Impact of classification difficulty on the weight matrices spectra in Deep Learning and application to early-stopping

Xuran Meng

Department of Statistics and Actuarial Science
The University of Hong Kong

May 18, 2023
Joint work with Jeff Yao
[JMLR, 2023]

Introduction of Deep Neural Networks (DNNs)

- Classification: data pair (h, y): $\{h \in \mathbb{R}^p$ with label y.
- Prediction of y based on the feature vector h using a DNN:

$$\hat{y}(h) = W_L \sigma(W_{L-1} \sigma(....\sigma(W_1 h))).$$

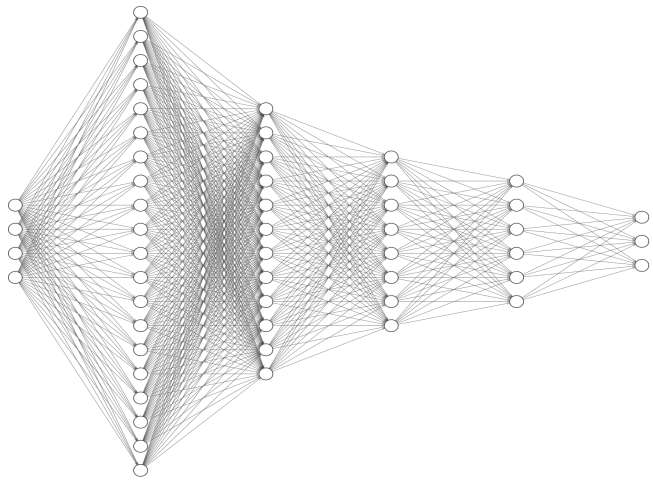
with activation function $\sigma(\cdot)$, weight matrix W_k connecting layer $k-1$ to k;

- The DNN is parametrized by its weight matrices: $\theta = \{W_k\}$
  - Loss of the prediction: $\mathcal{L}_\theta(y, \hat{y})$;
  - Given a set of labeled data

$$\mathcal{D} = \{(h_i, y_i) : \ 1 \le i \le N\},$$

we optimize weight matrices $\theta = \{W_k\}$ by minimizing the empirical loss function:

$$\mathcal{L}_N(\theta) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_\theta(y_i, \hat{y}(h_i)).$$



1.1: Deep Neural Network

May 18, 2023 Joint work with Jeff Y

Optimization using the Stochastic Gradient Descent (SGD) algorithm

- The optimization procedure uses Back propagation with a SGD search:

Algorithm 1 SGD Algorithm

Require: DataSet $\mathcal{D}$, learning rate η, batch size B, loss function $\mathcal{L}_\theta(y, \hat{y})$, initialized θ^0.

Iter $\leftarrow \lfloor N/B \rfloor$, Epoch $\leftarrow 0$, $t \leftarrow 0$

for Epoch =1:250 do
  for iter=1:Iter do
    Randomly select a subset of B-samples points $\{(h_i, y_i), 1 \le i \le B\}$:

$$\hat{\mathcal{L}}_{\theta^t} \leftarrow \frac{1}{B} \sum_{i=1}^{B} \mathcal{L}_{\theta^t}(y_i, \hat{y}(h_i)); \qquad \theta^{t+1} \leftarrow \theta^t - \eta \nabla_{\theta^t} \hat{\mathcal{L}}_{\theta^t} + \varepsilon_t$$

  end for

end for

- Other updating algorithms: AdaGrad, RMSProp, Adam, etc.

Mystery of Deep Learning

- DL has achieved impressive success in numerous areas;
- Any rational explanation of the success?
- The task is difficult becuse of the use of a collection of expert choices that determine the final structure of the DNNs: nonlinear activation, hidden layer architecture, loss function, back propagation algorithm, batch size B, etc.
- Empirical choices bringing non-linearity into the model, and non-convexity of optimization into the training process;
- Problem: lack of general guidelines about the "right choices" to design and train an effective DNN.

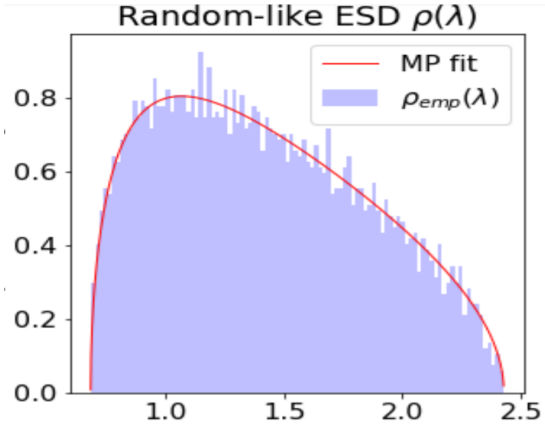
Can Random Matrix Theory help understand Deep Learning?

RMT in Deep Learning

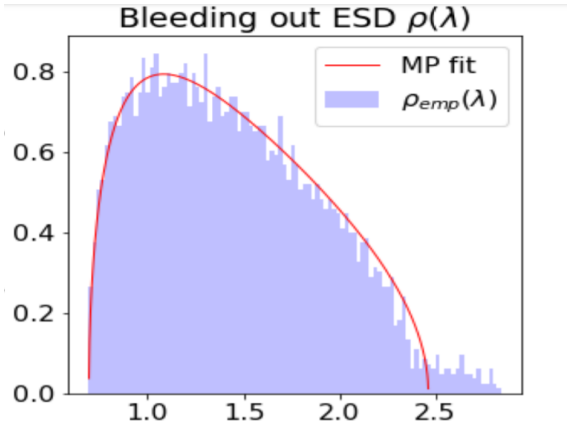
- Hessian Matrix: the final Hessian matrix of the empirical loss function can be realized as a Wishart matrix + Hermitian matrix(Pennington and Bahri,2017);
- Double/Triple Descent: theoretical results of generalization error calculated by RMT can explain the Double/Triple descent phenomena (Advani, Saxe, and Sompolinsky, 2020; Mei and Montanari, 2020);
- GAN: Identification of characteristics of generative networks in GAN via RMT(Seddik et al., 2020);
- Regularization in spectrum of a weight matrix: (Martin and Mahoney, 2018) studied the transition of weight matrix during the training time, established a 5+1 phases phenomenon, a connection between regularization and weight matrix spectrum.

Implicit Regularization (Martin and Mahoney,2018)

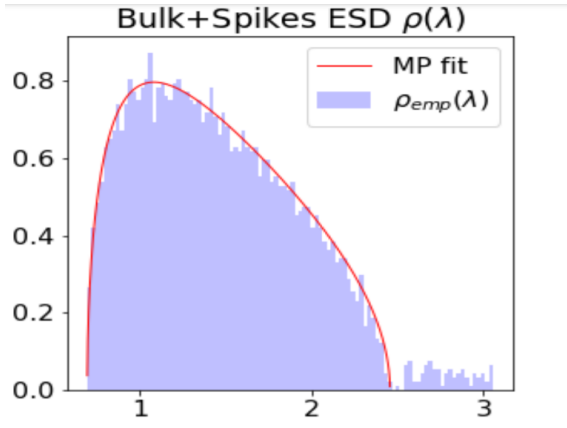
- Spectrum of $W^T W$ during the training process



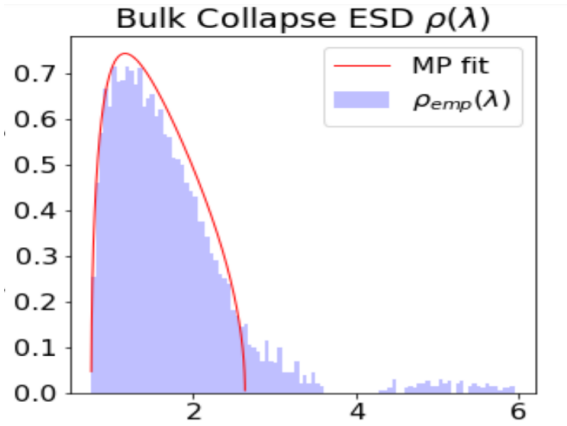
1.2: Phase1



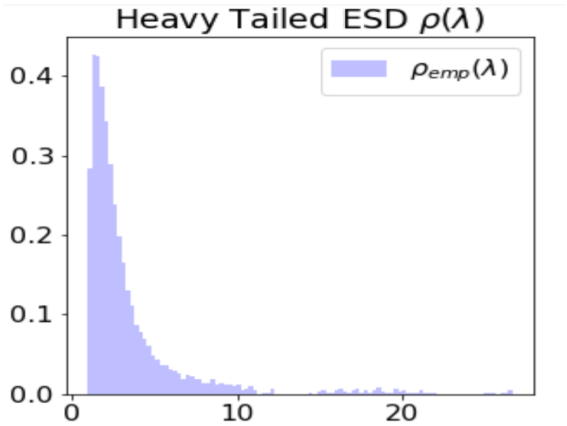
1.3: Phase2



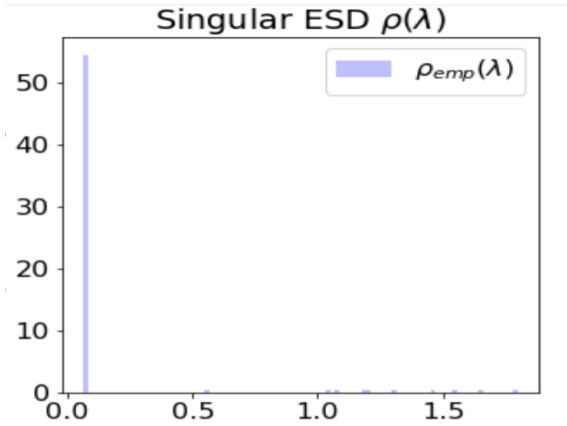
1.4: Phase3



1.5: Phase4



1.6: Phase5



1.7: Phase6

Some questions

- Different cases have different terminal types, modern neural networks training on modern data sets seem to achieve Heavy Tail, but neural networks on classical data sets always maintain MP Law. (Martin and Mahoney, 2018).
- How these different spectral shapes emerge? (,) pointed out that SGD could bring out heavy tails under linear settings.
- What are the factors that affect these spectral shapes? (Hodgkinson and Mahoney) pointed out some factors such as batch size, learning rate etc.
- Could we better understand these different spectral shapes?

Our contributions

- The difficulty of a classification problem is identified as a driving factor for the appearance of heavy tails in weight matrices spectra.
- Following the previous work of Martin and Mahoney, bulks of weight matrix spectra at final training stage are classified into three types:

  Light Tail (LT), Bulk Transition period (BT) and Heavy Tail (HT).

  While decreasing the classification difficulty, these spectrum bulks obey a phase transition from HT to BT, and then to LT.
- Leveraging on these findings, we propose a spectral criterion to guide the early stopping without using testing data.

  The HT(BT)-based spectral criterion could not only cut off a large training time with just a little drop of test accuracy, but also avoid over-fitting even when the training accuracy is increasing.

Settings of numeric experiments

A Gaussian ANOVA model

With K classes, data in class $k \in \{1, ..., K\}$ follows

$$h_{i,k} = \mu_k + \varepsilon_{i,k}, \quad 1 \le i \le n_k,$$

where $\mu_k \in \mathbb{R}^p$ is the class mean, $\varepsilon_{i,k} \overset{iid}{\sim} \mathcal{N}(0, \sigma^2 I_p)$ are Gaussian noise, n_k is the total number of observation from class k.

- Model $\mathcal{D}_1(\delta)$ for class means: set a random subset $I_k \subset \{1, ..., p\}$, of size $p/2$, and

$$\mu_k = m\mathbf{1}_{I_k} + (m + \delta)\mathbf{1}_{I_k^c}.$$

- Model $\mathcal{D}_2(t)$ for class means: the ETF structure (Papyan et al., 2020)

$$\mu_k = tv_k, \qquad v_k = \mathbf{1}_{\{j=k\}} - \frac{1}{K}\mathbf{1}_{\{1 \le j \le K\}}, \quad 1 \le j \le p.$$

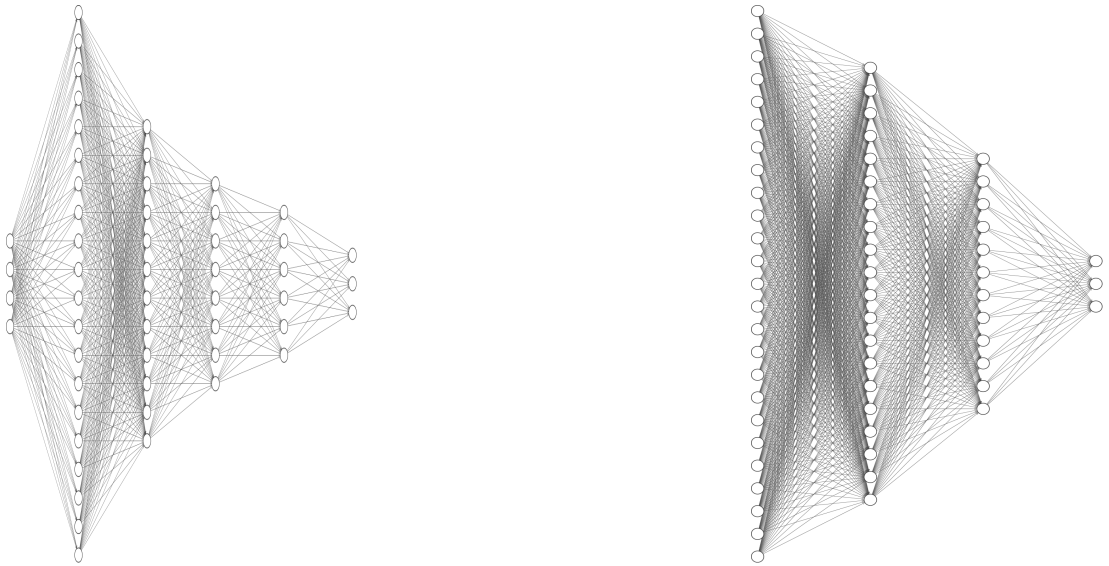
- Set $m = -0.2$, $\sigma = 1$, the tuning parameters are δ and t which tune data SNR:

$$\text{SNR} = \underset{\{k,k'\}}{\text{Ave}} \frac{||\mu_k - \mu_{k'}||}{\sigma}.$$

Settings

- The structure of two DNNs:

$$\text{NN1:} \quad 100 \to 1024 \to 512 \to 384 \to 192 \to K,$$
$$\text{NN2:} \quad 2048 \to 1024 \to 512 \to K.$$



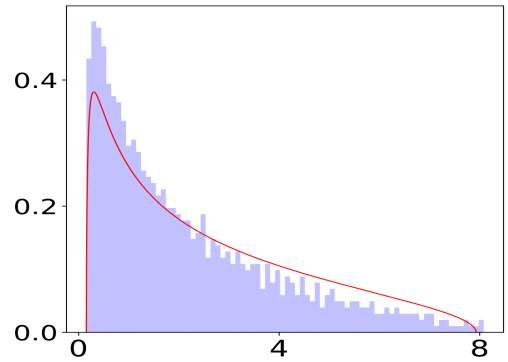
(a) NN1 (b) NN2

- The activation function is $\sigma(x) = \text{ReLU}(x) = \max(0, x)$.

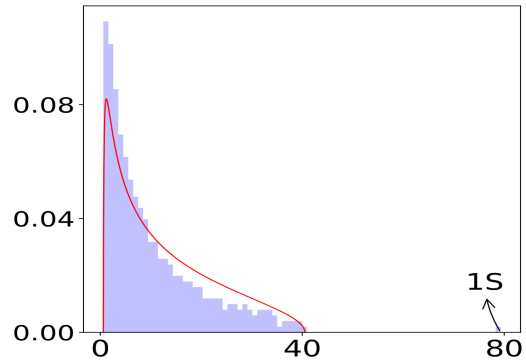
- $n_k = 7500$ for each class; cross-entropy loss with learning rate 0.01; $B = 64$ and Pytorch's default initialization.

Phase transition of spectra when the SNR increases

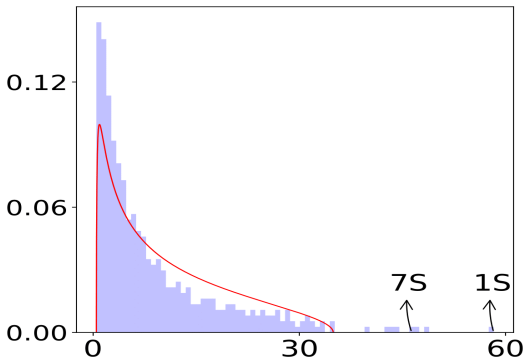
Heavy Tailed spectra:



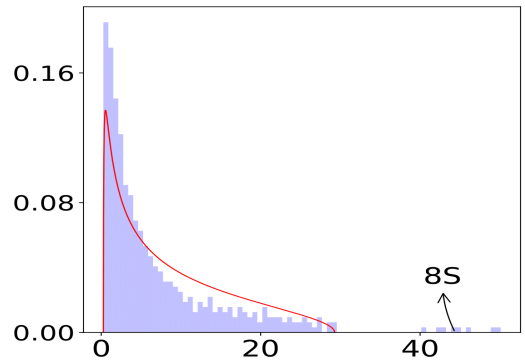
2.1: HT(0,0)



2.2: HT(0,1)



2.3: HT(7,1)

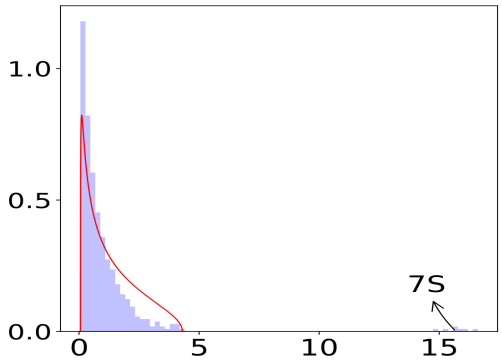


2.4: HT(0,8)

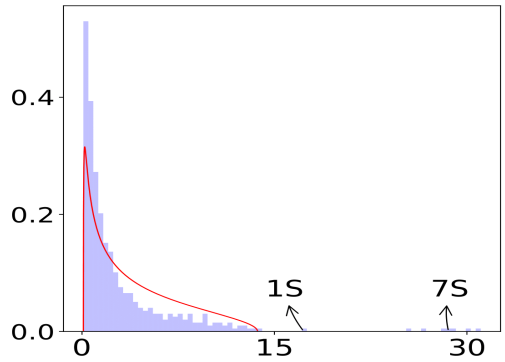
May 18, 2023 Joint work with Jeff Y

Phase transition of spectra when the SNR increases

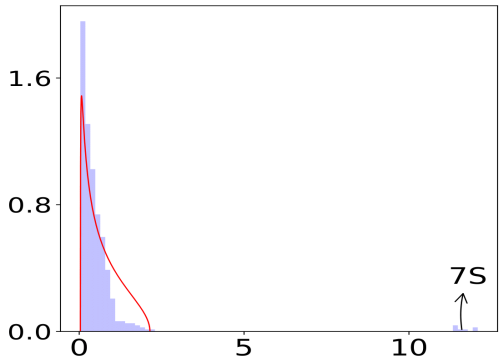
Bulk transition spectra:



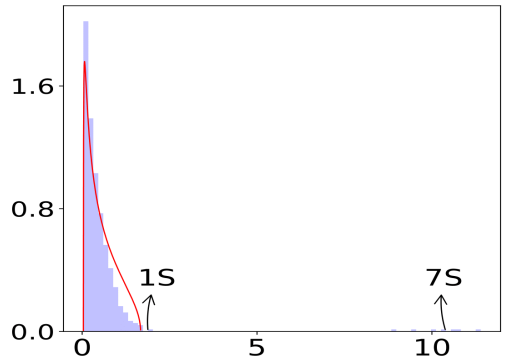
2.5: BT(0,7)



2.6: BT(1,7)



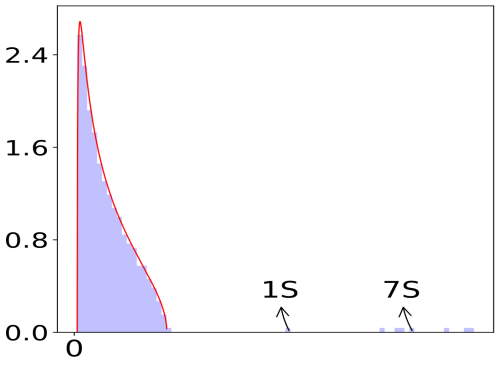
2.7: BT(0,7)



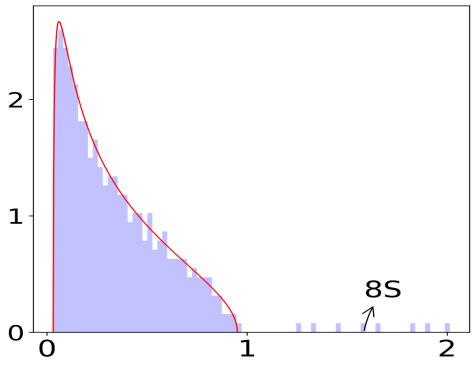
2.8: BT(1,7)

More detailed types of spectrum bulk

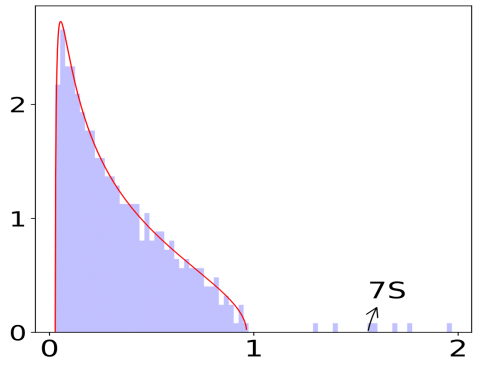
Light Tailed spectra (MP Laws):



2.9: LT(1,7)

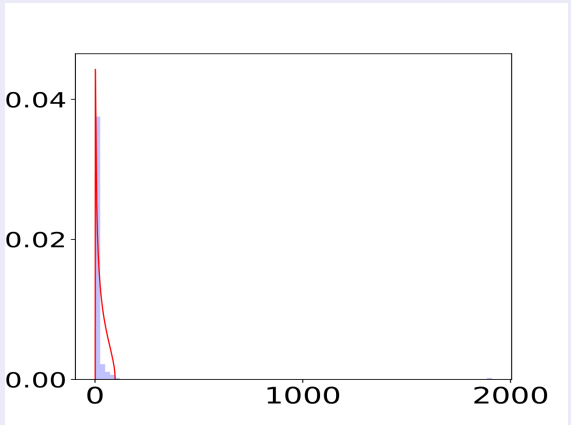


2.10: LT(0,8)



2.11: LT(0,7)

Special case of Rank collapse: when the SNR is very low



The impact of data SNR and class number K

Empirical findings:

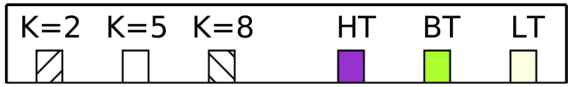
- The transition of spectrum types direction is

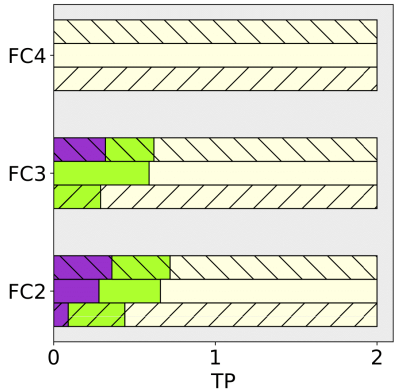
  $$\text{Heavy Tail} \to \text{Bulk Transition} \to \text{Light Tail},$$

  as the data SNR increases. The Bulk Transition starts when the $K-1$ spikes become distant as shown in Figures 2.5 and 2.6.

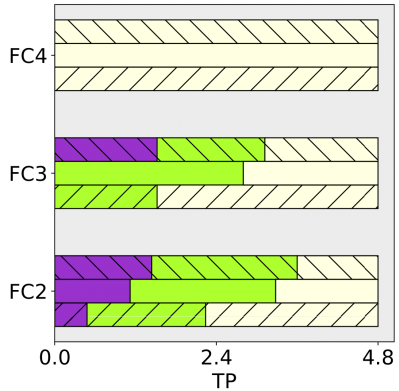
- When the data SNR fixed, the spectrum types have the tendency from Light Tail $\to$ Heavy Tail when the class number K increases.

The impact of data SNR and class number K

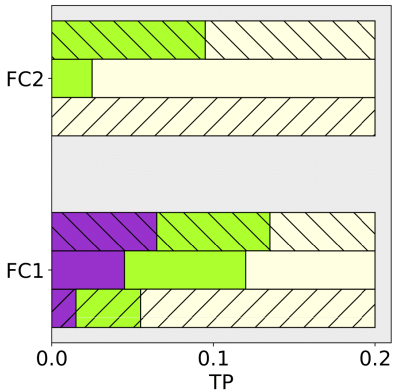




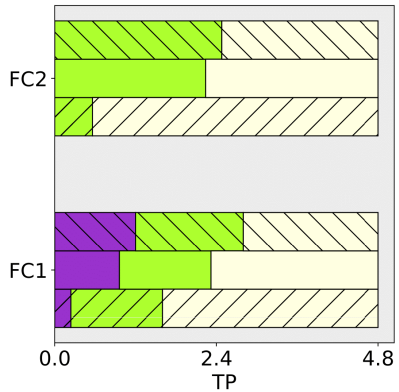
(b) NN1+$\mathcal{D}_1$



(c) NN1+$\mathcal{D}_2$



(d) NN2+$\mathcal{D}_1$



(e) NN2+$\mathcal{D}_2$

The impact of data SNR and class number K

- The four subfigures for the NNs/datasets combinations all contain the Same Transition Period Direction:

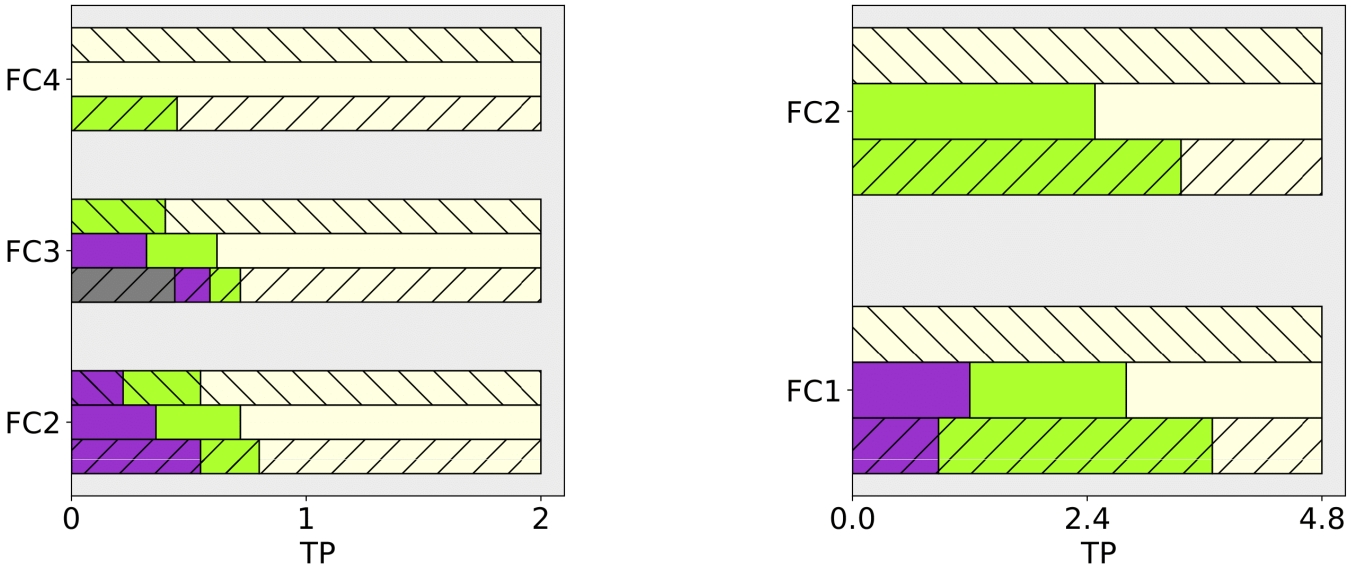
  $$\text{Heavy Tail} \rightarrow \text{Bulk Transition} \rightarrow \text{Light Tail.}$$

- The more classes the data set has, the higher difficulty to classify them correctly. For a given layer in the neural network at the same SNR level, the tails of spectrum bulks become heavier as the number of classes K increases.

- In line with the previous work in Hodgkinson and Mahoney, the statement that the wider layers will more likely to bring in HT is validated in our experiments. Practioners are thus suggested to design wider layers for learning process monitoring.

Additional experiments on batch sizes

- The Phase Transition of data SNR

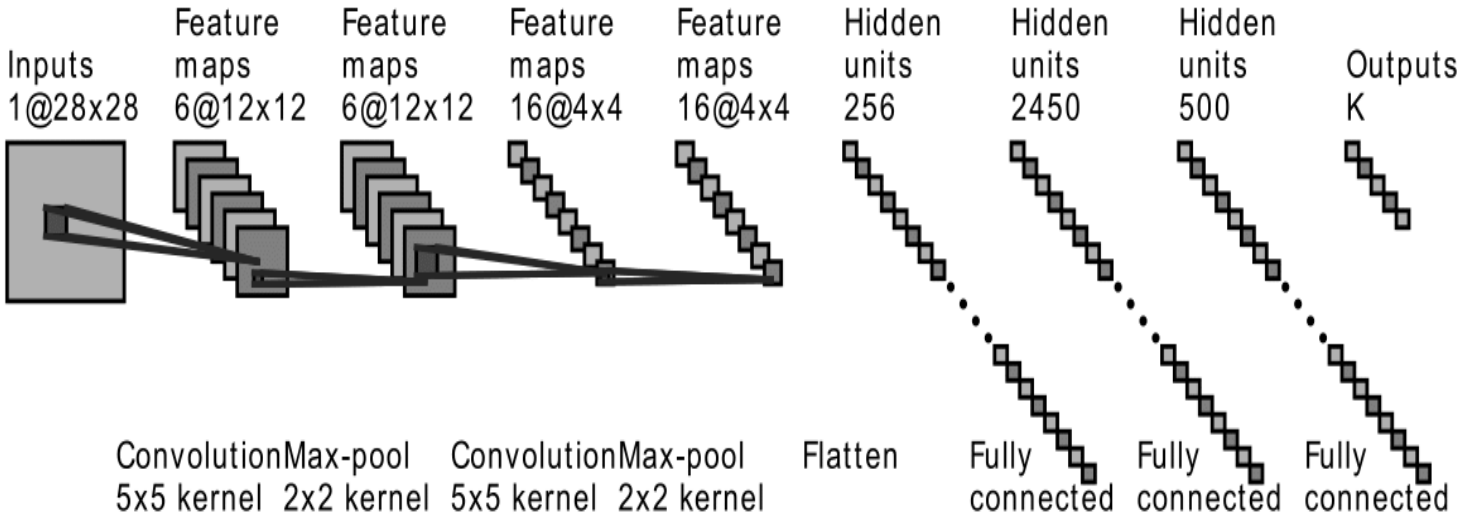


(a) NN1+$\mathcal{D}_1$ (b) NN2+$\mathcal{D}_2$

2.14: Transition Period

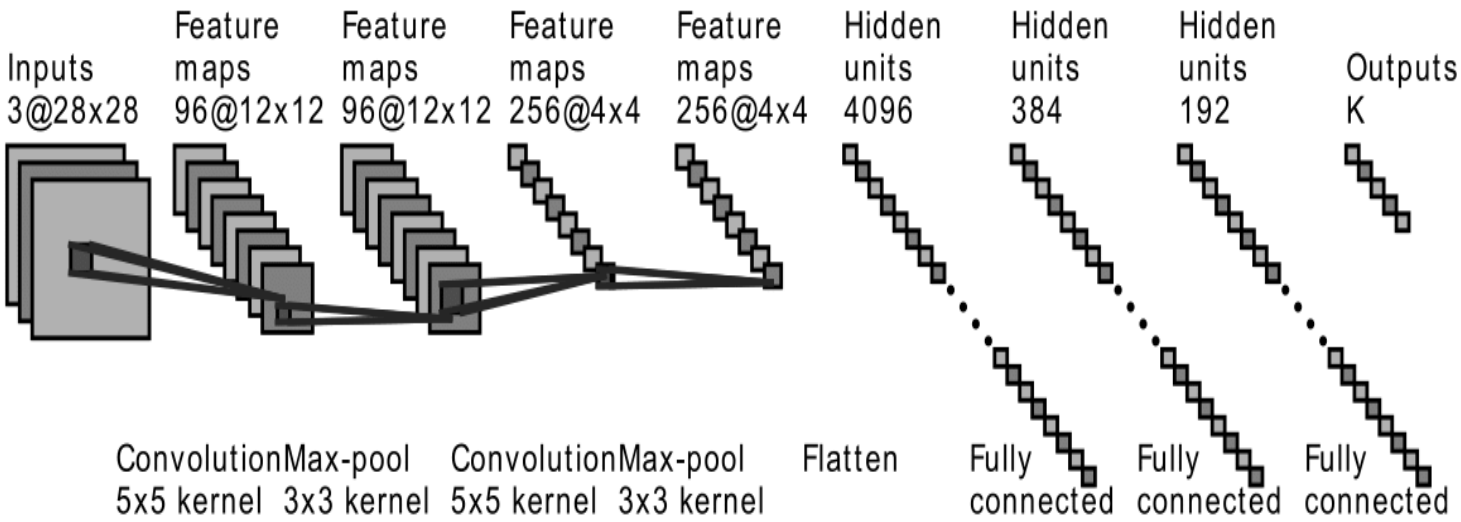
Settings: Real Data experiments

The neural networks are LeNet , MiniAlexNet and VGG, trained on MNIST and CIFAR10 respectively.



2.15: The structure of LeNet

Settings: Real Data experiments

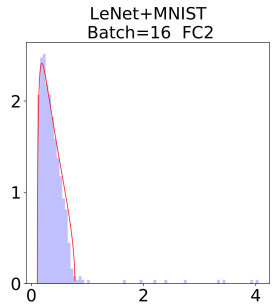


2.16: The structure of MiniAlexNet

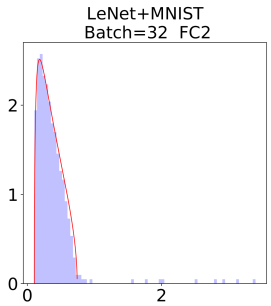
Comparison in LeNet

LeNet trained on MNIST and CIFAR10 at final epoch,

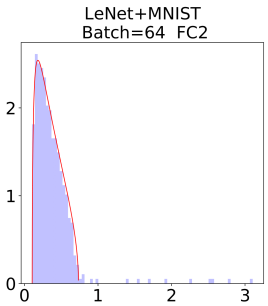
with batch size =16,32,64,128 and 256, respectively.



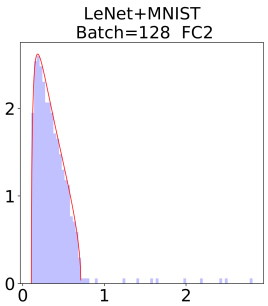
2.17: MNIST-16



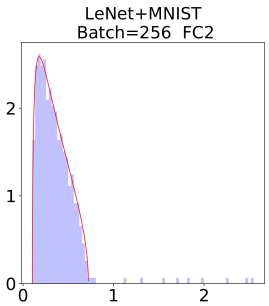
2.18: MNIST-32



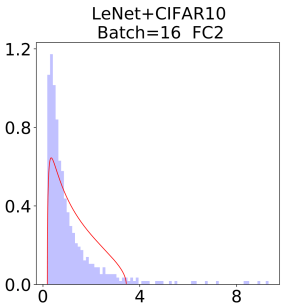
2.19: MNIST-64



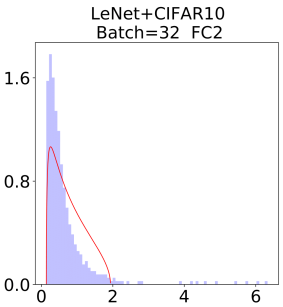
2.20: MNIST-128



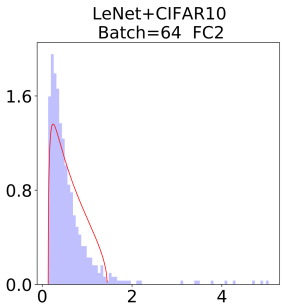
2.21: MNIST-256



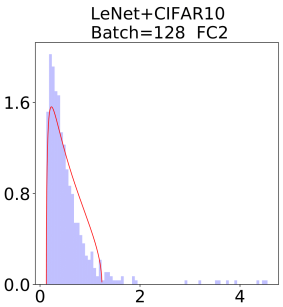
2.22: CIFAR-16



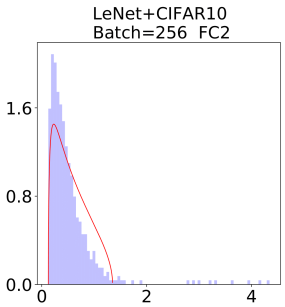
2.23: CIFAR-32



2.24: CIFAR-64



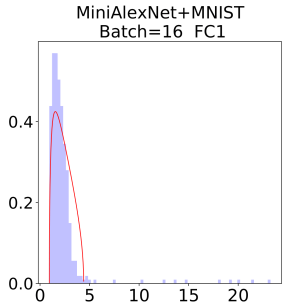
2.25: CIFAR-128



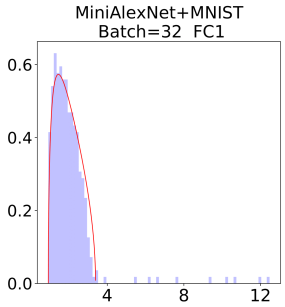
2.26: CIFAR-256

Comparison in MiniAlexNet

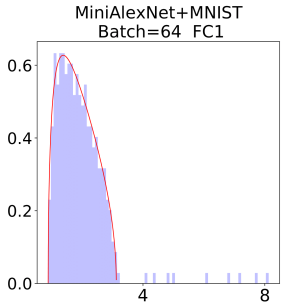
MiniAlexNet training on MNIST and CIFAR10 at final epoch respectively.



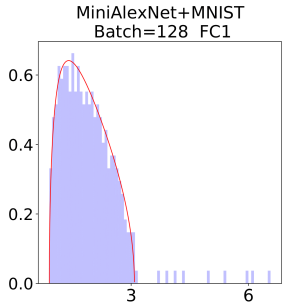
2.27: MNIST-16



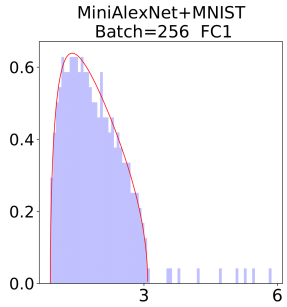
2.28: MNIST-32



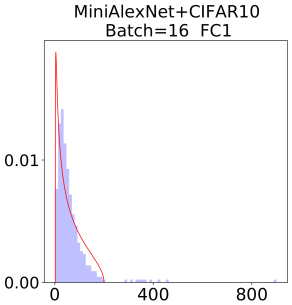
2.29: MNIST-64



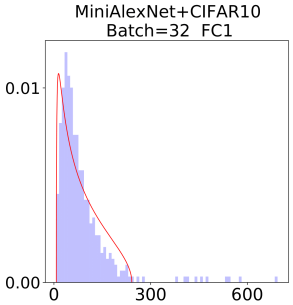
2.30: MNIST-128



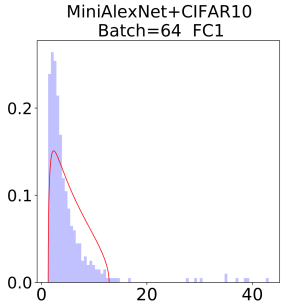
2.31: MNIST-256



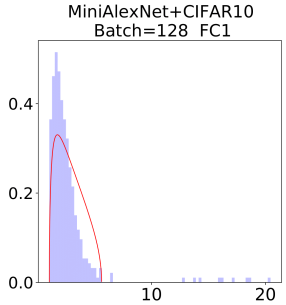
2.32: CIFAR-16



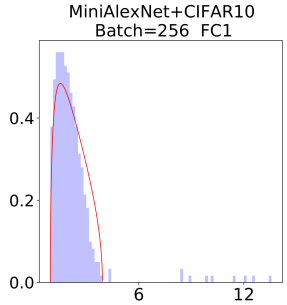
2.33: CIFAR-32



2.34: CIFAR-64



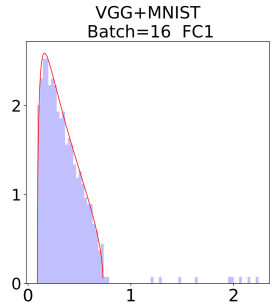
2.35: CIFAR-128



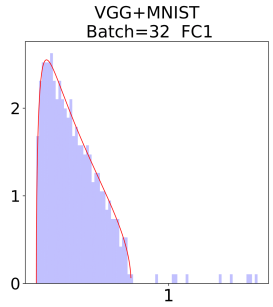
2.36: CIFAR-256

Comparison in VGG

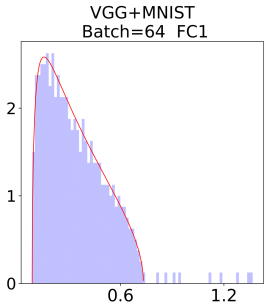
VGG training on MNIST and CIFAR10 at final epoch respectively.



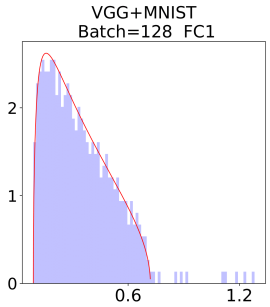
2.37: MNIST-16



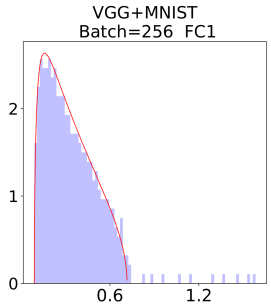
2.38: MNIST-32



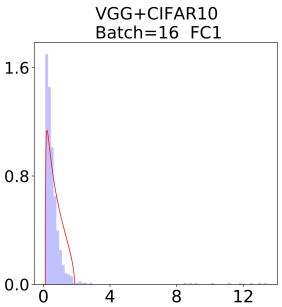
2.39: MNIST-64



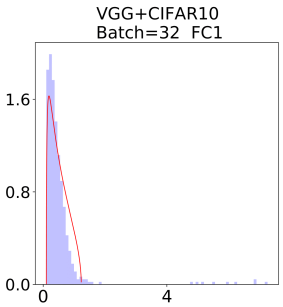
2.40: MNIST-128



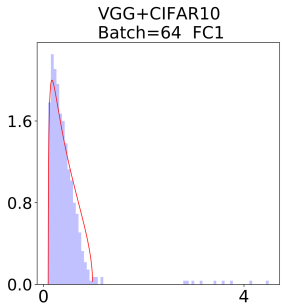
2.41: MNIST-256



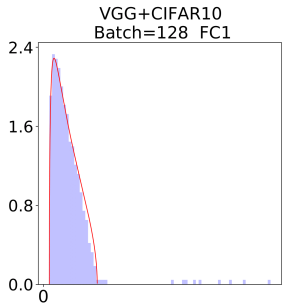
2.42: CIFAR-16



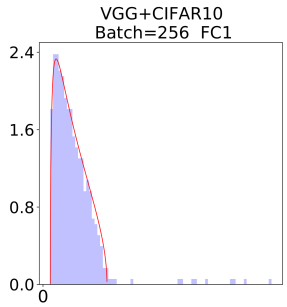
2.43: CIFAR-32



2.44: CIFAR-64



2.45: CIFAR-128



2.46: CIFAR-256

Conclusions from experiments on MNIST and CIFAR10

- In summary, these experiments show that the spectrum type of the weight matrix depends much more on the data set itself than the NN architecture.

- Different levels of classification difficulty in MNIST and CIFAR10 as shown by the detection rates on the test data of the trained NNs with the three architectures:

Data Set \ NN	LeNet	MiniAlexNet	VGG11
MNIST	99% (BT/LT)	99% (BT/LT)	99% (LT)
CIFAR10	64% (HT)	76% (HT)	81% (HT)

- The differences in testing accuracy give evidence that CIFAR10 has much more complex features than MNIST, and the classification problem is more difficult for CIFAR10 than for MNIST.

- In a sense, complex features in a data set will bring in complex correlations in weight matrix entries, thus generating heavy tails in their spectrum.

Summary from both experiments on synthetic and real data

- The degree of difficulty of a classification problem has a great impact on the spectra of weight matrices. We study the phenomenon from three aspects: the SNR, the number of classes and the complexity of data features.
- Heavy tails could be regarded as a training information encoder.
- According to the observed results, we propose spectral criterion to guide the early stopping.

Introduction of spectral criterion

- Classical method is to split data into training and testing parts;
- However, it may happen that test data are not available, e.g. when using a pre-trained DNN;
- An important question here is:

  Without test data, is there any guidance for early stopping?

- We propose a spectral criterion to guide such early stopping.

A spectral criterion

Key idea: a good early stopping time is when HT spectrum appears!

- An indication of the poor quality in the training data or the whole system. The emergence of Heavy Tail could be treated as an alarm for the hidden and problematic issues in the network.

- An indication of a regularized structure that has acquired considerable information from the training data. The emergence of Heavy Tail will somehow ensure the testing accuracy of the whole system, and additional training will not bring much improvement.

A spectral criterion

- Formally, consider a $n \times n_0$ weight matrix ($n \leq n_0$) and let $X_1, ..., X_n$ be the non-zero eigenvalues of the square matrix WW^T (The initialization W has been rescaled with $1/\sqrt{N}$).

- With respect to RMT Results, the density function of the standard MP Law MP_{c,σ^2} is

$$p_{c,\sigma^2}(x) = MP_{c,\sigma^2}(x) = \frac{1}{2\pi c\sigma^2 x}\sqrt{(b-x)(x-a)},$$

  with $a = \sigma^2(1-\sqrt{c})^2$ and $b = \sigma^2(1+\sqrt{c})^2$.

- $X_1, X_2, ..., X_n$ are supported on the interval $[a, b]$, with $0 < a < b$. Consider a mesh net on the interval on M bins of binsize $(b-a)/M$,

$$B_j = \left(a + (j-1)\frac{b-a}{M}, a + j\frac{b-a}{M}\right], \quad 1 \leq j \leq M.$$

  The histogram estimator for the density function of the data is

$$\hat{p}_M(x) = \frac{M}{n(b-a)}\sum_{i=1}^{n} I(X_i \in B(x)),$$

  $B(x)$ is the bin x belongs to.

A spectral criterion

- We use the following L_1 distance between the two density functions to measure the departure of the data points $\{X_i\}$ from the MP law:

$$s_n = \int_a^b |\hat{p}_M(x) - p_{c,\sigma^2}(x)|dx.$$

- In practice, we do not know the parameters c and σ^2 of the reference MP density $p_{c,\sigma^2}(x)$. Then we use the observed extreme statistics $\hat{a} = X_{(1)}$, and $\hat{b} = X_{(n)}$ to estimate a and b, respectively. These lead to corresponding estimates $\hat{c}$ and $\hat{\sigma}^2$ for the parameters c and σ^2, respectively. The MP density function with estimated parameters is then

$$p_{\hat{c},\hat{\sigma}^2}(x) = \frac{1}{2\pi\hat{c}\hat{\sigma}^2 x}\sqrt{(\hat{b}-x)(x-\hat{a})}I(\hat{a} \le x \le \hat{b}).$$

- The final L_1 distance statistic between the data set $\{X_1, \ldots, X_n\}$ and the MP law is

$$\hat{s}_n = \int_a^b |\hat{p}_M(x) - p_{\hat{c},\hat{\sigma}^2}(x)|dx.$$

Property of the distance statistic under the MP law assumption

So we reformulate the detection of HT spectrum as a deviation to the null hypothesis:

H_0 : the eigenvalues of the weight matrix WW^T obeys the MP law.

Proposition

Under the null hypothesis H_0, the distance statistic $\hat{s}_n$ satisfies

$$\hat{s}_n = O_P\left(\frac{1}{n^{1/3}} + \frac{1}{M} + \sqrt{\frac{M \log n}{n}}\right). \tag{1}$$

Definition of the spectral criterion

Spectral criterion.

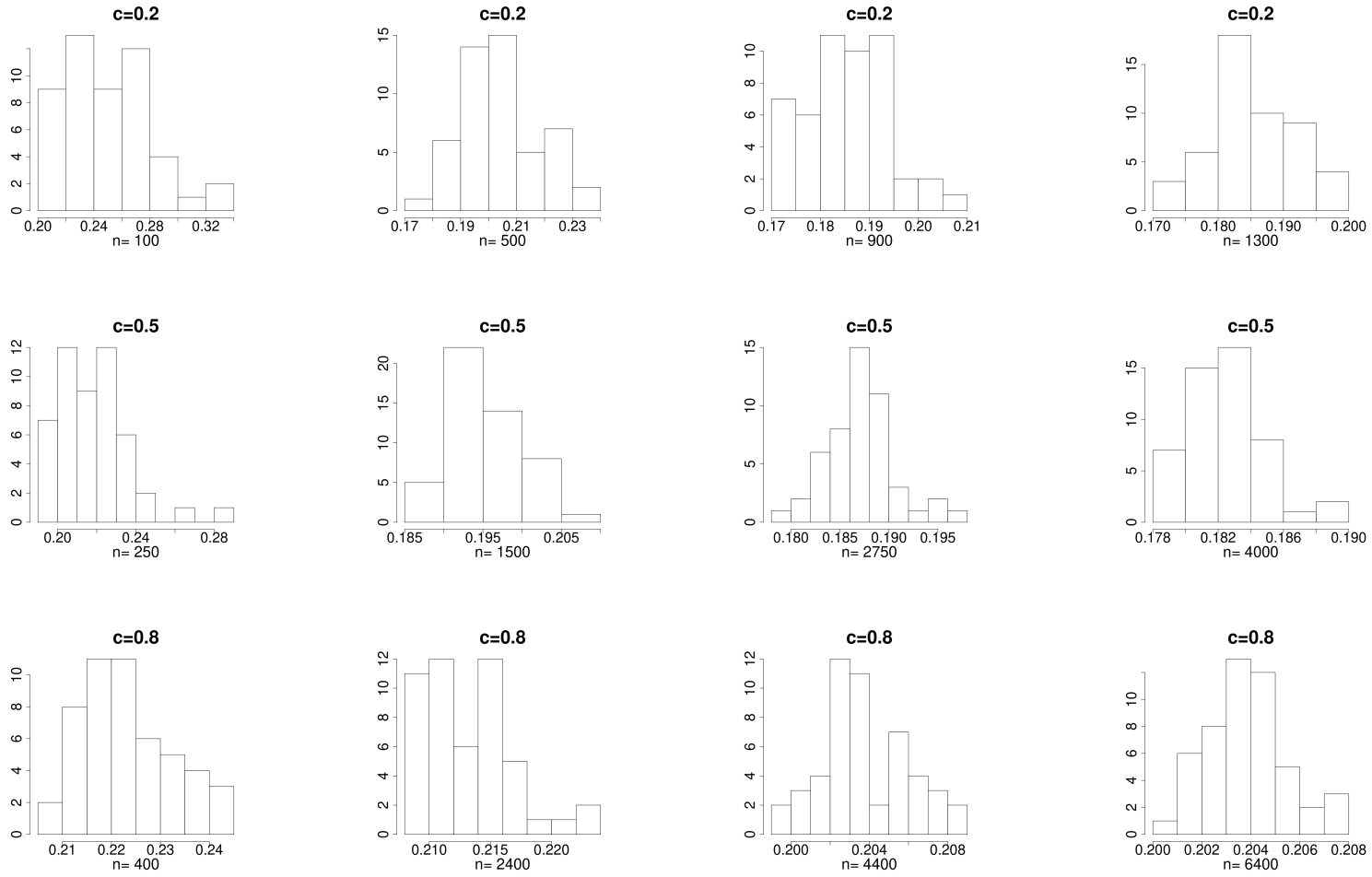
Set $M = 2\lfloor n^{\frac{1}{3}} \rfloor$ and consider a thresold value $s_* = C * \sqrt{\log n}/n^{\frac{1}{3}}$ with $C = 0.4$.

For each training epoch, calculate $\hat{s}_n$ as previously and stop the training if $\hat{s}_n > s_*$.

Remark

- In the simulation for different MP Laws, the value $\hat{s}_n n^{\frac{1}{3}}/\sqrt{\log n}$ always lies in the interval [0.15,0.25].
- We empirically suggest the critical line with $C = 0.4$ from theoretical simulations with comparison to the extreme value 0.35 displayed in Figure 3.1.
- Empirical results selected by the spectral criterion from $C = 0.4$ to $C = 0.6$ are all acceptable in our experiments. Basically, any value of C in the range of $[0.4, 0.6]$ can be recommended for the spectral criterion from our experimental results.

Simulations for the scaled statistic $\hat{s}_n n^{\frac{1}{3}} / \sqrt{\log n}$



3.1: Histograms of $\hat{s}_n n^{\frac{1}{3}} / \sqrt{\log n}$ from different c and n: For each pair of (c, n), the eigenvalues are generated from standard MP Law with 50 repetitions that lead to 50 values of the statistic.

Application of spectral criterion in Numeric experiments

The combination NN1+$\mathcal{D}_1$

Typical TP	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC2)	Test Acc	epoch(FC3)	Test Acc	FC1	FC2	Test Acc
0.15	7	25.84%	10	23.23%	HT	HT	20.17%
0.2	7	32.70%	12	27.48%	HT	HT	27.03%
0.3	7	49.36%	12	45.48%	HT	HT	44.80%
0.6	8	88.52%	32	88.32%	MPB	MPB	88.30%
0.9	-		-		MP	MP	99.13%

The combination NN1+$\mathcal{D}_1$

Typical TP	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC2)	Test Acc	epoch(FC3)	Test Acc	FC1	FC2	Test Acc
0.15	8	24.58%	16	20.44%	HT	HT	20.17%
0.2	8	31.50%	16	25.83%	HT	HT	27.03%
0.3	8	49.09%	12	45.48%	HT	HT	44.80%
0.6	9	87.96%	-		MPB	MPB	88.30%
0.9	-		-		MP	MP	99.13%

Application of spectral criterion in Numeric experiments

The combination NN1+$\mathcal{D}_2$

Typical TP	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC2)	Test Acc	epoch(FC3)	Test Acc	FC1	FC2	Test Acc
0.24	9	14.69%	16	13.89%	HT	HT	13.08%
1.2	7	38.61%	12	35.84%	HT	HT	32.98%
2.4	7	77.19%	16	74.55%	HT	HT	75.92%
3.2	9	92.11%	-		HT	MPB	92.64%
4.8	-		-		MP	MP	99.73%

The combination NN1+$\mathcal{D}_2$

Typical TP	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC2)	Test Acc	epoch(FC3)	Test Acc	FC1	FC2	Test Acc
0.24	9	14.69%	-		HT	HT	13.08%
1.2	8	39.31%	16	32.11%	HT	HT	32.98%
2.4	8	76.75%	20	74.29%	HT	HT	75.92%
3.2	10	91.94%	-		HT	MPB	92.64%
4.8	-		-		MP	MP	99.73%

Application of spectral criterion in Numeric experiments

The combination NN2+$\mathcal{D}_1$

Typical TP	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
0.02	6	14.89%	7	15.84%	HT	MPB	16.02%
0.04	8	24.78%	7	23.34%	HT	MPB	25.38%
0.07	5	48.31%	6	48.63%	HT	MPB	50.12%
0.13	6	87.03%	-		MPB	MP	87.50%
0.2	-		-		MP	MP	99.14%

The combination NN2+$\mathcal{D}_1$

Typical TP	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
0.02	-		-		HT	MPB	16.02%
0.04	-		-		HT	MPB	25.38%
0.07	-		-		HT	MPB	50.12%
0.13	-		-		MPB	MP	87.50%
0.2	-		-		MP	MP	99.14%

Application of spectral criterion in Numeric experiments

The combination NN2+$\mathcal{D}_2$

Typical TP	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
0.24	10	13.08%	6	12.89%	HT	HT	13.44%
1.2	12	34.22%	5	34.63%	HT	HT	36.31%
2.4	5	72.59%	16	74.61%	MPB	MPB	75.12%
3.2	-		-		MP	MP	91.20%
4.8	-		-		MP	MP	99.59%

The combination NN2+$\mathcal{D}_2$

Typical TP	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
0.24	12	13.48%	7	13.19%	HT	HT	13.44%
1.2	24	35.80%	5	34.63%	HT	HT	36.31%
2.4	6	73.80%	36	74.86%	MPB	MPB	75.12%
3.2	-		-		MP	MP	91.20%
4.8	-		-		MP	MP	99.59%

Application of spectral criterion in Real Data experiments

The combination LeNet+MNIST

batchsize	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	-		16	99.08%	MP	MPB	99.17%
32	-		40	99.13%	MP	MPB	99.17%
64	-		68	98.98%	MP	MPB	98.98%
128	-		124	98.91%	MP	MP	99.03%
256	-		-		MP	MP	98.96%

The combination LeNet+MNIST

batchsize	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	-		32	99.19%	MP	MPB	99.17%
32	-		-		MP	MPB	99.17%
64	-		-		MP	MPB	98.98%
128	-		-		MP	MP	99.03%
256	-		-		MP	MP	98.96%

Application of spectral criterion in Real Data experiments

The combination LeNet+CIFAR10

batchsize	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	24	61.37%	8	61.62%	MPB	HT	64.99%
32	60	64.78%	10	57.94%	MP	HT	64.57%
64	-		28	59.19%	MP	HT	62.49%
128	-		60	61.38%	MP	HT	61.83%
256	-		84	54.23%	MP	HT	60.49%

The combination LeNet+CIFAR10

batchsize	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	28	61.66%	8	61.62%	MPB	HT	64.99%
32	-		20	61.06%	MP	HT	64.57%
64	-		32	60.27%	MP	HT	62.49%
128	-		60	61.38%	MP	HT	61.83%
256	-		92	58.33%	MP	HT	60.49%

Application of spectral criterion in Real Data experiments

The combination MiniAlexNet+MNIST

batchsize	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	4	99.23%	-		MPB	MP	99.49%
32	20	99.42%	-		MP	MP	99.41%
64	-		-		MP	MP	99.42%
128	-		-		MP	MP	99.39%
256	-		-		MP	MP	99.31%

The combination MiniAlexNet+MNIST

batchsize	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	5	98.64%	-		MPB	MP	99.49%
32	-		-		MP	MP	99.41%
64	-		-		MP	MP	99.42%
128	-		-		MP	MP	99.39%
256	-		-		MP	MP	99.31%

Application of spectral criterion in Real Data experiments

The combination MiniAlexNet+CIFAR10

batchsize	spectral criterion $C = 0.4$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	3	69.05%	9	72.02%	HT	RC	10%(explode)
32	4	72.17%	16	74.64%	HT	RC	10%(explode)
64	5	71.61%	28	76.35%	HT	MP	77.94%
128	10	74.14%	-		HT	MP	77.43%
256	24	75.70%	-		MPB	MP	75.93%

The combination MiniAlexNet+CIFAR10

batchsize	spectral criterion $C = 0.6$				Final Epoch 248		
	epoch(FC1)	Test Acc	epoch(FC2)	Test Acc	FC1	FC2	Test Acc
16	4	71.01%	36(RC)	55.6%	HT	RC	10%(explode)
32	4	72.17%	196(RC)	62.84%	HT	RC	10%(explode)
64	6	73.03%	-		HT	MP	77.94%
128	12	74.31%	-		HT	MP	77.43%
256	28	75.87%	-		MPB	MP	75.93%

Conclusions

- We find a connection between spectrum type of weight matrices in a DNN and the degree of difficulty of the classification task at hand.
  The more difficult to classify, the higher chance for the Heavy Tail to emerge.

- The classification difficulty can depend on the data SNR, the number of classes and the complexity of the data;

- Further, we show the evidence that the weight matrices spectra could be regarded as a training information encoder; especially when HT appear.

- The spectral analysis in the weight matrices provides a new way of understanding the whole training procedure. We propose a spectral criterion in order to avoid over-training (in case of poor data quality).

Future works

- Why the spikes in weight matrices match class number well?
- How SGD generates Heavy Tail in the poor data quality in DNNs?
- Beyond our empirical findings, is it possible to give a model that generates the observed phase transition in weight matrix spectra?

[1] Madhu S. Advani, Andrew M. Saxe, and Haim Sompolinsky. "High-dimensional dynamics of generalization error in neural networks". In: Neural Networks 132 (2020), pp. 428–446. ISSN: 0893-6080.

[2] Liam Hodgkinson and Michael Mahoney. "Multiplicative noise and heavy tails in stochastic optimization". In: International Conference on Machine Learning. PMLR. 2021, pp. 4262–4274.

[3] Charles H. Martin and Michael W. Mahoney. Implicit Self-Regularization in Deep Neural Networks: Evidence from Random Matrix Theory and Implications for Learning. 2018. arXiv: 1810.01075 [cs.LG].

[4] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. 2020. arXiv: 1908.05355 [math.ST].

[5] Jeffrey Pennington and Yasaman Bahri. "Geometry of Neural Network Loss Surfaces via Random Matrix Theory". In: Proceedings of the 34th International Conference on Machine Learning. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 2798–2806.

[6] Mohamed El Amine Seddik et al. Random Matrix Theory Proves that Deep Learning Representations of GAN-data Behave as Gaussian Mixtures. 2020. arXiv: 2001.08370 [cs.LG].