Deep Neural Nets and Products of Random Matrices

Mihai Nica, University of Toronto

Based on joint work with Boris Hanin, Texas A&M

March 27, 2019

Part 1: Deep neural nets

- Mathematical definitions
- Limit theorem for a random neural net

Part 1: Deep neural nets

- Mathematical definitions
- Limit theorem for a random neural net

Part 2: Products of Random Matrices

- Connection to Neural Nets
- Limit theorem for products of random matrices

Part 1: Deep neural nets

- Mathematical definitions
- Limit theorem for a random neural net

Part 2: Products of Random Matrices

- Connection to Neural Nets
- Limit theorem for products of random matrices

Part 3: Proof Ideas

- Moments: Path counting
- Kolmogorov-Smirnov Distance: Martingales

Part 1: Neural Nets

- Mathematical Definitions
- Limit theorem for a random neural net

Definition - Fully Connected Neural Net

Definition - Fully Connected Neural Net

Fix a **depth** $d \in \mathbb{N}$, and **layer widths** $n_0, n_1 \ldots n_d \in \mathbb{N}$.

Definition - Fully Connected Neural Net

Fix a **depth** $d \in \mathbb{N}$, and **layer widths** $n_0, n_1 \ldots n_d \in \mathbb{N}$. A **neural net** is a function:

$$f^{n_0 \to n_d} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_d}$$

Definition - Fully Connected Neural Net

Fix a **depth** $d \in \mathbb{N}$, and **layer widths** $n_0, n_1 \ldots n_d \in \mathbb{N}$. A **neural net** is a function:

$$f^{n_0 \to n_d} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_d}$$

It depends d **weight matrices** and d **bias vectors**:

$$\underline{W}^{(i)} \text{ an } n_i \times n_{i-1} \text{ matrix}$$

$$\vec{b}^{(i)} \text{ a vector in } \mathbb{R}^{n_i}$$

Definition - Fully Connected Neural Net

Fix a **depth** $d \in \mathbb{N}$, and **layer widths** $n_0, n_1 \dots n_d \in \mathbb{N}$. A **neural net** is a function:

$$f^{n_0 \to n_d} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_d}$$

It depends d **weight matrices** and d **bias vectors**:

$$\underline{W}^{(i)} \text{ an } n_i \times n_{i-1} \text{ matrix}$$
$$\vec{b}^{(i)} \text{ a vector in } \mathbb{R}^{n_i}$$

and a (non-linear) **activation function** $\phi : \mathbb{R} \to \mathbb{R}$.

Definition - Fully Connected Neural Net

Fix a **depth** $d \in \mathbb{N}$, and **layer widths** $n_0, n_1 \ldots n_d \in \mathbb{N}$. A **neural net** is a function:

$$f^{n_0 \to n_d} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_d}$$

It depends d **weight matrices** and d **bias vectors**:

$$\underline{W}^{(i)} \text{ an } n_i \times n_{i-1} \text{ matrix}$$

$$\vec{b}^{(i)} \text{ a vector in } \mathbb{R}^{n_i}$$

and a (non-linear) **activation function** $\phi : \mathbb{R} \to \mathbb{R}$. Each **layer** of the network gives a function $f^{n_{i-1} \to n_i} : \mathbb{R}^{n_{i-1}} \to \mathbb{R}^{n_i}$:

$$f^{n_{i-1} \to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right) \text{ (applied entry-wise)}$$

Definition - Fully Connected Neural Net

Fix a **depth** $d \in \mathbb{N}$, and **layer widths** $n_0, n_1 \ldots n_d \in \mathbb{N}$. A **neural net** is a function:

$$f^{n_0 \to n_d} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_d}$$

It depends d **weight matrices** and d **bias vectors**:

$$\underline{W}^{(i)} \text{ an } n_i \times n_{i-1} \text{ matrix}$$

$$\vec{b}^{(i)} \text{ a vector in } \mathbb{R}^{n_i}$$

and a (non-linear) **activation function** $\phi : \mathbb{R} \to \mathbb{R}$. Each **layer** of the network gives a function $f^{n_{i-1} \to n_i} : \mathbb{R}^{n_{i-1}} \to \mathbb{R}^{n_i}$:

$$f^{n_{i-1} \to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right) \text{ (applied entry-wise)}$$

Finally, $f^{n_0 \to n_d}$ is the composition of these:

$$f^{n_0 \to n_d} := f^{n_{d-1} \to n_d} \circ f^{n_{d-2} \to n_{d-1}} \circ \ldots \circ f^{n_0 \to n_1}$$

How to find the parameters

Supervised Learning: Problem

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n \to m}$ minimizes the **prediction error**:

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n \to m}$ minimizes the **prediction error**:

$$\text{Error}(W, b) = \sum_{e \in \text{Examples}} \|f^{n \to m}(\vec{x}_e) - \vec{y}_e\|^2$$

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n \to m}$ minimizes the **prediction error**:

$$\text{Error}(W, b) = \sum_{e \in \text{Examples}} \|f^{n \to m}(\vec{x}_e) - \vec{y}_e\|^2$$

Supervised Learning: Solution Idea

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n \to m}$ minimizes the **prediction error**:

$$\text{Error}(W, b) = \sum_{e \in \text{Examples}} \|f^{n \to m}(\vec{x}_e) - \vec{y}_e\|^2$$

Supervised Learning: Solution Idea

0. **Invent** the architecture: depth d and layer widths $n_1, \ldots, n_{d-1}$. Set $n_0 = n$, $n_d = m$

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n \to m}$ minimizes the **prediction error**:

$$\text{Error}(W, b) = \sum_{e \in \text{Examples}} \|f^{n \to m}(\vec{x}_e) - \vec{y}_e\|^2$$

Supervised Learning: Solution Idea

0. **Invent** the architecture: depth d and layer widths $n_1, \ldots, n_{d-1}$. Set $n_0 = n$, $n_d = m$
1. **Initialization:** Pick parameters $\underline{W}, \vec{b}$ at **random**.

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e\in\text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e\in\text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n\to m}$ minimizes the **prediction error**:

$$\text{Error}(W, b) = \sum_{e\in\text{Examples}} \|f^{n\to m}(\vec{x}_e) - \vec{y}_e\|^2$$

Supervised Learning: Solution Idea

0. **Invent** the architecture: depth d and layer widths $n_1, \ldots, n_{d-1}$. Set $n_0 = n$, $n_d = m$
1. **Initialization:** Pick parameters $\underline{W}, \vec{b}$ at **random**.
2. **Modify** parameters to **shrink** Error(W, b) by **gradient descent**:

$$\text{new } W^{(i)}_{j,k} := \text{ old } W^{(i)}_{j,k} - \partial_{W^{(i)}_{j,k}} \text{Error}(W, b)$$

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n \to m}$ minimizes the **prediction error**:

$$\text{Error}(W, b) = \sum_{e \in \text{Examples}} \left\| f^{n \to m}(\vec{x}_e) - \vec{y}_e \right\|^2$$

Supervised Learning: Solution Idea

0. **Invent** the architecture: depth d and layer widths $n_1, \ldots, n_{d-1}$.
   Set $n_0 = n$, $n_d = m$
1. **Initialization:** Pick parameters $\underline{W}, \vec{b}$ at **random**.
2. **Modify** parameters to **shrink** Error(W, b) by **gradient descent**:

$$\text{new } W_{j,k}^{(i)} := \text{ old } W_{j,k}^{(i)} - \partial_{W_{j,k}^{(i)}} \text{Error}(W, b)$$

3. Repeat step 2 **many times**.

How to find the parameters

Supervised Learning: Problem

Given examples $\{\vec{x}_e\}_{e \in \text{Examples}} \subset \mathbb{R}^n$, and labels $\{y_e\}_{e \in \text{Examples}} \subset \mathbb{R}^m$how to find parameters $\underline{W}^{(i)}$ and $\vec{b}^{(i)}$ so that $f^{n \to m}$ minimizes the **prediction error**:

$$\text{Error}(W, b) = \sum_{e \in \text{Examples}} \|f^{n \to m}(\vec{x}_e) - \vec{y}_e\|^2$$

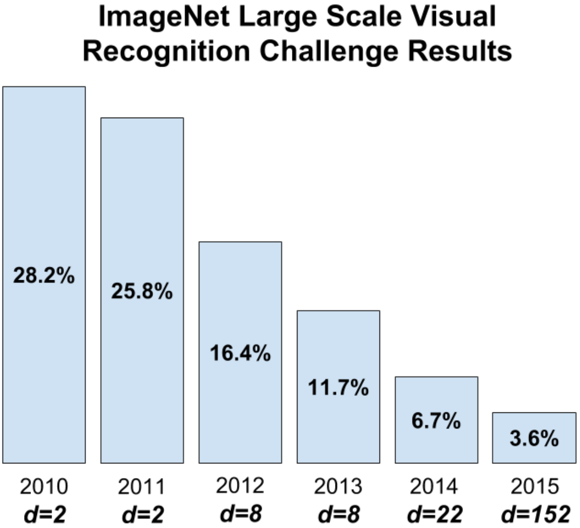
Supervised Learning: Solution Idea

0. **Invent** the architecture: depth d and layer widths $n_1, \ldots, n_{d-1}$. Set $n_0 = n$, $n_d = m$
1. **Initialization:** Pick parameters $\underline{W}, \vec{b}$ at **random**.
2. **Modify** parameters to **shrink** $\text{Error}(W, b)$ by **gradient descent**:

$$\text{new } W_{j,k}^{(i)} := \text{ old } W_{j,k}^{(i)} - \partial_{W_{j,k}^{(i)}} \text{Error}(W, b)$$

3. Repeat step 2 **many times**. **Hope** that the error is now small.

Which architecture is best?



Empirical Issue: d large has **vanishing and exploding gradients**:

Empirical Issue: d large has **vanishing and exploding gradients**: On **random initialization**, $\partial_{W_{j,k}^{(i)}} Error(W, b)$ is very large or small.

Empirical Issue: d large has **vanishing and exploding gradients**: On **random initialization**, $\partial_{W^{(i)}_{j,k}} Error(W, b)$ is very large or small.

Definition

The **aspect ratio** of a network is defined by:

$$\beta = \sum_{i=1}^{d} \frac{1}{n_i}$$

Empirical Issue: d large has **vanishing and exploding gradients**: On **random initialization**, $\partial_{W^{(i)}_{j,k}} Error(W, b)$ is very large or small.

Definition

The **aspect ratio** of a network is defined by:

$$\beta = \sum_{i=1}^{d} \frac{1}{n_i}$$

Our mathematical result:
If β is **large**, $\partial_{W^{(i)}_{j,k}} Error(W, b)$ will be very large or very small with **high probability**.

Definition. The **Input-Output Jacobian** matrix $\underline{J} = \mathrm{Jac}(f^{n_0 \to n_d})$ is the $n_d \times n_0$ matrix

$$J_{ij}(\vec{x}) := \partial_j f_i^{n_0 \to n_d}(\vec{x})$$

Definition. The **Input-Output Jacobian** matrix $\underline{J} = \mathrm{Jac}(f^{n_0 \to n_d})$ is the $n_d \times n_0$ matrix

$$J_{ij}(\vec{x}) := \partial_j f_i^{n_0 \to n_d}(\vec{x})$$

Remark: $\partial_{W_{j,k}^{(i)}} Error(W, b)$ can be written in terms of $\underline{J}$.

Definition. The **Input-Output Jacobian** matrix $\underline{J} = \mathrm{Jac}(f^{n_0 \to n_d})$ is the $n_d \times n_0$ matrix

$$J_{ij}(\vec{x}) := \partial_j f_i^{n_0 \to n_d}(\vec{x})$$

Remark: $\partial_{W_{j,k}^{(i)}} Error(W, b)$ can be written in terms of $\underline{J}$.

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$

Definition. The **Input-Output Jacobian** matrix $\underline{J} = \mathrm{Jac}(f^{n_0 \to n_d})$ is the $n_d \times n_0$ matrix

$$J_{ij}(\vec{x}) := \partial_j f_i^{n_0 \to n_d}(\vec{x})$$

Remark: $\partial_{W_{j,k}^{(i)}} Error(W, b)$ can be written in terms of $\underline{J}$.

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W}^{(i)}, \vec{b}^{(i)}$ are **chosen randomly**

Definition. The **Input-Output Jacobian** matrix $\underline{J} = \mathrm{Jac}(f^{n_0 \to n_d})$ is the $n_d \times n_0$ matrix

$$J_{ij}(\vec{x}) := \partial_j f_i^{n_0 \to n_d}(\vec{x})$$

Remark: $\partial_{W_{j,k}^{(i)}} Error(W, b)$ can be written in terms of $\underline{J}$.

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W}^{(i)}, \vec{b}^{(i)}$ are **chosen randomly** then for almost every $\vec{x} \in \mathbb{R}^{n_0}$, the vector:

$$\underline{J}(\vec{x})\vec{1}$$

where $\vec{1} = \frac{1}{\sqrt{n_0}}(1 \ldots, 1) \in \mathbb{R}^{n_0}$, has norm whose distribution depends on $\beta = \sum_{i=1}^{d} \frac{1}{n_i}$:

Definition. The **Input-Output Jacobian** matrix $\underline{J} = \mathrm{Jac}(f^{n_0 \to n_d})$ is the $n_d \times n_0$ matrix

$$J_{ij}(\vec{x}) := \partial_j f_i^{n_0 \to n_d}(\vec{x})$$

Remark: $\partial_{W_{j,k}^{(i)}} Error(W, b)$ can be written in terms of $\underline{J}$.

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W^{(i)}}, \vec{b}^{(i)}$are **chosen randomly** then for almost every $\vec{x} \in \mathbb{R}^{n_0}$, the vector:

$$\underline{J}(\vec{x})\vec{1}$$

where $\vec{1} = \frac{1}{\sqrt{n_0}}(1 \ldots, 1) \in \mathbb{R}^{n_0}$, has norm whose distribution depends on $\beta = \sum_{i=1}^{d} \frac{1}{n_i}$:

$$\left\| \underline{J}(\vec{x})\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W}^{(i)}, \vec{b}^{(i)}$ are **chosen randomly** then for almost every $\vec{x} \in \mathbb{R}^{n_0}$:

$$\left\| \underline{J}(\vec{x})\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\mathcal{N}(0,1)$ a Gaussian, $\vec{1} = \frac{1}{\sqrt{n_0}}(1\ldots,1) \in \mathbb{R}^{n_0}$, $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$.

Conditions on $\underline{W}^{(i)}, \vec{b}^{(i)}$:

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W}^{(i)}, \vec{b}^{(i)}$ are **chosen randomly** then for almost every $\vec{x} \in \mathbb{R}^{n_0}$:

$$\left\| \underline{J}(\vec{x})\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\mathcal{N}(0,1)$ a Gaussian, $\vec{1} = \frac{1}{\sqrt{n_0}}(1 \ldots, 1) \in \mathbb{R}^{n_0}$, $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$.

Conditions on $\underline{W}^{(i)}, \vec{b}^{(i)}$:

- All entries are independent

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W}^{(i)}, \vec{b}^{(i)}$ are **chosen randomly** then for almost every $\vec{x} \in \mathbb{R}^{n_0}$:

$$\left\| \underline{J}(\vec{x})\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\mathcal{N}(0,1)$ a Gaussian, $\vec{1} = \frac{1}{\sqrt{n_0}}(1 \ldots, 1) \in \mathbb{R}^{n_0}$, $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$.

Conditions on $\underline{W}^{(i)}, \vec{b}^{(i)}$:

- All entries are independent
- All entries are symmetrically distributed (i.e. $X \overset{d}{=} -X$)

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W}^{(i)}, \vec{b}^{(i)}$are <u>**chosen randomly**</u> then for almost every $\vec{x} \in \mathbb{R}^{n_0}$:

$$\left\| \underline{J}(\vec{x})\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\mathcal{N}(0,1)$ a Gaussian,$\vec{1} = \frac{1}{\sqrt{n_0}}(1 \ldots, 1) \in \mathbb{R}^{n_0}$, $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$.

Conditions on $\underline{W}^{(i)}, \vec{b}^{(i)}$:

- All entries are independent
- All entries are symmetrically distributed (i.e. $X \overset{d}{=} -X$)
- All entries of $\underline{W}^{(i)}$ are mean 0 and variance $2n_i^{-1}$

Limit Theorem for Random Neural Nets - Hanin, N. 2018

If $\phi(x) = \max\{x, 0\}$ and $\underline{W}^{(i)}, \vec{b}^{(i)}$ are **chosen randomly** then for almost every $\vec{x} \in \mathbb{R}^{n_0}$:

$$\left\| \underline{J}(\vec{x})\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\mathcal{N}(0,1)$ a Gaussian, $\vec{1} = \frac{1}{\sqrt{n_0}}(1 \ldots, 1) \in \mathbb{R}^{n_0}$, $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$.

Conditions on $\underline{W}^{(i)}, \vec{b}^{(i)}$:

- All entries are independent
- All entries are symmetrically distributed (i.e. $X \stackrel{d}{=} -X$)
- All entries of $\underline{W}^{(i)}$ are mean 0 and variance $2n_i^{-1}$
- All entries have finite moments of all order and no atoms

Vanishing and Exploding Gradients

Theorem

$$\left\| J\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$, and $\vec{1} = \frac{1}{\sqrt{n_0}}(1\ldots,1) \in \mathbb{R}^{n_0}$

Question: Which architectures have the vanishing/exploding gradient problem?

Vanishing and Exploding Gradients

Theorem

$$\left\| J\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$, and $\vec{1} = \frac{1}{\sqrt{n_0}}(1\ldots,1) \in \mathbb{R}^{n_0}$

Question: Which architectures have the vanishing/exploding gradient problem?
Answer: Those for which aspect ratio β is large! (i.e. the deep+skinny networks)

Vanishing and Exploding Gradients

Theorem

$$\left\| J\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta} \cdot \mathcal{N}(0,1) - \frac{5}{2}\beta \right)$$

where $\beta := \sum_{i=1}^{d} \frac{1}{n_i}$, and $\vec{1} = \frac{1}{\sqrt{n_0}}(1\ldots,1) \in \mathbb{R}^{n_0}$

Question: Which architectures have the vanishing/exploding gradient problem?
Answer: Those for which aspect ratio β is large! (i.e. the deep+skinny networks)

Conjecture

Other, fancier types neural net, (e.g. Convolutional Nets, ResNets) are also log-normal with a different formula for β.

Theorem (Precise Version)

$\left\| \underline{J}\vec{1} \right\|^2 \approx \exp\left(\sqrt{5\beta}\mathcal{N}(0,1) - \frac{5}{2}\beta\right)$, where "$\approx$" means:

Theorem (Precise Version)

$\left\|\underline{J}\vec{1}\right\|^2 \approx \exp\left(\sqrt{5\beta}\mathcal{N}(0,1) - \frac{5}{2}\beta\right)$, where "$\approx$" means:

Moments: For any $k \geq 0$, have:

$$\mathbf{E}\left[\left\|\underline{J}\vec{1}\right\|^{2k}\right] = \exp\left(5\binom{k}{2}\beta + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$

Theorem (Precise Version)

$\left\|\underline{J}\vec{1}\right\|^2 \approx \exp\left(\sqrt{5\beta}\mathcal{N}(0,1) - \frac{5}{2}\beta\right)$, where "$\approx$" means:

Moments: For any $k \geq 0$, have:

$$\mathbf{E}\left[\left\|\underline{J}\vec{1}\right\|^{2k}\right] = \exp\left(5\binom{k}{2}\beta + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$

Kolmogorov-Smirnov distance: $\exists C$ s.t. the cumulative distribution functions, Φ, for the random variables are close in L^∞ norm:

$$\left\|\Phi_{\ln\left(\left\|\underline{J}\vec{1}\right\|^2\right)} - \Phi_{\sqrt{5\beta}\cdot\mathcal{N}(0,1)-\frac{5}{2}\beta}\right\|_\infty \leq C\left(\frac{\sum n_i^{-2}}{\sum n_i^{-1}}\right)^{1/5}$$

Part 2: Products of Random Matrices

- Connection to Neural Nets
- Limit theorem for products of random matrices

J when $\phi(x) = \max\{x, 0\}$

Recall $f^{n_{i-1}\to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right)$ and want to compute

$$\underline{J} = \mathrm{Jac}\left(f^{n_{d-1}\to n_d} \circ f^{n_{d-2}\to n_{d-1}} \circ \ldots \circ f^{n_1\to n_0}\right)$$

J when $\phi(x) = \max\{x, 0\}$

Recall $f^{n_{i-1}\to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right)$ and want to compute

$$\underline{J} = \text{Jac}\left(f^{n_{d-1}\to n_d} \circ f^{n_{d-2}\to n_{d-1}} \circ \ldots \circ f^{n_1\to n_0}\right)$$

Since $\phi(x) = \max\{x, 0\}$, $\phi'(x) = 1\{x > 0\}$, then the gradient of each layer is:

J when $\phi(x) = \max\{x, 0\}$

Recall $f^{n_{i-1}\to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right)$ and want to compute

$$\underline{J} = \mathrm{Jac}\left(f^{n_{d-1}\to n_d} \circ f^{n_{d-2}\to n_{d-1}} \circ \ldots \circ f^{n_1\to n_0}\right)$$

Since $\phi(x) = \max\{x, 0\}$, $\phi'(x) = 1\{x > 0\}$, then the gradient of each layer is:

$$\mathrm{Jac}\left(f^{n_{i-1}\to n_i}\right) = \mathrm{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right)\underline{W}^{(i)}$$

J when $\phi(x) = \max\{x, 0\}$

Recall $f^{n_{i-1} \to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right)$ and want to compute

$$\underline{J} = \text{Jac}\left(f^{n_{d-1} \to n_d} \circ f^{n_{d-2} \to n_{d-1}} \circ \ldots \circ f^{n_1 \to n_0}\right)$$

Since $\phi(x) = \max\{x, 0\}$, $\phi'(x) = 1\{x > 0\}$, then the gradient of each layer is:

$$\text{Jac}\left(f^{n_{i-1} \to n_i}\right) = \text{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right)\underline{W}^{(i)}$$

Since all **random variables** are **symmetric**:

J when $\phi(x) = \max\{x, 0\}$

Recall $f^{n_{i-1}\to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right)$ and want to compute

$$\underline{J} = \text{Jac}\left(f^{n_{d-1}\to n_d} \circ f^{n_{d-2}\to n_{d-1}} \circ \ldots \circ f^{n_1\to n_0}\right)$$

Since $\phi(x) = \max\{x, 0\}$, $\phi'(x) = 1\{x > 0\}$, then the gradient of each layer is:

$$\text{Jac}\left(f^{n_{i-1}\to n_i}\right) = \text{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right)\underline{W}^{(i)}$$

Since all **random variables** are **symmetric**:

$$\text{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right) \stackrel{d}{=} \text{Diag}\left(\vec{X}^{(i)}\right)$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim \textit{Bernoulli}(\frac{1}{2})$.

J when $\phi(x) = \max\{x, 0\}$

Recall $f^{n_{i-1}\to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right)$ and want to compute

$$\underline{J} = \text{Jac}\left(f^{n_{d-1}\to n_d} \circ f^{n_{d-2}\to n_{d-1}} \circ \ldots \circ f^{n_1\to n_0}\right)$$

Since $\phi(x) = \max\{x, 0\}$, $\phi'(x) = 1\{x > 0\}$, then the gradient of each layer is:

$$\text{Jac}\left(f^{n_{i-1}\to n_i}\right) = \text{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right)\underline{W}^{(i)}$$

Since all **random variables** are **symmetric**:

$$\text{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right) \overset{d}{=} \text{Diag}\left(\vec{X}^{(i)}\right)$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim \textit{Bernoulli}(\frac{1}{2})$.
By **chain rule**, we should expect

$$\underline{J} \overset{?}{=} \text{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \ldots \cdot \text{Diag}(\vec{X}^{(1)})W^{(1)}$$

J when $\phi(x) = \max\{x, 0\}$

Recall $f^{n_{i-1}\to n_i}(\vec{x}) := \phi\left(\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)}\right)$ and want to compute

$$\underline{J} = \text{Jac}\left(f^{n_{d-1}\to n_d} \circ f^{n_{d-2}\to n_{d-1}} \circ \ldots \circ f^{n_1\to n_0}\right)$$

Since $\phi(x) = \max\{x, 0\}$, $\phi'(x) = 1\{x > 0\}$, then the gradient of each layer is:

$$\text{Jac}\left(f^{n_{i-1}\to n_i}\right) = \text{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right)\underline{W}^{(i)}$$

Since all **random variables** are **symmetric**:

$$\text{Diag}\left(1\left\{\underline{W}^{(i)}\vec{x} + \vec{b}^{(i)} > 0\right\}\right) \stackrel{d}{=} \text{Diag}\left(\vec{X}^{(i)}\right)$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim Bernoulli(\frac{1}{2})$.
By **chain rule**, we should expect

$$\underline{J} \stackrel{?}{=} \underbrace{\text{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \ldots \cdot \text{Diag}(\vec{X}^{(1)})W^{(1)}}_{:=\underline{M}}$$

Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \mathrm{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \dots \cdot \mathrm{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim \mathit{Bernoulli}(\frac{1}{2})$.

Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \mathrm{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \ldots \cdot \mathrm{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim \mathit{Bernoulli}(\frac{1}{2})$.

Proposition

$$\left\|\underline{M}\vec{1}\right\|^2 \stackrel{d}{=} \left\|\underline{J}\vec{1}\right\|^2$$

Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \text{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \dots \cdot \text{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim Bernoulli(\frac{1}{2})$.

Proposition

$$\left\|\underline{M}\vec{1}\right\|^2 \stackrel{d}{=} \left\|\underline{J}\vec{1}\right\|^2$$

Conditions: $\underline{W}$ and $\vec{b}$ are chosen so that

Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \text{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \cdots \cdot \text{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim Bernoulli(\frac{1}{2})$.

Proposition

$$\left\|\underline{M}\vec{1}\right\|^2 \stackrel{d}{=} \left\|\underline{J}\vec{1}\right\|^2$$

Conditions: $\underline{W}$ and $\vec{b}$ are chosen so that

- All entries are independent

Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \text{Diag}(\vec{X}^{(d)}) W^{(d)} \cdot \cdots \cdot \text{Diag}(\vec{X}^{(1)}) W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim Bernoulli(\frac{1}{2})$.

Proposition

$$\left\| \underline{M}\vec{1} \right\|^2 \stackrel{d}{=} \left\| \underline{J}\vec{1} \right\|^2$$

Conditions: $\underline{W}$ and $\vec{b}$ are chosen so that

- All entries are independent
- All entries are symmetrically distributed (i.e. $X \stackrel{d}{=} -X$)

Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \mathrm{Diag}(\vec{X}^{(d)}) W^{(d)} \cdot \dots \cdot \mathrm{Diag}(\vec{X}^{(1)}) W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim \mathit{Bernoulli}(\frac{1}{2})$.

Proposition

$$\left\| \underline{M}\vec{1} \right\|^2 \stackrel{d}{=} \left\| \underline{J}\vec{1} \right\|^2$$

Conditions: $\underline{W}$ and $\vec{b}$ are chosen so that

- All entries are independent
- All entries are symmetrically distributed (i.e. $X \stackrel{d}{=} -X$)
- $\phi(x) = \max\{x, 0\}$ is the ReLU function

Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \mathrm{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \cdots \cdot \mathrm{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid entries $X_j^{(i)} \sim \mathit{Bernoulli}(\frac{1}{2})$.

Proposition

$$\left\| \underline{M}\vec{1} \right\|^2 \stackrel{d}{=} \left\| \underline{J}\vec{1} \right\|^2$$

Conditions: $\underline{W}$ and $\vec{b}$ are chosen so that

- All entries are independent
- All entries are symmetrically distributed (i.e. $X \stackrel{d}{=} -X$)
- $\phi(x) = \max\{x, 0\}$ is the ReLU function

Proof Idea

Can show $\underline{M} \stackrel{d}{=} \underline{J}$ up to **conjugation** by random ± 1 Bernoulli's.

Limit Theorem for Product of Random Matrices - Hanin, N.

Fix $p \in (0, 1]$.

Limit Theorem for Product of Random Matrices - Hanin, N.

Fix $p \in (0,1]$. Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \mathrm{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \cdots \cdot \mathrm{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid $\{0,1\}$-valued entries: $X_j^{(i)} \sim \mathit{Bernoulli}(p)$.

Limit Theorem for Product of Random Matrices - Hanin, N.

Fix $p \in (0,1]$. Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \mathrm{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \cdots \cdot \mathrm{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid $\{0,1\}$-valued entries: $X_j^{(i)} \sim \mathit{Bernoulli}(p)$. If $\underline{W}^{(i)}$ independent, mean 0, variance $(n_i p)^{-1}$, finite moments, then:

Limit Theorem for Product of Random Matrices - Hanin, N.

Fix $p \in (0,1]$. Define the $n_d \times n_0$ product random matrix $\underline{M}$:

$$\underline{M} := \mathrm{Diag}(\vec{X}^{(d)})W^{(d)} \cdot \cdots \cdot \mathrm{Diag}(\vec{X}^{(1)})W^{(1)}$$

where $\vec{X}^{(i)} \in \mathbb{R}^{n_i}$ has iid $\{0,1\}$-valued entries: $X_j^{(i)} \sim \mathit{Bernoulli}(p)$. If $\underline{W}^{(i)}$ independent, mean 0, variance $(n_i p)^{-1}$, finite moments, then:

$$\left\|\underline{M}\vec{1}\right\|^2 \approx \exp\left(\sqrt{\left(\frac{3}{p}-1\right)\beta} \cdot \mathcal{N}(0,1) - \frac{1}{2}\left(\frac{3}{p}-1\right)\beta\right)$$

Theorem (Precise Version)

$$\left\| \underline{M}\vec{1} \right\|^2 \approx \exp\left(\sqrt{\left(\frac{3}{p} - 1\right)} \beta \mathcal{N}(0,1) - \frac{1}{2}\left(\frac{3}{p} - 1\right) \beta \right)$$

where "$\approx$" means:

Theorem (Precise Version)

$$\left\|\underline{M}\vec{1}\right\|^2 \approx \exp\left(\sqrt{\left(\frac{3}{p}-1\right)}\,\beta\mathcal{N}(0,1) - \frac{1}{2}\left(\frac{3}{p}-1\right)\beta\right)$$

where "≈" means:

Moments: For any $k \geq 0$, have:

$$\mathbf{E}\left[\left\|\underline{M}\vec{1}\right\|^{2k}\right] = \exp\left(\left(\frac{3}{p}-1\right)\binom{k}{2}\beta + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$

Theorem (Precise Version)

$$\left\| \underline{M} \vec{1} \right\|^2 \approx \exp\left(\sqrt{\left(\frac{3}{p} - 1\right)} \beta \mathcal{N}(0,1) - \frac{1}{2}\left(\frac{3}{p} - 1\right)\beta \right)$$

where "$\approx$" means:

Moments: For any $k \geq 0$, have:

$$\mathbf{E}\left[\left\| \underline{M} \vec{1} \right\|^{2k} \right] = \exp\left(\left(\frac{3}{p} - 1\right) \binom{k}{2} \beta + O\left(\sum_{i=0}^{d} \frac{1}{n_i^2} \right) \right)$$

Kolmogorov-Smirnov distance: $\exists C$ s.t. the cumulative distribution functions are close in L^∞ norm:

$$\left\| \Phi_{\ln\left(\left\|\underline{M}\vec{1}\right\|^2\right)} - \Phi_{\sqrt{\left(\frac{3}{p}-1\right)\beta} \cdot \mathcal{N}(0,1) - \frac{1}{2}\left(\frac{3}{p}-1\right)\beta} \right\|_\infty \leq C \left(\frac{\sum n_i^{-2}}{\sum n_i^{-1}} \right)^{1/5}$$

Part 3: Proof Ideas

- Where does the $\frac{3}{p}$ comes from?!?!
- Moments: Path counting
- Kolmogorov-Smirnov Distance: Martingales

Proposition

The k-th moment of $\left\|\underline{M}\vec{1}\right\|^2$ is

$$\mathbf{E}\left[\left\|\underline{M}\vec{1}\right\|^{2k}\right] = \exp\left(\left(\frac{3}{p}-1\right)\binom{k}{2}\sum_{i=1}^{d}\frac{1}{n_i} + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$

$$\approx \mathbf{E}\left[\exp\left(\sqrt{\left(\frac{3}{p}-1\right)\beta}\cdot\mathcal{N}(0,1) - \left(\frac{3}{p}-1\right)\frac{\beta}{2}\right)^k\right]$$

Proposition

The k-th moment of $\left\|\underline{M}\vec{1}\right\|^2$ is

$$\mathbf{E}\left[\left\|\underline{M}\vec{1}\right\|^{2k}\right] = \exp\left(\left(\frac{3}{p}-1\right)\binom{k}{2}\sum_{i=1}^{d}\frac{1}{n_i} + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$

$$\approx \mathbf{E}\left[\exp\left(\sqrt{\left(\frac{3}{p}-1\right)\beta}\cdot\mathcal{N}(0,1) - \left(\frac{3}{p}-1\right)\frac{\beta}{2}\right)^k\right]$$

Remark: Proof goes by counting paths in the neural network: a kind of "neural network" version of moments of Wigner's semi-circle law proof.

Proposition

The k-th moment of $\left\|\underline{M}\vec{1}\right\|^2$ is

$$\mathbf{E}\left[\left\|\underline{M}\vec{1}\right\|^{2k}\right] = \exp\left(\left(\frac{3}{p}-1\right)\binom{k}{2}\sum_{i=1}^{d}\frac{1}{n_i} + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$
$$\approx \mathbf{E}\left[\exp\left(\sqrt{\left(\frac{3}{p}-1\right)\beta}\cdot\mathcal{N}(0,1) - \left(\frac{3}{p}-1\right)\frac{\beta}{2}\right)^k\right]$$

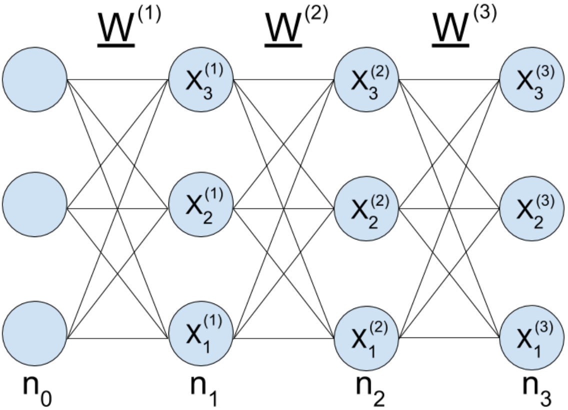
Remark: Proof goes by counting paths in the neural network: a kind of "neural network" version of moments of Wigner's semi-circle law proof.

Proposition

The result when $k = 1$ is:

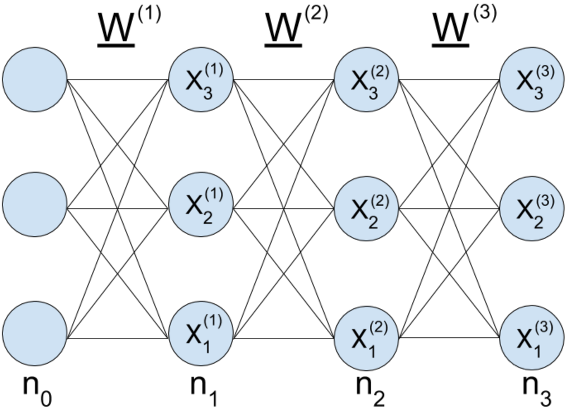
$$\mathbf{E}\left[\left\|\underline{M}\vec{1}\right\|^2\right] = 1$$

Proof Idea for $\mathbf{E}[||M\vec{1}||^2]$



Think of $\underline{M} := \left(\text{Diag}(\vec{X}^{(d)})\underline{W}^{(d)}\right) \cdots \left(\text{Diag}(\vec{X}^{(1)})\underline{W}^{(1)}\right)$ as a graph.

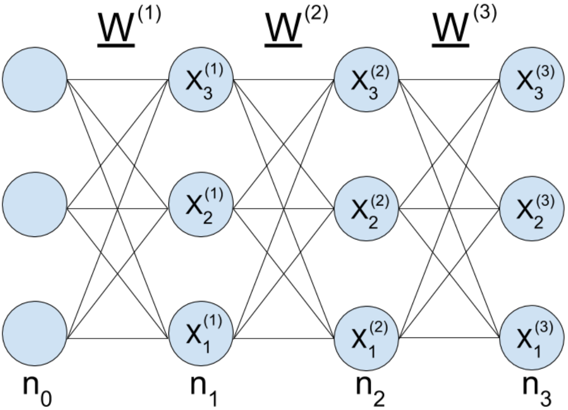
Proof Idea for $\mathbf{E}[||M\vec{1}||^2]$



Think of $\underline{M} := \left(\text{Diag}(\vec{X}^{(d)})\underline{W}^{(d)}\right) \cdots \left(\text{Diag}(\vec{X}^{(1)})\underline{W}^{(1)}\right)$ as a graph.

Edges represent the weights $W^{(i)}_{a,b}$.

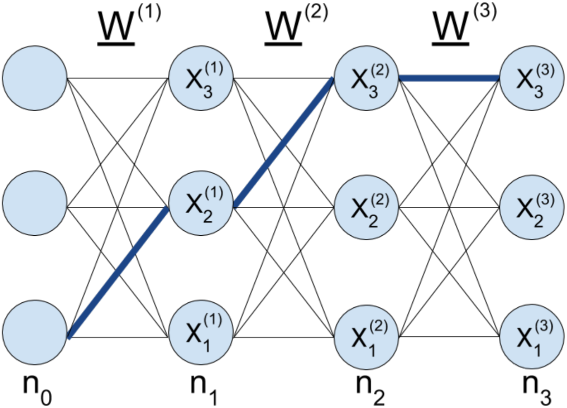
Proof Idea for $\mathbf{E}[||M\vec{1}||^2]$



Think of $\underline{M} := \left(\text{Diag}(\vec{X}^{(d)})\underline{W}^{(d)}\right) \cdots \left(\text{Diag}(\vec{X}^{(1)})\underline{W}^{(1)}\right)$ as a graph.

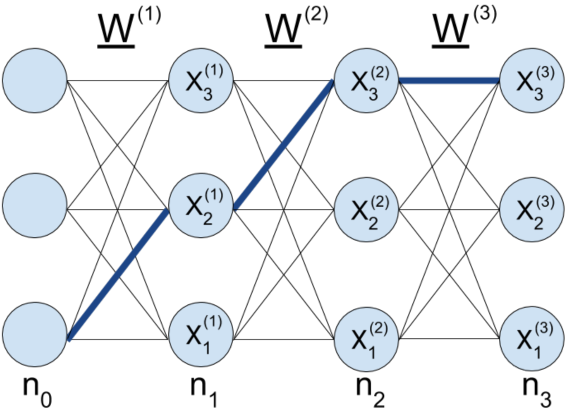
Edges represent the weights $W^{(i)}_{a,b}$. Vertices represent the Bernoulli's$X^{(i)}_a$.

Proof Idea for $\mathbf{E}[||M\vec{1}||^2]$



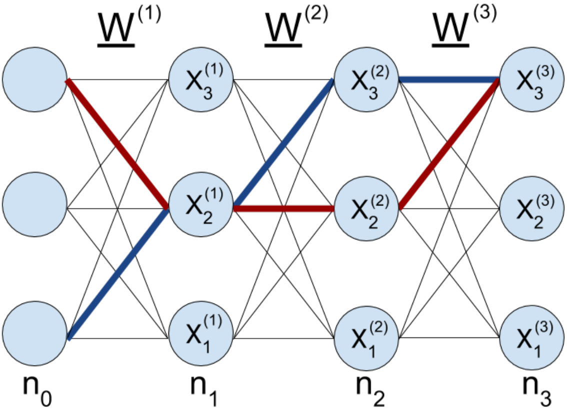
$\underline{M}_{a,b}$ is the sum over ALL paths starting at $b \in \{1, 2 \ldots, n_0\}$ and ending at $a \in \{1, 2 \ldots, n_d\}$.

Proof Idea for $\mathbf{E}[||M\vec{1}||^2]$



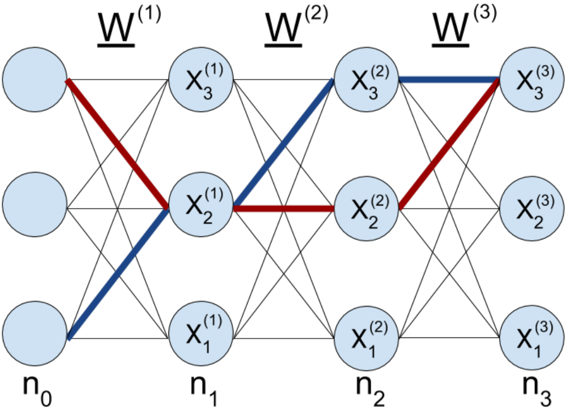
$\underline{M}_{a,b}$ is the sum over ALL paths starting at $b \in \{1, 2 \ldots, n_0\}$ and ending at $a \in \{1, 2 \ldots, n_d\}$. The weight of each path is the product of weights along path. I.e. $\underline{M}_{a,b} = \sum_\pi \prod_{i=1}^d X_{\pi_i}^{(i)} W_{\pi_{i-1},\pi_i}^{(i)}$.

Proof Idea for $\mathbf{E}||M\vec{1}||^2$



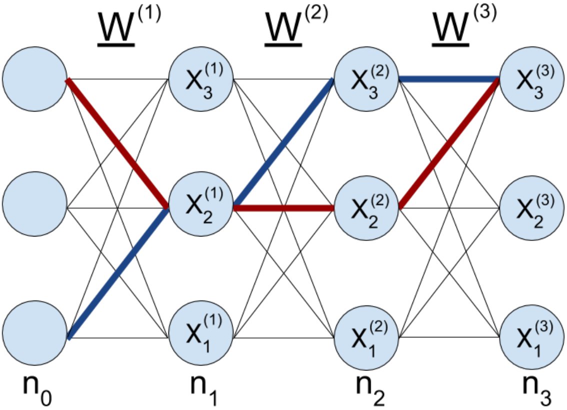
$||M\vec{1}||^2$ is a sum over **pairs of paths** that end at the same point.

Proof Idea for $\mathbf{E}||M\vec{1}||^2$



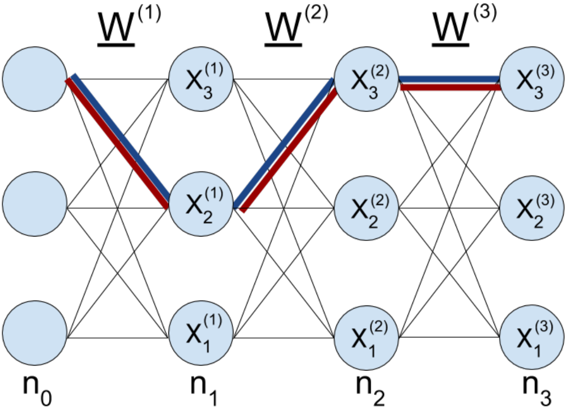
$||M\vec{1}||^2$ is a sum over **pairs of paths** that end at the same point.The weight of pair of paths is the product over edge & vertex weights.

Proof Idea for $\mathbf{E}||M\vec{1}||^2$



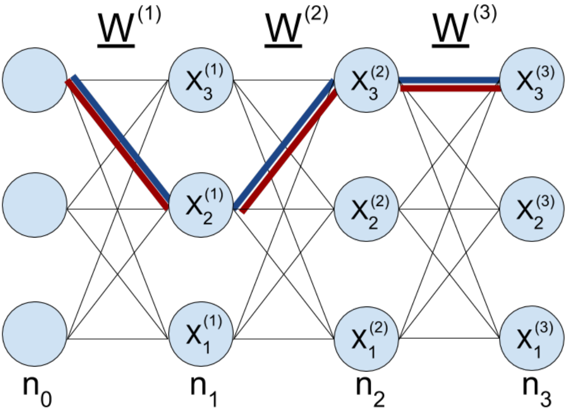
Most pairs of paths have $\mathbf{E}\left[\prod X^{(i)}_{\pi_i} W^{(i)}_{\pi_{i-1},\pi_i}\right] = 0$, because the weights $W^{(i)}_{a,b}$ are **independent** and **mean zero** ($\mathbf{E}\left[W^{(i)}_{a,b}\right] = 0$)

Proof Idea for $\mathbf{E}||M\vec{1}||^2$



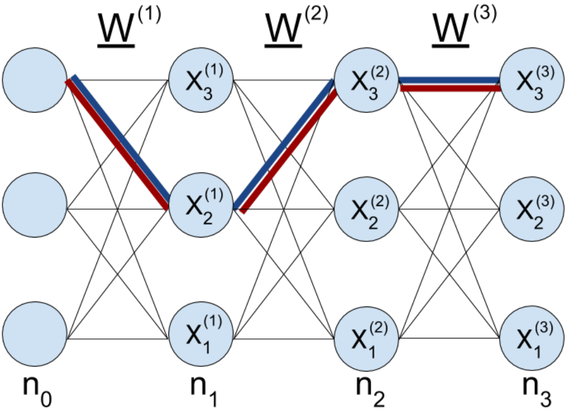
Non-zero contribution only if **the pair of paths overlap!**

Proof Idea for $\mathbf{E}||M\vec{1}||^2$



$$\mathbf{E}\left[\left(W_a^{(i)}\right)^2\right] = \frac{1}{pn_i}, \mathbf{E}\left[\left(X_b^{(i)}\right)^2\right] = p, \ \#\{\text{paths}\} = \prod_{i=1}^{d} n_i$$

Proof Idea for $\mathbf{E}||M\vec{1}||^2$



$$\mathbf{E}\left[\left(W_a^{(i)}\right)^2\right] = \frac{1}{pn_i}, \mathbf{E}\left[\left(X_b^{(i)}\right)^2\right] = p, \ \#\{\text{paths}\} = \prod_{i=1}^d n_i$$

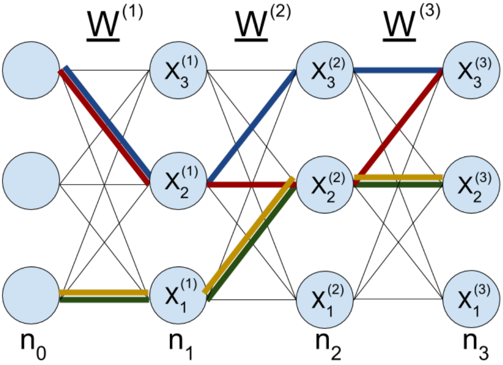
$$\mathbf{E}\left[||M\vec{1}||^2\right] = \#\{\text{paths}\}\left(\prod_{i=1}^d \mathbf{E}\left[\left(W_a^{(i)}\right)^2\right]\mathbf{E}\left[\left(X_b^{(i)}\right)^2\right]\right) = 1$$

Proposition

The second moment of $\left\|M\vec{1}\right\|^2$ is

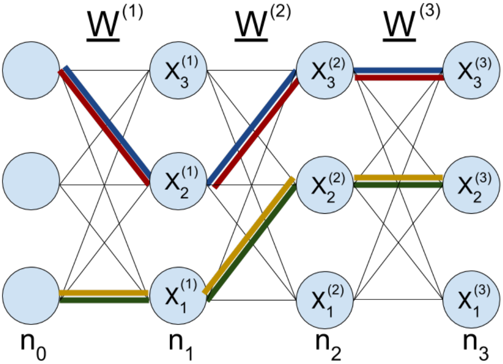
$$\mathbf{E}\left[\left\|\underline{M}\vec{1}\right\|^4\right] = \exp\left(\left(\frac{3}{p} - 1\right)\sum_{i=1}^{d}\frac{1}{n_i} + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



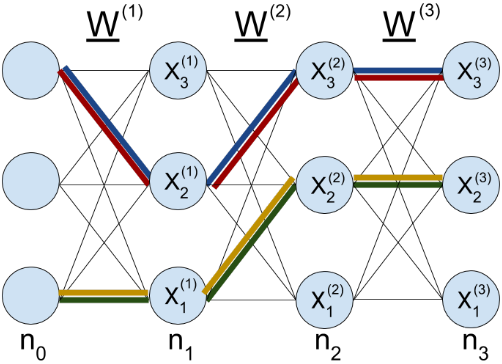
$||\underline{M}\vec{1}||^4$ is a sum over **4-tuples of paths** that end in **pairs** at the right.
(Must have: Red with Blue, Green with Yellow at right endpoint.)

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



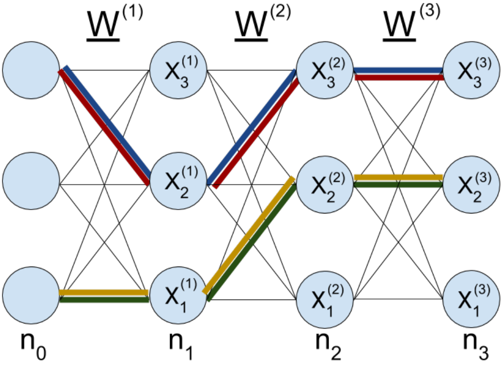
Non-zero contribution to $\mathbf{E}||M\vec{1}||^4$ when every edge is covered an **even number of times**.

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



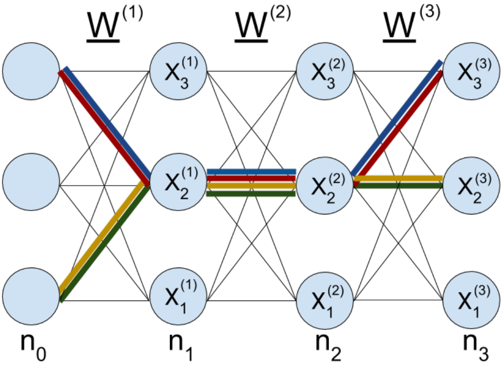
Non-zero contribution to $\mathbf{E}||M\vec{1}||^4$ when every edge is covered an **even number of times**. Interaction between the pairs of paths will make $\mathbf{E}||M\vec{1}||^4 \neq \left(\mathbf{E}||M\vec{1}||^2\right)^2$

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



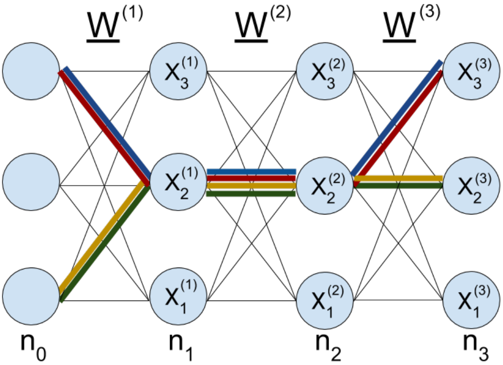
Non-zero contribution to $\mathbf{E}||M\vec{1}||^4$ when every edge is covered an **even number of times**. Interaction between the pairs of paths will make $\mathbf{E}||M\vec{1}||^4 \neq \left(\mathbf{E}||M\vec{1}||^2\right)^2$ Since $\mathbf{E}\left[\left\|M\vec{1}\right\|^2\right] = 1$ can think of the pairs of paths chosen "at random".

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



An edge covered **more than twice is rare**.

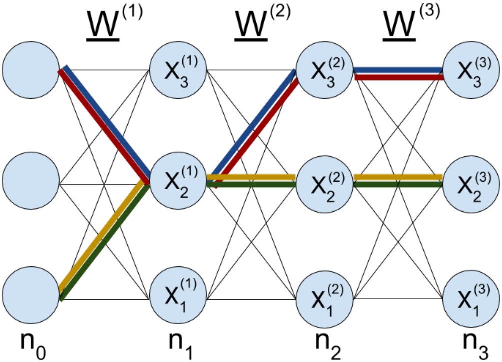
Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



An edge covered **more than twice is rare**.

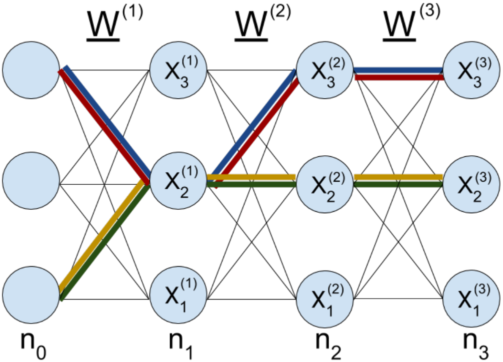
Contribution like $Cn_{i-1}^{-1}n_i^{-1} = O\left(n_{i-1}^{-2}\right) + O\left(n_i^{-2}\right)$.

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



A **simple collision** gives an extra factor of $\frac{1}{p}$.

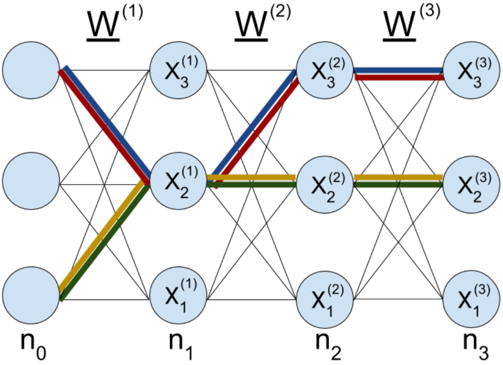
Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



A **simple collision** gives an extra factor of $\frac{1}{p}$.

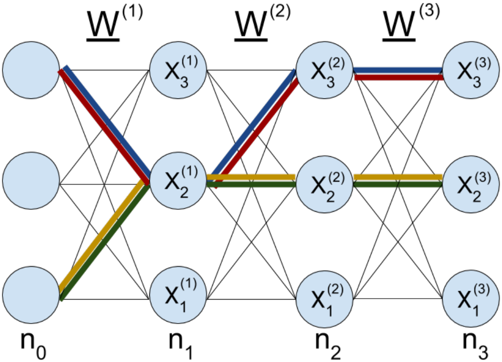
(Since $\mathbf{E}\left[\left(X_a^{(i)}\right)^4\right] = p$ but $\mathbf{E}\left[\left(X_a^{(i)}\right)^2\right]\mathbf{E}\left[\left(X_b^{(i)}\right)^2\right] = p^2$)

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



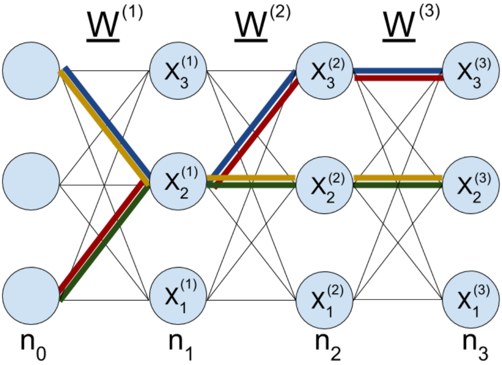
For each **simple collision**: There are **3 ways** to **group the 4 paths into 2 pairs.**

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



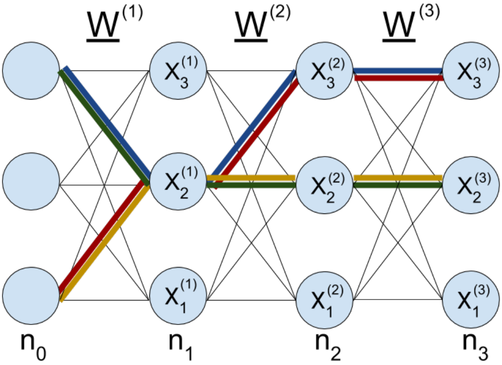
For each **simple collision**: There are **3 ways** to **group the 4 paths into 2 pairs.** You can pair Red↔Blue, Yellow↔Green. (The "boring" pairing)

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



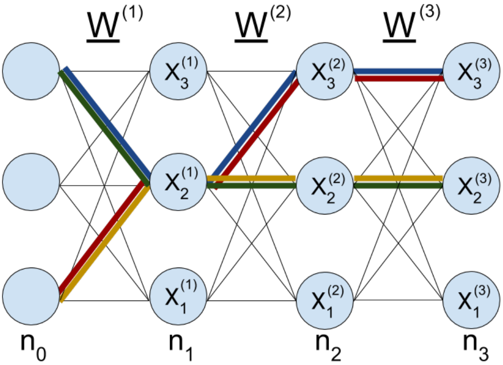
For each **simple collision**: There are **3 ways** to **group the 4 paths into 2 pairs.** ... OR Yellow↔Blue, Red↔Green.

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



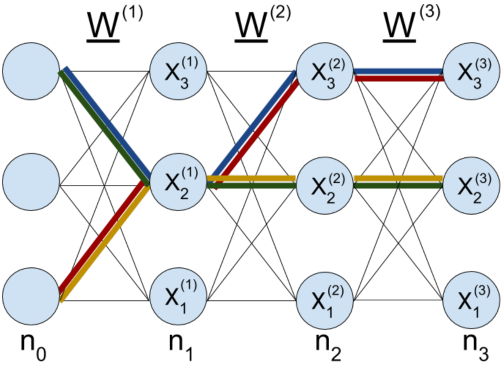
For each **simple collision**: There are **3 ways** to **group the 4 paths into 2 pairs.** ... OR Green$\leftrightarrow$Blue, Red$\leftrightarrow$Yellow.

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



$$\mathbf{E}\left[||\underline{M}\vec{1}||^4\right] \approx \mathcal{E}_{\text{paths}}\left[\left(\frac{3}{p}\right)^{\#\text{ of collisions}}\right] \approx \prod_{i=1}^{d}\left(1\left(1-\frac{1}{n_i}\right)+\frac{3}{p}\frac{1}{n_i}\right)$$

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^4$



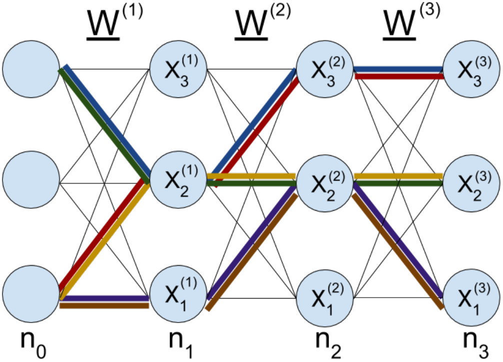
$$\mathbf{E}\left[||\underline{M}\vec{1}||^4\right] \approx \mathcal{E}_{\text{paths}}\left[\left(\frac{3}{p}\right)^{\#\text{ of collisions}}\right] \approx \exp\left(\left(\frac{3}{p}-1\right)\sum_{i=1}^{d}\frac{1}{n_i}\right)$$

Proposition

The k-th moment of $\left\|M\vec{1}\right\|^2$ is

$$\mathbf{E}\left[\left\|\underline{M\vec{1}}\right\|^{2k}\right] = \exp\left(\left(\frac{3}{p}-1\right)\binom{k}{2}\sum_{i=1}^{d}\frac{1}{n_i} + O\left(\sum_{i=0}^{d}\frac{1}{n_i^2}\right)\right)$$

Proof idea for $\mathbf{E}||\underline{M}\vec{1}||^{2k}$



$$\mathbf{E}\left[||\underline{M}\vec{1}||^{2k}\right] \approx \mathcal{E}_{\text{paths}}\left[\left(\frac{3}{p}\right)^{\#\text{collisions}}\right] \approx \prod_{i=1}^{d}\left(1\left(1-\binom{k}{2}\frac{1}{n_i}\right)+\frac{3}{p}\binom{k}{2}\frac{1}{n_i}\right)$$

$$\approx \exp\left(\left(\frac{3}{p}-1\right)\binom{k}{2}\sum_{i=1}^{d}\frac{1}{n_i}\right)$$

Proposition

$$\ln\left(\left\|\underline{M}\vec{1}\right\|^2\right) \approx \left(\frac{3}{p}-1\right)\beta\mathcal{N}(0,1)-\frac{1}{2}\left(\frac{3}{p}-1\right)\beta$$

in the sense that the Kolmogorov-Smirnov distance $d(X,Y)=\sup_t\left|\mathbf{P}(X\leq t)-\mathbf{P}\left(Y\leq t\right)\right|$ is small.

Proof Idea for ln $||\underline{M}\vec{1}||^2$

Define:

$$\vec{x}^{(j)} = \underline{B}^{(j)}\underline{W}^{(j)} \cdots \underline{B}^{(1)}\underline{W}^{(1)}\vec{1}$$

and let $\mathcal{F}_j$ be the filtration for first j layers.

Proof Idea for $\ln ||\underline{M}\vec{1}||^2$

Define:

$$\vec{x}^{(j)} = \underline{B}^{(j)}\underline{W}^{(j)} \cdots \underline{B}^{(1)}\underline{W}^{(1)}\vec{1}$$

and let $\mathcal{F}_j$ be the filtration for first j layers. Then:

$$\ln \left\| \underline{M}\vec{1} \right\|^2 = \ln \left\| \vec{x}^{(d)} \right\|^2 = \sum_{i=1}^{d} \ln \left(\frac{\left\| \vec{x}^{(i)} \right\|^2}{\left\| \vec{x}^{(i-1)} \right\|^2} \right)$$

$$= \sum_{i=1}^{d} \left\{ \ln \left(\frac{\left\| \vec{x}^{(i)} \right\|^2}{\left\| \vec{x}^{(i-1)} \right\|^2} \right) - \mathbf{E}\left[\ln \left(\frac{\left\| \vec{x}^{(i)} \right\|^2}{\left\| \vec{x}^{(i-1)} \right\|^2} \right) |\mathcal{F}_{i-1} \right] \right\} \quad (1)$$

$$+ \sum_{i=1}^{d} \mathbf{E}\left[\ln \left(\frac{\left\| \vec{x}^{(i)} \right\|^2}{\left\| \vec{x}^{(i-1)} \right\|^2} \right) |\mathcal{F}_{i-1} \right] \quad (2)$$

Proof Idea for $\ln ||\underline{M}\vec{1}||^2$

Define:

$$\vec{x}^{(j)} = \underline{B}^{(j)}\underline{W}^{(j)} \cdots \underline{B}^{(1)}\underline{W}^{(1)}\vec{1}$$

and let $\mathcal{F}_j$ be the filtration for first j layers. Then:

$$\ln \left\|\underline{M}\vec{1}\right\|^2 = \ln \left\|\vec{x}^{(d)}\right\|^2 = \sum_{i=1}^{d} \ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right)$$

$$= \sum_{i=1}^{d} \left\{ \ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right) - \mathbf{E}\left[\ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right) |\mathcal{F}_{i-1} \right] \right\} \quad (1)$$

$$+ \sum_{i=1}^{d} \mathbf{E}\left[\ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right) |\mathcal{F}_{i-1} \right] \quad (2)$$

(1) is a martingale difference sequence with increments of variance $\approx \left(\frac{3}{p} - 1\right) n_i^{-1}$ and fourth moments $O\left(n_i^{-2}\right) \implies$ close to Gaussian.

Proof Idea for $\ln ||\underline{M}\vec{1}||^2$

Define:

$$\vec{x}^{(j)} = \underline{B}^{(j)}\underline{W}^{(j)} \cdots \underline{B}^{(1)}\underline{W}^{(1)}\vec{1}$$

and let $\mathcal{F}_j$ be the filtration for first j layers.Then:

$$\ln \left\|\underline{M}\vec{1}\right\|^2 = \ln \left\|\vec{x}^{(d)}\right\|^2 = \sum_{i=1}^{d} \ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right)$$

$$= \sum_{i=1}^{d} \left\{ \ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right) - \mathbf{E}\left[\ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right) |\mathcal{F}_{i-1} \right] \right\} \quad (1)$$

$$+ \sum_{i=1}^{d} \mathbf{E}\left[\ln \left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2} \right) |\mathcal{F}_{i-1} \right] \quad (2)$$

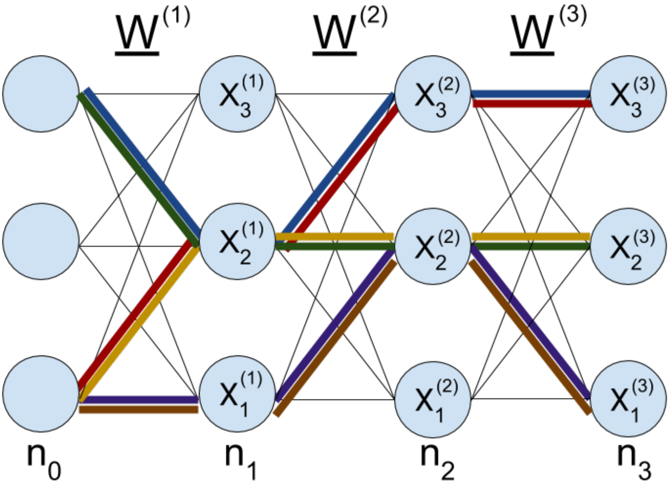
(1) is a martingale difference sequence with increments of variance $\approx \left(\frac{3}{p} - 1\right) n_i^{-1}$ and fourth moments $O\left(n_i^{-2}\right) \implies$ close to Gaussian.

(2) is approximately constant $\approx \frac{1}{2}\left(\frac{3}{p} - 1\right) n_i^{-1} + O\left(n_i^{-2}\right)$.

(2) is approximately constant

$$
\begin{aligned}
&\mathbf{E}\left[\ln\left(\frac{\left\|\vec{x}^{(i)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2}\right)|\mathcal{F}_{i-1}\right] \\
=&\mathbf{E}\left[\ln\left(1+\frac{\left\|\vec{x}^{(i)}\right\|^2-\left\|\vec{x}^{(i-1)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2}\right)|\mathcal{F}_{i-1}\right] \\
\approx&\mathbf{E}\left[\frac{\left\|\vec{x}^{(i)}\right\|^2-\left\|\vec{x}^{(i-1)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2}|\mathcal{F}_{i-1}\right]+\frac{1}{2}\mathbf{E}\left[\left(\frac{\left\|\vec{x}^{(i)}\right\|^2-\left\|\vec{x}^{(i-1)}\right\|^2}{\left\|\vec{x}^{(i-1)}\right\|^2}\right)^2|\mathcal{F}_{i-1}\right] \\
\approx&0+\frac{1}{2}\left(\frac{3}{p}-1\right)\frac{1}{n_i}+\frac{\mu_4-3}{2n_ip}\frac{\left\|\vec{x}^{(i-1)}\right\|_4^4}{\left\|\vec{x}^{(i-1)}\right\|_2^4}
\end{aligned}
$$

The end!



Limit Theorem for Product of Random Matrices - Arbitrary vectors

If $\vec{x}$ is an arbitrary vector, then:

$$\|\underline{M}\vec{x}\|^2 \approx \log-\text{normal}\left(\left(\frac{3}{p}-1\right)\sum_{i=1}^{d}\frac{1}{n_i}+\frac{\mu_4-3}{n_1 p}\frac{\|\vec{x}\|_4^4}{\|\vec{x}\|_2^4}\right)$$

where $\mu_4 = \mathbf{E}\left[\left(W_{j,k}^{(i)}\right)^4\right]$ is the fourth moment of the random weights $W_{j,k}^{(i)}$