TÉCNICO LISBOA

Aprendizagem 2022/23

Homework III

Deadline 28/10/2022 23:59 via Fenix as PDF

- Submission Gxxx.PDF in Fenix where xxx is your group number. Please note that it is possible to submit several times on Fenix to prevent last-minute problems. Yet, only the last submission is considered valid
- Use the provided report template. Include your programming code as an Appendix
- Exchange of ideas is encouraged. Yet, if copy is detected after automatic or manual clearance, homework is nullified and IST guidelines apply for content sharers and consumers, irrespectively of the underlying intent
- Please consult the FAQ before posting questions to your faculty hosts

I. Pen-and-paper [12v]

Consider the problem of learning a regression model from 5 univariate observations $\big((0.8),(1),(1.2),(1.4),(1.6)\big)$ with targets (24,20,10,13,12).

1) [5v] Consider the basis function, $\phi_j(x) = x^j$, for performing a 3-order polynomial regression,

$$\hat{z}(x, \mathbf{w}) = \sum_{j=0}^{3} w_j \phi_j(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 \,.$$

Learn the Ridge regression (l_2 regularization) on the transformed data space using the closed form solution with $\lambda = 2$.

Hint: use `numpy` matrix operations (e.g., `linalg.pinv` for inverse) to validate your calculus.

2) [1v] Compute the training RMSE for the learnt regression model.
3) [6v] Consider a multi-layer perceptron characterized by one hidden layer with 2 nodes. Using the activation function $f(x) = e^{0.1x}$ on all units, all weights initialized as 1 (including biases), and the half squared error loss, perform one batch gradient descent update (with learning rate $\eta = 0.1$) for the first three observations (0.8), (1) and (1.2).

II. Programming and critical analysis [8v]

Consider the following three regressors applied on `kin8nm.arff` data (available at the webpage):

- linear regression with Ridge regularization term of 0.1
- two MLPs – MLP_1 and MLP_2 – each with two hidden layers of size 10, hyperbolic tangent function as the activation function of all nodes, a maximum of 500 iterations, and a fixed seed (`random_state=0`). MLP_1 should be parameterized with early stopping while MLP_2 should not consider early stopping. Remaining parameters (e.g., loss function, batch size, regularization term, solver) should be set as default.

Using a 70-30 training-test split with a fixed seed (`random_state=0`):

4) [4v] Compute the MAE of the three regressors: linear regression, MLP_1 and MLP_2.
5) [1.5v] Plot the residues (in absolute value) using two visualizations: boxplots and histograms. *Hint*: consider using `boxplot` and `hist` functions from `matplotlib.pyplot` to this end
6) [1v] How many iterations were required for MLP_1 and MLP_2 to converge?
7) [1.5v] What can be motivating the unexpected differences on the number of iterations? Hypothesize one reason underlying the observed performance differences between the MLPs.

END