Introduction to Machine Learning Course

Exercise 2 –
The Elections Challenge – Data Preparation

At last we are about to start handling the Elections challenge. Similarly to any other machine learning project, our first step is the data preparation. By the end of this task, you should have 3x2 data sets, as follows:

- Split the data to – train (50-75%), validation, (25-15%), test (25-10%)
- For each set –
    - Keep a copy of the raw-data
        - These data-sets will serve as references, in case you would like to re-examine (and possibly reconsider) the transformations you've made
    - Prepare the data for use
        - Clean, scaled, transformed, without missing values, while keeping only the relevant attributes

Mandatory Assignment

Write a Python script that will execute the following

1. Load the Election Challenge data from the ElectionsData.csv file
    - Can be found at the "The Election Challenge" section in the course site
2. Identify and set the correct type of each attribute
    - It is advised to do it with Pandas
    - The "Vote" attribute serves as the label
3. Perform the following data preparation tasks using ALL the data
    - Imputation
        - Filling up missing values
    - Data Cleansing
        - Outlier detection
        - Type/Value modification – Use mainly for Nominal attributes
    - Normalization (scaling)
        - Use with care, do not destroy attribute dependencies
    - Feature Selection
        - Use at least one filter method and one wrapper method
4. Split the data to train, test, validation sets
5. Save the 3x2 data sets in CSV files

Please submit

1. The list of selected features
2. The Python files
3. A short documentation that explains your process and any significant decision/insight you would like to share

Non-Mandatory (Bonus) Assignments

The following list includes additional, non-mandatory, assignments. You are highly encouraged to do at least some of them. Each functional implementation of ANY assignment will get a bonus, but more importantly – it will help you in getting better results

A. Perform data preparation actions only on the training set and apply the resulted transformations to both the test and the validation sets
    - Compare with results you obtained when using ALL the data and discuss whether you've witnessed any significant degradation in results
B. Construct a hybrid feature selection scheme
    - A hybrid scheme – gets the selected attributes from the various methods and makes the final selection
C. Identify the role of the attributes with respect to the classes (the "Vote" label)
    - Provide an explanation (in the submitted documentation)
D. Implement the Relief algorithm, and use it for feature selection
    - Provide a python file with your own implementation
        - Calling a package that includes Relief doesn't count … ☺
    - Compare and discuss pros and cons of the feature selection capabilities of Relief vs the other methods that you have used
        - Where there features that were only/not selected by Relief?

Triplets Mandatory Assignments (Bonus for Pairs)

The following list includes mandatory assignments for triplets. Triplets must submit all the assignments. Pairs are highly encouraged to submit at least some of them, and for pairs it will be considered as bonus assignments

A. Implement generic Sequential Forward Selection (SFS) wrapper, capable of getting as a parameter a base model. The base model will be (re)trained and used for scoring throughout the progress of the forward selection strategy. Use SFS with at least two base algorithms for feature selection
    - Provide a python file with your own implementation
        - Calling a package that includes SFS doesn't count … ☺
    - Compare and discuss pros and cons of the feature selection capabilities of SFS vs the other methods that you have used
    - Where there features that were only/not selected by SFS?
B. Implement EM training for Multivariate Gaussian
    - Provide a python file with your own implementation
        - Calling a package that includes EM for Gaussian doesn't count … ☺
    - Provide a script that conduct a (toy) comparison for various options of Bivariate (2D) Gaussian training and plot the results
        - See the "Multivariate Gaussians Parameter Estimation using Partially Observed Data" slide

Comments

- The three “prepared” data sets that were generated by your process should NOT include ANY missing value
- The minimal set of useful attributes includes less than half of the original attributes, and the set is not unique
    - Namely, there are redundancies among the useful attributes
- It is difficult (might be even impossible) for you at this point to identify and select exactly the set of useful attributes
    - Our grading policy will favor approximating this set as close as possible, but will “punish” any loss of a useful attribute
    - This means that you should employ a conservative approach!
        - Recall that it is difficult to recover, at later stages, from the loss of relevant features at the data preparation stage
- Keep plotting and looking at the data in various forms and aspects
    - You’ll be amazed how much insights can be gained from visualization
- The leading Python Packages that you should use are Pandas, Scikit-learn, and Matplotlib. These packages include a rich set of very useful examples
    - You are highly encouraged to look at it, “steal” some ideas, and shorten your implementation time
    - Moreover, It is permissible, and actually recommended, to look for additional Python packages and implementations, and either use explicitly or borrow ideas

This exercise can be submitted in pairs or triplets!

- **You should submit only one copy but remember to document who are the contributors**
- **“No Couples/Triplets Swapping” during the semester**
    - **At least not without my formal approval**
- **Triplets must submit the bonus assignments that are marked mandatory for triplets**