STAT 510: Homework 06

David Dalpiaz

Due: Monday, March 28, 11:59 PM

Exercise 1 (Method of Moments)

Let $X_1, X_2, \ldots, X_n \sim \text{Gamma}(\alpha, \beta)$. Find the method of moments estimator of α and β.

Solution

We first find the first and second moment of the gamma distribution.

$$\mathbb{E}[X] = \frac{\alpha}{\beta}, \quad \mathbb{V}[X] = \frac{\alpha}{\beta^2}, \quad \mathbb{E}[X^2] = \frac{\alpha^2 + \alpha}{\beta^2}$$

Next, define the first two sample moments.

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad \overline{X^2} = \frac{1}{n}\sum_{i=1}^{n} X_i^2$$

Next, equate the population and sample moments.

$$\frac{\alpha}{\beta} = \bar{X}, \quad \frac{\alpha^2 + \alpha}{\beta^2} = \overline{X^2}$$

Lastly, solve for the parameters.

$$\boxed{\hat{\alpha} = \frac{\bar{X}^2}{\overline{X^2} - \bar{X}^2}} \qquad \boxed{\hat{\beta} = \frac{\bar{X}}{\overline{X^2} - \bar{X}^2}}$$

Alternatively, we could use the second central moment, that is, set the population variance equal to the sample variance. Doing so, we would also obtain a MOM estimator.

$$\boxed{\hat{\alpha} = \frac{\bar{X}^2}{S^2}} \qquad \boxed{\hat{\beta} = \frac{\bar{X}}{S^2}}$$

Exercise 2 ("Numeric" Maximum Likelihood)

Let $X_1, X_2, \ldots, X_n \sim \text{Exponential}(\lambda)$. That is

$$f(x) = \lambda e^{-\lambda x}.$$

Consider each of the following potential values of λ.

```r
lambda = seq(0.001, 1, by = 0.001)
```

We create some data and store it in a vector named `some_data`.

```r
set.seed(42)
some_data = rexp(n = 100, rate = 0.2)
```

For each value λ, calculate the log-likelihood given the data above. Plot the results and report the "MLE" based on this procedure.

Solution

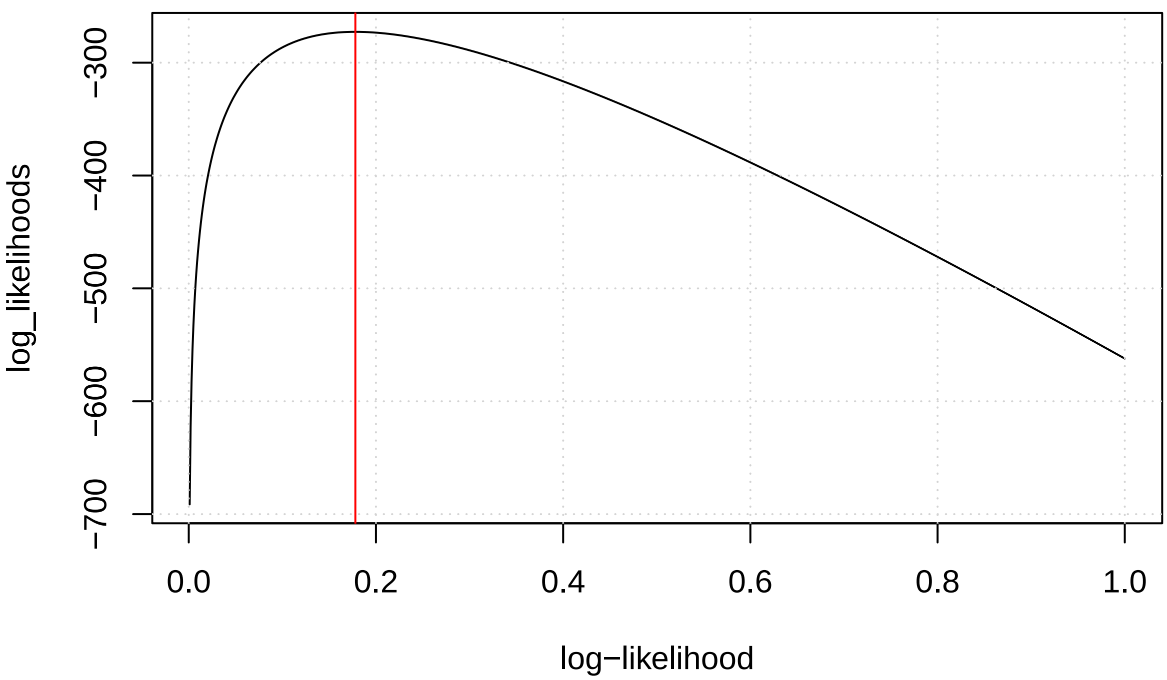
```r
calc_log_lik = function(data, theta) {
 sum(dexp(x = data, rate = theta, log = TRUE))
}
```

```r
log_likelihoods = sapply(lambda, calc_log_lik, data = some_data)
```

```r
# the "MLE"
lambda[which.max(log_likelihoods)]
```

```
## [1] 0.178
```

```r
plot(x = lambda, y = log_likelihoods, type = "l",
     xlab = "log-likelihood")
abline(v = lambda[which.max(log_likelihoods)], col = "red")
grid()
```



Exercise 3 (Estimating Allele Frequency)

In genetics, single nucleotide polymorphisms (SNPs) are locations in the (human) genome that exhibit variation across the population. SNPs cause the differences we see in traits such as hair color. Each SNP typically has two possible alleles – say A and a – and each person's genotype at the SNP is either AA, Aa, or aa, where one allele comes from the person's mother and one from the father. Let X be the number of A

alleles at a particular SNP, and suppose we collect a random sample of people from some population. Under some assumptions (such as "random mating" and "no selection") we may assume that

$$X_1, X_2, \ldots, X_n \ \sim \ \text{Binom}(2, p),$$

where p is called the *allele frequency* of allele A. What is the maximum likelihood **estimator** of p? What is the maximum likelihood **estimate** of the allele frequency of allele A if our sample consists of five people with genotypes

$$AA, aa, Aa, aa, Aa$$

at this particular SNP?

Solution

The likelihood is:

$$\begin{aligned} L(p) &= \prod_{i=1}^{n} \binom{2}{x}_i p^{x_i} (1-p)^{2-x_i} \\ &= \left\{ \prod_{i=1}^{n} \binom{2}{x}_i \right\} p^{\sum_{i=1}^{n} x_i} (1-p)^{2n - \sum_{i=1}^{n} x_i} \end{aligned}$$

The log-likelihood is:

$$\log L(p) = \log \left\{ \prod_{i=1}^{n} \binom{2}{x}_i \right\} + \log(p) \sum_{i=1}^{n} x_i + \log(1-p) \left(2n - \sum_{i=1}^{n} x_i \right)$$

The derivative of the log-likelihood is:

$$\frac{d}{dp} \log L(p) = \frac{\sum_{i=1}^{n} x_i}{p} - \frac{2n - \sum_{i=1}^{n} x_i}{1-p}$$

Thus, setting that equal to 0 yields:

$$\frac{\sum_{i=1}^{n} x_i}{p} = \frac{2n - \sum_{i=1}^{n} x_i}{1-p} \iff \sum_{i=1}^{n} x_i - p \sum_{i=1}^{n} x_i = 2np - p \sum_{i=1}^{n} x_i \iff \frac{1}{2n} \sum_{i=1}^{n} x_i = p$$

Thus, our candidate for the MLE is

$$\hat{p} = \frac{1}{2n} \sum_{i=1}^{n} x_i.$$

To verify that this is a maximum, we calculate:

$$\frac{d^2}{dp^2} \log L(p) = -\frac{\sum_{i=1}^{n} x_i}{p^2} - \frac{2n - \sum_{i=1}^{n} x_i}{(1-p)^2}$$

which is always negative, so we have found a maximum, and we can conclude the MLE is

$$\boxed{\hat{p} = \frac{1}{2n}\sum_{i=1}^{n} x_i}$$

Converting the observed genotypes into random variables $X_1, \ldots, X_5$, we have data: $2, 0, 1, 0, 1$. Thus, the maximum likelihood estimate of the allele frequency is:

$$\boxed{\hat{p} = \frac{1}{10} \cdot 4 = 0.4}$$

Exercise 4 (Corn!)

Consider two corn varieties, A and B, both grown in the Morrow Plots. Illinois is very serious about our corn. Rumor has it, if a student is found trespassing in the Morrow Plots, they will be expelled...

Suppose that $X_1, X_2, \ldots, X_n$, representing yields per acre for corn variety A, constitute a random sample from a normal distribution with mean μ_1 and variance θ. (In more usual notation, $\theta = \sigma^2$, but we are using θ here to make the notation easier in this problem.) Also, $Y_1, Y_2, \ldots, Y_m$, representing yields for corn variety B, constitute a random sample from a normal distribution with mean μ_2 and variance θ. If the X_i and Y_j are all mutually independent, find the maximum likelihood **estimator** for the common variance θ. Assume that μ_1 and μ_2 are **known.**

Solution

The likelihood is:

$$\begin{aligned} L(\theta) &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\theta}} \exp\left\{-\frac{(x_i - \mu_1)^2}{2\theta}\right\} \cdot \prod_{j=1}^{m} \frac{1}{\sqrt{2\pi\theta}} \exp\left\{-\frac{(y_j - \mu_2)^2}{2\theta}\right\} \\ &= \frac{1}{(2\pi)^{\frac{n}{2}}} \frac{1}{\theta^{\frac{n}{2}}} \exp\left\{-\frac{1}{2\theta}\sum_{i=1}^{n}(x_i - \mu_1)^2\right\} \frac{1}{(2\pi)^{\frac{m}{2}}} \frac{1}{\theta^{\frac{m}{2}}} \exp\left\{-\frac{1}{2\theta}\sum_{j=1}^{m}(y_j - \mu_2)^2\right\} \end{aligned}$$

The log-likelihood is:

$$\log L(\theta) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log\theta - \frac{1}{2\theta}\sum_{i=1}^{n}(x_i - \mu_1)^2 - \frac{m}{2}\log(2\pi) - \frac{m}{2}\log\theta - \frac{1}{2\theta}\sum_{j=1}^{m}(y_j - \mu_2)^2$$

The derivative of the log-likelihood is:

$$\begin{aligned} \frac{d}{d\theta}\log L(\theta) &= -\frac{n}{2} \cdot \frac{1}{\theta} + \frac{1}{2\theta^2}\sum_{i=1}^{n}(x_i - \mu_1)^2 - \frac{m}{2} \cdot \frac{1}{\theta} + \frac{1}{2\theta^2}\sum_{j=1}^{m}(y_j - \mu_2)^2 \\ &= -\frac{n+m}{2} \cdot \frac{1}{\theta} + \frac{1}{2\theta^2}\left\{\sum_{i=1}^{n}(x_i - \mu_1)^2 + \sum_{j=1}^{m}(y_j - \mu_2)^2\right\} \end{aligned}$$

Setting that equal to 0 and solving, we get:

$$
\begin{aligned}
\frac{n+m}{2}\cdot\frac{1}{\theta} &= \frac{1}{2\theta^2}\left\{\sum_{i=1}^{n}(x_i-\mu_1)^2+\sum_{j=1}^{m}(y_j-\mu_2)^2\right\} \\
\iff \theta &= \frac{1}{n+m}\left\{\sum_{i=1}^{n}(x_i-\mu_1)^2+\sum_{j=1}^{m}(y_j-\mu_2)^2\right\}
\end{aligned}
$$

Thus our candidate is

$$
\hat{\theta} = \frac{1}{n+m}\left\{\sum_{i=1}^{n}(x_i-\mu_1)^2+\sum_{j=1}^{m}(y_j-\mu_2)^2\right\}.
$$

Taking the second derivative of the log-likelihood, we have:

$$
\begin{aligned}
\frac{d^2}{d\theta^2}\log L(\theta) &= \frac{n+m}{2}\cdot\frac{1}{\theta^2} - 2\cdot\frac{1}{2\theta^3}\left\{\sum_{i=1}^{n}(x_i-\mu_1)^2+\sum_{j=1}^{m}(y_j-\mu_2)^2\right\} \\
&= \frac{n+m}{2}\cdot\frac{1}{\theta^2} - \frac{1}{\theta^3}(n+m)\hat{\theta} \\
&= (n+m)\left(\frac{1}{2\theta^2}-\frac{\hat{\theta}}{\theta^3}\right)
\end{aligned}
$$

so that:

$$
\frac{d^2}{d\theta^2}\log L\left(\hat{\theta}\right) = (n+m)\left(\frac{1}{2\hat{\theta}^2}-\frac{\hat{\theta}}{\hat{\theta}^3}\right) = (n+m)\left(\frac{1}{2\hat{\theta}^2}-\frac{2}{2\hat{\theta}^2}\right) = -(n+m)\frac{1}{2\hat{\theta}^2} < 0
$$

Thus, we have found our maximum, and the MLE is:

$$
\boxed{\hat{\theta} = \frac{1}{n+m}\left\{\sum_{i=1}^{n}(x_i-\mu_1)^2+\sum_{j=1}^{m}(y_j-\mu_2)^2\right\}}
$$

Exercise 5 (A "Fun and Easy" MLE)

Let X_1, X_2 be independent random variables from Poisson distributions with parameters λ_1 and λ_2 respectively. That is

$$
f(x_i) = \frac{\lambda_i^{x_i}e^{-\lambda_i}}{x_i!}, \quad x_i = 0, 1, 2, \ldots
$$

- When $\theta = -1$, we have $\lambda_1 = 2.3$ and $\lambda_2 = 5.6$.
- When $\theta = 1$, we have $\lambda_1 = 4.4$ and $\lambda_2 = 3.2$.

Suppose we observe $x_1 = 3$ and $x_2 = 4$. Based on this data, what is the maximum likelihood estimate of θ? Justify your answer!

Solution

$$\mathcal{L}(\theta = -1) = \frac{2.3^{x_1} e^{-2.3}}{x_1!} \cdot \frac{5.6^{x_1} e^{-5.6}}{x_2!}$$

$$\mathcal{L}(\theta = 1) = \frac{4.4^{x_1} e^{-4.4}}{x_1!} \cdot \frac{3.2^{x_1} e^{-3.2}}{x_2!}$$

```
prod(dpois(x = c(3, 4), lambda = c(2.3, 5.6)))
```

```
## [1] 0.03080681
```

```
prod(dpois(x = c(3, 4), lambda = c(4.4, 3.2)))
```

```
## [1] 0.03104255
```

After plugging in the observed values, we see that the likelihood is maximized when $\theta = 1$.

$$\boxed{\hat{\theta} = 1}$$

Exercise 6 (Method of Moments with Uniform)

Let $X_1, X_2, \ldots, X_n \sim \text{Uniform}(a, b)$ where $a < b$. Find the method of moments estimators for a and b.

Solution

First, note that

$$\mathbb{E}[X] = \frac{a + b}{2}.$$

Next, we see that

$$\mathbb{E}[X^2] = \int_a^b \frac{x^2}{b - a} dx = \frac{b^2 + ab + a^2}{3}$$

Next, equate the population and sample moments.

$$\frac{a + b}{2} = \bar{X}, \quad \frac{b^2 + ab + a^2}{3} = \overline{X^2}$$

Lastly, solve for the parameters while noting that $a < b$.

$$\boxed{\hat{a} = \bar{X} - \sqrt{3}\sqrt{\overline{X^2} - \bar{X}^2}} \qquad \boxed{\hat{b} = \bar{X} + \sqrt{3}\sqrt{\overline{X^2} - \bar{X}^2}}$$

Exercise 7 (Maximum Likelihood with Uniform)

Let $X_1, X_2, \ldots, X_n \sim \text{Uniform}(a, b)$ where $a < b$. Find the maximum likelihood estimators for a and b. Also find the MLE of

$$\tau = \int x dF(x)$$

Solution

Via the intuition developed in lecture, we immediately conclude that

$$\boxed{\hat{a} = \min(X_1, X_2, \ldots, X_n)} \qquad \boxed{\hat{b} = \max(X_1, X_2, \ldots, X_n)}$$

We then use the invariance of the MLE to obtain the MLE for τ.

$$\boxed{\hat{\tau} = \frac{\hat{a} + \hat{b}}{2}}$$

Exercise 8 (Maximum Likelihood versus Empirical Distribution)

Let $X_1, X_2, \ldots, X_n \sim \text{Poisson}(\lambda)$ and consider the following data generated according to this model:

```
set.seed(7)
some_data = rpois(n = 20, lambda = 2)
some_data
```

```
##  [1] 6 1 0 0 1 3 1 5 1 2 1 1 3 0 2 0 2 0 6 1
```

Consider two probabilities:

- $P(X > 3)$
- $P(X > 7)$

For both:

- Provide an estimate using maximum likelihood.
- Provide an estimate using the empirical distribution.
- Provide the true value.

Solution

```
c(
  mle = 1 - ppois(3, lambda = mean(some_data)),
  empirical = mean(some_data > 3),
  true = 1 - ppois(3, lambda = 2)
)
```

```
##       mle empirical      true
## 0.1087084 0.1500000 0.1428765
```

```
c(
  mle = 1 - ppois(7, lambda = mean(some_data)),
  empirical = mean(some_data > 7),
  true = 1 - ppois(7, lambda = 2)
)
```

```
##          mle    empirical         true
## 0.0005615272 0.0000000000 0.0010967190
```

Exercise 9 (Parametric Bootstrap)

Let $X_1, X_2, \ldots, X_n \sim \text{Normal}(\mu, \sigma)$. Given the data below, find the MLE of $P(X > 5)$. Use the parametric bootstrap to estimate the standard error of this MLE.

```r
set.seed(42)
some_data = rnorm(n = 100, mean = 4, sd = 2)
```

Solution

```r
# calculate MLE for mu, sigma, and parameter of interest
mu_hat = mean(some_data)
sd_hat = sqrt(mean((some_data - mu_hat) ^ 2))
mle = pnorm(q = 5, mean = mu_hat, sd = sd_hat, lower.tail = FALSE)
```

```r
# function to generate parametric bootstrap replicate
gen_boot_rep = function(n, mu, sd) {
  boot_resample = rnorm(n = n, mean = mu, sd = sd)
  mu_hat = mean(boot_resample)
  sd_hat = sqrt(mean((boot_resample - mu_hat) ^ 2))
  boot_replicate = pnorm(q = 5, mean = mu_hat, sd = sd_hat, lower.tail = FALSE)
  return(boot_replicate)
}
```

```r
# generate bootstrap replicates
boot_replicates = replicate(
  n = 5000,
  gen_boot_rep(n = 100, mu = mu_hat, sd = sd_hat)
)
```

```r
# calculate SE
sqrt(mean((boot_replicates - mle) ^ 2))
```

```
## [1] 0.03822419
```

Exercise 10 (Numeric MLE)

Let $X_1, X_2, \ldots, X_n \sim \text{Exponential}(\lambda)$. That is

$$f(x) = \lambda e^{-\lambda x}.$$

We create some data and store it in a vector named `some_data`.

```r
set.seed(42)
some_data = rexp(n = 100, rate = 0.2)
```

Use Newton–Raphson to find the MLE numerically. (Note that numerical optimization is not actually necessary in this example, so you can easy check your work analytically.) Consider three different initial values for λ:

- `1e-10`
- `0.3`
- `0.5`

Use any reasonable stopping criteria. Comment on the differences based on initial values.

Solution

```r
loglik_prime = function(lambda, data) {
  length(data) / lambda - sum(data)
}
```

```r
loglik_primeprime = function(lambda, data) {
  -length(data) / lambda ^ 2
}
```

```r
do_nr = function(theta_initital) {

  theta_previous = Inf
  theta = theta_initital

  while (abs(theta - theta_previous) > .Machine$double.eps) {

    theta_previous = theta
    theta = theta_previous - loglik_prime(lambda = theta_previous, data = some_data) /
      loglik_primeprime(lambda = theta_previous, data = some_data)

    if (theta == -Inf | theta == Inf) {
      warning("Algorithm did not converge. \n")
      return(theta)
    }

  }

  return(theta)

}
```

```r
do_nr(theta_initital = 1e-10)
```

```
## [1] 0.1778868
```

```r
do_nr(theta_initital = 0.3)
```

```
## [1] 0.1778868
```

```r
do_nr(theta_initital = 0.5)
```

```
## Warning in do_nr(theta_initital = 0.5): Algorithm did not converge.
```

```
## [1] -Inf
```

Both `1e-10` and `0.3` converge to the MLE, but `0.3` converges faster. Using an intial value of `0.5` leads to convergence issues. Thus, it is important with Newton–Raphson to select good initial guesses. Often this is done by using the MoM estiamtor as the initial guess.

Exercise 11 (EM for Mixture of Normals)

This is a challenge question. You will likely need to do some "Googling" to complete this question.

The following code generates data according to a mixture model. In particular, we have a mixture of three univariate normals.

```r
mu = c(0, 5, 10)
sd = sqrt(c(2, 1, 0.5))
mix = c(0.7, 0.1, 0.2)
```

```r
set.seed(42)
components = sample(1:3, prob = mix, size = 1000, replace = TRUE)
some_data = rnorm(n = 1000, mean = mu[components], sd = sd[components])
```

Use the EM algorithm assuming a three component mixture of normals to estimate the mixing parameters, means, and standard deviations. State how you initialized the parameters, and how you decided to stop iterating. Plot a histogram of the data. Overlay both the true and estimated densities. Do not use any built in functions or packages for fitting mixture models except to check your work.

Solution

$$\theta = (\pi_1, \pi_2, \pi_3, \mu_1, \mu_2, \mu_3, \sigma_1, \sigma_2, \sigma_3)$$

$$\boldsymbol{X} = (X_1, X_2, \dots X_n)$$

E Step

$$r_k^{(i)} = P(Z_i = k \mid \boldsymbol{X}, \theta) = \frac{\pi_K \cdot f\left(x^{(i)}; \mu_k, \sigma_k\right)}{\sum_{j=1}^{3} \pi_j \cdot f\left(x^{(i)}; \mu_j, \sigma_j\right)}$$

M Step

$$\pi_k = \frac{1}{n} \sum_{i=1}^{n} r_k^{(i)}$$

$$\mu_k = \frac{1}{\sum_{i=1}^{n} r_k^{(i)}} \sum_{i=1}^{n} r_k^{(i)} x^{(i)}$$

$$\sigma_k = \sqrt{\frac{1}{\sum_{i=1}^{n} r_k^{(i)}} \sum_{i=1}^{n} r_k^{(i)} \left(x^{(i)} - \mu_k\right)^2}$$

```
# function to calculate desnity of mixture of normals
mix_density = function(x, mu = mu, sd = sd, mix = mix) {
  sum(mix * dnorm(x, mean = mu, sd = sd))
}
```

```
# function to calculate the posterior probability of each component
mix_post = function(x, mu, sd, mix) {
  mix * dnorm(x, mean = mu, sd = sd) / sum(mix * dnorm(x, mean = mu, sd = sd))
}
```

```
# function to calculate the log-likelihood of a mixture of three normals
mix_loglik = function(data, mu, sd, mix) {
  sum(log(sapply(data, mix_density, mu = mu, sd = sd, mix = mix)))
}
```

```
# function to perform EM algorithm for mixture of three normals
do_em = function(data, mu_init, sd_init, mix_init, verbose = FALSE) {

  # get sample size
  n = length(data)

  # set initial values of mu, sd, mix, and the log-likelihood
  mu = mu_init
  sd = sd_init
  mix = mix_init
```

```
  loglik = mix_loglik(data = data, mu = mu, sd = sd, mix = mix)

  # set initial values of mu, sd, mix, and the log-likelihood
  # set such that while loop will be entered at least once
  mu_prev = -Inf
  sd_prev = -Inf
  mis_prev = -Inf
  loglik_previous = Inf

  # update parameter values
  while (abs(loglik - loglik_previous) > .Machine$double.eps) {

    # store current value of loglik as previous
    loglik_previous = loglik

    # get posterior probabilities
    post = t(sapply(data, mix_post, mu = mu, sd = sd, mix = mix))

    # update mixing parameters
    n_k = colSums(post)
    mix = n_k / n

    # update mu
    mu[1] = (1 / n_k[1]) * sum(post[, 1] * data)
    mu[2] = (1 / n_k[2]) * sum(post[, 2] * data)
    mu[3] = (1 / n_k[3]) * sum(post[, 3] * data)

    # update sigma
    sd[1] = sqrt((1 / n_k[1]) * sum(post[, 1] * (data - mu[1]) ^ 2))
    sd[2] = sqrt((1 / n_k[2]) * sum(post[, 2] * (data - mu[2]) ^ 2))
    sd[3] = sqrt((1 / n_k[3]) * sum(post[, 3] * (data - mu[3]) ^ 2))

    # update lok-likelihood
    loglik = mix_loglik(data = data, mu = mu, sd = sd, mix = mix)

    if (verbose) {
      print(loglik)
    }

  }

  # return estimated paramters
  return(list(mu = mu, sd = sd, mix = mix))

}
```

```
# check data to determine reasonable values for init mu
range(some_data)
```

```
## [1] -4.268001 11.855091
```

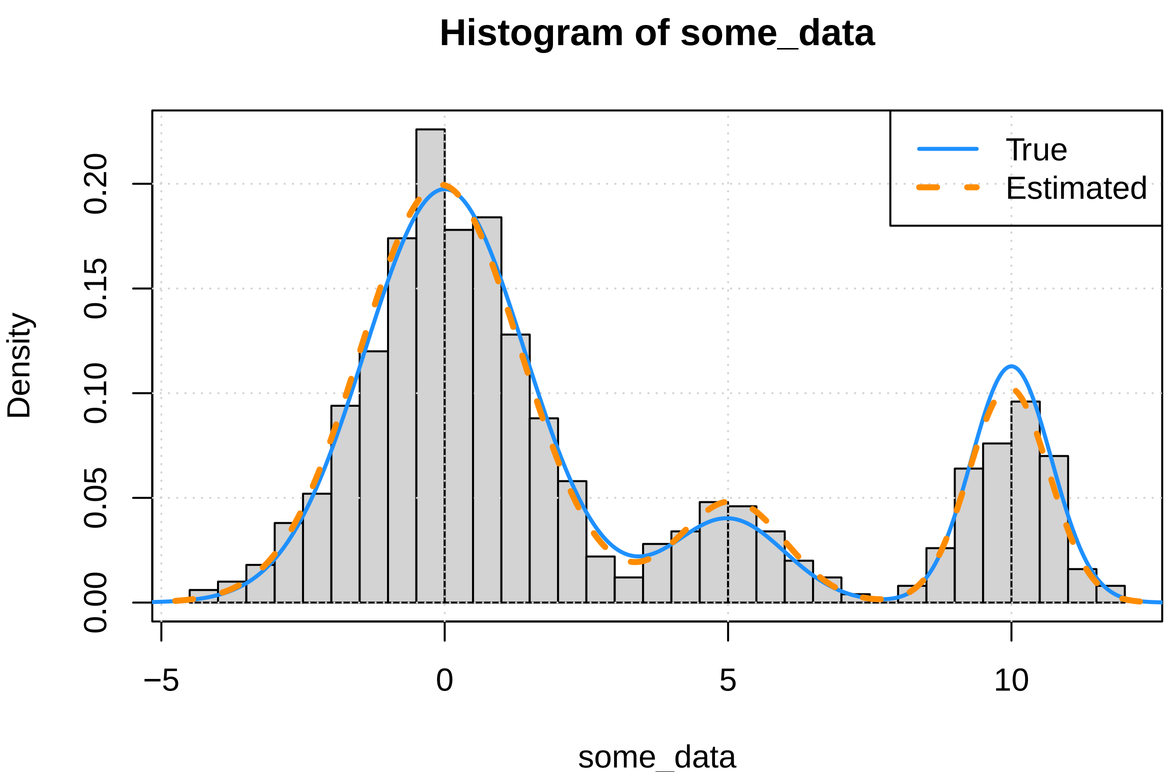
```
mean(range(some_data))
```

```
## [1] 3.793545
```

```r
# run EM with "reasonable" mu initial values and "flat" sigma and mixture values
results = do_em(data = some_data,
      mu_init = c(-4, 4 , 11),
      sd_init = c(1, 1, 1),
      mix_init = c(0.34, 0.33, 0.33))
```

```r
# calculate values for plotting
x = seq(-20, 20, by = 0.01)
dens_true = sapply(x, mix_density, mu = mu, sd = sd, mix = mix)
dens_esti = sapply(x,
                   mix_density,
                   mu = results$mu,
                   sd = results$sd,
                   mix = results$mix)
```

```r
hist(some_data, probability = TRUE, breaks = 50)
box()
grid()
lines(x, dens_true, col = "dodgerblue", lwd = 2)
lines(x, dens_esti, col = "darkorange", lwd = 3, lty = 2)
legend(
  "topright",
  legend = c("True", "Estimated"),
  lty = c(1, 2),
  lwd = c(2, 3),
  col = c("dodgerblue", "darkorange")
)
```




```r
results
```

```
## $mu
```

```
## [1] -0.07611362  5.06703238  9.95566175
##
## $sd
## [1] 1.4085045 0.9377182 0.7085354
##
## $mix
## [1] 0.7046813 0.1131016 0.1822171
```