Solutions to exercises of Day 4

Exercise 4.1

(a) At the finest level, there are 14 clusters. These small clusters are organised in three larger clusters.
(b) This is an artificial problem. However, the desired number of clusters will usually depend on the application. If the small clusters have a physical meaning, such as e.g. individual species while the three larger clusters represent animal kingdoms, then the specific problem which the biologist is studying will determine the appropriate number of clusters, i.e. the level of detail considered.

Exercise 4.2

(b) There are no vertical stems that are distinctly longer than the other. The tree grows gradually, not in leaps and bounds.
(c) In terms of the lengths of the stems there is no difference.

Exercise 4.3

(a) These lengths correspond to the distances (complete linkage) between the two clusters being joined by the bridge consisting of two stems and a horizontal bar.
(b) The closer together the samples in a cluster and the further apart the clusters, the better the clustering.
(c) Long stems correspond to large distances between clusters and therefore a potential point to cut the dendrogram.

Exercise 4.4

(a) Yes, in the single linkage dendrogram, there are basically two lengths of vertical stems: those indicating the joining of the smallest clusters, and those indicating the joining of the larger three clusters. This is less pronounced with complete linkage.
(b) The average linkage dendrogram is roughly similar to the complete linkage dendrogram.

Exercise 4.5

(a) Two is a good solution, since there are two clear elongated Gaussians.
(b) Yes, the two cigars are separated, although some outliers make the interpretation of the clustering solution more complex.
(c) No, complete linkage does not succeed in finding the two cigar-shaped clusters, since its linkage scheme favors circular clusters. Half of the samples from the other cigar are closer (in the complete link sense) than the rest of the samples in the same cluster.

Exercise 4.6

(a) Two clusters appear OK, but a case could be made for some other configurations: the dendrogram is not absolutely clear.
(b) Absolutely not; there are two outliers in one cluster and the rest are in the other cluster.

(c) This is unclear, since there is little structure in the dendrogram.

Exercise 4.8

(a) When samples become prototypes, the prototype is always amongst the samples. This obviously reduces the number of possible initial conditions and has the disadvantage of being more susceptible to suboptimal local minima. An advantage is that it reduces the possibility of a situation where one prototype ends up without any samples.
(b) K-means is better suited for datasets with spherical clusters, and should therefore work better on the `messy` dataset.

Exercise 4.9

(b) The algorithm has converged, i.e. no alteration of the prototype positions results in a decrease of the within-scatter. However, the within-scatter of this solution is significantly higher than the within-scatter associated with solutions where there is a single prototype per cluster. In order to reach the global minimum, where each prototype is in a single cluster, one of the prototypes now sharing a cluster with another prototype needs to move towards the cluster without a prototype. In order to achieve this, the distances between the objects already assigned to the prototype that needs to move will increase, which will result in an *initial* increase in the within-scatter. Since the K-means algorithm only decreases the within-scatter, it remains stuck in this local minimum.
(c) It does not suffer from this particular problem, since it is a deterministic algorithm, independent of the random initial conditions. However, one might argue that most variants of hierarchical clustering avoid this problem by simply not minimizing an objective function at all. Note that also for hierarchical clustering randomization can be useful, for example to assess cluster stability.
(d) Run the algorithm several times with different initial settings, and choose the best result.

Exercise 4.10

A large jump in the fusion graph implies that two clusters were merged that were, relative to the other cluster distances, quite far apart.

Exercise 4.11

(a) From $g = 2$ to $g = 3$.
(b) The fusion graph is ambiguous about the exact number of clusters: either two or three clusters are associated with large fusion jumps of roughly the same magnitude. This is due to the complete linkage distance: the distance between furthest samples of clusters at $(0, 0)$ and $(1, 1)$ (which are merged to result in a total of two clusters) is roughly half the distance between furthest samples in the merged cluster at $(0, 0)/(1, 1)$ and the cluster at $(2, 2)$, which are merged to result in a single cluster. In single linkage this does not occur, since the minimal distances between the clusters at $(1, 1$ and $(2, 2)$ is the same as the minimal distance between the merged cluster at $(1, 1)/(2, 2)$ and the cluster at $(0, 0)$.

Exercise 4.12

(a) There are two pronounced jumps, at 3 and 14 clusters.

Exercise 4.13

(a) At three clusters, since the largest fusion jump occurs between $g = 2$ and $g = 3$.

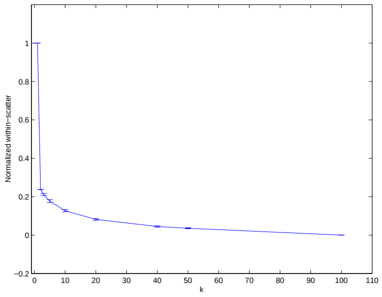
(b) No, three outliers constitute the members of two of the three clusters with all the other samples in the third cluster.
(c) While the size of the fusion jump between $g = 1$ and $g = 2$ is not very convincing (not much larger than the other jumps) the resulting clustering is better. This stems from the fact that complete linkage is less prone to outliers than single linkage.

Exercise 4.14

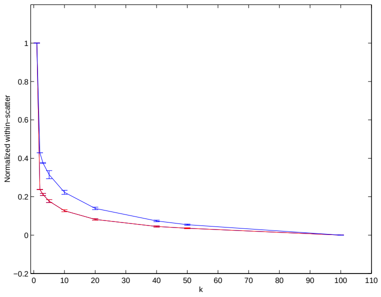
(b) Example code:

```
g = [1 2 3 5 10 20 40 50 100]; % number of clusters to try
nreps = 10; % number of repeats
for i=1:length(g)
   for j=1:nreps
     [p,labs,ws(i,j)] = kmclust(a,g(i),0,0); % store within-scatter matrix ws
   end
end
WS_mean = mean(ws,2); % calculate average within-scatter
WS_std = std(ws,0,2); % calculate standard deviation
errorbar(g,WS_mean/WS_mean(1),WS_std/WS_mean(1)); % plot normalized within scatter
```

(c) 1) The within-scatter (figure (a) below) decreases monotonically as g increases, and for $g = n$, it is zero. 2) The largest decrease in within-scatter occurs when we go from one to two clusters, revealing that there are probably two clusters in the data.



(a)



(b)

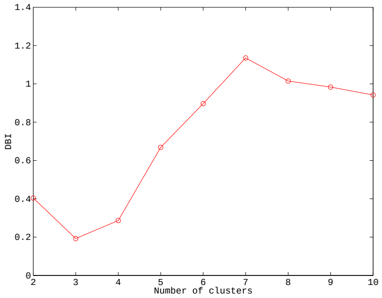
(d) For `d = 10` (red line in figure (b) above) the decrease in within-scatter when going from one cluster to two clusters is much larger than for `d = 5` (blue line). This indicates that for `d = 10` we have a "stronger" clustering than for `d = 5`. This is to be expected since the parameter `d` specifies the distance between class means in the first dimension.

Exercise 4.15

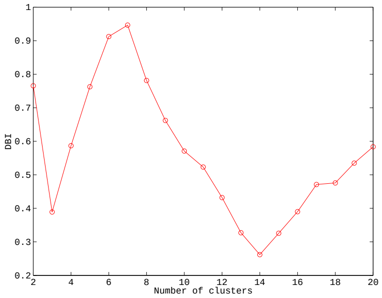
There should be no large jump between $g = 1$ and $g = 2$.

Exercise 4.16

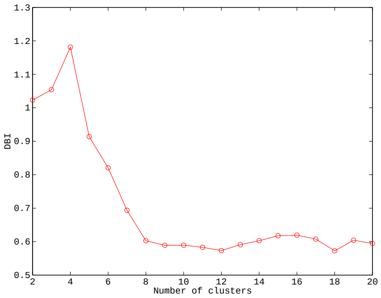
(b) It should have a minimum at three clusters.
(c) The DBI curve is shown in the figure below, with a clear minimum at three clusters.



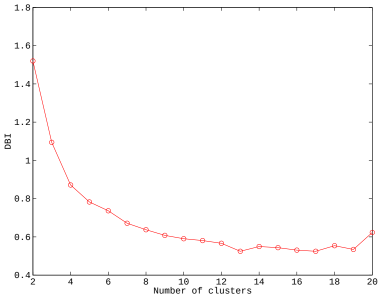
(d) There is a pronounced minimum at 3 clusters and a slightly lower minimum at 14 clusters. The first minimum (3 clusters) corresponds to the situation where the smaller clusters are grouped in three large clusters (four in top-left, four in top-right and six at the bottom) while the minimum at 14 corresponds to the fine cluster structure in the data. The valley at fourteen is more pronounced since the ratio of the maximal within-scatter to the minimal distance between any pair of these clusters is smaller than for the three cluster configuration.



(e) The DBI curve is not minimal at two clusters, and starts flattening at around eight clusters. Roughly the same result is obtained for complete and single linkage (figures (a) and (b) below). The problem is that DBI assumes *circular* clusters, while the cigars dataset consists of strongly elongated clusters.



(a)



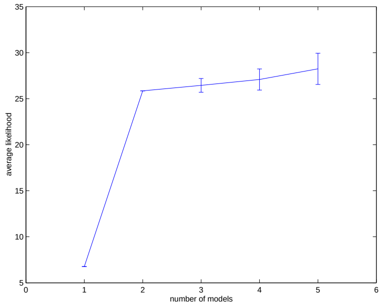
(b)

Exercise 4.17

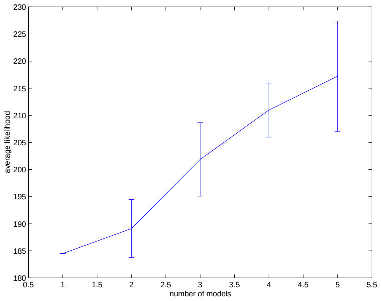
(a) `'circular'`: diagonal covariance matrix with *equal* variances on the diagonal
`'aligned'`: diagonal covariance matrix with *unequal* variances on the diagonal
`'gauss'`: full covariance matrix, i.e. no constraints on the entries.
(b) The optimal g should be 2.
(c) When the likelihood curve flattens, it is an indication of very small increases in likelihood with more clusters. The point where it starts flattening is usually an indication of the desired number of clusters. The curve of likelihood as a function of the number of models is as shown

in figure (a) below. Note that results may vary due to local minima for $g = 2$ (and often the error bar will be larger). Example code:

```
ncomp = [1:5] % array of number of clusters to try
nreps = 20; % number of repeats
for i=ncomp
   for j=1:nreps
     lik(j) = em(a,i,'gauss',0,0,0); % fit unconstrained Gaussian mixture model
   end
   L_mean(i) = mean(lik); % calculate mean likelihood
   L_std(i) = std(lik); % calculate standard deviation
end
errorbar(ncomp,L_mean,L_std); % likelihood as function of the number of components
```



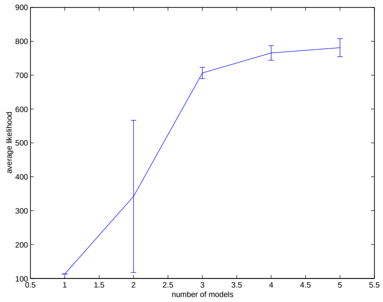
(a)



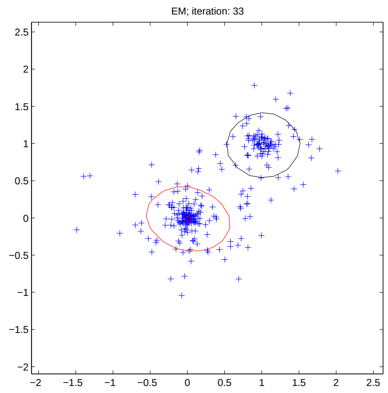
(b)

(d) The log-likelihood (figure (b) above) increases roughly linearly with the number of clusters, i.e. there is no clear flattening of the curve indicating a preference for a given number of clusters.

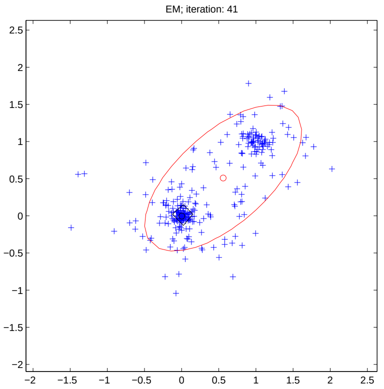
(e) The optimal number of clusters is two, with a covariance matrix of type 'circular' (the other types give similar performance). The log-likelihood vs. the number of clusters is depicted in figure (a), the desired solution with two clusters in (b) and two frequently occurring undesirable solutions with two clusters in figures (c) and (d). These undesirable solutions are the reason why the variance is so large at two clusters and why the curve does not flatten at two, but at three clusters.



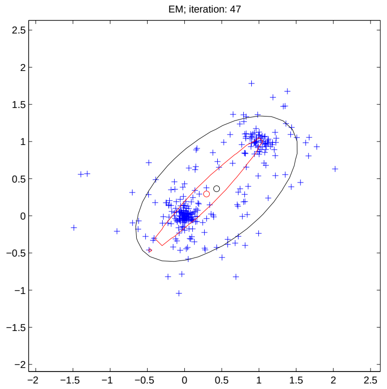
(a)



(b)



(c)



(d)

Exercise 4.18

(a) The probabilities of the two sequences given the models are

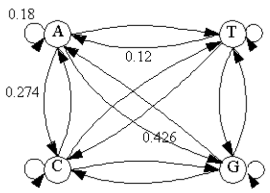
- IID:
  $P(CCGAT) = P(C)P(C)P(G)P(A)P(T) = 0.1 \times 0.1 \times 0.1 \times 0.2 \times 0.6 = 0.00012$
  $P(CATAT) = P(C)P(A)P(T)P(A)P(T) = 0.1 \times 0.2 \times 0.6 \times 0.2 \times 0.6 = 0.0014$

- Weight matrix:
  $P(CCGAT) = P_1(C)P_2(C)P_3(G)P_4(A)P_5(T) = 0.1 \times 0.2 \times 0.6 \times 0.1 \times 0.15 = 0.00018$
  $P(CATAT) = P_1(C)P_2(A)P_3(T)P_4(A)P_5(T) = 0.1 \times 0.3 \times 0.05 \times 0.1 \times 0.15 = 0.0000225$

- First-order Markov chain:
  $P(CCGAT) = P(C)P(C|C)P(G|C)P(A|G)P(T|A) = 0.1 \times 0.1 \times 0.1 \times 0.3 \times 0.05 = 0.000015$
  $P(CATAT) = P(C)P(A|C)P(T|A)P(A|T)P(T|A) = 0.1 \times 0.35 \times 0.05 \times 0.6 \times 0.05 = 0.0000525$

Exercise 4.19

(a) The sum of all terms in a row must be one, since this is the probability that the base corresponding to a specific row is followed by any of the bases.
(b) Element a_{ij} of the transition matrix is the probability $P(q_t = j|q_{t-1} = i)$ of going from state i to state j. In a Markov chain this is indicated by a directed edge from state i to state j (see figure below, only the transition probabilities of going out from state A are indicated).

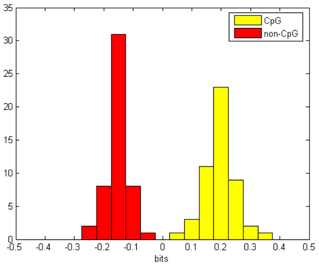


(c) The transition probability matrices were estimated on a modest number (50) of sequences and bases (on average 500 per sequence). Maximum likelihood estimates only converge to the true probabilities with much more training data. You might want to edit the script `cpgislands.m` to see the effect of increasing the number of sequences and the maximum number of bases on the estimated transition matrices.

(d) The log-likelihood ratio matrix of CpG versus non-CpG is

$$LL = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{c} \begin{array}{cccc} A & C & G & T \end{array} \\ \begin{pmatrix} -0.7370 & 0.41865 & 0.5799 & -0.8074 \\ -0.9131 & 0.3044 & 1.8126 & -0.6915 \\ -0.6233 & 0.4626 & 0.3316 & -0.7347 \\ -1.1638 & 0.5708 & 0.3951 & -0.6820 \end{pmatrix} \end{array}$$

Positive values correspond to transitions that are more likely in CpG islands and negative values to transitions that are more likely in non-CpG islands. Indeed, the log-odds for ratio for the transition from C to G is a large positive value. See the figure below for a histogram of the log-odds (in bits) for all sequences in the test set. CpG islands are shown in yellow and non-CpG islands are shown in red.



(e) Consecutive stretches of length at least 100 are: 19011–19123 and 63353–63492. You can use the script `cpgstretches.m` to find these. Similar results are also obtained with a simpler method implemented in `cpgplot` (`https://www.ebi.ac.uk/Tools/seqstats/emboss_cpgplot/`).
(f) The main disadvantage of our approach is the arbitrary fixed window size. One would expect CpG islands to be of arbitrary length and have sharp boundaries. A hidden Markov model is probably better since we can make a single model for the entire sequence (details in Durbin et al., pp. 51–53).

Exercise 4.20

We could use the full-fledged Viterbi algorithm on a trellis for this calculation. However, it is easier to note that only four paths can generate a sequence of length seven such as the consensus sequence `ACACATC`. Using the general equation for the joint probability of an observed sequence x and a state sequence Q in terms of transition probabilities a_{ij} and emission probabilities $b_j(x)$ (see lecture)

$$P(x,Q) = \prod_{t=1}^{N} P(q_t|q_{t-1}) \prod_{t=1}^{N} P(x_t|q_t) = \prod_{t=1}^{N} a_{q_{t-1},q_t} \prod_{t=1}^{N} b_{q_t}(x_t),$$

we have:

$P(ACACATC \text{ \& } 1234567) = 0.8\times1\times0.8\times1\times0.8\times0.6\times0.4\times0.6\times1\times1\times0.8\times1\times0.8 = 0.047$
$P(ACACATC \text{ \& } 1234456) = 0.8\times1\times0.8\times1\times0.8\times0.6\times0.4\times0.4\times0.2\times0.6\times0\times1\times0 = 0$
$P(ACACATC \text{ \& } 1234445) = 0.8\times1\times0.8\times1\times0.8\times0.6\times0.4\times0.4\times0.2\times0.4\times0.2\times0.6\times0 = 0$
$P(ACACATC \text{ \& } 1234444) = 0.8\times1\times0.8\times1\times0.8\times0.6\times0.4\times0.4\times0.2\times0.4\times0.2\times0.4\times0.4 =$ 0.00013

The optimal state sequence is the one with the highest probability: 1-2-3-4-5-6-7.