APRENENTATGE AUTOMÀTIC (APA)
Grau en Enginyeria Informàtica - UPC
Practical work 2017-2018

Lluís A. Belanche
belanche@cs.upc.edu

Abstract

This is the **guide** for the correct development of the practical work of the APA ("Machine Learning") course. The students must apply the different concepts and models lectured during the course to solve a real problem. The students must write a complete report describing the work carried out, the problems encountered and the solutions envisaged, as well as the final results and conclusions of the study.

1 General information

All students enrolled in APA are required to complete a term project. The goal is to develop a **classification or regression model** to solve one of the problems that can be obtained from selected repositories (detailed in 3.1). You can choose to explore any problem that motivates you, and even bring your own proposal.

You are expected to write a complete **report** describing the work carried out, its motivation, the problems encountered and the solutions envisaged, and the final results and conclusions of the study. The main text is *strictly* limited to 15 pages (this includes graphics, tables and references; note the code must be submitted *separately*).

The main programming language used for the modelling part must be `R`.[1] Remember that there are *many* packages for `R` which probably contain useful routines you can use; just be sure to mention them in your final document.[2] Other software can be used as long as it serves a specific or secondary purpose. Notice also that `R` can be interfaced with many other languages.[3] Any additional information on the methods or on the problems should be **acknowledged and/or properly cited**.

2 Evaluation

The grade will be partly based on the **clarity** of your report, so please make sure your final report is well organized and clearly written. There should be an introductory part explaining the basics of your work, and a conclusions section, basically stating what you know compared to what you knew before the work started; also any gaps, possible extensions or limitations in your development should be noted and explained.

Your work will also be evaluated based on **technical quality**. This means that the techniques you use should be reasonable, the stated results should be accurate, and the technical results should be correct and complete.

[1] http://cran.r-project.org/

[2] https://cran.r-project.org/web/packages/available_packages_by_name.html

[3] https://cran.r-project.org/manuals.html#R-admin

In summary, these are the conditions for a high score (in this order):

1. The (good) use of techniques and methods presented in class
2. The care and rigor for obtaining the results (validation protocol, statistical significance)
3. The quality of the obtained results (generalization error, simplicity)
4. The quality of the written report (conciseness, completeness, clarity).

In addition, as you probably know, there is a **generic competence** (or *skill*) associated to this course: "*Effective communication*"[4], which is worth an additional 10% of the final grade. In order to help you deal with this, the **rubric** with which the competence will be evaluated is available to everyone prior to delivery.

3 Detailed information

The **final report** that you must deliver should include the following **sections**:

1. A brief but self-contained description of the work and its goals, and of the available data, and any additional information that you have gathered and used
2. A brief description of related previous work and results
3. The relevant data exploration process (pre-processing, feature extraction/selection, clustering and visualization)
4. The resampling protocol (training/test, cross-validation, etc) that you have used (see 3.3)
5. The results obtained using **at least three linear/quadratic** methods (indicating the best set of parameters for each one):
   (a) If the task is **classification**, any of: logistic regression, multinomial regression (single-layer MLP), LDA, QDA, RDA, Naive Bayes, nearest-neighbours, linear SVM, quadratic SVM
   (b) If the task is **regression**, any of: linear regression, ridge regression, the LASSO, nearest-neighbours, linear SVM, quadratic SVM
6. The results obtained using **at least two general non-linear** methods (indicating the best set of parameters for each one); for both **classification** and **regression** tasks, any of: one-hidden-layer MLP, the RBFNN, the SVM with RBF kernel, a Random Forest
7. A description and justification of the final model chosen, and a honest estimation of its generalization error (see 3.3)
8. A final part (one to two pages) containing:
   (a) A self-assessment of successes, failures and doubts (I suggest this to be a list of one-line items)
   (b) Scientific and personal conclusions
   (c) Possible extensions and known limitations
9. References to all your used sources: books, web pages, code, scientific papers, ...

Mechanism for delivery: You will be required to submit a written report (a pdf file) and the **full code** (in separate files, and only electronically) and a brief text file with instructions on how to execute your code. The report should *not* include explanations of the methods seen in class. All deliveries are exclusively by means of the "Racó" in a **single compressed file**; you do not have to print anything.

[4] Expressió oral i escrita.

3.1 Obtaining the problems

Be sure to browse the following repositories:

1. The UCI machine learning repository:
   `http://archive.ics.uci.edu/ml/`
2. The School of Informatics (University of Edinburgh) repository:
   `http://www.inf.ed.ac.uk/teaching/courses/dme/html/datasets0405.html`
3. The datasets that arose from the Delve project:
   `http://www.cs.toronto.edu/∼delve/data/datasets.html`

Then choose *one* of the problems (most of which are real-world tasks and many are quite challenging). Their origins are very diverse, not only regarding the area of work (biology, geophysics, medicine, etc) but because they show different data characteristics. For example, there are great differences in the number of variables and examples, number of classes, intrinsic difficulty, lost values, various errors, mixed nominal and/or continuous variables, etc. Some problems are synthetic (they have been generated by a program), and their characteristics are completely known. However, their study is interesting for a number of reasons, including meaningful (as well as significant) comparisons of different learning algorithms.

Some problems are then easier in some aspects and more difficult in others. Therefore, the selection of the particular problem does not have much importance for the grade. In particular, it is not at all advisable that you start to test problems to see how they "behave". It is recommended that you base the decision on the interest that it raises in you.

3.2 On data pre-processing

Each problem requires a different approach in what concerns data cleaning and preparation, and the selection of the particular information you are going to use can vary; this pre-process is very important because it can have a deep impact on future performance; it can easily take you a significant part of the time. It is then strongly advised that you analyse well the data before doing anything, in order to gauge the best way to pre-process it [5]. In particular, you shall pay attention to the following aspects (not necessarily in this order):

1. treatment of lost values (missing values)
2. treatment of anomalous values (outliers)
3. treatment of incoherent or incorrect values
4. coding of non-continuous or non-ordered variables (nominal or binary)
5. possible elimination of irrelevant or redundant variables (feature selection)
6. creation of new variables that can be useful (feature extraction)
7. normalization of the variables (*e.g.* standardization)
8. transformation of the variables (*e.g.* correction of serious skewness and/or kurtosis)

3.3 On model selection and estimation of performance

In accordance with the problem and the available data, you should design a set of experiments based on valid protocols to select models and to honestly estimate the generalization error (or any other measure of future performance) of the final proposed model or solution.

Some problems come with their own test data (data used for the estimation of true generalization error), some do not; in the latter case, you must obtain test data by splitting the full available data (once or several times, depending on the data size). For model selection, k-fold cross-validation will probably be necessary (the selection of the best value for k is

[5] See Laboratory 0.

your decision). It is methodologically prohibited to use as test data information that has already been used for the creation, adjustment or selection of the solution.

Important information:

1. Delivery date: **January 17, 2018** (this date is **strict**)
2. You must form teams of 2 people (3 are possible upon **explicit permission**)
3. The project and its generic competence are worth 50% of the final grade

---

You must decide which **problem** you want to attack and communicate your choice to the course coordinator no later than **November 6, 2017**. Please indicate **all** the group member's names.