EM Algorithm for Fish Length Data with Some Unknown Age Groups

Tianyi Han, Meagan Neves, Holly Stoner, Erna Kuginyte

28/10/2022

MT4113 Group Assignnment

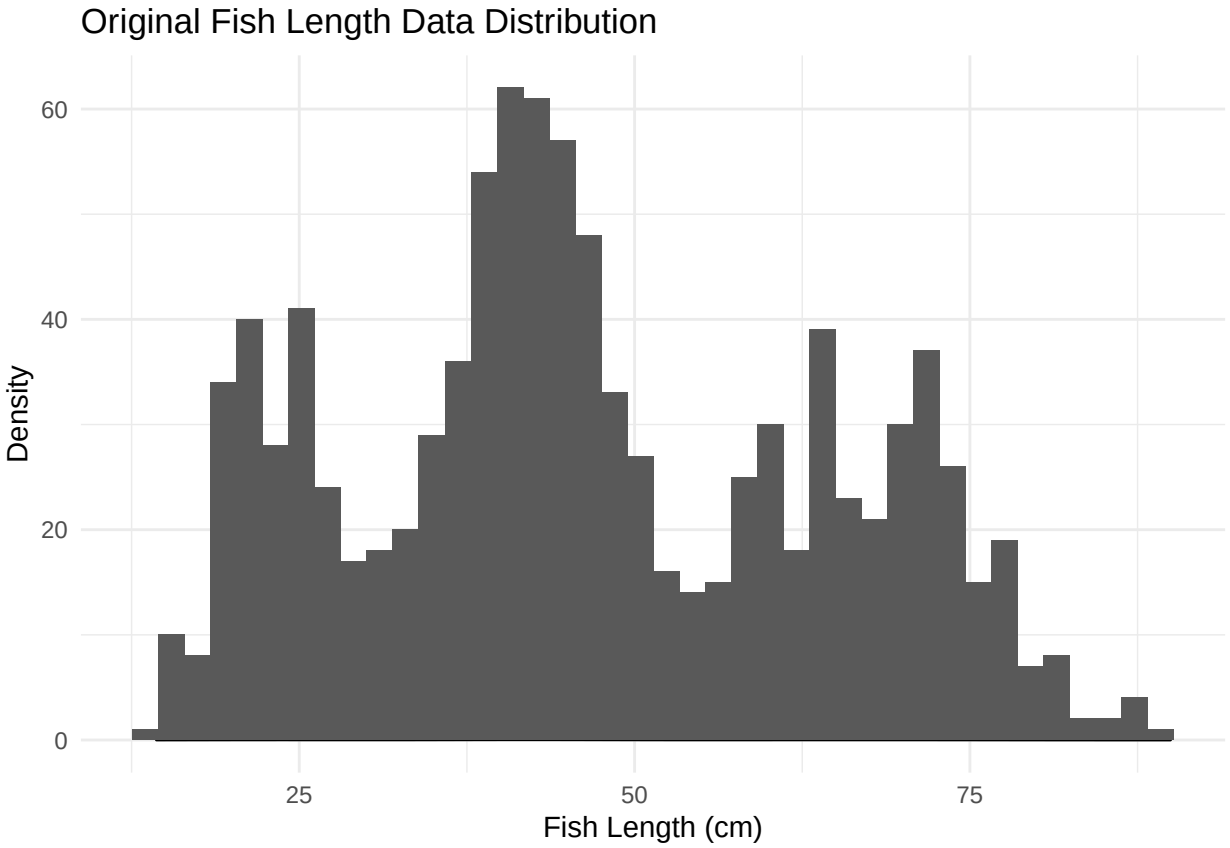
Introduction

The data we received for this project is a data frame of 1000 observations of fish with 3 types of information: ID (FishID), length of the fish (Length), and the age group they belong to (Age). 100 of the observations have been determined to belong to a known age group - 1, 2, or 3. The age group of the other 900 observations are unknown. The following are some examples of observations in FishLengths.RData and the histogram of all fish lengths.

Table 1: First few rows from the original data frame

	FishID	Length	Age
776	76	58.48	NA
562	62	37.88	NA
387	87	49.18	NA
14	14	22.17	1
757	57	64.77	3
628	28	41.99	NA

Since the age of a fish and its length are correlated, it is a reasonable approach to determine the age groups of fish populations by analyzing the distribution of their lengths.



The histogram above presents 3 peaks among all fish lengths, indicating that there could be 3 main length groups and three corresponding age groups within the 1000 observations. We assume that lengths within each group are normally distributed and adopt a Gaussian mixture model for the data. The research question of this project is to find estimates of the mean (mu) and standard deviation (sigma) of the normal distributions of the 3 age groups and the probability (lambda) for a fish to be in each of the groups.

Implementation of EM algorithm

We applied the Expectation-Maximization (EM) algorithm in order to find the most probable distribution of the age groups.[1] The first step is to assign the fish of unknown age into one of the age groups based on their length and the estimated parameters of the normal distributions for each age group. As a start, data from fish of known age groups are used to compute mu, sigma, and lambda for the 3 age groups. In the following iterations, estimates from the previous iterations are used. The second step is to calculate the posterior probability of a fish belonging to each age group given the observation of its length. The third step is to make new estimates of mu, sigma, and lambda based on the posterior probabilities. In the final step, a log-likelihood is calculated using the new estimates. Convergence is tested by comparing the log-likelihood of the current iteration and the previous. If the difference is smaller than a desired tolerance value, which is typically small, then convergence is met. These are the main components in a single iteration of the EM algorithm. The four steps will keep repeating until either convergence is met or reaches a maximum of iterations.

[1]This sentence will be printed as a footnote.

Original EM Algorithm Flow Chart

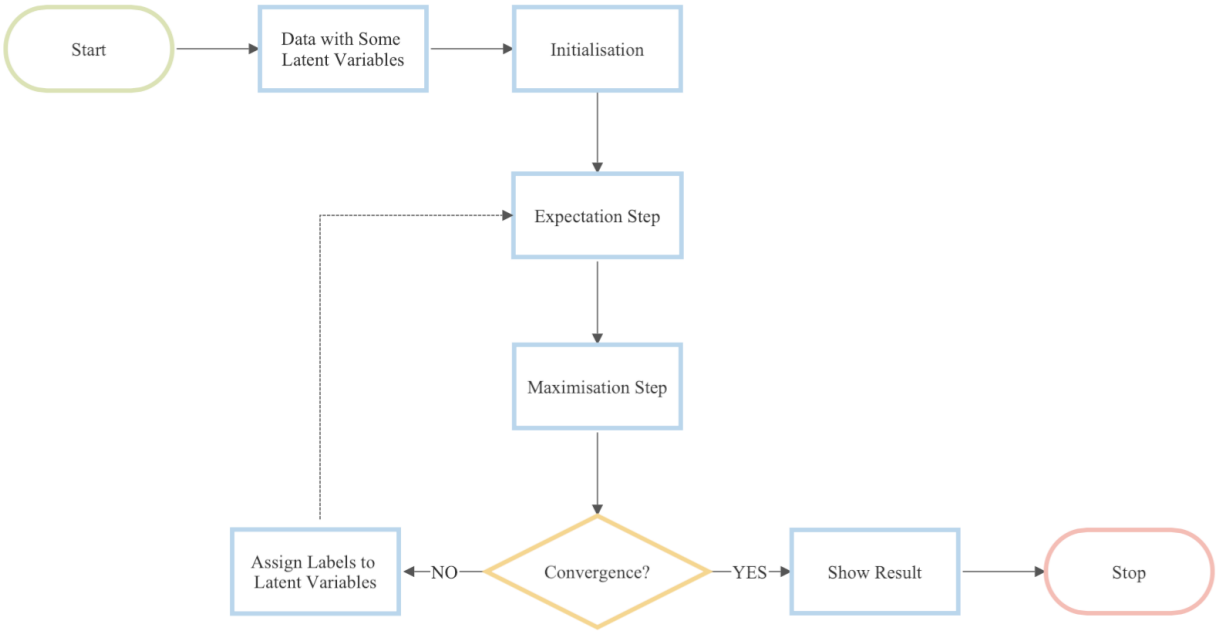


Figure 1: EM Flow Chart

Pseudo-Code

```
# Pseudo code for function teamEM

# Inputs:
#   data - data frame with N observations and 3 columns: FishID, Length,  Age (k numbered groups for known, and NA for
    unknown).
#   epsilon - desired tolerance value for convergence.
#   maxit - maximum number of iterations.

# Outputs:
#   inits - dataframe with mu, sigma, lambda for k age groups.
#   posterior - posterior probabilities of N observations in k age groups.
#   estimates - new mu, sigma, lambda for k age groups.
#   converged - TRUE if convergence is met before the maximum iteration, FALSE otherwise.
#   likelihood - vector of length equals to number of iterations, maximum
#   length equals to maxit.


teamEM <- function(data, epsilon = 1e-08, maxit = 1000) {

  # 1. Initialisation
  # Assign labels to fish with latent age variables based on data from fish
  # with known age:
  # Take the the values of the probability density functions for each fish length, compare them, and assign the age groups
in
  #   accordance with the highest pdf value.
  # Take the data with the known and assigned values and calculate mean (mu), sd (sigma) for each age group.
  # Calculate probability (lambda) for a fish to be in each age group.

  # Repeating expectation, maximization and testing of convergence.
  # Create a for-loop with maximum iterations of maxit times.
  for (i in 1:maxit) {

    # 2. Expectation
    # Calculate probabilities for fish belonging to each age group given the observed lengths.
    # Use inits as a start and then use estimates from previous iterations

    # 3. Maximization
    # Calculate new estimates for mean, sd, and probability for each age group based on the posterior probabilities.

    # 4. Testing Convergence
    # Calculate the log-likelihood with parameters of this iteration.
    # If the difference between log-likelihood of this iteration and the previous is smaller than tolerance value, then
convergence is met.
    # Break the for-loop if convergent
    if (log(ith likelihood) - log((i-1)th likelihood) < epsilon)
      break.
    # If not convergent, reassign Age groups to latent variables in the dataframe the same way it has been done at the
    #   Initialisation step.
    #   Go back to the E step. Maximum amount of iterations - maxit.
  }
}
```

Figure 2: Pseudo-Code

Testing

Initialisation Function Test

Run two sample ks.test() to compare the initial known length values of each fish age group to the ones that were assigned by the initialise() function.

```
## Warning in ks.test.default(x = lengths_2, y = assigned_2): p-value will be
## approximate in the presence of ties
```

```
## Warning in ks.test.default(x = lengths_3, y = assigned_3): p-value will be
## approximate in the presence of ties
```

```
## [1] "Two-sample Kolmogorov-Smirnov tests don't indicate statistically significant difference in sampl
```

teamEM.r Function Test

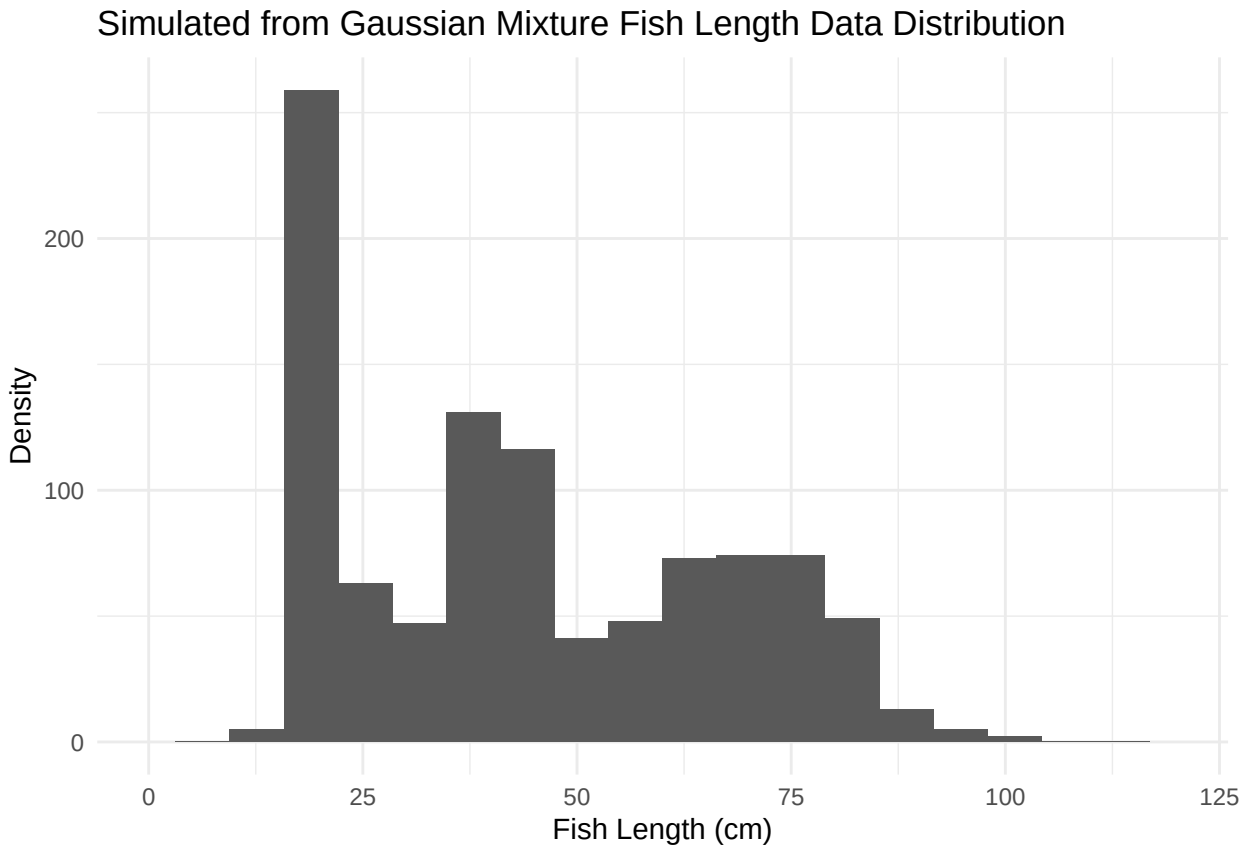
The teamEM() function outputs the correct data, the data converges.

```
## [1] "No of iterations is  55"
## [1] "Posterior has 1000 rows and 3 columns: TRUE"
## [1] "Posterior probability for each observation adds up to 1: TRUE"
## [1] "Difference in (likelihood(i+1) and \n              likelihood(i) is  1.0125404514838e-08 \n
## [1] "Does converge output match with what is confered from difference in likelihoods:  TRUE"
## [1] "Correct number of outputs: TRUE"
```

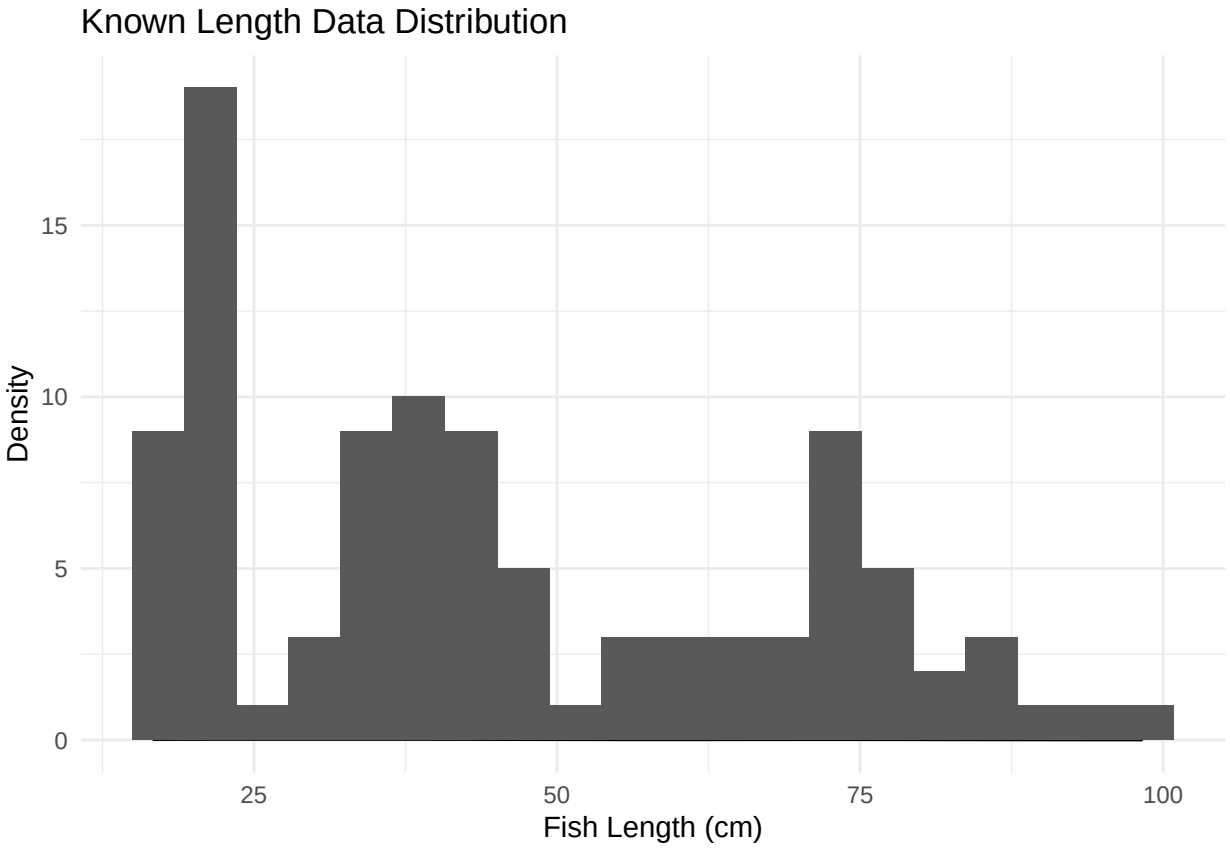
Simulation from Gaussian Mixture

As part of the testing of the teamEM function, we simulated data from a known Guassian mixture, implemented the teamEM function on this data and compared the estimates with the known results. The simulation testing function (found in file 'Simulation_Testing.R') takes distribution inputs from the user, creates the simulated Gaussian mixture, runs the 'teamEM' function on this data and evaluates the accuracy of the estimates via confidence intervals. In order to obtain the simulated data, the function samples 1000 values from 3 different Normal distributions with means and standard deviations sourced from the inputs of the 'simulation' function. For our testing, we used means of 20, 40, 70 and standard deviations of 2, 5, 10 respectively. In the simulation data, 100 of the values will have known 'Age' data; the rest (900) of the 'Age' values will be 'NA'.

```
## Warning: Removed 2 rows containing missing values (geom_bar).
```


Simulated from Gaussian Mixture Fish Length Data Distribution
Density
200
100
0
0
25
50
75
100
125
Fish Length (cm)

```
## [1] 100
```




```
## [1] "No of iterations is  284"
```

The simulated data has 3 distinct peaks (much like the FishLengths.RData file) with each peak having a different standard deviation. The next step was running the teamEM function on the simulated data which had the following outputs:

simulation(data, means = c(20, 40, 70), sds = c(2, 5, 10))

[1] "No of iterations is 78" [1] "The values for the gaussian mixture are mu = 20 , 40 , 70 with sd = 2, 5, 10. The EM algorithm produced means of mu = 22.19 , 41 , 66.82 . Hence, the confidence intervals for the means are (20.56 , 23.81) , (36.19 , 45.81) , (51.37 , 82.28)"

The user will be able to use this to evaluate if the true means lie in the confidence intervals. Please note: since the seed has not been set in this function, the estimates do change on every re-run. In this testing example, the 'teamEM' function took 78 iterations to converge to the means with a system time of 0.68 seconds.

Results

[1] "No of iterations is 55"

Table 2: Estimates

	mu	sigma	lambda
Age1	23.1127137279	3.85285226238	0.201919437641
Age2	41.8099489678	5.62986777882	0.452678184357
Age3	66.8606052700	8.35678860689	0.345402378001

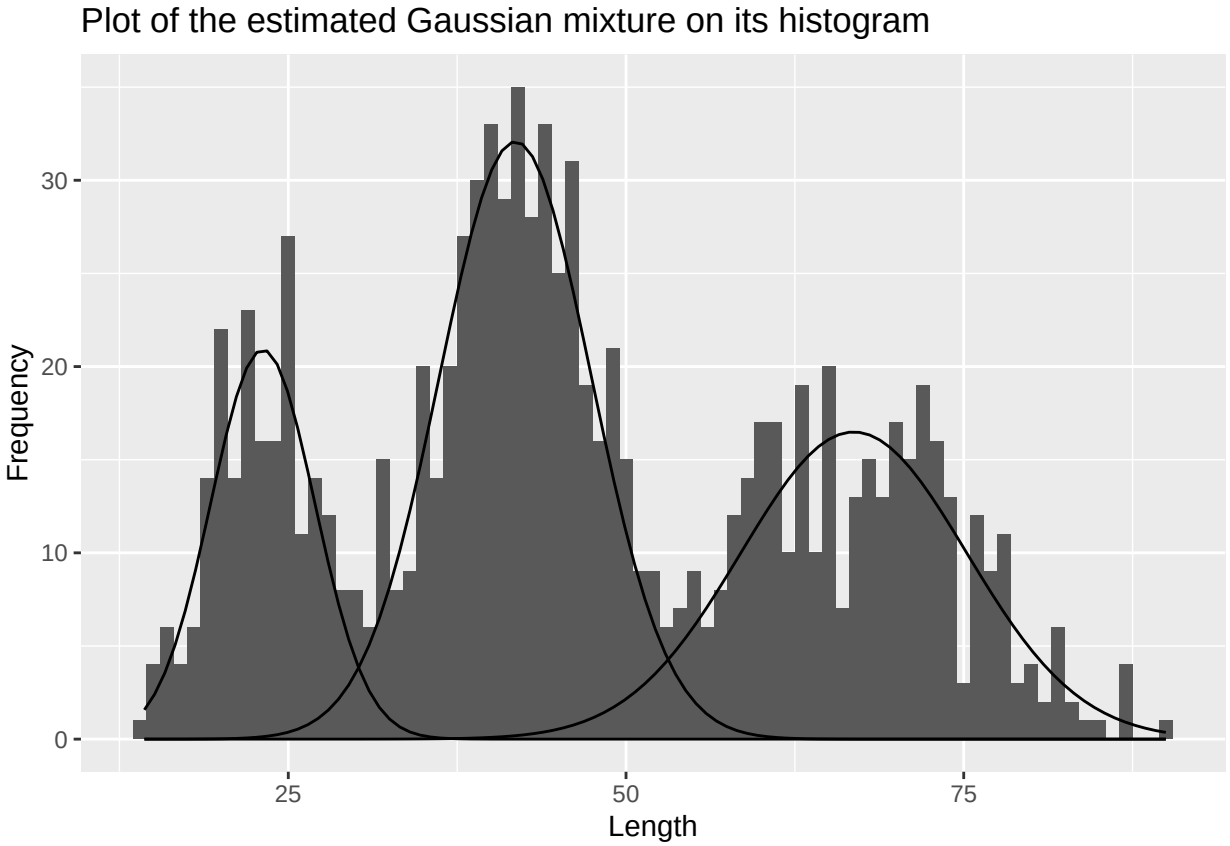
The results obtained from the estimates table show the estimated mean and standard deviation of the length

of fish contained in each age group of fish, Age 1, Age 2 and Age 3. This means the EM algorithm calculated that the average length of fish in age group 1 was 23.11, in age group 2 was 41.81 and age group 3 was 66.86. These results are in line with the distribution of known fish lengths and therefore are a sensible set of estimates.

Plot

The plot show how the model fits to the original data. The three distribution curves follow the histogram trend, the teamEM algorithm works effectively.

```
## [1] "No of iterations is  55"
```



Bibliography

1. Do, C.B. and S. Batzoglou. 2008. What is the expectation maximization algorithm? Nature Biotechnology 26:897-899.