Documentation – ARCode/
Aakash Gupta

1. AG_Scripts
    - Contains several scripts I wrote as side projects
    - Nothing in here should be of use

2. clips_healthy
    - Contains the generated clips files for the healthy control
    - One file for each healthy control

3. clips_patients
    - Contains the generated clips files for the C-Brace patients
    - One file for each C-Brace patient for each brace

4. code
    - Contains all the code and subfolders
    - More detailed explanation on page 2

5. features_healthy
    - Contains the features files for the C-Brace patients
    - One file for each C-Brace patient

6. features_patient
    - Contains the features files for the C-Brace patients
    - One file for each C-Brace patient for each brace

7. home_data
    - Create one folder for each patient and follow the naming format
    - Each patient folder contains both CBR and SCO braces
    - Each patient folder should have the home data file and wearing data file generated from ActiLife
    - README.txt contains the information for how to format the filenames and wearing data file

8. home_labeled
    - Create one folder for each patient and brace and follow the naming format
    - The folder contains all the home labeled clips
    - The features .mat file and classifiers .mat file is generated and outputted to the home_labeled folder

9. raw
    - One folder for each patient and brace
    - One folder for each healthy control
    - Each folder contains the labeled lab sessions

Documentation – ARCode/code/

1. Generate clips and features from the lab session .csv files
   - Use *generateClips.m*
     - Lab session files should be present in the **raw** folder under the patient folder
     - Add "_p" to the end of the folder names in **raw** that you want to generate clips for
     - Will output the clips .mat file for each patient in the **clips_healthy** or **clips_patient** folder
   - Use *dataPreprocessingPAR.m*
     - Add "_p" to the end of the folder names in **raw** that you want to generate features for
     - The code will access the files in the clips folder and generate features in the correct patient or healthy folder
     - The code will also ask whether this is "healthy" or "patients" being analyzed
   - Use *classifierDataCreate.m*
     - Asks whether this is "healthy" or "patients" being analyzed
     - Will consolidate ALL the features files in either **features_healthy** or **features_patient** folders
     - Outputs a .mat file in the **code** folder
     - Filenames generated are "trainData_healthy.mat" and "trainData_patient.mat" in **code** folder

2. Use classifier for cross-validation (CV) on lab data
   - Subject-wise, session-wise, and artificial models (3 CV models)
     - Support Vector Machine – *kSubject_SVM.m, kFold_SVM_Session.m, kFold_SVM_Artificial.m*
     - Random Forest – *kSubject_RF.m, kFold_RF_Session.m, kFold_RF_Artificial.m*
     - Naïve Bayesian – *kSubject_NB.m, kFold_NB_Session.m, kFold_NB_Artificial.m*
     - HMM – *kSubject_RF_HMM.m, kFold_RF_HMM_Session.m, kFold_RF_HMM_Artificial.m*
   - Using the SVM files
     - SVM requires an optimized sigma value when using the Gaussian (RBF) kernel
     - An Excel spreadsheet of optimized sigma values is **ARCode/SVM RBF Sigma Results.xlsx**
   - Calculate optimized sigma values
     - *kSubject_SVM_SIGMA.m, kFold_SVM_Session_SIGMA.m, kFold_SVM_Artificial_SIGMA.m*
     - Enter desired sigma values in the code in the sigma variable
     - Codes will plot the accuracy against sigma value on a semilog scale
     - Write down the optimized sigma value corresponding to the highest accuracy

3. Plotting
   - *plot_acc.m*
     - Plot representative accelerations graph for poster
     - Import lab csv file and label first four columns as t, x, y, and z variables
   - *plot_scatter3D.m*
     - Open a features file from either of the features folder
     - Script will plot the 5 activities in 3D feature space

4. Home Data (*HomeData.m*)
   - Uses **home_data** folder and opens the one month of home accelerations
   - Gives user the option to remove non-wearing data
   - Generates clips, calculates features, and uses lab sessions to predict labels on home data
   - Creates figures for activities profiles and confidence levels

5. Plot True-Positives and False-Positives (*TFPositives.m*)
   - Trains a model on lab data and tests on home labeled data in **home_labeled** folder
   - Plots histograms for true-positives and false-positives for each class