Lab Book

Machine Learning

Córdoba Romero, Javier
Corroto Martín, Juan José
Guerrero del Pozo, Álvaro

Beginning 10 October 2018

Contents

Wednesday, 10 October 2018

1 First steps

Done by Juan José Corroto, Javier Córdoba and Álvaro Guerrero

We've initialized the repo with `coookiecutter` directory structure and learnt the purpose of each directory. We've also learnt how to work with `pandas`, read and write a csv and how to work with spyder.

2 Data cleaning

We've picked the data of only the first day, we've done this by looking at the raw data and looking at the column *"TimeStemp"*.
Then we've deleted the *UUID, Version* and *TimeStemp* columns because they were non-numerical values.
Then we've tried deleting the rows that had *NaN* values, and we ended up deleting the whole dataframe. So, after scanning the dataframe, we have realized that the columns

1. *RotationVector_cosThetaOver2_MEAN*
2. *RotationVector_cosThetaOver2_MEDIAN*
3. *RotationVector_cosThetaOver2_MIDDLE_SAMPLE*

Had all their values as *Nan*.
After eliminating these 3 columns, we proceeded as before: we eliminated those rows that had *NaN* values and deleted 2 rows.
Then we saved this new data as processed data in its corresponding folder: data/interim/.

Wednesday, 17 October 2018

1 Feature Selection

Done by Juan José Corroto, Javier Córdoba and Álvaro Guerrero

First, we have modified how we pick the first day data: instead of getting the 2038 first rows, we convert the *'Timestemp'* to days and choosing the day 28 (the first one).

In addition to the columns dropped the last week, we have also dropped *UUID*. Then every *FFT* and *Middle Sample* rows have been dropped too.

We have chosen just two sensors to do an initial exploration: **Accelerometer** and **Linear acceleration**. The purpose is to try and explore the data related to the movement of the phone.
So, we create a new dataframe with only *Acelerometer* and *Linear Aceleration*. Finally, we drop *Covariances*, as we don't think they will be useful cause they are calculated as the relation *2-by-2* of the axis.

2 Visualization - Clustering

We then calculate the *PCA* with an *explained_variance_ratio* of **0.78** and **0.14**, which is a very good representation of the original 18 features we were studying.
With this results, we plot a *scatterplot* to visualize it(Figure 1).
After that we run the *k-means* algorithm with different number of centroids and print the *silhouette* and *distortion*. We initialize the centroids using the *K-means++*algorithm as well. You can see the measurements of distortion and silhouette of KMeans with different number of centroids in figures 2 and 3 respectively.

By looking at the plots, we have decided to choose **4** as the final number of centroids.

Finally, we run the *k-means* algorithm with said number of centroids and plot the results, each cluster having a different color, and the centroids being colored in red.

Before trying to get meaning from the clusters, we have decided to repeat this experiment by doing some exploration on the features.

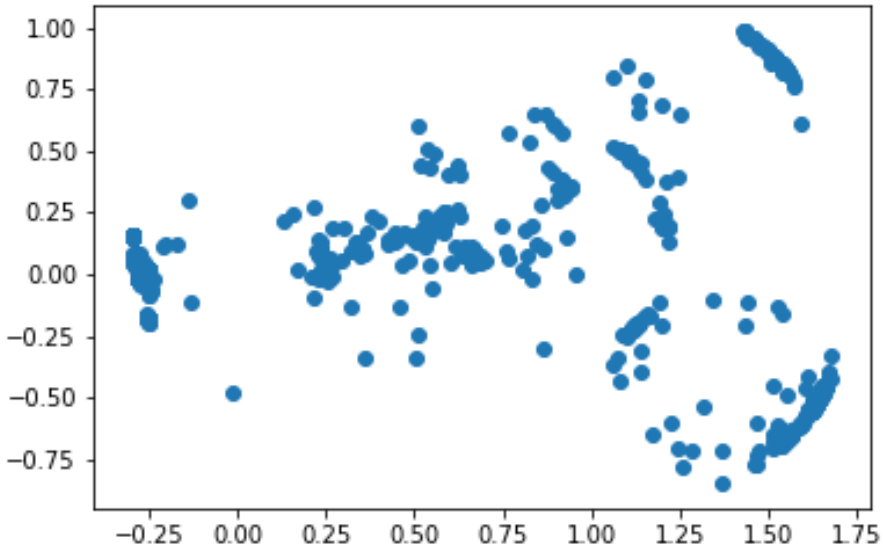

Figure 1: Plot of the data after PCA

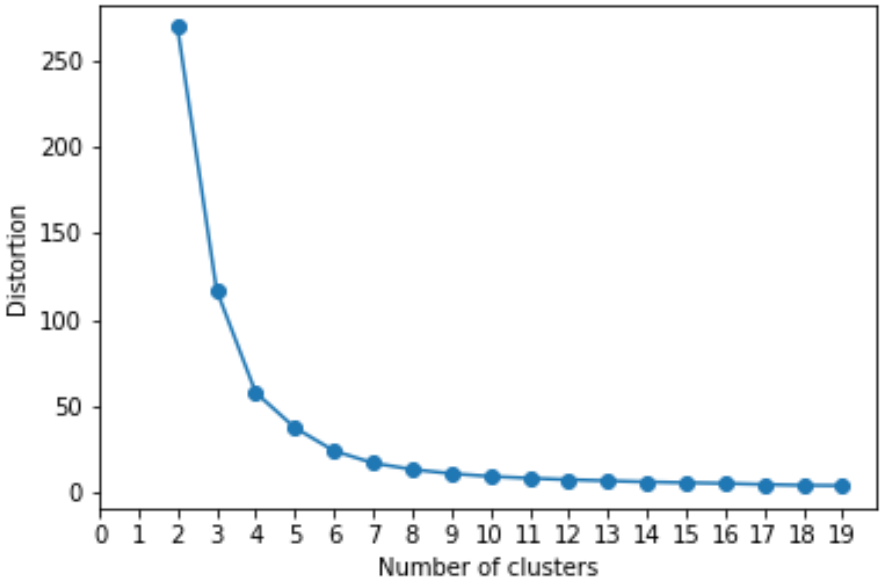

Figure 2: Distortion of K-means with different number of centroids

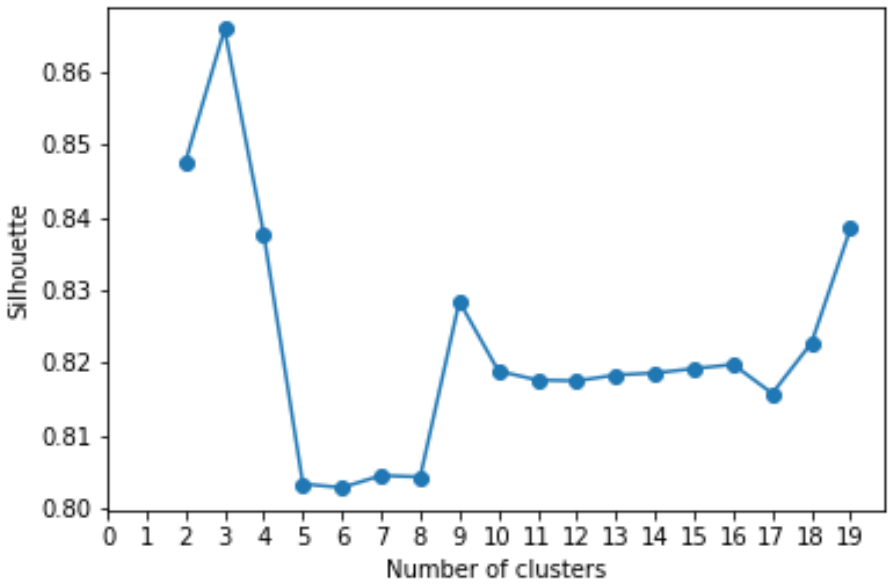

Figure 3: Silhouette of K-means with different number of centroids

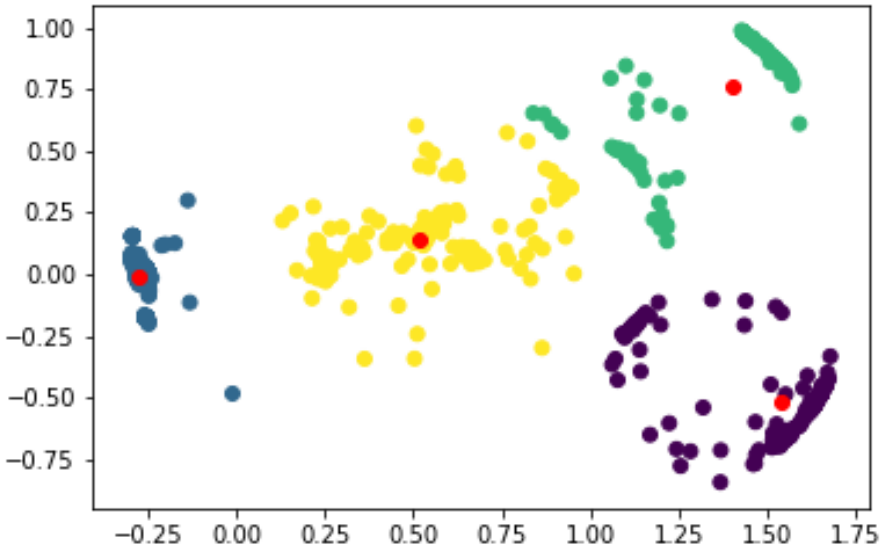

Figure 4: Plot of the data after K-means (centroids in red)

Friday, 26 October 2018

1 Feature Selection done well

Done by Juan José Corroto.
We have repeated the same process as before: pick the day 1 data from the whole database, but this time, we are going to perform some analysis on the features in order to remove those features that gives us less value due to redundancy, instead of removing some columns randomly. First, we still remove all columns that have to do with the *Fast Fourier transform*, since we don't know how it works and we can not extract knowledge from them. The same with the *Middle Sample* columns. After that we remove *NaN* columns and rows as we were doing proviously.

Now we are going to see the correlation between the features:

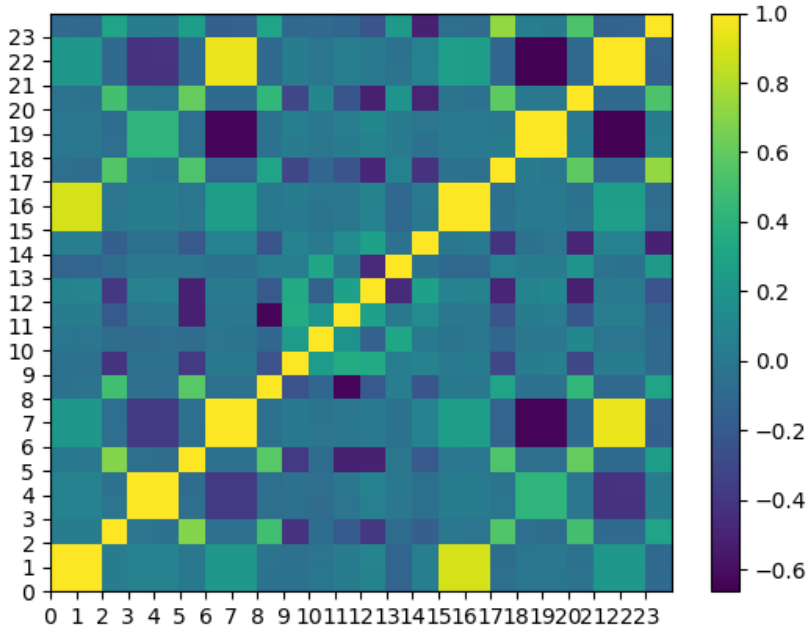

Figure 1: Correlation of the 24 features we now have

We can see some variables with a very high correlation in figure 1. There are some couple

of features with a correlation near to 1 (The maximum value), like features 0 and 1, or 21 and 22. This features are the mean and median values of every axis, so we remove all median values from our features.

Having a very similar median and mean values gives us some information: The mean of a value is an stimator strongly affected by outliers, while the median is not. If the median and the mean have similar values, this means little ammount of outliers in our data (It does not mean they are non existent).

From the correlation matrix we can also see some features with a very big correlation: the mean values of each axis between both sensors. This means the value of x for the Accelerometer and the LinearAcceleration are highly correlated, so we remove one of those (In this case, the value of the Accelerometer is dropped). The result can be seen in Figure 2.

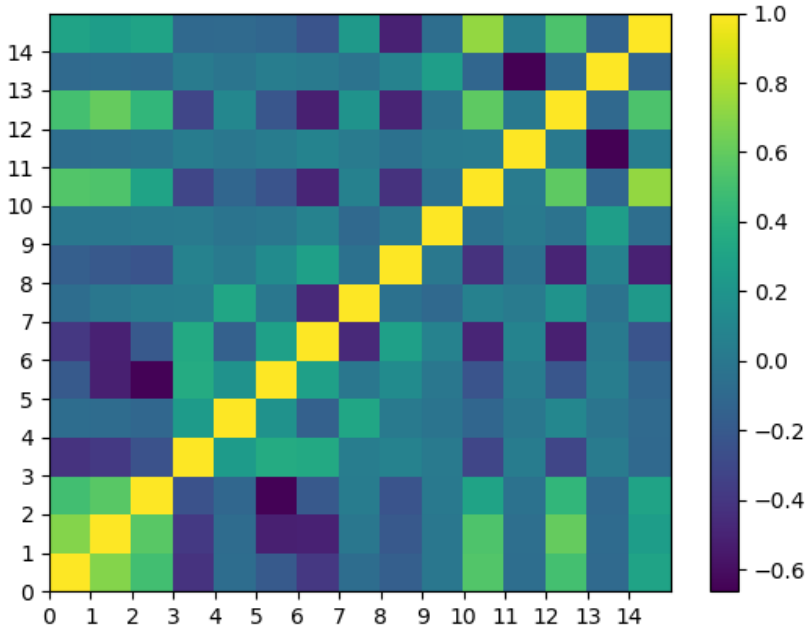

Figure 2: Correlation of the 15 features we now have after removing some

There are a couple of features that are still highly correlated: The variances of the accelerometer, but this correlation is not high enough to remove them straight up, so we are going to use hierarchical clustering to see if we can extract more information. As seen in the Figure 3, there are some clusters of features, but the most similar ones are features 0, 1, 2 and 12. Those are the variances of the accelerometer. These are the features we were looking for, therefore, we can use only one of them.

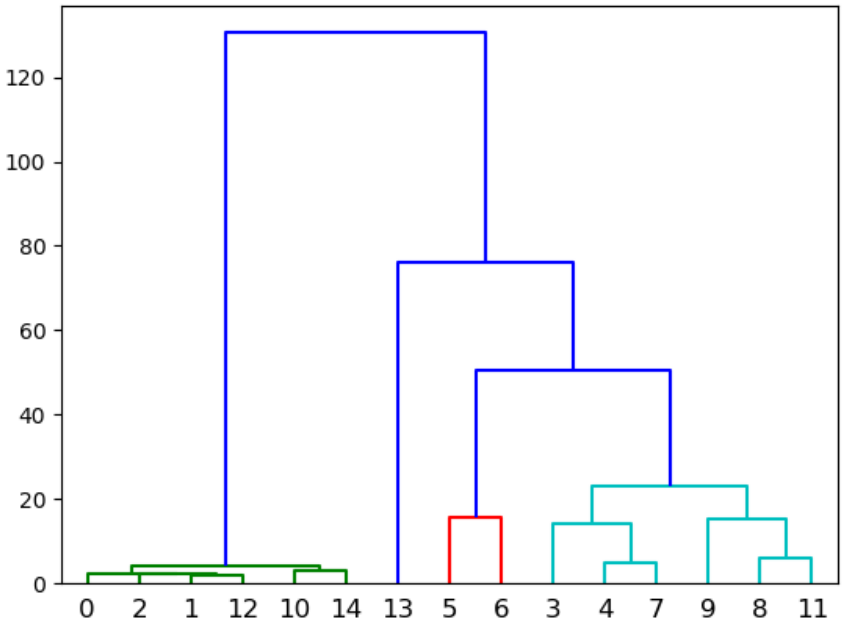

Figure 3: Features dendogram

2 Repeating the process

Done by Juan José Corroto.
With our new dataset finished from the previous experiment, we are going to repeat the process of our last day. First we apply PCA with an *explained_variance_ratio* of **0.76** and **0.14**. This ratio is very similar to the one we had in the previous iteration, and pretty good considering we have removed features that were highly correlated.

The data is plotted in Figure 4, and we can see some differences with respect to the plot we had in the previous day.
We run K-means several times again. We get very good values of Distortion and Silhouette again, thought this time they are even better for 4 clusters, so we select again this number of clusters and run K-means, with the plotted results shown in Figure 7.

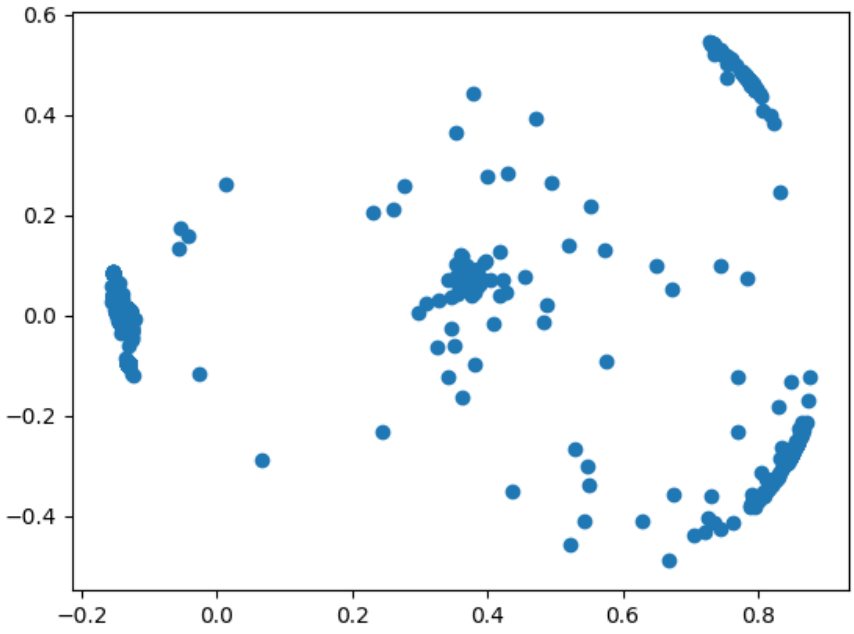

Figure 4: Plot of the data after PCA

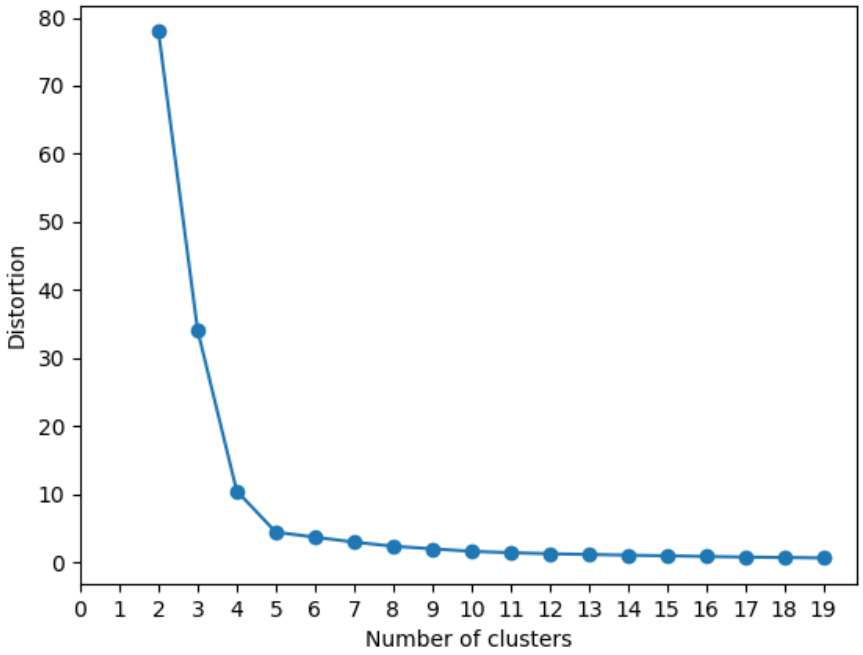

Figure 5: Distortion of K-means with different number of centroids

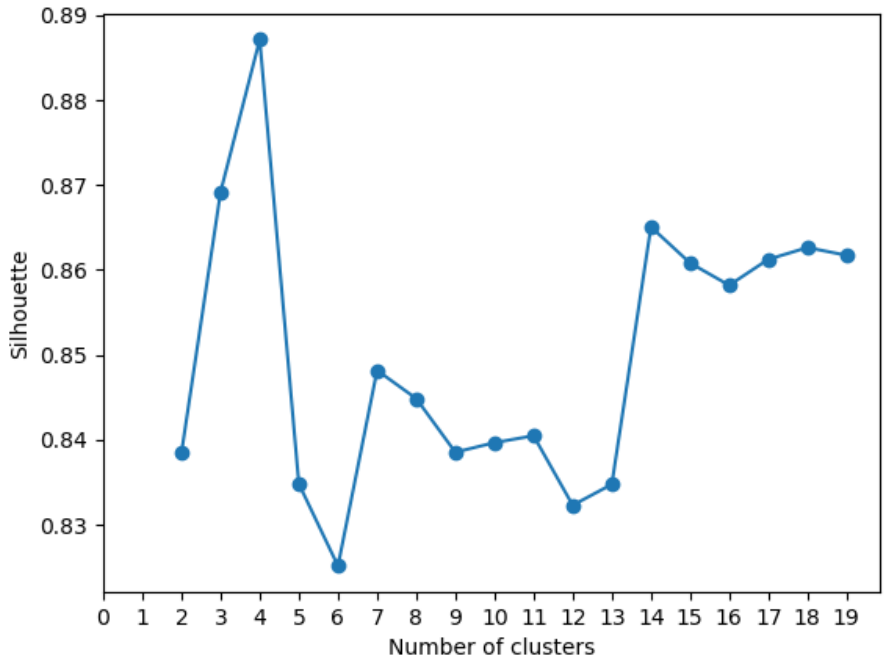

Figure 6: Silhouette of K-means with different number of centroids

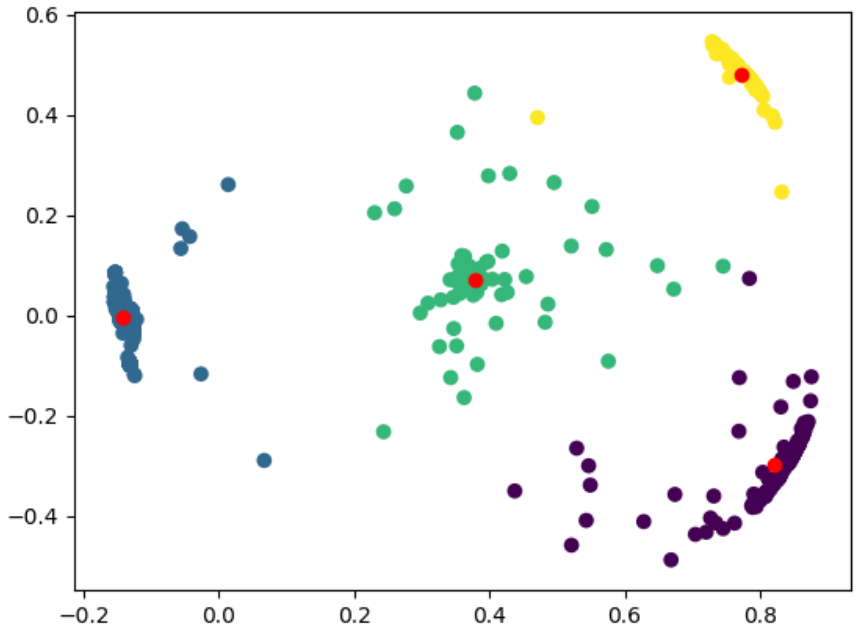

Figure 7: Plot of the data after K-means (centroids in red)

Saturday, 27 October 2018

1 Using other sensors

Done by Álvaro Guerrero del Pozo.
In this experiment, we have used what we have learnt until now, this time applied to other sensors. Again, we load the dataset, but just keep the data of the first day. Then, we drop any column that contains no information (i.e is null) and rows with null values. Then, we just keep columns related to *GyroscopeStat* and *RotationVector*
As before, we now plot correlations between features:

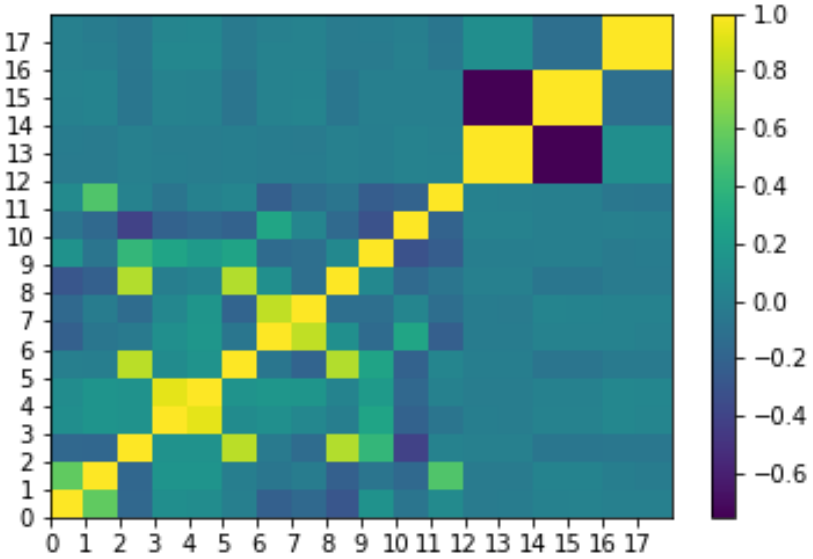

Figure 1: Correlation of the 18 features

As expected, *Mean* and *Median* values of each axis, of both sensors are highly correlated, so we can safely drop one of them, as the only add redundant informacion. We choose to remove *Median* values. But, there is an exception: X axis of the *Gyroscope*. The correlation between *Mean* and *Median* isn't high enough, so we don't remove any of them. Also, we can see that there is a surprisingly high (inverse) correlation between *Y Mean* and *X Mean* of the *Rotation Vector*.

It's not as high as other values (most of the previous ones had a correlation of near to 1), but still, with a correlation of around **-0.75**, we have decided to remove *Y Mean*. As a result, we are left with 12 features, whose correlations can be seen in the next figure:

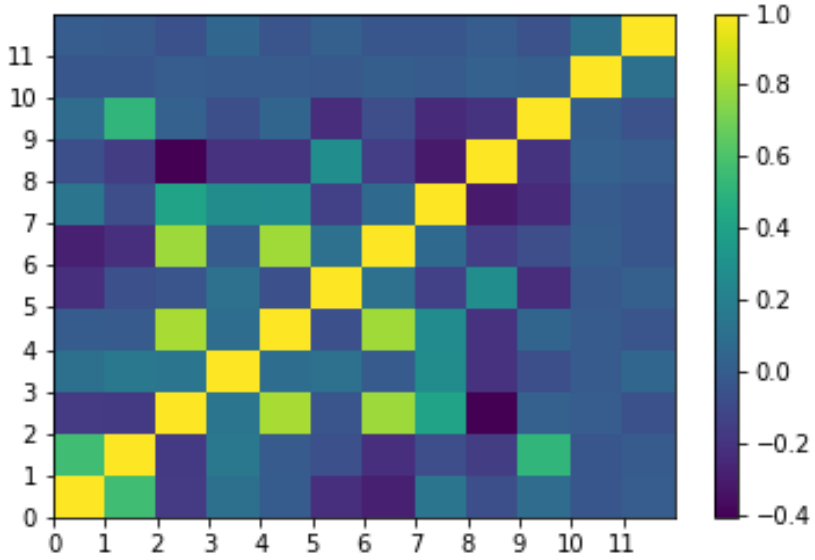

Figure 2: Correlation of the 12 features

After that, we plot the dendogram of the features (Figure 3).

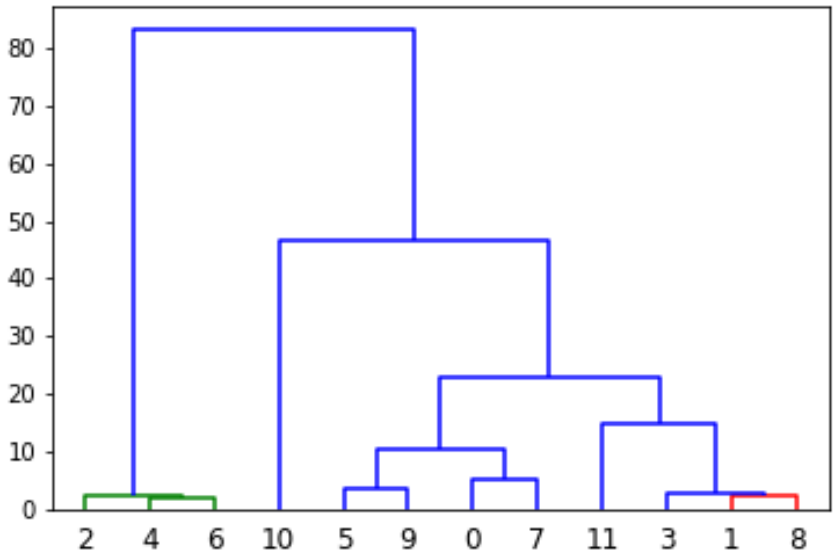

Figure 3: Dendogram

As we can see, the most similar features are the *Variances* of the three axis of the gyroscope (at, the left, in green color), so we drop two of them, as they don't add much informacion, and keep only *GyroscopeStat_x_VAR*. The next more similar features (right, red color), are *GyroscopeStat_x_MEDIAN* and *GyroscopeStat_COV_z_x*, wich is something we were expecting, due to the influence of the x axis of the *Gyroscope* in both features. Due to this fact, we don't consider this features similar enough so one of them gets removed, so we keep both.

Now that we have our data only with the features that are considered relevant, we proceed to visualize the data, by using *k-means*.

First, we apply PCA, and obtain an *explained_variance_ratio* of **0.825** and **0.130**. It's a high result, so we expect to obtain a good visualization. The plot of PCA is:

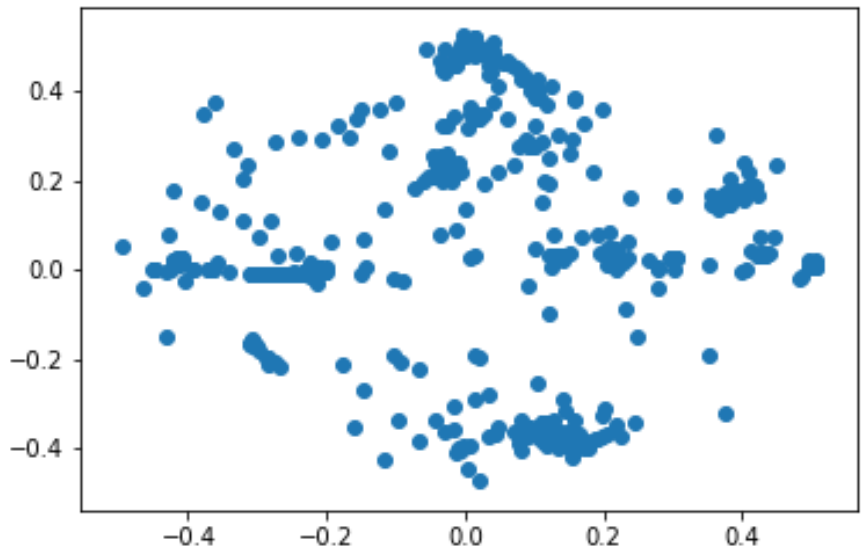

Figure 4: Plot of the data after PCA

After that, we run *k-means* several times, from 2 to 20 *centroids* and plot the Distortion 5 and Silhouette 6.

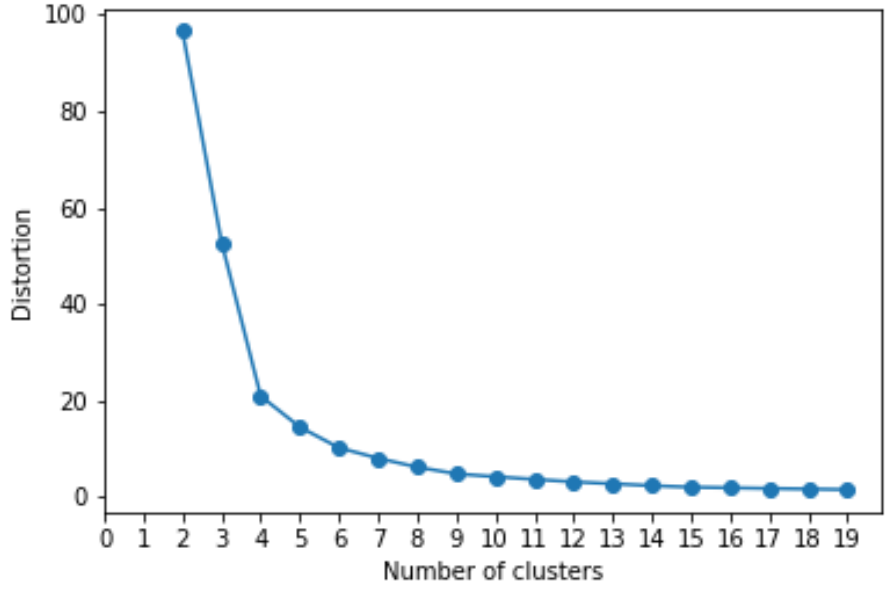

Figure 5: Distortion

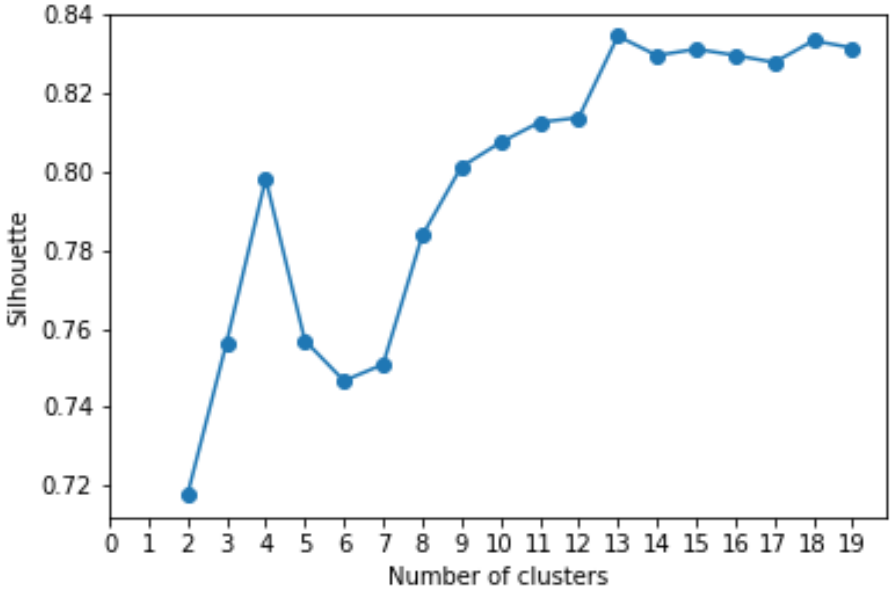

Figure 6: Silhouettes

We choose 4 as the best number of *centroids*, as there is a maximun in the *silhouette*, and the *distortion* is low. The resulting plot is shown below.

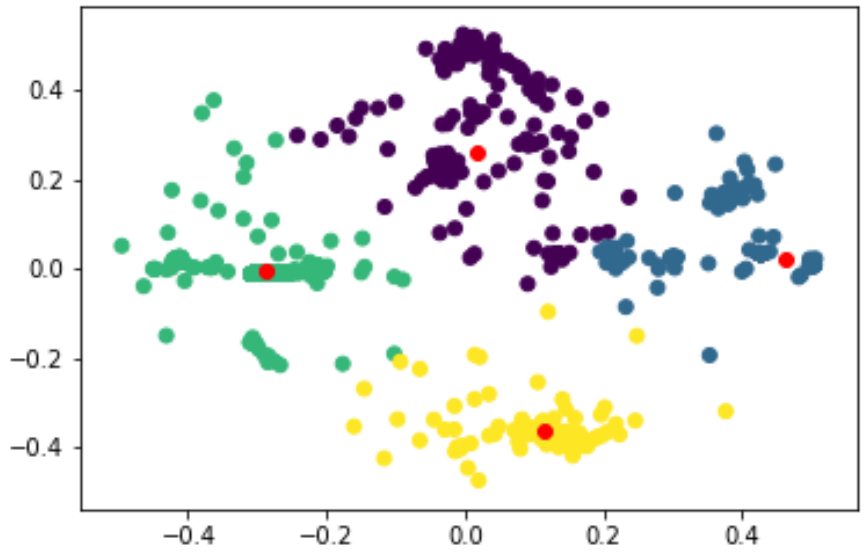

Figure 7: Plot of the data after K-means (centroids in red)

As we can see, the centroids have been placed where there are higher concentrain of points, but there is a decent amount of points that would be assigned to the neightbour cluster, should the centroids change slightly. Is a matter for another experiment to interpret this results, and decide wether they are good enough or not.

Tuesday, 30 October 2018

1 Selecting different data

The aim of this experiment is to select data from a different day, to see if it is significantly different than that of the first day. So, we are going to repeat the process of studying the *Acceleromenter* and *Linear Acceleration* variables.

The first thing to do is to remove all the features we can not use: as before, we remove the timestamp and the non-numerical values, and also the features that have something to do with the *First Fourier Transform*. After that we study the correlation between variables.

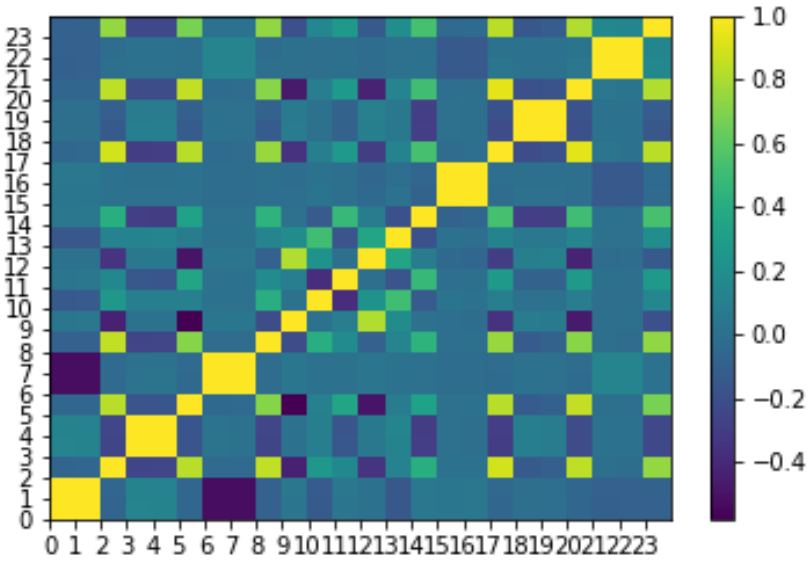

Figure 1: Correlation of the 24 features we have

As we can see in figures 1, 2 and 3, the situation is very similar: The mean and median of every sensor is highly correlated, and the variances of the accelerometer are highly correlated but not high enough too discard them. But when making the clustering we have proof enough that their information is very similar and we discard them as well, leaving us with the same 12 features we had on the previous experiment.

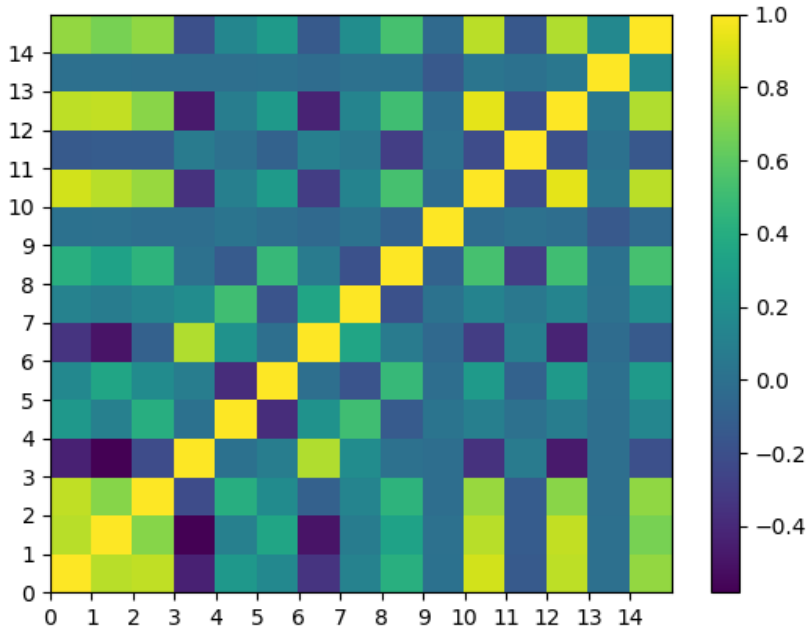

Figure 2: Correlation of the 15 features we now have after removing some

We are now going to plot the data, as seen in figure 4. For that we have to, of course, apply PCA. In this case, the *explained_variance_ratio* is a little lower: **0.51134397** and **0.17562308**. This can be explained because we have almost double the data as before. Anyways, it is still high enough to have a good representation of the data.
The next step is to do some clustering to the data. As before, we are going to use *K-means*, and we are also going to use the elbow method to see the best number of clusters.
In figures 5 and 6 we can infere that, again, a good number of clusters is 4. In Figure 7 we can see a plot of the data after clustering. The data looks a lot different than that of the first day, and the clusters seem less defined even. Maybe we should try a different cluster algorithm than K-means?

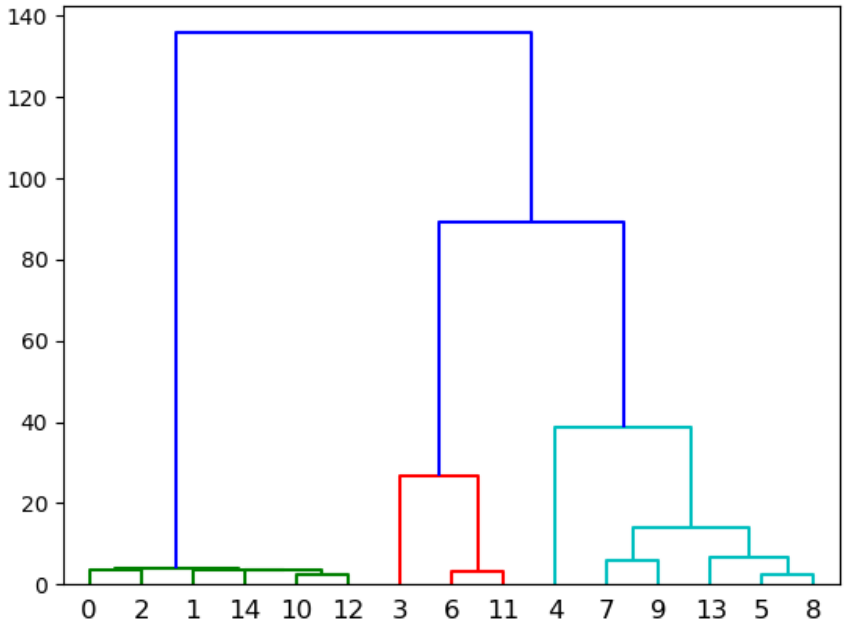

Figure 3: Features dendogram

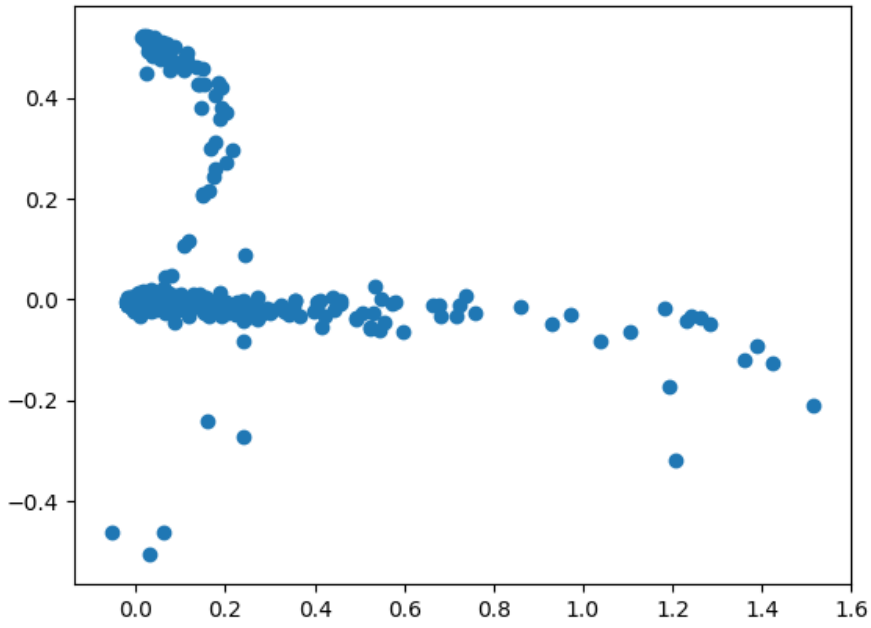

Figure 4: Plot of the data after PCA

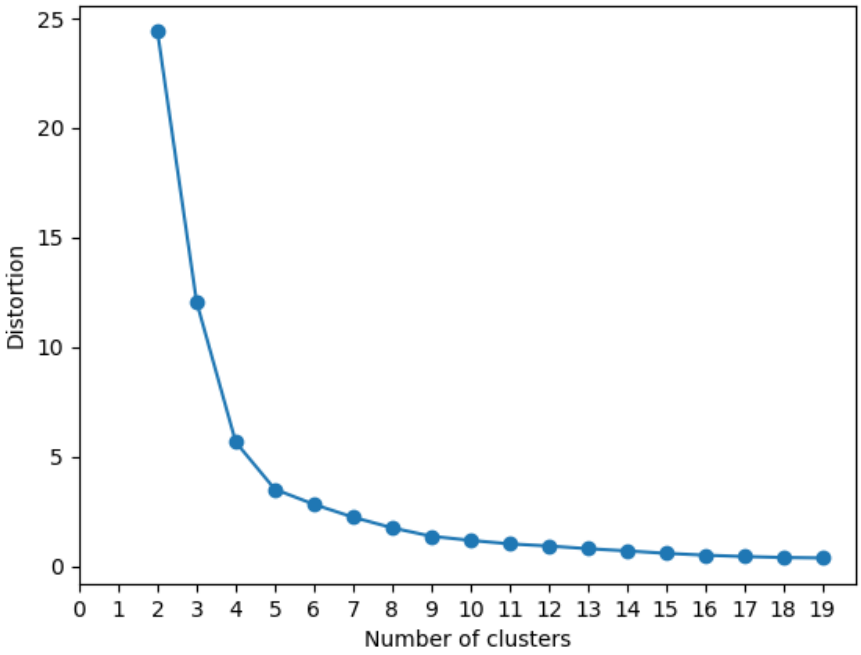

Figure 5: Distortion of K-means with different number of centroids

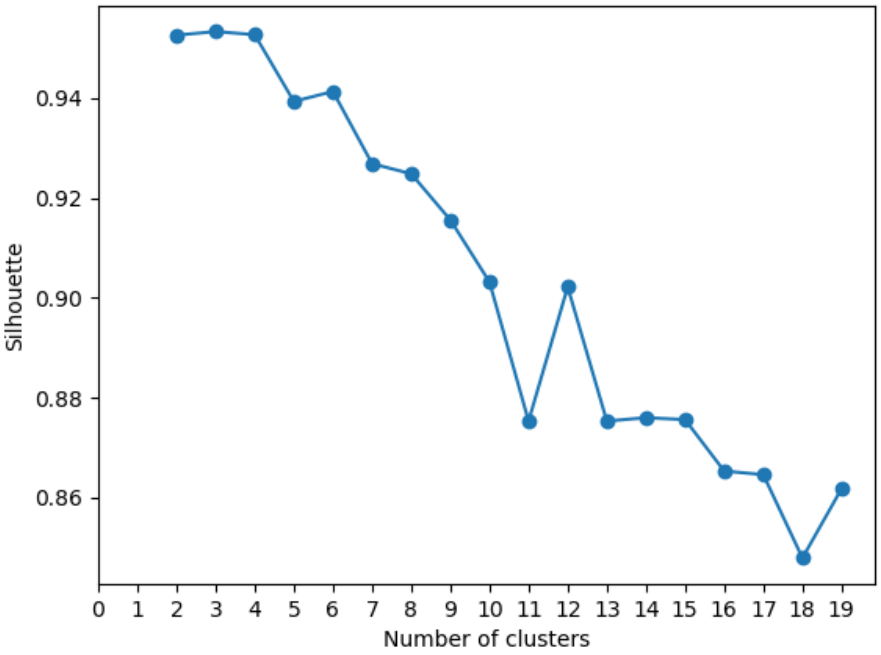

Figure 6: Silhouette of K-means with different number of centroids

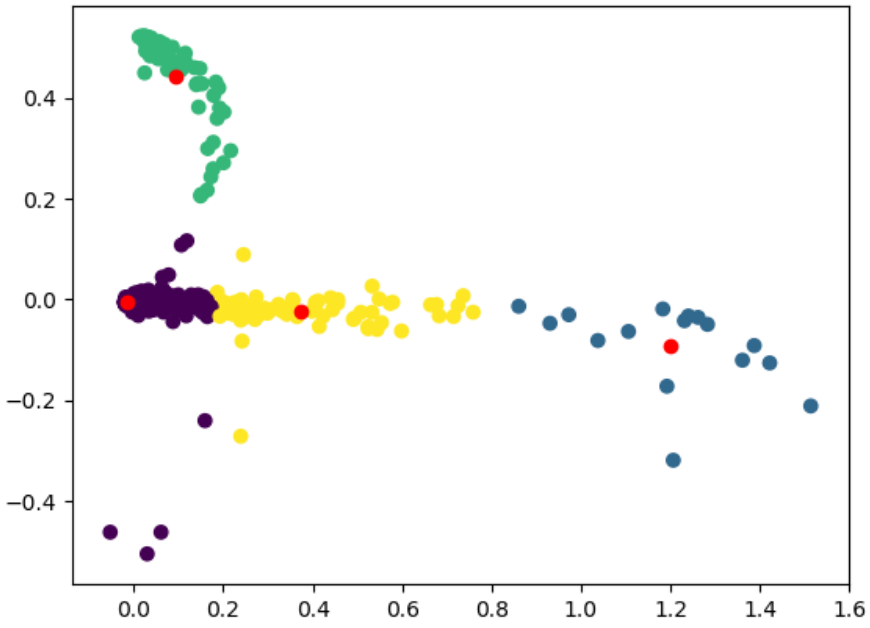

Figure 7: Plot of the data after K-means (centroids in red)

Thursday, 1st November 2018

1 Interpreting the results: Gyroscope

After all the previous experiments, the only remaining work is interpreting the results. In this section, we will interpret the results of the Gyroscope.
We are going to focus on the *Rotation_Vector*. The first part is understanding how the three axis of this sensor work.

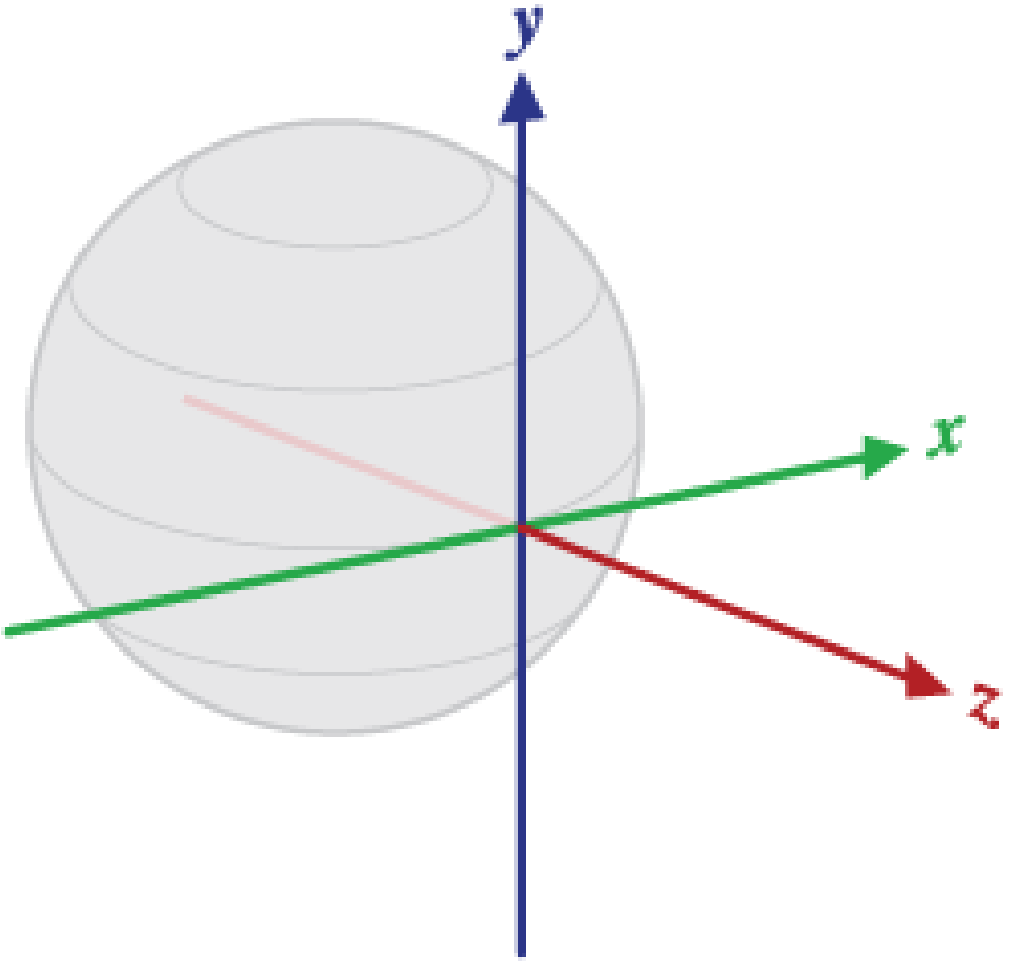

Figure 1: Rotation Vector Axis Globe

By looking at the image, we can interpret that the z axis is perpendicular to the phone screen, the y axis is parallel to the phone's longest side, and x axis is perpendicular to both of them.
The sensors *should* give a result of 0 when te phone is laying in an **horizontal surface**.

The next thing we do is plotting the results of the clustering along with the measures of the sensors. We'll use *seaborn* library for that.

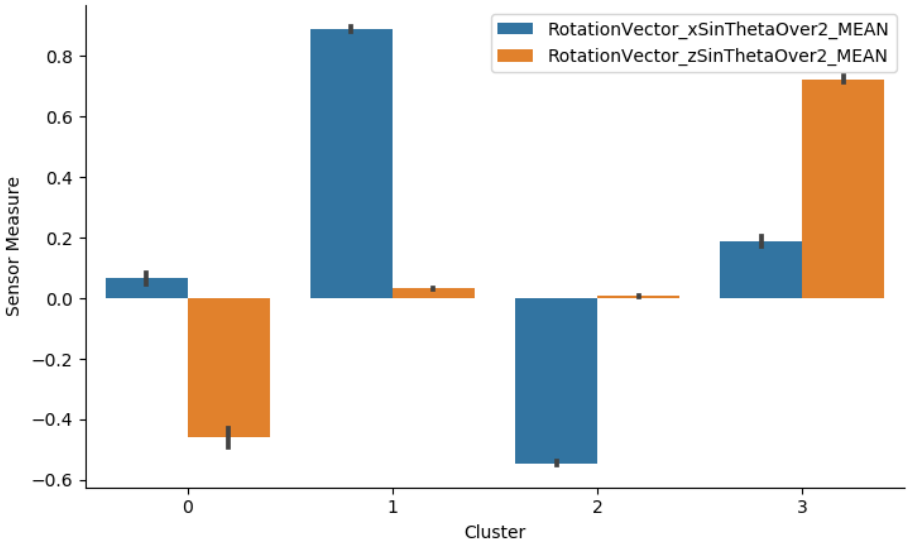

Figure 2: Rotation Vector Barplot

As we can see, the first cluster's (number 0) z axis has a value of **-0.5**. Taking into account this measures are a *sin* (and therefore, they can have values from 0 to 1), we assume this clusters includes the measures of when the user is using the phone, as the *arcsin* of **-0.5** is *30 degrees*. In other words, the phone is inclined so the user can see the screen.

In the next cluster, Cluster 1, we see high measures of the x axis. Taking into account that the sensor doesn't activate when the phone is laying horizontally in a surface, we have concluded that measures of (near to) 1, in the *Rotation Vector x axis* mean that the user is using the phone vertically, or with the phone in his/her trousers' pocket.

Looking at the Cluster 2, we can see that it's similar to the number 1: only x has measures, while z has values near to 0. Also, we can see that values for x axis are negative, instead of positive. We have decided that this measures represent the same actions than Cluster 1, but with the phone in the opposite position (i.e. instead of the screen facing the leg, it is facing to the direction the user is walking, or vice-versa).

Lastly, we have to talk about Cluster 3. Given that both the x axis and z axis have values different than 0, we know that the phone is inclined. The most common situation where this could happen, as our thinking, is the situation when we are sitting and the phone is in our trousers pocket. It is not exactly **on** our leg (all sensors would give values of 0, it would be the same as the phone laying in a table), but in our pocket. In this situation the phone tends to "fall" to the right if it's on the pocket of our right leg, or to the left otherwise, giving the results we see in the above figure.

Friday, 2nd November 2018

1 Interpreting the results: Acceleration, Day 1

Similar to what we did yesterday, we are going to try and interpret the results of our process. Today we are going to address our unsupervised learning of the Acceleration of the phone (Regarding the sensors *Accelerometer* and *Linear Acceleration*).
As said in the Android API, the axis of the phone are equal to those of the gyroscope: Positive X means acceleration towards the right, and positive Z means acceleration towards the sky (Considering the phone flat on a table).

In terms of understanding the data, our best bet is try and look at the mean value of each record; the covariances between the different axis are a lot harder to interpret. Besides, the values of the *Linear Acceleration* sensor exclude the gravitational pull, which means that an acceleration of 0 means no acceleration (In contrary to the *Accelerometer*, which does not exlude it, and therefore no acceleration in the z axis means free-falling) Considering that, we can look at the results of day 1 for the three axis for each cluster in Figure 1.
From this image, The easiest cluster to make sense of is cluster 2: The acceleration of the three axis are close to 0 (Considering the noise of the sensors). This can mean that the phone was standing flat on a table.
Cluster 1 can also be easy to interpret: The phone is constantly moving along its z axis (In the direction of the screen). If we take into account what we learned from the gyroscope, we know that the user of the phone has it in his pocket with the screen facing forward. This could mean that the user was travelling forward while standing, therefore the acceleration. The other two clusters are a lot more difficult to understand. It seems the phone is accelerating towards its left and down in cluster 0. In a similar fashion, the phone is moving to the right, up and contrary to its screen in cluster 3. Those kind of movements seem very wierd to us, and are then hard to define in terms of a person doing a clear activity. Another important thing is the number of points in each cluster. Clusters 1 and 3 have a lot less values than cluster 1. This seems to indicate that the person in day 1, according to our results, was accelerated forward most of the day.

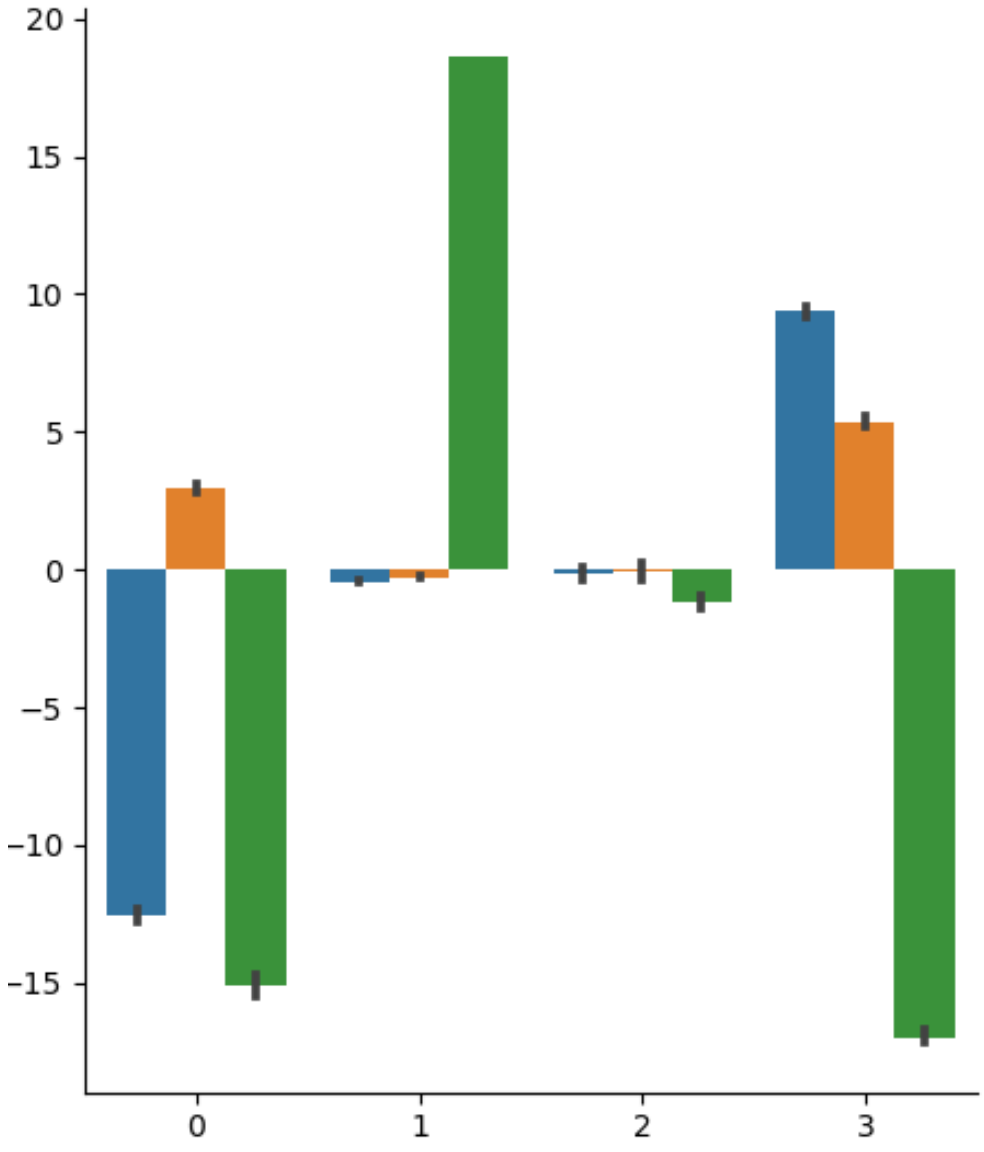

Figure 1: Acceleration Barplot

2 Interpreting the results: Acceleration, Day 2

The most interesting thing we saw when analyzing the same sensors on a different day was the difference in the data. The features had the same relations, same correlation, but the data was completely different. This can also be seen in the results of the clustering (Figure 2).

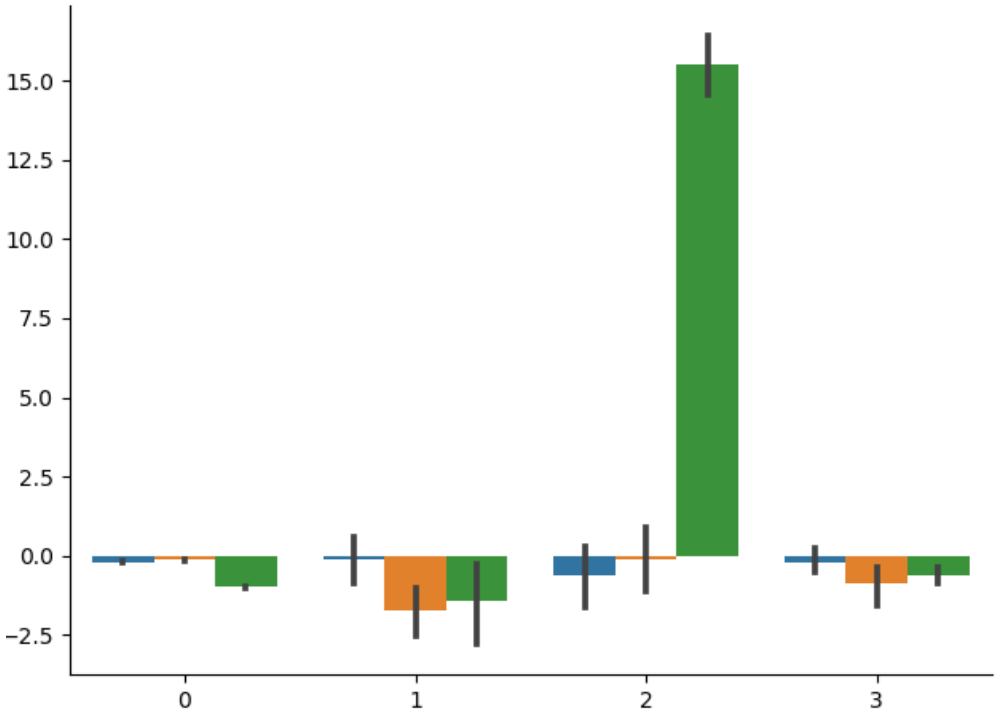

Figure 2: Acceleration Barplot Day 2

Two of the clusters we had are still present: Cluster 0 has all values close to 0, which means the phone is standing flat on a table. Cluster 0 has values close to 0 except for axis Z. This means, according to our interpretation, that the user is walking. The other 2 clusters are a lot different. On the one hand, most of their values have a very high variance, and also, their values are very scattered (One value can be of -3, while the next value, 15 seconds later, is 15). On the other hand, these 2 clusters (Cluster 1 and 3) have very little values. We are talking cluster 1 has 16 values out of more than 4000. These made us think that these clusters were bad results. Also, looking at the plot we obtained after applying k-means (7 from the 30th of October), we could think that these clusters were created due to the supposition of k-means that the distribution of the data is globular, which in this case is not. Since our data seems a lot more dense in a particular area, we applied DBScan. Looking at the results of the DBScan, we can see that these 2 wierd clusters end up merging with cluster 0. This gives us a result of 2 clusters on this day: One where the phone was not being moved, and therefore

not accelerated, and one where it was highly accelerated in the direction of the screen, presumably while moving (Like on Day 1).