ReadMe document for source code and Demo

1. **Background**

This document provides readme documentation for the source code used to generate the results for the manuscript entitled “Machine Learning of in vivo Tissue Electrophysiology in Patients with Heart Failure “.

There is also a demo code to show the results of the trained model for a sample dataset.

1. **Files attached**

* Code: The main code is presented in a Jupyter notebook format (.ipynb). The code provides information for all the steps and the functions that are used in the result. Comments are provided for each function.
  + The main code is saved in “Code\_for\_manuscript\_submission.ipynb” as jupyter notebook in the Code folder. The rest of the functions are helper functions saved as a .py file. The full source code will only run with full dataset (we only provide demo dataset).
* Data: Only a subset of the data is provided for demo purposes and can be found in the Demo folder.
* Demo Folder contents:
  + 20191219 is an excel sheet showing the actual labels for both VTVF and Mortality endpoints and whether they were in training/validation splits in the trained model
  + Calc\_metrics\_v2: helper function to calculate accuracy, sensitivity, specificity, NPV, PPV.
  + Demo\_Code.ipynb: Jupyter notebook to run demo. Running this demo will show results from the sample dataset provided.
  + Mortality\_CV1\_finalized\_mode.sav: trained model using cross validation 1 for mortality endpoint.
  + VTVF\_CV1\_finalized\_model.sav: trained model using cross validation 1 for VTVF endpoint.
  + Mortality\_labels\_demo & Mortality\_tsfresh\_features\_demo: input and true output for model (Mortality). First 5 digit contain patient ID, and last 4 digits contain beat ID.
  + VTVF\_labels\_demo & VTVF\_tsfresh\_features\_demo: input and true output for model (VTVF).
  + demo\_data\_20191219.npz: numpy (.npz) file that has the voltage-timeseries MAPs. Each point is 1msec apart and the values are voltages in mV.
* Results and Data folder: empty folder to hold and use data (this will not have any files in them, they are used to generate full results for full dataset).

1. **Hardware and Software**

This program runs on a desktop computer system with following specifications:

* Inter Core i9-9900K CPU @3.6Ghz, 3600 Mhz, \* cores, 16 Logical Processors
* Microsoft Windows 10 Pro
* 32 GB RAM

All computations were performed on Python 3.6 using Anaconda Navigator 1.9.7. The following packages were used:

* numpy 1.17.3
* pandas 0.23.4
* pandas-datareader 0.8.0
* scikit-learn 0.21.3
* scipy 1.3.1
* tsfresh 0.12.0
* xlrd 1.2.0
* xlsxwriter 1.2.6
* jupyter 1.0.0
* pickle 1.0 or higher
* matplotlib 3.3.0

1. **Time to execute**

All runtimes are based on the hardware specifications provided in section 3. Most commands run in less than 1 minute and a few take up to 3 minutes. The only part that takes a considerable time (~15 minutes) is “extract\_features\_without\_label” which extracts all the tsfresh features. This command is performed in the “Code\_for\_manuscript\_submission.ipynb” Jupyter file under section “Feature extraction using tsfresh” (this requires the full dataset to run. Only code is provided).

1. **Instructions to run**

If downloaded as a zipped folder, you will need to unzip the folder.

Running every cell in the Demo Jupyter file notebook in order will run the program and produce the results in the Jupyter notebook.

The Demo also allows to plot different MAP beats for the demo files (can be configured). The runtime for the demo file takes a few minutes (tested on machine with specifications in section 3, but it should run fast on any modern machine as well).

1. **Software License**

Creative Commons Attribution-NonCommercial-NoDerivatives