CSE572 – Data Mining

Submission data: February 12, 2020
Roey Ben Hayun

Activity Recognition

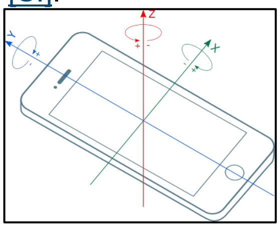
Background

Accelerometer

[1]Accelerometers are devices that measure acceleration, which is the rate of change of the velocity of an object. A raw accelerometer sensor measures changes in acceleration in 3 different directions, but is affected by gravity. The Accelerometer sensor is an inertial-frame sensor, this means that when the device is in free fall, the acceleration is 0 m/s2 in the falling direction, and when a device is laying flat on a table, the acceleration in upwards direction will be equal to the Earth gravity, i.e. g ≡ 9.8 m/s2 as it is measuring the force of the table pushing the device upwards.
https://www.sciencebuddies.org/science-fair-projects/references/accelerometer

Gyroscope

A gyroscope senses *angular velocity*, relative to itself, thus it measures its own rotation, using an inertial force called the Coriolis effect. The angular velocity is the rate at which the device rotates about a specified axis in a local coordinate system defined by the device. Its unit is the radian per second (rad/s) [SI].



Orientation

The orientation sensor is a fusion of the Accelerometer and Gyroscope and/or Magnometer, depends on the orientation type being measured (i.e relative orientation sensor, absolute orientation sensor, etc.). In our case, the orientation part of the IMU data is composed of four values: x,y,z, and w. Those are the representation of quaternion which is another alternate representation of device orientation.

Phase One: Data Cleaning and Organization

For this project, I have created four cleass member lists of 2d Numpy arrays:
self.__imu_fork_eating = []
self.__imu_fork_non_eating = []
self.__imu_spoon_eating = []
self.__imu_spoon_non_eating = []

Each list has entries as the number of users (30 in our case). Each column represents the IMU sensor values (10).
So, for the eating activities (fork, spoon), the number of rows is derived from the ground truth data and extracted from the raw sensor data using the following formula:
start_row = start_frame * 50 / 30
end_row = end_frame * 50 / 30

Where:
start_frame and end_frame are taken from the ground truth data. start_row and end_row will be the range of IMU data that will be extracted from the raw sensor data. The non-eating IMU data is what was left after the extraction of the eating activities .

Phase Two: Feature Extraction

Eating and none activities recognition/detection from IMU data are preceded by a feature extraction step. The assumption is that the IMU data collected was pre-processed (windowing, filtering, etc,) and is ready for the feature extraction phase.
Signal properties such as time-domain and frequency-domain are widely used for feature extraction. Time-domain features include mean, median, variance, max, range, etc. Frequency domain features include peak frequency, peak power, spectral power and more.

Mean

The mean and standard deviation describe the average value and the amplitude change of accelerations and angular velocity (gyroscope) in an eating period. The hypothesis is that the signals on each axis can better reflect eating activities and that we should a difference between eating and non-eating activities.

$\underline{Orientation}$:

$$Mean_{orientation} = [\mu_{Ox}, \mu_{Oy}, \mu_{Oz}, \mu_{Ow}]\ Where:$$

$$\mu_{ox} = \sum_{i=1}^{N} \frac{O_{x_i}}{N}, \quad \mu_{oy} = \sum_{i=1}^{N} \frac{O_{y_i}}{N}, \quad \mu_{oz} = \sum_{i=1}^{N} \frac{O_{z_i}}{N}, \mu_{ow} = \sum_{i=1}^{N} \frac{O_{w_i}}{N}$$

$\underline{Accelerometer}$:

$$Mean_{acc} = [\mu_{ax}, \mu_{ay}, \mu_{az}]\ Where:$$

$$\mu_{ax} = \sum_{i=1}^{N} \frac{A_{x_i}}{N}, \quad \mu_{ay} = \sum_{i=1}^{N} \frac{A_{y_i}}{N}, \quad \mu_{az} = \sum_{i=1}^{N} \frac{A_{z_i}}{N}$$

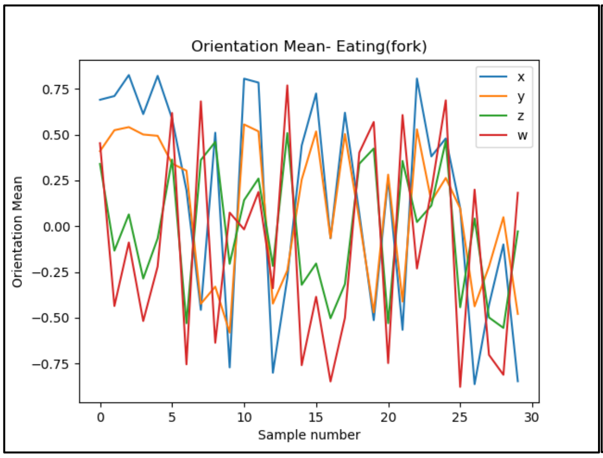
$\underline{Gyro}$:

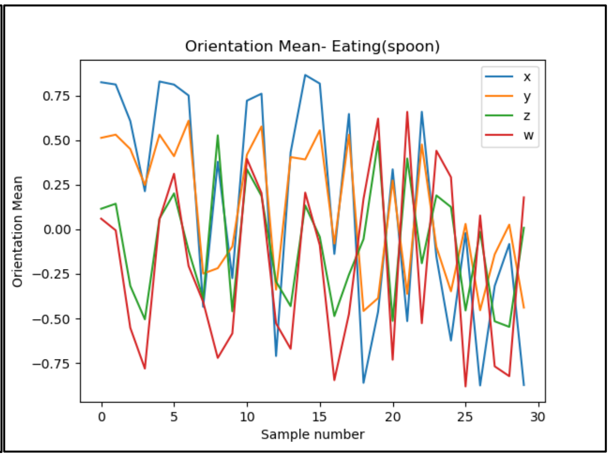
$$Mean_{gyro} = [\mu_{gx}, \mu_{gy}, \mu_{gz}]\ Where:$$

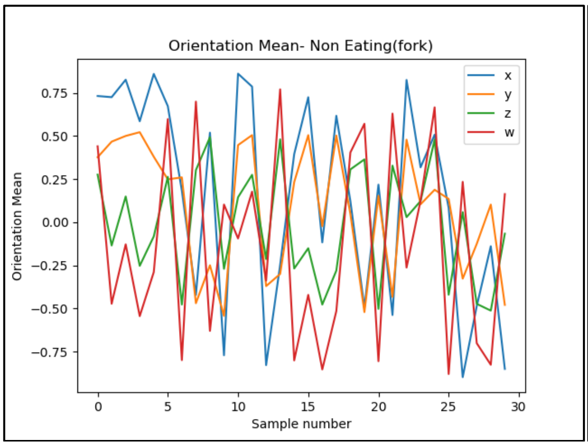
$$\mu_{ax} = \sum_{i=1}^{N} \frac{G_{x_i}}{N}, \quad \mu_{ay} = \sum_{i=1}^{N} \frac{G_{y_i}}{N}, \quad \mu_{az} = \sum_{i=1}^{N} \frac{G_{z_i}}{N}$$

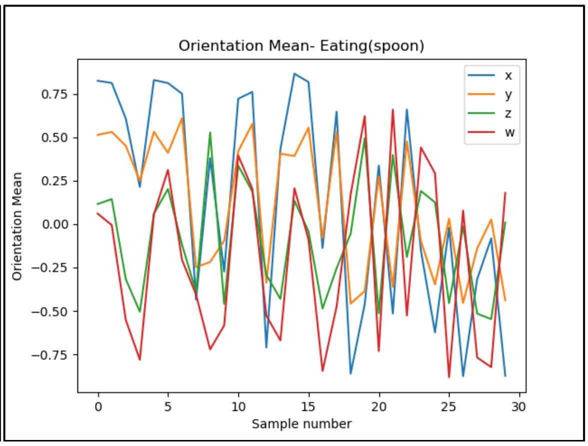
Plots

Orientation Mean Eating vs. Non-Eating

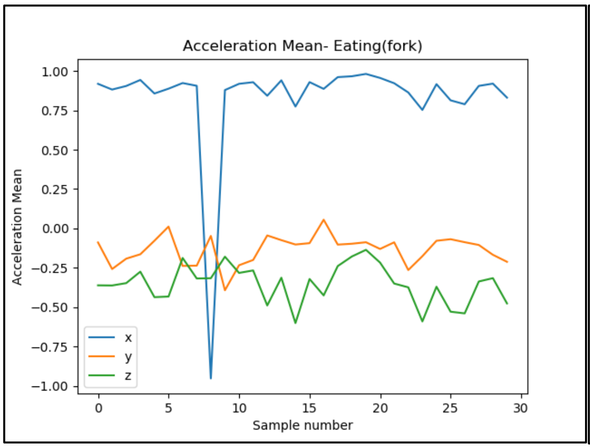


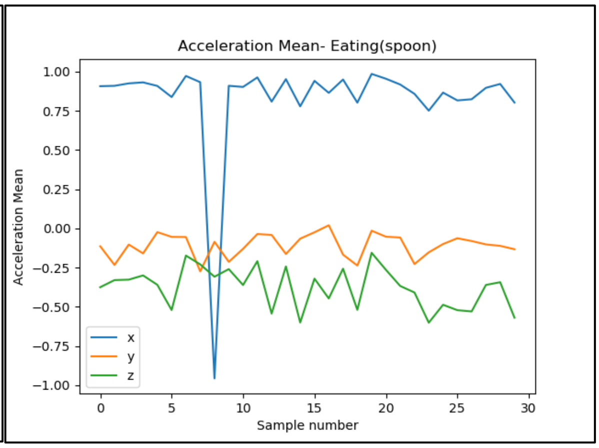


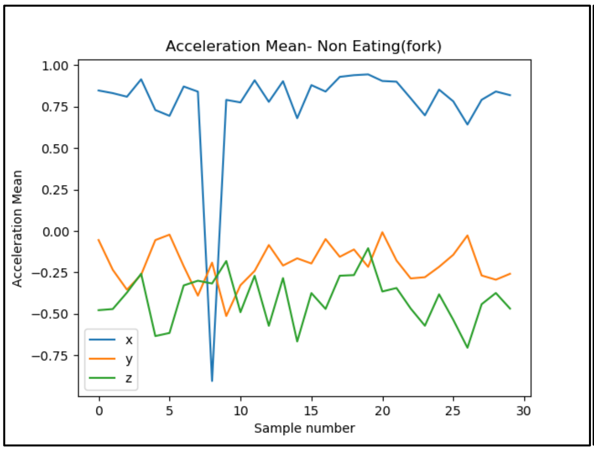


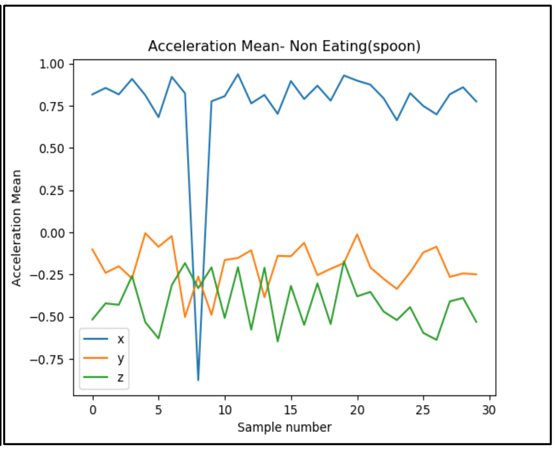


Acceleration Mean Eating vs. Non-Eating

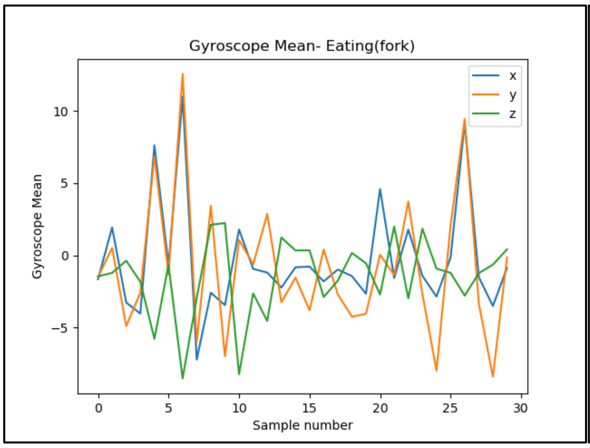


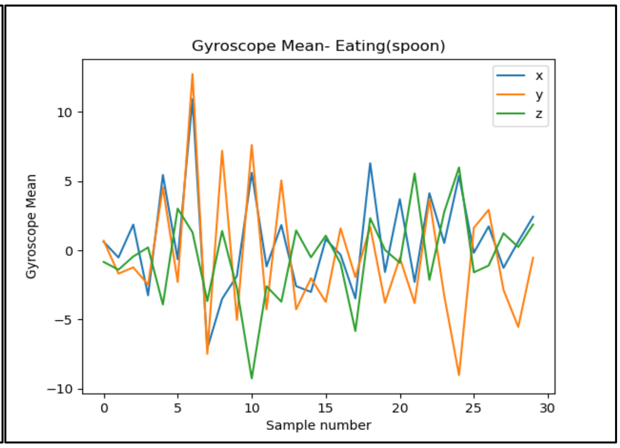


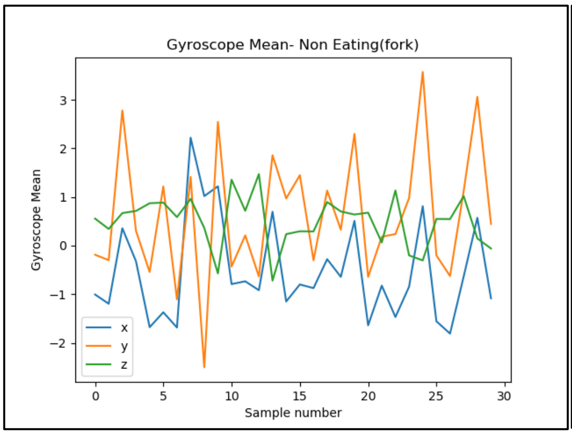


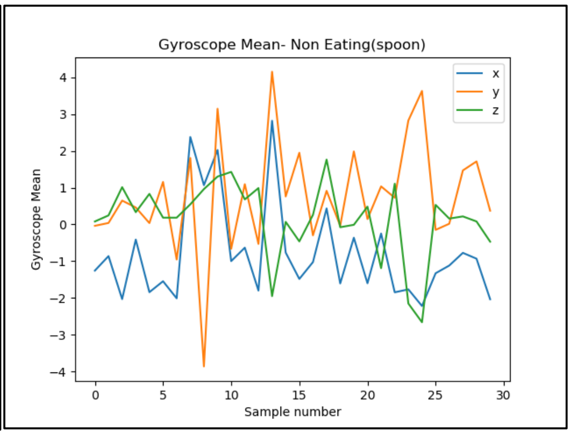


Gyroscope Mean Eating vs. Non-Eating









In this case, only minor differences are noticeable. Not as I had expected.

Variance

As explained before for the mean, the standard deviation describe the average value and the amplitude change of accelerations and angular velocity (gyroscope) in an eating period. I expect to observe differences between eating and non-eating activities.

$\underline{Orientation}$:

$$Variance_{orientation} = \left[\sigma^2{}_{ox}, \sigma^2{}_{oy}, \sigma^2{}_{oz}, \sigma^2{}_{ow}\right] \, Where:$$

$$\sigma^2{}_{ox} = \sum_{i=1}^{N} \frac{(\mu_{Ox} - O_{x_i})^2}{N}, \quad \sigma^2{}_{oy} = \sum_{i=1}^{N} \frac{(\mu_{Oy} - O_{y_i})^2}{N},$$
$$\sigma^2{}_{oz} = \sum_{i=1}^{N} \frac{(\mu_{Oz} - O_{z_i})^2}{N}, \quad \sigma^2{}_{ow} = \sum_{i=1}^{N} \frac{(\mu_{Ow} - O_{w_i})^2}{N}$$

$\underline{Accelerometer}$:

$$Variance_{acc} = \left[\sigma^2{}_{ax}, \sigma^2{}_{ay}, \sigma^2{}_{az}\right] \, Where:$$

$$\sigma^2{}_{ax} = \sum_{i=1}^{N} \frac{(\mu_{ax} - A_{x_i})^2}{N}, \quad \sigma^2{}_{ay} = \sum_{i=1}^{N} \frac{(\mu_{ay} - A_{y_i})^2}{N}, \sigma^2{}_{az} = \sum_{i=1}^{N} \frac{(\mu_{az} - A_{z_i})^2}{N}$$

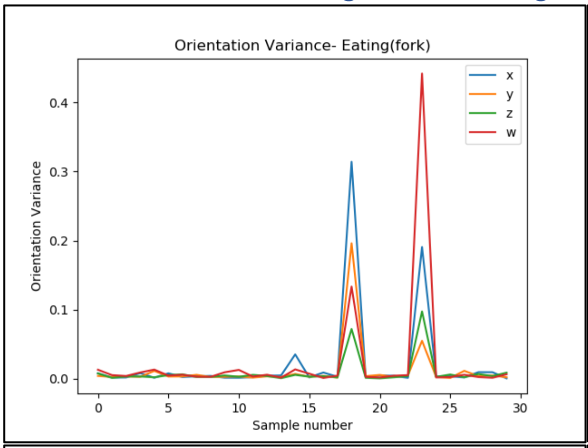
$\underline{Gyro}$:

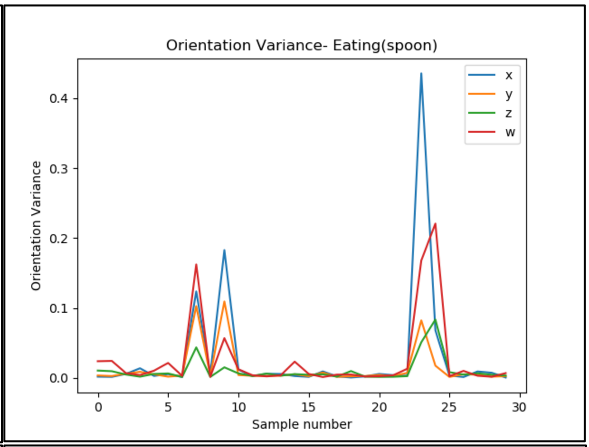
$$Variance_{gyro} = \left[\sigma^2{}_{gx}, \sigma^2{}_{gy}, \sigma^2{}_{gz}\right] \, Where:$$

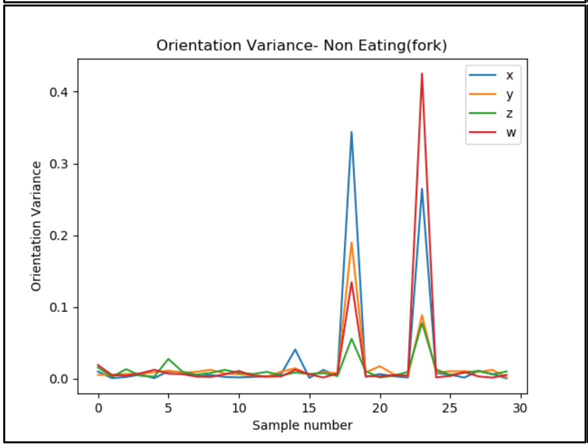
$$\sigma^2{}_{gx} = \sum_{i=1}^{N} \frac{(\mu_{gx} - G_{x_i})^2}{N}, \quad \sigma^2{}_{gy} = \sum_{i=1}^{N} \frac{(\mu_{gy} - G_{y_i})^2}{N}, \sigma^2{}_{gz} = \sum_{i=1}^{N} \frac{(\mu_{gz} - G_{z_i})^2}{N}$$

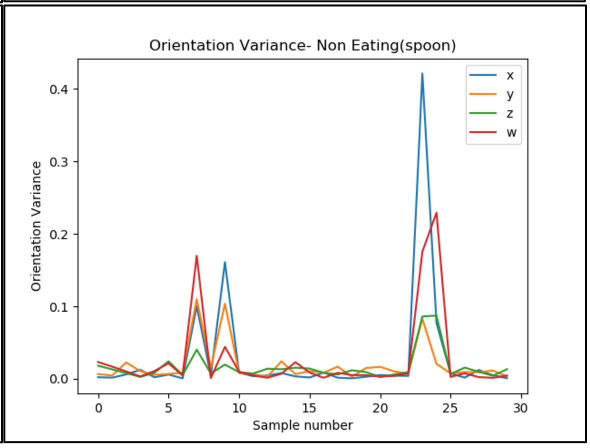
Plots

Orientation Variance Eating vs. Non-Eating

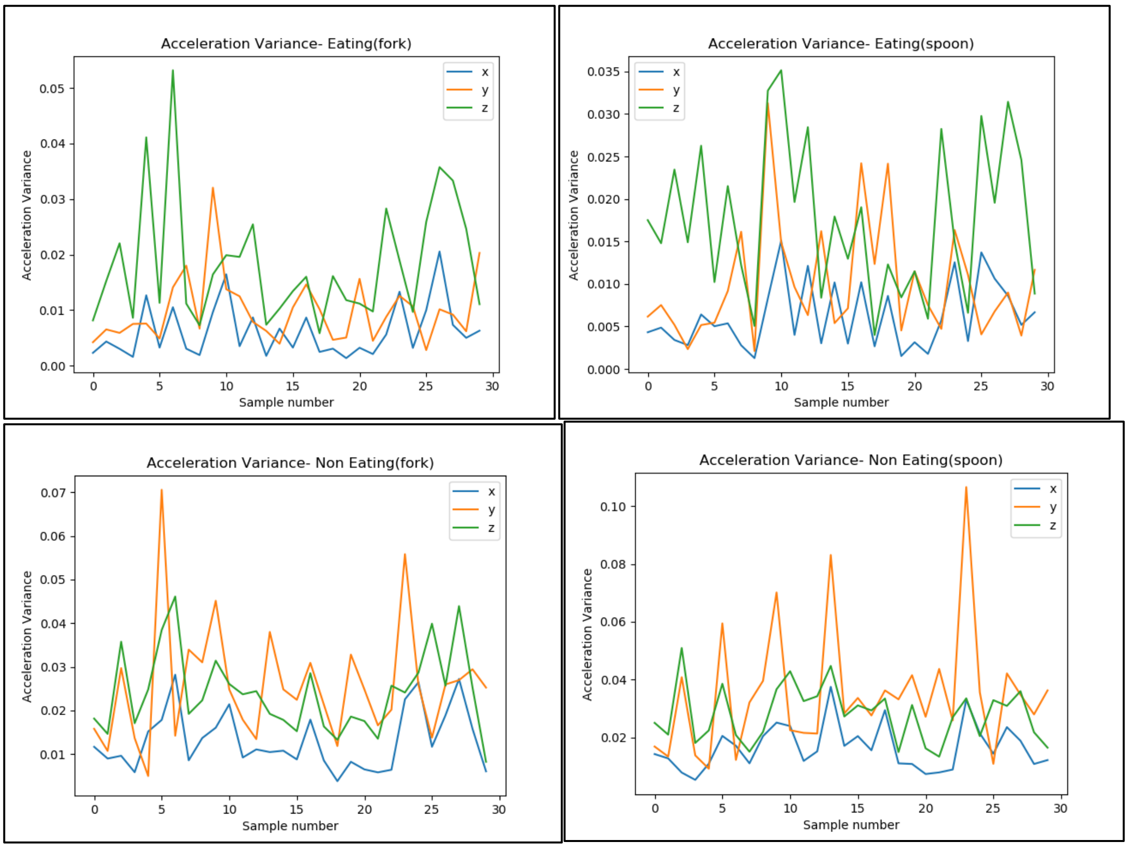






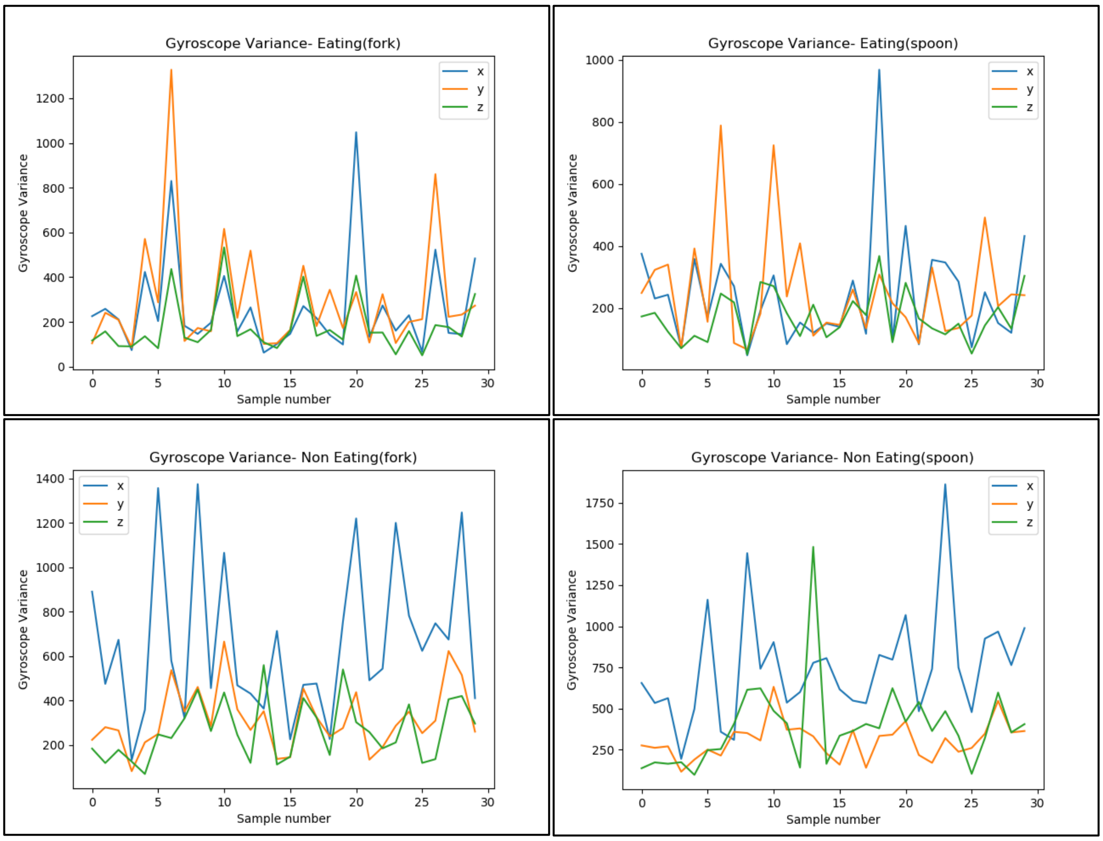


Acceleration Variance Eating vs. Non-Eating



We can see here a clear distinction between the variance of the acceleration in the y and z axis between eating and non-eating activities (both for spoon and fork). It looks like the minor changes that we observed with the mean feature are now more clear and noticeable with the variance feature. Since we do not have to much information about the orientation of the phone it hard to tell what is the meaning in the real world (experiment) of the x,y and z here.

Gyroscope Variance Eating vs. Non-Eating



The same here. You can see a clear distention between eating and non-eating activities (both for spoon and fork). The approving our intuition.

RMS

The RMS is a fundamental measurement of the magnitude/energy of a signal. My intuition is that we should observe a change in the energy measurements of the IMU sensors correlated with activity (eating and non-eating)

$\underline{Orientation}$:

$$RMS_{orientation} = [rms_{ox}, rms_{oy}, rms_{oz}, rms_{ow}]\ Where:$$

$$rms_{ox} = \sqrt{\sum_{i=1}^{N} \frac{{O_{x_i}}^2}{N}},\quad rms_{oy} = \sqrt{\sum_{i=1}^{N} \frac{{O_{y_i}}^2}{N}}, rms_{oz} = \sqrt{\sum_{i=1}^{N} \frac{{O_{z_i}}^2}{N}},\ rms_{ow} = \sqrt{\sum_{i=1}^{N} \frac{{O_w}^2}{N}}$$

$\underline{Accelerometer}$:

$$RMS_{acc} = [rms_{ax}, rms_{ay}, rms_{az}]\ Where:$$

$$rms_{ax} = \sqrt{\sum_{i=1}^{N} \frac{{A_{x_i}}^2}{N}},\quad rms_{ay} = \sqrt{\sum_{i=1}^{N} \frac{{A_{y_i}}^2}{N}}, rms_{az} = \sqrt{\sum_{i=1}^{N} \frac{{A_{z_i}}^2}{N}}$$

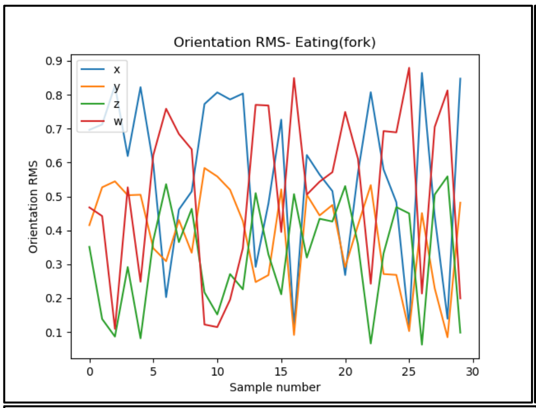
$\underline{Gyro}$:

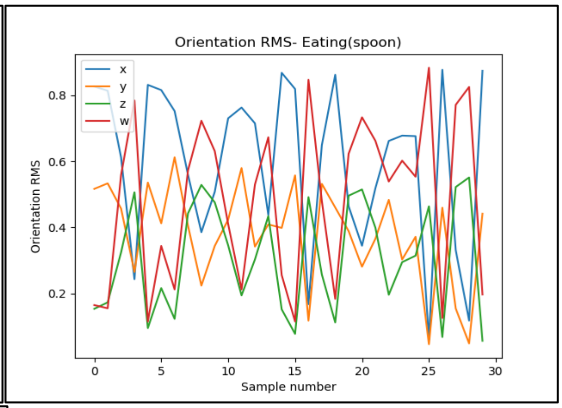
$$RMS_{gyro} = [rms_{gx}, rms_{gy}, rms_{gz}]\ Where:$$

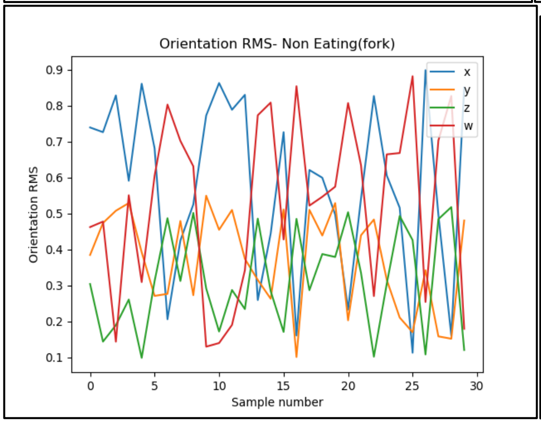
$$rms_{gx} = \sqrt{\sum_{i=1}^{N} \frac{{G_{x_i}}^2}{N}},\quad rms_{gy} = \sqrt{\sum_{i=1}^{N} \frac{{G_{y_i}}^2}{N}}, rms_{gz} = \sqrt{\sum_{i=1}^{N} \frac{{G_{z_i}}^2}{N}}$$

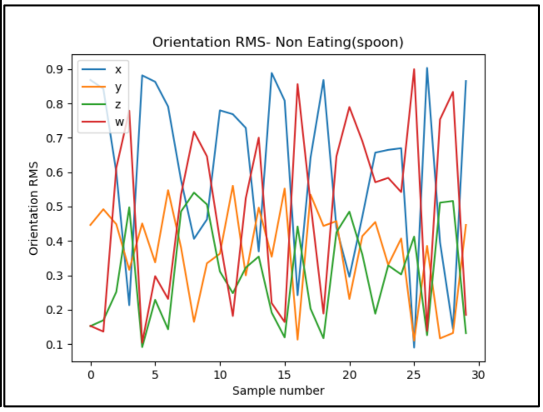
Plots

Orientation RMS Eating vs. Non-Eating



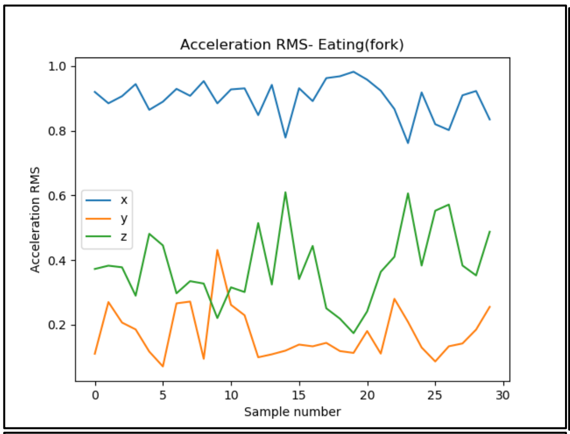


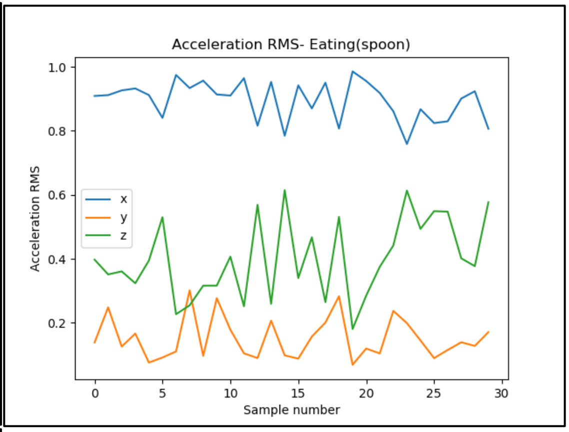


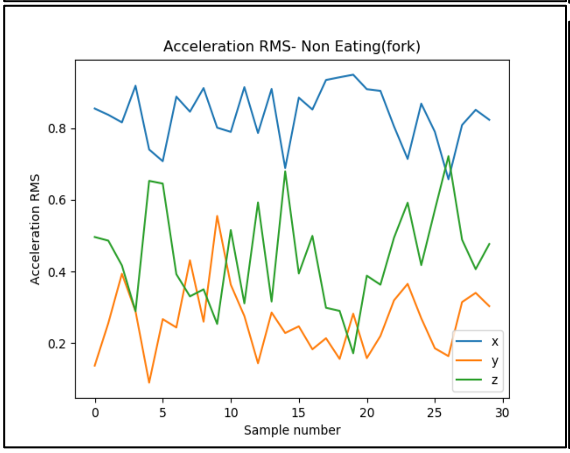


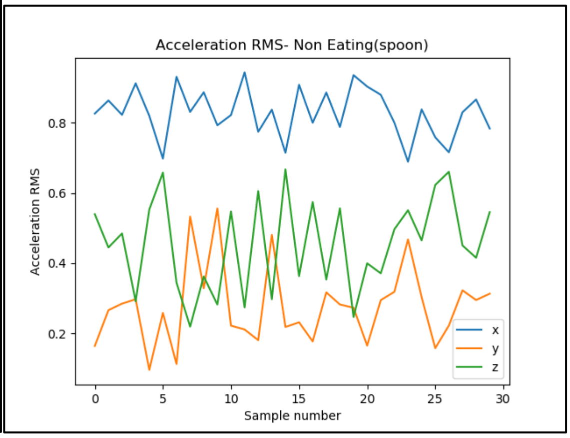
In the case of the RMS of the orientation sensors, my intuition was in correct. By just looking on the plots there is no way to make a distinction between eating and non-eating activities.

Acceleration RMS Eating vs. Non-Eating

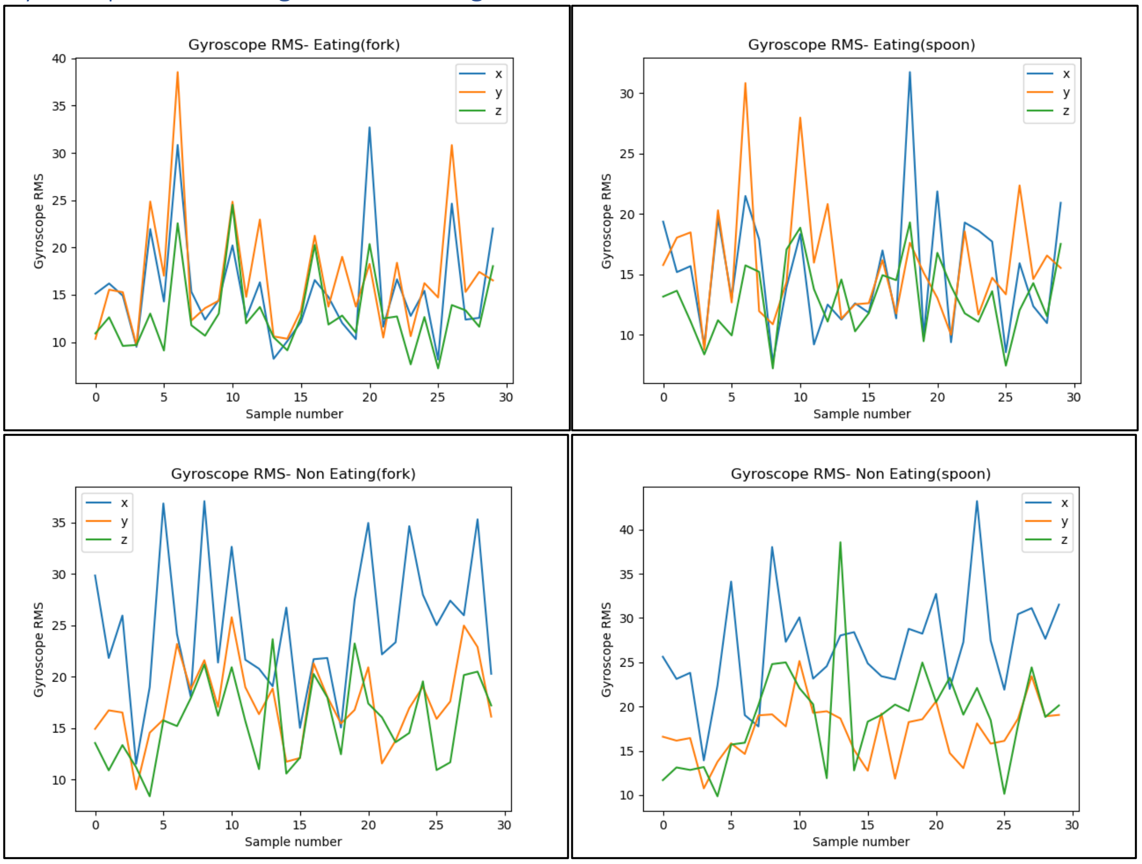








Gyroscope RMS Eating vs. Non-Eating



For the acceleration and gyroscope we can see that my intuition was correct. Again, like in the case of the variance feature, we can see a clean distinction between the RMS of the gyroscope /acceleration in eating and non-eating activities. There is an energy change which is caused by the movement of the hand (and that's also make sense).

Min

Find the minimum of IMU sensors. My intuition is that I will not be able to see any clear(major) distinction between the activities. Moreover, because some information about the position and the interpretation of the values is missing it is difficult to understand the mining of minimum by intuition. Nevertheless, some (minor) distinction would not surprise me.

$\underline{Orientation}$:

$$MIN_{orientation} = [min_{ox}, min_{oy}, min_{oz}, min_{ow}]\ Where:$$

$$min_{ox} = \min\{O_{x_1} \dots O_{x_i}\},\quad min_{oy} = \min\{O_{y_1} \dots O_{y_i}\},$$
$$min_{oz} = \min\{O_{z_1} \dots O_{x_i}\},\ min_{ow} = \min\{O_{w_1} \dots O_{w_i}\}$$

$\underline{Accelerometer}$:

$$MIN_{acc} = [min_{ax}, min_{ay}, min_{az}]\ Where:$$

$$min_{ax} = \min\{A_{x_1} \dots A_{x_i}\},\quad min_{ay} = \min\{A_{y_1} \dots A_{y_i}\}, min_{az} = \min\{A_{z_1} \dots A_{x_i}\}$$

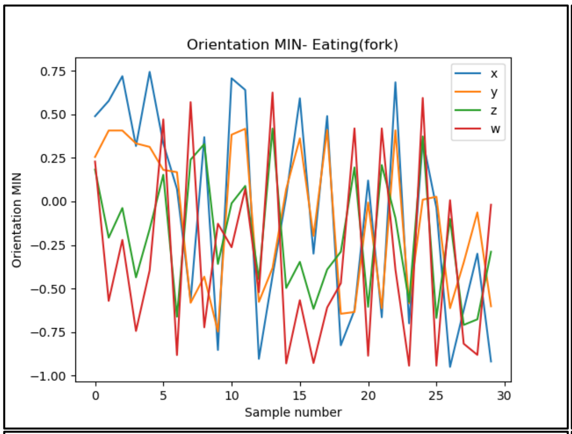
$\underline{Gyro}$:

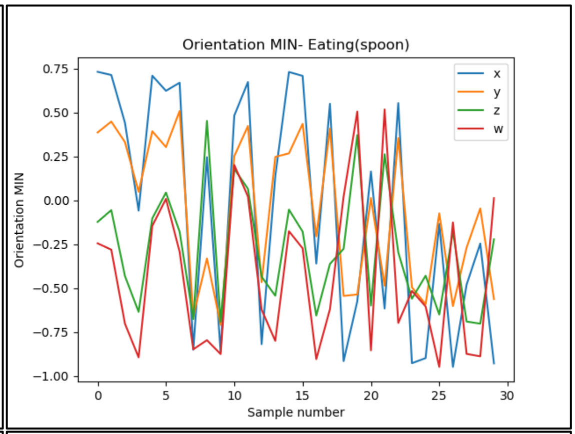
$$MIN_{gyro} = [min_{gx}, min_{gy}, min_{gz}]\ Where:$$

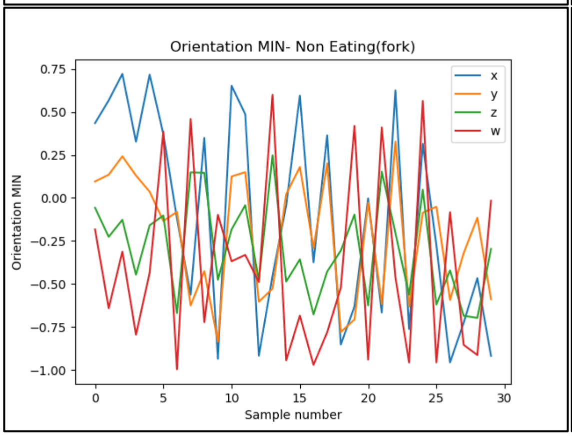
$$min_{gx} = \min\{G_{x_1} \dots G_{x_i}\},\quad min_{gy} = \min\{G_{y_1} \dots G_{y_i}\}, min_{gz} = \min\{G_{z_1} \dots G_{x_i}\}$$

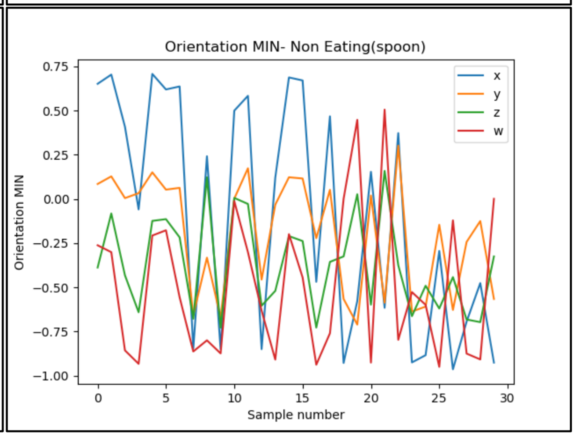
Plots

Orientation MIN Eating vs. Non-Eating



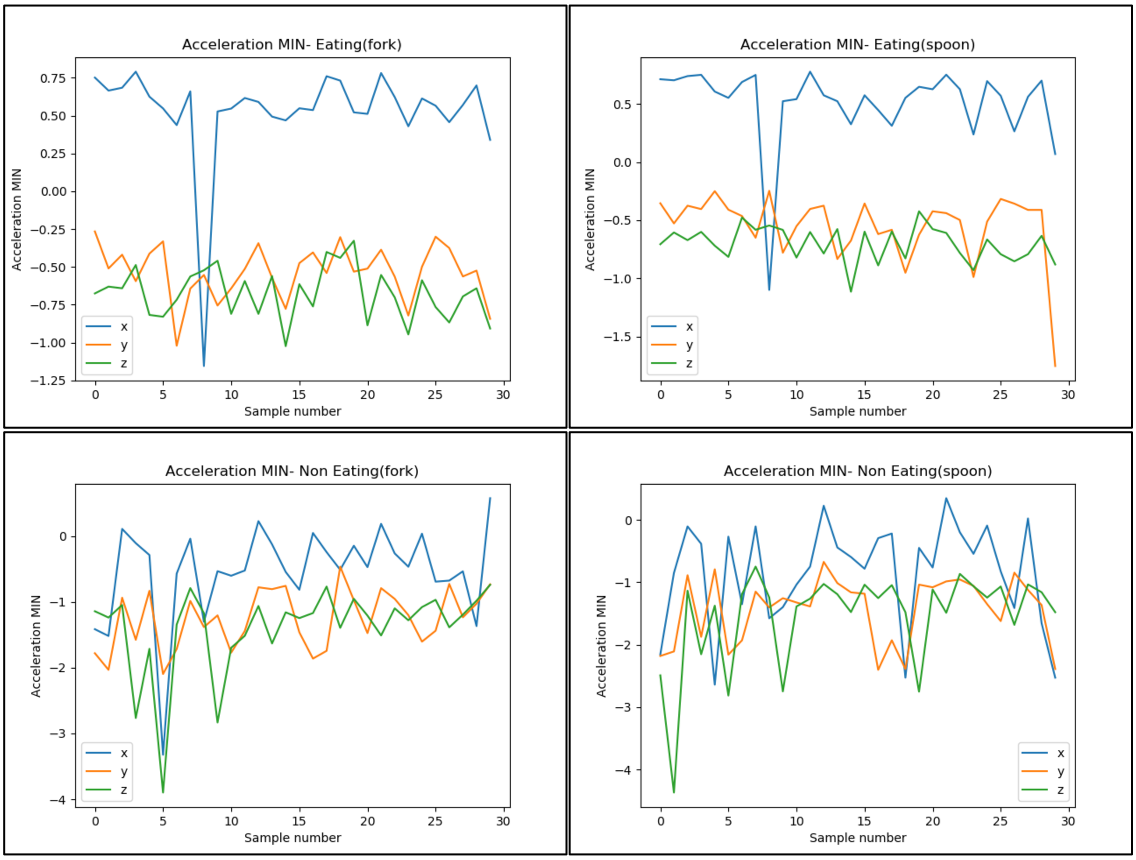






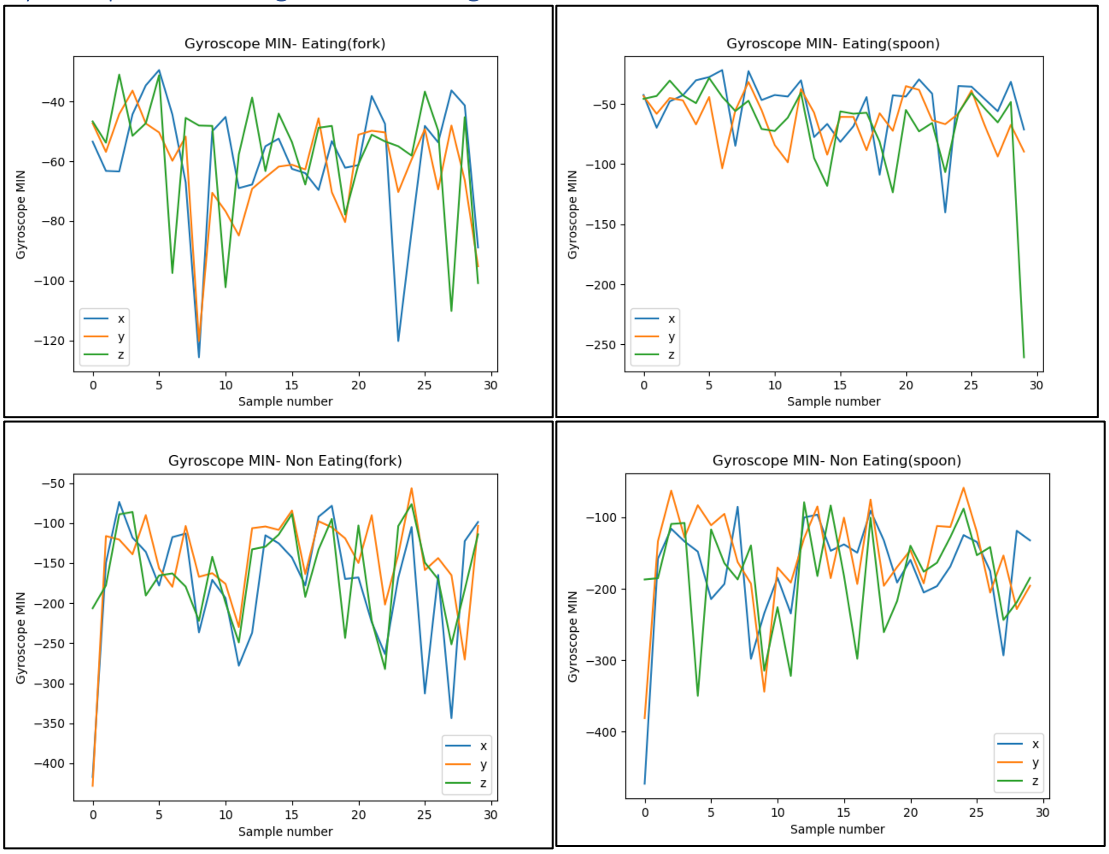
Again, I am not surprised with the orientation readouts. I am not sure at all about the validity of this sensor.

Acceleration MIN Eating vs. Non-Eating



Not surprised, distinction is possible but is hard to understand the physical meaning of it.

Gyroscope MIN Eating vs. Non-Eating



The same explanation here. A minor distinction. Very hard to understand the physical meaning of it.

Max

Find the maximum I will repeat the same paragraph (explanation). My intuition is that I will not be able to see any clear(major) distinction between the activities. Moreover, because some information about the position and the interpretation of the values is missing it is difficult to understand the mining of minimum by intuition. Nevertheless, some (minor) distinction would not surprise me

$\underline{Orientation}$:

$$MAX_{orientation} = \left[max_{ox}, max_{oy}, max_{oz}, max_{ow}\right] \; Where :$$

$$maz_{ox} = \; max\,\{O_{x_1} \dots O_{x_i}\}\;, \;\; max_{oy} = \; \max\,\{O_{y_1} \dots O_{y_i}\},$$
$$max_{oz} = \; \max\,\{O_{z_1} \dots O_{x_i}\}, \;\; max_{ow} = \max\,\{O_{w_1} \dots O_{w_i}\}$$

$\underline{Accelerometer}$:

$$MAX_{acc} = \left[max_{ax}, max_{ay}, max_{az}\right] \; Where :$$

$$max_{ax} = \; \max\,\{A_{x_1} \dots A_{x_i}\}\;, \;\; max_{ay} = \; \max\,\{A_{y_1} \dots A_{y_i}\}, max_{az} = \; \max\,\{A_{z_1} \dots A_{x_i}\}$$

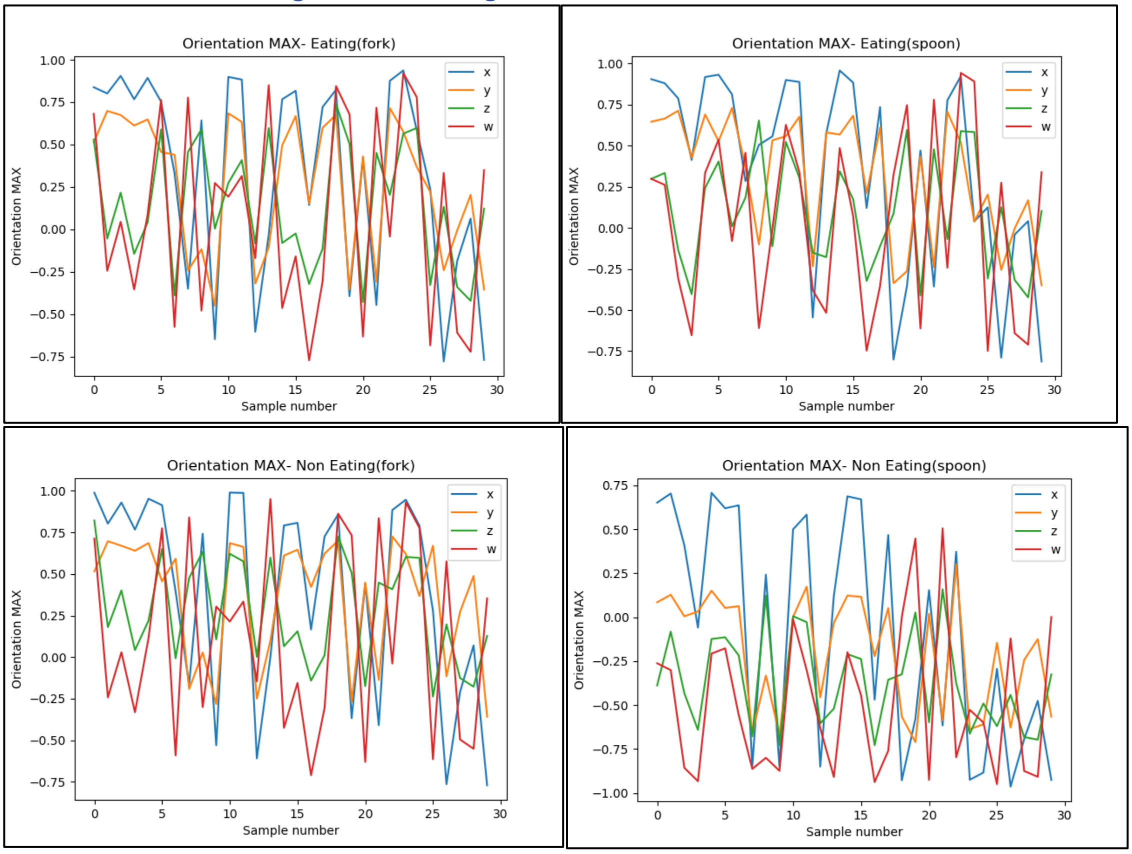
$\underline{Gyro}$:

$$MAX_{gyro} = \left[max_{gx}, max_{gy}, max_{gz}\right] \; Where :$$

$$max_{gx} = \; \max\,\{G_{x_1} \dots G_{x_i}\}\;, \;\; max_{gy} = \; \max\,\{G_{y_1} \dots G_{y_i}\}, max_{gz} = \; \max\,\{G_{z_1} \dots G_{x_i}\}$$

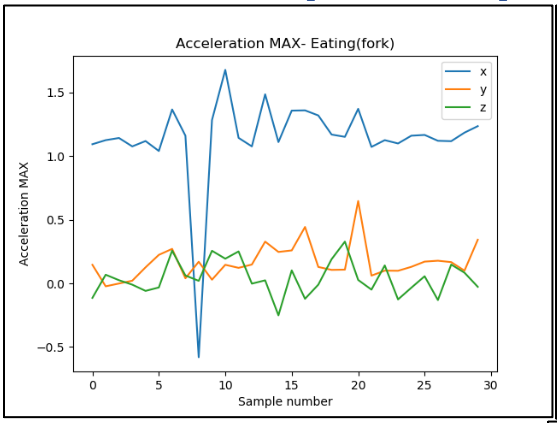
Plots

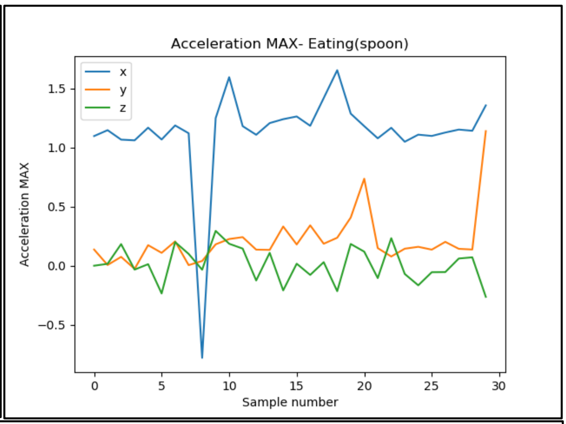
Orientation MAX Eating vs. Non-Eating

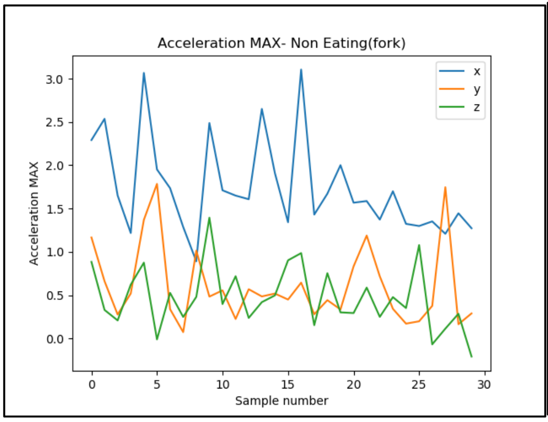


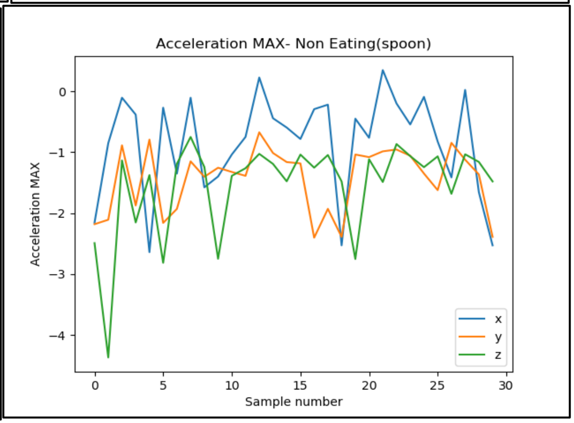
Again, I am not surprised with the orientation readouts. I am not sure at all about the validity of this sensor.

Acceleration MAX Eating vs. Non-Eating



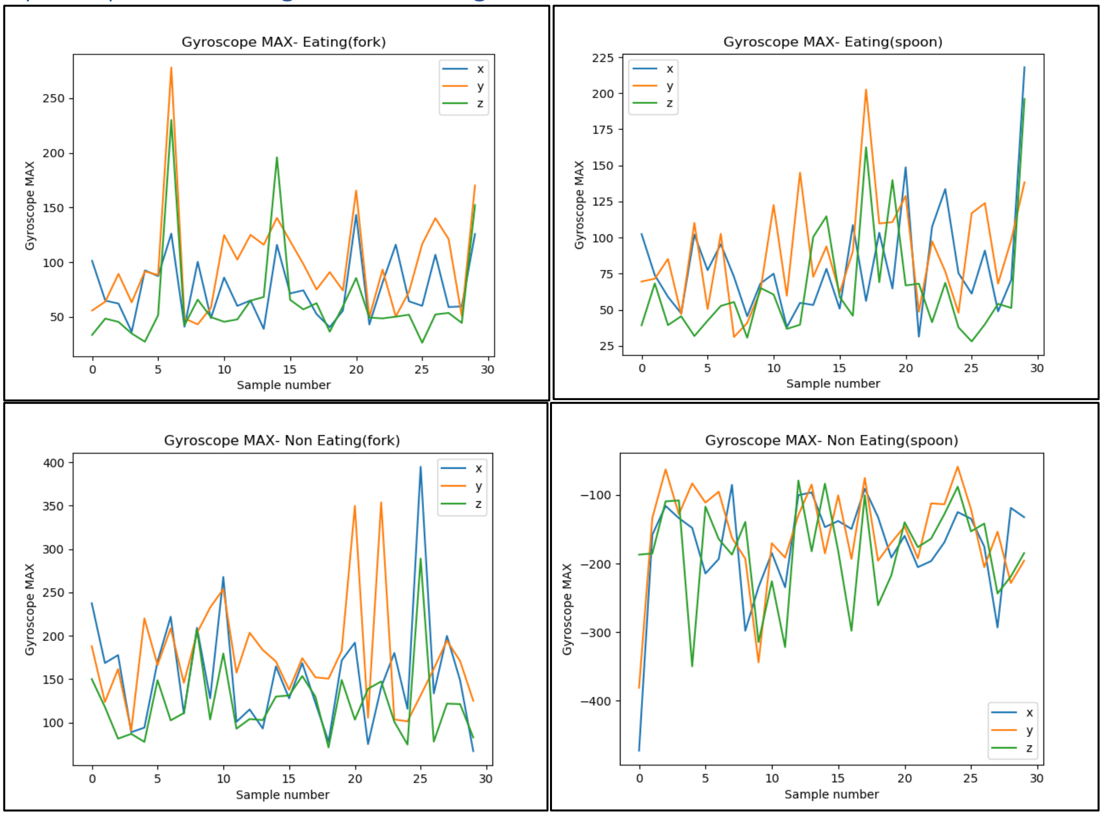






Same for the minimum feature.

Gyroscope MAX Eating vs. Non-Eating



Same for the minimum feature.

Phase Three: Feature Selection

[5]Principal Component Analysis (PCA) is a simple yet popular and useful linear transformation technique that is used in numerous applications, such as stock market predictions, the analysis of gene expression data, and many more.

The main goal of a PCA analysis is to identify patterns in data; PCA aims to detect the correlation between variables. If a strong correlation between variables exists, the attempt to reduce the dimensionality only makes sense. In a nutshell, this is what PCA is all about: Finding the directions of maximum variance in high-dimensional data and project it onto a smaller dimensional subspace while retaining most of the information.

Arranging the feature matrix

In order to work with PCA there is a need to arrange the feature matrix such that the PCA will be able to extract the principal components correctly.

Total number of features is 50 : Mean, Variacne, RMS, MIN and MAX calculated on orientation_x orientation_y orientation_z orientation_w, acceleration_x, acceleration_y, acceleration_z, gyro_x, gyro_y, gyro_z
These was calculated for each user for eating and non-eating activities therefor the feature matrix dimension is 60x50. To get matrix, first I have stacked horizontally the features of the eating activities:

```
X_1 = np.hstack\
            ((self.__mean_eating_fork,\
            self.__variance_eating_fork, \
            self.__rms_eating_fork, \
            self.__min_eating_fork, \
            self.__max_eating_fork  ))
```

Then I did the same with the features of the non-eating activities:

```
X_2 = np.hstack\
            ((self.__mean_non_eating_fork, \
            self.__variance_non_eating_fork, \
            self.__rms_non_eating_fork, \
            self.__min_non_eating_fork, \
            self.__max_non_eating_fork))
```

To get the final feature matrix I have stacked vertically the eating and non-eating matrices into one matrix:

```
X_fork = np.vstack((X_1, X_2))
```

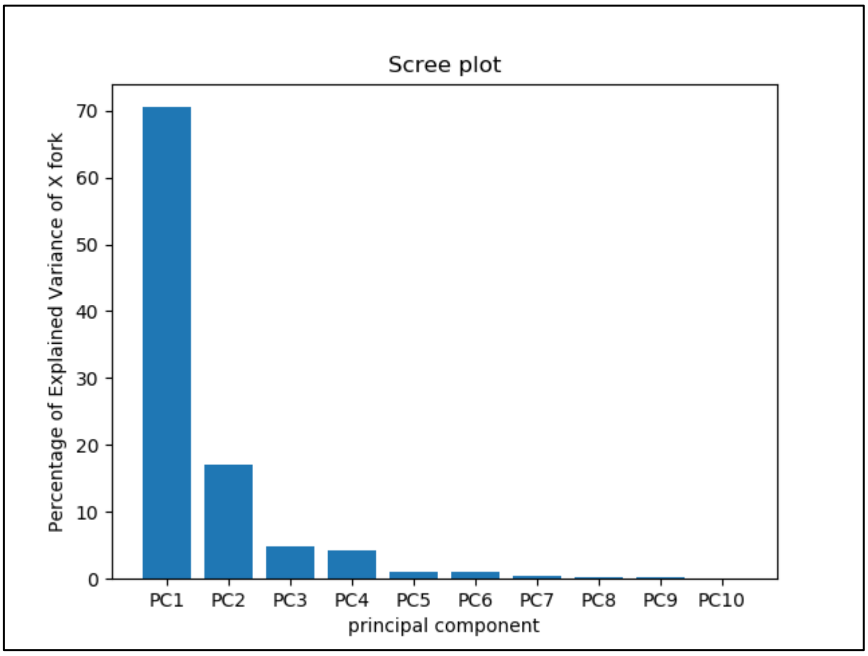
Where the dim of X_fork is (60,50)
PCA is performed on the covariance matrix (50x50) where each element represents the covariance between two features. Next, the eigen decomposition is done on the covariance matrix. The output of that is the Eigenvectors and Eigenvalues. The eigenvectors with the lowest eigenvalues bear the least information about the distribution of the data; those are the one that can be dropped.

To find out how many principal components we are going to choose for our nee feature subspace I have plotted a Scree plot. From the plot is we can estimate what is the number of PC we should select.
In the last step the 50x2 (number of features X number of principal components) dimensional projection matrix **W** transformed our samples onto the new subspace via the following equation:
Y= X x **W**, where **X** is our feature matrix (60x50) and **W** is the projection matrix (50x2). **Y** matrix (60x2) is our transformed samples.

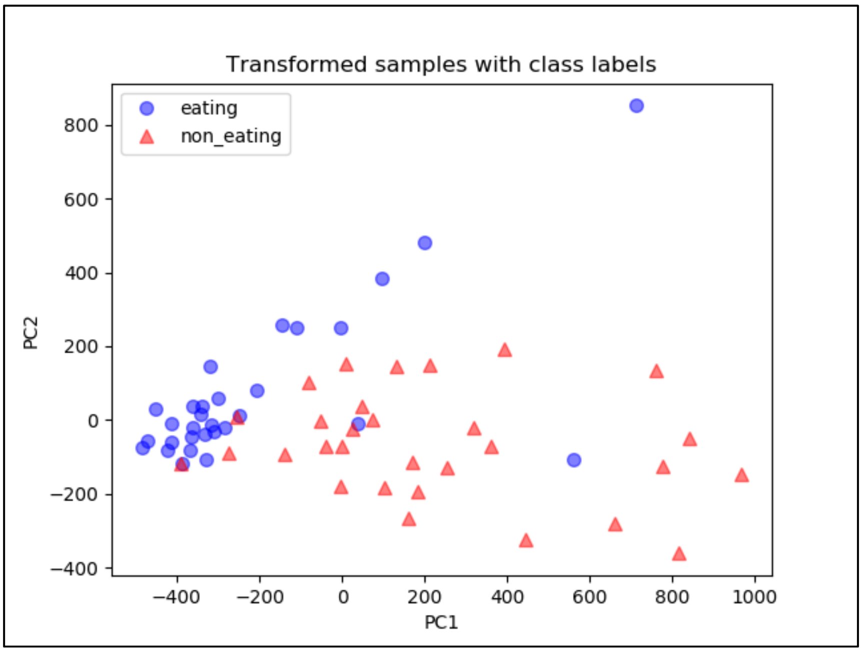
Note – the steps which I have described, is implemented under the hood by scikit-learn python package which is being used in this project the calculate PCA.

PCA Execution

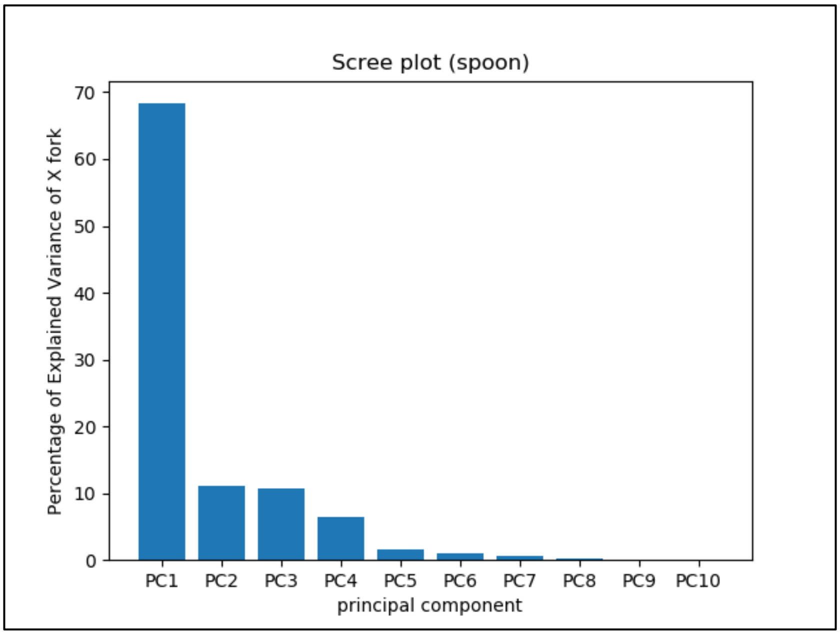
As described in, I have been using scikit-learn for the PCA calculation.
I have used the Scree plot below to find how many components to keep in the principal component analysis (2 principal components)



Plotting the principal components along with the labeled data (eating and non eating) can help us to visualize the eating and non-eating clusters using two dimensions in the new feature space



Note – This part was done on the features of the eating with fork and non-eating with fork activities. Applying the same process on the spoon dataset will yield the following plots:



PCA Results

Be using PCA we can see it strength in the case where the high dimensional problem (50 dims in our case) has been reduced two only two dimensions.
When looking on the data set that's very difficult to find the most important (that contains a lot of information) features and that's because there are many features. On the other hand, when taking a look on the features(graphs) which were described earlier for some of them we can say (with high probability) that they are not important/relevant only be looking on their graphs. However, for some features we can see a clear distinction between eating and non-eating activities (for example variance of gyroscope and variance of accelerometer).

Note on standardization:
One of the problems of PCA is that there is a different scale between the feature set. Therefore, there is a need to standardize the data before computing PCA. In our case I have assumed that's not the case (standardize is a one liner using skelern).

Code and environment setup

Python3
Linux/OSx
Dependencies:

```
import scipy.io
import matplotlib.pyplot as plt
import numpy as np
import os
import math
from tempfile import TemporaryFile
import re
from sklearn.decomposition import PCA
import pandas as pd
import seaborn as sns
from sklearn.pipeline import make_pipeline
from sklearn import preprocessing
from sklearn.preprocessing import StandardScaler
import errno
import os
import sys
```

How to run

python3 main.py

This will create two directories:
'out' and 'plot'
In the 'out' directory you will find all the extracted IMU data for all users and activities (eating, non-eating, spoon and fork).
In the 'plot' directory you will find the plots (which I have used for this report) organized in sub directories.
Important – make sure to delete of the 'out' and 'plots' directories before running again.

Reference:

1. D *Physical Human Activity Recognition Using Wearable Sensors*, Sensors Journal 2015,15,31314-31338;
2. https://www.w3.org/TR/motion-sensors/#accelerometer-sensor
3. https://www.researchgate.net/publication/223300172_Comparative_study_on_classifying_human_activities_with_miniature_inertial_and_magnetic_sensors
4. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6308535/#B7-sensors-18-04189
5. https://plot.ly/python/v3/ipython-notebooks/principal-component-analysis/