**UVA CS 6316: Machine Learning : 2019 Fall**
**Course Project: Deep2Reproduce @**
https://github.com/qiyanjun/deep2reproduce/tree/master/2019Fall

# Modern Neural Networks Generalize on Small Data Sets

Preproduced by: Jingyuan Chou & Tairan Song

12/05/2019

# Motivation

- Explain the generalization ability of very high capacity neural networks:
  - Zhang[1] suggest SGD may provide implicit regularization by encouraging low complexity solutions to the optimization.
  - Others Explore the effect of margins on generalization error
- Explain Why the neural network is surprisingly resistant to overfitting
- Large datasets needed for training properly
- The way neural network reaches to variance reduction is mysterious.
- How about on much richer class of small datasets?

# Background

- Great deal of recent research are aimed to explain the generalization ability of very high capacity neural networks:
- There exists limitations in previous experiments, concentrated on a small set of image classification tasks:
  - Over half of the papers in NIPS 2017, ICML 2017
  - MNIST, CIFAR-10, CIFAR-100, ImageNet share same characteristics
  - Similar problem domain, very low noise rates, balanced classes, relatively large training sizes

# Related Work

- Zhang,C.(2017)[1] **Understanding deep learning requires rethinking generalization, ICML**
  - Stochastic Gradient Descent may provide implicit regularization by encouraging low complexity solutions to the neural network optimization problem

- Bartlett P.L et al.(2017).[2] **Spectrally-normalized margin bounds for neural network. NIPS**
- Liang, T. at al.(2017).[3] **Fisher-rao metric, geometry, and complexity pf neural netowrks. arXiv**
  - Explore the effect of margins on generalization error, similar to the margin-based view of Adaboost in the boosting literature that bound test performance in terms of the classifier's confidence in its predictions.
- Other:
  - Investigate the sharpness of local minimum found by training a neural network with SGD

# Claim / Target Task

- The central aim of the paper is to **identify the variance stabilization that occurs when training a deep neural network**
  - **Dedicated to decomposing a neural network into an ensemble of sub-networks (low bias, low variance)**
- **Similar manner as random forest**

# Proposed Solution I

- To establish a view that a network has a natural representation as an ensemble classifier
- Network Decomposition:
  - Given a regular network in a binary classification setting. In the case of a network with L hidden layers, each layer with M hidden nodes:

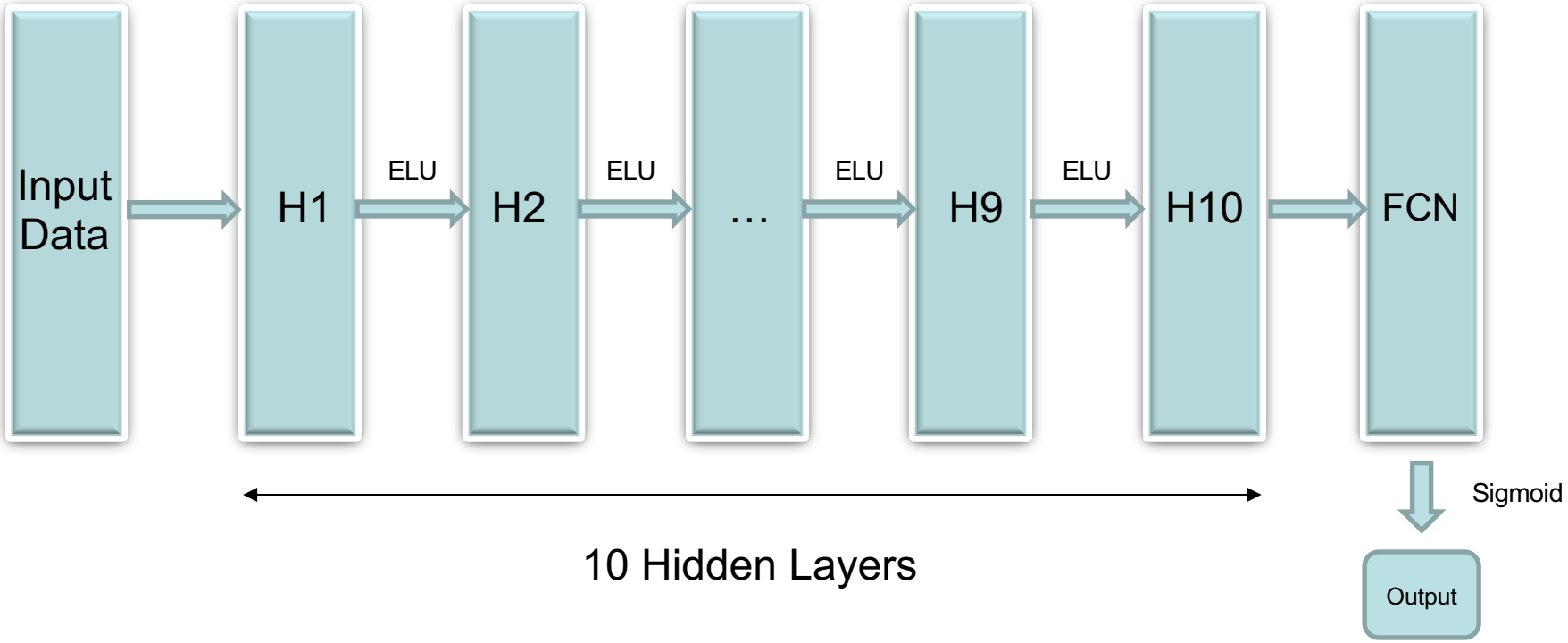
    - $z^{l+1} = W^{l+1} g(z^l), with\ l = 0, \dots \dots, L$
    - $f(x) = \sigma(z^{L+1})$

  - $\sigma$ is sigmoid function, g is activation function, $W^{l+1} \in R^{1*M}, W^1 = R^{M*P}, and\ W^L \in R^{M*M}$ for $l = 2, \dots, L, and\ z^0 = x$
  - In this paper, L = 10, M = 100, g is ELU activation function

# Proposed Solution II

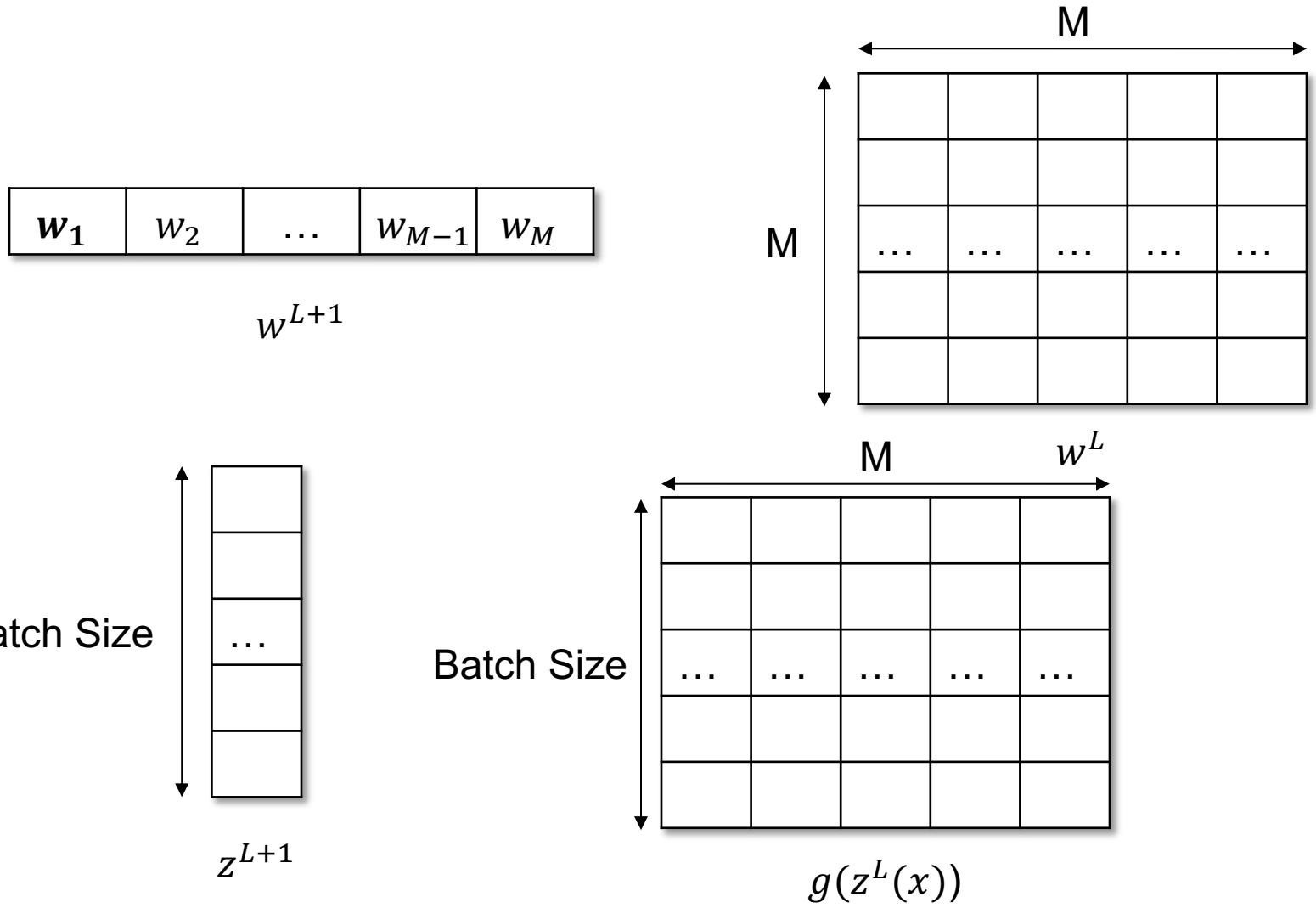
- One way for decomposition is at the final hidden layer:
  - Fix an integer $K \leq M$; and another matrix $\alpha \in R^{M*K}$, with $\sum_{k=1}^{K} \alpha_{m,k} = W_{1,m}^{L+1}$ for m = 1,…..,M
  - Final logit output as a combination of units from the final hidden layer:
    - $Z^{l+1}(x) = W^{L+1} g\left(z^{L}(x)\right)$
    - $= \sum_{m=1}^{M} W_{1,m}^{L+1} g(z_m^l(x))$
    - $= \sum_{m=1}^{M} \sum_{k=1}^{K} \alpha_{m,k} g(z_m^l(x))$
    - $= \sum_{k=1}^{K} \sum_{m=1}^{M} \alpha_{m,k} g(z_m^l(x))$
    - $= \sum_{k=1}^{K} f_k(x)$
  - With $f_k(x) = \sum_{m=1}^{M} \alpha_{m,k} g(z_m^l(x))$
  - In words, we have decomposed the final layer of the network into a sum of component networks at the logit level

# Model (Binary Setting)




**Adapt to Softmax function when under multiclass classification setting**

# Proposed Solution II



# Proposed Solution III

- Ensemble Hunting:
- We want to search for a set of ensemble components that are both **diverse** and **low-bias**.
  - Low-bias: impose restriction that each sub-network achieves very high training accuracy, 100% in the setting, for each sub-network $f_k$
  - Diversity: Desire each in the ensemble should be built from a different part of the full network, to make this happen, require the columns of α are sparse, non-overlapping, and the approach is to just simply force a random selection of half the entries of each column to be zero
  - For each of the K columns of α, sampled integers $(m_{1,k}, m_{2,k} \ldots, m_{M/2,k})$ uniformly without replacement from 1 to M

# Proposed Solution IV

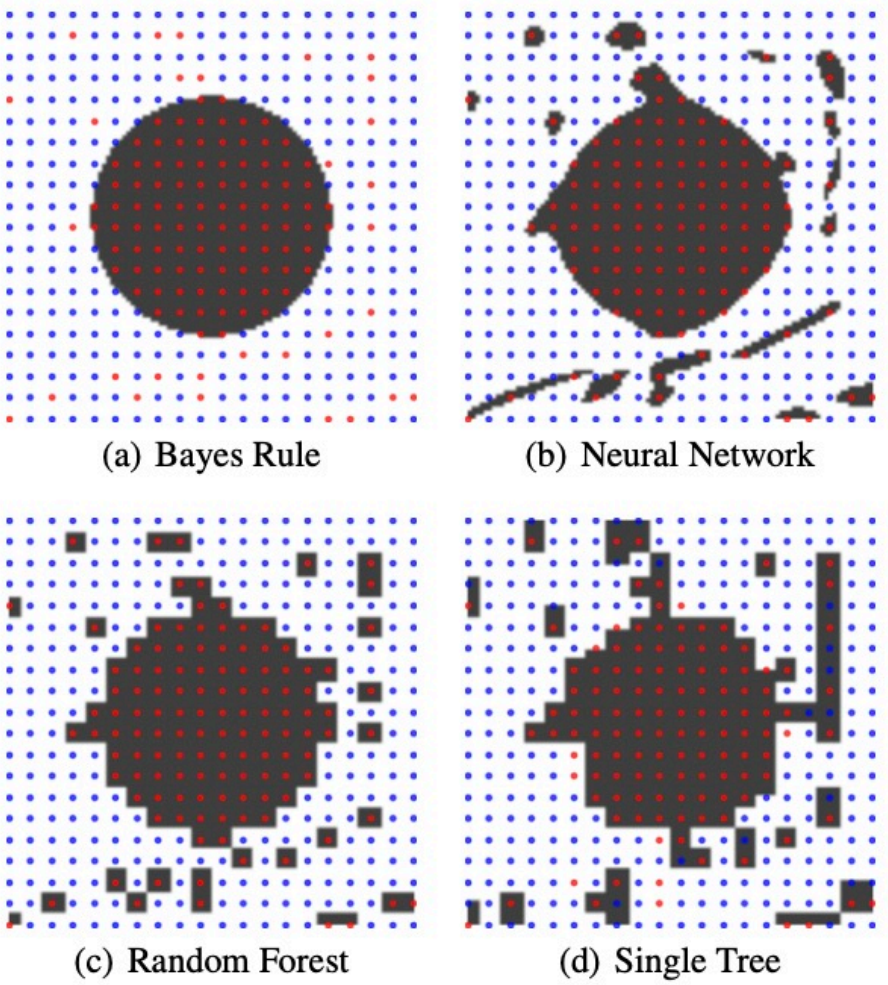
- We need to use linear programming to find a matrix α∈ $R^{M*K}$ that satisfied the required constraints:
  - $\sum_{k=1}^{K} \alpha_{m,k} = W_{1,m}^{L+1}$ , $1 \le m \le M$
  - $\alpha_{m_{j,k}\, k} = 0,\ 1 \le j \le \frac{M}{2}, 1 \le k \le K$
  - $\sum_{m=1}^{M} \alpha_{m,k} g(z_m^l(x))\, y_i \ge 0, 1 \le i \le n, 1 \le k \le K$

# An Intuitive Figure Showing WHY Claim I

Sample 400 points as training set under the following distribution:

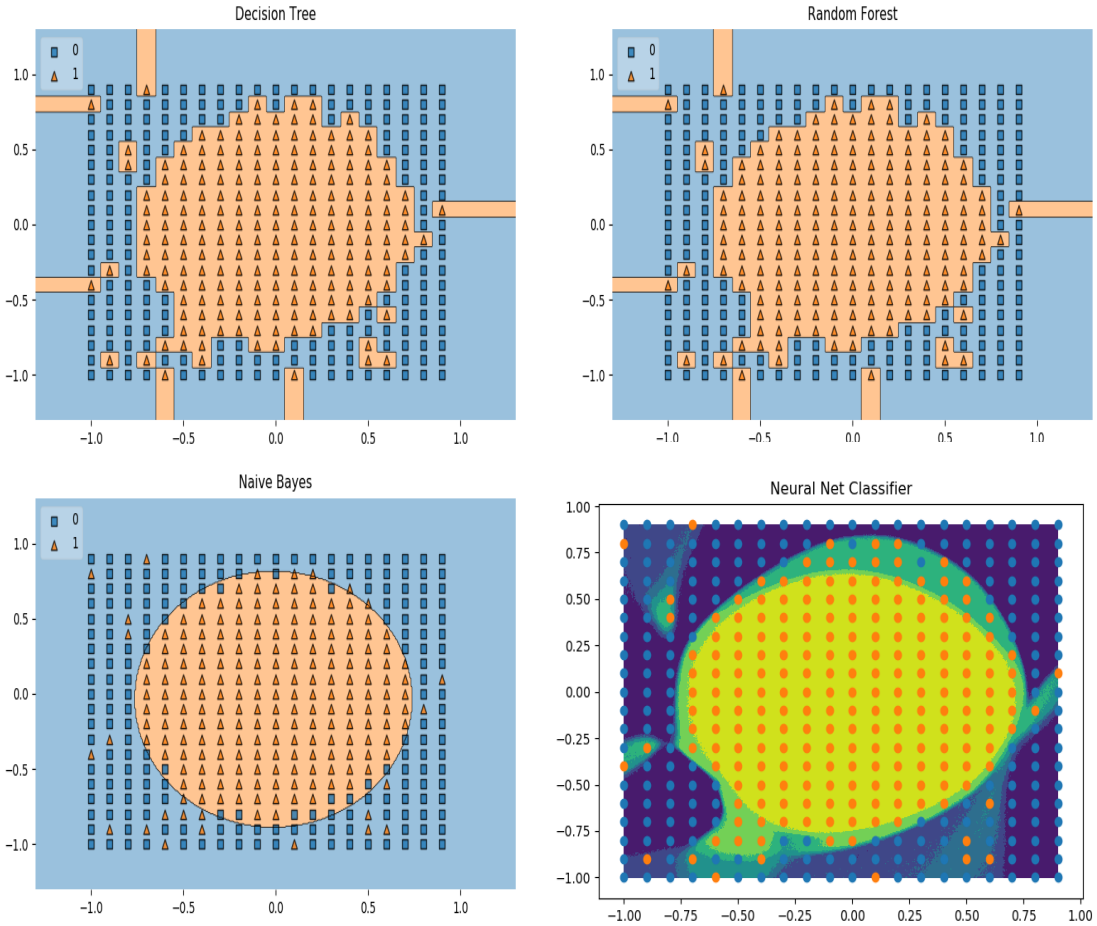
$$p(y = 1|x) = \begin{cases} 1 & ||x||_2 \leq 0.3 \\ 0.15 & otherwise \end{cases}$$

10000 points as test set

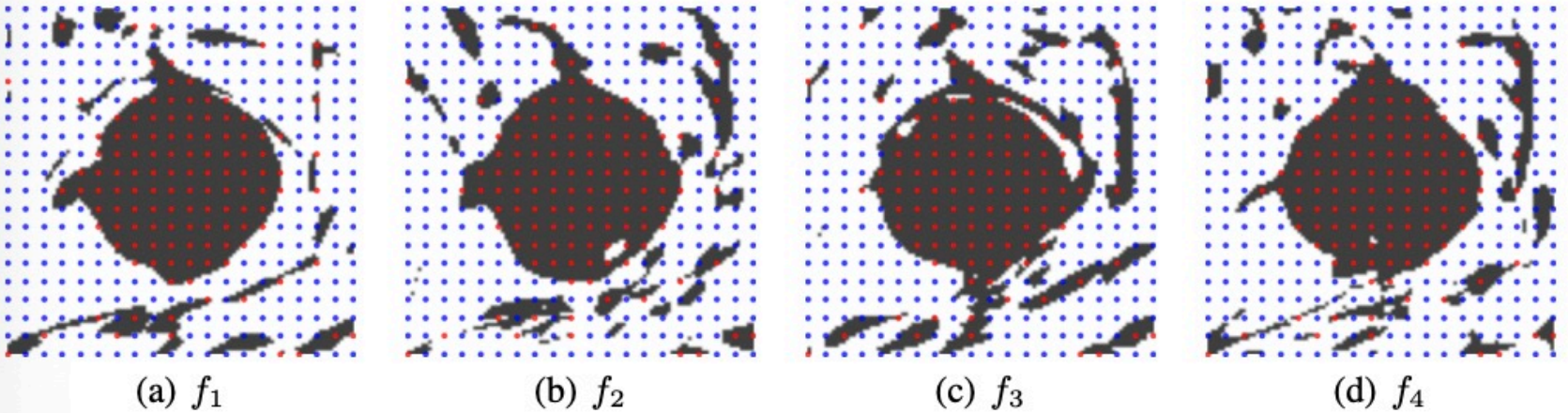


# Decision Boundary for 4 classifiers

Reproduction



# Sub-Network Decision Boundary



(a) $f_1$ (b) $f_2$ (c) $f_3$ (d) $f_4$

(e) $f_5$ (f) $f_6$ (g) $f_7$ (h) $f_8$

# Implementation

- Three classifier trained, random forest classifier; neural networks with dropout; neural networks without dropout
- Neural Networks:
  - 10 hidden layers
  - 100 nodes per layer
  - 200 epochs of gradient descent using Adam optimizer with learning rate of 0.001
  - He-initialization for each hidden layer
  - Elu activation function
  - Dropout with keep rate 0.85, serving as regularization, ridge-type penalty
- Random forest:
  - 500 trees, $\sqrt{p}, p\ is\ the\ number\ of\ features\ in\ the\ dataset$

# Data Summary

- Total 116 small-sized datasets from UCI Data Repository
- Datasets span a wide variety of domains, including agriculture, credit scoring, health outcomes, ecology, and engineering applications etc.
- Highly imbalanced, non-trivial Bayes error rates, discrete features
- The median number of training cases is 601, the smallest only 10 observations.
- Number of features range from 3 to 262, categorical features included in half of the datasets, number of classes range from 2 to 100

# Data Summary

| | CATEGORICAL | CLASSES | FEATURES | N |
|---|---|---|---|---|
| MIN | 0 | 2 | 3 | 10 |
| 25% | 0 | 2 | 8 | 208 |
| 50% | 4 | 3 | 15 | 601 |
| 75% | 8 | 6 | 33 | 2201 |
| MAX | 256 | 100 | 262 | 67557 |

Table 1: Dataset Summary

# Experimental Results

- RF outperforms unregularized NN on 72 out of 116 datasets by small margin, the mean difference in accuracy is 2.4%, with P value less than 0.01 through Wilcox signed rank test

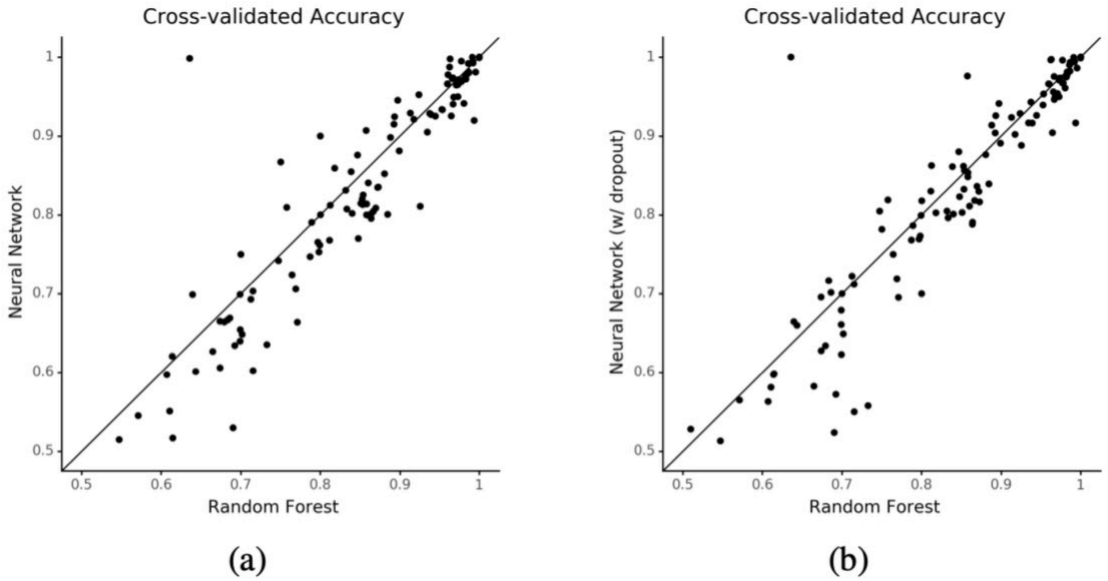



Figure 3: Plots of cross-validated accuracy. Each point corresponds to a data set.

# Experimental Analysis

- Using dropout helps when fitting noisy data sets, it is surprising that the absence of dropout doesn't lead to a collapse in performance.
- Using K=10 to decompose the network, each with 100% training accuracy, applied on datasets with at lease 500 observations, 80-20 train/test split randomly 25 times.
- Errors made by the sub-networks tend to have low correlation, which is the precise motivation for the random forest algorithm

# Conclusion and Future Work

- Large neural network generalize well on small, noisy, data sets.
- neural networks can be trained on small data sets with minimal tuning
- Neural Networks have a natural interpretation as an ensemble of low-bias classifiers whose pairwise correlations are less than one.

- Future work aims to discern a mechanism for the decorrelation observed, and explore the link between decorrelation and generalization

# Main References


- Zhang,C., Bengio ,S., Hardt,M., Recht, B., Vinyals,O.(2017) ***Understanding deep learning requires rethinking generalization, ICML***
- Bartlett P.L., Foster, D.J., and Telgarsky, M.J.(2017). ***Spectrally-normalized margin bounds for neural network. NIPS***
- Liang, T., Poggio. T., Rakhlin, A., and Strokes, J.(2017). ***Fisher-rao metric, geometry, and complexity pf neural netowrks. arXiv***