MY475: Formative Assignment

WT 2025

The goals of this assignment are to 1) check your understanding of key mathematical concepts, 2) consolidate your understanding of building neural networks in PyTorch, and 3) submit a prediction model in the correct format.

Remember that this assignment is *formative* -- the mark is not what is important, the feedback is!

Exercise 1

a. Consider the following objects:

$$\mathbf{A} = \begin{bmatrix} 3 & 7 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 1 & 3 \\ 2 & 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 0 & 1 & 3 & 5 \\ 1 & 0 & 0 & 3 \end{bmatrix}.$$

Calculate the following:

i. $\mathbf{AB}$

ii. $\mathbf{BC}$

iii. $\mathbf{(AB)C}$

iv. $\mathbf{A(BC)}$

v. $\mathbf{BA}$

b. Consider the model:

$$y_{\text{Pred.}} = \sigma(wX + b)$$

where

$$\mathbf{X} = \begin{bmatrix} 3 & 1 & 4 \\ 2 & 2 & 7 \\ 1 & 0 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} 31 \\ 39 \\ 8 \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}, \quad b = 0, \quad \sigma(z) = \text{ReLU}(z).$$

i. Suppose we wanted to backpropagate the squared error from this model. Draw the corresponding computation graph. You may use a package like `graphviz` or draw it by hand and insert it as a scan/image.

ii. What is the partial derivative of the loss function, with respect to w_1 (i.e. $\frac{\partial l}{\partial w_1}$) -- state your answer as a general equation, not for any specific example in $\mathbf{X}$

iii. Calculate the loss for the first example in $\mathbf{X}$ (think carefully about the dimensions).

iv. Suppose $\mathbf{X}$ is our batch. Calulate the value of $\mathbf{w}$ after one backwards pass of this batch, using mini-batch gradient descent and a learning rate of 0.01.

v. A colleague suggests that this model is always superior to a linear regression, because deep networks can approximate non-linearities in the data generating process. Do you agree with your colleague, and why (not)?

You do not need to show your workings.

Exercise 2

In this exercise, suppose you are a data scientist working for a city's government. They are interested in boosting levels of cycling, and want to make sure enough bikes are available for the public at any given time. You have been tasked with building a neural network model to predict the total number of bike rentals in the city, based on historical data. We have provided you with data recorded at the hour-level, which includes variables such as the weather, temperature, and time of day.

Your model should be designed to predict the total number of city bike rentals, on an hourly basis (i.e. each observation in the test/train data is the total number of bikes being rented for a given hour). You should therefore focus on predicting the variable 'cnt' -- which is the total count of rentals.

Further information on this dataset, including variable descriptions, can be found on the UCI Machine Learning Repository here (https://archive.ics.uci.edu/dataset/275/bike+sharing+dataset). You should read this description carefully, and be mindful of your choice of variables etc.

We have separated out a random test set from this data. Your goal is to train as predictive a model as possible using the provided formative_train.csv file. Once you are happy with your model, you should then predict the values for the formative_test_data.csv file. Note this file does not contain values for the outcome of interest.

Specifically, you should:

- Define a neural network model class called `my475_mod` that is capable of taking in data, passing it through a neural network, and outputting a prediction.
    - You may rely on functions outside this class definition to help you build/train the model
- Train an instance of that model on the provided training data and save your final trained model checkpoint to a file (this file should be named " <CANDIDATE_NUMER>.pt")

- Pass our test data--`formative_test.csv`--through your trained model and save the predictions as a .csv file. The file should contain a single column of predictions, with each row corresponding to a row in the test data. The first row should contain the header "prediction". Save this file as "<CANDIDATE_NUMER>.csv"

We will assess both the quality of your code (i.e. the legibility of your definitions, the soundness of the architecture etc.) and the performance of your model on the test set.

Submission requirements

You should submit separately:

1. Your answers to exercise 1 as an HTML file
2. A zipped directory containing your work for exercise 2:
    - Your model class definition as a `.py` script -- please omit any training/testing code
    - Your test predictions in `.csv` format
    - Your model checkpoint file as a `.pt` file

You should name each of these files with your candidate number i.e. `1234567.html` *and* `123456.zip` *containing* `1234567.py`, `1234567.csv`, `1234567.pt`

Please note: we will not make allowances for discrepancies between your model structure and the test script (e.g. the class name must be `my475_mod`).

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js