Machine Learning and Computational Physics
Fall 2021

Assignment 2

Due: 09/22/2021, 11:59:59 PM PDT

Selecting the Correct Regularization Parameter

In this assignment, you will train a deep network to approximate a scalar function. The primary objective is to (a) learn how to select the optimal value of the regularization parameter for a problem, and (b) to understand the effect of regularization on the prediction of a network.

1. Consider the function developed in Assignment 1 to define a neural network. Extend it so that:
   (a) It accepts the regularization parameter (a scalar) as input.
   (b) The `tanh` activation function is used in each hidden layer, with no activation in the output layer.
   (c) L_2 regularization is applied to the weights and biases.
   (d) The is model compiled using the `Adam` optimizer with a learning rate `lr = 1e-3`, using the `mse` loss function and the `mse` metric.

   The following TensorFlow tools might be useful
   - `keras.regularizers.l2()`
   - `keras.optimizers.Adam()`
   - `keras.model.compile()`

2. Create a dataset using the function
$$f(x) = x + \sin(10\pi x) + \omega(x)$$
   where $\omega \sim \mathcal{N}(0, 0.3)$ is an additive Gaussian noise at each evaluation point. Evaluate the function at 125 uniformly spaced points between $[0, 1]$. Use 100 of these points (selected randomly) for training and the remainder for validation (finding the optimal regularization parameter). Sort the values in these data sets in increasing order. Fix a random generator seed (say to 1), to ensure the same split into training and validation data each time your code is run. In a single plot, plot the training and validation sets.

3. Create a network with `width = 15`, `depth = 10` and `input_dim = output_dim = 1`. Consider three values of the regularization parameter, $\{1e-6, 1e-3, 1e-2\}$. Your goal is to find the "optimal" value of the parameter. For each parameter value, train the network 4 times with different weight initializations. Set maximum epochs = 5000, batch size = 25 and shuffling on. You will need to use the `model.fit()` command to train.

   For each value of the regularization parameter:

(a) Save the history of the metric evaluated on training set (`mse`) and the validation set (`val_mse`) for each of the four runs. Compute the average of this history across the four runs. In a single plot (one per regularization regularization parameter) plot the training and validation metrics as a function of epoch number for each run, as well as the average.

(b) Report the average value of training and validation metrics at the final epoch. Based on this value, select the optimal value for the regularization parameter. Justify your selection.

(c) At the completion of each run, evaluate the prediction of the trained network for a set of uniformly spaced $1,000$ points in the interval $[0, 1]$. In a single plot (one per regularization regularization parameter) plot this prediction as a function of the input. There should be four curves in each plot. Explain the effect of regularization on these predictions. Which regularization parameter yields prediction curves that are closest to the noise-free target function.

Instructions:

- You need to submit your work as a single notebook saved as `A2_FirstName_LastName.ipnyb` (for example `A2_Tommy_Trojan.ipnyb`). You can create this notebook locally (on your computer using Jupyter notebook) or on cloud using Google Colab (which we recommend). If you are using Google Colab, then please make sure that you are signed in to your USC Google account before starting. This will make sharing your saved work little easier.
- At the very beginning of your notebook insert a text cell and write your name.
- Make sure that your entire notebook runs successfully on Google Colab before submitting it. It is your responsibility to ensure this.
- Once you finish the assignment save it and share it with `hramaswa@usc.edu`. (If you are using Google Colab, then the notebook will automatically be saved to your Google Drive. Once you locate it in your Google Drive, right click on it and share it with `hramaswa@usc.edu`). While sharing make sure that you enable "editor" option, so that we can run your notebook on our end while grading it.
- Submit you notebook only once. Resubmissions will not be considered.