



STANFORD
UNIVERSITY



UNIFYID

DO DEEP NEURAL NETWORKS LEARN SHALLOW LEARNABLE EXAMPLES FIRST?

Karttikeya Mangalam , Vinay Uday Prabhu

Stanford University

mangalam@stanford.edu

Workshop on Identifying and Understanding Deep Learning Phenomena

36th International Conference on Machine Learning 2019

MOTIVATION

Prior work on Characterizing generalization trajectories of deep networks

- U-shaped validation error explained explained with classic bias-variance tradeoff (Vapnik, 1998)
 - Generalization Error decreases until saturation, then overfitting sets in

MOTIVATION

Prior work on Characterizing generalization trajectories of deep networks

- U-shaped validation error explained explained with classic bias-variance tradeoff (Vapnik, 1998)
 - Generalization Error decreases until saturation, then overfitting sets in
- DNNs Learn *simple pattern first* before memorizing (Bengio et al., 2017)

MOTIVATION

Prior work on Characterizing generalization trajectories of deep networks

- U-shaped validation error explained explained with classic bias-variance tradeoff (Vapnik, 1998)
 - Generalization Error decreases until saturation, then overfitting sets in
- DNNs Learn *simple pattern first* before memorizing (Bengio et al., 2017)
- Information Bottleneck: DNNs learn compressed representations of input that maximize the mutual information between the input and the prediction task in a Markov chain (Tishby & Zaslavsky, 2015)

MOTIVATION

Prior work on Characterizing generalization trajectories of deep networks

- U-shaped validation error explained explained with classic bias-variance tradeoff (Vapnik, 1998)
 - Generalization Error decreases until saturation, then overfitting sets in
- DNNs Learn *simple pattern first* before memorizing (Bengio et al., 2017)
- Information Bottleneck: DNNs learn compressed representations of input that maximize the mutual information between the input and the prediction task in a Markov chain (Tishby & Zaslavsky, 2015)
- Input domains consist of a subsets of both task relevant and task irrelevant information and representations first learn to effectively compress the task irrelevant information (Saxe et al. 2018)

RESEARCH QUESTIONS INVESTIGATED

- How similar is the *notion of (classification) easiness* for models with as different parameterizations and architectures as shallow machine learning models and *deep networks*? And hence is attached to the example independently of a model?

RESEARCH QUESTIONS INVESTIGATED

- How similar is the *notion of (classification) easiness* for models with as different parameterizations and architectures as shallow machine learning models and *deep networks*? And hence is attached to the example independently of a model?
- If we are to investigate the examples that a DNN learns to correctly classify over the training batches, *do we observe a shallow learnable to deep learnable regime change*?

RESEARCH QUESTIONS INVESTIGATED

- How similar is the *notion of (classification) easiness* for models with as different parameterizations and architectures as shallow machine learning models and deep networks? And hence is attached to the example independently of a model?
- If we are to investigate the examples that a DNN learns to correctly classify over the training batches, *do we observe a shallow learnable to deep learnable regime change?*
- Are there examples that are shallow learnable but for some reason a DNN with a far better overall accuracy fails to classify? At the heart of this quest is to understand *if shallow learnability is a good proxy for the (classification) easiness of an example.*

EXPERIMENTAL SETUP

- Given a **trained** classical machine learning model M and a **randomly initialized** deep neural network D , we propose to track the training trajectory of D in the following way:

EXPERIMENTAL SETUP

- Given a **trained** classical machine learning model M and a **randomly initialized** deep neural network D , we propose to track the training trajectory of D in the following way:
- After every training step calculate the **contingency matrix** T on the validation/test set:

EXPERIMENTAL SETUP

- Given a **trained** classical machine learning model M and a **randomly initialized** deep neural network D , we propose to track the training trajectory of D in the following way:
- After every training step calculate the **contingency matrix** T on the validation/test set:

	M incorrect	M correct
D incorrect	T_{00}	T_{01}
D correct	T_{10}	T_{11}

test examples that M classifies **correctly** but D after that training step makes a **mistake** on

METRICS TRACKED

- Accuracy of D after each training step and Accuracy of M are straightforward:

$$\text{Accuracy } (M) = \frac{T_{01} + T_{11}}{T_{11} + T_{00} + T_{10} + T_{01}} \quad \text{Accuracy } (D) = \frac{T_{10} + T_{11}}{T_{01} + T_{11} + T_{10} + T_{00}}$$

METRICS TRACKED

- Accuracy of D after each training step and Accuracy of M are straightforward:

$$\text{Accuracy (M)} = \frac{T_{01} + T_{11}}{T_{11} + T_{00} + T_{10} + T_{01}} \quad \text{Accuracy (D)} = \frac{T_{10} + T_{11}}{T_{01} + T_{11} + T_{10} + T_{00}}$$

- Marginal Accuracies of D on M -correct (R_+) and M -incorrect (R_-) subsets can be tracked :

$$R_+ = \frac{T_{11}}{T_{11} + T_{01}} \quad R_- = \frac{T_{10}}{T_{10} + T_{00}}$$

METRICS TRACKED

- Accuracy of D after each training step and Accuracy of M are straightforward:

$$\text{Accuracy (M)} = \frac{T_{01} + T_{11}}{T_{11} + T_{00} + T_{10} + T_{01}} \quad \text{Accuracy (D)} = \frac{T_{10} + T_{11}}{T_{01} + T_{11} + T_{10} + T_{00}}$$

- Marginal Accuracies of D on M -correct (R_+) and M -incorrect (R_-) subsets can be tracked :

$$R_+ = \frac{T_{11}}{T_{11} + T_{01}} \quad R_- = \frac{T_{10}}{T_{10} + T_{00}}$$

- Finally, the Ratio of marginal accuracies R_{\pm}

$$R_{\pm} = \frac{T_{11}T_{10}}{T_{11}T_{10} + T_{11}T_{00} + T_{01}T_{10} + T_{01}T_{00}}$$

DATASETS & MODELS

- **Datasets:** • MNIST • CIFAR10 • CIFAR100

DATASETS & MODELS

- **Datasets:** • MNIST • CIFAR10 • CIFAR100
- **Classical Machine Learning Models:**
 - Support Vector Machine with RBF Kernel
 - Random Forests with early stopping

DATASETS & MODELS

- **Datasets:** • MNIST • CIFAR10 • CIFAR100
- **Classical Machine Learning Models:**
 - Support Vector Machine with RBF Kernel
 - Random Forests with early stopping
- **Deep Learning Models:**
 - 2 Layer CNN (MNIST)
 - DenseNet 121 (CIFAR 10)
 - ResNet 101 (CIFAR 100)

DATASETS & MODELS

- **Datasets:** • MNIST • CIFAR10 • CIFAR100

- **Classical Machine Learning Models:**

- Support Vector Machine with RBF Kernel
- Random Forests with early stopping

- **Deep Learning Models:**

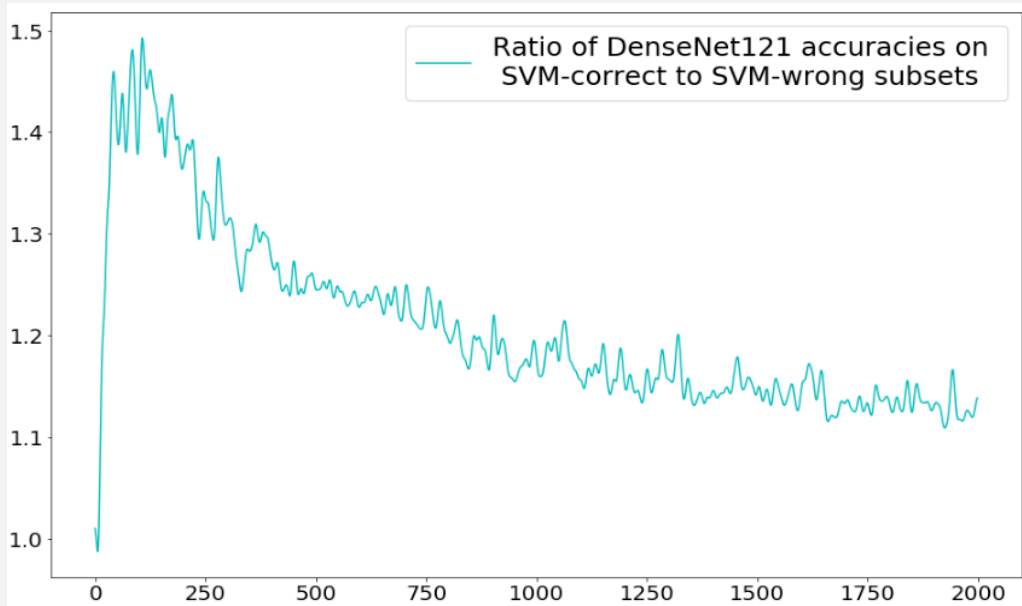
- 2 Layer CNN
- DenseNet 121
- ResNet 101

	MNIST	CIFAR10	CIFAR100
SVM	97.92%	40.08 %	14.42 %
Random Forests	96.14%	35.86 %	14.26 %
Deep Network	98.8%	95.04%	77.78 %
	(2 layer CNN)	(DenseNet121)	(ResNet 101)

Final maximum accuracies achieved by these classifiers

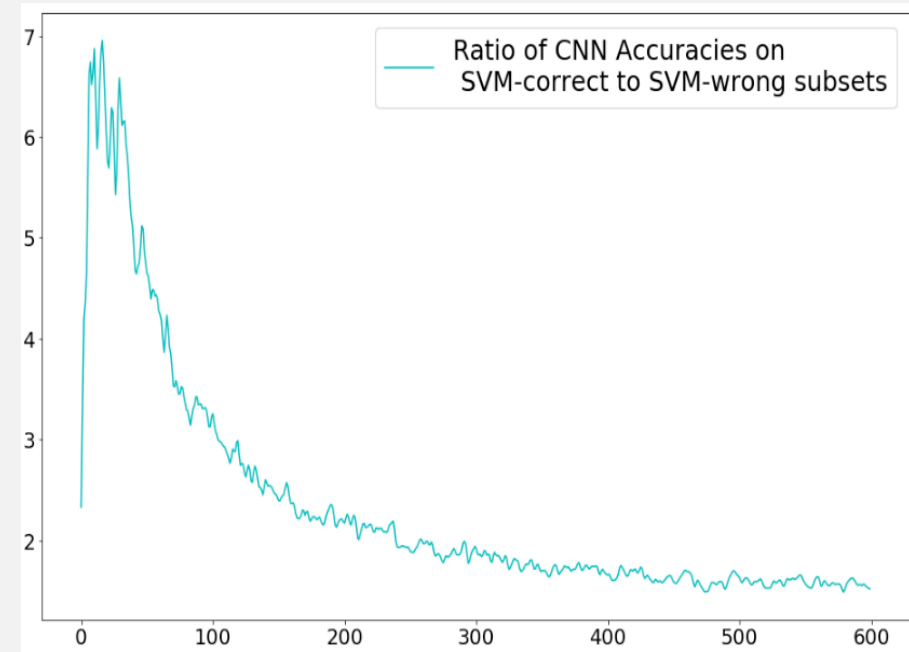
KEY OBSERVATIONS

- Across all the {Dataset, Deep Learning Model (D), Machine Learning model (M)} triplets, the ratio of accuracies retains a **right skewed unimodal shape with a sharp peak**.



R_{\pm} (Ratio of Accuracies)

for {CIFAR10, D = DenseNet121, M = SVM-RBF}



R_{\pm} (Ratio of Accuracies)

for {MNIST, D = 2 layer CNN, M = Random Forest}

KEY OBSERVATIONS

- Across all the {Dataset, Deep Learning Model (D), Machine Learning model (M)} triplets, the ratio of accuracies retains a **right skewed unimodal shape with a sharp peak**.
 - Other reasonable shapes that did not happen:
 - A more or less constant ratio around 1.0
 - ↔
 - M -correct and M -incorrect examples are irrelevant to generalization of D

KEY OBSERVATIONS

- Across all the {Dataset, Deep Learning Model (D), Machine Learning model (M)} triplets, the ratio of accuracies retains a **right skewed unimodal shape with a sharp peak**.
- Other reasonable shapes that did not happen:

A more or less constant ratio around 1.0

↔

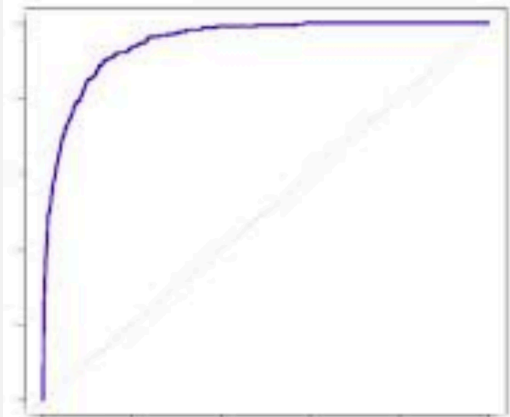
M -correct and M -incorrect examples are irrelevant to generalization of D

Or,

A steady rise from 1.0 to a final value (not 1.0) with no maxima

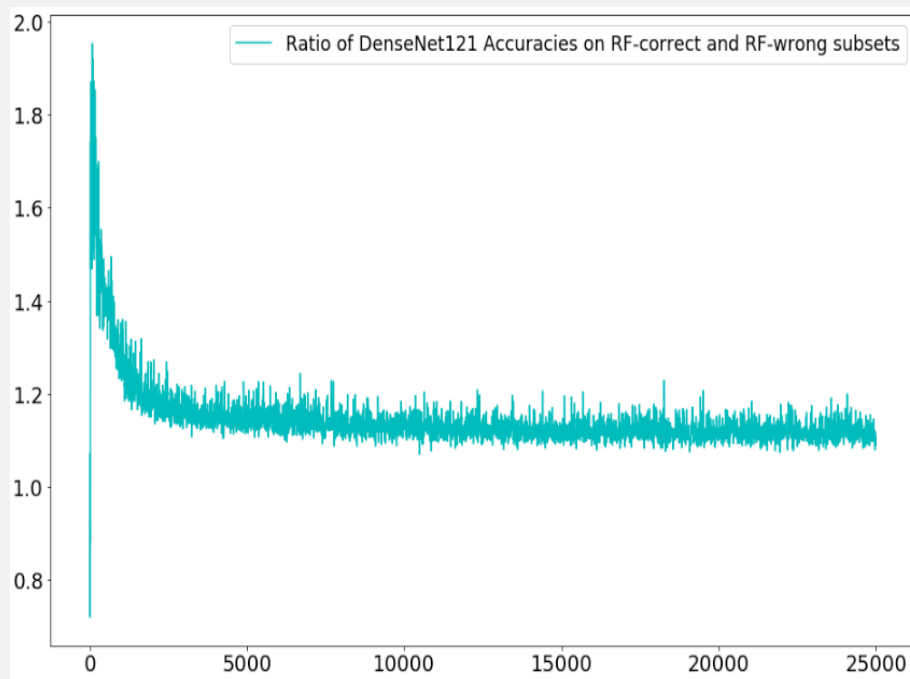
↔

M -correct are easier to generalize to but are learnt concurrently with the M -incorrect examples



KEY OBSERVATIONS

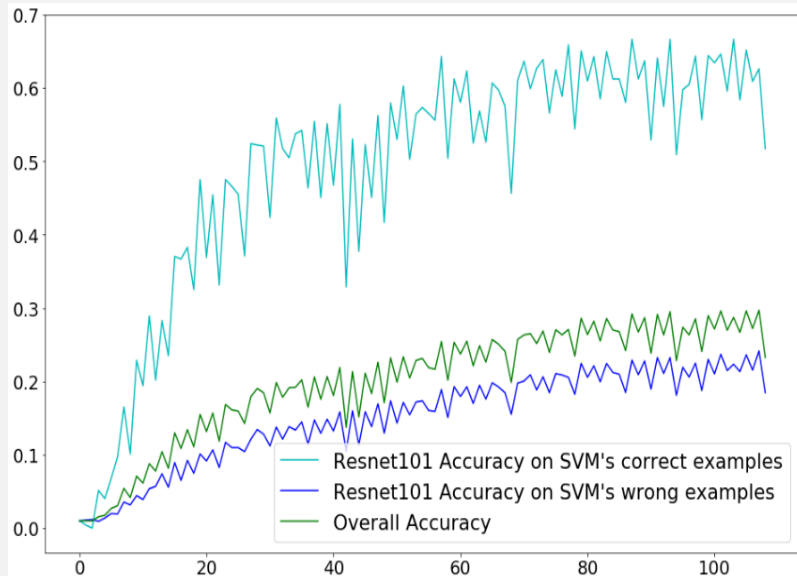
- Across all the {Dataset, Deep Learning Model (D), Machine Learning model (M)} triplets, the ratio of accuracies retains a **right skewed unimodal shape with a sharp peak**.
- The Ratio of accuracies does start from 1.0 (random initialization) but **peaks rapidly** (sometimes as fast as after less than one fifth of the training epoch), **sharply** and **very slowly settles down** to the final value **not equal to 1.0**



R_{\pm} (Ratio of Accuracies)
for {CIFAR10, $D = \text{DenseNet121}$, $M = \text{RF}$ }

KEY OBSERVATIONS

- Across all the {Dataset, Deep Learning Model (D), Machine Learning model (M)} triplets, the ratio of accuracies retains a **right skewed unimodal shape with a sharp peak**.
- The Ratio of accuracies does start from 1.0 (random initialization) but **peaks rapidly** (sometimes as fast as after less than one fifth of the training epoch), **sharply** and **very slowly settles down** to the final value **not equal to 1.0**
 - This implies that even after multiple hundreds of epochs and at convergence, M – correct test examples are more often correct than M – incorrect examples.



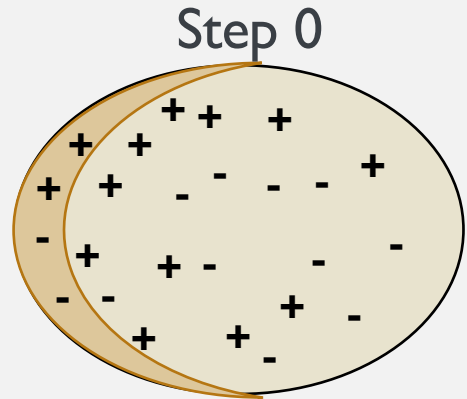
R_+ (Accuracy of D on M -correct),
Overall Accuracy (combined) and R_-
(Accuracy of D on M -incorrect) for
{CIFAR100, Resnet101, SVM} triplet

KEY OBSERVATIONS

- Across all the {Dataset, Deep Learning Model (D), Machine Learning model (M)} triplets, the ratio of accuracies retains a **right skewed unimodal shape with a sharp peak**.
- The Ratio of accuracies does start from 1.0 (random initialization) but **peaks rapidly** (sometimes as fast as after less than one fifth of the training epoch), **sharply** and **very slowly settles down** to the final value **not equal to 1.0**
 - This implies that even after multiple hundreds of epochs and at convergence, M – correct test examples are more often correct than M – incorrect examples.
- Even after convergence T_{01} is non zero (ie there exists examples that M classifies correctly but D gets wrong) on all M, D and all datasets except MNIST.

CONCLUSION

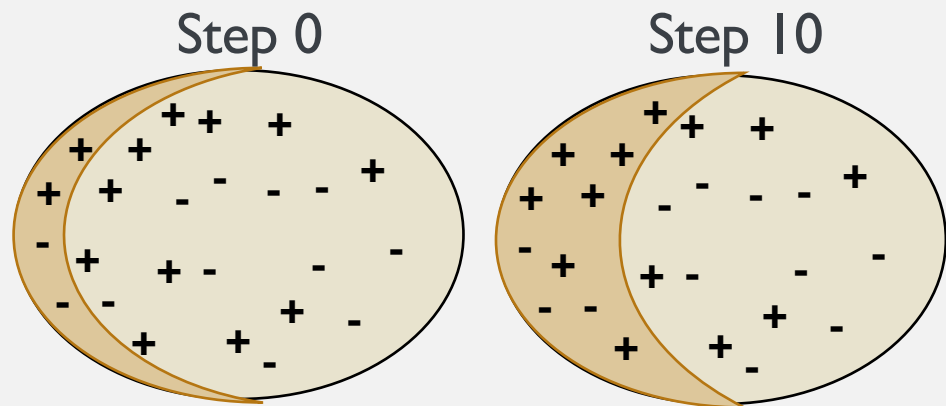
This infographic summarizes our observations succinctly



Random Initialization
Equal number of + & -

Oval represents the test set. +/- represent the test examples correctly/incorrectly classified for the shallow learning model. Finally, Golden(Gray) represents the region correctly(incorrectly) classified by the deep learning model.

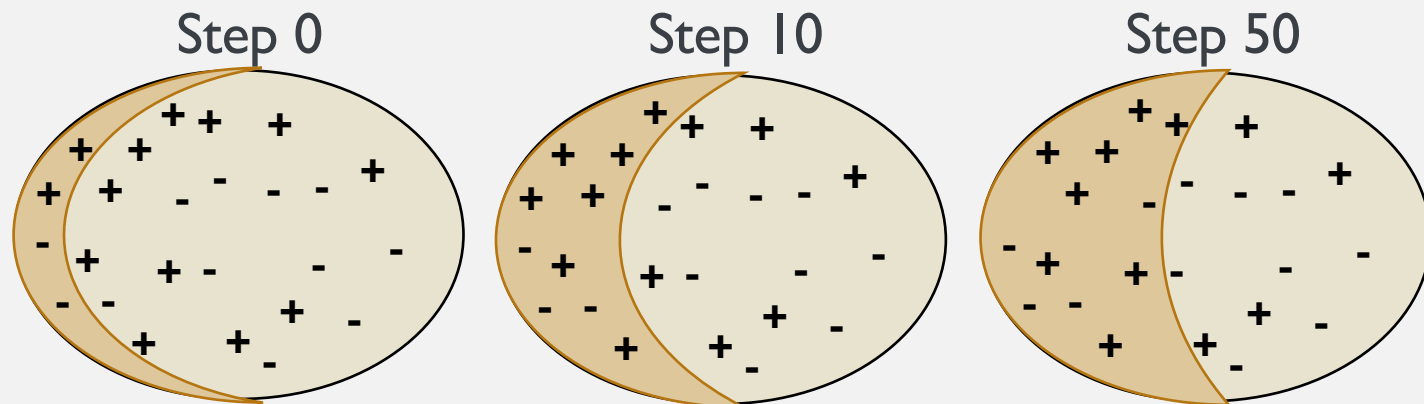
CONCLUSION



Random Initialization Rapid learning of
Equal number of + & - + with few - learnt

Oval represents the test set. +/- represent the test examples correctly/incorrectly classified for the shallow learning model. Finally, Golden(Gray) represents the region correctly(incorrectly) classified by the deep learning model.

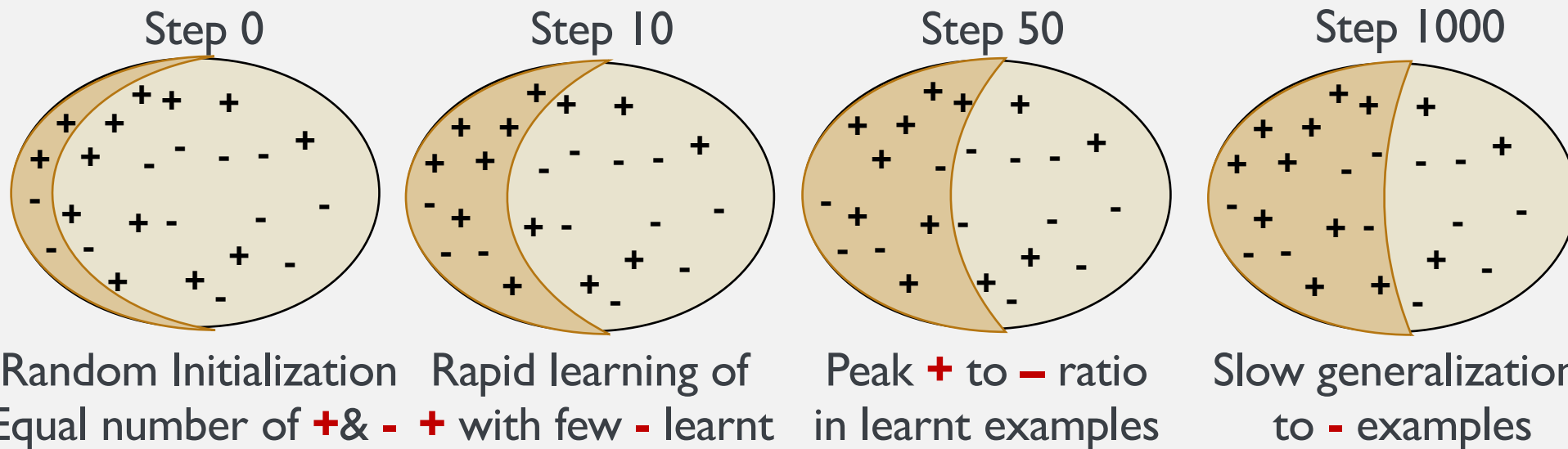
CONCLUSION



Random Initialization Rapid learning of Peak **+** to **-** ratio
Equal number of **+** & **-** **+** with few **-** learnt in learnt examples

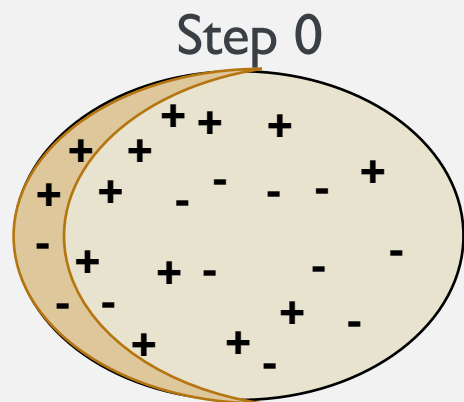
Oval represents the test set. +/- represent the test examples correctly/incorrectly classified for the shallow learning model. Finally, Golden(Gray) represents the region correctly(incorrectly) classified by the deep learning model.

CONCLUSION

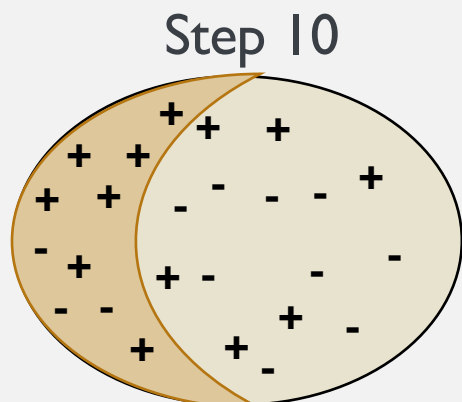


Oval represents the test set. +/- represent the test examples correctly/incorrectly classified for the shallow learning model. Finally, Golden(Gray) represents the region correctly(incorrectly) classified by the deep learning model.

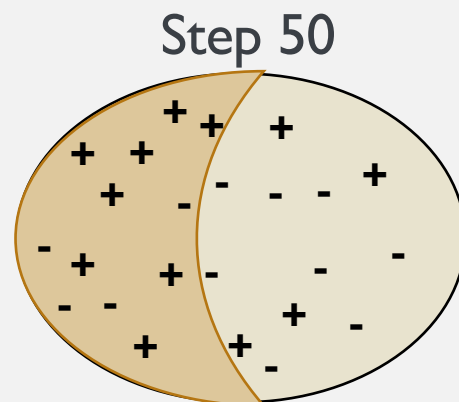
CONCLUSION



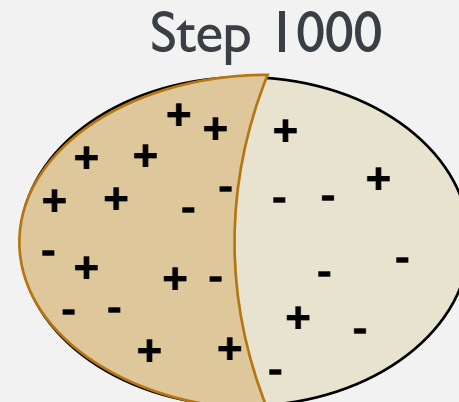
Random Initialization
Equal number of **+** & **-**



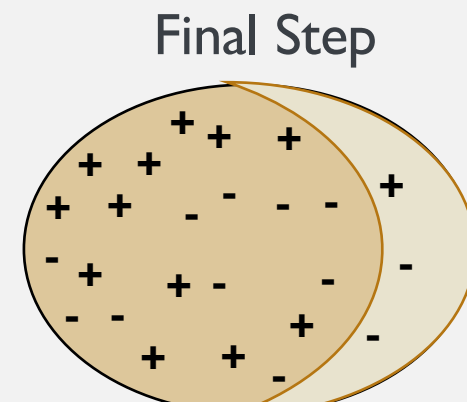
Rapid learning of **+**
with few **-** learnt



Peak **+** to **-** ratio
in learnt examples



Slow generalization
to **-** examples



- more prevalent in
not learnt set than **+**

Oval represents the test set. +/- represent the test examples correctly/incorrectly classified for the shallow learning model. Finally, Golden(Gray) represents the region correctly(incorrectly) classified by the deep learning model.

THANK YOU!



Do Deep Neural Networks Learn Shallow Learnable Examples First ?

Karttikeya Mangalam, Vinay Uday Prabhu
Email: mangalam@stanford.edu



Motivation

What characterizes the generalization process of a deep learning network as training progresses?

- Generalization error decreases first then overfitting sets in
- U-shaped test error curve explained by Bias-Variance tradeoff [1]
- DNNs learn simple patterns first before memorizing [2]
- Input domains consist of a subsets of both task relevant and task irrelevant information and representations first learn to effectively compress the task irrelevant information [3]

Core Questions Investigated

- Is the notion of *easiness for classification same for models* with as different parameterizations and architectures as classical machine learning models and deep networks the same? And hence is largely related to the example independently of model?
- As training progresses, is there a *shallow learnable to deep learnable regime change* viewed through the test set?
- Are there examples that are shallow learnable but for some reason a DNN with a far better overall accuracy fails to classify?

Datasets & Models

- Datasets:**
To study the phenomenon on a wide range of examples we perform experiments on:
 - MNIST • CIFAR10 • CIFAR100
- Classical Machine Learning Models:**
To compare the learning process against different classical machine learning models we use the following models:
 - Support Vector Machine (RBF Kernel) • Random Forests
- Deep Learning Models:**
We choose diverse network architectures to account for *different inductive biases* like skip connections, dense networks etc. and also according to the *dataset simplicity and size*. With these considerations, we study the generalization process of the following three deep learning networks:
 - 2 layer Convolution Neural Network (MNIST)
 - DenseNet 121 (CIFAR10)
 - ResNet 101 (CIFAR100)

Note that each DNN is compared against both the ML models.

Experimental Procedure

Tracking the Learning Process

Traditionally, generalization performance on a held out set is tracked.

Given models **M** and **D** we propose to keep track of the contingency matrix **T** as training of **D** progresses.

Several other interesting metrics are obtained from **T**

- Accuracy**
Accuracy of models **D** and **M** can be found simply as:

	M incorrect	M correct
D incorrect	T_{00}	T_{01}
D correct	T_{10}	T_{11}

$$\text{Accuracy (M)} = \frac{T_{01} + T_{11}}{T_{11} + T_{00} + T_{10} + T_{01}} \quad \text{Accuracy (D)} = \frac{T_{10} + T_{11}}{T_{01} + T_{11} + T_{10} + T_{00}}$$

Marginal Accuracy

Accuracy of **D** on subsets that **M** classifies correct (R_+) & incorrect (R_-)

$$R_+ = \frac{T_{11}}{T_{11} + T_{01}} \quad R_- = \frac{T_{10}}{T_{10} + T_{00}} \quad R_{\pm} = \frac{R_+}{R_-}$$

Ratio of Accuracies

Ratio of marginal accuracies R_{\pm} is also obtained which serves as a measure of how the correctly classified by **D** overlap with the those classified by **M**.

Results & Observations

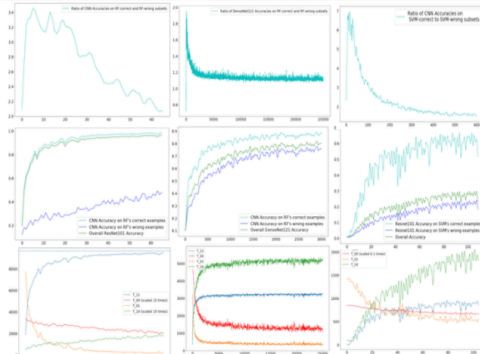


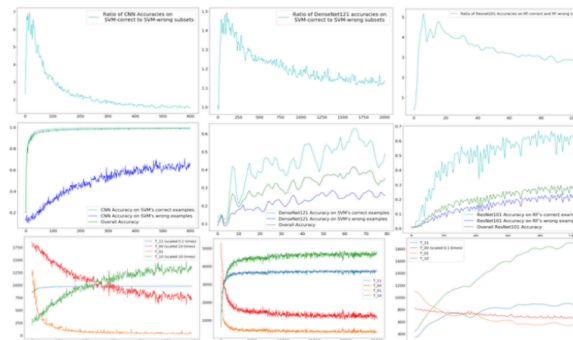
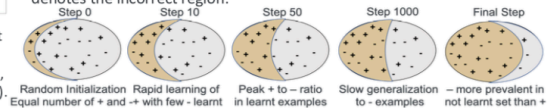
Figure 1. Various metrics tracked as training progresses with **M** as Support Vector Machine. Plots of R_{\pm} (Top Row), Marginal Accuracies (R_+ , R_-) (Middle Row) and **T** (Bottom Row) on the pairs of {MNIST, CNN} (Left Col), {CIFAR10, DenseNet121} (Middle Col) & {CIFAR100, ResNet101} (Right Col).

Key Observations:

- R_{\pm} has a right skewed unimodal shape. Of the two subsets of testing data, **M**-correct and **M**-incorrect were completely irrelevant for generalization process of **D**, R_{\pm} would stay identically at 1.
- Instead, the observed peak indicates that **D** learns **M**-correct examples much earlier in the training than **M**-incorrect. Then slowly over the epochs generalized to *harder* **M**-incorrect set.
- Plots of R_+ , R_- (middle row) validate this observation where R_+ can sometimes be as high as 60% where the overall accuracy is still only 20% and R_- is still around 15%.

Conclusion

The following infographic succinctly expresses our findings. The Oval denotes the entire test set littered with + and - which denote **M** correct and incorrect examples. Finally, golden color denotes the region **D** classifies correctly and gray denotes the incorrect region.



Equivalent Results to Figure 1 with **M** as Random Forests.

Relevant Previous Work

- Vapnik, V. N. Statistical learning theory. Adaptive and learning systems for signal processing, communications and control series, 1998.
- Arpit, D., Jastrzebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., et al. A closer look at memorization in deep networks. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp. 233–242. JMLR [3] Saxe, A. M., Bansal, Y., Dapello, J., Advani, M., Kolchinsky, A., Tracey, B. D. & Cox, D. D. On the information bottleneck theory of deep learning.

Please come to our **poster** for a closer look at the findings.

Our paper can also be found here: <http://bit.ly/icml19>

The code is also available at: https://github.com/karttikeya/Shallow_to_Deep