COMP6246 Machine Learning Technologies Coursework 2025/26

Effort, deliverables and deadlines

Expected Effort: 75 hours (10 days full time)

Deliverable	Deadline	Feedback	Weight
Machine learning pipelines implementation 1. Source_code.ipynb 2. Requirements.txt The deliverable is used to provide evidence of practical work.	Week 11 - 9 December 2025, Tue @ 4pm	Week 16	15%
Final report to be submitted as Report.pdf Top scoring final reports will characterize the use case problem and data, and connect these characteristics to attributes of the chosen pre-processing, feature selection, dimensionality reduction for the **three** machine learning algorithms. They will provide a rigorous evaluation of the implementation and justify chosen designs in the context of other algorithm options available (from course text or wider literature).	Week 11 - 9 December 2025, Tue @ 4pm	Week 16	35%

Introduction

The goal of the assignment is to assess how you put in practice a Machine Learning Project Checklist (Appendix A of the course's textbook). To align to the expected effort of the coursework, we provide you some steps and simplify others:
1. Problem Frame: We provide it
2. Get the data: We provide you with the dataset (Test and Training)
3. Exploration and Preparation: Your full responsibility
4. Shortlist Models: Instead of shortlisting from everything in the module, we ask you to choose 3 and compare them
5. Fine-Tune: Your responsibility
6. Present your solution: A workplace-style report

Problem Frame

You have been engaged by SO-Lively is a health-tech startup focused on developing AI-driven solutions for personal wellness, fitness monitoring, and eldercare support.

Define the objective in business terms. How will your solution be used?

Develop a machine learning-based Human Activity Recognition (HAR) system to accurately classify physical activities (e.g., walking, sitting, standing) from sensor data from wearable devices proprietary to the company. The primary goal is to integrate this model into our smart fitness and healthcare monitoring applications to deliver personalized insights and improve safety and engagement.

More technically, given a time series of readings from wearable devices on a human, return a partition of the time series where each sub-series is labelled with the activity the human was doing during that time.

What are the current solutions/workarounds (if any)?

HAR is a classical problem in Machine Learning, however no solution has been tested with the SO-Lively wearable devices and the company wants to know what solutions can be used.

How should you frame this problem (supervised/unsupervised, online/offline, etc.)?

We want to assess your use of each type of ML algorithm covered in the module, so you will implement:

- One Unsupervised: Choose between K-means and GMM

- One Classic Supervised: Choose between Random Forest and SVM

- One Deep Learning: Choose between CNN and RNN

How should performance be measured? What is the minimum performance to reach the business objective?

Precision per class >= 75 %; Recall per class >= 50%

What are comparable problems? Can you reuse experience or tools?

The problem is comparable to transport mode recognition that you learned in the use case lectures (how convenient....)

Is human expertise available?
No.

Dataset

The dataset and ground truth labels are provided on the module wiki page (link to be updated). The dataset contains measurements taken in lab conditions from subjects wearing

a SO-Lively device on their back and a sensor on their thigh and doing a number of activity the following:

1. timestamp: date and time of recorded sample
2. back_x: acceleration of back sensor in x-direction (down) in the unit g
3. back_y: acceleration of back sensor in y-direction (left) in the unit g
4. back_z: acceleration of back sensor in z-direction (forward) in the unit g
5. thigh_x: acceleration of thigh sensor in x-direction (down) in the unit g
6. thigh_y: acceleration of thigh sensor in y-direction (right) in the unit g
7. thigh_z: acceleration of thigh sensor in z-direction (backward) in the unit g
8. label: code of the activity being performed, as per the list below:

1: walking   **2**: running   **3**: shuffling   **4**: stairs (ascending)   **5**: stairs (descending)   **6**: standing   **7**: sitting   **8**: lying   **13**: cycling (sit)   **14**: cycling (stand) **130**: cycling (sit, inactive) **140**: cycling (stand, inactive)

Colleagues have already set apart data from 4 users (~20%) to use as a test set. You will find it on the folder "test_set"

Note: The dataset is based on existing research data for the HAR problem [1], but we have modified it and added some synthetic data to adapt it to the learning objectives of this assessment.

Report

The final report comprises **35 out of 50 marks**. Imagine is the report you will present to your line manager about your work. It MUST have the following five sections:

1. **Data Pre-processing and exploration (1 page max):**
   a. Report the class balance of the whole dataset.
   b. Write a function to drop data labelled with any type of cycling, the company is not currently interested in them. Apply it to the dataset and report the new dataset size.
   c. Write a function to merge the stairs (ascending) and stairs (descending) labels in a single class "stairs" with code 9, as the company wants to consider them as a single action. Apply it to the dataset
   d. Use appropriate sampling and visualisations to report and interpret data patterns of the "Walking" class in the dataset.

2. **Data Cleaning and Preparation (2 page max including figures or tables):**
   a. Colleagues that did Data Collection warn you there might be issues with Subject S007's sensor: analyse and report the data quality of that subset. You only need to report the issues you find, no need to report the whole exploration, also no need to have all charts in your implementation, only add those that allowed you to spot the quality problem. Assume the other subjects data is good enough (sadly we cannot have the whole coursework about exploration ^_^)

b. Based on the results of (a), Choose and implement an imputation (or deletion) strategy for low quality data in S007. Report and justify your choice and the effect on dataset size (if any).

c. Write a function that receives the dataset and outputs sliding windows of 2 seconds size, with an overlap of 1 second (0-2s, 1-3s, 2-4s...etc). Report the number of data points with this window size.

   (1 **page max**)

3. **Pipelines (2 page max each):** For each of the three machine learning algorithms you chose:

   a. **Baseline:** Establish a baseline with an initial set of 10 features (for feature – based algorithms) and a minimum number of layers for NN algorithms. Implement and report any additional transformation required for the algorithm you need. References [1], [2], [3] and [4] provide a large number of features and pointers to other papers. Note our dataset contains only raw data acceleration, while some features in the literature require the decomposition of the raw data into gravity and movement components as described in reference [6]. To estimate the movement component from raw data, you need to use the ENMO formula: The Euclidean Norm of the acceleration vector minus one. (see reference [6] for a full description)

   b. **Fine-Tuning**: Improve over the baseline you established by fine-tuning the algorithm (the fine tuning differs according to the type of algorithm). For feature-based algorithms you can use at most 20 features (we know this is suboptimal, but the report becomes too long if we ask you for all features). In general we expect you to use appropriate techniques for fine-tuning according to the algorithm you chose.

   c. **Measure training and generalisation error.** Use Cross-Validation where appropriate.

   d. **Characterise errors**: What type of errors the fine-tuned model makes?

4. **Conclusion (1 page max):** Compare the three pipelines according to their performance and error types and state the one model you would recommend for deployment in production, justifying your choice.

The report PDF document should be between 7 and 10 pages long. The 10-page limit is not a target to aim for, and shorter reports that present information concisely are better - find your perfect balance. Use tables and figures to show data, and highlight key information clearly such as the main findings. Use any document style (e.g. reference style) as long as it's clear and easy to read. Reports over 10 pages long may incur a 5-mark penalty for demonstrating a poor ability to summarize key information.

Implementation

15 out of the 50 marks.
The Implementation of all 4 sections of the report in a Python based Jupyter Notebook:

1. **Data Pre-processing**: Code that produces the figures or tables you reported **(1 mark)**
2. **Data Exploration and Preparation**: Implementation of transformation functions, windowing and imputation. **(4 marks)**

3. **Pipelines**: For each pipeline, implementation of training, fine tuning, error characterisation and generalisation error **(9 marks, 3 marks each)**
4. **Conclusion:** Code to produce visualisation for comparing the models you trained. **(1 mark)**

FAQs

Can I use external data?

No. Use only training and test data from the assignment ZIP file. We crafted it specifically for this assignment.

What software can I use for the implementation?

The code must be implemented using Python and submitted as a Jupyter notebook. You can make use of lab sessions and materials to help you with the coursework. You can also use any python libraries included in the virtual environment such as numpy, sklearn, nltk, etc. that help

Can I use time series specific libraries and classifiers such as sktime?

No because they are not covered in the module, we need to assess what was covered.

Can I use ensemble methods? That's one of the steps described on Appendix A of the book.

Indeed, in a real world setup one should try an ensemble method combining the three pipelines to get a full system, however, we realised that asking for that would make the coursework too long. If you do it you won't get penalised, but won't get any marks either.

Plagiarism

Both the machine learning algorithm implementation and the final report need to be the student's own work unless mentioned otherwise.

For the machine learning algorithm, you can reuse example code from the course textbook or lab materials, but anything else needs to be clearly acknowledged in the report and when submitting.

You are allowed to use ideas and strategies reported in academic papers in addition to those in the references, as long as you implement these strategies yourselves and acknowledge the papers in your report. In case of doubt, feel free to ask! This is important as any violations, deliberate or otherwise, will be automatically reported to the Academic Integrity Officer.

Late submissions

Late submissions will be penalised according to University policy.

AI Usage

You can use AI assistance to generate and debug code for the cells of your notebook, like for example, the visualisations you need for exploration. For the report you can use it to check syntax/grammar and a translation aid, but do not use it for writing sections of the report or suggesting outlines.   See the accompanying GenAI form for further details.

Marking scheme

Data Pre-Processing (3 marks)
   a. 0,5 mark
   b. 0,5 mark
   c. 0,5 mark
   d. 1,5 mark

Data Preparation (5 marks)
 Correct calculation and reporting of each of the steps asked for
   a. Correctly identifies all quality issues (Can't tell you the value of each because it reveals what they are ^_^): 3 marks
   b. 1 mark
   c. 1 mark

Pipelines (8 marks each = 24 marks)
For each pipeline
   • Correct baseline establishing. (2 marks)
   • Fine tuning of the pipeline improves at least 10% over baseline, correctly applying appropriate methods (4 marks)
   • Correct measure of training and generalisation error (1 mark)
   • Complete error characterisation and interpretation (1 marks)

Conclusion (3 marks)
(0) There is no conclusion
(2 marks) The conclusion summarizes the methods and evaluation results but does not describe any insights gained, or ideas for future improvement.
(3 marks) The conclusion summarizes the method and evaluation results and includes information on insights gained and ideas for future improvement.

Implementation (15 marks)

Clarity (5 marks) - the code clearly describes the various steps and commentaries facilitate understanding.

Correctness (10 marks) - You start with 10 marks, every cell that fails or gives a result/figure inconsistent with the report incurs in a –1 penalty, down to 0 marks.

References

[1] Logacjov, A., Bach, K., Kongsvold, A., Bårdstu, H. B., & Mork, P. J. (2021). HARTH: A Human Activity Recognition Dataset for Machine Learning. *Sensors*, *21*(23), 7853. https://doi.org/10.3390/s2123785

[2] O. D. Lara and M. A. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors," in *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1192-1209

[3] Silvia González, Javier Sedano, José R. Villar, Emilio Corchado, Álvaro Herrero, Bruno Baruque, Features and models for human activity recognition, Neurocomputing, Volume 167, 2015.

[4] Rosati, S., Balestra, G., & Knaflitz, M. (2018). Comparison of Different Sets of Features for Human Activity Recognition by Wearable Sensors. *Sensors*, *18*(12)

[5] Kaixuan Chen, Dalin Zhang, Lina Yao, Bin Guo, Zhiwen Yu, and Yunhao Liu. 2021. Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges, and Opportunities. ACM Comput. Surv. 54,

[6] an Hees, V.T. et al. (2013) 'Separating Movement and Gravity Components in an Acceleration Signal and Implications for the Assessment of Human Daily Physical Activity', *PLoS ONE*. Edited by M. Müller, 8, p. e61691. doi:10.1371/journal.pone.0061691.