A.7 EARLY STOPPING IN DATA SCARCITY SCENARIOS

Labelled data is expensive in most domains of interest, which results in small data sets or low quality of the labels. We investigate the following experimental set-ups: (1) Reducing the training data set size and (2) Permuting a fraction of the training labels. We train a fully connected network ($[500, 500, 200]$ architecture) on 'MNIST' and 'Fashion-MNIST'. In the experiments, we compare the following measures for stopping the training: i) Stopping at the optimal test accuracy. ii) Fixed stopping after the burn in period. iii) Neural persistence patience criterion. iv) Training loss patience criterion. v) Validation loss patience criterion. For a description of the patience criterion, see Algorithm 2. All measures, except validation loss, include the validation datasets (20%) in the training process to simulate a larger data set when no cross-validation is required. We report the accuracy on the non-reduced, non-permuted test sets. The batch size is 32 training instances. The stopping measures are evaluated every quarter epoch.

Figure A.12 shows the results averaged over 10 runs (the error is the standard deviation). The difference between the top and the bottom panel is the data set and the patience parameters. The x-axis depicts the fraction of the data set, which is warped for better accessibility. In each panel, the left-hand side subplots depict the results of the reduced data set experiment where the right-hand side subplots depict the result of the permutation experiments. The y-axis of the top subplot shows the accuracy on the non-reduced, non-permuted test set. The y-axis of the bottom subplot shows when the stopping criterion was triggered.

We note the following observations, which hold for both panels: More, non-permuted data yields higher test accuracy. Also, as expected, the optimal stopping gives the highest test accuracy. The fixed early stopping results in inferior test accuracy when only a fraction of the data is available. The neural persistence based stopping is triggered late when only a fraction of the data is available which results in a slightly better test accuracy compared to training and validation loss. The training loss stopping achieves similar test accuracies compared to the persistence based stopping (for all regimes except the very small data set) with shorter training, on average. We note that, it is generally not advisable to use training loss as a measure for stopping because the stability of this criterion also depends on the batch size. When only a fraction of the data is available, the validation loss based stopping stops on average after the same number of training epochs as the training loss, which results in inferior test accuracy because the network has seen in total fewer training samples. Most strikingly, validation loss based stopping is is triggered later (sometimes never) when most training and validation labels are randomly permuted which results in overfitting and poor test accuracy.

To conclude, the neural persistence based stopping achieves good performance without being affected by the batch size and noisy labels. The authors also note that the result is consistent for multiple architectures and most patience parameters.

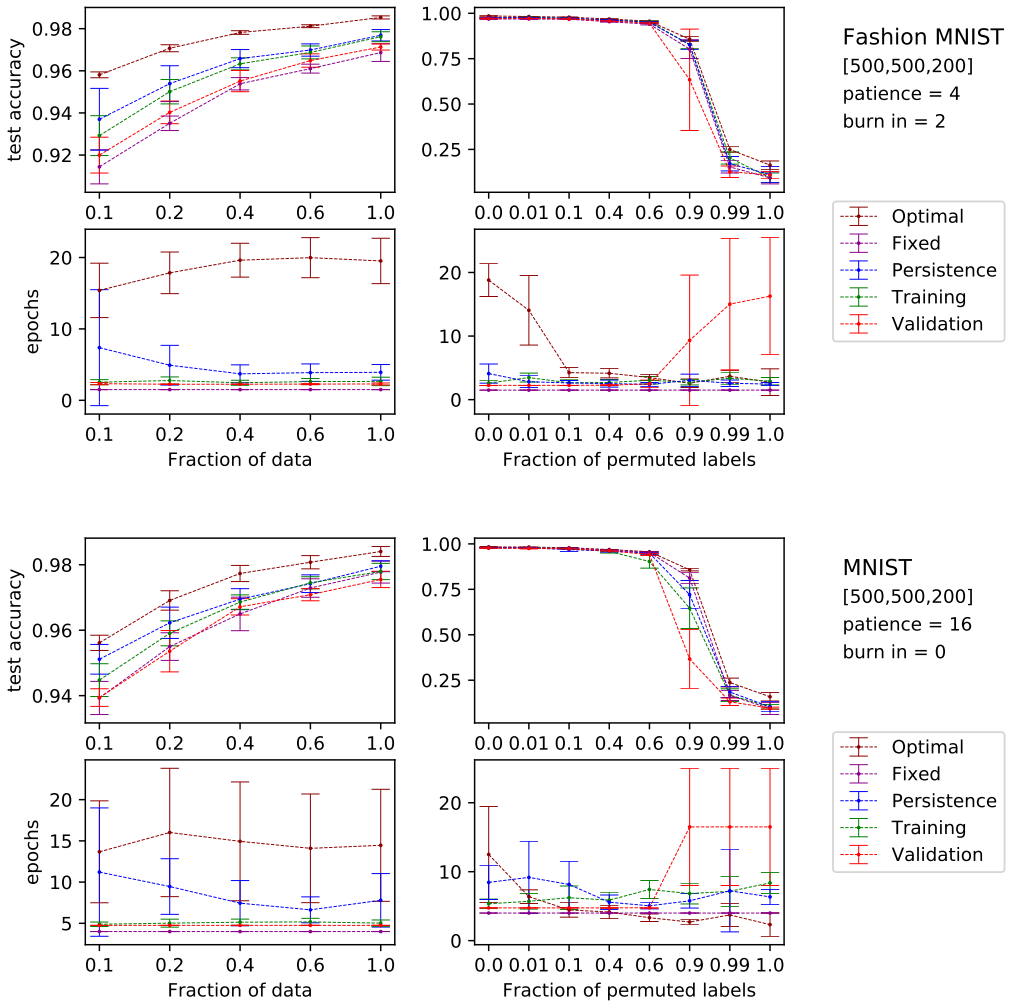

Figure A.12: On MNIST and Fashion-MNIST $\overline{\text{NP}}$ (in blue) stops later than validation and training loss when fewer training samples are available (left-hand side) which results in a higher test accuracy. For increasing noise in the training labels (right-hand side), the stopping of $\overline{\text{NP}}$ remains stable, in contrast to the validation loss stopping, which leads to lower test accuracy after longer training at a high fraction of permuted labels. The patience and burn in parameters are reported in quarter epochs.

Table A.1: Parameters and hyperparameters for the experiment on best practices and neural persistence. Dropout and batch normalization were applied after the first hidden layer. Throughout the networks, *ReLU* was the activation function of choice.

Data set	# Runs	# Epochs	Architecture	Optimizer	Batch Size	Hyperparameters
MNIST	50	40	$[650, 650]$	Adam	32	$\eta = 0.0003$ $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$ $\eta = 0.0003$ $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$, Batch Normalization $\eta = 0.0003$ $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$, Dropout 50%

Table A.2: Parameters and hyperparameters for the experiment on early stopping. Throughout the networks, *ReLU* was the activation function of choice.

Data set	# Runs	# Epochs	Architecture	Optimizer	Batch Size	Hyperparameters
(Fashion-)MNIST	100	10	Perceptron $[50, 50, 20]$	Minibatch SGD	100	$\eta = 0.5$
		40	$[300, 100]$ $[20, 20, 20]$	Adam	32	$\eta = 0.0003$ $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$
CIFAR-10	10	80	$[800, 300, 800]$	Adam	128	$\eta = 0.0003$ $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$
IMDB	5	25	$[128, 64, 16]$	Adam	128	$\eta = 1 \times 10^{-5}$ $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$