

# Statistical Computation - Assignment 2

Philip Dahlqvist-Sjöberg

9th of April 2020

## 1 Two-dimensional minimization

We have the functions given  $x \in [-1.5, 4]$  and  $y \in [-3, 4]$ ,

$$f(x, y) = \sin(x + y) + (x - y)^2 - 1.5x + 2.5y + 1 \quad (1)$$

$$\frac{\partial f}{\partial x} = \cos(x + y) + 2x - 2y - 1.5 \quad (2)$$

$$\frac{\partial f}{\partial y} = \cos(x + y) + 2x - 2y + 2.5 \quad (3)$$

$$Hessian = \begin{bmatrix} 2 - \sin(x + y) & -2 - \sin(x + y) \\ -2 - \sin(x + y) & 2 - \sin(x + y) \end{bmatrix} \quad (4)$$

a) See Figure 1.

b) Here i have used the Newton Raphson method to find the local minimum. It uses equation (2), (3) and (4) to search for an optimum. The stopping rule is the absolute criteria, see R-code, and from looking at the contour plot, a good starting point seems to be at  $(0, -1)$ . This starting point is marked as blue in the left graph in Figure 1. With the algorithm, it found the local minimum of  $-1.913$  after five iterations at point  $(-0.55, -1.55)$ , marked as red in the same graph.

c) To find a possible local minimum at the border, the algorithm is constructed to hold one axis

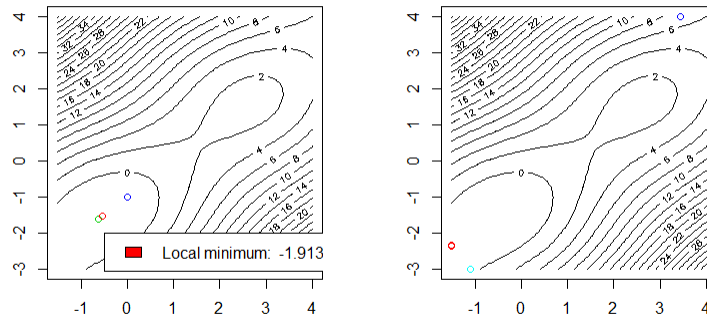


Figure 1: Contour plot, with points for b) in left and points for c) in right graph.

constant for each of the sides. From the starting values for  $(left, right, top, bottom) = (y = -2, y = -2, x = 1, x = 1)$ , all except of the left border converged out of the allowed limits. The left border found a local minimum of  $-1.253$  at  $y = -2.38$  after 4 iterations. However, this minimum is larger than the local minimum obtained within the borders. This is the red point in the right graph in Figure 1.

Table 1: EM-algorithm for three normal populations

$p_1$	$p_2$	$p_3$	$\mu_1$	$\mu_2$	$\mu_3$	$\sigma_1$	$\sigma_2$	$\sigma_3$
0.25	0.23	0.52	3.83	1.07	6.58	0.72	1.20	0.60

## 2 EM algorithm for mixture of three normal components

a) See R-code for algorithm of three normal distributions.

b) A stopping criteria that will work regardless of the scaling, is a relative absolute criteria. It will standardize the parameter change, to work with the same stopping limit. However, in this case one would still use the "p" parameter of group allocation in the stopping criteria. Hence, in the formula

$$cc = \sum \frac{|P_{new} - P_{old}|}{P_{ratio}}$$

where  $P$  is a vector of parameters of interest, and the "p" parameter in  $P_{ratio}$  is 1, to still take this change into account.

c) In Figure 2, we see the *threepops* data plotted in a histogram. At the x-axis, the means from running the EM-algorithm are marked to illustrate the three observed populations.

We observe three populations defined as  $X_1 \sim N(3.83, 0.72)$ ,  $X_2 \sim N(1.07, 1.2)$ ,  $X_3 \sim N(6.58, 0.6)$  after 200 iterations. The parameters observed are shown in Table 1.

d) In Figure 3-5, the five parameters are shown over the 200 iterations until stopping criteria.

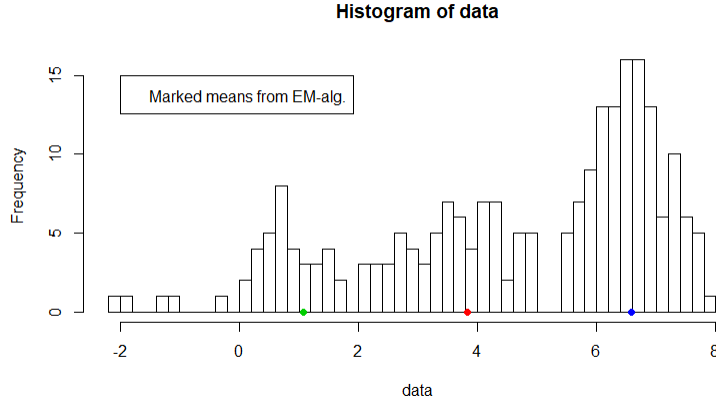


Figure 2: Histogram of the *threepops* data, with marked means from running the EM algorithm in a).

Comparing the mean and sigma from Figure 3 and histogram in Figure 2, we can see (the colors are matching between the graphs) that it makes sense how the means and sigmas have converged. In Figure 2, we can distinguish three peaks, all representing a normal population. Looking at the green, far left, population, its variance seems to have a tail on its left side. This would give a reason that this sigma did not converge to as small a value as the other two populations.

Looking at Figure 4 and 5, we can see that " $p_1$ " seem to converge to a small value, however the other two do not converge at all. I suspect that there is something fishy in the calculations of " $p$ " in my algorithm, but can't figure it out. One would assume that they have converged properly, due to the well fitted population parameters shown in the histogram.

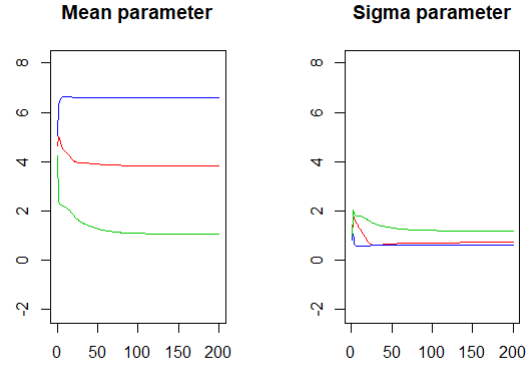


Figure 3: Parameter convergence of the three different means in left graph. Parameter convergence of the three different sigmas in the right graph.

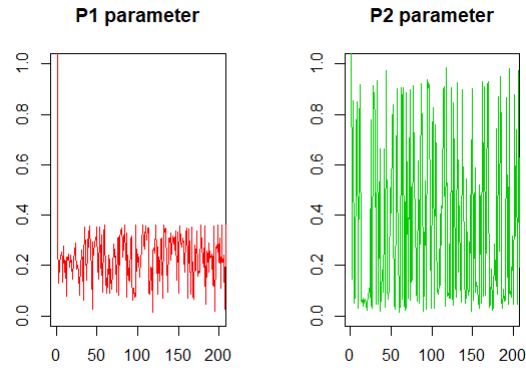


Figure 4: Parameter convergence of " $p_1$ " in left graph, and " $p_2$ " in right graph.

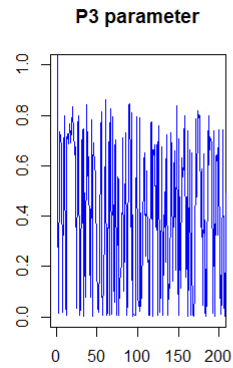


Figure 5: Parameter convergence of " $p_3$ ".

### 3 EM-algorithm for bivariate normal mixture

a) See R-code for the algorithm.

Table 2: EM-algorithm for bivariate normal populations

Iterations	Expected Q	$p_1$	$p_2$	$\mu_{1_1}$	$\mu_{1_2}$	$\mu_{2_1}$	$\mu_{2_2}$
58	-3.06	0.56	0.44	2.19	3.65	3.13	2.71
$\sigma_{1_{11}}$	$\sigma_{1_{12}}$	$\sigma_{1_{21}}$	$\sigma_{1_{22}}$	$\sigma_{2_{11}}$	$\sigma_{2_{12}}$	$\sigma_{2_{21}}$	$\sigma_{2_{22}}$
0.53	0.04	0.04	0.42	1.09	0.56	0.56	1.04

b) In Figure 6 we can see a two-dimensional scatter plot of the variables from the *bivardat* data. From looking at the graph, it is hard to distinguish large differences and clusters of observations. However, there looks to be a cluster around the point (2.2, 3.5) and (3, 3). From these points, the variance in both x and y axis looks to be (1.5, 1.5) in the first point and (1, 1) in the second point. since there is a lot of spread in the data, I assume a low covariance of 5% for all points. Since there looks to be more observations in the first point, a good starting "p" would be (0.7, 0.3).

c) With the starting values discussed in b),

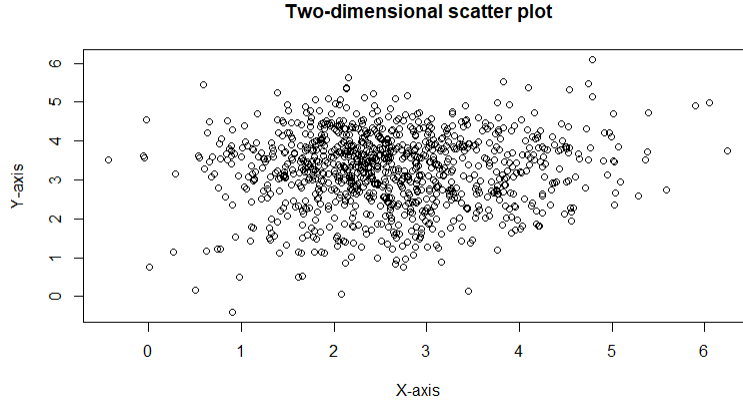


Figure 6: Two-dimensional plot of the variables in *bivardat* data.

$$\Sigma_1 = \begin{bmatrix} 1.5 & 0.05 \\ 0.05 & 1.5 \end{bmatrix} \Sigma_2 = \begin{bmatrix} 1 & 0.05 \\ 0.05 & 1 \end{bmatrix}$$

$$\mu_1 = \begin{bmatrix} 2.2 \\ 3.5 \end{bmatrix} \mu_2 = \begin{bmatrix} 3 \\ 3 \end{bmatrix} P = \begin{bmatrix} 0.7 \\ 0.3 \end{bmatrix}$$

the algorithm is run. The converged parameters are shown in Table 2. After 58 iterations, the expected log-likelihood Q is -3.06.

## 4 Gauss-Hermite integration

a) See Figure 7, where the weight are plotted against the nodes. With a line, the nodes form an approximate distribution function is illustrated.

b) In Table 4 the Gauss-Hermite integration results are shown using the formulas given in the R-code. These integrations are calculated using the 11 nodes shown in Table 3. Since we know that the integral is one for these familiar density distributions, the results are very close to the true value, from only a few nodes. Looking at the plot, it look symmetric and centered around zero. This indicates that the distribution from where these nodes are selected from, is of standard normal distribution. Hence, the *iii*)  $\mu = 0$  should be the best possible distribution to approximate with the Gauss-Hermite integration, which is clear from the results, since it is actually one. Also the *t*-distribution and double-exponential distribution should, and did perform close from only a few nodes.

Table 3: Nodes of the distribution of interest

Node $i$	Node $x_i$	Weights $A_i$
1	-3.66847	1.43956e-06
2	-2.78329	0.000346819
3	-2.02595	0.0119114
4	-1.32656	0.117228
5	-0.65681	0.42936
6	0	0.654759
7	0.65681	0.42936
8	1.32656	0.117228
9	2.02595	0.0119114
10	2.78329	0.000346819
11	3.66847	1.43956e-06

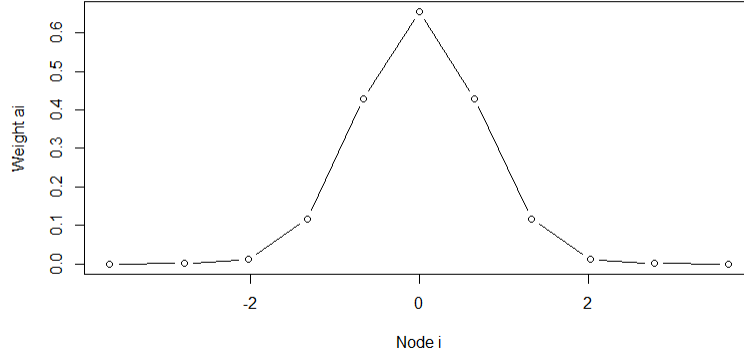


Figure 7: Weight for node  $i$ .

Table 4: Gauss-Hermite integration

	Integration $\approx 1$
i)	0.9762
ii)	1.0214
iii) mean 0	1
iii) mean 1	0.9998
iii) mean 2	0.9912
iii) mean 3	0.9912