SGD vs GD : Rank Deficiency in Linear Networks

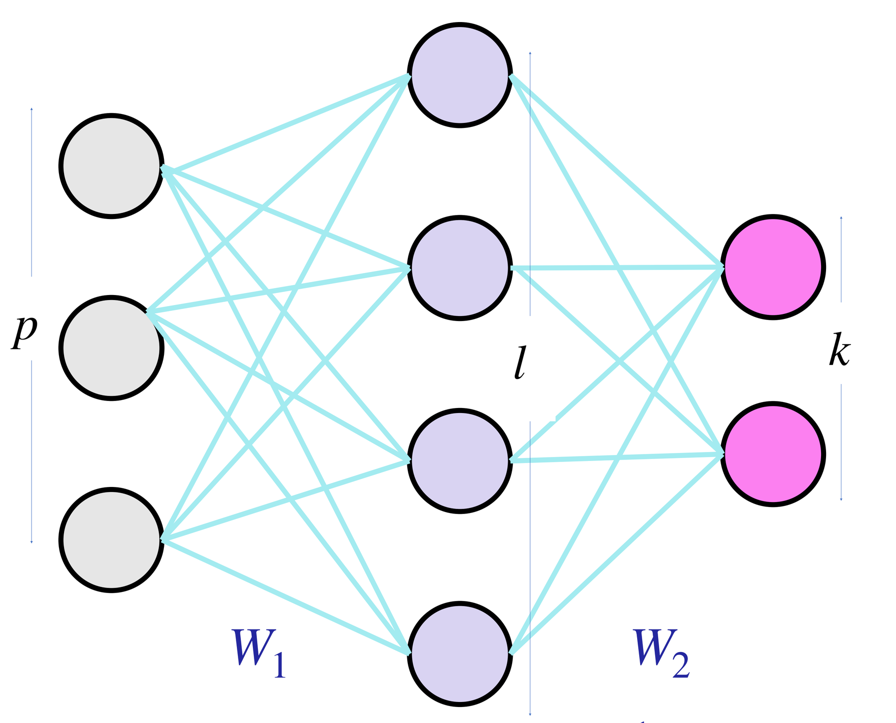
Aditya Varre, Margarita Sagitova, Nicolas Flammarion @ Theory of Machine Learning Lab (TML), Lausanne

EPFL — NEURAL INFORMATION PROCESSING SYSTEMS

Regression with Linear Network

Setup: The $(x_i)_{i=1}^n$ from $\mathbb{R}^p$ and w.l.o.g assume the labels are generated by $y_i = \mathbf{U}_*^\top x_i \in \mathbb{R}^k$ and $\mathbf{U}_* \in \mathbb{R}^{d \times k}$.

A linear network representing function f

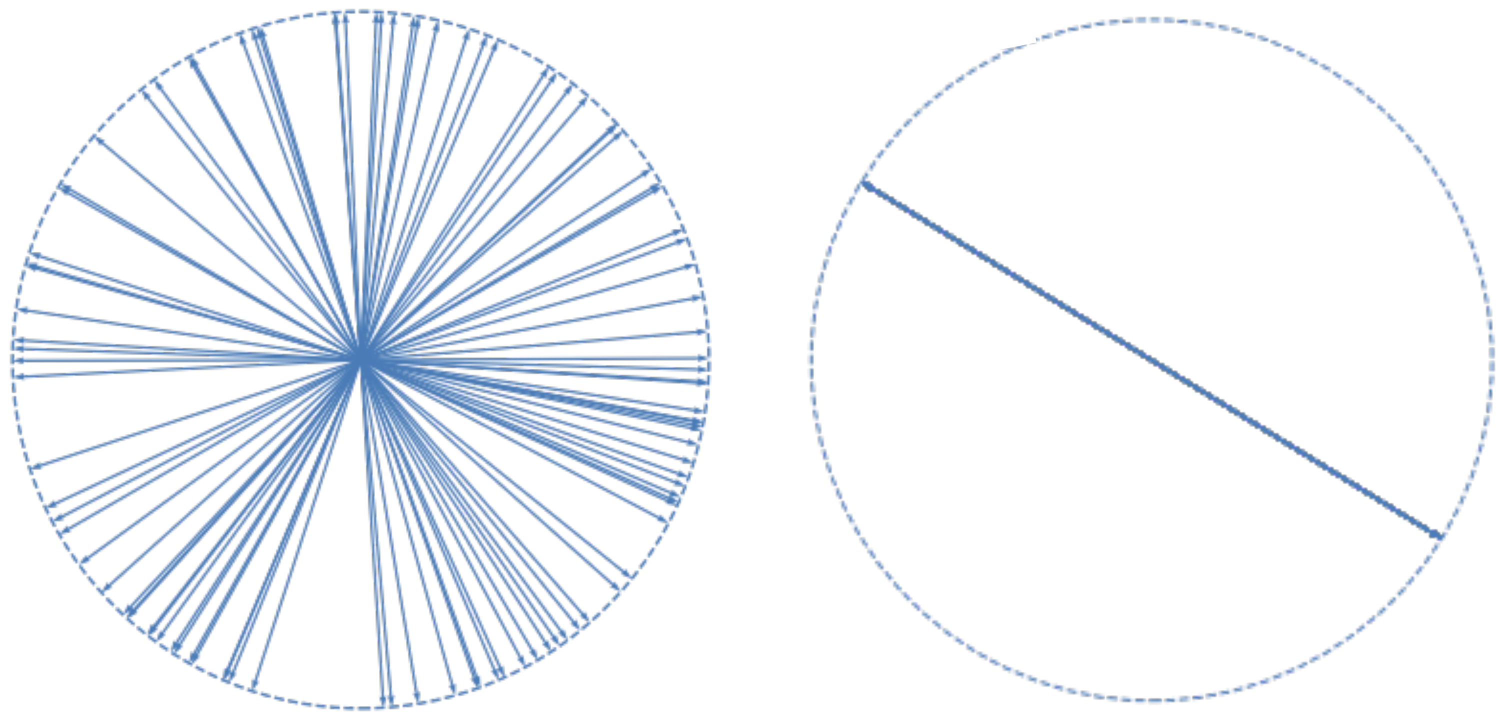


Linear Network: Let hidden layer be $W_1 \in \mathbb{R}^{d \times l}$ and weight layer $W_2 \in \mathbb{R}^{l \times k}$. The network represents the function

$$f(x) = W_2^\top W_1^\top x.$$

Square Loss: $L(W_1, W_2) = \frac{1}{2}\|Y - XW_1W_2\|^2$ where X, Y denotes the data in a matrix form.

A simple illustration: Let us consider the case with $k = 1$, we train GD and SGD with same initialization



(a) Gradient Descent (b) Stochastic Gradient Descent

The directions of columns of W_1 at the end of training linear network.

SGD and simple structures:

a) Limiting dynamics of SGD - Blanc et. al. 2020, Damian et. al. 2020, Li et. al. 2021.

b) SGD on diagonal networks and bias to sparse predictor- Pesme et. al. 2021, Pillaud-Vivien et.al. 2022

c) Empirical works in deep networks - Andruischenko et. al. 2023, Chen et. al. 2023.

How does **stochasticity** enable the emergence of simple structures?

Gradient Flow: Gradient flow on linear networks can be written as

$$\mathrm{d}W_1 = -\nabla_{W_1} L(W_1, W_2)\mathrm{d}t = X^\top (Y - XW_1W_2)W_2^\top \mathrm{d}t,$$
$$\mathrm{d}W_2 = -\nabla_{W_2} L(W_1, W_2)\mathrm{d}t = W_1^\top X^\top (Y - XW_1W_2)\mathrm{d}t\,.$$

Let the equivalent linear predictor $\beta = W_1W_2,\ \nabla L(\beta) = X^\top(X\beta - Y)$.

Block Matrix: Using a block matrix $\theta = [W_1^\top \mid W_2] \in \mathbb{R}^{l \times (p+k)}$.

$$\mathrm{d}\theta = \theta\ J\ \mathrm{d}t \quad \text{where} \quad J := \begin{bmatrix} 0_{p \times p} & -\nabla L(\beta) \\ -\nabla L(\beta)^\top & 0_{k \times k} \end{bmatrix},$$

reveals the inherent **multiplicative nature** of the gradient in linear networks.

Stochastic Gradient Flow: We choose a continuous version of Label noise gradient descent. With $\mathbf{B}_t \in \mathbb{R}^{n \times k}$ and δ being the standard Brownian motion and noise level,

$$\mathrm{d}\theta = \theta\ [J\mathrm{d}t + \sqrt{\eta\delta}\mathrm{d}\xi] \quad \mathrm{d}\xi := \begin{bmatrix} 0_{p \times p} & X^\top \mathrm{d}\mathbf{B}_t \\ \left(X^\top \mathrm{d}\mathbf{B}_t\right)^\top & 0_{k \times k} \end{bmatrix},$$

Dichotomy via Determinant

Theorem : For the SGF, the following property holds for the evolution of determinant

$$\mathrm{d}\left(\det\left(\theta^\top\theta\right)\right) = -2\eta\delta k\ \mathrm{Tr}\{\left(X^\top X\right)\}\ \det\left(\theta^\top\theta\right)\mathrm{d}t\,.$$

Hence, $\det\left(\theta^\top\theta\right) = \det\left(\theta_0^\top\theta_0\right)\exp\left\{-2\delta k \mathrm{Tr}\{\left(X^\top X\right)\}t\right\}$ where θ_0 is the initialization.

Comments :

a) For GF, $\det\left(\theta^\top\theta\right) = \det\left(\theta_0^\top\theta_0\right)$ -> **rank preserving invariant.**

b) For SGF, $\lim_{t \to \infty} \det\left(\theta(t)^\top\theta(t)\right) \to 0$ -> **SGF diminishes the rank** and simplifies the parameters.

Limitations :

a) Diminishing only the **smallest singular value** is sufficient to reduce the determinant. Therefore the above result does not capture the complete simplification of the parameters

b) The dichotomy is effective only when $\det\left(\theta_0^\top\theta_0\right) \neq 0$, which occurs when $l \geqslant p + k$, thereby limiting the result to networks with large widths

Repulsive Force between Singularvalues

From the SDE on the matrix-valued process on θ, we can derive the SDE governing the singular values [Dyson,1962, Bru, 1989].

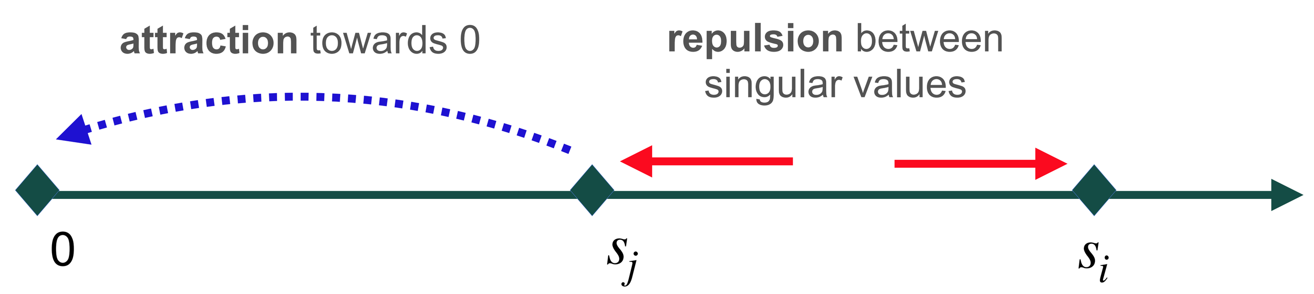
Time rescaled SGF for scalar regression, $W_1 = W, W_2 = a, X = I$,

$$\mathrm{d}W = \ \mathrm{d}\mathbf{X}a^\top; \quad \mathrm{d}a = W^\top \mathrm{d}\mathbf{X}, \text{ where } \mathrm{d}\mathbf{X} = \frac{1}{\eta\delta}(Y - Wa)\mathrm{d}t + \mathrm{d}\mathbf{B}_t$$

SVD: Let $W = U\Sigma V^\top$. Let $s_1, s_2, \ldots s_l$ be the eigenvalues of $W^\top W$ (squared singular values of W)

Theorem: Until the collision time, the evolution of eigenvalues is given by

$$\mathrm{d}\left(s_i\right) = pc_i^2\ \mathrm{d}t + \sum_{j \neq i}^{l} \frac{s_i c_j^2 + s_j c_i^2}{s_i - s_j}\mathrm{d}t + 2\sqrt{s_i c_i^2}\ (\mathrm{d}\tilde{\mathbf{X}})_i$$



Forces at play:

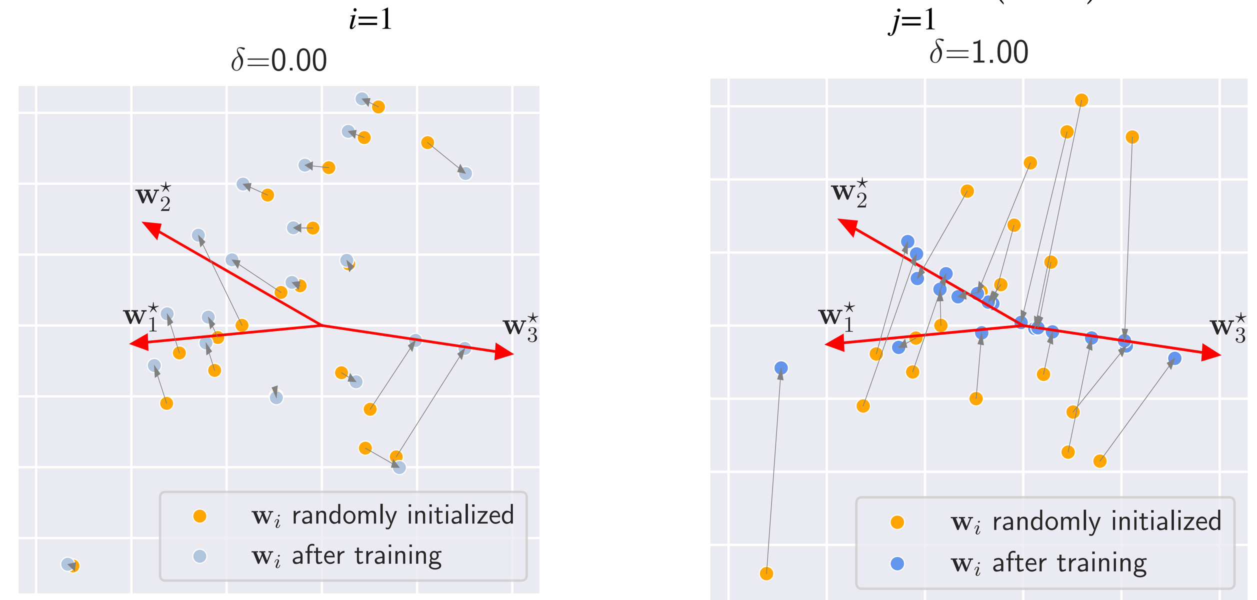
(a) A repulsion force between the eigenvalues due to the skew-symmetric term

(b) A pull towards zero due to the Geometric Brownian motion

$$\mathrm{d}\log\left(s_i\right) = p\frac{c_i^2}{s_i}\ \mathrm{d}t - 2\frac{c_i^2}{s_i}\ \mathrm{d}t + \frac{1}{s_i}\sum_{j \neq i}^{l} \frac{s_i c_j^2 + s_j c_i^2}{s_i - s_j}\mathrm{d}t + 2\sqrt{\frac{c_i^2}{s_i}}\ (\mathrm{d}\tilde{\mathbf{X}})_i$$

Empirics with ReLU: Let $k = 1$, we train one hidden layer ReLU network with GD and SGD from the same initialization.

Teacher : $y = \sum_{i=1}^{3} \sigma(x^\top w_i^*)$ **Student:** $y = \sum_{j=1}^{m} \sigma\left(w_j^\top x\right)$



Neurons in ReLU networks aligns along the teacher directions when trained with a noise level $\delta = 1$.

References:

a) Blanc et. al. Implicit regularization for deep neural networks driven by an ornstein-uhlenbeck like process, COLT 2020.

b) Li et. al. What happens after sgd reaches zero loss?–a mathematical framework., ICLR 2021

c) M.-F. Bru. Diffusions of perturbed principal component analys 1989

d) Chen et. al. Stochastic collapse: How gradient noise attracts SGD dynamics towards simpler subnetworks, NeurIPS 2023