Assignment 3
Total Marks: 24
Deadline: April 21, 11:59 PM

Create your own repo and write code in a similar fashion as you did for earlier assignments. We are not providing any sample code this time. Any commits after the deadline will not be acceptable.

If you have any questions, please ask them on MS Teams below this post. Please copy Mrinal and Nidhin on Teams.

1. Implement Unregularised Logistic regression for 2-class problem using:
    a. Update rules mentioned in class slides with gradient descent [Slide 26 from https://nipunbatra.github.io/ml2021/lectures/logistic-regression.pdf] [1 mark]
    b. Use Jax/Autograd to automatically compute gradient and solve with gradient descent [1 mark]
    c. Using breast cancer dataset and K=3 folds present the overall accuracy. [1 mark]
    d. Plot decision boundary for 2d input data where you can choose any two pairs of features to visualise [1 mark]
2. Implement L1 and L2 regularised logistic regression for 2-class problem using:
    a. Jax/Autograd [1 mark]
    b. Using nested cross-validation find the optimum lambda penalty term for L2 and L1 regularisation. From the L1 regularisation, can you infer the “more” important features? [2 marks]
3. Implement K-class Logistic regression using:
    a. Update rules in slides [Slide 38 from https://nipunbatra.github.io/ml2021/lectures/logistic-regression.pdf] [1 mark]
    b. Jax/Autograd [1 mark]
    c. Using Digits dataset and stratified (K=4 folds) visualise the confusion matrix and present overall accuracy. Which two digits get the most confused? Which is the easiest digit to predict? [3 mark]
    d. Use PCA (as blackbox) from sklearn and project the digit data to 2 dimensions and make a scatter plot with different colours showing the different digits. What can you infer from this plot? [1 mark]
4. What is time and space complexity for Logistic regression learning and prediction? [1 mark]
5. Create a fully connected NN (MLP) where the input is X, y, [n1, n2, …, nh] where ni is the number of neurons in i^th hidden layer, [g1, g2, …, gh] where gi in {‘relu’, ‘identity’, ‘sigmoid’} are the activations for i^th layer. You should use Jax for backpropagation. You should write the forward pass yourself. [3 marks]
6. Test NN code for simple classification (Digits dataset) and regression dataset (Boston housing) both using 3-fold CV. You can choose the number of layers and activations of your choice. [3 marks]

7. In this question, you have to compare the performance of: transfer learning, VGG1, and VGG1 with data augmentation on an image dataset. Refer this article: https://machinelearningmastery.com/how-to-develop-a-convolutional-neural-network-to-classify-photos-of-dogs-and-cats/ You do not need to write your own code. Reuse the code from the post. You need to create the dataset on your own based on your first and last name. For instance, my name is “**N**ipun **B**atra”. I will thus choose a data set of my liking: nightingale v/s bear. You can refer to: https://course.fast.ai/videos/?lesson=3 or https://course.fast.ai/images or plainly download 40 images of both classes (total 80 images). Of these 40 images of each class, we will use 30 for training and 10 for testing. You may choose any two objects for the dataset. For example, I could have created: Novak Djokovic v/s Boris Becker or Notebook v/s Ballpen. The absolute value of accuracy you obtain is immaterial. [4 marks]

Datasets:

1. Digits dataset
   https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_digits.html#sklearn.datasets.load_digits
2. Breast cancer dataset
   https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_breast_cancer.html#sklearn.datasets.load_breast_cancer
3. Boston housing dataset
   https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_boston.html#sklearn.datasets.load_boston