Machine Learning Human Rights and Wrongs

This repository contains replication materials for the paper, *Kevin Greene, Michael Colaresi, and Baekkwan Park "Machine Learning Human Rights and Wrongs: How Supervised Learning Using Texts Can Inform the Debate about Changing Standards of Human Rights"* in *Political Analysis*.

For questions, contact Baekkwan Park (baekkwan.park@gmail.com)

Directory structure and Files (See Directory Structure Section)

- **code**: All the scripts to run the models and plot figures in the paper and the appendix.
- **data**: documents and annotated list for the documents
- **outputs**: all the outputs from the models
  - *console*: log outputs for models (.txt)
  - *model_output*: model outputs (.csv)
  - *plots*: plots (.pdf)
  - *vi_outputs*: variable importance outputs (.csv)

Requirements

- Python3
  - scikit-learn 0.19.1
  - SciPy 1.0.0
  - Numpy 1.13.3
- R 3.4.2

Setup and Executing the Scripts

1. Run Step by Step (Estimated Running Time: 30 hours)

- If you already have all the requirements listed above on your computer, you can run directly. If not, install the following softwares and packages:

  - Python3
    - scikit-learn 0.19.1

        - SciPy 1.0.0
        - Numpy 1.13.3
    - R 3.4.2

- Make sure your current working directory is **REPLICATION_FILE**

- In the folder **REPLICATION_FILE**, you should be able to see the following subfolders and files.

- In the ***code*** folder, you have all the code. In your terminal, you can run them one by one in the following order.

  First, run **main.py** file for python.

Second, run **Rolling_Accuracy.R** file for Rscript.

Third, run **Feature_Importance.R** file for Rscript.

- All the results and figures will be exported to **outputs** folder.

2. Run with Virtual Environment (Estimated Running Time: 30 hours)

- If you want to run with Python's virtual environment, first, make sure that you have Python3.

- Make sure your current working directory is **REPLICATION_FILE**
- There are two bash files: **setup.sh__ and __execute.sh**. Simply run one by one, then it will run all the code.
- **setup.sh**: this will create a virtual environment for python and install all the requirements (scikit learn, numpy, scipy, etc.) to run the script.

```
$ ./setup.sh
```

- **execute.sh**: this will run all the models and plot figures.

```
$ ./execute.sh
```

Directory Structure (REPLICATION_FILE)

1. Folder Structure Only

In the **REPLICATION_FILE** folder, you can find the following sub-folders and instruction files.

```
REPLICATION_FILE
├── README.md
├── README.pdf
├── Figures_Table_in_the_Paper
├── setup.sh
├── execute.sh
├── code
├── data
└── outputs
    ├── model_output
    ├── console
    │   ├── fixed_5
    │   │   └── all
    │   ├── fixed_10
    │   │   ├── all
    │   │   └── appendix_tables
    │   └── sequential
    ├── plots
    └── vi_output
```

2. Files in Each directory and Comments

There are files and sub-directories in each directory.

```
REPLICATION FILE
|
├── README.md:
|         This is the file that you are reading right now.
|
├── setup.sh:
|         As explained above, this will set up for python virtual
|         environment and installing all the requirements.
|
├── execute.sh:
|         As explained above, by running this file, you run all
|         the models and draw plots.
|
├── Figures_Table_in_the_Paper.pdf:
|         This explains all the input and output files used to generate
|         tables and figures in the article.
|
├── code
|    |
|    ├── main.py:
|    |       This is the file that runs all the models the paper.
|    |       You do not need to run the rest of the code individually
|    |       because they are all imported and used as modules in this file.
|    |
|    ├── csv_sequence.py:
|    |       It calls the raw documents from annotated_list to
|    |       data_processing.
|    |
|    ├── data_processing.py:
|    |       It converts all the raw documents into document term matrices for
|    |       models.
|    |
|    ├── classifier.py:
|    |       It estimates all the models (classifiers) used in the article.
|    |       (Naive Bayes, Logistic Regression, SVM, Random Forests)
|    |
|    ├── model_testing.py
|    |       It runs all the classifiers for rolling window prediction.
|    |
|    ├── voting_classifier.py
|    |       It runs majority vote classifier for rolling window predictions.
|    |
|    ├── bias.py
|    |       It calculates in-window and out-window bias for each model.
|    |
|    ├── random_forest_features.py
|    |       It calculates feature importance for random forests models.
```

```
│   │
│   ├── output.py
│   │       It writes, save and export all the results from each model in
│   │       *.csv and *.txt format.
│   │
│   ├── Rolling_Accuracy.R:
│   │       It draws line plots for accuracies for all models.
│   │
│   ├── Feature_Importance.R:
│   │       It draw plots for feature importance outputs.
│   │
│   └── requirements.txt:
│           This is for installing all the requirements when using virtual
│           environment.
│
├── data
│   │
│   ├── documents:
│   │       This directory contains about 5,000 txt files from the State
│   │       Department Reports (1978 - 2010).
│   │
│   └── annotated_list.csv:
│           This file lists classification labels for all the documents.
│
└── outputs
    │
    ├── model_output:
    │   │   This folder contains results from rolling window predictions
    │   │   (5 years and 10 years respectively) and sequential
    │   │    window predictions (accuracy and bias for each window).
    │   │
    │   ├── fixed_5_model_evaluation.csv:
    │   │     This is 5 year rolling window prediction results.
    │   │
    │   ├── fixed_10_model_evaluation.csv:
    │   │     This is 10 year rolling window prediction results.
    │   │
    │   └── sequential_model_evaluation.csv:
    │         This is forward sequential window prediction results.
    │
    ├── console
    │   │   This directory contains log outputs for all the models. You can
    │   │   find all the outputs for cross validation, tuning parameters,
    │   │   various forms of evaluation metrics(precision, recall, F1,
    │   │   accuracies, confusion matrix, etc.)
    │   │
    │   ├── fixed_5
    │   │   │
    │   │   └── all:
    │   │     This contains all the raw log outputs for fixed 5 year
    │   │     rolling window prediction.
    │   │
    │   ├── fixed_10
    │   │   │
    │   │   ├── all:
    │   │   │   This contains all the raw log outputs for fixed 10 year
```

```
│   │   │   rolling window prediction.
│   │   │
│   │   └── appendix_tables:
│   │       This folder contains all the raw log outputs used to create
│   │       Table 2 in the Appendix and the confusion matrices.
│   │
│   │
│   └── sequential:
│         This contains all the raw log outputs for sequential rolling
│         window prediction.
│
├── plots:
│   │    This directory contains all the plots used in the paper.
│   │     (see "Figures_Table_in_Paper.pdf" file for detail.)
│   │
│   ├── 5_Rolling.pdf
│   ├── 10_Rolling_bw.pdf
│   ├── 10_Rolling.pdf
│   ├── Seq_FWD.pdf
│   ├── SimulatedAccuraciesOutAndIn.pdf
│   ├── VarImpFullEarly1.pdf
│   ├── VarImpFullLate1.pdf
│   ├── VarImpMain_bw.pdf
│   └── VarImpMain_color.pdf
│
└── vi_output
        This contains all the results of feature importance analysis.
```