Problem 3

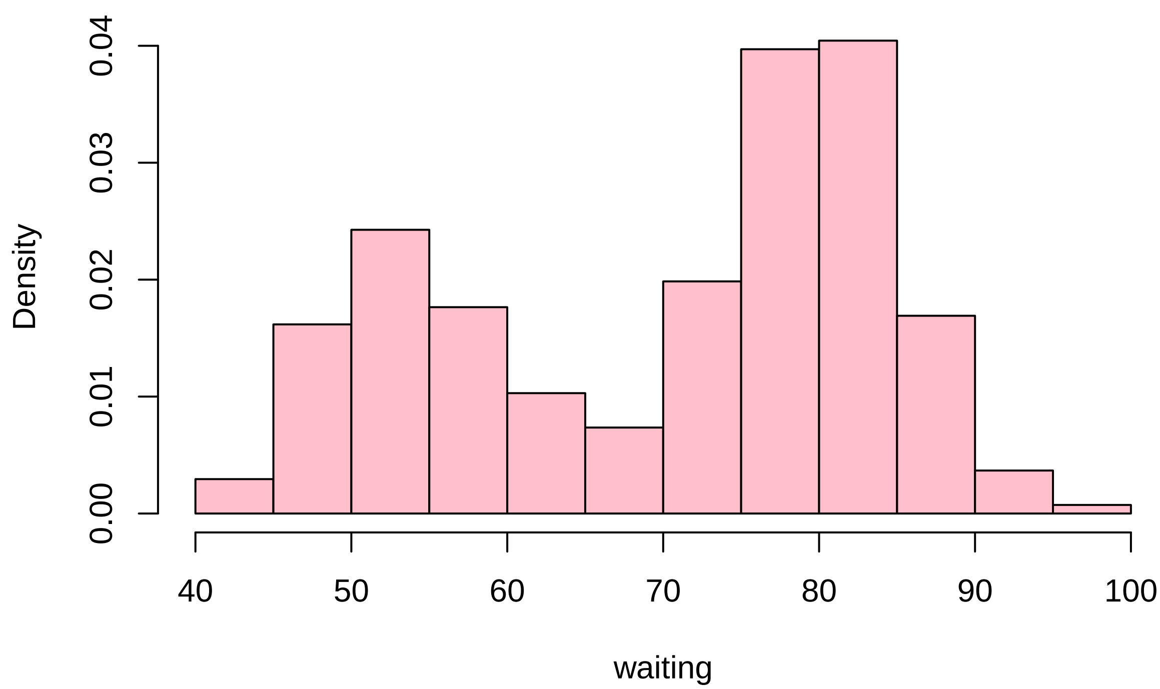
Nidhi

2022-11-16

Problem 3: Analysis of `faithful` datasets.

Consider the `faithful` datasets:

```
attach(faithful)
hist(faithful$waiting,xlab = 'waiting',probability = T,col='pink',main='')
```



Fit following three models using MLE method and calculate **Akaike information criterion** (aka., AIC) for each fitted model. Based on AIC decides which model is the best model? Based on the best model calculate the following probability

$$\mathbb{P}(60 < \texttt{waiting} < 70)$$

(i) **Model 1**:

$$f(x) = p * Gamma(x|\alpha, \sigma_1) + (1-p)N(x|\mu, \sigma_2^2), \quad 0 < p < 1$$

(ii) **Model 2**:

$$f(x) = p * Gamma(x|\alpha_1, \sigma_1) + (1-p)Gamma(x|\alpha_2, \sigma_2), \quad 0 < p < 1$$

(iii) **Model 3**:

$$f(x) = p * logNormal(x|\mu_1, \sigma_1^2) + (1-p)logNormal(x|\mu_1, \sigma_1^2), \quad 0 < p < 1$$

```
attach(faithful)
```

```
## The following objects are masked from faithful (pos = 3):
##
##     eruptions, waiting
```

```
head(faithful)
```

```
##   eruptions waiting
## 1     3.600      79
## 2     1.800      54
## 3     3.333      74
## 4     2.283      62
## 5     4.533      85
## 6     2.883      55
```

```
## parameters = (alpha, sigma1, mu, sigma2, p)

model1 <- function(parameters, data){

  alpha <- parameters[1]
  sigma1 <- parameters[2]
  mu <- parameters[3]
  sigma2 <- parameters[4]
  p <- parameters[5]


  loglikely <- 0
  n = length(data)

  for (i in 1:n){

    first <- p*dgamma(data[i], shape=alpha, scale=sigma1)
    second <- (1-p)*dnorm(data[i], mean=mu, sd=sigma2)
    loglikely <- loglikely + log(first + second)
  }
  return(-loglikely)
}

initial1 <- c(3,0.45,80,9,0.35)

fit1 <- optim(initial1,
             model1,
             data=waiting,
             control=list(maxit=1500))
```

```r
aic1 <- length(fit1$par)*2 + 2*fit1$value

cat("Parameters:", fit1$par, "\n", "AIC for Model 1:", aic1)
```

```
## Parameters: 82.75992 0.6620515 80.16155 5.811328 0.3652798
##  AIC for Model 1: 2076.18
```

```r
################################################################################

model2 <- function(parameters, data){

  alpha1 <- exp(parameters[1])
  sigma1 <- exp(parameters[2])
  alpha2 <- exp(parameters[3])
  sigma2 <- exp(parameters[4])
  p <- exp(parameters[5])/(1 + exp(parameters[5]))


  loglikely <- 0
  n = length(data)

  for (i in 1:n){

    first <- p*dgamma(data[i], shape=alpha1, scale=sigma1)
    second <- (1-p)*dgamma(data[i], shape=alpha2, scale=sigma2)
    loglikely <- loglikely + log(first + second)
  }
  return(-loglikely)
}

initial2 <- c(4, 0, 4, 0, 0.4)

fit2 <- optim(initial2,
              model2,
              data=waiting,
              control=list(maxit=1500))

aic2 <- length(fit2$par)*2 + 2*fit2$value

cat("Parameters:", fit2$par, "\n", "AIC for Model 1:", aic2)
```

```
## Parameters: 5.298486 -0.9128245 4.378315 -0.3715949 0.528566
##  AIC for Model 1: 2076.117
```

```r
################################################################################

model3 <- function(parameters, data){

  mu1 <- parameters[1]
  sigma1 <- exp(parameters[2])
  mu2 <- parameters[3]
  sigma2 <- exp(parameters[4])
```

```r
  p <- exp(parameters[5])/(1 + exp(parameters[5]))


  loglikely <- 0
  n = length(data)

  for (i in 1:n){

    first <- p*dlnorm(data[i], meanlog=mu1, sdlog=sigma1)
    second <- (1-p)*dlnorm(data[i], meanlog=mu2, sdlog=sigma2)
    loglikely <- loglikely + log(first + second)
  }
  return(-loglikely)
}

initial3 <- c(2.76,-2.25,4.4,-2.6,0.35)

fit3 <- optim(initial3,
              model3,
              data=waiting,
              control=list(maxit=1500))

aic3 <- length(fit3$par)*2 + 2*fit2$value

cat("Parameters:", fit3$par, "\n", "AIC for Model 3:", aic3)
```

```
## Parameters: 4.003638 -2.164784 4.384334 -2.663589 -0.4918291
##  AIC for Model 3: 2076.117
```

```r
density <- function(x, theta){

  mu1 <- theta[1]
  sigma1 <- exp(theta[2])
  mu2 <- theta[3]
  sigma2 <- exp(theta[4])
  p <- exp(theta[5])/(1 + exp(theta[5]))

  first <- p*dlnorm(x, meanlog=mu1, sdlog=sigma1)
  second <- (1-p)*dlnorm(x, meanlog=mu2, sdlog=sigma2)
  return(first + second)
}

solution <- integrate(density, 60, 70, fit3$par)

cat("The required probability is:", as.numeric(solution[1]))
```

```
## The required probability is: 0.09112692
```