Statistical Analysis and Document Mining

Final exam

May 2022

You have 3 hours. All handwritten documents allowed, as well as the electronic documents on your computer. No calculator or mobile phone allowed.

No internet connection is available from your computer.

To do this practical exam, you have to use `RStudio`. Open each `Rmd` file located in the `$HOME/exam/` directory

Answer the questions directly in each `Rmd` file, ensure that you can knit it into an `html` file, and save the `Rmd` file. The different `Rmd` files correspond to three independent exercises.

Please answer all questions in English only.

Any answer not contained into the .Rmd file will not be taken into account, unless instructed explicitly by you or the teacher into the `Rmd` file (you can state into a `Rmd` file: 'Please see figure `myfig.png`' for example).

The contents of the $HOME/exam/ directory are

- `Part1.Rmd`, `Part2.Rmd`, `Part3.Rmd` (the files you have to complete and save)
- `references` (lecture notes, slides, books)
- `igraph.pdf` (documentation for the `igraph` package)
- `instructions.pdf` (this document)
- `rmarkdown_cheatsheet.pdf` (basic RMarkdown commands)
- `datasets/*`: datasets used in Part 1 and Part 3

Please check the presence of the `whoami.txt` file. This file is automatically generated at login and should already contain your first and last names, login and computer identifier. If the file is missing, please create and fill it.

Important – You can (and you should!) save the current state of your whole home directory using the icons on your desktop. Before that, save your currently open files using your editor(s) – particularly the `Rmd` and `html` files, which you should try to knit. These icons allow you to either:

1. Save your current work and continue your work;
2. Save your current work and stop the session. You have to perform this operation before leaving the room.

The backup operation is based on `rsync`, and does not store every intermediate saved version. If your PC crashes, your latest backup can be recovered with the help of the teaching staff.

Part 1: Multiple regression (7 points)

The dataset contains data on test performance, school characteristics and student demographic backgrounds. The data obtained in 1998 and 1999 are from Californian districts. The objective of the exercise is to predict test scores and find determinants of test scores (which combine math and reading tests) based on sociodemographic variables. Reading scores have been obtained from students in last school year of elementary school. The available variables in the dataset are the following:

- **enrltot**: number of kids at school
- **teachers**: number of full-time teachers
- **calwpct**: percentage of students in a public assistance program
- **mealpct**: percentage of students that qualify for a reduced price lunch
- **computer**: number of computers per class room
- **testscr**: test score
- **compstu**: number of computers per student
- **expnstu**: expenditure per student
- **str**: student teacher ratio
- **avginc**: district average income (when multiplied by 1000)
- **elpct**: percentage of students for whom English is a second language

We start by loading the dataset in `R` and attaching the `MASS` package.

```
data<-read.csv("Caschool.csv",sep=";")
library(MASS)
```

Q1: How many californian districts are represented in the dataset?

Q2: Using multiple regression, find which variables are significantly associated with test scores. Are the effects significant in the expected directions? (`testscr`).

Q3: Is the variable `calwpct` significant when using a simple linear regression (one predictive variable only)? Explain the difference with the result of multiple regression.

Q4: Using model selection based on an information criterion (e.g. with the `aic` command in `R`), find a parsimonious regression model that explains the test score. Explain the computational procedure and provide the parsimonious model.

Q5: Train the complete model (all variables) on all districts but the first 100 districts and evaluate the mean square prediction error (i.e the test score) using the first 100 districts.

Q6: Plot the first 100 fitted values as a function of the true value of the test score and evaluate if the model over or underestimates test scores. Confirm your results using numerical computations.

Q7: Compute the mean square prediction using the parsimonious model found in Q4. Compare prediction accuracies and evaluate if the reduced model under or over estimates test score.

Part 2: Classification (7 points)

Consider a simulated dataset which you will generate as follows:

(1) Set the seed of your `R` script with `set.seed(42)`.

(2) For each data point i, sample its label from a Bernoulli distribution $y_i \sim \mathcal{B}(p)$, i.e. $y_i = 1$ with probability p and $y_i = 0$ with probability $1 - p$. You can use the `rbern` function from `R` here.

(3) Then, depending on the label $y_i \in \{0, 1\}$ the associated data point x_i is sampled as follows:

$$y_i = 0 \quad \Rightarrow \quad x_i \sim 0.5\, \mathcal{N}(\mu_0^{(a)}, C_0^{(a)}) + 0.5\, \mathcal{N}(\mu_0^{(b)}, C_0^{(b)})$$

$$y_i = 1 \quad \Rightarrow \quad x_i \sim \mathcal{N}(\mu_1, C_1)$$

where

$$\mu_0^{(a)} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{and} \quad \mu_0^{(b)} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \quad \text{and} \quad \mu_1 = \begin{bmatrix} \varepsilon \\ 0 \end{bmatrix}$$

and $C_0^{(a)} = C_0^{(b)} = 0.5\ \mathbf{I}_2$ and $C_1 = \mathbf{I_2}$ with $\mathbf{I}_n$ the n-dimensional identity matrix. You can use the `rnorm` function from `R` to generate samples from a univariate Gaussian distribution and then transform them so to become 2D vectors with the statistics described above.

We will denote a set of n data points $(x_i, y_i)_{i \leq n}$ simulated with ε and p as $\mathcal{D}(n \mid \varepsilon, p)$.

Q1: Consider for now that $p = 0.5$ and $\varepsilon = 2$ and simulate two datasets:

$$\mathcal{D}_{\text{train}} = \mathcal{D}(200 \mid 2, 0.5) \quad \text{and} \quad \mathcal{D}_{\text{test}} = \mathcal{D}(1000 \mid 2, 0.5)$$

(a) What is the mathematical expression for the optimal Bayes classifier in this setting? And for its boundary region?

(b) Plot the boundary region for the Bayes classifier overlayed with the scattered data points of $\mathcal{D}_{\text{train}}$. Use different colors for each class and use function `contour` for plotting the boundary region.

(c) Estimate the error of the Bayes classifier on the samples from $\mathcal{D}_{\text{test}}$

(d) Train a LDA and a Logistic Regression classifier on $\mathcal{D}_{\text{train}}$ and estimate their errors on the samples from $\mathcal{D}_{\text{test}}$. How do these errors compare to the value obtained in (c)? Comment your results.

Q2: Consider now that $p = 0.5$ remains fixed and that ε can vary.

Simulate 51 datasets $\mathcal{D}_{\text{train}}^{(i)} = \mathcal{D}(200 \mid \varepsilon_i, 0.5)$ and $\mathcal{D}_{\text{test}}^{(i)} = \mathcal{D}(1000 \mid \varepsilon_i, 0.5)$ with $\varepsilon_i = 1 + i/10$ and $i = 0, \ldots, 50$.

(a) Calculate the test error for the Bayes classifier, the LDA, and the Logistic Regression classifier for each one of these datasets, i.e. train on $\mathcal{D}_{\text{train}}^{(i)}$ and test on $\mathcal{D}_{\text{test}}^{(i)}$.

(b) Plot a curve showing the error with each of these classifiers as a function of ε. Comment your results.
Remark: You should notice that for large values of ε the fitting of the logistic regression throws an error. Can you explain what is happening? What would be a good approach for limiting this problem?

Part 3: Community detection (6 points)

In this exercise, we will consider a famous example of a social network from the sociology literature, Wayne Zachary's "karate club" network. This network represents the pattern of friendships between members of a karate club at a US university, as determined by direct observation of the club's members over an extended period.

The network is interesting because during the period of observation a dispute arose among the members of the club over whether to raise the club's fees and as a result the club eventually split into two parts, of 18 and 16 members respectively, the latter departing to form their own club. It is the two factions in this split, as reported by Zachary, that will form the ground truth for us when applying different algorithms for community detection.

Start by loading the dataset into R as follows

```
library(igraph)
load('./karate.rda')
```

so the graph of interest will be stored at the `karate` variable of the workspace.

Q1: What does the following command line do?

```
plot(karate, vertex.size=degree(karate))
```

Can you interpret the result it outputs and why it might be useful for better understanding this graph of relations?

Q2: The true factions to which each member of the Karate club belongs to can be obtained via `V(karate)$Faction`. Use this information to calculate the modularity of the graph with for this ground truth setup.

Q3: Calculate the modularity matrix of the graph using the function `modularity_matrix` and obtain its eigenvectors. Interpret the magnitude of the coordinates of the leading eigenvector (i.e. the one related to the largest eigenvalue) and explain how it can be related to importance of each vertex in the graph. How can we use this eigenvector to split the graph into two communities?

Q4: Run algorithms `cluster_louvain` and `cluster_edge_betweenness` and comment the results you obtain, e.g. plot the graphs with communities detected by each algorithm, compare to the ground truth, check the modularity of split that we obtain, etc.