

Implementation.

extract_features.py

```
~/Code/academic/mtp/reegmo master*  
datacon > python extract_features.py --help  
usage: extract_features.py [-h] [-e] [-m] [-b] [-s] [-a] [--all] [-w WINSIZE]  
  
Extract features from EEG sessions  
  
optional arguments:  
  -h, --help                show this help message and exit  
  -e, --extract-only        Only extract each session into pickle  
  -m, --merge-only          Only merge all sessions into a single pickle  
  -b, --baseline            Subtract mean value of the respective channel from each channel  
  -s, --standardize         Standardize each channel  
  -a, --average             Average out all channels  
  --all                    Extract all possible combinations  
  -w WINSIZE, --winsize WINSIZE  
                           Window Size in seconds
```


Implementation.

classify.py

```
~/Code/academic/mtp/reegmo master* 6s
datacon > python classify.py --help
usage: classify.py [-h] [-b] [-s] [-a] [-d {random,subjectid} [{random,subjectid} ... ]]
                  [-m {random_forest,xgboost} [{random_forest,xgboost} ... ]]
                  [-r RANDOM_STATE] [-tr TEST_SIZE] [-ts TEST_SUBJECT_SIZE]
                  [-c {2,3,-1}] [--all]

Supervised classification of EEG features

optional arguments:
  -h, --help                show this help message and exit
  -b, --baseline            Subtract mean value of the respective channel from each channel
                           -- Use this feature
  -s, --standardize        Standardize each channel -- Use this feature
  -a, --average            Average out all channels -- Use this feature
  -d {random,subjectid} [{random,subjectid} ... ], --divtype {random,subjectid} [{random,subjectid} ... ]
                           How to split the data into train and test sets
  -m {random_forest,xgboost} [{random_forest,xgboost} ... ], --model {random_forest,xgboost} [{random_forest,xgboost} ... ]
                           Model(s) to use for classification
  -r RANDOM_STATE, --random-state RANDOM_STATE
                           Random State for Shuffle
  -tr TEST_SIZE, --test-size TEST_SIZE
                           Test size in fraction (of total data points)
  -ts TEST_SUBJECT_SIZE, --test-subject-size TEST_SUBJECT_SIZE
                           Test size in fraction (of number of subjects)
  -c {2,3,-1}, --num-classes {2,3,-1}
                           Number of classes into which y will be divided
  --all                    Use features from all possible combinations
```