CSE 572 - Data mining

Group 11
Spring 2018

Task 1. Raw sensor data has 17 streams for each hand (Left and Right) i.e total 34 data streams in addition to Kinect data. Kinect data will be ignored for further analysis. Upon observation, we notice that of all the 10 gestures, only **CAN** and **DECIDE** use left hand. Hence, it will be safe to assume that classification can be done by analyzing sensor data streams only from right hand sensors.
We have taken data from the following groups = DM11, DM26, DM27, DM28, DM30, DM32, DM34, DM35, DM36, DM37 The following data types are selected from the raw data

1. Accelerometer (Right hand X,Y,Z axes) = **3**
2. Gyroscope (Right hand X,Y,Z axes) = **3**
3. Orientation (Right hand - Roll, Pitch, Yaw) = **3**
4. EMG (Right hand) = **8**

`procecssData.m` prompts user to select directory where all data files are kept. It outputs 10 csv files, one for each action in `processed_data` directory. All other scripts must select this directory when prompted for path. In each ACTION csv, rows represent invidual sensor type. Multiple repetitions of the same action are row appended in their respective files. This means each file will have total rows = 17xNumber of repetitions for Action. Sensor data values at each instant depending upon the sensor frequency are saved in columns.

Task 2. We need to choose 5 feature extraction methods for this task. They are as follows

1. **Number of zero crossings**
   (a) Intuition
   Since the gestures are performed in air, hand acceleration and angles of wrist must change directions during the course of the action. It is possible that each gesture have their unique counts for how many times this direction changes hinting their spatial trajectories.
   (b) Feature extraction
   The number of zero crossings are recorded for all axes of accelerometer and gyroscope i.e. 6 data streams. This gives a 6x1 vector for each instance of an action. For N instances of the same action, a mean vector is produced. The code can be found in `zero_crossings.m`
   $X_{action} = \frac{1}{N} \sum_{i=1}^{N} x_i \quad where \quad x = [n_{ax} n_{ay} n_{az} n_{gx} n_{gy} n_{gz}]$

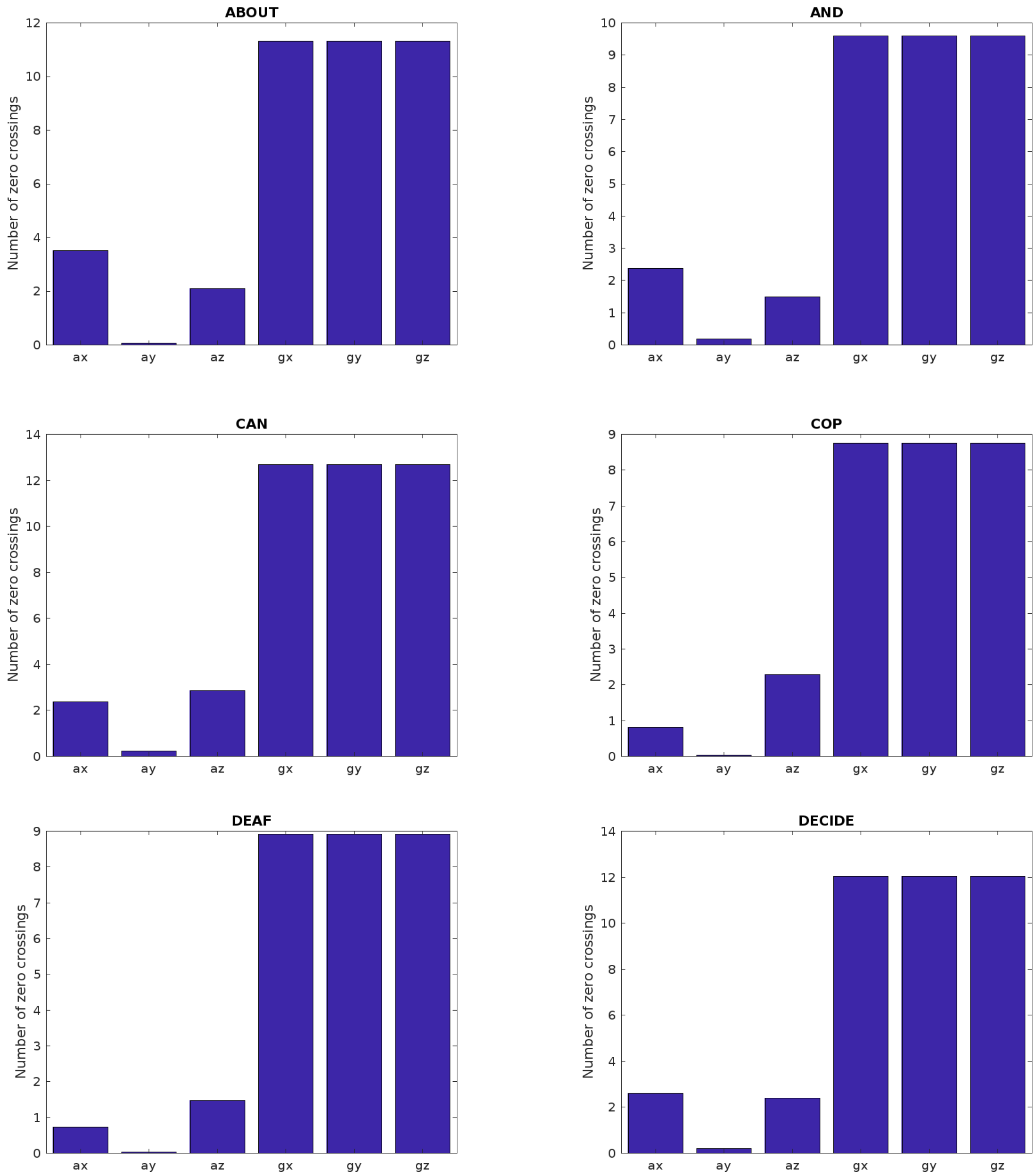

Figure 1: Zero crossings

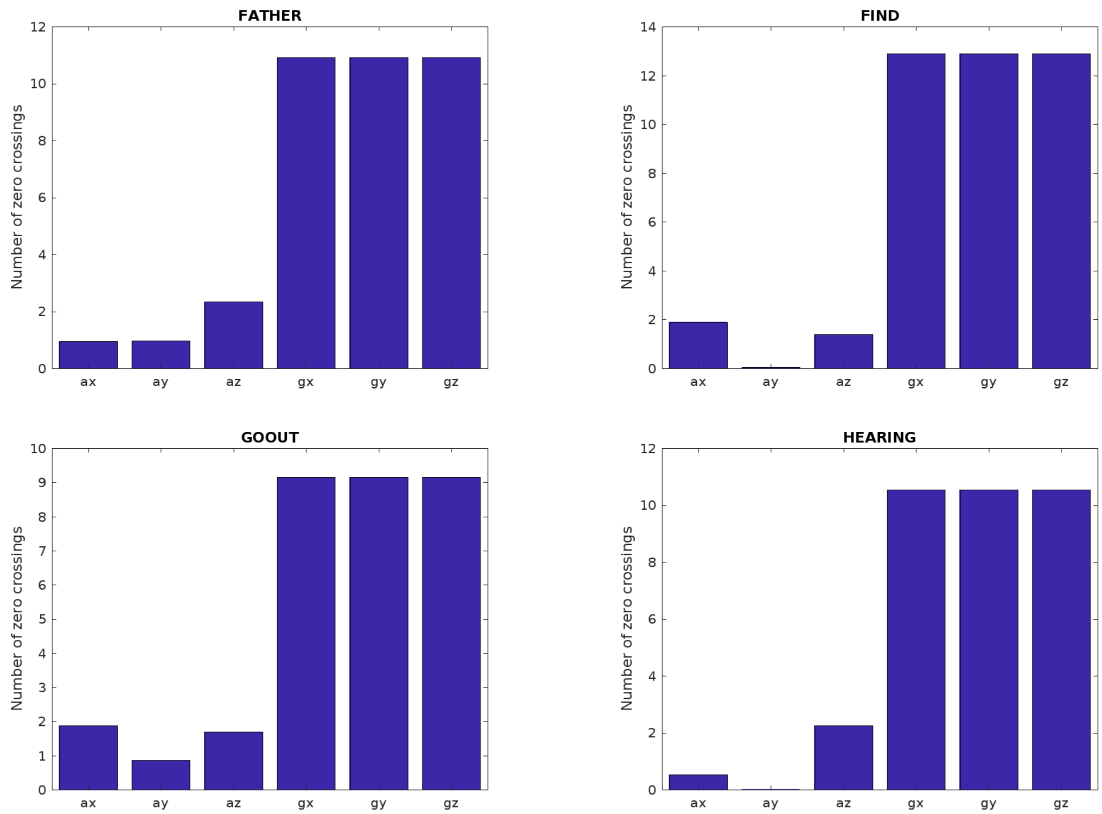

Figure 2: Zero crossings

(c) Plots

(d) Observations : Different gestures show different number of zero crossings for acceleration and gyroscope readings. Although the variance between different gestures is not too large for either acceleration or angle.

2. Temporal location of max value

   (a) Intuition : It is assumed that each gesture must have its unique point in time when muscle activity/acceleration/angular velocity/pitch/yaw/roll etc peaks since max value event depends on the gestures pattern in space.

   (b) Feature extraction : The index of max value for each sensor type is recorded. For repeated actions, the values are averaged over the samples. The code can be found in `max_value.m`

   (c) Plots

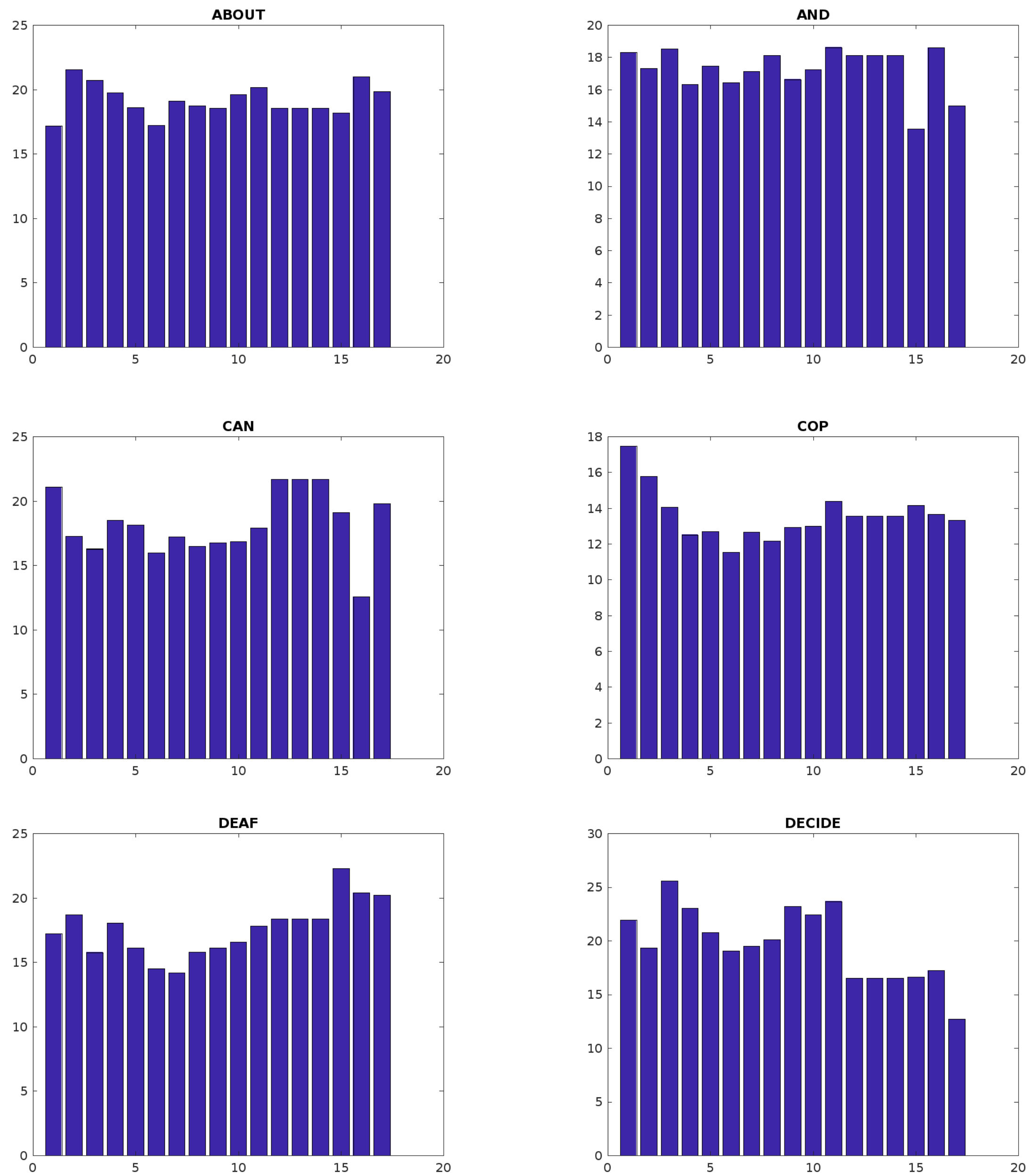

Figure 3: Max Indices for features[1:17]

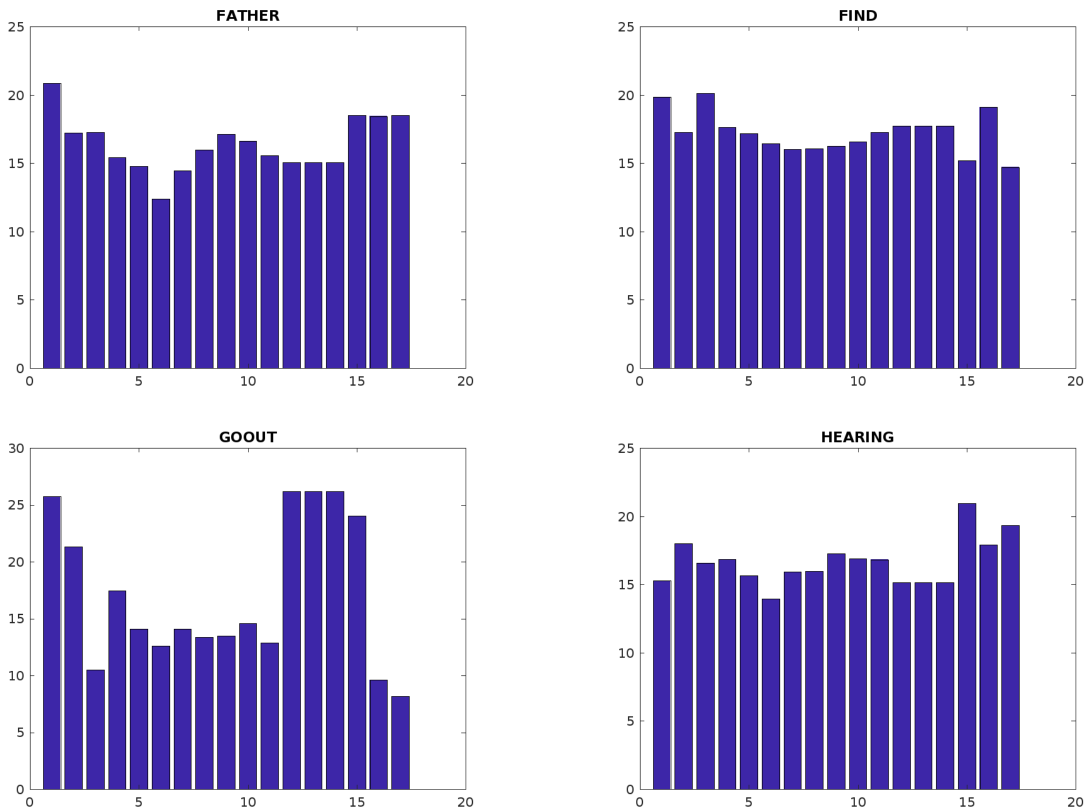

Figure 4: Max Indices for features[1:17]

(d) Observarions : As expected, peaks value shows significant variation among gestures. It is also worth noticing that gyroscope values for all 3 axes are same for all gestures. This could be due to gyroscope configured to measure values along single axis during the experiment.

3. Fast fourier transform

(a) Intuition : Fast fourier transform helps us to look at the frequency spectrum of the gestures. The EMG records electrical activity in muscles. During the course of performing gesture, muscles contract and relax depending upon the gesture. We extract this feature since it is likely that different gestures have different dominant frequencies which can be observed in the magnitude spectrum.

(b) Feature extraction
Since we have 8 channels in EMG, to reduce the dimension data, rows from 8 channels are added before performing FFT. This is reasonable since FFT performs decomposition. This means in the frequency domain, amplitude from each channel simply gets added for a given frequency. For each sample we have 44 data points. To increase the resolution of FFT, we zero pad the signal to 128 length. After that, half of the spectrum is discarded since it is a mirror of the other half for real signals, which is the case for us. This gives us 1x64 dimension vector for

each action. Code can be found in `feature_fft.m`

4. Plots

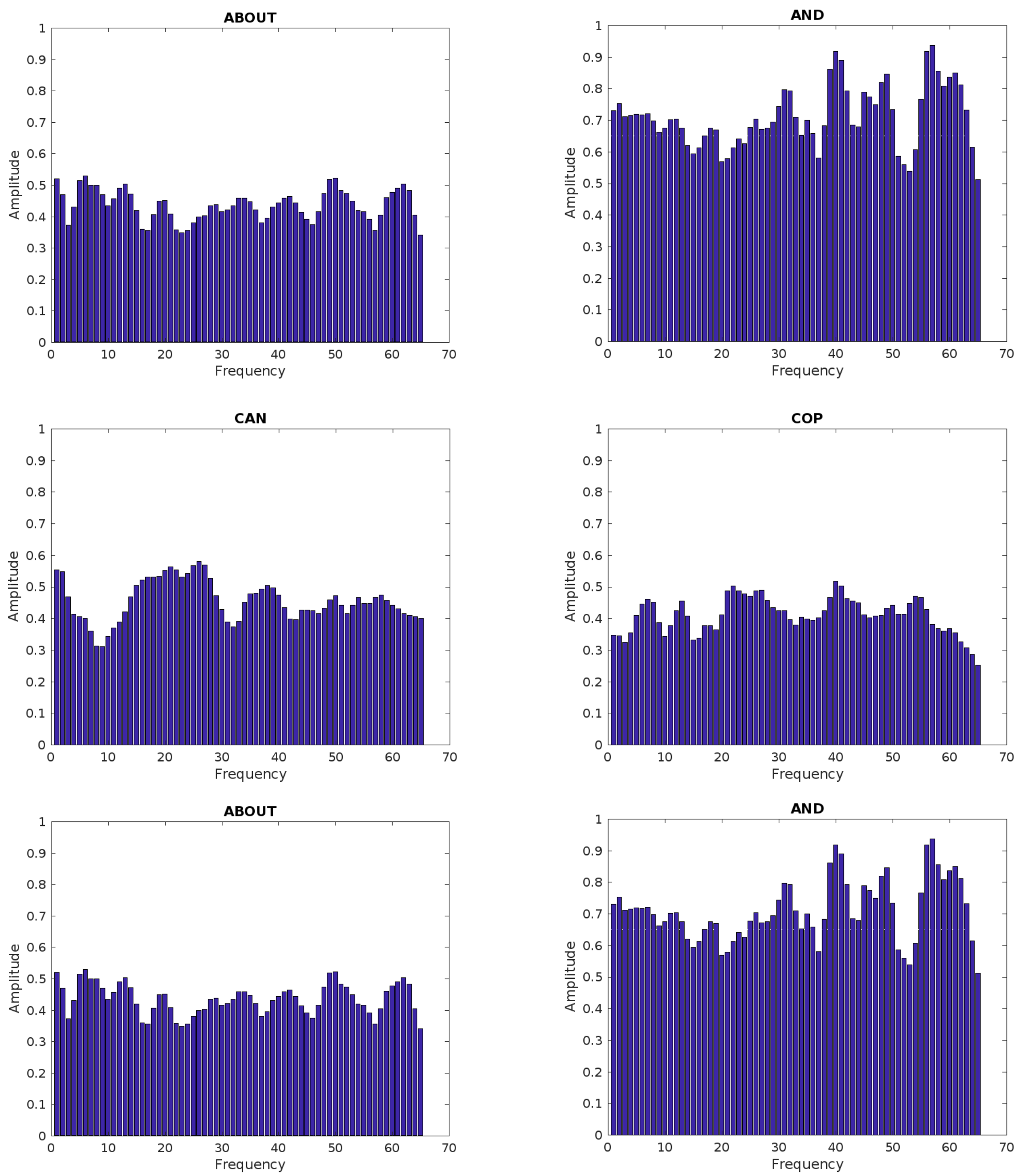

Figure 5: FFT for EMG signal (1x64)

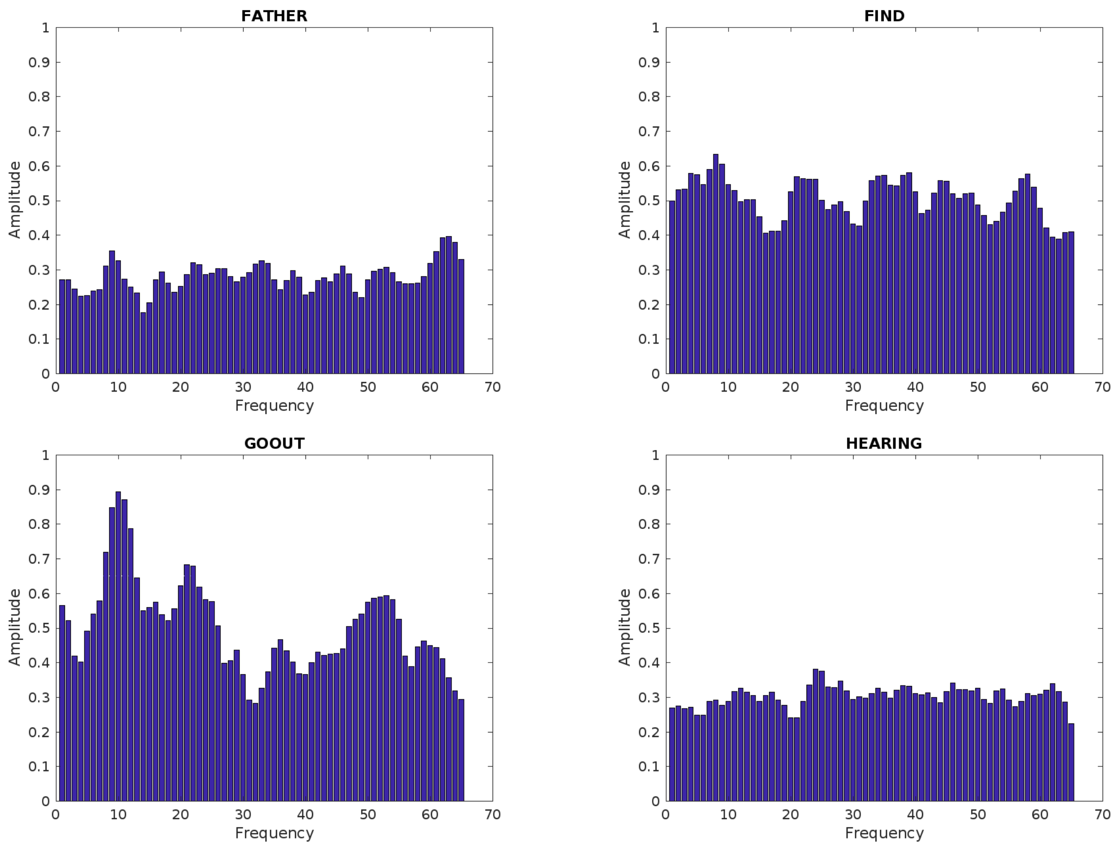

Figure 6: FFT for EMG signal (1x64)

5. Observation : Our intuition is largely correct. There is significant difference in maximum amplitude as we all the frequencies at which they occur.

6. Standard deviation

   (a) Intuition : This is a standard statistical feature for time series. Different sensors are likely to exhibit different variances for a given gesture.

   (b) Feature extraction : Standard deviation for all 17 sensor types are calculated and then averaged over multiple repetitions of the action. This gives us a 1x17 feature vector. The code can be found in `feature_std.m`

   (c) Plot :

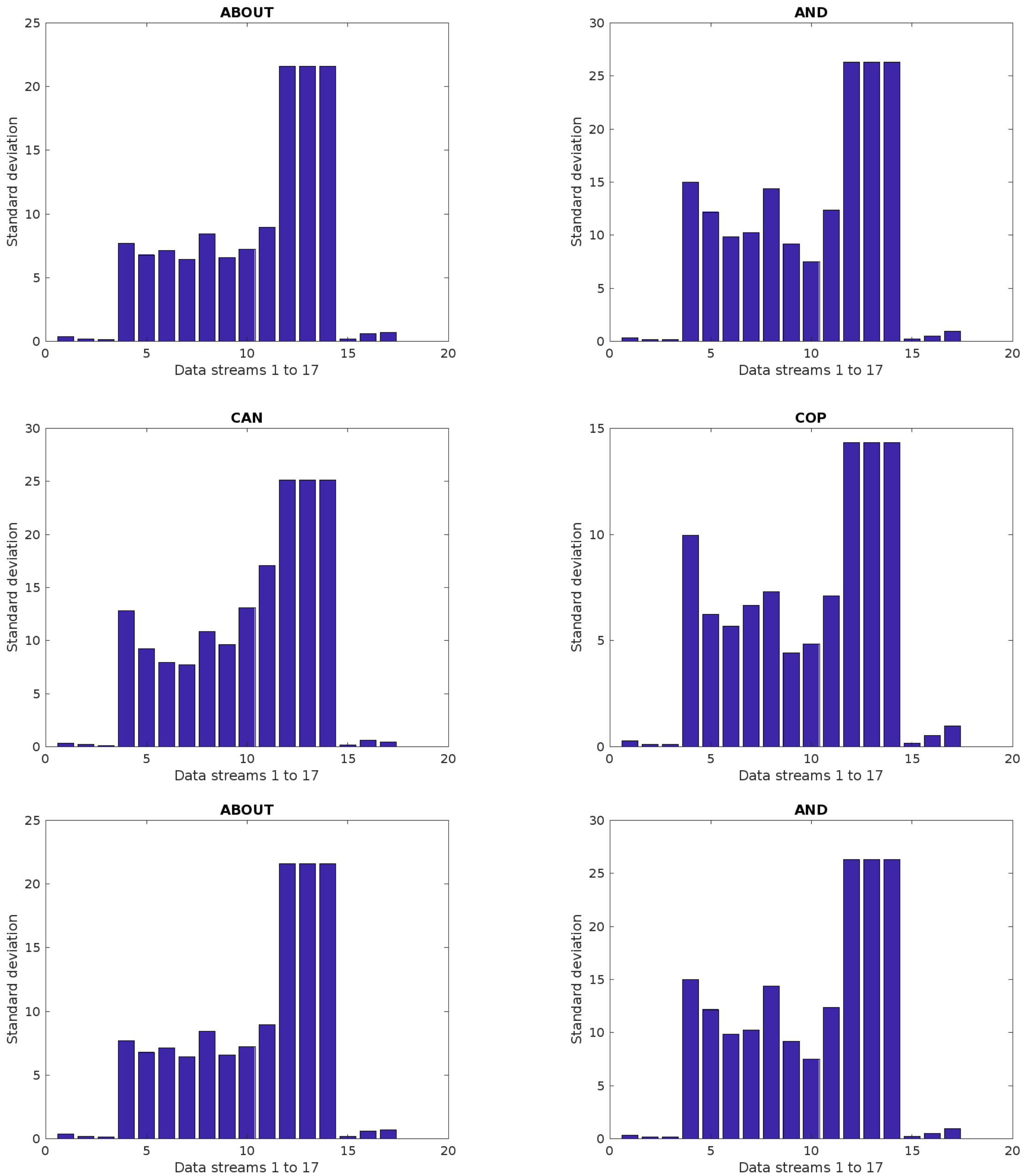

Figure 7: Standard deviation for sensors

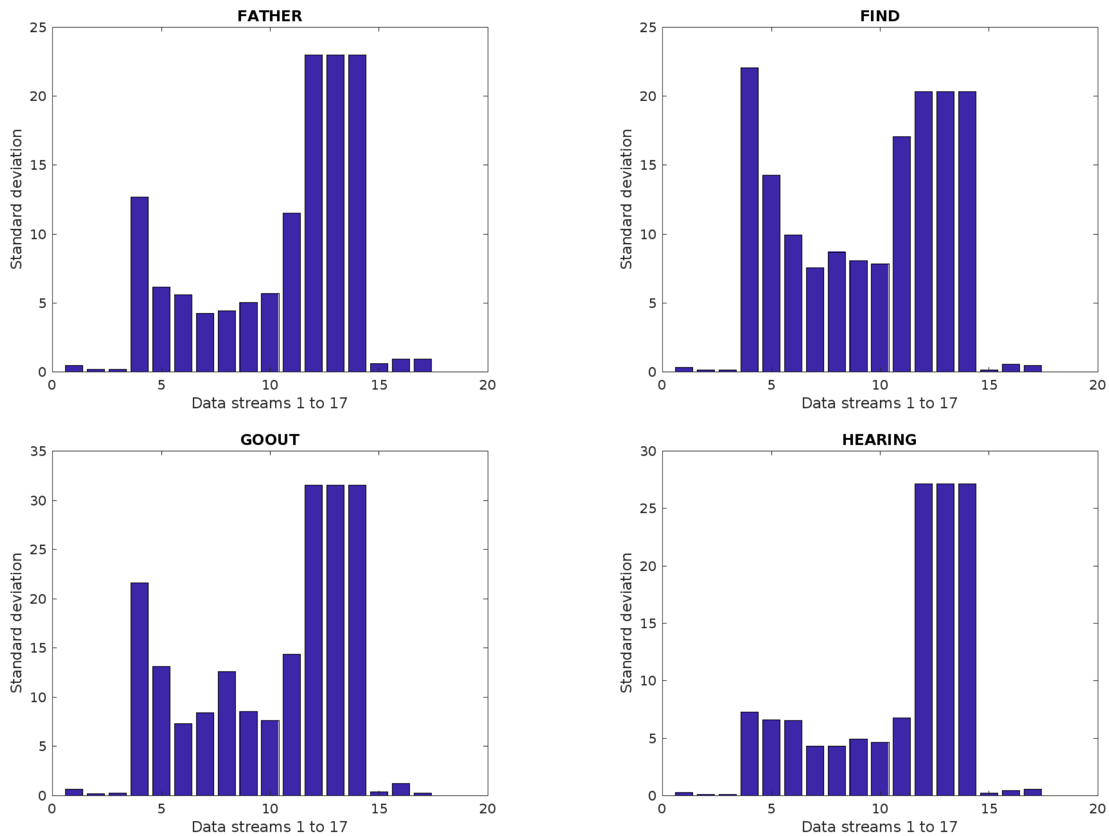

Figure 8: Standard deviation for sensors

(d) Observation : The standard deviation for acceleration and orientation (pitch,yaw,roll) is very small. Hence, only the remaining sensor types will be used to construct the feature vector of size (1x11)

7. Mean

   (a) Intuition : Intuition behind this feature is that different gestures are performed with different speed,acceleration,muscle activity,roll/pitch/yaw along all their corresponding axes. Different mean values are expected for each action.

   (b) Feature extraction : This feature results in (1x3) vector where means are calculated along a_x, a_y and a_z. Code can be found in `mean_all.m`

   (c) Plot :

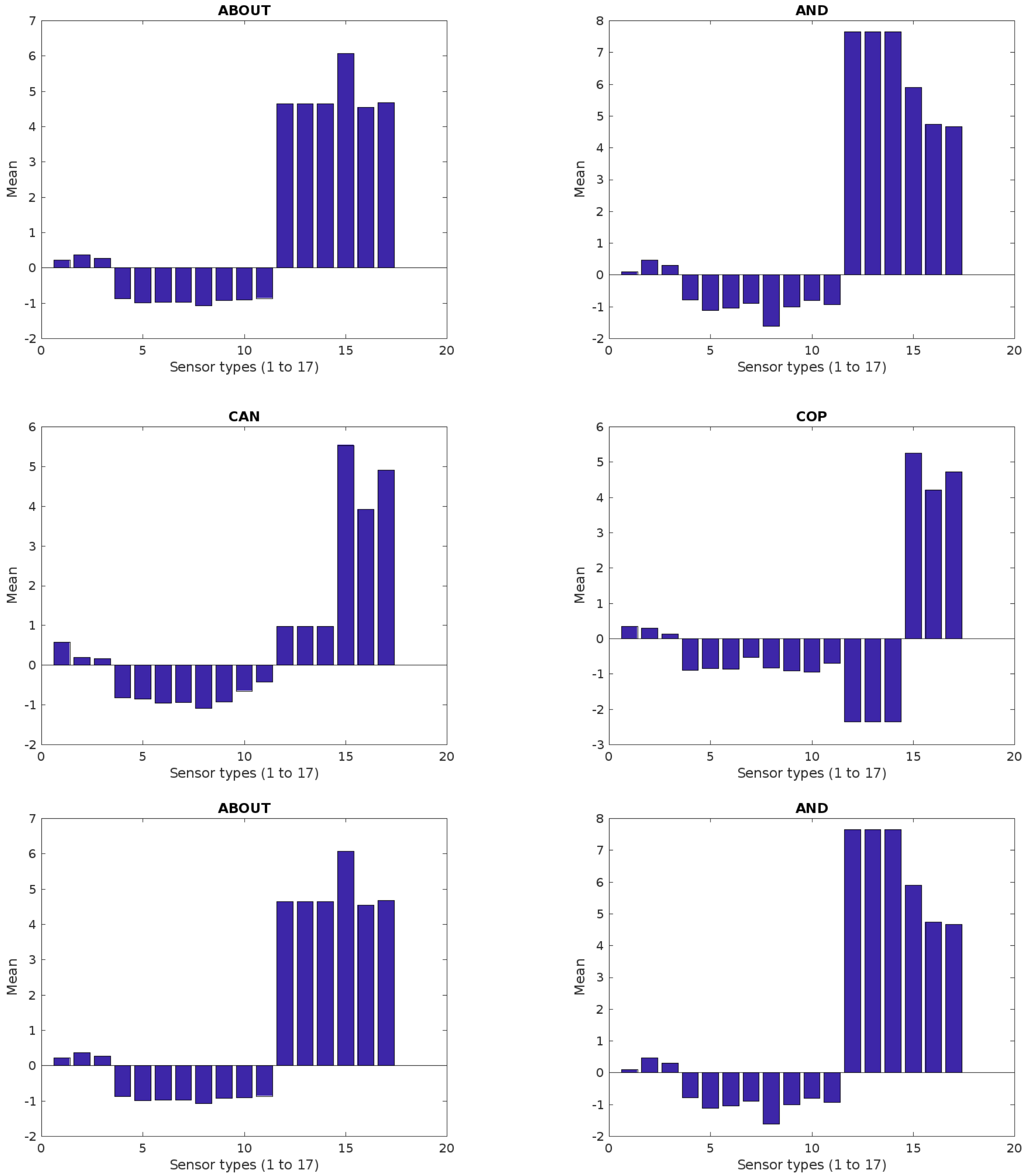

Figure 9: Feature means

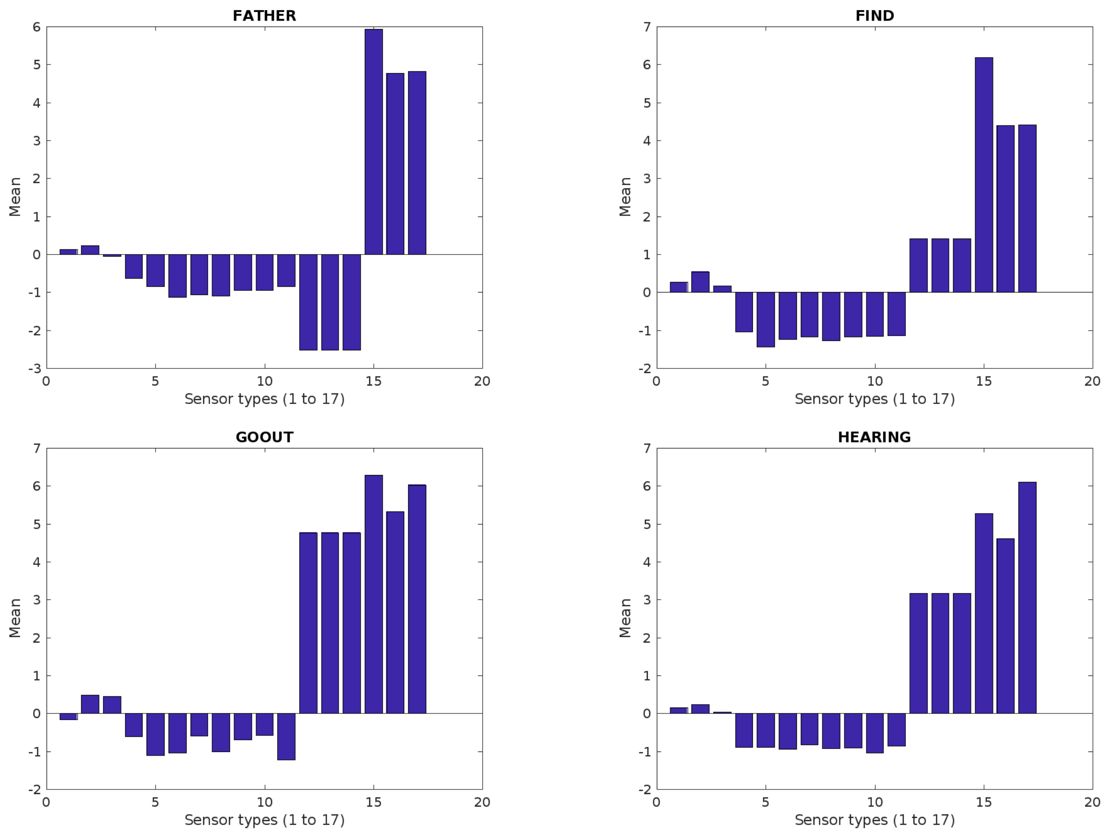

Figure 10: Feature means

8. Observation: While some sensors show variation in mean other show less deviation. The feature is extracted for all sensor types since it is assumed PCA will filter out only informative columns.

Task 3. Principal Component Analysis
The objective of this task is to reduce our feature space. We have 121 features which are indexed as following

1. Zero crossing [1:6]

2. Index of Max Value [7:23]

3. Fast fourier transform [24:87]

4. Standard deviation [88:104]

5. Mean [104:121]

We run MATLAB's `pca` function to perform Principal Component Analysis. `pca` is run individually for all our 10 actions. Each action has 20 instances (number of repetitions for the action). This gives us a 20x121 matrix as input to `pca`. Output is a coefficient matrix where columns are in decreasing order on principal component variances. The `biplot` function in

MATLAB helps us visualize which features out of the 121 display large variance. The steps to extract top features carrying most information is as follows. The code can be found in `pca_analysis.m`:

1. for *action* in LIST_OF_ACTIONS
2. Create *feature_matrix* 20x121 matrix by calculating all features
3. Perform PCA, coeff = pca(feature_matrix)
4. Get Top 3 feautures
5. Take union of all 3 top features collected from all LIST_OF_ACTIONS

This gives us a vector of **17 features**

$$0, 7, 9, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 95, 99, 100, 101$$

Plot :

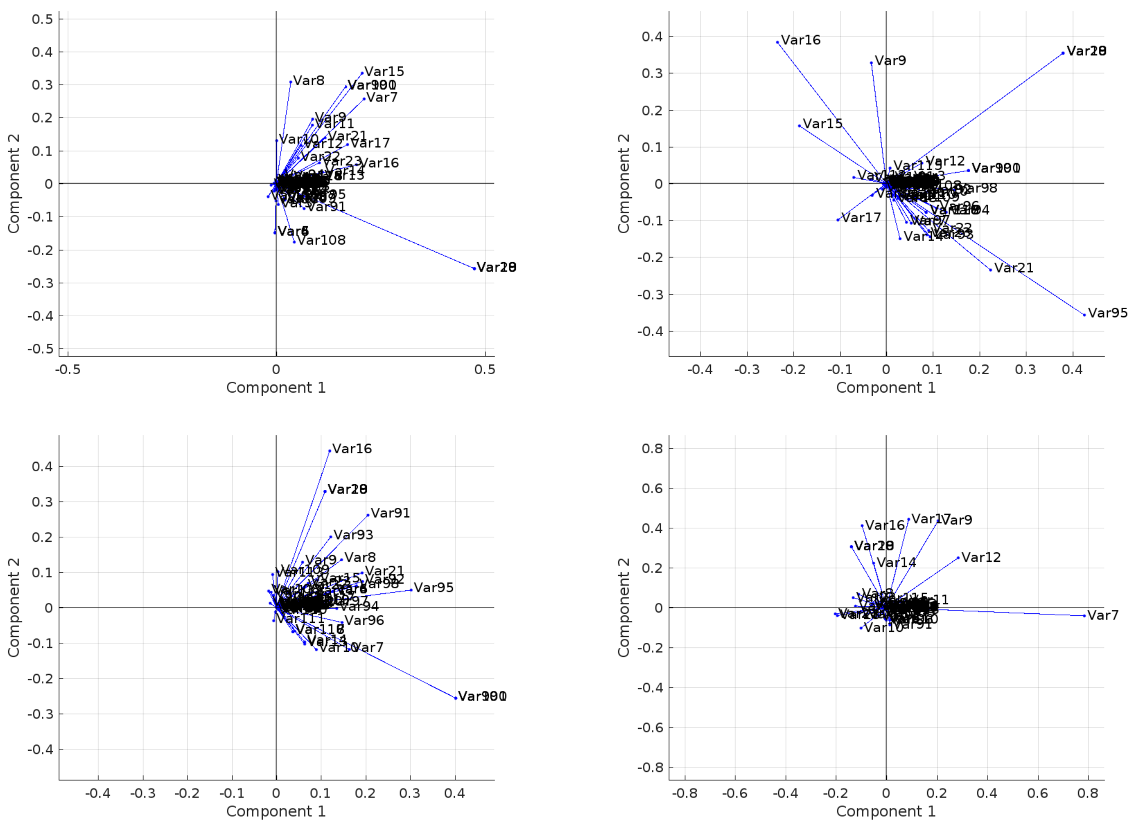

Figure 11: PCA biplot

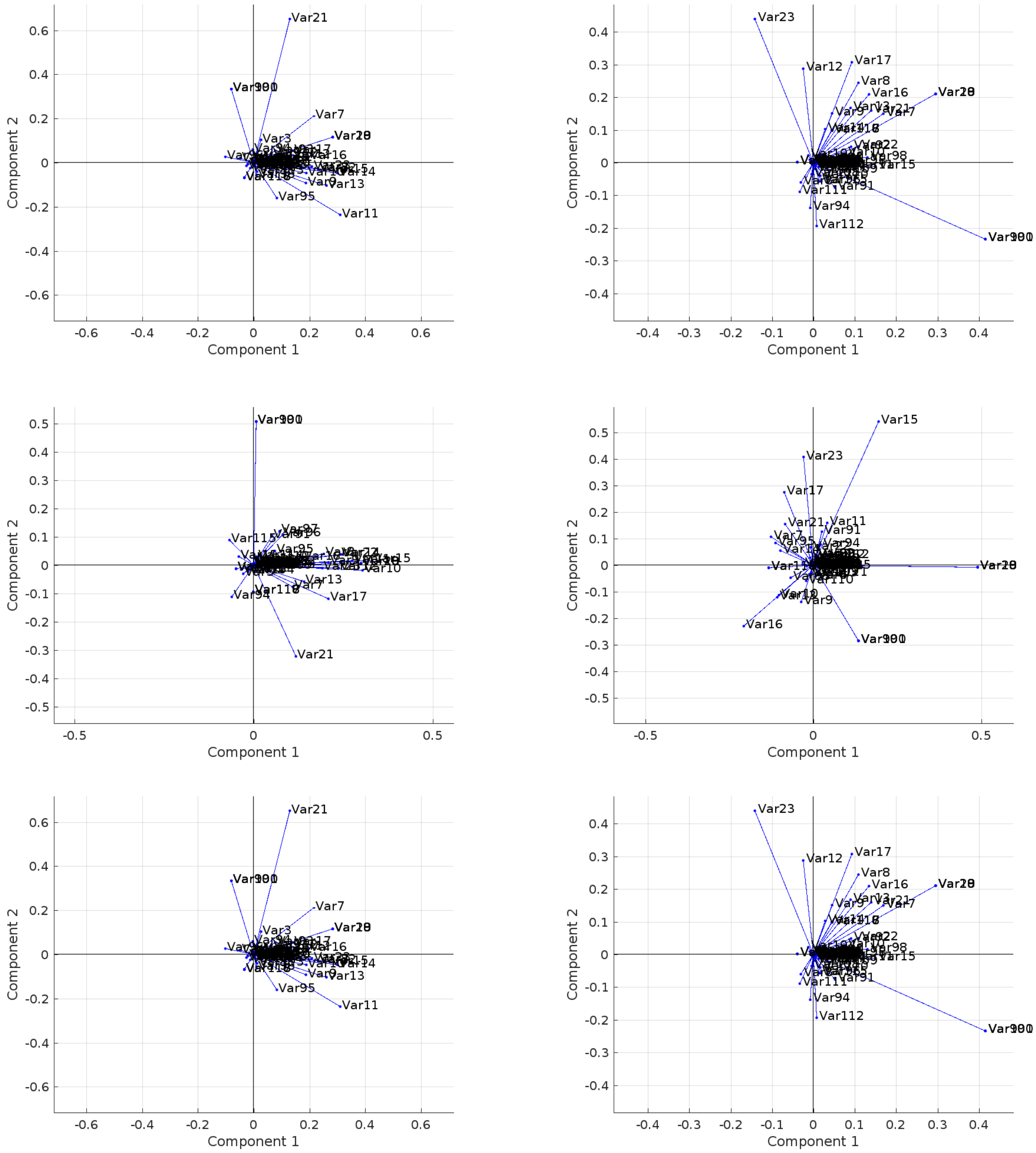

Figure 12: PCA biplot