EPFL, Autumn 2025
Yoav Zemel
Guillaume Obozinski Ho Yun

Statistical Machine Learning

Exercise sheet 14

Exercise 14.1 In this exercise, we will train a three-layer artificial neural network (i.e., an input layer, a hidden layer, and an output layer) to learn a 1D function. There are many packages/libraries for neural networks in R (as well as in Python). This exercise references the `ANN2` package in R, but feel free to use the package that you prefer in your implementations. Make sure to set a seed at the beginning of your program for reproducibility.

(a) Generate $n = 350$ simulated data $\{(x_1, y_1), \dots (x_n, y_n)\}$ based on the relationship

$$y_i = \sin(x_i)(1 + 0.7\sin(3x_i)) + \epsilon_i,$$

where $x_1, \dots, x_n \overset{\text{iid}}{\sim} \mathcal{N}(0, 4)$, and $\epsilon_1, \dots, \epsilon_n \overset{\text{iid}}{\sim} \mathcal{N}(0, 0.1)$ for $i = 1, \dots, n$. Let $\{(x_1, y_1), \dots (x_{250}, y_{250})\}$ be the training set and $\{(x_{251}, y_{251}), \dots (x_{350}, y_{350})\}$ be the test set. Plot the training set.

(b) Using the `ANN2` package, fit a neural network with 300 hidden units with sigmoid activation to the training set. Using the SGD optimizer, train the network for 3000 epochs and have 10% of the training set to be the validation set. Additionally, set `regression=TRUE`, `loss.type= 'squared'` and leave the remaining parameter settings to be the default. Plot the training and validation curves, plot the prediction curve and calculate the MSE of the test set.

(c) Repeat (b), this time using the 2000 hidden units. Compare the results to those of (b) and comment on any differences.

(d) Repeat (b), this time using the ReLU activation. Compare the results to those of (b), and (c) and comment on any differences.

(e) Repeat (b), this time using the 2000 hidden units with ReLU activation. Compare the results to those of (b), (c), and (d) and comment on any differences.

(f) Let us compare the neural network regression model to kernel ridge regression model. Using the `CVST` package[1], fit a Gaussian ridge regression model with kernel bandwidth $\sigma = 400$ and penalty $\lambda = 0.001$ (or select them using cross validation) to the training set. Plot the prediction curve and calculate the MSE of the test set. Compare the Gaussian ridge regression model to the neural network models of (b), (c), (d), and (e). Which you these models in your opinion has the best fit? Which of these models achieved the lowest test error?

[1] See `http://www.stats.ox.ac.uk/~sejdinov/teaching/atsml19/KRR.html` for an example.

(g) **Bonus.** Create an ensemble model simply by taking the average of the predictions of the neural network models. What is the test error of this model? If this was a classification problem, how would your ensemble model make a final prediction?

(h) **Bonus.** Repeat the exercise under difference settings. Does adding L2 regularization to the neural network models make a difference? Try different noise level ϵ_i and functions.