Scaling ResNets in the large-depth regime

ROUEN, AUGUST 2022

Gérard Biau
Team

Adeline Fermanian  
MINES PARIS - PSL

Pierre Marion  
SORBONNE UNIVERSITY

Jean-Philippe Vert  
GOOGLE RESEARCH
Agenda

Learning with ResNets

Scaling deep ResNets

Scaling in the continuous-time setting

Beyond initialization
Agenda

- Learning with ResNets
- Scaling deep ResNets
- Scaling in the continuous-time setting
- Beyond initialization
How most people see the supervised learning problem

Learn how to build an image-recognizing convolutional neural network with Python and Keras in less than 15 minutes!

https://towardsdatascience.com/cat-dog-or-elon-musk-145658489730
How machine learners see the supervised learning problem

Goal: understand the relationship between $x \in \mathbb{R}^{n_{in}}$ and $y \in \mathbb{R}^{n_{out}}$. 
How statisticians see the supervised learning problem

▷ Goal: understand the relationship between \( x \in \mathbb{R}_{\text{in}}^{n_{\text{in}}} \) and \( y \in \mathbb{R}_{\text{out}}^{n_{\text{out}}} \).

▷ Data: \((x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}_{\text{in}}^{n_{\text{in}}} \times \mathbb{R}_{\text{out}}^{n_{\text{out}}}, \) i.i.d. \( \sim (x, y) \).
How statisticians see the supervised learning problem

- **Goal**: understand the relationship between $x \in \mathbb{R}^{n_{in}}$ and $y \in \mathbb{R}^{n_{out}}$.
- **Data**: $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^{n_{in}} \times \mathbb{R}^{n_{out}}$, i.i.d. $\sim (x, y)$.
- **Model**: $\{F_\pi : \mathbb{R}^{n_{in}} \mapsto \mathbb{R}^{n_{out}}, \pi \in \Pi\}$. 

- **Loss function** $\ell : \mathbb{R}^{n_{out}} \times \mathbb{R}^{n_{out}} \to \mathbb{R}^+$. 

- **Regression**: $\ell(F_\pi(x), y) = (y - F_\pi(x))^2$.
- **Binary classification**: $\ell(F_\pi(x), y) = 1[\ yF_\pi(x) \leq 0]$. 

- **Theoretical risk minimization**: choose $\pi^\star \in \arg\min_{\pi \in \Pi} L(\pi) = \mathbb{E}(\ell(F_\pi(x), y))$. 

- **Empirical risk minimization**: choose $\pi_n \in \arg\min_{\pi \in \Pi} L_n(\pi) = \frac{1}{n} \sum_{i=1}^{n} \ell(F_\pi(x_i), y_i)$. 

How statisticians see the supervised learning problem

- **Goal**: understand the relationship between \( x \in \mathbb{R}^{n_{\text{in}}} \) and \( y \in \mathbb{R}^{n_{\text{out}}} \).
- **Data**: \((x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^{n_{\text{in}}} \times \mathbb{R}^{n_{\text{out}}}\), i.i.d. \( \sim (x, y) \).
- **Model**: \( \{F_\pi : \mathbb{R}^{n_{\text{in}}} \mapsto \mathbb{R}^{n_{\text{out}}}, \pi \in \Pi\} \).
- **Loss function** \( \ell : \mathbb{R}^{n_{\text{out}}} \times \mathbb{R}^{n_{\text{out}}} \rightarrow \mathbb{R}_+ \).
How statisticians see the supervised learning problem

- **Goal**: understand the relationship between $x \in \mathbb{R}^{n_{in}}$ and $y \in \mathbb{R}^{n_{out}}$.
- **Data**: $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^{n_{in}} \times \mathbb{R}^{n_{out}}$, i.i.d. $\sim (x, y)$.
- **Model**: $\{F_\pi : \mathbb{R}^{n_{in}} \rightarrow \mathbb{R}^{n_{out}}, \pi \in \Pi\}$.
- **Loss function** $\ell : \mathbb{R}^{n_{out}} \times \mathbb{R}^{n_{out}} \rightarrow \mathbb{R}_+$. 
- **Regression**: $\ell(F_\pi(x), y) = (y - F_\pi(x))^2$
How statisticians see the supervised learning problem

- **Goal**: understand the relationship between $x \in \mathbb{R}^{n_{\text{in}}}$ and $y \in \mathbb{R}^{n_{\text{out}}}$.  

- **Data**: $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^{n_{\text{in}}} \times \mathbb{R}^{n_{\text{out}}}$, i.i.d. $\sim (x, y)$.  

- **Model**: $\{F_\pi : \mathbb{R}^{n_{\text{in}}} \mapsto \mathbb{R}^{n_{\text{out}}}, \pi \in \Pi\}$.  

- **Loss function** $\ell : \mathbb{R}^{n_{\text{out}}} \times \mathbb{R}^{n_{\text{out}}} \rightarrow \mathbb{R}^+$.  

- **Regression**: $\ell(F_\pi(x), y) = (y - F_\pi(x))^2$  
  **Binary classification**: $\ell(F_\pi(x), y) = 1[yF_\pi(x) \leq 0]$.  

Goal: understand the relationship between $x \in \mathbb{R}^{n_{in}}$ and $y \in \mathbb{R}^{n_{out}}$.

Data: $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^{n_{in}} \times \mathbb{R}^{n_{out}}$, i.i.d. $\sim (x, y)$.

Model: $\{F_\pi : \mathbb{R}^{n_{in}} \mapsto \mathbb{R}^{n_{out}}, \pi \in \Pi\}$.

Loss function $\ell : \mathbb{R}^{n_{out}} \times \mathbb{R}^{n_{out}} \rightarrow \mathbb{R}_+$.

Regression: $\ell(F_\pi(x), y) = (y - F_\pi(x))^2$  
Binary classification: $\ell(F_\pi(x), y) = 1[yF_\pi(x) \leq 0]$.

Theoretical risk minimization: choose

$$\pi^* \in \arg\min_{\pi \in \Pi} \mathcal{L}(\pi) = \mathbb{E}(\ell(F_\pi(x), y)).$$
How statisticians see the supervised learning problem

- **Goal**: understand the relationship between $x \in \mathbb{R}^{n_{in}}$ and $y \in \mathbb{R}^{n_{out}}$.
- **Data**: $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^{n_{in}} \times \mathbb{R}^{n_{out}}$, i.i.d. $\sim (x, y)$.
- **Model**: $\{F_{\pi} : \mathbb{R}^{n_{in}} \mapsto \mathbb{R}^{n_{out}}, \pi \in \Pi\}$.
- **Loss function** $\ell : \mathbb{R}^{n_{out}} \times \mathbb{R}^{n_{out}} \to \mathbb{R}_+$.
- **Regression**: $\ell(F_{\pi}(x), y) = (y - F_{\pi}(x))^2$  Binary classification: $\ell(F_{\pi}(x), y) = 1[yF_{\pi}(x) \leq 0]$.
- **Theoretical risk minimization**: choose
  \[
  \pi^* \in \arg\min_{\pi \in \Pi} \mathcal{L}(\pi) = \mathbb{E}(\ell(F_{\pi}(x), y)).
  \]
- **Empirical risk minimization**: choose
  \[
  \pi_n \in \arg\min_{\pi \in \Pi} \mathcal{L}_n(\pi) = \frac{1}{n} \sum_{i=1}^{n} \ell(F_{\pi}(x_i), y_i).
  \]
Goal: understand the relationship between $x \in \mathbb{R}^{n_{\text{in}}}$ and $y \in \mathbb{R}^{n_{\text{out}}}$.

Data: $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^{n_{\text{in}}} \times \mathbb{R}^{n_{\text{out}}}$, i.i.d. $\sim (x, y)$.

Model: $\{F_\pi : \mathbb{R}^{n_{\text{in}}} \mapsto \mathbb{R}^{n_{\text{out}}}, \pi \in \Pi \}$.

Loss function $\ell : \mathbb{R}^{n_{\text{out}}} \times \mathbb{R}^{n_{\text{out}}} \to \mathbb{R}^+$.

Regression: $\ell(F_\pi(x), y) = (y - F_\pi(x))^2$ Binary classification: $\ell(F_\pi(x), y) = \mathbb{1}[yF_\pi(x) \leq 0]$.

Theoretical risk minimization: choose

$$\pi^* \in \arg\min_{\pi \in \Pi} \mathcal{L}(\pi) = \mathbb{E}(\ell(F_\pi(x), y)).$$

Empirical risk minimization: choose

$$\pi_n \in \arg\min_{\pi \in \Pi} \mathcal{L}_n(\pi) = \frac{1}{n} \sum_{i=1}^{n} \ell(F_\pi(x_i), y_i).$$
Residual neural networks (ResNets)

Sequence of hidden states $h_0, \ldots, h_L \in \mathbb{R}^d$ defined by recurrence:
Residual neural networks (ResNets)

- Sequence of hidden states $h_0, \ldots, h_L \in \mathbb{R}^d$ defined by recurrence:

\[
\begin{align*}
    h_0 &= A x, \\
    h_{k+1} &= h_k + f(h_k, \theta_{k+1}), \\
    F_\pi(x) &= B h_L.
\end{align*}
\]
Residual neural networks (ResNets)

Sequence of hidden states $h_0, \ldots, h_L \in \mathbb{R}^d$ defined by recurrence:

$$h_0 = Ax, \quad h_{k+1} = h_k + f(h_k, \theta_{k+1}), \quad F_\pi(x) = B h_L.$$
Residual neural networks (ResNets)

Sequence of hidden states $h_0, \ldots, h_L \in \mathbb{R}^d$ defined by recurrence:

$$h_0 = Ax, \quad h_{k+1} = h_k + f(h_k, \theta_{k+1}), \quad F_\pi(x) = B h_L.$$ 

Different forms for $f : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d =$ different architectures.
Residual neural networks (ResNets)

- Sequence of hidden states $h_0, \ldots, h_L \in \mathbb{R}^d$ defined by recurrence:
  \[
  h_0 = Ax, \quad h_{k+1} = h_k + f(h_k, \theta_{k+1}), \quad F_\pi(x) = Bh_L.
  \]

- Different forms for $f : \mathbb{R}^d \times \mathbb{R}^p \rightarrow \mathbb{R}^d = \text{different architectures}$.

**Original Parametric Simple General ResNet**

\[
  f(h_k, \theta_{k+1}) = V_{k+1} \text{ReLU}(W_{k+1}h_k + b_{k+1})
\]

- $\text{ReLU}(x) = \max(x, 0) = \text{activation function}$
- $\theta_k = (W_k, b_k) = \text{weight matrice + bias}$
- $\pi = (A, B, (V_k)_{1 \leq k \leq L}, (\theta_k)_{1 \leq k \leq L})$

He et al. (2016)
Residual neural networks (ResNets)

- Sequence of hidden states $h_0, \ldots, h_L \in \mathbb{R}^d$ defined by recurrence:
  \[ h_0 = Ax, \quad h_{k+1} = h_k + f(h_k, \theta_{k+1}), \quad F_\pi(x) = Bh_L. \]

- Different forms for $f : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d = \text{different architectures.}$

**Original Parametric Simple General ResNet**

\[ f(h_k, \theta_{k+1}) = V_{k+1} \sigma(W_{k+1} h_k + b_{k+1}) \]

- $\sigma = \text{activation function}$
- $\theta_k = (W_k, b_k) = \text{weight matrice + bias}$
- $\pi = (A, B, (V_k)_{1 \leq k \leq L}, (\theta_k)_{1 \leq k \leq L})$

He et al. (2016)
Residual neural networks (ResNets)

- Sequence of hidden states \( h_0, \ldots, h_L \in \mathbb{R}^d \) defined by recurrence:

\[
h_0 = Ax, \quad h_{k+1} = h_k + f(h_k, \theta_{k+1}), \quad F_\pi(x) = B h_L.
\]

- Different forms for \( f : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d \) = different architectures.

Original Parametric Simple General ResNet

- \( f(h_k, \theta_{k+1}) = V_{k+1} \sigma(h_k) \)
  - \( \sigma = \text{activation function} \)
  - \( \theta_k = \emptyset \)
  - \( \pi = (A, B, (V_k)_{1 \leq k \leq L}) \)

He et al. (2016)
Residual neural networks (ResNets)

- Sequence of hidden states $h_0, \ldots, h_L \in \mathbb{R}^d$ defined by recurrence:
  \[
  h_0 = Ax, \quad h_{k+1} = h_k + f(h_k, \theta_{k+1}), \quad F_\pi(x) = Bh_L.
  \]

- Different forms for $f : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d = \text{different architectures.}$

### Original Parametric Simple General ResNet

- $f(h_k, \theta_{k+1}) = V_{k+1}g(h_k, \theta_{k+1})$
  
  - $g : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d$
  - $\theta_k = \text{parameters}$
  - $\pi = (A, B, (V_k)_{1 \leq k \leq L}, (\theta_k)_{1 \leq k \leq L})$

He et al. (2016)
The revolution of ResNets

Examples from the ImageNet dataset

https://blog.roboflow.com/introduction-to-imagenet
The revolution of ResNets

ImageNet performance over time

https://semiengineering.com/
new-vision-technologies-for-real-world-applications
The revolution of ResNets

ImageNet performance over time

https://semiengineering.com/new-vision-technologies-for-real-world-applications
Deep learning $\rightarrow$ neural ODE $\leftarrow$ ODE

Traditional neural networks

$$h_{k+1} = f(h_k, \theta_{k+1})$$
Deep learning → neural ODE ← ODE

- **Traditional** neural networks

\[ h_{k+1} = f(h_k, \theta_{k+1}) \]

- **Residual** neural networks (He et al., 2016)

\[ h_{k+1} = h_k + f(h_k, \theta_{k+1}) \]
Deep learning $\rightarrow$ neural ODE $\leftarrow$ ODE

- **Traditional** neural networks
  \[ h_{k+1} = f(h_k, \theta_{k+1}) \]

- **Residual** neural networks (He et al., 2016)
  \[ h_{k+1} = h_k + \frac{1}{L} f(h_k, \theta_{k+1}) \]
Deep learning $\rightarrow$ neural ODE $\leftarrow$ ODE

- **Traditional** neural networks
  \[ h_{k+1} = f(h_k, \theta_{k+1}) \]

- **Residual** neural networks (He et al., 2016)
  \[ h_{k+1} = h_k + \frac{1}{L} f(h_k, \theta_{k+1}) \]

- **Neural ODE** (Chen et al., 2018)
  \[ dH_t = f(H_t, \Theta_t) dt \]
Deep learning $\rightarrow$ neural ODE $\leftarrow$ ODE

- **Traditional** neural networks
  \[ h_{k+1} = f(h_k, \theta_{k+1}) \]

- **Residual** neural networks (He et al., 2016)
  \[ h_{k+1} = h_k + \frac{1}{L} f(h_k, \theta_{k+1}) \]

- **Neural ODE** (Chen et al., 2018)
  \[ dH_t = f(H_t, \Theta_t) \, dt \]
New network architectures: Runge-Kutta networks

Benning et al. (2019)
New network architectures: antisymmetric networks

Chang et al. (2019)
In summary

<table>
<thead>
<tr>
<th>ResNet</th>
<th>Neural ODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_0 = Ax$</td>
<td>$H_0 = Ax$</td>
</tr>
<tr>
<td>$h_{k+1} = h_k + \frac{1}{L}f(h_k, \theta_{k+1})$</td>
<td>$dH_t = f(H_t, \Theta_t)dt$</td>
</tr>
<tr>
<td>$F_\pi(x) = Bh_T$</td>
<td>$F_\Pi(x) = BH_1$</td>
</tr>
<tr>
<td>$f(h, \theta) = V\sigma(Wh + b)$</td>
<td></td>
</tr>
</tbody>
</table>
In summary

\begin{align*}
\textbf{ResNet} & \quad \textbf{Neural ODE} \\
 h_0 &= Ax & H_0 &= Ax \\
 h_{k+1} &= h_k + \frac{1}{L}f(h_k, \theta_{k+1}) & dH_t &= f(H_t, \Theta_t)dt \\
 F_\pi(x) &= Bh_T & F_\Pi(x) &= BH_1 \\
 f(h, \theta) &= V\sigma(Wh + b)
\end{align*}

⚠️ ResNet ≠ RNN
Agenda

Learning with ResNets

Scaling deep ResNets

Scaling in the continuous-time setting

Beyond initialization
Stability at initialization

Original ResNet:

\[
\begin{align*}
    h_0 &= Ax \\
    h_{k+1} &= h_k + V_{k+1} \text{ReLU}(W_{k+1}h_k) \\
    F_\pi(x) &= Bh_L.
\end{align*}
\]
Stability at initialization

Original ResNet:

\[ h_0 = Ax \]
\[ h_{k+1} = h_k + V_{k+1} \text{ReLU}(W_{k+1}h_k) \]
\[ F_\pi(x) = Bh_L. \]

At initialization: \( A, B, (V_k)_{1 \leq k \leq L}, \) and \( (W_k)_{1 \leq k \leq L} \) are i.i.d. Gaussian matrices.
Stability at initialization

▷ **Original ResNet:**

\[
\begin{align*}
  h_0 &= Ax \\
  h_{k+1} &= h_k + V_{k+1} \text{ReLU}(W_{k+1}h_k)
\end{align*}
\]

\[
F_\pi(x) = Bh_L.
\]

▷ **At initialization:** \( A, B, (V_k)_{1 \leq k \leq L}, \text{ and } (W_k)_{1 \leq k \leq L} \) are i.i.d. Gaussian matrices.

![Graph showing \( \frac{\|h_L\|}{\|h_0\|} \) vs. \( L \)]
Stability at initialization

▷ **Original** ResNet:

\[
h_0 = Ax \\
h_{k+1} = h_k + V_{k+1} \text{ReLU}(W_{k+1}h_k) \\
F_\pi(x) = Bh_L.
\]

▷ **At initialization**: \( A, B, (V_k)_{1 \leq k \leq L}, \) and \((W_k)_{1 \leq k \leq L}\) are i.i.d. Gaussian matrices.

▷ **Solution**: batch normalization or scaling.
A scaling factor $\frac{1}{L^\beta}$:

$$h_{k+1} = h_k + \frac{1}{L^\beta} V_{k+1} \text{ReLU}(W_{k+1} h_k).$$
Scaling ResNets

- A scaling factor $\frac{1}{L^\beta}$:

  $$h_{k+1} = h_k + \frac{1}{L^\beta} V_{k+1} \text{ReLU}(W_{k+1} h_k).$$

- **Question**: choice of $\beta$. 
Scaling ResNets

- A scaling factor $\frac{1}{L^\beta}$:

  $$h_{k+1} = h_k + \frac{1}{L^\beta} V_{k+1} \text{ReLU}(W_{k+1} h_k).$$

- **Question**: choice of $\beta$.

- $\beta = 0$ (original ResNets)?
Scaling ResNets

A scaling factor $1/L^\beta$:

$$h_{k+1} = h_k + \frac{1}{L^\beta} V_{k+1} \text{ReLU}(W_{k+1} h_k).$$

**Question**: choice of $\beta$.

- $\beta = 0$ (original ResNets)? $\beta = 1$ (neural ODE)?
Scaling ResNets

A scaling factor $\frac{1}{L^\beta}$:

$$h_{k+1} = h_k + \frac{1}{L^\beta} V_{k+1} \text{ReLU}(W_{k+1} h_k).$$

Question: choice of $\beta$.

$\beta = 0$ (original ResNets)? $\beta = 1$ (neural ODE)?

Many empirical studies, no consensus.
Scaling ResNets

A scaling factor $\frac{1}{L^\beta}$:

$$h_{k+1} = h_k + \frac{1}{L^\beta} V_{k+1} \text{ReLU}(W_{k+1} h_k).$$

Question: choice of $\beta$.

$\beta = 0$ (original ResNets)? $\beta = 1$ (neural ODE)?

Many empirical studies, no consensus.

Our approach: mathematical analysis at initialization.
Scaling with standard initialization

\( \frac{\| h_L - h_0 \|}{\| h_0 \|}, \beta = 1 \) (a)

\( \frac{\| h_L - h_0 \|}{\| h_0 \|}, \beta = 0.25 \) (b)

\( \frac{\| h_L - h_0 \|}{\| h_0 \|}, \beta = 0.5 \) (c)

With an i.i.d. initialization, the critical value for scaling is \( \beta = \frac{1}{2} \).

Similar results (identity/explosion/stability) for the gradients.

Not the ODE scaling!
Scaling with standard initialization

\[(a) \| h_L - h_0 \| / \| h_0 \|, \beta = 1\]

\[(b) \| h_L - h_0 \| / \| h_0 \|, \beta = 0.25\]

\[(c) \| h_L - h_0 \| / \| h_0 \|, \beta = 0.5\]

With an i.i.d. initialization, the critical value for scaling is \( \beta = \frac{1}{2} \).
Scaling with standard initialization

(a) $\|h_L - h_0\|/\|h_0\|$, $\beta = 1$

(b) $\|h_L - h_0\|/\|h_0\|$, $\beta = 0.25$

(c) $\|h_L - h_0\|/\|h_0\|$, $\beta = 0.5$

- With an i.i.d. initialization, the critical value for scaling is $\beta = 1/2$.
- Similar results (identity/explosion/stability) for the gradients.
Scaling with standard initialization

(a) $\|h_L - h_0\| / \|h_0\|, \beta = 1$

(b) $\|h_L - h_0\| / \|h_0\|, \beta = 0.25$

(c) $\|h_L - h_0\| / \|h_0\|, \beta = 0.5$

- With an i.i.d. initialization, the critical value for scaling is $\beta = 1/2$.
- Similar results (identity/explosion/stability) for the gradients.
- Not the ODE scaling! 😐
Theorem

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.
Scaling with standard initialization

Theorem

**Assumption**: the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$
2. If $\beta < \frac{1}{2}$
3. If $\beta = \frac{1}{2}$
Scaling with standard initialization

**Theorem**

**Assumption:** the entries of $\sqrt{d}V_k$ and $\sqrt{d}W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\|h_L - h_0\|/\|h_0\| \xrightarrow{\mathbb{P}} 0$.

2. If $\beta < \frac{1}{2}$

3. If $\beta = \frac{1}{2}$
Scaling with standard initialization

**Theorem**

**Assumption:** the entries of $\sqrt{d}V_k$ and $\sqrt{d}W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\frac{\|h_L - h_0\|}{\|h_0\|} \overset{\mathbb{P}}{\rightarrow} 0$. → identity

2. If $\beta < \frac{1}{2}$

3. If $\beta = \frac{1}{2}$
Scaling with standard initialization

**Theorem**

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\frac{\| h_L - h_0 \| / \| h_0 \|}{P_{L \to \infty}} \to 0$. → identity
2. If $\beta < \frac{1}{2}$ then $\frac{\| h_L - h_0 \| / \| h_0 \|}{P_{L \to \infty}} \to \infty$.
3. If $\beta = \frac{1}{2}$
Theorem

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\|h_L - h_0\|/\|h_0\| \xrightarrow{\mathbb{P}} 0$. \rightarrow \text{identity}
2. If $\beta < \frac{1}{2}$ then $\|h_L - h_0\|/\|h_0\| \xrightarrow{\mathbb{P}} \infty$. \rightarrow \text{explosion}
3. If $\beta = \frac{1}{2}$
Theorem

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\|h_L - h_0\|/\|h_0\| \xrightarrow{\mathbb{P}} 0$. → identity

2. If $\beta < \frac{1}{2}$ then $\|h_L - h_0\|/\|h_0\| \xrightarrow{\mathbb{P}} \infty$. → explosion

3. If $\beta = \frac{1}{2}$ then, with probability at least $1 - \delta$,

$$\exp \left( \frac{3}{8} - \sqrt{\frac{22}{d \delta}} \right) - 1 < \frac{\|h_L - h_0\|^2}{\|h_0\|^2} < \exp \left( 1 + \sqrt{\frac{10}{d \delta}} \right) + 1.$$
Scaling with standard initialization

**Theorem**

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > 1/2$ then $\|h_L - h_0\|/\|h_0\| \xrightarrow{P} 0$. → identity

2. If $\beta < 1/2$ then $\|h_L - h_0\|/\|h_0\| \xrightarrow{P} \infty$. → explosion

3. If $\beta = 1/2$ then, with probability at least $1 - \delta$,

\[
\exp \left( \frac{3}{8} - \sqrt{\frac{22}{d\delta}} \right) - 1 < \frac{\|h_L - h_0\|^2}{\|h_0\|^2} < \exp \left( 1 + \sqrt{\frac{10}{d\delta}} \right) + 1. \quad \rightarrow \text{stability}
\]
Gradients

Objective: assess the backwards dynamics of the gradients $p_k = \frac{\partial L_n}{\partial h_k} \in \mathbb{R}^d$. 

Target: $\|p_0 - p_L\| / \|p_L\|$ when $L$ is large.

Backpropagation formula:

$$p_k = p_{k+1} + L \beta \partial g(h_k, \theta_{k+1})^\top \partial h V_{k+1} p_{k+1} \rightarrow \text{wrong way}.$$ 

Our approach: with $q_k(z) = \partial h_k \partial h_0 z$, $q_{k+1}(z) = q_k(z) + L \beta V_{k+1} \partial g(h_k, \theta_{k+1}) \partial h q_k(z) \rightarrow \text{flow of information} = \text{!}.$

Conclusion with $\|p_0\|_2 \|p_L\|_2 = E_{z \sim N(0, I_d)} (\|p_L\|_2)$. 

$\|$
Gradients

- **Objective:** assess the backwards dynamics of the gradients $p_k = \frac{\partial L}{\partial h_k} \in \mathbb{R}^d$.
- **Target:** $\|p_0 - p_L\| / \|p_L\|$ when $L$ is large.
Gradients

- **Objective**: assess the backwards dynamics of the gradients \( p_k = \frac{\partial L_n}{\partial h_k} \in \mathbb{R}^d \).

- **Target**: \( \| p_0 - p_L \| / \| p_L \| \) when \( L \) is large.

- **Backpropagation formula**:

\[
p_k = p_{k+1} + \frac{1}{L\beta} \frac{\partial g(h_k, \theta_{k+1})^\top}{\partial h} \left( V_{k+1}^\top p_{k+1} \right)
\]
Objective: assess the backwards dynamics of the gradients $p_k = \frac{\partial \mathcal{L}_n}{\partial h_k} \in \mathbb{R}^d$.

Target: $\frac{\|p_0 - p_L\|}{\|p_L\|}$ when $L$ is large.

Backpropagation formula:

$$p_k = p_{k+1} + \frac{1}{L\beta} \frac{\partial g(h_k, \theta_{k+1})}{\partial h}^\top V_{k+1}^\top p_{k+1} \quad \rightarrow \text{wrong way}.$$
Objective: assess the backwards dynamics of the gradients $p_k = \frac{\partial L_n}{\partial h_k} \in \mathbb{R}^d$.

Target: $\|p_0 - p_L\|/\|p_L\|$ when $L$ is large.

Backpropagation formula:

$$p_k = p_{k+1} + \frac{1}{L\beta} \frac{\partial g(h_k, \theta_{k+1})}{\partial h} V_{k+1} p_{k+1} \quad \rightarrow \text{wrong way}.$$ 

Our approach: with $q_k(z) = \frac{\partial h_k}{\partial h_0} z$,

$$q_{k+1}(z) = q_k(z) + \frac{1}{L\beta} V_{k+1} \frac{\partial g(h_k, \theta_{k+1})}{\partial h} q_k(z)$$
Gradients

- **Objective:** assess the backwards dynamics of the gradients \( p_k = \frac{\partial L_n}{\partial h_k} \in \mathbb{R}^d \).
- **Target:** \( \|p_0 - p_L\|/\|p_L\| \) when \( L \) is large.
- **Backpropagation formula:**
  \[
p_k = p_{k+1} + \frac{1}{L \beta} \left( \frac{\partial g(h_k, \theta_{k+1})}{\partial h} \right)^\top V_{k+1}^\top p_{k+1} \quad \rightarrow \text{wrong way.}
  \]
- **Our approach:** with \( q_k(z) = \frac{\partial h_k}{\partial h_0} z \),
  \[
  q_{k+1}(z) = q_k(z) + \frac{1}{L \beta} V_{k+1} \left( \frac{\partial g(h_k, \theta_{k+1})}{\partial h} \right) q_k(z) \quad \rightarrow \text{flow of information = } \checkmark.
  \]
Gradients

- **Objective:** assess the backwards dynamics of the gradients $p_k = \frac{\partial L_n}{\partial h_k} \in \mathbb{R}^d$.

- **Target:** $\|p_0 - p_L\|/\|p_L\|$ when $L$ is large.

- **Backpropagation formula:**

  $$p_k = p_{k+1} + \frac{1}{L^\beta} \left( \frac{\partial g(h_k, \theta_{k+1})}{\partial h} \right) ^\top V_{k+1} p_{k+1} \rightarrow \text{wrong way}.$$  

- **Our approach:** with $q_k(z) = \frac{\partial h_k}{\partial h_0} z$,

  $$q_{k+1}(z) = q_k(z) + \frac{1}{L^\beta} V_{k+1} \left( \frac{\partial g(h_k, \theta_{k+1})}{\partial h} \right) q_k(z) \rightarrow \text{flow of information} = \checkmark.$$  

- **Conclusion with**

  $$\frac{\|p_0\|^2}{\|p_L\|^2} = E_{z \sim \mathcal{N}(0,I_d)} \left( \left( \frac{p_L}{\|p_L\|} \right)^\top q_L(z) \right)^2.$$  

Scaling with standard initialization – Gradients

(a) $\|p_0 - p_L\|/\|p_L\|$, $\beta = 1$

(b) $\|p_0 - p_L\|/\|p_L\|$, $\beta = 0.25$

(c) $\|p_0 - p_L\|/\|p_L\|$, $\beta = 0.5$
Scaling with standard initialization – Gradients

**Theorem**

**Assumption**: the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.
Theorem

**Assumption:** the entries of $\sqrt{d}V_k$ and $\sqrt{d}W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$
2. If $\beta < \frac{1}{2}$
3. If $\beta = \frac{1}{2}$
Theorem

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\frac{\|p_0 - p_L\|}{\|p_L\|} \xrightarrow{\text{P}} 0$.

2. If $\beta < \frac{1}{2}$

3. If $\beta = \frac{1}{2}$
Scaling with standard initialization – Gradients

**Theorem**

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > 1/2$ then $\|p_0 - p_L\|/\|p_L\| \xrightarrow{P} 0$. \quad \rightarrow \text{identity}

2. If $\beta < 1/2$

3. If $\beta = 1/2$
Scaling with standard initialization – **Gradients**

**Theorem**

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\frac{\|p_0 - p_L\|}{\|p_L\|} \xrightarrow{P} 0$. $\rightarrow$ identity

2. If $\beta < \frac{1}{2}$ then $\mathbb{E}(\frac{\|p_0 - p_L\|}{\|p_L\|}) \xrightarrow{P} \infty$.

3. If $\beta = \frac{1}{2}$
Theorem

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > \frac{1}{2}$ then $\|p_0 - p_L\|/\|p_L\| \xrightarrow{P} 0$. \(\rightarrow\) identity

2. If $\beta < \frac{1}{2}$ then $\mathbb{E}(\|p_0 - p_L\|/\|p_L\|) \xrightarrow{P} \infty$. \(\rightarrow\) explosion

3. If $\beta = \frac{1}{2}$
Theorem

Assumption: the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > 1/2$ then $\|p_0 - p_L\|/\|p_L\| \xrightarrow{L \to \infty} 0$. → identity

2. If $\beta < 1/2$ then $\mathbb{E}(\|p_0 - p_L\|/\|p_L\|) \xrightarrow{L \to \infty} \infty$. → explosion

3. If $\beta = 1/2$ then

\[
\exp\left(\frac{1}{2}\right) - 1 \leq \mathbb{E}\left(\frac{\|p_0 - p_L\|^2}{\|p_L\|^2}\right) \leq \exp(4) - 1.
\]
Scaling with standard initialization – Gradients

**Theorem**

**Assumption:** the entries of $\sqrt{d} V_k$ and $\sqrt{d} W_k$ are symmetric i.i.d. sub-Gaussian.

1. If $\beta > 1/2$ then $\|p_0 - p_L\| / \|p_L\| \xrightarrow{P \text{-} \text{a.s.}} 0$. \[ \rightarrow \text{identity} \]

2. If $\beta < 1/2$ then $\mathbb{E}(\|p_0 - p_L\| / \|p_L\|) \xrightarrow{L \rightarrow \infty} \infty$. \[ \rightarrow \text{explosion} \]

3. If $\beta = 1/2$ then

$$\exp \left( \frac{1}{2} \right) - 1 \leq \mathbb{E} \left( \frac{\|p_0 - p_L\|^2}{\|p_L\|^2} \right) \leq \exp(4) - 1. \rightarrow \text{stability}$$
Stability – output/Gradients

(a) Distribution of $\frac{\|h_L\|}{\|h_0\|}$

(b) Distribution of $\frac{\|\frac{\partial L}{\partial h_0}\|}{\|\frac{\partial L}{\partial h_L}\|}$
How to interpret the critical value $\beta = 1/2$?

\[ h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k). \]
How to interpret the critical value $\beta = \frac{1}{2}$?

- Simple ResNet: $h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k)$.

- The entries of $V_k$ are i.i.d. $\mathcal{N}(0, \frac{1}{d})$. 
How to interpret the critical value $\beta = \frac{1}{2}$?

- **Simple ResNet:** $h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k)$.
- The entries of $V_k$ are i.i.d. $\mathcal{N}(0, 1/d)$.
- For $B : [0, 1] \rightarrow \mathbb{R}^{d \times d}$ a $(d \times d)$-dimensional Brownian motion.
How to interpret the critical value $\beta = \frac{1}{2}$?

- **Simple ResNet**: $h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k)$.

- The entries of $V_k$ are i.i.d. $\mathcal{N}(0, 1/d)$.

- For $B : [0, 1] \to \mathbb{R}^{d \times d}$ a $(d \times d)$-dimensional Brownian motion

$$B_{(k+1)/L, i, j} - B_{k/L, i, j} \sim \mathcal{N}(0, \frac{1}{L}).$$
How to interpret the critical value $\beta = 1/2$?

- **Simple ResNet:** $h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k)$.

- The entries of $V_k$ are i.i.d. $\mathcal{N}(0, 1/d)$.

- For $B : [0, 1] \to \mathbb{R}^{d \times d}$ a $(d \times d)$-dimensional Brownian motion

  $$B_{(k+1)/L, i, j} - B_{k/L, i, j} \sim \mathcal{N}(0, \frac{1}{L}).$$

- Consequence:

  $$h_0 = Ax, \quad h_{k+1} = h_k + \frac{1}{\sqrt{d}} \sigma(h_k^T)(B_{(k+1)/L} - B_{k/L}), \quad 0 \leq k \leq L - 1.$$
### SDE regime

<table>
<thead>
<tr>
<th><strong>ResNet</strong></th>
<th><strong>Neural SDE</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_0 = Ax$</td>
<td>$H_0 = Ax$</td>
</tr>
<tr>
<td>$h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k)$</td>
<td>$dH_t^\top = \frac{1}{\sqrt{d}} \sigma(H_t^\top) dB_t$</td>
</tr>
<tr>
<td>$F_\pi(x) = Bh_L$</td>
<td>$F_{\Pi}(x) = BH_1$</td>
</tr>
</tbody>
</table>

- Proposition
- Assumption: the entries of $V_k$ are i.i.d. Gaussian $N(0, 1/d)$ and $\sigma$ is Lipschitz continuous.
- Then the SDE has a unique solution $H$ and, for any $0 \leq k \leq L$, $E(\|H_k^L - h_k\|) \leq C \sqrt{L}$. 


SDE regime

**ResNet**

\[ h_0 = Ax \]

\[ h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k) \]

\[ F_\pi(x) = B h_L \]

**Neural SDE**

\[ H_0 = Ax \]

\[ dH_t^\top = \frac{1}{\sqrt{d}} \sigma(H_t^\top) dB_t \]

\[ F_\Pi(x) = B H_1 \]

**Proposition**

**Assumption**: the entries of \( V_k \) are i.i.d. Gaussian \( \mathcal{N}(0, \frac{1}{d}) \) and \( \sigma \) is Lipschitz continuous.
**SDE regime**

<table>
<thead>
<tr>
<th>ResNet</th>
<th>Neural SDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>( h_0 = Ax )</td>
<td>( H_0 = Ax )</td>
</tr>
<tr>
<td>( h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1} \sigma(h_k) )</td>
<td>( dH_t^\top = \frac{1}{\sqrt{d}} \sigma(H_t^\top) dB_t )</td>
</tr>
<tr>
<td>( F_\pi(x) = Bh_L )</td>
<td>( F_{\Pi}(x) = BH_1 )</td>
</tr>
</tbody>
</table>

**Proposition**

**Assumption**: the entries of \( V_k \) are i.i.d. Gaussian \( \mathcal{N}(0, 1/d) \) and \( \sigma \) is Lipschitz continuous.

Then the SDE has a unique solution \( H \) and, for any \( 0 \leq k \leq L \),

\[
\mathbb{E}(\|H_{k/L} - h_k\|) \leq \frac{C}{\sqrt{L}}.
\]
Summary so far

For deep ResNets with i.i.d. initialization:
For deep ResNets with i.i.d. initialization:

- the critical value for scaling is $\beta = \frac{1}{2}$
- this value corresponds in the deep limit to a SDE.
Summary so far

For deep ResNets with i.i.d. initialization:

▷ the critical value for scaling is $\beta = 1/2$
▷ this value corresponds in the deep limit to a SDE.

Remaining questions:

▷ Can we obtain other limits? For example ODEs?
▷ Do they correspond to the same critical value?
Summary so far

For deep ResNets with i.i.d. initialization:

- the critical value for scaling is $\beta = \frac{1}{2}$
- this value corresponds in the deep limit to a SDE.

Remaining questions:

- Can we obtain other limits? For example ODEs?
- Do they correspond to the same critical value?

Key: link between $\beta$ and the weight distributions.
Summary so far

For deep ResNets with i.i.d. initialization:

- the critical value for scaling is $\beta = \frac{1}{2}$
- this value corresponds in the deep limit to a SDE.

Remaining questions:

- Can we obtain other limits? For example ODEs?
- Do they correspond to the same critical value?

Key: link between $\beta$ and the weight distributions.
Agenda

Learning with ResNets

Scaling deep ResNets

Scaling in the continuous-time setting

Beyond initialization
Leaving the i.i.d. world behind

Idea: the weights \((V_k)_{1 \leq k \leq L}\) and \((\theta_k)_{1 \leq k \leq L}\) are discretizations of smooth functions.

Model:

\[
h_0 = Ax, \quad h_k + 1 = h_k + 1 L V_k + 1 g(h_k, \theta_k + 1),
\]

where \(V_k = V_k / L\) and \(\theta_k = \Theta_k / L\).

Assumption: the stochastic processes \(V\) and \(\Theta\) are a.s. Lipschitz continuous and bounded.

Example: the entries of \(V\) and \(\Theta\) are independent Gaussian processes with zero expectation and covariance \(K(x, x') = \exp(-\frac{(x - x')^2}{\ell^2})\).
Leaving the i.i.d. world behind

- Idea: the weights \((V_k)_{1 \leq k \leq L}\) and \((\theta_k)_{1 \leq k \leq L}\) are discretizations of smooth functions.

\[(V_k)_{1 \leq k \leq L} \hookrightarrow \mathcal{V} : [0, 1] \rightarrow \mathbb{R}^{d \times d}\]
Leaving the i.i.d. world behind

- Idea: the weights $\left( V_k \right)_{1 \leq k \leq L}$ and $\left( \theta_k \right)_{1 \leq k \leq L}$ are discretizations of smooth functions.

- $\left( V_k \right)_{1 \leq k \leq L} \hookrightarrow \mathcal{V} : [0, 1] \rightarrow \mathbb{R}^{d \times d}$ and $\left( \theta_k \right)_{1 \leq k \leq L} \hookrightarrow \Theta : [0, 1] \rightarrow \mathbb{R}^p$. 

Assumption: the stochastic processes $V$ and $\Theta$ are a.s. Lipschitz continuous and bounded.

Example: the entries of $V$ and $\Theta$ are independent Gaussian processes with zero expectation and covariance $K(x, x') = \exp(-\frac{(x-x')^2}{\ell^2})$. 

Leaving the i.i.d. world behind

- Idea: the weights \((V_k)_{1 \leq k \leq L}\) and \((\theta_k)_{1 \leq k \leq L}\) are discretizations of smooth functions.

- \((V_k)_{1 \leq k \leq L} \hookrightarrow \mathcal{V} : [0, 1] \to \mathbb{R}^{d \times d}\) and \((\theta_k)_{1 \leq k \leq L} \hookrightarrow \Theta : [0, 1] \to \mathbb{R}^p\).

- Model:
  \[h_0 = Ax, \quad h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}), \quad 0 \leq k \leq L - 1,\]

  Assumption: the stochastic processes \(V\) and \(\Theta\) are a.s. Lipschitz continuous and bounded.

- Example: the entries of \(V\) and \(\Theta\) are independent Gaussian processes with zero expectation and covariance \(K(x, x') = \exp\left(-\frac{(x - x')^2}{\ell^2}\right)\).
Leaving the i.i.d. world behind

- Idea: the weights \(( V_k)_{1 \leq k \leq L}\) and \((\theta_k)_{1 \leq k \leq L}\) are discretizations of smooth functions.

\[(V_k)_{1 \leq k \leq L} \hookrightarrow \mathcal{V} : [0, 1] \rightarrow \mathbb{R}^{d \times d}, \quad (\theta_k)_{1 \leq k \leq L} \hookrightarrow \Theta : [0, 1] \rightarrow \mathbb{R}^p.\]

- Model:

\[
h_0 = Ax, \quad h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}), \quad 0 \leq k \leq L - 1,
\]

where \(V_k = \mathcal{V}_{k/L}\) and \(\theta_k = \Theta_{k/L}\).
Leaving the i.i.d. world behind

- **Idea:** the weights \(( V_k )_{1 \leq k \leq L}\) and \(( \theta_k )_{1 \leq k \leq L}\) are discretizations of smooth functions.

- \(( V_k )_{1 \leq k \leq L} \hookrightarrow \mathcal{V} : [0, 1] \rightarrow \mathbb{R}^{d \times d}\) \(( \theta_k )_{1 \leq k \leq L} \hookrightarrow \Theta : [0, 1] \rightarrow \mathbb{R}^p\).

- **Model:**

\[
    h_0 = Ax, \quad h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}), \quad 0 \leq k \leq L - 1,
\]

where \( V_k = \mathcal{V}_{k/L} \) and \( \theta_k = \Theta_{k/L} \).

**Assumption:** the stochastic processes \( \mathcal{V} \) and \( \Theta \) are a.s. Lipschitz continuous and bounded.
Leaving the i.i.d. world behind

- **Idea:** the weights \((V_k)_{1 \leq k \leq L}\) and \((\theta_k)_{1 \leq k \leq L}\) are discretizations of smooth functions.

\[
(V_k)_{1 \leq k \leq L} \hookrightarrow \mathcal{V} : [0, 1] \rightarrow \mathbb{R}^{d \times d}, \quad (\theta_k)_{1 \leq k \leq L} \hookrightarrow \Theta : [0, 1] \rightarrow \mathbb{R}^{p}.
\]

- **Model:**

\[
h_0 = Ax, \quad h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}), \quad 0 \leq k \leq L - 1,
\]

where \(V_k = \mathcal{V}_{k/L}\) and \(\theta_k = \Theta_{k/L}\).

**Assumption:** the stochastic processes \(\mathcal{V}\) and \(\Theta\) are a.s. Lipschitz continuous and bounded.

- **Example:** the entries of \(\mathcal{V}\) and \(\Theta\) are independent Gaussian processes with zero expectation and covariance \(K(x, x') = \exp(-\frac{(x-x')^2}{2\ell^2})\).
Scaling and weight regularity

(a) i.i.d.
Scaling and weight regularity

(a) i.i.d.

(b) Smooth
ODE regime

**ResNet**

\[ h_0 = Ax \]
\[ h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}) \]
\[ F_\pi(x) = B h_L \]

**Neural ODE**

\[ H_0 = Ax \]
\[ dH_t = \mathcal{V}_t g(H_t, \Theta_t) dt \]
\[ F_\Pi(x) = B H_1 \]
### ResNet

\[ h_0 = Ax \]
\[ h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}) \]
\[ F_\pi(x) = B h_L \]

### Neural ODE

\[ H_0 = Ax \]
\[ dH_t = \mathcal{V}_t g(H_t, \Theta_t) \, dt \]
\[ F_\Pi(x) = B H_1 \]

---

**Proposition**

**Assumption**: the function \( g \) is **Lipschitz continuous** on compact sets.
### ODE regime

<table>
<thead>
<tr>
<th>ResNet</th>
<th>Neural ODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>( h_0 = Ax )</td>
<td>( H_0 = Ax )</td>
</tr>
<tr>
<td>( h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}) )</td>
<td>( dH_t = \mathcal{V}_t g(H_t, \Theta_t) dt )</td>
</tr>
<tr>
<td>( F_\pi(x) = Bh_L )</td>
<td>( F_\Pi(x) = BH_1 )</td>
</tr>
</tbody>
</table>

### Proposition

**Assumption**: the function \( g \) is **Lipschitz continuous** on compact sets.

Then the ODE has a **unique solution** \( H \) and, a.s., for any \( 0 \leq k \leq L \),

\[
\| H_{k/L} - h_k \| \leq \frac{c}{L}.
\]
Scaling with a smooth initialization

\( \frac{\| h_L - h_0 \|}{\| h_0 \|}, \beta = 2 \)  
\( \frac{\| h_L - h_0 \|}{\| h_0 \|}, \beta = 0.5 \)  
\( \frac{\| h_L - h_0 \|}{\| h_0 \|}, \beta = 1 \)

Again 3 cases: identity/explosion/stability. With a smooth initialization, the critical scaling is \( \beta = 1 \). It is the scaling that corresponds in the deep limit to an ODE.
Scaling with a smooth initialization

Again 3 cases: identity/explosion/stability.
Scaling with a smooth initialization

> Again 3 cases: identity/explosion/stability.
> With a smooth initialization, the critical scaling is \( \beta = 1 \).
Scaling with a smooth initialization

\[
\|h_L - h_0\|/\|h_0\|, \beta = 2
\]

\[
\|h_L - h_0\|/\|h_0\|, \beta = 0.5
\]

\[
\|h_L - h_0\|/\|h_0\|, \beta = 1
\]

➤ Again 3 cases: identity/explosion/stability.
➤ With a smooth initialization, the critical scaling is $\beta = 1$.
➤ It is the scaling that corresponds in the deep limit to an ODE.
Scaling with a smooth initialization

**Theorem**

**Assumption:** $\mathcal{W}$ and $\Theta$ are a.s. Lipschitz continuous and bounded.
Scaling with a smooth initialization

**Theorem**

**Assumption:** $\mathcal{W}$ and $\Theta$ are a.s. Lipschitz continuous and bounded.

1. If $\beta > 1$
2. If $\beta = 1$
3. If $\beta < 1$
Scaling with a smooth initialization

**Theorem**

**Assumption:** \( \mathcal{V} \) and \( \Theta \) are a.s. Lipschitz continuous and bounded.

1. If \( \beta > 1 \) then, a.s., \( \| h_L - h_0 \| / \| h_0 \| \xrightarrow{L \to \infty} 0 \).  
2. If \( \beta = 1 \)
3. If \( \beta < 1 \)
Scaling with a smooth initialization

**Theorem**

**Assumption:** \( \mathcal{V} \) and \( \Theta \) are a.s. Lipschitz continuous and bounded.

1. If \( \beta > 1 \) then, a.s., \( \| h_L - h_0 \| / \| h_0 \| \xrightarrow{L \to \infty} 0 \). \( \to \) identity

2. If \( \beta = 1 \)

3. If \( \beta < 1 \)
Scaling with a smooth initialization

**Theorem**

**Assumption:** $\mathcal{V}$ and $\Theta$ are a.s. Lipschitz continuous and bounded.

1. If $\beta > 1$ then, a.s., $\|h_L - h_0\|/\|h_0\| \xrightarrow{L \to \infty} 0$. → identity

2. If $\beta = 1$ then, a.s., $\|h_L - h_0\|/\|h_0\| \leq c$.

3. If $\beta < 1$
Scaling with a smooth initialization

**Theorem**

**Assumption:** \( \mathcal{V} \) and \( \Theta \) are a.s. Lipschitz continuous and bounded.

1. If \( \beta > 1 \) then, a.s., \( \| h_L - h_0 \| / \| h_0 \| \xrightarrow{L \to \infty} 0. \)  \( \to \) **identity**

2. If \( \beta = 1 \) then, a.s., \( \| h_L - h_0 \| / \| h_0 \| \leq c. \)  \( \to \) **stability**

3. If \( \beta < 1 \)
Scaling with with a smooth initialization

**Theorem**

**Assumption**: \( \mathcal{W} \) and \( \Theta \) are a.s. Lipschitz continuous and bounded.

1. If \( \beta > 1 \) then, a.s., \( \| h_L - h_0 \| / \| h_0 \| \xrightarrow{L \to \infty} 0 \). \( \rightarrow \) identity

2. If \( \beta = 1 \) then, a.s., \( \| h_L - h_0 \| / \| h_0 \| \leq c \). \( \rightarrow \) stability

3. If \( \beta < 1 \) + assumptions, then \( \max_k \frac{\| h_k - h_0 \|}{\| h_0 \|} \xrightarrow{L \to \infty} \infty \).
Scaling with a smooth initialization

**Theorem**

<table>
<thead>
<tr>
<th>Assumption: ( \mathcal{V} ) and ( \Theta ) are a.s. Lipschitz continuous and bounded.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. If ( \beta &gt; 1 ) then, a.s., ( \frac{|h_L - h_0|}{|h_0|} \xrightarrow{L \to \infty} 0. ) ( \rightarrow ) <strong>identity</strong></td>
</tr>
<tr>
<td>2. If ( \beta = 1 ) then, a.s., ( \frac{|h_L - h_0|}{|h_0|} \leq c. ) ( \rightarrow ) <strong>stability</strong></td>
</tr>
<tr>
<td>3. If ( \beta &lt; 1 ) + assumptions, then ( \max_k \frac{|h_k - h_0|}{|h_0|} \xrightarrow{L \to \infty} \infty. ) ( \rightarrow ) <strong>explosion</strong></td>
</tr>
</tbody>
</table>
Intermediate regimes

- **Challenge:** describe the transition between the i.i.d. and smooth cases.

- We initialize the weights as increments of a fractional Brownian motion \( B_H^t \) for \( t \in [0, 1] \).

- Recall: \( B_H^t \) is Gaussian, starts at zero, has zero expectation, and covariance function
  \[
  E(B_H^s B_H^t) = \frac{1}{2} (|s|^{2H} + |t|^{2H} - |t-s|^{2H}),
  \]
  for \( 0 \leq s, t \leq 1 \).

- The Hurst index \( H \in (0, 1) \) describes the raggedness of the process.
Challenge: describe the transition between the i.i.d. and smooth cases.

We initialize the weights as increments of a fractional Brownian motion \( (B_t^H)_{t \in [0,1]} \).
Intermediate regimes

- **Challenge**: describe the transition between the i.i.d. and smooth cases.
- We initialize the weights as increments of a fractional Brownian motion \((B^H_t)_{t\in[0,1]}\).
- **Recall**: \(B^H\) is Gaussian, starts at zero, has zero expectation, and covariance function

\[
\mathbb{E}(B^H_s B^H_t) = \frac{1}{2} (|s|^{2H} + |t|^{2H} - |t - s|^{2H}), \quad 0 \leq s, t \leq 1.
\]
Intermediate regimes

- **Challenge**: describe the transition between the i.i.d. and smooth cases.
- We initialize the weights as increments of a fractional Brownian motion \((B^H_t)_{t \in [0,1]}\).
- **Recall**: \(B^H\) is Gaussian, starts at zero, has zero expectation, and covariance function
  \[
  \mathbb{E}(B^H_s B^H_t) = \frac{1}{2} (|s|^{2H} + |t|^{2H} - |t - s|^{2H}), \quad 0 \leq s, t \leq 1.
  \]
- The **Hurst index** \(H \in (0, 1)\) describes the raggedness of the process.
(a) $H = 0.2$

(b) $H = 0.5$

(c) $H = 0.8$

$\Delta H = \frac{1}{2}$: standard Brownian motion (SDE regime).

$\Delta H < \frac{1}{2}$: the increments are negatively correlated.

$\Delta H > \frac{1}{2}$: the increments are positively correlated.

$\Delta$ When $H \to 1$: the trajectories converge to linear functions (ODE regime).
\( H = 0.2 \) (a)

\( H = 0.5 \) (b)

\( H = 0.8 \) (c)

\( H = \frac{1}{2} \): standard Brownian motion (SDE regime).
\( H = 1/2 \): standard Brownian motion (SDE regime).

\( H < 1/2 \): the increments are negatively correlated.

\( H > 1/2 \): the increments are positively correlated.
(a) $H = 0.2$
(b) $H = 0.5$
(c) $H = 0.8$

- $H = \frac{1}{2}$: standard Brownian motion (SDE regime).
- $H < \frac{1}{2}$: the increments are negatively correlated.
- $H > \frac{1}{2}$: the increments are positively correlated.
- When $H \to 1$: the trajectories converge to linear functions (ODE regime).
A continuum of intermediate regularities
A continuum of intermediate regularities
Learning with ResNets

Scaling deep ResNets

Scaling in the continuous-time setting

Beyond initialization
Training

Before training

After training

I.i.d. initialization, $\beta = \frac{1}{2}$
Training

Before training

After training

I.i.d. initialization, $\beta = 1$
Smooth initialization, $\beta = 1$
Smooth initialization, $\beta = 1$

The weights after training still exhibit a strong structure as functions of the layer.
The weights after training still exhibit a strong structure as functions of the layer.
Their regularity is influenced by both the initialization and the choice of $\beta$. 

Smooth initialization, $\beta = 1$
Performance after training

(a) On MNIST

(b) On CIFAR-10
Conclusion

Deep limits allow to understand scaling and initialization strategies for ResNets.
Deep limits allow to understand scaling and initialization strategies for ResNets.

With standard initialization the correct scaling is $\beta = \frac{1}{2}$. 

Perspectives: what about training? how should we choose the regularity for a given problem?

To know more: arXiv:2206.06929.
Deep limits allow to understand scaling and initialization strategies for ResNets.

With standard initialization the correct scaling is $\beta = 1/2$.

To train very deep ResNets, it is important to adapt scaling to the weight regularity.
Conclusion

- Deep limits allow to understand scaling and initialization strategies for ResNets.
- With standard initialization the correct scaling is $\beta = \frac{1}{2}$.
- To train very deep ResNets, it is important to adapt scaling to the weight regularity.
- Perspectives: what about training? how should we choose the regularity for a given problem?

To know more: arXiv:2206.06929.
Conclusion

- Deep limits allow to understand scaling and initialization strategies for ResNets.
- With standard initialization the correct scaling is $\beta = \frac{1}{2}$.
- To train very deep ResNets, it is important to adapt scaling to the weight regularity.
- Perspectives: what about training? how should we choose the regularity for a given problem?
- To know more: arXiv:2206.06929.
Thank you!

- gerard.biau@sorbonne-universite.fr
- perso.lpsm.paris/∼biau