10-605 Deep Double Descent

Introduction

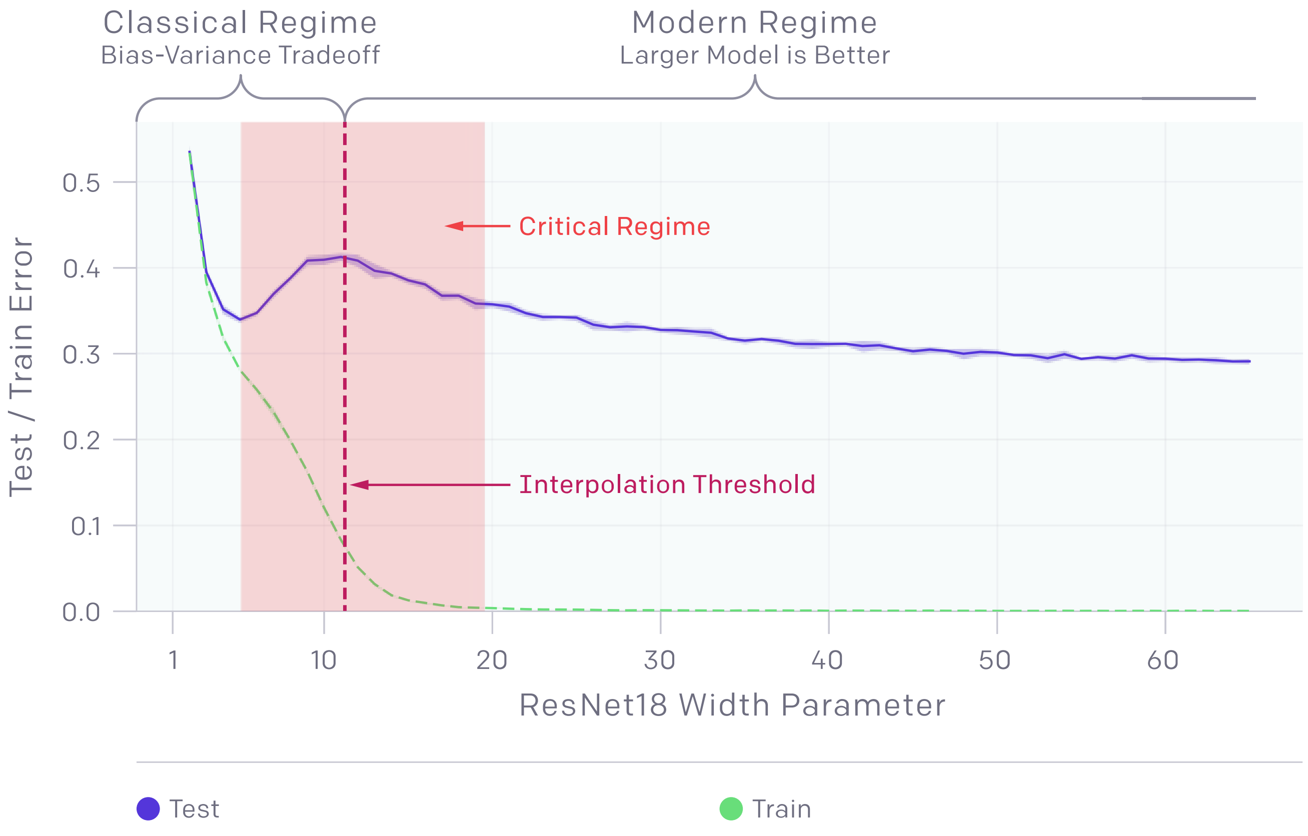
As the world increasingly relies on more automated ways of improving processes using machine learning, we see a greater number of models rely on more data to create the most accurate predictions and classifications and capture various trends. Within applications like Natural Language Processing and Computer Vision, more complex models are being used to understand intent, which results in large corpuses of texts being used. These corpuses are coupled with very complex models to solve a variety of tasks, from language generation to image processing. However, with larger datasets and more complex deep learning models such as CNNs and ResNets, we have to understand our limitations when it comes to the sample size used for training and the number of parameters included in our model. Within this blog, we will cover the phenomenon of "Double Descent," which further illustrates the tradeoff between training and testing error and model complexity. Understanding double descent will allow us to create the most optimized models to solve a variety of tough problems.

Background

"Deep Double Descent: Where Bigger Models and More Data Hurt," unravels the behavior between model complexity and training data size. Many conventional theories of machine learning, such as "more data is always better," are challenged in this paper using evidence that modern deep learning models have different behaviors in different regimes. This paper addresses different circumstances where model performance could either improve or decline based on the complexity of the model and defines a new measurement called effective model complexity. EMC is explained as the maximum number of samples on which the model can achieve close to zero training error. EMC is also defined mathematically below:

equation

A common concept in statistics is that there is a bias-variance tradeoff: higher complexity models have lower bias and higher variance, while lower complexity models have higher bias and lower variance. When model complexity is low and the number of training samples is high, deep learning models exhibit the typical bias-variance tradeoff. This is called the under-parameterized regime. However, when model complexity is large compared to the number of samples, then increasing model complexity lowers training error. This phenomenon, first defined by Belkin et al. (2018), is called double descent.



From https://openai.com/blog/deep-double-descent/ (https://openai.com/blog/deep-double-descent/)

"Double descent" occurs after a point of interpolation, which is when the training error decreases and approaches zero while the test error increases. The point of interpolation is also the point where the model perfectly fits the dataset. This means that as the model complexity increases (or as EMC increases on the x-axis), we start to see the train and test error decrease. The figure above shows a Res18 neural network based model trained on different complexity parameters and demonstrates how double descent occurs after the point of interpolation. Visually, we can see the direct relationship between model complexity, testing error, and training error. As the model increases in complexity, the training error consistently decreases but the test error initially decreases and then increases due to the model being in the "under-parameterized" regime. Within the classical regime, deep learning models will typically follow this behavior due to the bias-variance tradeoff. However, after the point of interpolation, the training error will continue to approach zero while the test error will experience a second descent after the interpolation threshold.

As model complexity increases, we see three different states where the model fluctuates in test error, although the training error is consistently decreasing. These states relate to the “Deep Double Descent” hypothesis. This paper describes three different states for a neural network based training process regime; below are the formal definitions for the three regimes.

Under-Parameterized Regime: Given the case where the Effective Model Complexity is smaller than the size of the dataset, then any training procedure that increases the effective model complexity (such as adding more parameters) will decrease the test error.

Over-Parameterized Regime: Given the case where the Effective Model Complexity is greater than the size of the dataset, then any training procedure that increases the effective model complexity (such as adding more parameters) will decrease the test error.

Critically Parameterized regime: A set of parameters where the behavior of test error is uncertain since increasing model complexity can result in a decrease or increase in test error.

Experimental Setup

This paper builds on previous research done by Belkin et al. (2019), who defined the double descent phenomenon for decision trees and 2-layer neural networks. Further, it shows that double descent occurs in many other cases, such as for CNNs, ResNets, and transformers.

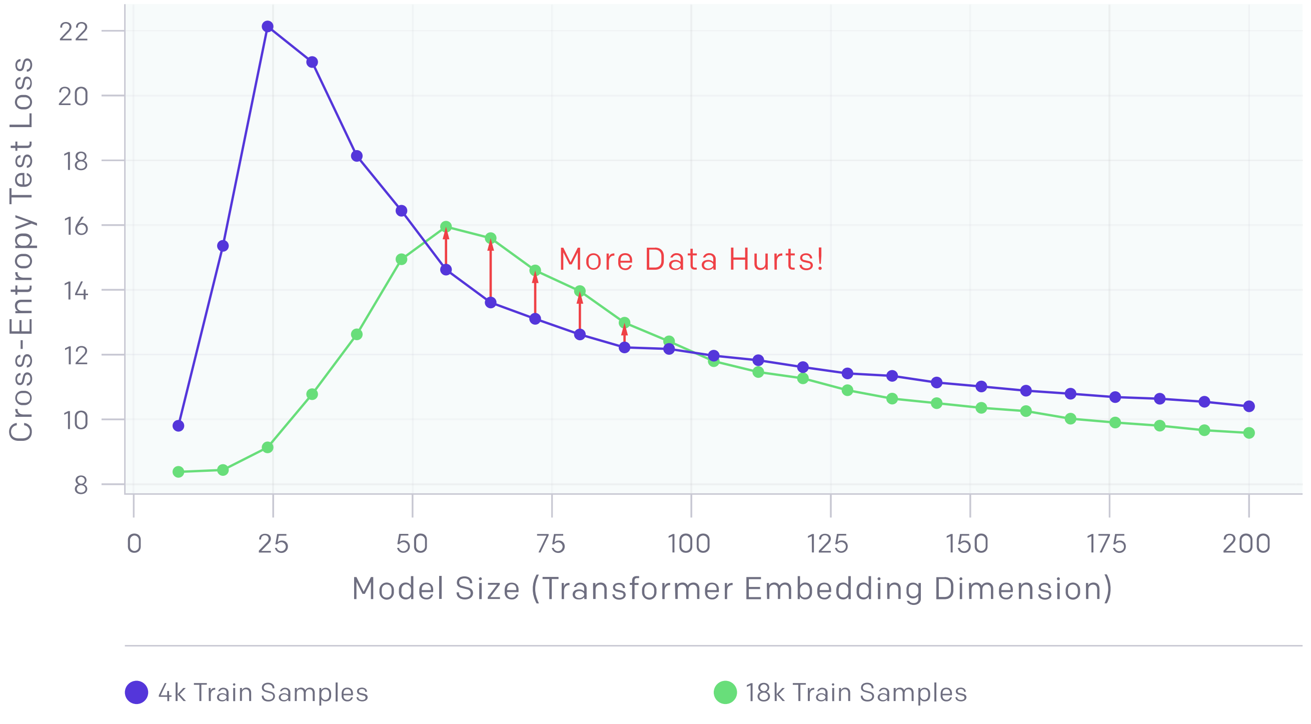
This paper tests the double descent hypothesis on three different families of architectures: ResNets, Standard CNNs, and Transformers. All models were trained using Adam, which is an optimization algorithm typically used to train deep neural networks, with a learning rate of 0.0001. This optimization algorithm and learning rate were used because they optimize well and do not require hyperparameter tuning; however, the authors found that these results generalize to other optimizers and learning rates.

Results and Discussion

The authors trained these three models on several different datasets, such as the IWSLT ’14 German to English dataset and the WMT ’14 English to French dataset, and calculated the training and testing error. They obtained the following results:

Firstly, the paper identifies the critical regime, which is a region where increasing model complexity leads to an increase in training and testing error. In this regime, models are barely large enough to fit the training data. The location of the critical regime can be affected by the type of optimization algorithm used, the number of training samples, or the amount of label noise. Notably, the model-wise double descent phenomenon is most easily observed when there is added label noise.

Secondly, the paper also finds that there is a regime where increasing the sample size increases the cross-entropy test loss. This occurs in the critical region, when the model size is just large enough to accommodate the number of training samples. In this region, adding more data to the training sample may not improve, and can even hurt, train and test error. This phenomenon, referred to as sample-wise non-monotonicity, was observed for Transformers trained on language translation tasks. However, this result is not unique to only deep neural networks; adding more data can also negatively impact linear models in this regime.



From https://openai.com/blog/deep-double-descent/ (https://openai.com/blog/deep-double-descent/)

Lastly, the paper discusses epoch-wise double descent, which is a regime where training longer can reverse overfitting. In this regime, for a fixed model size, as the training time increases, test and train error decreases, increases, and then decreases again. This trend can be explained as a function of EMC; when the EMC is smaller than the number of samples, performance follows the classical U-curve, but once the EMC is larger than the number of samples, performance only improves.

The reasons behind these results are not fully understood. However, the intuition is that at the point of interpolation, there are no models that can both interpolate the train set and fit the test set. However, once we get past the point of interpolation, there are many models that can fit the training data, leading to lower train and test error.

Applications

We don’t only notice double descent for standard training procedures for deep learning methods. We can notice this phenomenon in different training methods as well.

From Deep Double Descent: Where Bigger Models and More Data Hurt

From Deep Double Descent: Where Bigger Models and More Data Hurt

The paper mainly covers neural network based models for deep learning. For example, ensemble methods are common techniques used to create highly accurate machine learning models. Another experiment conducted for this paper used an ensemble of five RestNet18 models with label noise to determine whether double descent would occur. The ensemble classifier was determined by a plurality vote from the five base models. The ensemble method error is tested against the standard model, which is a single Resnet18 without label noise. We see on the figure to the left, that the ensemble method reduced the increase in test error within the critical regime. Furthermore, we see that the test error is far less than the standard test error of a single RestNet18 model trained on the dataset with labeled noise. The figure on the right shows the same effect; however, the ensemble models were CNNs and they were not trained on a dataset without labeled noise. We see similar results from the previous ensemble experiment with Resnet19 models. The test error is lower, and the double descent effect is less prominent within the critical regime, with a slight increase at a width parameter of around 10 for the CNN. This experiment proves that even for different types of training procedures, double descent can still occur. For both of the ensembled experiments, we notice that the model goes through the three different regimes outlined in the hypothesis such that when the model complexity increases, there is still a critical regime where the test error increases and then decreases for the ensembled model.

Conclusion

The double descent phenomenon can be observed for a variety of deep learning models trained on different datasets. This phenomenon can be characterized in terms of Effective Model Complexity. The paper finds that due to double descent, there can be a scenario where training on more data hurts model performance. These results are useful because they demonstrate that if the model is just barely able to fit the training data, then changing the model may lead to unexpected behavior, or even a decrease in performance. Furthermore, neural network training typically involves “early stopping,” where training is stopped as soon as the test error fails to improve. With “early stopping,” however, it is difficult (although not impossible) to observe double descent; therefore, future research could focus on where the optimal early stopping point is in relation to double descent.

As datasets grow larger and the field of machine learning advances, it is important to consider how to best train deep neural networks and other machine learning models to obtain the most accurate results. This paper presents an interesting and relevant phenomenon in relation to testing and training deep learning models; fully understanding and exploring the mechanisms behind double descent may provide us with even more insight into the power of machine learning.

Bibliography

@misc{nakkiran2019deep, title={Deep Double Descent: Where Bigger Models and More Data Hurt}, author={Preetum Nakkiran and Gal Kaplun and Yamini Bansal and Tristan Yang and Boaz Barak and Ilya Sutskever}, year={2019}, eprint={1912.02292}, archivePrefix={arXiv}, primaryClass={cs.LG} }