UL FRI Data Science - Machine Learning for Data Science 1, 2023-2024
Homework 4 report

Artificial Neural Networks

Ivan Nikolov[1]

[1] *in7357@student.uni-lj.si, 63190378*

Implementation

For this homework, I implemented multilayer fully-connected artificial neural networks for classification and regression.

Both types of neural networks inherit the same base class, `ANN`. The network uses sigmoid activation functions in the intermediate layers. The L2 regularization is specified using the parameter λ, In the descendant classes, we define the different activation functions (softmax for classification, and plain linear layer for regression). The losses (cross entropy for classification and MSE for regression), and methods for modifying the matrix shapes. The rest of the code for weight initialization, forward and backward pass and model prediction is identical.

For optimizing the cost function I used the L-BFGS-B method which supports passing calculated gradients from backpropagation.

Numerical verification

In order to check if my gradient calculations are correctly calculated for the cost function, I implemented numerical calculation of gradients.

$$\frac{\partial J}{\partial w_i} \approx \lim_{h \to 0} \frac{f(w_1,\ldots,w_i+h,\ldots,w_n) - f(w_1,\ldots,w_i+h,\ldots)}{2h}$$

I checked the gradients from backpropagation in a classification network with multiple layers with the numerical calculations, and they matched by to the fifth decimal. It is important to note that the MSE loss needs to be divided by 2 in order for the numerical and backpropagation gradients to match. This should not be a concern as we are only dividing by a constant and the direction of the gradients still remains the same.

Housing dataset evaluation

I compared the neural network implementation with Multinomial Logistic regression from the third homework for classification, and a basic Linear regression for the regression task. Before training the models, I standardized the continuous features, such that it has a mean of 0 and a standard deviation of 1. For comparing the model performance, I used log loss for classification and MSE for regression and quantified the uncertainty using bootstrap with 1000 iterations. The evaluation was done with a holdout train/test split with a test size of 30% of the whole dataset.

Table 1. Log. loss by model on Housing3

	Log. loss $\pm$ SE
Multinomial	0.379 ± 0.079
ANN	0.278 ± 0.050

On table 1 we can see that the ANN with one hidden layer with 10 nodes and a λ of 0.001 performed better than the multinomial regression.

Table 2. MSE by model on Housing2r

	MSE $\pm$ SE
Ridge regression	22.432 ± 5.153
ANN	10.112 ± 1.925

ANN performed better for the regression task too and beat ridge regression.

Final dataset

For this part, for optimal parameter finding, I created a holdout validation set that consisted of 20% of the train set. Before training the model, I standardized the dataset based on the values from the training set. Due to time constraints, I manually tested different network architectures and λ values.

Table 3. Log. loss on validation set

	Log. loss $\pm$ SE
ANN [100], $\lambda = 0.001$	0.537 ± 0.007
Multinomial (Baseline)	0.643 ± 0.010

On table 3 we can see that ANN with one layer with 100 nodes performs better than the baseline. I compared different parameters also, but did not log the results and only included the best one.

Typical training time is around 7 minutes on a 3-year-old laptop.