University of Helsinki
Data Science for the Internet of Things (DS4IoT) - Spring 2024

Exercise 4

Due on 15th April 2024 by 23:59, Helsinki time.

Instruction: All course participants are requested to submit their solutions (in English) through Moodle by the due date. Please take into account the following guidelines about including AI-generated text: https://studies.helsinki.fi/instructions/article/using-ai-support-learning

Submission: You can submit your homework using one of the two following options:

- *Option 1:* submit both the report of non-programming tasks and the solution of the programming tasks in Jupyter Notebook format (*.ipynb*).
- *Option 2:*
  - Submit the report of non-programming tasks in PDF file format.
  - Submit the solutions and the source code of the programming tasks in Python using Jupyter Notebook format (*.ipynb*).

Use the following format for naming the files: *[last name_first name]_[your file name], (i.e., Nguyen_Ngoc_Exercise 1 solution.ipynb).*

Assessment: Participants are encouraged to review course materials to answer the problems and in some cases write computer programs to derive solutions. In all the exercises, do not just give the answers, but also justify, derive and contextualise the answers in Data Science for the Internet of Things contexts. The assessment parameters for each of the tasks are: solution (25%), justification and derivation (50%), contextualisation (25%).

1 Non-programming Tasks

1.1 Learning diary (Compulsory, 4 pts.)

Choose two concepts from the past two lectures (one from Lecture 7 and one from Lecture 8) and describe **what** they mean and **why** they are relevant for the Data Science for the Internet of Things. For full points give an example (or examples) of the concept in a Data Science for the Internet of Things context or otherwise illustrate their importance. Do not copy the definitions from Wikipedia or other sources, but explain the concepts in your own words and relate them with Data Science for the Internet of Things. Use about 100–150 words to describe each of the concepts.

1.2 Features and Data Modelling (4 pts)

Figure 1 shows examples of everyday plastic objects and a time series of light intensity measurements taken from the objects. Assume your task is to develop a data model that uses features extracted from the light measurements to classify objects based on their plastic type.

(a) What frame length would be suitable for the measurements and why?

(b) Describe 5 features that would be relevant for classifying the objects. Justify and contextualise each of the selected features. In which context(s) they will work better?

(c) Describe how you can implement a template-based (kNN) classifier for classifying the objects. Which similarity would you use and why?

(d) Would feature scaling affect the design of the template-based classifier in any way? Why / why not?

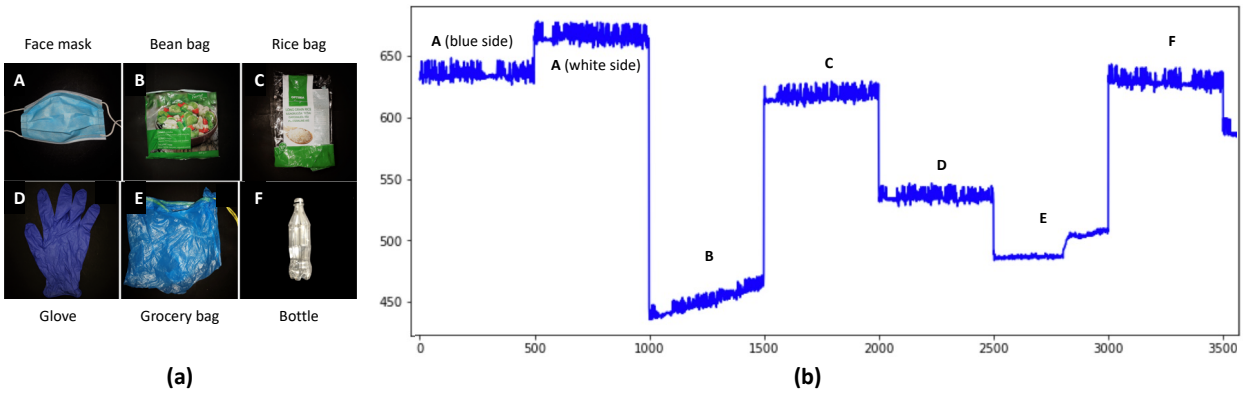

Figure 1: Plastic objects and light intensity.

1.3 Clustering (4 pts)

Consider the applications described below. Which type of clustering method(s) would be best suited for the applications and why?

1. Traffic modeling that attempts to identify areas with high traffic using PM and CO_2 sensors.
2. Accelerometer, gyroscope, luminosity, audio and heart rate data from wearables to identify patterns affecting sleep quality.
3. Thermal camera images for detecting hidden surveillance devices.
4. Speaker recognition from voice input in smart homes.

2 Programming Tasks

2.1 Feature Extraction (8 pts)

The file HR.csv contains heart rate measurements of a user while performing different everyday activities. The measurements were collected using two sensors placed at different parts of the body: (i) at the chest and (ii) at the wrist. You can access the dataset in Moodle.

1. Plot the heart rate measurements and observe their variation over time. How many different activities can you identify? For full points include the period where the activities occur (HINT: Use change point detection to identify the boundaries. Python has a library called *ruptures* that includes a ready-function to make that: *ruptures.detection.pelt.Pelt*).

2. Use the measurements collected at the chest and extract the following features: mean, median, min, max, and standard deviation. Perform the analysis in frames of 250 seconds, with 50% overlap and a frequency of 1Hz. For the preprocessing use moving average with a window size = 50 seconds. HINT: You can use the following code as reference to extract frames from the measurements:

```
frame_size = $size_of_frames$
overlap_50 = frame_size/2
window_size = $size_of_windows_for_preprocessing$
signal = data_frame[$sensor_name$]
signal_feature = []
for i in range(0, len(signal), overlap_50):
    signal_frame = signal[i:i+frame_size]
    signal_preprocessed = preprocessing_signal(signal_frame, window_size)
    extracted = extract_feature(signal_preprocessed, $type_of_feature$)
    signal_feature.append(extracted)
plt.plot(signal_feature)
```

3. Plot the extracted features. Which features most and least representative for identifying the different activities? Why? How many different activities can you identify with the most representative feature?

4. Use the measurements collected at the wrist and extract the feature you chose as the most representative for measurements at the chest. Plot the extracted feature. Can you identify the same number of activities as in step 3? Elaborate your answer to get full points. NOTE: You should apply the same considerations of framing and preprocessing for the samples collected from the chest and from the wrist.