Machine Learning Project

Madalina Olteanu & Fabrice Rossi

The goal of the machine learning project is to analyse a realistic data set both from an explanatory point of view (unsupervised learning) and from a predictive point of view (supervised learning). The analyses can be implemented either in Python (using Pandas, Scikit-learn, etc.) or in R (with the tidyverse packages).

1 Data files

The data studied in this project are stored in several CSV files documented below.

1.1 Persons

The main file `learn.csv` contains a description of 100000 French persons (from Metropolitan France). Persons are described by the following variables:

- `UID`: primary key (unique identifier);
- `F_name`: first name;
- `LASTNAME`: last name;
- `Age_2019`: age;
- `SEX`: sex;
- `household_type`: family type;
- `highest_degree`: highest diploma;
- `Activity_type`: activity type;
- `Is_student`: true if the person is a student;
- `Occupation_42`: socio-professional category (PCS 2003 norm, see below);
- `Insee`: INSEE code of the city of residence.

1.2 Job

The current jobs of persons with an employee status are described in the `learn_job.csv` file. The description uses the following variables:

- `UID`: foreign key to the person table;
- `ECONOMIC_ACTIVITY`: economic sector of the job;
- `Terms_of_emp`: work contract type;

- Job_category: type of job (regular, intersnship, etc.);
- JOB_CONDITION: job terms (full-time, part-time, etc.);
- WORKING_HOURS: total annual working hours (this variable has missing values);
- Employer_category: type of employers;
- employee_count: size of the company;
- JOB_DEP: department in which the job is located;
- Work_desc: description of the job according to the PCS-ESE 2017 norm (see below);
- EMOLUMENT: annual salary of the person.

Moreover all persons with a job have a EMP variable that describes their type of job. This is given in the file learn_EMP.csv. It contains a foreign key UID to the person table and the associated EMP variable. Notice that persons with a job do not necessarily have an employee status.

1.3 Sport

Some persons belong to a sport club. This is documented in the file learn_Sports.csv. It contains a foreign key UID to the person table and the associated Sports variable.

1.4 Test set

The predictive task of the project consists in building a predictive model for the EMOLUMENT variable. This applies only to persons with an employee status.

A test set with 10174 persons is provided for evaluation purposes. It does not contain the EMOLUMENT variable. To ease the implementation of the task, the test set is a join between person level information and job level information (including the EMP variable). This is provided in the test.csv file. Sport related information is provided in test_Sports.csv.

1.5 Categorical variables and geography

Most variables are categorical. The possible values are listed and documented in CSV files named after the variables (e.g. code_highest_degree.csv for the highest_degree variable). Notice that those files have been produced by INSEE and are written in French. The PCS-ESE 2017 INSEE norm is described by the following files:

- code_Work_desc.csv contains the association between codes and profession;
- code_Work_desc_map.csv contains a mapping between the complete codes (N3) used in the data set and two coarser representations (N1 and N2);
- code_Work_desc_n2.csv contains the association between codes and profession groups a the N2 level;
- code_Work_desc_n1.csv contains the association between codes and profession groups a the N1 level.

The PCS 2003 is a complementary norm which adds modalities to the N2 level of the PCS-ESE 2017. Codes are given in code_Occupation_42.csv

Geographical and administrative information about metropolitan French cities is contained in several files:

- city_adm.csv contains administrative information:
  - Nom de la commune: city name
  - Insee: INSEE code of the city;
  - Dep: code of the department of the city;
  - TOWN_TYPE: city type (modalities are administrative city category);
- city_loc.csv contains geographical information, the GPS coordinates of the cities expressed in the WSG 84 system[1] as well as in the Lambert-93 projection[2]. The Lambert-93 coordinates can be used to compute distances (in meters) between cities with a reasonable precision in metropolitan France. Attributes:
  - Insee: INSEE code of the city;
  - Lat: latitude;
  - long: longitude;
  - X: X Lambert coordinate;
  - Y: Y Lambert coordinate;
- city_pop.csv contains population information:
  - Insee: INSEE code of the city;
  - Inhabitants: population of the city.
- deparments.csv contains departments information:
  - Nom du département: department name;
  - Dep: code of the department;
  - Reg: code of the region to which the department belongs.
- regions.csv contains region information (from 2018):
  - Nom de la région: region name;
  - Reg: code of the region.

2 Expected results

Two different analyses must be conducted on the data set. They can be conducted mostly independently. The analyses must use all the persons in the data set, including those with missing data.

2.1 Explanatory analysis

The goal of the exploratory analysis is to characterise French departments from the point of view of their composition in terms of general population and in terms of working population. The analysis should be implemented in two steps.

The first step should consist in creating two data sets aggregated at the department level. In these data sets, each French metropolitan department should be described by statistics

[1] https://en.wikipedia.org/wiki/World_Geodetic_System

[2] https://en.wikipedia.org/wiki/Lambert_conformal_conic_projection

summarising the population composition. For instance, one could describe a department by the percentage of its Female inhabitants, its number of inhabitants, the distribution of the inhabitants education degrees, of the salaries, etc. It is recommend to include numerical variables only in the descriptions to ease the analysis. The first data set should correspond to the departments as characterised by their general population, while in the second data set departments have to be described using the working population. The report must contain a description of the statistics used to represent the departments.

In a second step, the two data sets must be analysed using clustering and visualisation algorithms. The goal of the analysis is to assess whether French departments are homogeneous in terms of population composition or, on the contrary, separated into different groups. You may use any clustering algorithms seen during the class, and subsequently assess cluster composition and relative positioning using linear and/or nonlinear mapping, statistical tests based on some analysis of variance, etc. Comparing the inhabitants based description to the workers based one is also very important. Eventually, you may consider illustrating whether the resulting clusters are spatially correlated, using a geographical mapping.

2.2 Predictive analysis

The goal of the predictive analysis is to build a predictive model for the `EMOLUMENT` given the other variables. A predictive model must be constructed using only the learning set (`learn.csv` and all `learn_*.csv` files, possibly augmented, see below). Then the selected model must be applied to the test set (`test.csv`) to produce predicted values of `EMOLUMENT`. Those predictions must be saved in a CSV file with a column named `EMOLUMENT` containing the predictions and a column name `UID` used to identified the persons. This CSV file must be included in the submission. Notice that only the CSV format is allowed, with standard US representation of decimal numbers.

The analysis must follow these rules:

- the use of a resampling technique to select the best model is mandatory (this can be for instance cross-validation for general models, and leave-one-out or out-of-bag estimates for specific ones);
- the project report must include an estimate of the expected performances of the model on the test set;
- the quality of the predictions will be taken into account to grade the project and it is therefore important to use state-of-the-art predictive methods.

In addition, it is recommended:

- to complement the core data sets by the geographical information available in the complementary data set;
- to use category simplification/grouping if needed for categorical variables. In particular, most predictive models will have difficulties with the `Work_desc` variable if used directly. It is acceptable to use of external data to simplify the categories;
- external data can be used to complement the features;
- as pointed out in the data description, the `WORKING_HOURS` variable has missing value, but the corresponding persons cannot be discarded from the analysis, neither at the learning phase, nor at the testing phase.

3 Project submission

The results of the project must be submitted as a single zip file containing:

- a report on the exploratory analysis (exclusively in pdf format, other format will be discarded). This report should describe the analysis, using graphical illustration, conditional tables, etc. The report should focus on the findings more than on the way to get to them. It must include a description of the features used to characterized departments;
- a report on the predictive analysis (exclusively in pdf format, other format will be discarded): this report should be short and very precise. It should outline the methodology used to construct the chosen model. It must contain an estimation of the quality of the prediction on the test set;
- a file named `predictions.csv` with the predicted values of `EMOLUMENT` on the test set (`test.csv`);
- the full code used to perform the analysis (in two separated files, one for the exploratory analysis, one for the predictive analysis).

For R users, it is strongly recommended to use the Rmarkdown system to build reports directly from the code used to perform the analyses. In this case, the full code is the markdown file itself. If you use Rmarkdown, you should submit a zip file containing 5 files (2 reports in pdf, a csv file, 2 Rmarkdown files).

For Python users, the use of notebooks is recommended. The report can be the pdf rendering of the notebook[3], but this is not recommended as this leads to barely readable reports. A possible solution is provided by Jupyter Book[4]. The code must be submitted as both the notebook file itself and an extracted Python code. Thus if you use a notebook, you should submit a zip file containing 7 files (2 reports in pdf, a csv file, 2 notebooks and 2 python files extracted from the notebooks).

Notice that no manual editing of the data files via e.g. excel is permitted. In particular, if data files must be combined, this has to be done with R/Python.

[3] It might by necessary to install pyppeteer for pdf export.

[4] https://jupyterbook.org/intro.html