Slide set 5 assignments

NOTE : you can merge the subprojects analyses into the same document, zip can contain all the exercise codes, there is no need to do separate zip packages, just one.
NOTE2: The classifier/predictor codes are expected to have training and testing phases and thus the data must be divided (close to) 80% / 20% ratio for training and testing.

NOTE3: DeepAR exercises are mainly omitted, due to numerous reasons, starting from internal problems, package installation problems, incomplete code, and some DeepAR:s are using LSTM/GRU models. There are few easy DeepAR exercises at the end with very severe precautions.

Exercise 1

Make the example from the link:

https://stackabuse.com/time-series-prediction-using-lstm-with-pytorch-in-python/

and improve the result by modifying the model architecture. It is also possible to include nn.Linear layers before the LSTM layers to improve preprocessing. Include figures of the prediction in your analysis (+zipped code).

NOTE: there is an error on the code, move the initialization

```
model.hidden_cell = (torch.zeros(1, 1, model.hidden_layer_size),
                     torch.zeros(1, 1, model.hidden_layer_size))
```

in the loop to some place before before the loop.

Max 3 p.

Exercise 2

Run the code from link:

https://charlieoneill11.github.io/charlieoneill/python/lstm/pytorch/2022/01/14/lstm2.html

Verbally:

- Explain exactly what the multivariate data represents in input and what the LSTM network tries to predict?
- Does this dataset have any internal correlation?

Include the MSE error to test measurement. Improve the multivariate correlation by changing the parameters, architecture and show the improved result in MSE error (original+improved) and figures (original + improved) and zipped code

Max 3 p.

Exercise 3

Make the example from the link:

https://stackabuse.com/time-series-prediction-using-lstm-with-pytorch-in-python/

Do not change the architecture but change the LSTM layers first to RNN and then to GRU (bidirectional=False). Compute summed errors from each test sequences. Include figures from all three predictions and sums of test errors in your analysis and conclude which one is the best. (+zipped code)

Max 8 p.

Exercise 4a

Do the example presented in
https://www.statology.org/step-by-step-guide-to-linear-regression-in-python/ . Return your analysis with images and zipped code.

Max 2 p.

Exercise 4b

Modify the exercise 2 for this new dataset in link 4a. Compare the results and dataset from 4a to modified exercise 2 predictor results. Is the linear regressor better than the LSTM predictor? Return your analysis with images and zipped code.

Max 7 p.

Exercise 5

Load the code "Anomaly_detection_autoencoder.py" and the related data ECG5000.zip as well as the arff2pandas-master.zip. Extract the packages in the same folder where the anomaly detection python code is.

Run the code and verify that it is working. The running might take time.
Can you improve the already high classification and what you have to change?

Verbally: what the autoencoder is doing in this code and what is its purpose?

Verbally:
what is the best strategy to improve the code that is slow in the first place?
In order to save time is there any way to monitor the improvements on the fly?

Max 5 p.

NOTE4: you can ONLY do the 6a-b if you have Python 3.10 installed. It is currently not working with Python 3.11+ due to pytorch-forecast package installation problems.

Exercise 6a

Copy and run the code (for example in Jupyter Notebook) from:

https://pytorch-forecasting.readthedocs.io/en/stable/tutorials/deepar.html

using the default dataset. Modify the code to show mean square error per sample (as in slideset 2c) between the prediction and the test data.

You can do the MSE by yourself or use some python package based MSE.

Show the error to all individual test sequences (i.e. part of the loop).

Max 3 p.

Exercise 6b

Improve the previous code to get as small as possible MSE for all test sequences. NOTE: there is a hint at the bottom of the tutorial how to improve the code. You can also change other parameters.

Max 2 p.