.167           356 8.166667
## 4 1957.250           348 8.250000
## 5 1957.333           355 8.333333
## 6 1957.417           422 8.416667
```

```r
tail(data.test)
```

```
##        time AirPassengers        t
## 43 1960.500           622 11.50000
## 44 1960.583           606 11.58333
## 45 1960.667           508 11.66667
## 46 1960.750           461 11.75000
## 47 1960.833           390 11.83333
## 48 1960.917           432 11.91667
```

We can fit the model with `lm` and estimate the `MLE` of the parameters.

```r
omega=2*pi
fit.lm = lm(log(AirPassengers)~t+sin(omega*t)+cos(omega*t),data=data.train)
summary(fit.lm)
```

```
##
## Call:
## lm(formula = log(AirPassengers) ~ t + sin(omega * t) + cos(omega *
##     t), data = data.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.22754 -0.05417  0.01297  0.05833  0.16000
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     4.779379   0.016693 286.317  < 2e-16 ***
## t               0.133880   0.003647  36.709  < 2e-16 ***
## sin(omega * t)  0.037284   0.011907   3.131  0.00233 **
## cos(omega * t) -0.131349   0.011856 -11.078  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08212 on 92 degrees of freedom
## Multiple R-squared:  0.9419, Adjusted R-squared:   0.94
## F-statistic: 497.4 on 3 and 92 DF,  p-value: < 2.2e-16
```

```r
data.test$y_hat = exp(predict(fit.lm,newdata = data.test))
head(data.test)
```

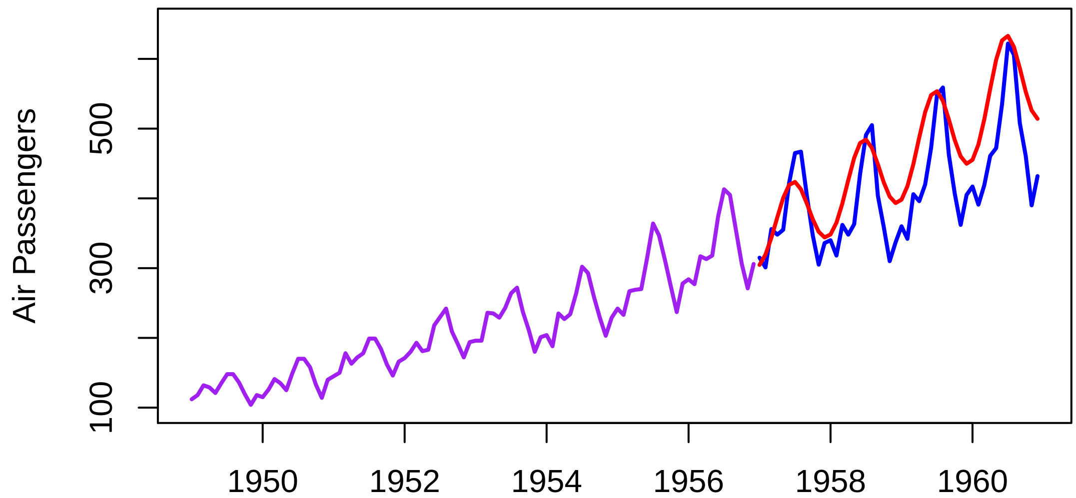
```
##       time AirPassengers        t    y_hat
## 1 1957.000           315 8.000000 304.6222
## 2 1957.083           301 8.083333 319.4077
## 3 1957.167           356 8.166667 343.5556
## 4 1957.250           348 8.250000 372.8497
## 5 1957.333           355 8.333333 400.6193
## 6 1957.417           422 8.416667 419.3050
```

```r
tail(data.test)
```

```
##         time AirPassengers        t    y_hat
## 43 1960.500           622 11.50000 632.9249
## 44 1960.583           606 11.58333 617.2468
## 45 1960.667           508 11.66667 586.8103
## 46 1960.750           461 11.75000 552.9063
## 47 1960.833           390 11.83333 526.1918
## 48 1960.917           432 11.91667 514.0868
```

- We plot the predicted path `y_hat` in the following graph with the red line.

```r
plot(NULL,xlim=c(tme_min,tme_max),ylim=c(100,650)
     ,xlab='',ylab='Air Passengers')
lines(tme[1:96],data_train,lwd=2,col='purple')
lines(tme[97:144],data_test,lwd=2,col='blue')
lines(tme[97:144],data.test$y_hat,lwd=2,col='red')
```



a) Assume $Cauchy(0, 1)$ distribution on $\alpha_0, \alpha_1, \beta_1, \beta_2$ and $Half - Cauchy(0, 1)$ on σ as prior and implement the **posterior mode** using the `optim` function
b) Implement the MCMC sampling. Simulate 10000 MCMC sampling after 1000 burn-in samples.
c) Make the trace-plot of the MCMC samples
d) For each simulated samples, you can estimate predicted path of `AirPassangers` in test data sets and plot the first 100 predicted path on the same plot.
e) Calculate 95% Bayesian interval for the predicted path.
f) Can you come up with a model which can predict the path more accurately?