Computer Lab 3 - Unsupervised, semisupervised and active learning

Machine Learning 7.5 credits

Mattias Villani and Frank Miller, Department of Statistics, Stockholm University

INSTRUCTIONS:

- The sections named Intro do **not** have any problems for you. Those sections contain code used to set up the data and do some initial analysis, so just read and follow along by running each code chunk.
- Your problems are clearly marked out as Problem 1, Problem 2 etc. You should answer all problems by adding code chunks and text below each question.
- Your submission in Athena should contain two files:
  - This Rmd file with your answers.
  - A PDF version of this file (use the knit to PDF option above).
- You can also write math expression via LaTeX, using the dollar sign, for example´ β.
- You can navigate the sections of this file clicking (Top Level) in the bottom of RStudio's code window.

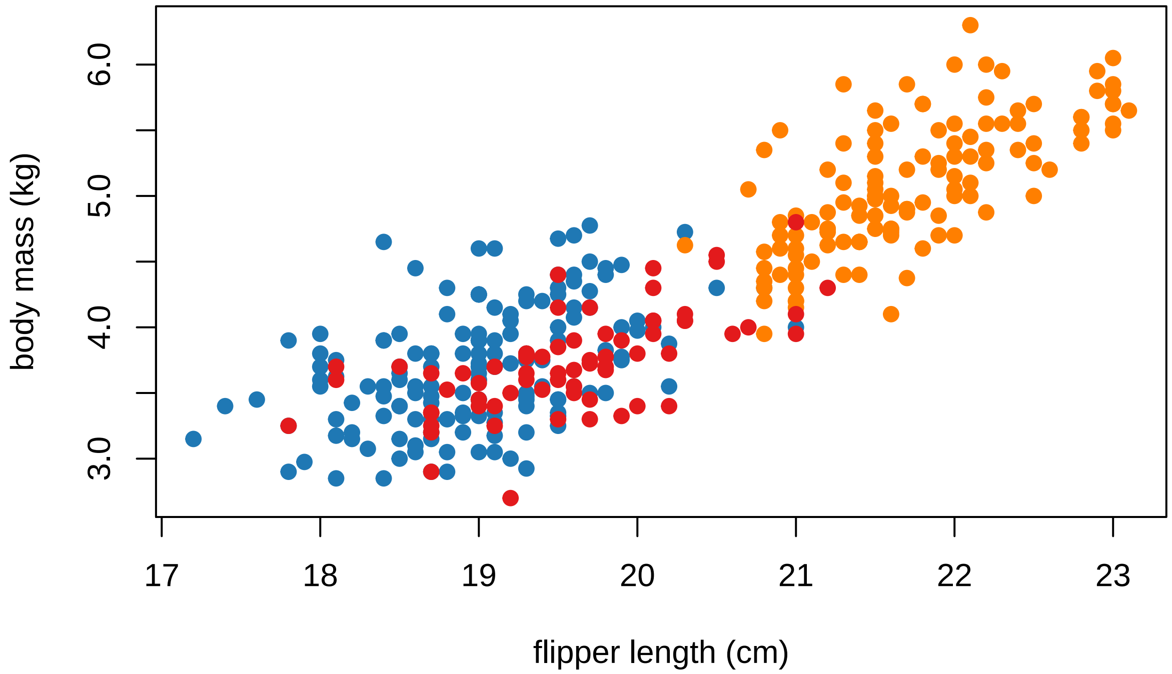
Intro1 - Loading packages and data

Loading some packages first. Do `install.packages()` for each package the first time you use a new package.

```
library("RColorBrewer") # for pretty prettyColors
prettyColors = brewer.pal(12, "Paired")[c(1,2,7,8,3,4,5,6,9,10)];
options(repr.plot.width = 12, repr.plot.height = 12, repr.plot.res = 100) # plot size
set.seed(12332)         # set the seed for reproducability
```

The aim of this lab to explore supervised, unsupervised and semi-supervised learning using Gaussian Mixture models. The data set used here contains measures of body mass and length of flippers for 342 penguins. See this blog post for some information, and note that I have excluded 2 penguins due to missing data. The penguins belong to three different species (Adelie, Chinstrap, and Gentoo) which will be used as labels for the observations. The code below loads the data and plots it.

```
penguins = read.csv("https://github.com/mattiasvillani/MLcourse/raw/main/Data/PalmerPenguins.csv")
xmin = min(penguins[,"flipper_length_cm"])
xmax = max(penguins[,"flipper_length_cm"])
ymin = min(penguins[,"body_mass_kg"])
ymax = max(penguins[,"body_mass_kg"])
plot(penguins[penguins[,"species"]=="Adelie","flipper_length_cm"],
     penguins[penguins[,"species"]=="Adelie","body_mass_kg"],
     col = prettyColors[2], xlim = c(xmin,xmax), ylim = c(ymin,ymax), pch = 19,
     xlab = "flipper length (cm)", ylab = "body mass (kg)")
points(penguins[penguins[,"species"]=="Gentoo","flipper_length_cm"],
       penguins[penguins[,"species"]=="Gentoo","body_mass_kg"], col = prettyColors[4], pch = 19)
points(penguins[penguins[,"species"]=="Chinstrap","flipper_length_cm"],
       penguins[penguins[,"species"]=="Chinstrap","body_mass_kg"], col = prettyColors[8], pch = 19)
```



Problem 1 - Supervised GMM - LDA and QDA Analyze the Penguin data using supervised Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA). Write your own code. The end result of your analysis should included a figure similar to Figure 10.4 in the MLES book. Are the assumptions in LDA plausible for this dataset? [Hint: you can use the mvtnorm package to get the multivariate normal density].

Problem 2 - Unsupervised GMM Pretend now that the labels of the Penguins are unknown. Use the EM for multivariate GMM code on the course web page (under Lecture 9) GMM_EM_Multi.R. Use the code to fit a Gaussian mixture model to the penguin data for M=1, 2 and 3 mixture components. Set reasonable initial values for the EM algorithm (at least take into account the scale of the data).

Problem 3 - Semi-supervised GMM Pretend now that the label for every odd observation in the dataset is known, but the label for every even observation is unknown. Modify the GMM_EM.R code to semi-supervised GMM; the function mixtureMultiGaussianEM should have an additional argument which contains a vector of labels (NA for unknown labels). Analyze the penguin data using a semi-supervised Gaussian mixture model with three mixture components.

Problem 4 - Active learning - logistic regression Use the Penguin data with the two species "Adelie" and "Chinstrap", only. Pretend then for the beginning that the species (labels) are not known, but keep this information such that the oracle can label queried data points later. Choose randomly 15 data points and label them. Run then active learning to query additional 45 data points based on a logistic regression model. Use both uncertainty sampling and variance reduction with an E-optimal design. Plot the labeled dataset after 60 labeled observations and compare between uncertainty sampling and E-optimality. Report the parameter estimates or the decision boundaries.