The Implicit Regularization of SGD

in Least Square Problems and Beyond

Jingfeng Wu, 01/2023

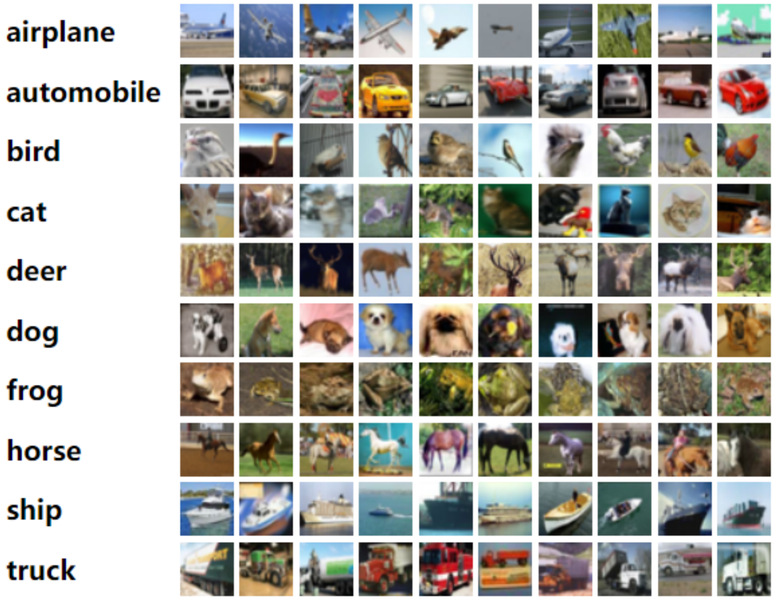
DL 101: How to train DNNs on CIFAR-10?

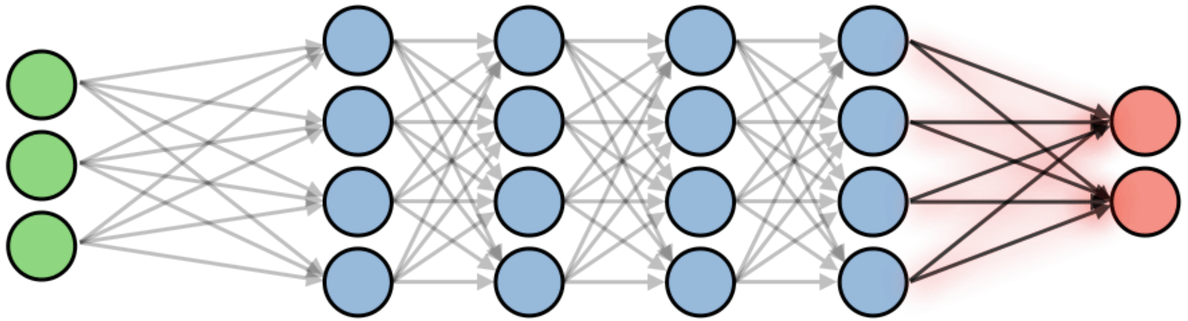
Bag of tricks

- Data augmentation
- Weight decaying
- Dropout
- Batch normalization

…

- SGD (LR decaying + early stopping)





Algorithms induce regularization

- “Unregularized”:

$$w \leftarrow \arg\min L(w)$$

- “Explicit” regularization: weight decaying / ridge

$$w \leftarrow \arg\min L(w) + \lambda \|w\|_2^2$$

- “Implicit” regularization: SGD

$$w \leftarrow \text{SGD}(\eta; \text{dataset})$$

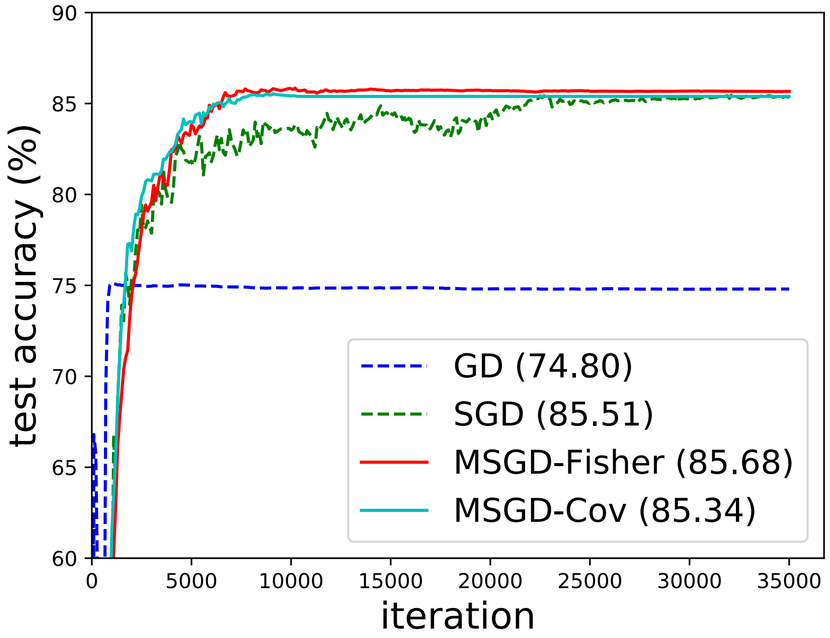
SGD in Practice

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \cdot \nabla \ell(\mathbf{x}_i, y_i; \mathbf{w})$$

- Overparameterized model => Tons of ERM & some may not generalize
- SGD: small batch + large initial LR + LR decaying (early stopping)
- SGD solution generalizes well

Algo implicitly imposes regularization!

But how?



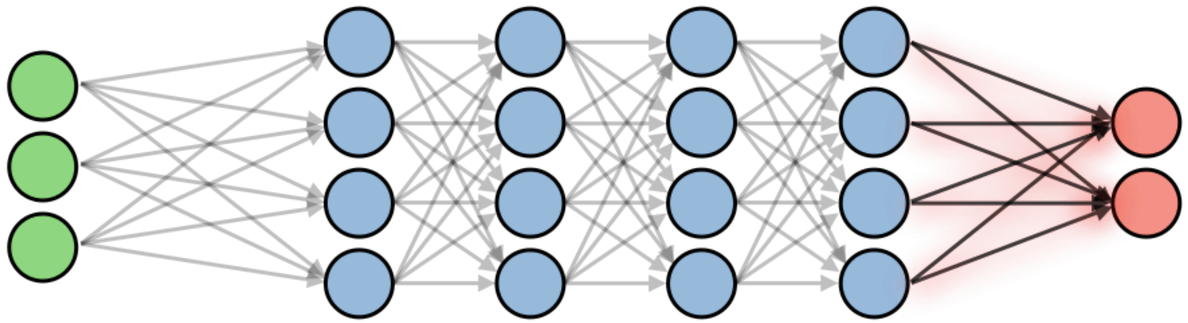
Wu, Jingfeng, Wenqing Hu, Haoyi Xiong, Jun Huan, Vladimir Braverman, and Zhanxing Zhu. "On the noisy gradient descent that generalizes as sgd." In International Conference on Machine Learning, pp. 10367-10376. PMLR, 2020.

Q: For a new problem, which trick to try/tune first?

SGD or WD?

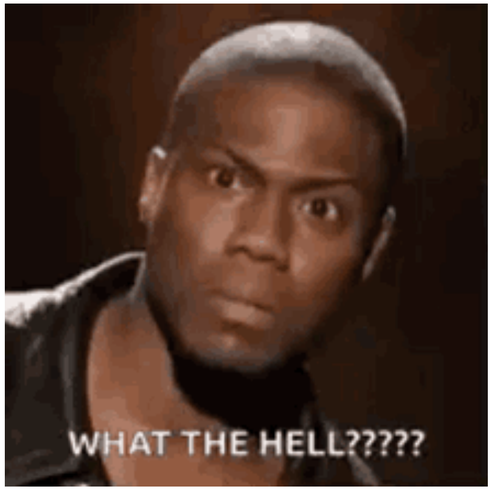
Understand SGD

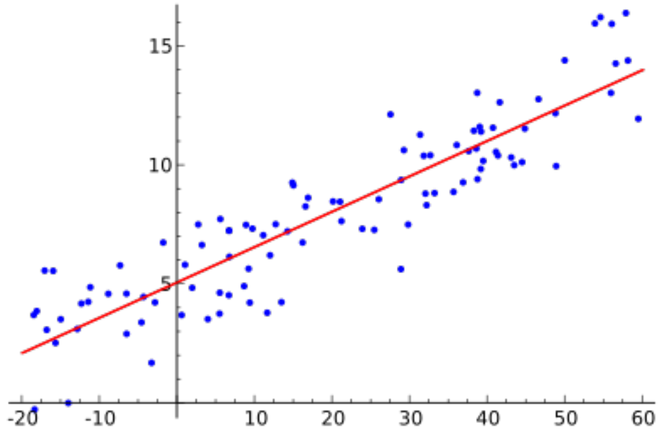
Problem simplification



- # param $> 10^7$
- ReLU non-linearity
- Layer structure

…





Maybe…

Start with modeling high-dim?

Head up: + ReLU shortly

A high-dim linear regression

- d dimensional regression

$$y = \mathbf{x}^\top \mathbf{w}_* + \mathcal{N}(0,1), \quad \|\mathbf{w}_*\|_2 \le 1, \quad \mathbf{x} \sim \mathcal{N}(0, \mathbf{H})$$

- A small training set (iid), $n < d$

$$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$$

- Test error

$$\Delta(\mathbf{w}) := \|\mathbf{w} - \mathbf{w}^*\|_{\mathbf{H}}^2$$

Wait, is this even possible to solve?

In general, no

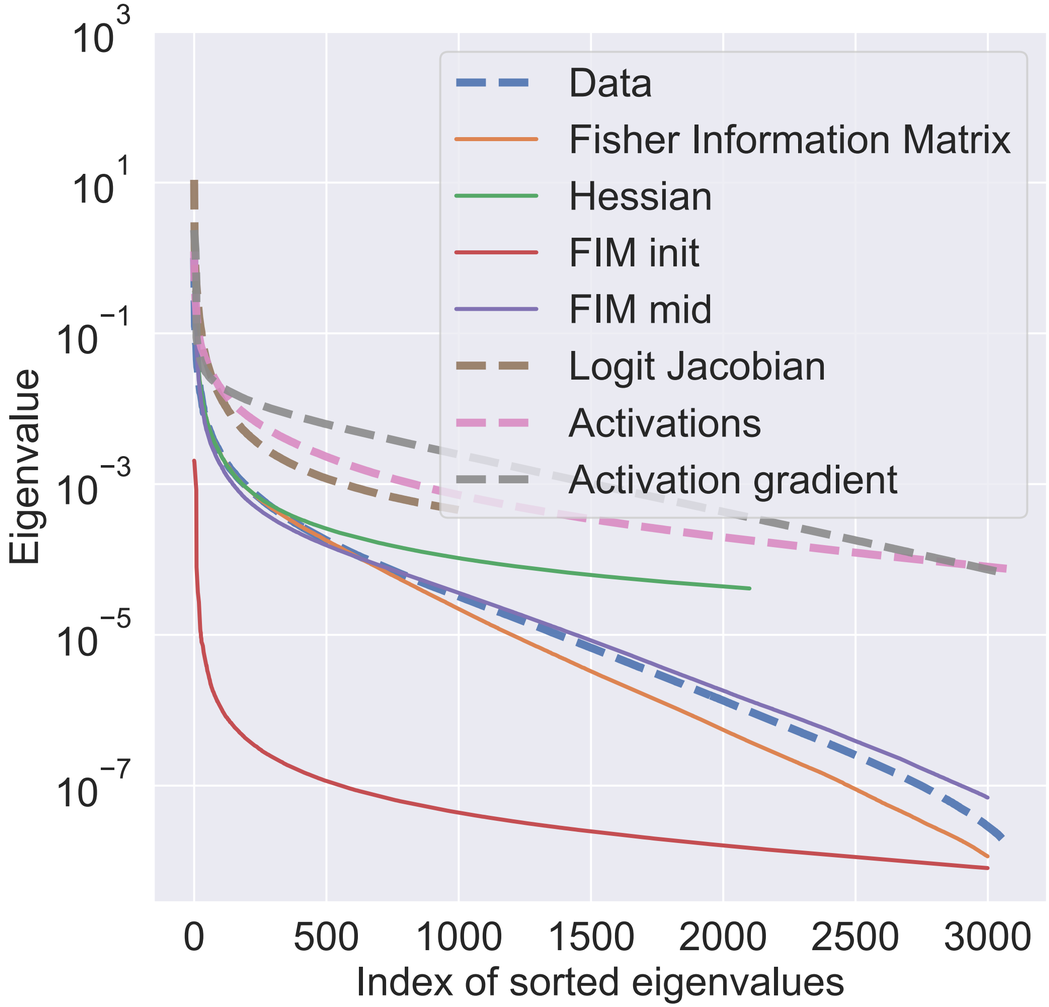
[Classical Result]

When $\mathbf{H} = \mathbf{I}_d$, any reasonable algorithm suffers $\Delta \geq \frac{d}{n} \geq \Omega(1)$

[Intuition]

- probe each dim induces a unit uncertainty
- d important directions to probe

But why we can solve them in practice?



In practice, $\mathbf{H}$ only has a few large eigenvalues.

$$\mathbf{H} \ll \mathbf{I}$$

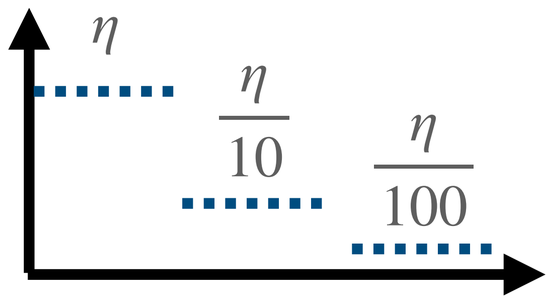
Yang, Rubing, Jialin Mao, and Pratik Chaudhari. "Does the data induce capacity control in deep learning?." In International Conference on Machine Learning, pp. 25166-25197. PMLR, 2022.

Q: How SGD performs when $\mathbf{H} \ll \mathbf{I}$?

Algorithm simplifaction

One-pass SGD (i.e., early stopping)

$$\mathbf{w}_t = \mathbf{w}_{t-1} - \eta_t \cdot (\mathbf{x}_t^\top \mathbf{w}_{t-1} - y_t) \cdot \mathbf{x}_t, \quad t = 1, \ldots, n$$



A Glance to Our Theory

Classical result

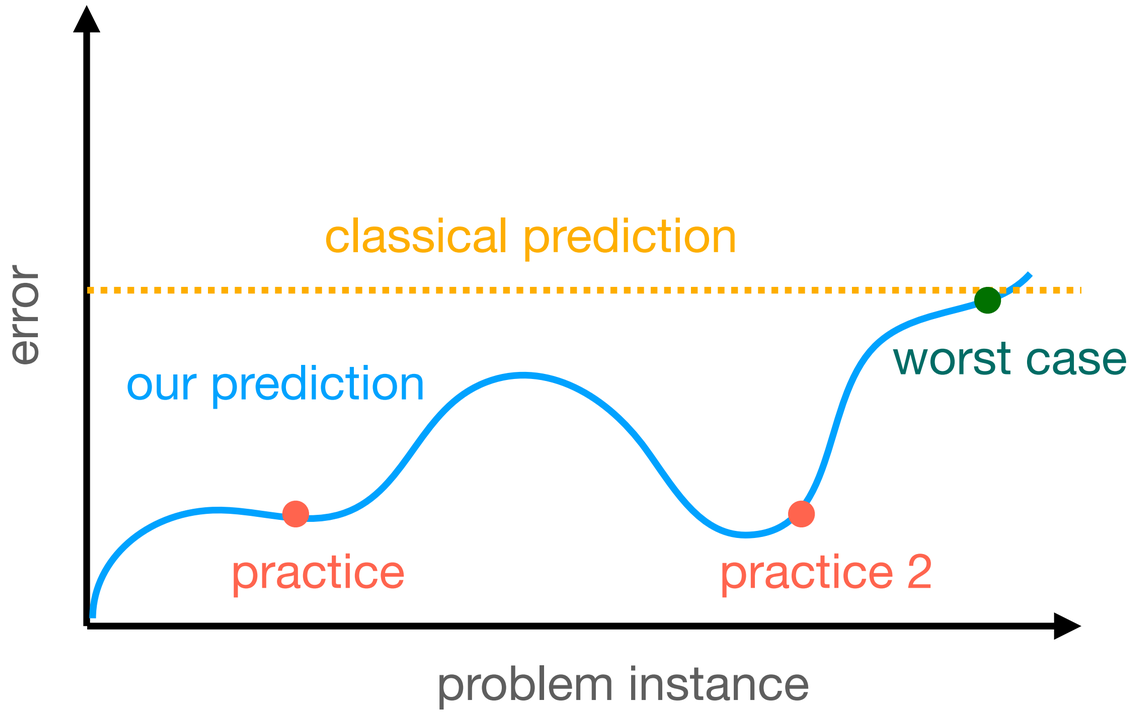
$$\Delta \le \frac{\texttt{const}}{n} \cdot d$$

Our result

$$\frac{\texttt{const1}}{n} \cdot \texttt{DIM} \le \Delta \le \frac{\texttt{const2}}{n} \cdot \texttt{DIM}$$

$$\texttt{DIM} := \#\left\{ \lambda_i \ge \frac{1}{\eta n} \right\} + \eta^2 n^2 \sum_{\lambda_i < \frac{1}{\eta n}} \lambda_i^2$$

eigenvalues of $\mathbf{H} = \mathbb{E}[\mathbf{x}\mathbf{x}^\top]$



- *Zou, Difan, Jingfeng Wu, Vladimir Braverman, Quanquan Gu, and Sham Kakade. "Benign overfitting of constant-stepsize sgd for linear regression." In Conference on Learning Theory, pp. 4633-4635. PMLR, 2021.*
- *Wu, Jingfeng, Difan Zou, Vladimir Braverman, Quanquan Gu, and Sham M. Kakade. "Last Iterate Risk Bounds of SGD with Decaying Stepsize for Overparameterized Linear Regression." International Conference on Machine Learning, 2022.*

SGD can solve well-structured high-dim problems

SGD or WD?

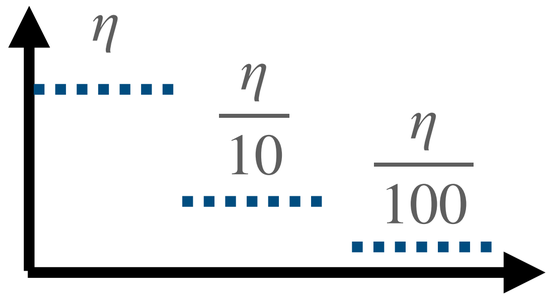
SGD vs. WD

- Ridge regularization (WD)

$$\mathbf{w} = \arg\min \sum_{t=1}^{n} \|\mathbf{x}_n^\top \mathbf{w} - y_n\|_2^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

- One-pass SGD (i.e., early stopping)

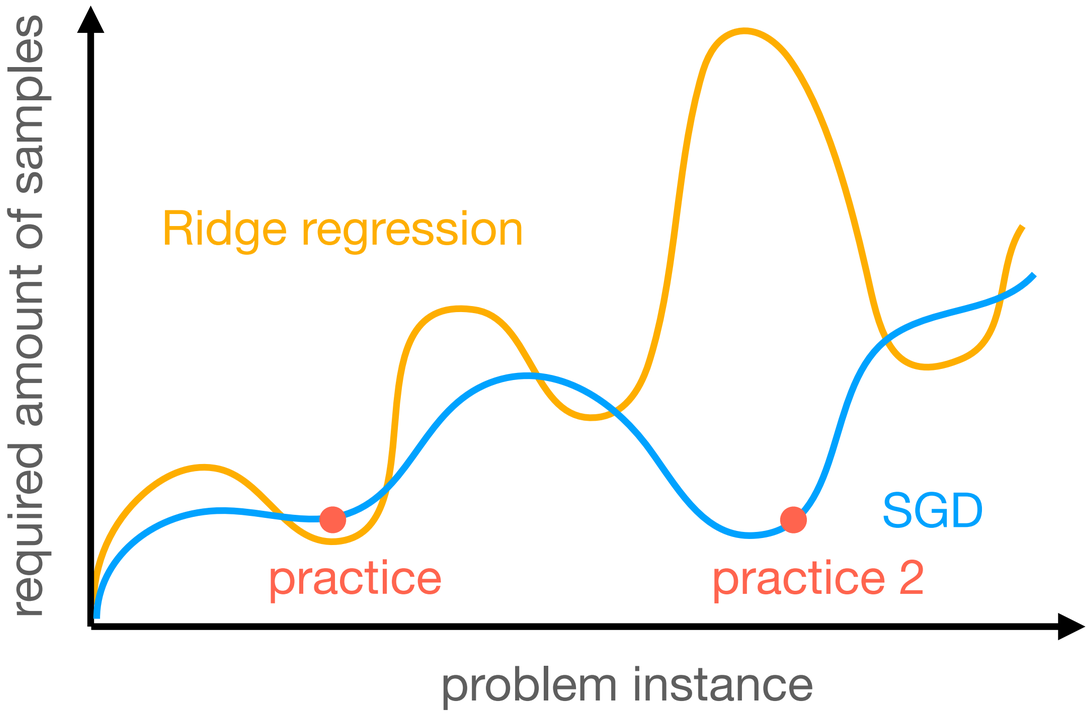
$$\mathbf{w}_t = \mathbf{w}_{t-1} - \eta_t \cdot (\mathbf{x}_t^\top \mathbf{w}_{t-1} - y_t) \cdot \mathbf{x}_t, \quad t = 1, \ldots, N$$



SGD vs. WD

Ridge could be bad ☹️

SGD is alway nearly good 🙂



Tune both SGD and ridge to their best.

Let m and n be their sample complexity

- For each problem in set:

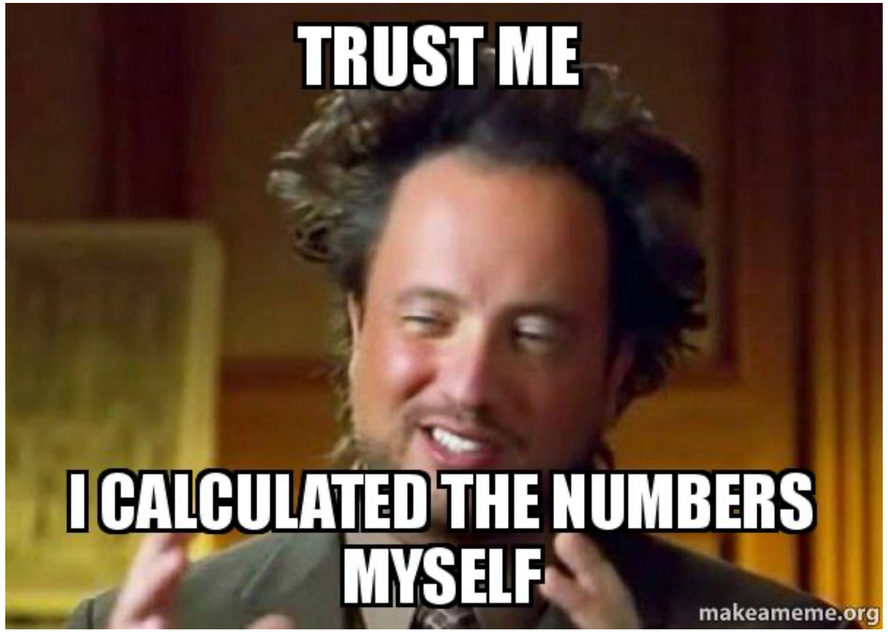
$$m \lesssim n \log^2(n)$$

- There is a problem in the set:

$$n \geq m^2/\log^4(m)$$

- *Tsigler, Alexander, and Peter L. Bartlett. "Benign overfitting in ridge regression." arXiv preprint arXiv:2009.14286 (2020).*
- *Zou, Difan, Jingfeng Wu, Vladimir Braverman, Quanquan Gu, Dean P. Foster, and Sham Kakade. "The benefits of implicit regularization from sgd in least squares problems." Advances in Neural Information Processing Systems 34 (2021): 5456-5468.*

Tuning SGD is probably more worthy than tuning WD

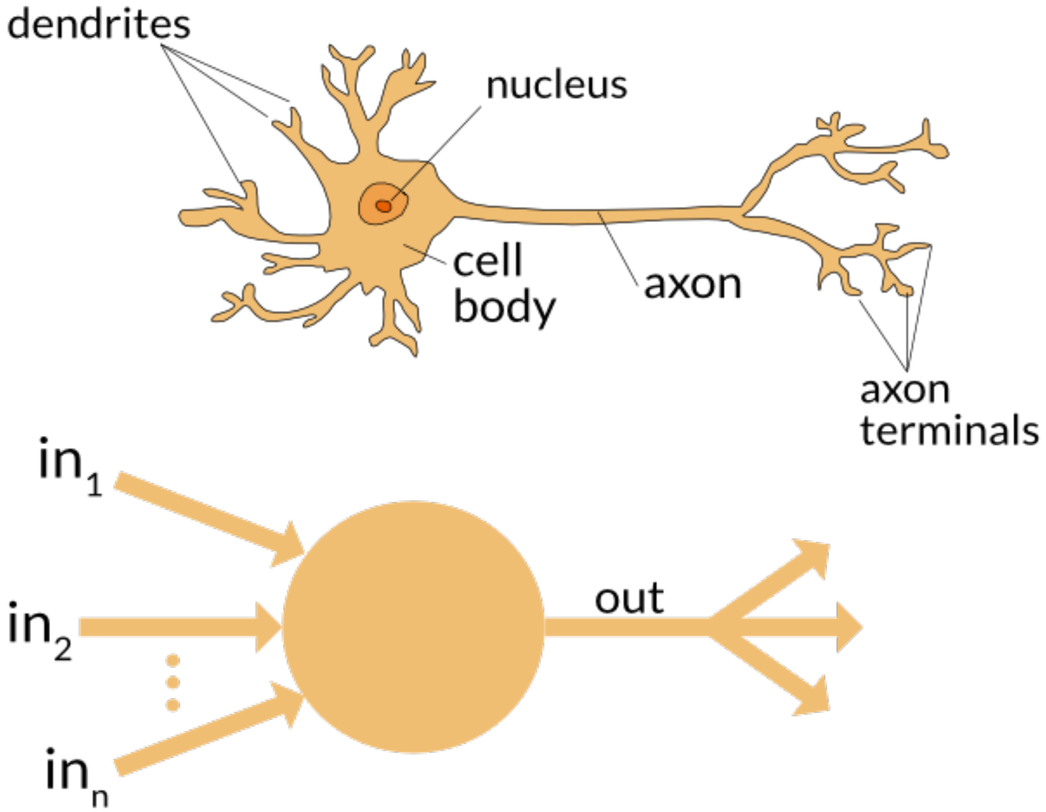


Beyond Linearity

A high-dim ReLU regression

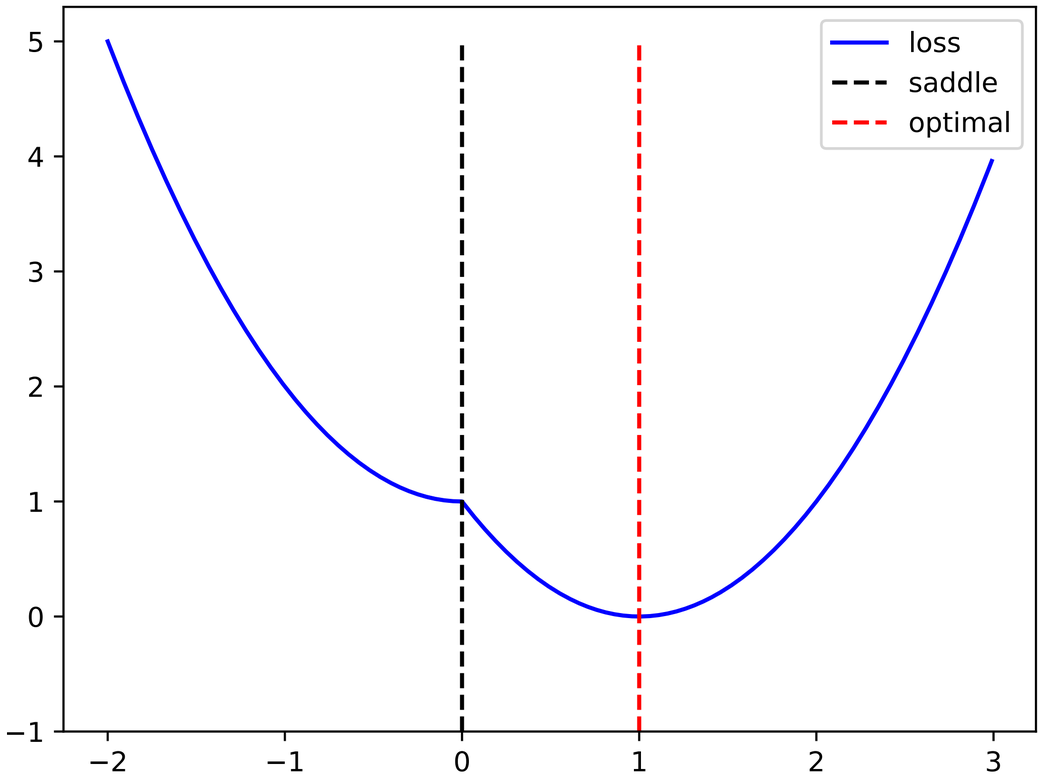
$$y = \texttt{ReLU}(\mathbf{x}^\top \mathbf{w}_*) + \mathcal{N}(0,1)$$

- Non-convex -> very hard
- SGD no longer a good idea
- But, can fix it!



A loss landscape example

- Non-convex
- SGD, initialized from left, will stuck
- Exact-gradient based updates may not be good

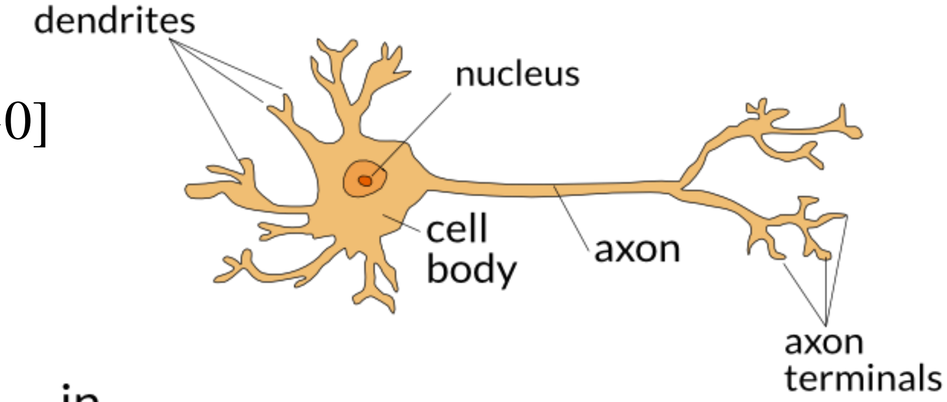


ML101: Perceptron

$$L(\mathbf{w}) = \sum_{(\mathbf{x},y)} \left(\texttt{ReLU}(\mathbf{x}^\top \mathbf{w}_*) - y\right)^2$$

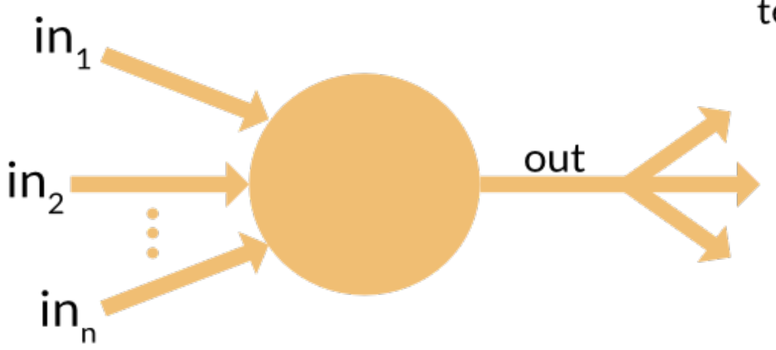
- SGD

$$\mathbf{w}_t = \mathbf{w}_{t-1} - \eta_t \cdot (\texttt{ReLU}(\mathbf{x}_t^\top \mathbf{w}_{t-1}) - y_t) \cdot \mathbf{x}_t \cdot 1_{[\mathbf{x}_t^\top \mathbf{w}_{t-1} > 0]}$$



- Perceptron (iteratively error correction)

$$\mathbf{w}_t = \mathbf{w}_{t-1} - \eta_t \cdot (\texttt{ReLU}(\mathbf{x}_t^\top \mathbf{w}_{t-1}) - y_t) \cdot \mathbf{x}_t$$

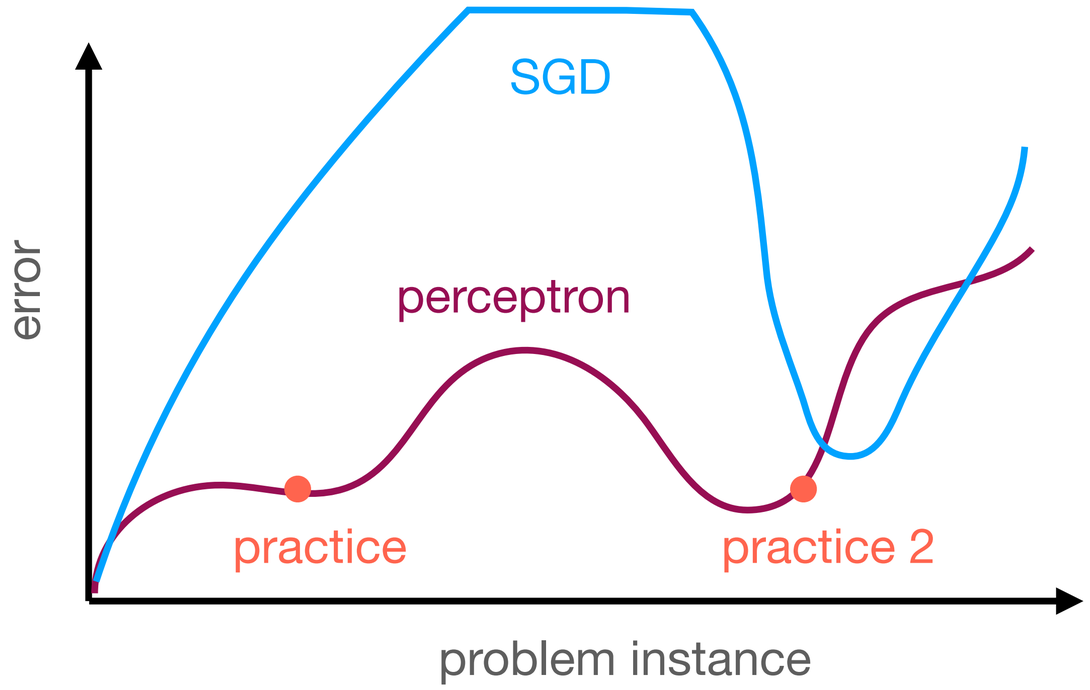


Perceptron solves ReLU regression

$$\frac{\texttt{const1}}{n} \cdot \texttt{DIM} \le \Delta \le \frac{\texttt{const2}}{n} \cdot \texttt{DIM}$$

$$\texttt{DIM} := \#\left\{ \lambda_i \ge \frac{1}{\eta n} \right\} + \eta^2 n^2 \sum_{\lambda_i < \frac{1}{\eta n}} \lambda_i^2$$

eigenvalues of $\mathbf{H} = \mathbb{E}[\mathbf{x}\mathbf{x}^\top]$



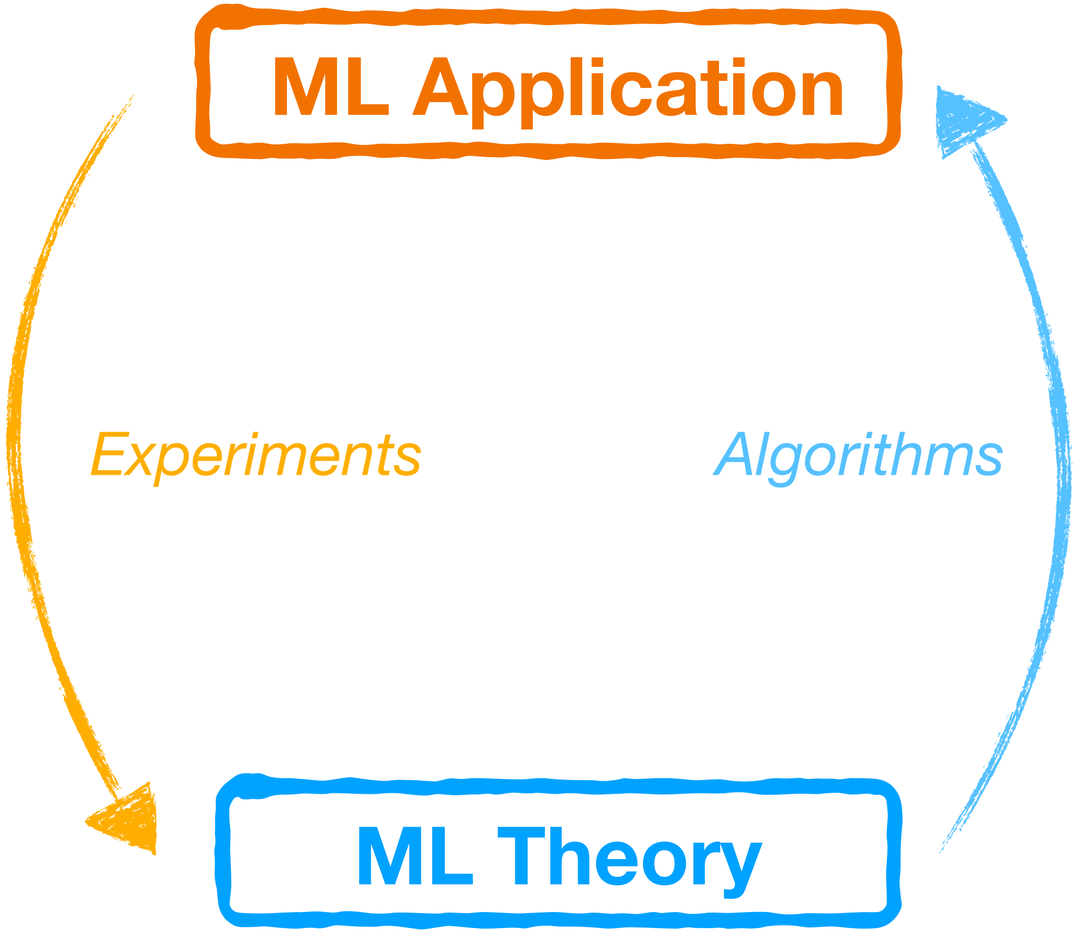
Open Question:

How to extend perceptron to DL?

Better theory

- High-dim
- Non-convex
- Average cases

…



Better practice

- Algo. choice
  - SGD > WD
- Algo. design
  - Explore structure
  - Adjust gradient?

…

Take Home & Acknowledgement

- Algorithmic regularization
- SGD > WD
- ReLU: perceptron > SGD

Vova Bravermen

Dean P. Foster

Quanquan Gu

Sham M. Kakade

Difan Zou