Supplementary Material for Training Robust Models Using Random Projection

Nguyen Xuan Vinh Sarah Erfani Sakrapee Paisitkriangkrai
James Bailey Christopher Leckie Kotagiri Ramamohanarao
Department of Computing and Information Systems
The University of Melbourne, VIC 3010, Australia. Correspondence Email: vinh.nguyen@unimelb.edu.au

Classification Accuracy on UCI Data

We also tested different regularizers on a number of data sets from the UCI repository [1]. These data sets vary in the number of samples and features, as detailed in Table I. The network structure consists of 2 hidden layers of $\{1024-1024\}$ nodes. Rectified linear units were used. For each data set, we rescale the input to $[-1,1]$. The learning rate was set to either 0.1 or 0.01 for different data sets (but fixed to the same value for all different regularizers), and momentum set to 0.9. The regularizers being tested are L2 (weight decay) with weight 1e-2, 1e-3 and 1e-4, Random Projection with $k = 10$ and 20 projections, dropout and a combination of dropout and Random Projection. As a reference, we also provide the performance of an RBF-kernel SVM of which the parameters were fine-tuned using 5-fold cross validation. Note that as opposed to SVM, for all neural network models, no effort was put into fine-tuning the hyper parameters, e.g., number of layers and nodes, learning rate and momentum. As our primary goal is to compare the effectiveness of different regularizers for NNs, this setting suffices.

From Table I, it can be observed that amongst all the different regularizers for NNs, while there is no absolute clear winner, the Random Projection regularizers perform competitively overall, either as a stand-alone regularizer or in conjunction with dropout. Furthermore, it is worth noting that these regularizers do not exclude the use of each other, and therefore can be used in tandem. From this experimental evidence, we promote the Random Projection regularizer as a new tool to add into existing NN toolkits.

Model Capacity vs. Number of RPs

The Random Projection regularizer admits one tuneable parameter: the number of random projections k. This parameter plays a role similar to the penalty weight in L1 or L2 regularizers. Given a NN with fixed architecture, increasing the number of random projections increases the amount of perturbation and variety in the augmented data set, making it increasingly harder to learn a precise (but potentially overfitted) decision boundary for any individual projection. We carry out an experiment on the MNIST data set as follows. Three small networks of two hidden layers of sizes $\{100-100\}$, $\{300-300\}$ and $\{500-500\}$ were tested with the number of projections k ranging from 1 (i.e., no augmentation) to 100. The networks were trained with hyper-parameters as per the previous section for 200 epochs, without any other form of regularization. The results of this experiment are presented in Fig. 1(b). It can be observed that, on the smallest $\{100-100\}$ network, the RP regularizers with $k = 50$ and 100 created an overly strong regularization effect, with error rates higher than the baseline vanilla network. This indicates that the network does not have sufficient learning capacity to capture all the variety within the large augmented data sets. On the other hand, the RP regularizers with $k = 5, 10$ and 20 perform well. When increasing the network size, we observed that the RP regularizers with $k = 50$ and 100 gradually perform better and eventually outperform the baseline on the $\{500 - 500\}$ network.

References

[1] K. Bache and M. Lichman, "UCI machine learning repository," http://archive.ics.uci.edu/ml, 2013.

Table I
5-Fold Cross Validation Error Rate on UCI Data (%)

Datasets	**n**	**d**	**Neural networks of {1024-1024} hidden nodes**									**SVM**
			No Reg.	**L2 (1e-2)**	**L2(1e-3)**	**L2(1e-4)**	**Ensemble(10)**	**RP(10)**	**RP(20)**	**Dropout**	**Dropout+RP(10)**	
Arrythmia	430	257	24.7±4.8	23.5±2.8	24.9±4.5	24.7±3.1	22.8±4.5	23.0±3.6	**20.7±3.0**	23.5±1.5	23.0±1.9	24.0±3.7
SRBCT	84	2308	2.4±5.3	3.5±5.3	2.4±5.3	2.4±5.3	1.2±2.6	1.2±2.6	**0.0±0.0**	3.5±5.3	2.4±3.2	0.0±0.0
Colon	62	2000	19.8±15.4	**16.4±13.6**	19.8±15.4	19.8±15.4	17.7±10.7	19.8±15.4	23.1±19.7	19.8±15.4	19.8±15.4	19.8±19.0
Lung	73	325	35.4±21.5	35.4±21.5	35.4±21.5	34.1±20.9	38.5±6.8	23.2±12.9	23.0±15.1	37.0±18.0	**21.8±15.4**	17.5±18.6
Optdigits	3823	64	1.6±0.3	3.0±0.6	1.6±0.3	1.5±0.3	1.5±0.4	**1.1±0.4**	1.3±0.4	1.3±0.5	1.1±0.6	0.9±0.4
Waveform	5000	21	16.5±1.2	**12.8±1.2**	14.4±1.3	15.9±1.1	14.5±0.8	15.1±0.9	14.6±1.6	14.7±0.6	14.2±1.3	13.3±1.3
HDR	2000	649	2.0±0.4	2.1±0.3	2.0±0.3	2.0±0.4	1.8±0.6	2.0±0.5	1.8±0.4	**1.6±0.5**	**1.6±0.5**	1.7±0.9
Lymphoma	96	4026	11.2±12.2	12.2±12.5	13.2±12.7	12.2±12.5	14.6±11.5	**8.2±11.6**	9.3±11.7	12.2±12.5	10.2±10.8	3.0±6.7
Leukemia	73	7129	4.2±6.3	2.9±6.4	4.2±6.3	4.2±6.3	4.2±6.3	5.5±5.9	**1.4±3.2**	2.8±3.8	2.9±6.4	2.9±6.4
Advertisement	3279	1558	2.7±0.9	2.5±0.9	2.5±0.8	2.6±0.9	2.7±0.7	2.4±0.8	2.3±0.8	2.5±0.9	**2.1±0.7**	2.7±0.9
Promoter	106	57	20.8±7.3	20.7±7.1	19.8±6.3	19.8±6.3	**18.9±7.6**	20.8±10.0	22.6±5.2	21.7±7.2	19.8±9.1	18.9±3.4
Musk2	6598	166	0.4±0.2	2.3±0.6	0.4±0.2	0.4±0.2	**0.2±0.2**	0.6±0.2	0.7±0.3	0.4±0.2	0.7±0.2	0.3±0.2
Spambase	4601	57	6.5±1.0	8.7±1.2	6.7±1.2	6.4±1.1	6.8±0.7	**6.0±1.0**	6.3±1.0	6.2±1.0	7.0±0.9	6.1±0.6

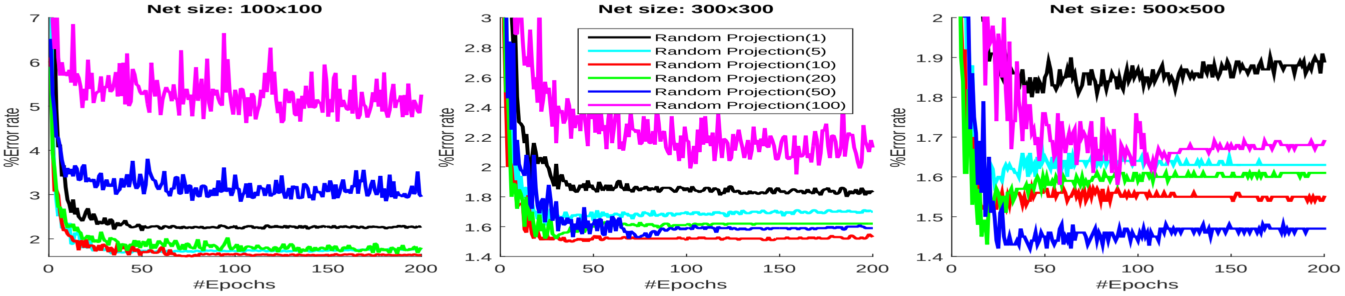

Figure 1. Experiments on the MNIST data set: Model capacity vs. number of RPs. Model capacity needs to be increased to accommodate the extra data complexity introduced by RPs.