Machine Learning
Programming Assignment II-a

Ujjwal Sharma and dr. Stevan Rudinac

The following assignments will test your understanding of topics covered in the first three weeks of the course. These assignments **will count towards your grade** and should be submitted through Canvas by **30.11.2018 at 12:59 (CET)**. You can choose to work individually or in pairs. You can get at most 6 points for these assignments, which is 6% of your final grade.

Submission

You can submit either a Jupyter Notebook (*.ipynb) or a python program file (*.py). To test the code we will use Anaconda Python 3.6. Please state the names and student is of the authors (at most two) at the top of the submitted file.

1 Implementation Details

In this assignment, we will be looking at data preprocessing as well as classification tasks. Since they have a fixed sequence of execution, you will be required to use the sklearn `Pipeline` functionality to encapsulate your preprocessing transforms as well as classification models into a single estimator. In the following assignments, you should perform fitting and prediction with a `Pipeline` estimator.

Any grid search should also be performed on the `Pipeline`, not on independent transforms.

2 Data

Data for this assignment is available here. In the zip file, you will find:

- A data file titled `votes.csv`. You will need to split this file yourself. For this assignment, you can use the default split ratio in `sklearn.model_selection.train_test_split`.
- An additional file `county_facts_dictionary.csv` containing descriptions for demographic data attributes in the accompanying data.

The supplied data consists of demographic statistics for all counties in the United States as well as vote counts for Hillary Clinton and Donald J. Trump. Your task is to use this demographic data to train a classifier to predict the electoral outcomes in these counties.

3 Classification

Building on the classifiers introduced in the previous week, we will be working with the `LinearSVC` and the `LogisticRegression` classifiers this week. This homework contains three distinct tasks:

✉ u.sharma@uva.nl, s.rudinac@uva.nl

3.1 Data Preprocessing

As before, `pandas` is an immensely helpful tool to work with tabular data. Before starting, you will need to perform two tasks:

1. Cut out the relevant features to be used as input. The following attributes need to be separated as input data:

   ```
   ['population2014', 'population2010', 'population_change', 'POP010210',
   'AGE135214', 'AGE295214', 'age65plus', 'SEX255214', 'White', 'Black',
   'RHI325214', 'RHI425214', 'RHI525214', 'RHI625214', 'Hispanic', 'RHI825214',
   'POP715213', 'POP645213', 'NonEnglish', 'Edu_highschool', 'Edu_batchelors',
   'VET605213', 'LFE305213', 'HSG010214', 'HSG445213', 'HSG096213', 'HSG495213',
   'HSD410213', 'HSD310213', 'Income', 'INC110213', 'Poverty', 'BZA010213',
   'BZA110213', 'BZA115213', 'NES010213', 'SBO001207', 'SBO315207', 'SBO115207',
   'SBO215207', 'SBO515207', 'SBO415207', 'SBO015207', 'MAN450207', 'WTN220207',
   'RTN130207', 'RTN131207', 'AFN120207', 'BPS030214', 'LND110210']
   ```

   You will find descriptions for these attributes in the `county_facts_dictionary.csv` file.

2. You will have to create a target value. Since the data only contains the percentage of votes in favor of both the candidates, you will need to create a new column containing a binary variable that denotes which candidate won a majority in that county.

3. Finally, you will scale the data and choose an optimal method for this. For this task, you should experiment with the data preprocessing methods discussed in class and settle on what works best.

4. For both the classifiers, you must choose an optimal cross-validation strategy. As always, please explain the reasoning behind your choice in a text box.

3.2 Classification

You will be required to implement the `LinearSVC` and `LogisticRegression` classifiers.

1. Create a `LinearSVC` classifier with default parameters. Fit your classifier on scaled as well as unscaled versions of the data and report the estimator scores. Additionally, analyze the effects of data preprocessing on the classification performance.

   In no more than 50 words, explain your observations and your assessment of the underlying situation. You can use a text box (i.e. Markdown Cell) in Jupyter to write down your analysis.

2. Create a `LogisticRegression` classifier. Additionally use `GridSearchCV` to find an optimal value for the parameter `C`.

   As before, you will be required to run your estimator on scaled as well as unscaled versions of the data and report your observations for the estimator score. Additionally also analyze the effects of regularization strength on the performance of the estimator [1]. In not more than 50 words, explain your observations and your assessment of the effects of `C` and data scaling on the classification performance.

Note: If you make a design choice with regards to any aspect of this homework, please explain the underlying reasoning in an appropriately placed Markdown cell. For example, if you choose to select a certain scaling strategy for data preprocessing, explain why you feel that this strategy is better suited to the problem than the alternatives. Your final grade will incorporate points for these notes.

[1] The `LogisticRegression` hyperparameter C represents the *inverse* of regularization strength. So a smaller value of C is, in fact, stronger regularization.