Modern Statistics and Big Data Analysis, Exercises 4

Deadline: end of Thursday 2 November.

1. (3 points) Analyse the Tetragonula bees data set in R-package `prabclus`. This data set gives genetic data for 236 Tetragonula (Apidae) bees from Australia and Southeast Asia. The data give pairs of alleles for 13 diploid microsatellite loci (these can be thought of as positions of genes, each characterised by two alleles). The interest here is in clustering the bees in order to find different bee species. A species is characterised by having a similar genetic makeup, whereas different species should be separated. The data set can be loaded by

   ```
   library(prabclus)
   data(tetragonula)
   ```

   In this form, the alleles are coded by numbers (every locus has a six digit number, with the first three digits referring to the first allele, and the digits 4-6 referring to the second allele). The data set needs some preprocessing in order to make it ready for work.

   ```
   ta <- alleleconvert(strmatrix=tetragonula)
   ```

   converts the allele codes to letters, so that each locus now has two letters (there are also some missing values coded "-"). Finally,

   ```
   tai <- alleleinit(allelematrix=ta)
   ```

   produces a collection of ways to represent the data (you don't need to understand much of this; if you are curious, consult the help page of `alleleinit`). Particularly, `tai$distmat` is now a matrix of genetic distances (the help page of `alleledist` explains how this was computed), which is the data set you are finally asked to work with.

   (a) Using `tai$distmat`, compute an MDS and show the MDS plot. Try out different dissimilarity-based cluster analysis methods and decide which one you think is best here. Also choose a number of clusters and visualise your final clustering using the MDS. Give reasons for your choices.

   (b) Try out the following alternative way of clustering the data: Take the points generated by the MDS and apply k-means clustering, Ward's method, or fitting a Gaussian mixture to them. Again choose a number of clusters.

   What are advantages and disadvantages of this approach compared to the hierarchical clustering? Do you think that this clustering is ultimately better? Do you think it would be better, for this task, to produce an MDS solution with $p > 2$?

2. (3 points) On Virtuale under "Data sets" you'll find the data set `wdbc.data`. This data set is taken from the UCI Machine Learning Repository `http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)`. Data are given about 569 breast cancer patients, and there are the two "true" classes of benign (357 cases) and malignant (212 cases) tumors. There are ten quantitative features in the dataset, the documentation of which is in the file `wdbc.names`, which I also put on the IOL/Moodle site, however you don't really need to know what's in there to do this exercise. Actually the features have been recorded for every cell nucleus in an image and three different statistics of these are in the original dataset so that the number of variables is 30, but I'm asking you to use only the first ten variables, otherwise clustering can be very unstable and computationally hard.

   You can load the data as follows (assuming it is in the same directory where you run R):

```
wdbc <- read.csv("wdbc.data",header=FALSE)
# The variables I ask you to use are variables 3 to 12,
# so wbbcc is the data set to be clustered:
wdbcc <- wdbc[,3:12]

# There is also a diagnosis whether cancer is benign or malign as variable 2
# in the data set:
wdbcdiag <- as.integer(wdbc[,2])
```

   Compute different clusterings of the data (use at least two different approaches including a Gaussian mixture model and try out numbers of clusters up to 10) and compare them first *without using the information about benign vs. malign cancers* in the diagnosis variable `wdbcdiag` (this also means you should ignore the information that there are 2 classes). Which clustering do you think is best?

   Only after you have made a decision about your favourite clustering, use the ARI to compare all these clusterings to `wdbcdiag`.

   Note that you don't get any marks for picking the best clustering according to ARI with `wdbcdiag` in the first place, so it won't do any good to cheat and use `wdbcdiag` for finding an ARI-optimal clustering. The idea of this exercise is that you learn from evaluating the clusterings against `wdbcdiag` that you first have constructed without that information, so that you can critically appraise the reasons which you used earlier to pick your favourite clustering.

   Note also that there may be other legitimate and meaningful clusters in the data about which we don't have information, so a clustering that has low ARI with `wdbcdiag` isn't necessarily bad.

3. (2 points) With given $c(1),\ldots,c(n)$ and $\hat{\mathbf{m}}_k = \frac{1}{n_k}\sum_{c(i)=k}\mathbf{x}_i,\ n_k = |C_k|$,

$$S(\mathcal{C},\mathbf{m}_1,\ldots,\mathbf{m}_K) = \sum_{i=1}^{n} d_{L2}^2(\mathbf{x}_i,\mathbf{m}_{c(i)})$$

and all further notation as in the definition of k-means clustering, show

$$S(\mathcal{C}, \hat{\mathbf{m}}_1, \ldots, \hat{\mathbf{m}}_K) = \sum_{k=1}^{K} \frac{1}{2n_k} \sum_{\mathbf{x}_i, \mathbf{x}_j \in C_k} d^2_{L2}(\mathbf{x}_i, \mathbf{x}_j).$$

Note that the notation for the sum "$\sum_{\mathbf{x}_i, \mathbf{x}_j \in C_k} d^2_{L2}(\mathbf{x}_i, \mathbf{x}_j)$" implies that i and j both take all possible values (as long as $c(i) = c(j) = k$), which in particular means that for any fixed pair of numbers (i, j) with, say, $i < j$, *both* $d^2_{L2}(\mathbf{x}_i, \mathbf{x}_j)$ and $d^2_{L2}(\mathbf{x}_j, \mathbf{x}_i)$ are summed up.

4. (2 points) I wrote a research paper comparing some distances for high dimensional continuous data:
Christian Hennig (2020) *Minkowski distances and standardisation for clustering and classification of high dimensional data.* arXiv:1911.13272, published in Imaizumi, Tadashi; Nakayama, Atsuho; Yokoyama, Satoru (Eds.) "Advanced Studies in Behaviormetrics and Data Science. Essays in Honor of Akinori Okada", Springer Singapore (2020), p. 103-118.
The paper can be accessed under `https://arxiv.org/abs/1911.13272`. Read the paper to the extent that you can answer the following questions (which means that you certainly don't need to understand and probably not even read everyting):

   (a) Focusing on Complete Linkage clustering, what is the best distance in these experiments (including the standardisation method)? Explain how this can be seen from the plots regarding the Complete Linkage results (i.e., explain roughly what these plots show).

   (b) Discuss how Complete Linkage and Partitioning Around Medoids clustering compare overall.