Assignment 1

Group 15: Xinyu Hu Andrei Udrişte Rui Pan

Vrije Universiteit Amsterdam

1 Theoretical part

1.1 Chapter 2

Q1.

(a) These users may be doing different things, so they will have different locations, accelerations, etc.
(b) The sensor data may be inaccurate, perhaps they used a different device to collect the data.
(c) Each person's data may contain different outliers.

Q2.

(a) The task of the model: Depending on the task that has to be realized the granularity of the measured data might change, if we have a classification task, we can use a higher granularity compared to a regression task;
(b) The dimension of the data set: If the data set is to small then a high granularity might result in a small set of data points, that would be hard to train on, and produce overfitting;
(c) The number of missing values in the dataset: If the number of missing values in a data set is high we can use a higher granularity factor and average over the values that are present in the data set, ignoring the missing values.
(d) Experiments: by doing experiments with different granulation levels we can determine the value that will fulfil our task the best. One thing to look for would be to not use random values, the best approach would be to have a pattern for the values used for granulation, and to also pay attention to the other constrains that we might have on the data set.

Q3.

(a) Focusing on unsupervised learning and reinforcement learning, we could perform two tasks: (1) a clustering problem, namely characterize the data, and make assumptions about certain properties, such as sensor, a cluster of intense activities, and one with limited activities, (2) a reaction problem, smart phone can react to the certain emergency according to the heart rate of a user based on a policy, which specifies when to alert user, and sensor's data.
(b) When adapting semi-supervised learning, we could perform a task: (3) a learning user's environment pattern problem, we could make use of labeled activity and unlabeled sensor value such as the light intensity and pressure value to tell whether the user is indoor or outdoor.

1.2 Chapter 3

Q2. A distance based algorithm is much more beneficial when we don't have a lot of data points and it is impossible to create a distribution for the data, in this case the only measurement we can rely on would be the distance between the points. So the main advantage of distance based algorithms would be that they do not assume that the data comes from a distribution.

Q4. The time complexity is $O(N^2)$, because the local outlier factor algorithm traverses all its neighbors where it computes local reachability density, which also needs to traverse all its neighbors. To improve the approach, one can reduce the number of distances that need to be computed. Otherwise, a powerful GPU can help to accelerate the efficiency of the algorithm.

1.3 Chapter 4

Q1. Assuming a data set collected from a phone light sensor, which records the change of light lux per second. We can summarize the data within a window size with four functions, the summarized values can be used to tell if the user is indoor or outdoor. Since the light doesn't change frequently, whichever method we use can summarize the data correctly.

Q6. Considering we wish to predict the user's mood, we can collect people's heart rate, finger pressure on the screen, and battery of the phone. Firstly, the faster your heart beats, the more emotional you are. Second, the harder you press on the screen, the more pressure you are under. Lastly, people often spend more time on their phones when they feel depressed.

Q7. The advantage is that it's straightforward to implement and fast to run, the disadvantage is that the stemmed word output might contain inaccuracies because we only care about the words to map to the same stem.

2 Practical part

2.1 Chapter 2

Q1. The data that was used to create Fig 1 is collected using only a smartphone. For the recording of the data we used the application "phyphox" offered by RWTH Aachen University. The app offer the possibility to record a variety of sensor that are incorporated in the smartphone, but for our project we decided to use the following sensors: *Accelerometer, Gyroscope, Light, Location* and the *Magnetic Field*. In Fig 1 we can observe the measured data for each of those sensors at two different levels of granulation. First we can have a look at what each sub-graph represents.The first sub-graph represents the three accelerations,

on X, Y and Z, measured by the Accelerometer. The second sub-graph represent the three rotations, on X, Y and Z axis, obtained using the Gyroscope. The next sub-graph contains the light intensity registered by the phone. The 4rt sub-graph is the location of the phone, recorded in Latitude, Longitude and Velocity. In the 5th sub-graph we have the last recorded measurement, more exactly the Magnetic field recorded by the phone in the three axis. The last sub-graph present all the labels and the total time for each recording.

One observation regarding the labels would be that all the recordings for each individual label take the exact amount of time, 300s. This is realized using one functionality of the app, that allows for a timed recording of data. This was done to facilitate the labeling of data.

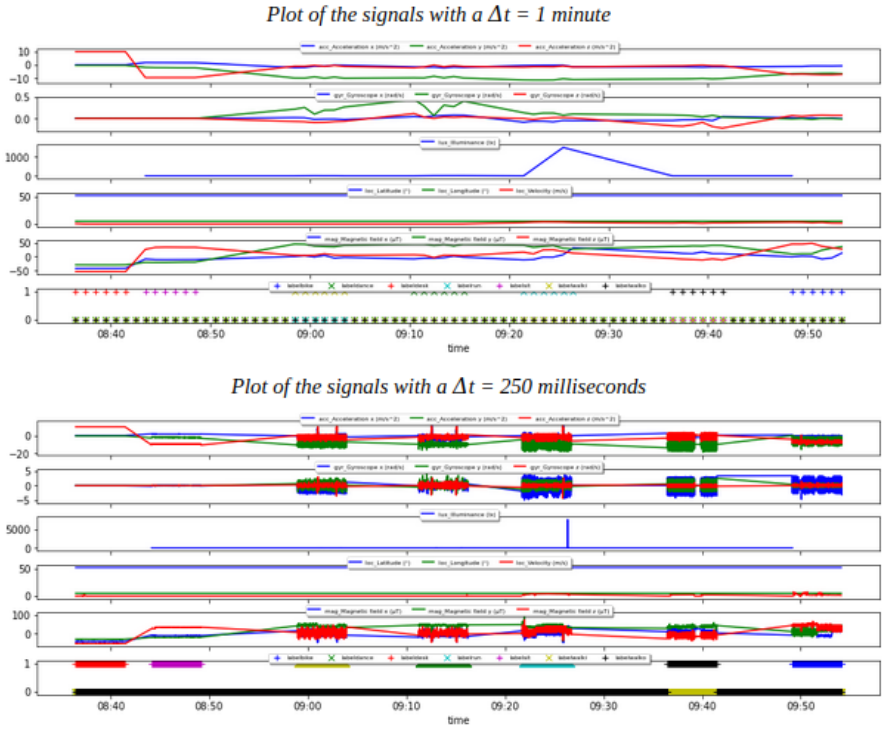

Fig. 1: All measured signals at two different Δt

The last observation concerning Fig 1, would be the difference between the two granulation levels. More exactly for a Δt = 1 minute we can observe that the data is averaged for each minute, this means that a lot of the data is normalized. This also translates in small fluctuations or the influence of outliers not being that impactful. In the second graph is presented the influence of a Δt = 250 ms.

We can clearly observe that in this case the fluctuations in time are much more clear than in the first graph, but we can also see that the outliers have a bigger influence on the data. As a conclusion we can observe that as the granulation level increases the outliers will become less relevant, but also, the fluctuations in data will become smaller and smaller, making it harder to distinguish between different activities.

Q2. In Fig 2 we can observe the difference between the two data sets. In the first plot we can observe the data obtained from the crowdsignals, while in the second plot we can see the data that we collected using our smartphones. At a first glance we could say that the two data sets are quite similar, but after a closer examination we can observe that there are quite some differences between the two.

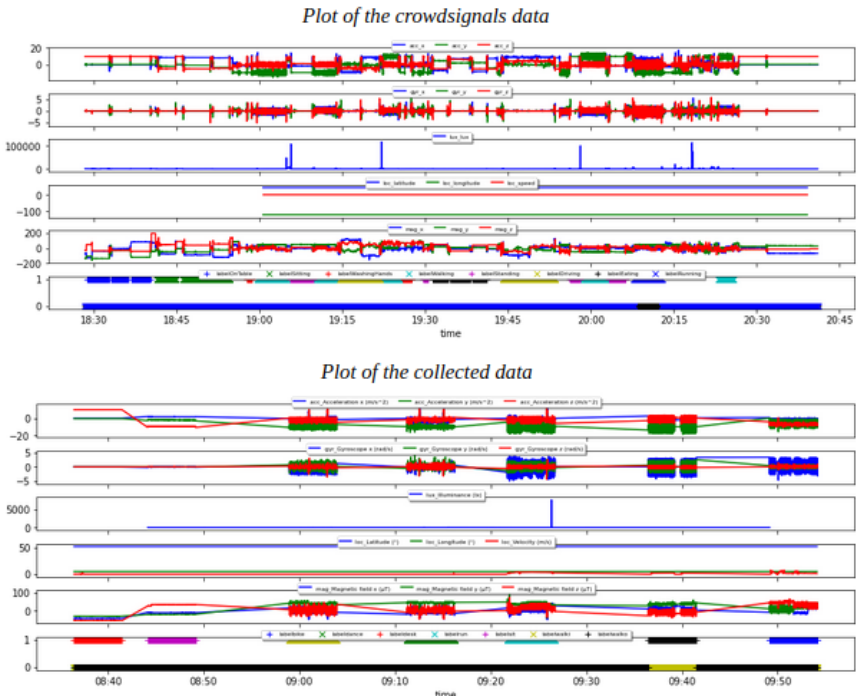

Fig. 2: Plots of the two data sets

The first difference would be the fact that the crowdsignals has a continuous recording of data, while the data collected by our smartphones is not continuous, this is a consequence of the method that was used to collect the data. The other difference would be the amount of data that was recorded, if we have a look at

the crowdsgnals data, we can see that they have over 2 hours of recorded data, while if we combine all the recorded data that we obtained, we get 35 minutes of recorded data. In conclusion we can see that the two data sets present some similarities, like the presence of outliers while recording the intensity of the light, we can clearly see that there are a lot of differences between the two data sets, mainly the recorded time and the way in which the data was recorded and labeled. Because of this we can conclude that the two data sets are quite different from one another.

Another option to verify the difference between the two data sets, would be to compare the data obtained for realizing the same activity. This could be done using the Pandas library, and by doing this we can obtain the mean, standard deviation and count of values for each of the labels. By comparing those values we can determine if there is a difference between the two data sets, more importantly if there is a difference between the sensors used. The advantage of this kind of comparison would be that even if the values are provided from different sensors that have different scales and the distribution might be different, we could just ajust the values of one distribution to be on the same scale as the other one and by doing that we can have a much more easier time comparing the differences between the two data sets.

2.2 Chapter 3

Q2. To investigate the influence of different parameter settings of Chauvenet's criterion, we vary the constant c from a list $[2, 10, 100, 1000, 10000]$. For the parameter *acc_phone_x*, the Chauvenet's criterion cannot find any outlier at all. However, for the parameter *light_phone_lux*, its outliers is getting reduced as the c getting larger at time 19:05. The testing of Mixture model shows that as the tuple c getting larger, the graphs for observed values and for its probability are all getting less dramatic. In addition, Simple Distance approach shows there are some outliers for both parameters. Thus, it is clear to say that the previous methods missed some outliers indeed. The local outlier factor shows that for different k the shape of the data points also changed to be less dramatic, and it indeed shows that the changes in values for the outliers.

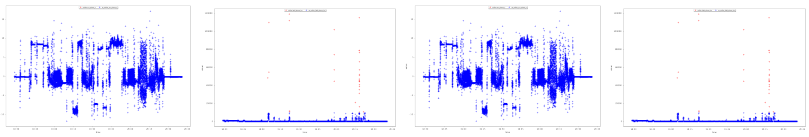

Fig. 3: Chau figures

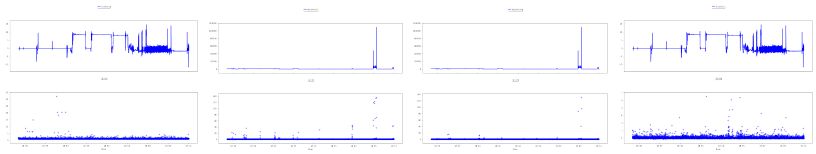

Fig. 4: LOF figures

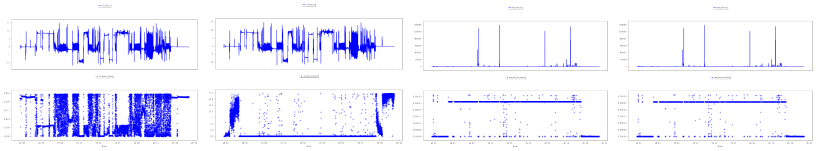

Fig. 5: Mixture figures

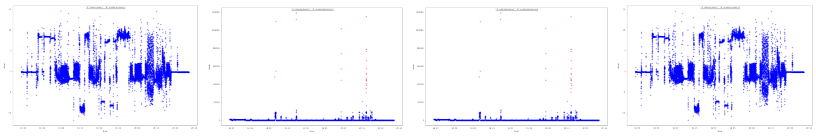

Fig. 6: Simple Distance figures

Q3. We use a model-based approach to impute the heart rate.

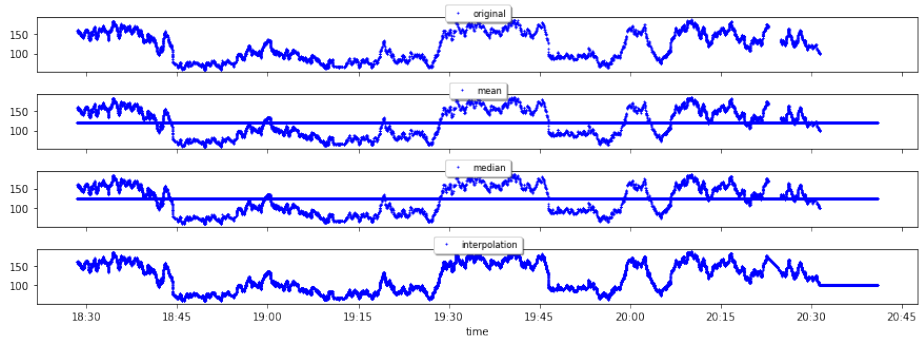

Fig. 7: Imputed heart rate

2.3 Chapter 4

Q1. The entropy plot shows that there exist certain patterns between the features with highest amplitudes. In addition to that, the individual labels, such as

running and walking, suggest that even for the same activity the amplitude is also changeable. Since running and walking are similar activities, they have the similar mean, std, max and min values for the maximum frequency and power spectral entropy. But they have very different frequency signal weighted average.

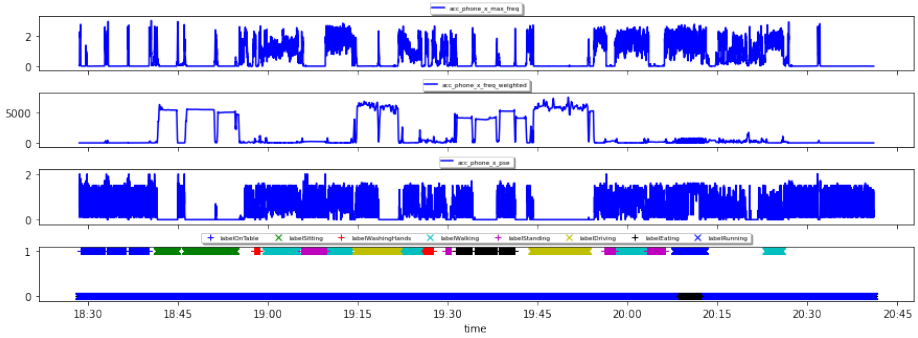

Fig. 8: Frequency

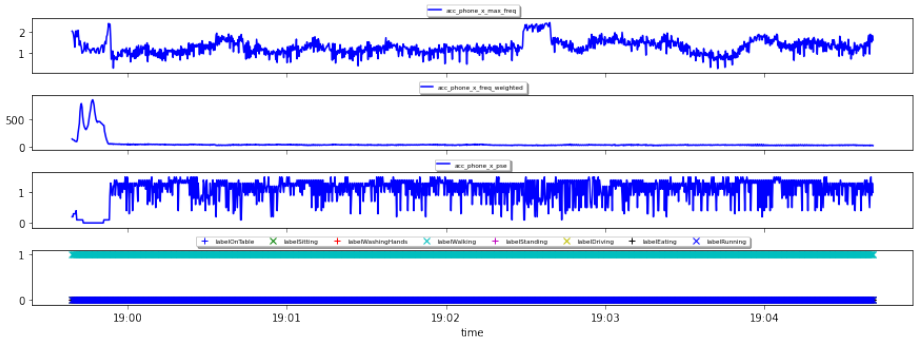

Fig. 9: Pattern of walking

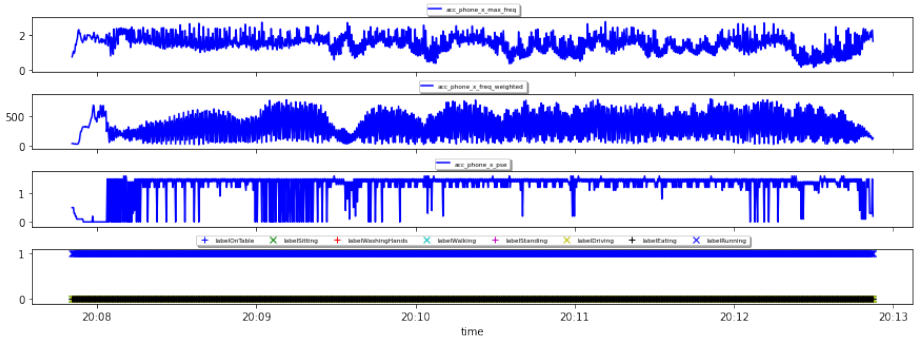

Fig. 10: Pattern of running

Q2. To study the usefulness of different metrics to time domain and frequency domain, we implement each domain 2 metrics. For Time domain, we implement *sum* and *count* as metric.

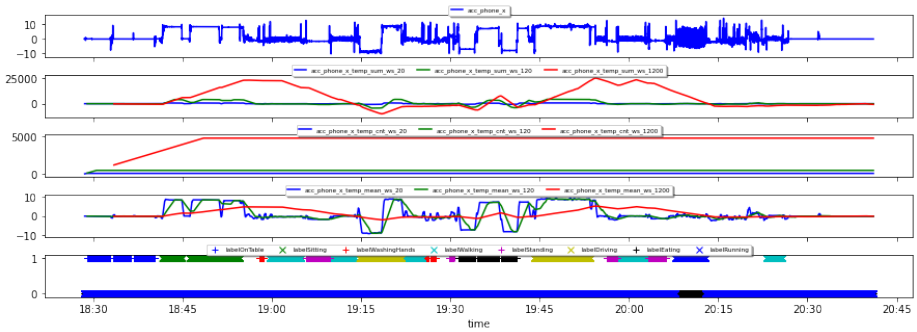

Fig. 11: Different metrics for Time domain

According to above results, we can notice that compared to mean metric, count and sum metric hardly show the pattern of data, which lost some important information. Hence, sum and count metric are not useful enough. For Frequency domain, we implement *minimum* and *variance* as metric. We override functions in FrequencyAbstraction.py.

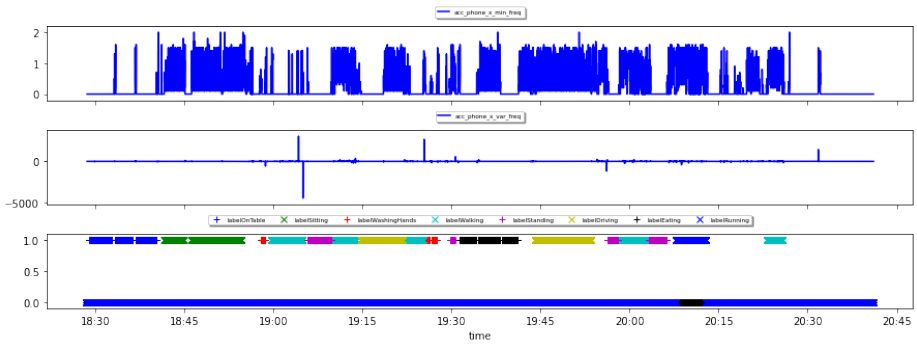

Fig. 12: Different metrics for Frequency domain

As we can see, the minimum and variance metrics neither can present data properly. Therefore, they are not useful.