Coding exercise:

The following will test your knowledge of python and ability to train a machine learning model. Before the exercise please install a conda environment as described below. **This may take several hours depending on your connection and will download several Gb of data**.

Installation instructions:

To enable equal comparison please install a new anaconda environment. First install conda on your system first. https://docs.conda.io/en/latest/.

Once conda is installed set up an environment to run the exercise. To set up the correct environment use the command:

```
        conda create --name inforideas  pytorch spyder pandas scipy matplotlib cpuonly
-c pytorch -c conda-forge
```

This will take several minutes to work out the correct dependencies and then you will be asked:

```
        Proceed ([y]/n)?
```

input y and press enter. This can take up to an hour to install. Once the environment has been installed you will notice that the terminal prompt has now changed to:

```
        (base) [user]@[machine]:~/$
```

where '(base)' is now used to denote the base/normal environment. To switch to the new environment for the exercise which you have created above use:

```
        conda activate inforideas
```

You will then see that the command prompt has changed to:

```
        (inforideas) [user]@[machine]:~/$
```

Start the spyder IDE by typing:

```
        spyder
```

From within the IDE type in the following:

```
import numpy as np
import torch
import torch.nn as nn
import numpy as np
from torch.autograd import Variable
from torch import optim
```

If these result in no errors you are ready to go.

Coding exercise.

i) Unzip the attached zip file and open the file load_data.py in spyder. You may need to change the directories to your local paths and then run the script. This will make available to you two variables ynp and Xnp. Ynp is the target, Xnp are the input variables. Ynp is ordered in time and so is a time series.

Xnp consists of 15 variables. Please select two of these variables for modelling of y. Explain your choice. (do not get delayed on this part, after 30 minutes move to the next part and return at the end if you have time).

ii) Given y and the reduced set of X (ie 2 variables from above), create the following model in pytorch:

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \beta_1 x_t^1 + \beta_2 x_t^2$$

where $(\alpha_1, \alpha_2, \beta_1, \beta_2)$ are the parameters of the model and y_{t-1}, y_{t-2} are delayed versions of y (i.e. regressors). x_t^1, x_t^2 are the two input variables you selected in part i).

Train this model to minimise the prediction MSE using a gradient descent based algorithm (ex:SGD or similar). Split the data as you see fit. We are looking for the approach you are using and not the model performance you achieve. Please explain each of your choices or experiments with short but clear comments in your code.

Deliverable:

Please deliver a single python script of ~100 lines with short comments explaining your choices.