GENERALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Since September 2019, I was reading about Robustness+ Network Compression. The more I am reading, the more I get that there is a strong correlation between robustness/stability, generalization, generalization bounds, and compression. I found some interesting papers and talks regarding this and I will cover them here.

GENERALIZATION AND COMPRESSION

The whole story starts with figure **??** . While we expect that with more parameters in model we have less generalization, in Deep Neural Network it is not the case.

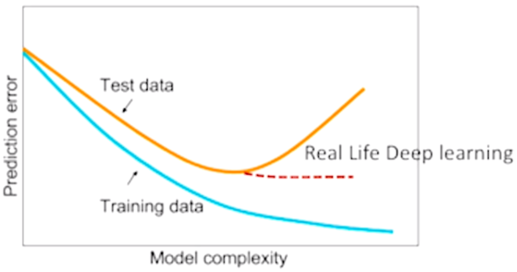
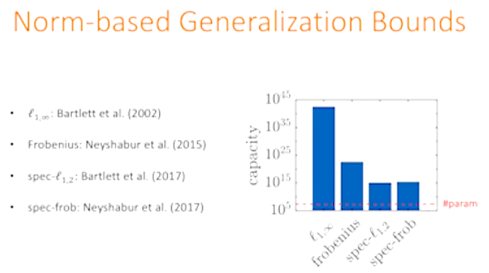


Figure 1: Generalizaion gets better with overparametrized deep neural networks(Arora et al. (2018))

Guiding Q: Why is it a good idea to train VGG19 (20M parameters) on CIFAR 10?
In order to solve this mystery we might have a look on generalization bounds in ML:

$$error_{test} \leq error_{training} + c\sqrt{\frac{capacity}{m}} \tag{1}$$

In which, the capacity roughly corresponds to the number of parameters and m is the number of training samples. However equation 1 is not working for DL because the number of params are way more than m(very loose bound). Many other bounds proposed but their common problem is that they are way more than number of params.


Norm-based Generalization Bounds

- $\ell_{1,\infty}$: Bartlett et al. (2002)
- Frobenius: Neyshabur et al. (2015)
- spec-$\ell_{1,2}$: Bartlett et al. (2017)
- spec-frob: Neyshabur et al. (2017)

Figure 2: Norm based generalization bounds are loose, alot more than number of parameters (Arora et al. (2018))

Arora et al. (2018) tries to answer this question: What property of network trained on real data can help to sharpen this bound? They defined **noise stability** of trained neural network: How injected gaussian noise at a layer affects higher layers? *i.e.*, noise propagation in layers. Noise stability Arora (2018): add gaussian η to output x of a layer ($|\eta| = |x|$), then measure change in higher layers, if small then network is **noise stable**. They injected gaussian noise into VGG. Fig figure **??** shows the result. Even for larger noise, i.e. close to the original signal, the factor of the noise is disappearing. They emprically found that noise stability correlates strongly with generalization error during training.

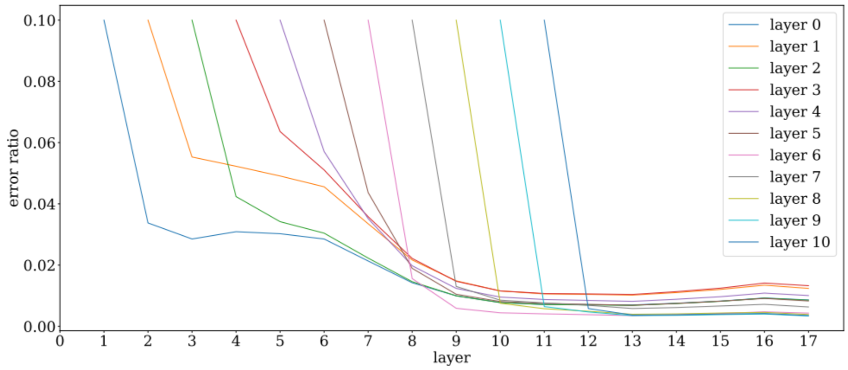


Figure 3: Attenuation of injected noise on a VGG-19 net trained on CIFAR-10. The x-axis is the index of layers and y-axis denote the relative error due to the noise ($\frac{\|\hat{x}^i - x^i\|_2}{\|x^i\|_2}$). A curve starts at the layer where a scaled Gaussian noise is injected to its input, whose l_2 norm is set to 10% of the norm of its original input. As it propagates up, the injected noise has rapidly decreasing effect on higher layers, i.e. The injected noise to higher layers get attenuated very much. This property is shown to imply compressibility (Arora et al. (2018))

noise stability to bound effective capacity. How do trained nets achieve noise stability? What does it have to do with generalization? They propose that when we prune the network we inject noise to the model.

- noise stability of one layer Arora (2018): Assume that there is no nonlinearity, *i.e.*, fully connected layer (Matrix weight M). This layer is noise stable iff ($\frac{|Mx|}{|x|} \gg \frac{|M\eta|}{|\eta|}$).
  What does this mean mathematically? $\frac{|Mx|}{|x|}$ is uppber bounded by the **top (larger) singular values** of matrix M, so $\frac{|Mx|}{|x|} \geq \sigma_{max}(M)$. As the gaussian noise is evenly distriuted in all directions, *e.g.*, singular directions, a simple calculation shows that the right term is related to the l_2 **norm of singular values** , *i.e.*, $\frac{|M\eta|}{|\eta|}) = \frac{\sqrt{\sum_i \sigma_i(M)^2}}{\sqrt{n}}$. This suggests that the singular values are concentrated. As figure **??** shows, there are few large singular values, and most of them (thousands) are close to zero. These are close to zero, but exactly zero, so they may contribute to the norm, but they are very few large singular values. Dicrading such small values is not a good idea.
  They propose a new compression method, in which at the end number of parameters $\ll$ number of data points (before compression, number of parameters $\gg$ number of data points). Important: compression method allowed to use any number of new random bits, provided they do not depend on data.
  proof sketch : noise stanility $\rightarrow$ deep net compressible with minimal change to training error.
  - idea 1: **Compress** a layer(randomized in such a way that error introduced are "Gaussian like"). Compression algorithm is like Random Linear Projection, where we take the projection of layer matrix on to these sign matrices (Nontrivial extension to convolutional nets).
  - Errors (gaussian noise) **attenuate** as they go through network, as noted earlier.

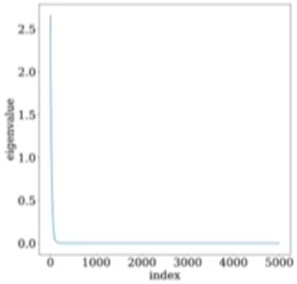


Figure 4: Distribution of singular values in a filter of layer 10 of VGG19. Such matrices are **compressible**(Arora (2018))

Algorithm 1 Compression

```
Generate $K$ random sign matrices $M_1, ..., M_k$ (important: picked before seeing the data)
$\hat{A} = \frac{1}{k} \sum_i^k < A, M_t > M_t$
```

The Quantitatie Bound. we can measure the below definitions on the desired network, and this gives us an upper bound on the capacity of such network. These are the properties that allow noise stability.

$$capacity \approx (\frac{depth \times activationcontraction}{layercushion \times interlayercushion})^2, \tag{2}$$

The Qulitatie Check. Correlation with Generalization.

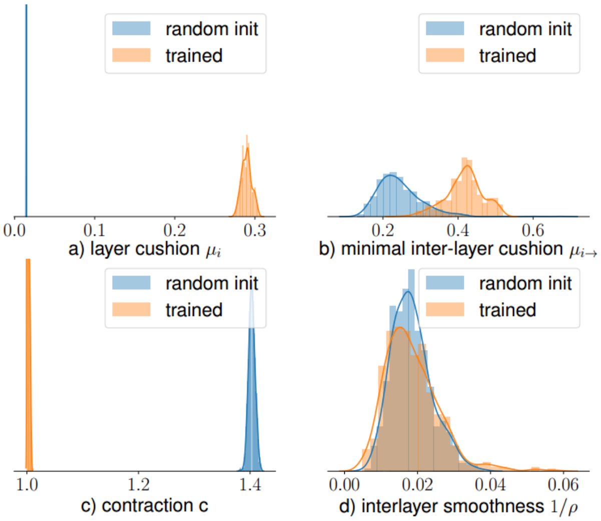


Figure 5

The Qulitatie Check. Correlation with Generalization.

References

Sanjeev Arora. Toward theoretical understanding of deep learning (icml 2018 tutorial), 2018. https://www.youtube.com/watch?v=rcR6P5O8CpU [Accessed: 07.10.2019].

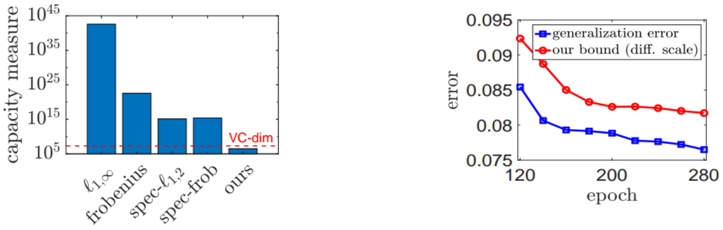


Figure 6: Correlation with Generalization

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. *arXiv preprint arXiv:1802.05296*, 2018.

Appendix

You may include other additional sections here.