Global convergence of gradient descent for non-convex learning problems

Francis Bach

INRIA - Ecole Normale Supérieure, Paris, France

Joint work with Lénaïc Chizat

Institut Henri Poincaré - April 5, 2019

Machine learning
Scientific context

- **Proliferation of digital data**
  - Personal data
  - Industry
  - Scientific: from bioinformatics to humanities

- **Need for automated processing of massive data**

Machine learning
Scientific context

- **Proliferation of digital data**
  - Personal data
  - Industry
  - Scientific: from bioinformatics to humanities
- **Need for automated processing of massive data**
- **Series of “hypes”**

  Big data $\rightarrow$ Data science $\rightarrow$ Machine Learning
  $\rightarrow$ Deep Learning $\rightarrow$ Artificial Intelligence

Machine learning
Scientific context

- **Proliferation of digital data**
  - Personal data
  - Industry
  - Scientific: from bioinformatics to humanities

- **Need for automated processing of massive data**

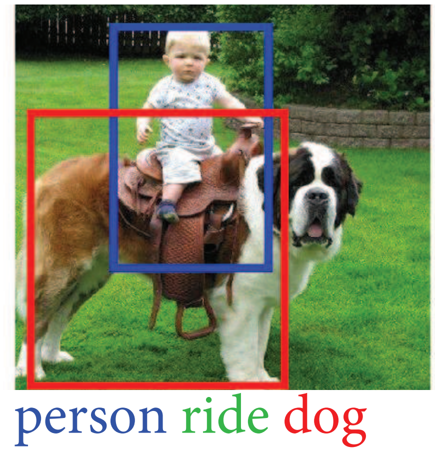
- **Series of "hypes"**

  Big data $\rightarrow$ Data science $\rightarrow$ Machine Learning
  $\rightarrow$ Deep Learning $\rightarrow$ Artificial Intelligence

- **Healthy interactions between theory, applications, and hype?**

Recent progress in perception (vision, audio, text)

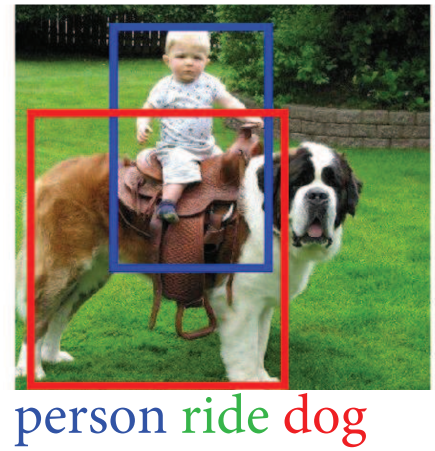
From translate.google.fr



From Peyré et al. (2017)

Recent progress in perception (vision, audio, text)

From translate.google.fr



From Peyré et al. (2017)

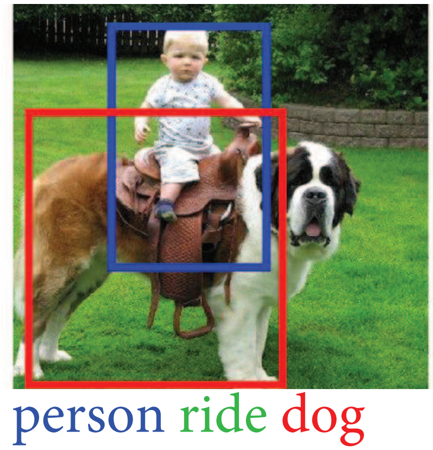
(1) **Massive data**

(2) **Computing power**

(3) **Methodological and scientific progress**

Recent progress in perception (vision, audio, text)

From translate.google.fr



From Peyré et al. (2017)

(1) **Massive data**

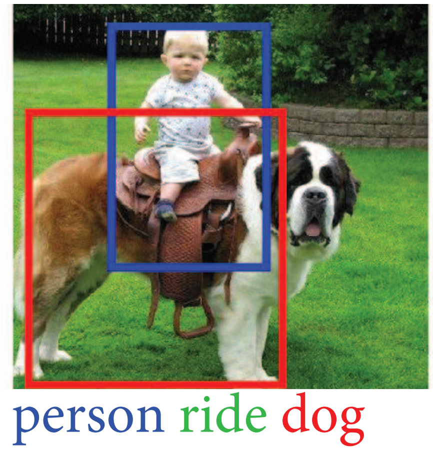
(2) **Computing power**

(3) **Methodological and scientific progress**

"Intelligence" = models + algorithms + data
+ computing power

Recent progress in perception (vision, audio, text)

From translate.google.fr



From Peyré et al. (2017)

(1) **Massive data**

(2) **Computing power**

(3) **Methodological and scientific progress**

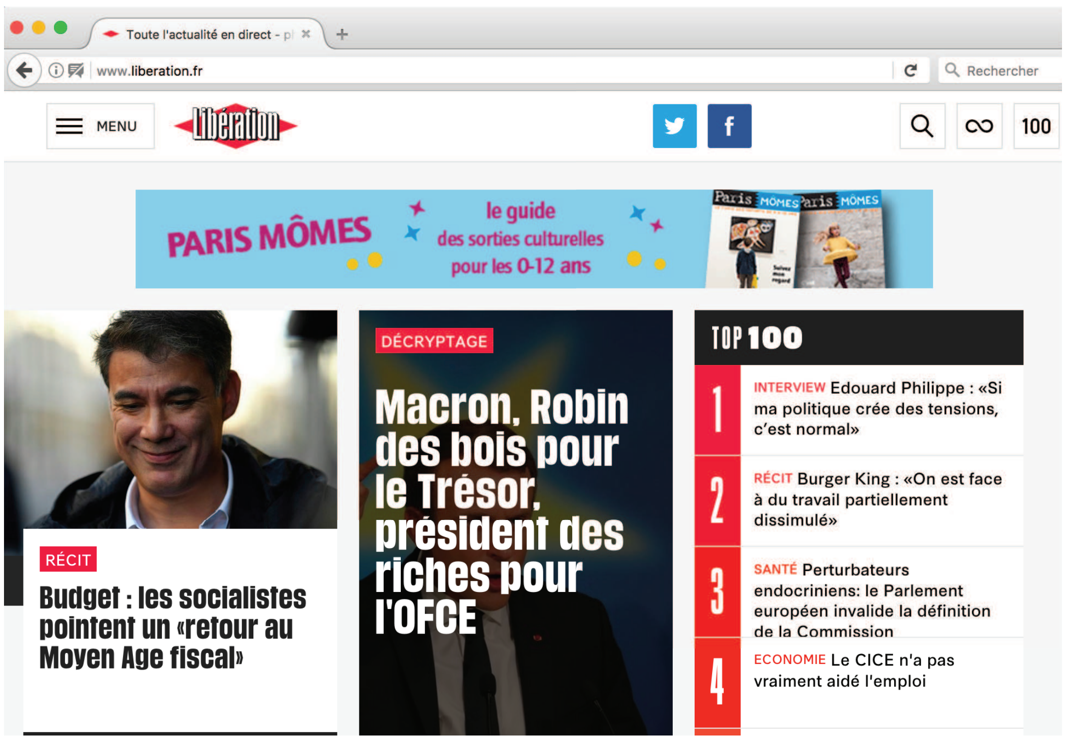
"Intelligence" = models + algorithms + data + computing power

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

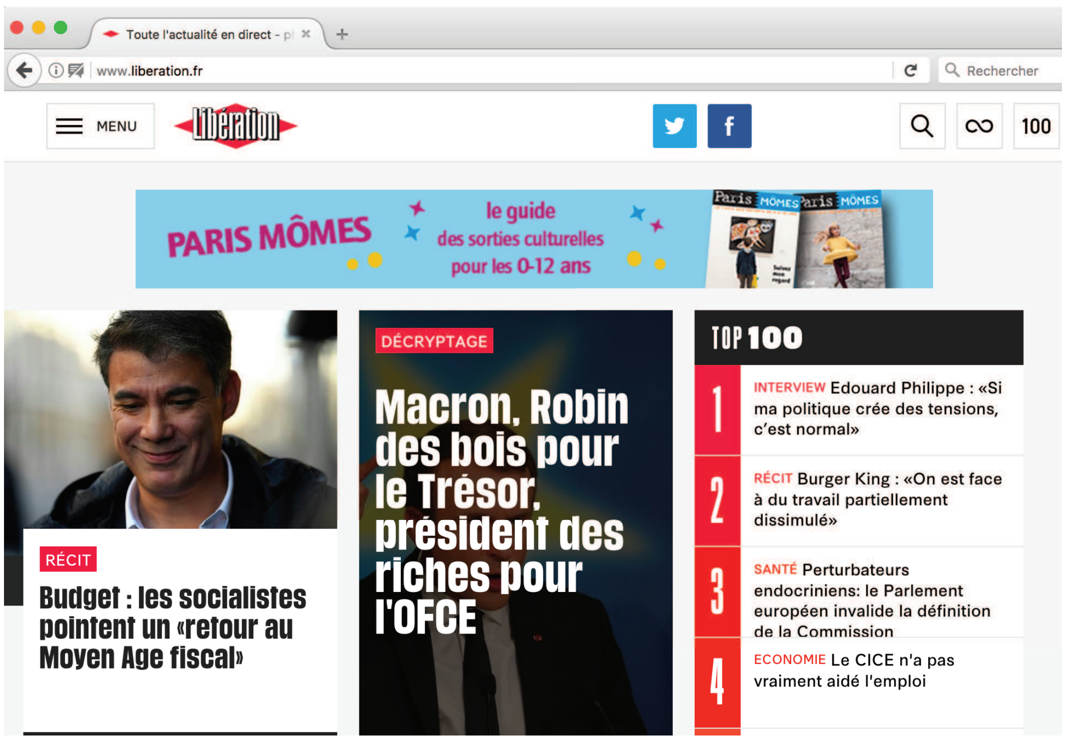


- **Advertising**: $n > 10^9$
  - $\Phi(x) \in \{0, 1\}^d$, $d > 10^9$
  - Navigation history + ad

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$



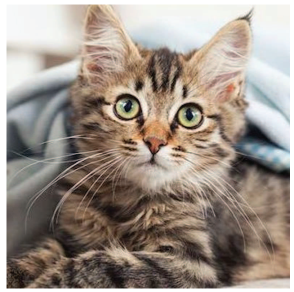
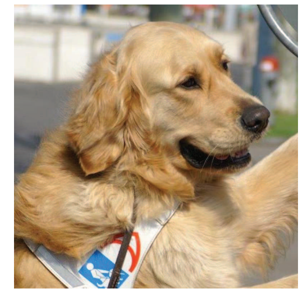
- **Advertising**: $n > 10^9$
  - $\Phi(x) \in \{0, 1\}^d$, $d > 10^9$
  - Navigation history + ad

- Linear predictions
  - $h(x, \theta) = \theta^\top \Phi(x)$

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \ldots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

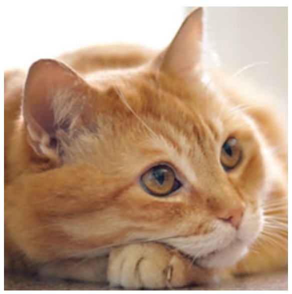
x_1	x_2	x_3	x_4	x_5	x_6
					
$y_1 = 1$	$y_2 = 1$	$y_3 = 1$	$y_4 = -1$	$y_5 = -1$	$y_6 = -1$

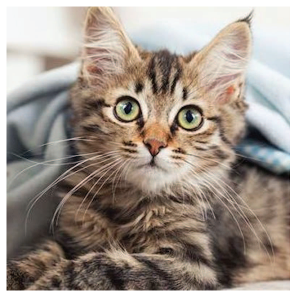
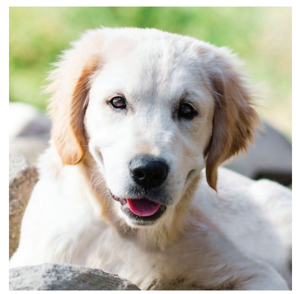
Parametric supervised machine learning

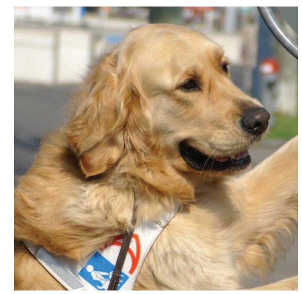
- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

x_1 x_2 x_3 x_4 x_5 x_6

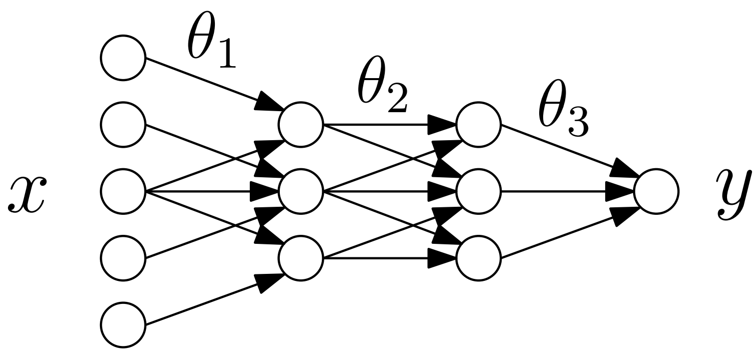




$y_1 = 1$ $y_2 = 1$ $y_3 = 1$ $y_4 = -1$ $y_5 = -1$ $y_6 = -1$

  – Neural networks $(n, d > 10^6)$: $h(x, \theta) = \theta_m^\top \sigma(\theta_{m-1}^\top \sigma(\cdots \theta_2^\top \sigma(\theta_1^\top x))$



Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \quad \ell\big(y_i, h(x_i, \theta)\big) \quad + \quad \lambda \Omega(\theta)$$

data fitting term $+$ regularizer

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \ \ell\big(y_i, h(x_i, \theta)\big) \quad + \quad \lambda \Omega(\theta)$$

data fitting term $+$ regularizer

- **Actual goal**: minimize test error $\mathbb{E}_{p(x,y)} \ell(y, h(x, \theta))$

Convex optimization problems

- **Convexity in machine learning**
  - Convex loss and linear predictions $h(x, \theta) = \theta^\top \Phi(x)$

Convex optimization problems

- **Convexity in machine learning**
  - Convex loss and linear predictions $h(x, \theta) = \theta^\top \Phi(x)$
- **(approximately) Matching theory and practice**
  - Fruitful discussions between theoreticians and practitioners
  - Quantitative theoretical analysis suggests practical improvements

Convex optimization problems

- **Convexity in machine learning**
  - Convex loss and linear predictions $h(x, \theta) = \theta^\top \Phi(x)$

- **(approximately) Matching theory and practice**
  - Fruitful discussions between theoreticians and practitioners
  - Quantitative theoretical analysis suggests practical improvements

- **Golden years of convexity in machine learning** (1995 to 201*)
  - Support vector machines and kernel methods
  - Inference in graphical models
  - Sparsity / low-rank models with first-order methods
  - Convex relaxation of unsupervised learning problems
  - Optimal transport
  - Stochastic methods for large-scale learning and online learning

Convex optimization problems

- **Convexity in machine learning**
  - Convex loss and linear predictions $h(x, \theta) = \theta^\top \Phi(x)$
- **(approximately) Matching theory and practice**
  - Fruitful discussions between theoreticians and practitioners
  - Quantitative theoretical analysis suggests practical improvements
- **Golden years of convexity in machine learning** (1995 to 201*)
  - Support vector machines and kernel methods
  - Inference in graphical models
  - Sparsity / low-rank models with first-order methods
  - Convex relaxation of unsupervised learning problems
  - Optimal transport
  - Stochastic methods for large-scale learning and online learning

Exponentially convergent SGD for smooth finite sums

- **Finite sums**: $\displaystyle \min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n}\sum_{i=1}^{n} f_i(\theta) = \frac{1}{n}\sum_{i=1}^{n} \Big\{ \ell\big(y_i, h(x_i, \theta)\big) + \lambda \Omega(\theta) \Big\}$

Exponentially convergent SGD for smooth finite sums

- **Finite sums**: $\displaystyle \min_{\theta \in \mathbb{R}^d} \ \frac{1}{n}\sum_{i=1}^n f_i(\theta) = \frac{1}{n}\sum_{i=1}^n \Big\{ \ell\big(y_i, h(x_i,\theta)\big) + \lambda\Omega(\theta) \Big\}$

- **Non-accelerated algorithms** (with similar properties)
  - SAG (Le Roux, Schmidt, and Bach, 2012)
  - SDCA (Shalev-Shwartz and Zhang, 2013)
  - SVRG (Johnson and Zhang, 2013; Zhang et al., 2013)
  - MISO (Mairal, 2015), Finito (Defazio et al., 2014a)
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014b), etc...

$$\theta_t = \theta_{t-1} - \gamma \Big[\nabla f_{i(t)}(\theta_{t-1}) \qquad\qquad\qquad\qquad \Big]$$

Exponentially convergent SGD for smooth finite sums

- **Finite sums**: $\min_{\theta \in \mathbb{R}^d} \ \frac{1}{n}\sum_{i=1}^n f_i(\theta) = \frac{1}{n}\sum_{i=1}^n \Big\{ \ell\big(y_i, h(x_i,\theta)\big) + \lambda \Omega(\theta) \Big\}$

- **Non-accelerated algorithms** (with similar properties)
  - SAG (Le Roux, Schmidt, and Bach, 2012)
  - SDCA (Shalev-Shwartz and Zhang, 2013)
  - SVRG (Johnson and Zhang, 2013; Zhang et al., 2013)
  - MISO (Mairal, 2015), Finito (Defazio et al., 2014a)
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014b), etc...

$$\theta_t = \theta_{t-1} - \gamma \Big[\nabla f_{i(t)}(\theta_{t-1}) + \frac{1}{n}\sum_{i=1}^n y_i^{t-1} - y_{i(t)}^{t-1} \Big]$$

Exponentially convergent SGD for smooth finite sums

- **Finite sums**: $\min_{\theta \in \mathbb{R}^d} \ \frac{1}{n}\sum_{i=1}^{n} f_i(\theta) = \frac{1}{n}\sum_{i=1}^{n} \Big\{ \ell\big(y_i, h(x_i,\theta)\big) + \lambda\Omega(\theta) \Big\}$

- **Non-accelerated algorithms** (with similar properties)
  - SAG (Le Roux, Schmidt, and Bach, 2012)
  - SDCA (Shalev-Shwartz and Zhang, 2013)
  - SVRG (Johnson and Zhang, 2013; Zhang et al., 2013)
  - MISO (Mairal, 2015), Finito (Defazio et al., 2014a)
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014b), etc...

- **Accelerated algorithms**
  - Shalev-Shwartz and Zhang (2014); Nitanda (2014)
  - Lin et al. (2015b); Defazio (2016), etc...
  - Catalyst (Lin, Mairal, and Harchaoui, 2015a)

Exponentially convergent SGD for finite sums

- **Running-time to reach precision** ε (with $\kappa =$ condition number)

Gradient descent	$d\times$	$n\kappa$	$\times \log \frac{1}{\varepsilon}$
Accelerated gradient descent	$d\times$	$n\sqrt{\kappa}$	$\times \log \frac{1}{\varepsilon}$

Exponentially convergent SGD for finite sums

- **Running-time to reach precision** ε (with κ = condition number)

Stochastic gradient descent	$d\times$	κ	$\times$	$\frac{1}{\varepsilon}$
Gradient descent	$d\times$	$n\kappa$	$\times \log$	$\frac{1}{\varepsilon}$
Accelerated gradient descent	$d\times$	$n\sqrt{\kappa}$	$\times \log$	$\frac{1}{\varepsilon}$
SAG(A), SVRG, SDCA, MISO	$d\times$	$(n+\kappa)$	$\times \log$	$\frac{1}{\varepsilon}$

Exponentially convergent SGD for finite sums

- **Running-time to reach precision** ε (with κ = condition number)

Stochastic gradient descent	$d\times \quad \kappa \quad \times \quad \frac{1}{\varepsilon}$
Gradient descent	$d\times \quad n\kappa \quad \times \log\frac{1}{\varepsilon}$
Accelerated gradient descent	$d\times \quad n\sqrt{\kappa} \quad \times \log\frac{1}{\varepsilon}$
SAG(A), SVRG, SDCA, MISO	$d\times \quad (n+\kappa) \quad \times \log\frac{1}{\varepsilon}$
Accelerated versions	$d\times \quad (n+\sqrt{n\kappa}) \quad \times \log\frac{1}{\varepsilon}$

NB: slightly different (smaller) notion of condition number for batch methods

Exponentially convergent SGD for finite sums

- **Running-time to reach precision** ε (with κ = condition number)

Stochastic gradient descent	$d\times \quad \kappa \quad \times \quad \frac{1}{\varepsilon}$
Accelerated gradient descent	$d\times \quad n\sqrt{\kappa} \quad \times \log \frac{1}{\varepsilon}$
SAG(A), SVRG, SDCA, MISO	$d\times \quad (n+\kappa) \quad \times \log \frac{1}{\varepsilon}$
Accelerated versions	$d\times \ (n+\sqrt{n\kappa}) \quad \times \log \frac{1}{\varepsilon}$

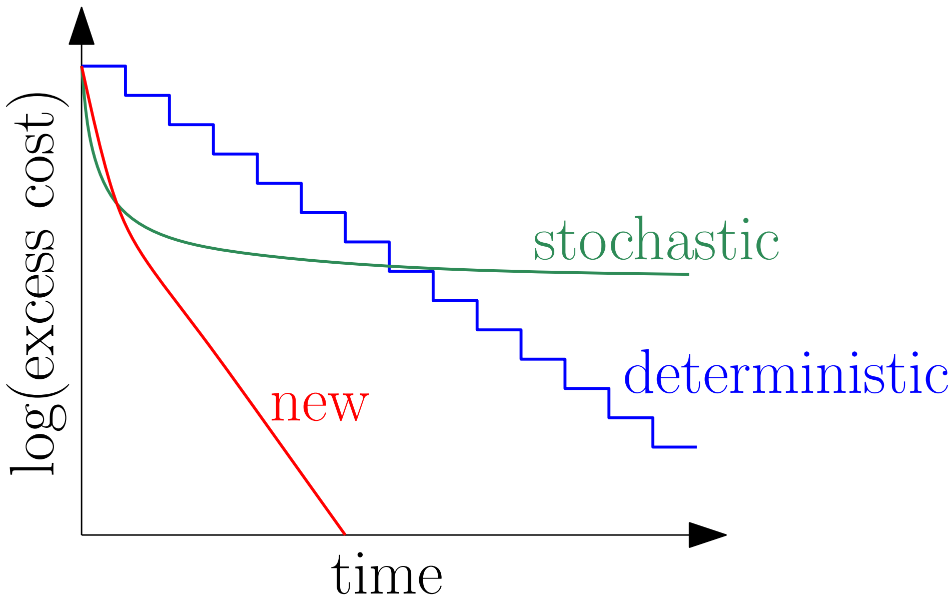
- **Beating two lower bounds** (Nemirovski and Yudin, 1983; Nesterov, 2004): with additional assumptions

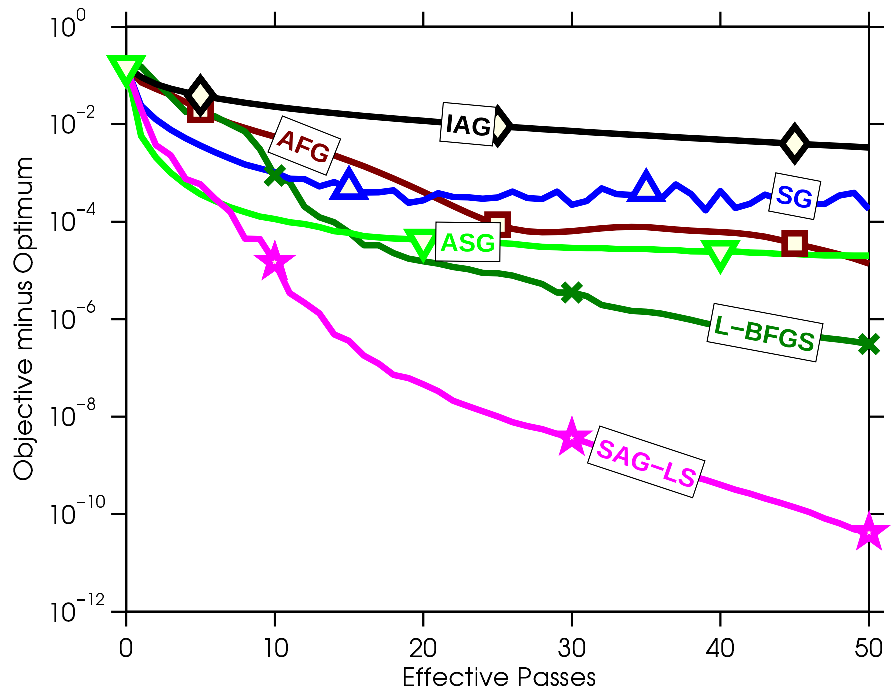
(1) stochastic gradient: exponential rate for finite sums

(2) full gradient: better exponential rate using the sum structure

- **Matching lower bounds** (Woodworth and Srebro, 2016; Lan, 2015)

Exponentially convergent SGD for finite sums
From theory to practice and vice-versa





- **Empirical performance “matches” theoretical guarantees**

- **Theoretical analysis suggests practical improvements**
  - Non-uniform sampling, acceleration
  - Matching upper and lower bounds

Convex optimization for machine learning
From theory to practice and vice-versa

- **Empirical performance "matches" theoretical guarantees**
- **Theoretical analysis suggests practical improvements**

Convex optimization for machine learning
From theory to practice and vice-versa

- **Empirical performance "matches" theoretical guarantees**

- **Theoretical analysis suggests practical improvements**

- **Many other well-understood areas**
  - Single pass SGD and generalization errors
  - From least-squares to convex losses
  - Non-parametric and high-dimensional regression
  - Randomized linear algebra
  - Bandit problems
  - etc...

Convex optimization for machine learning
From theory to practice and vice-versa

- **Empirical performance “matches” theoretical guarantees**

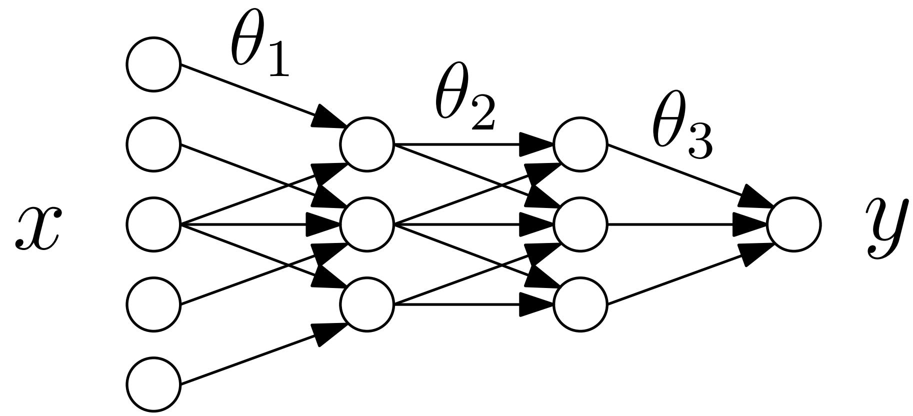
- **Theoretical analysis suggests practical improvements**

- **Many other well-understood areas**
  - Single pass SGD and generalization errors
  - From least-squares to convex losses
  - Non-parametric and high-dimensional regression
  - Randomized linear algebra
  - Bandit problems
  - etc...

- **What about deep learning?**

Theoretical analysis of deep learning

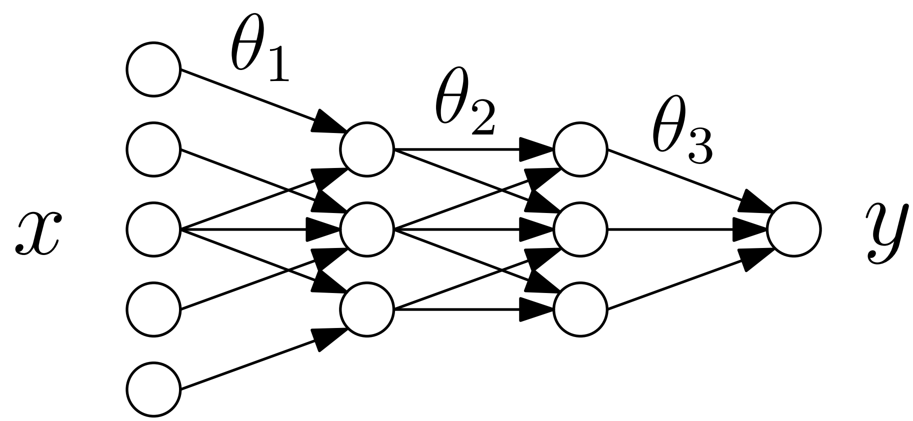
- **Multi-layer neural network** $h(x,\theta) = \theta_m^\top \sigma(\theta_{m-1}^\top \sigma(\cdots \theta_2^\top \sigma(\theta_1^\top x))$



  – NB: already a simplification

Theoretical analysis of deep learning

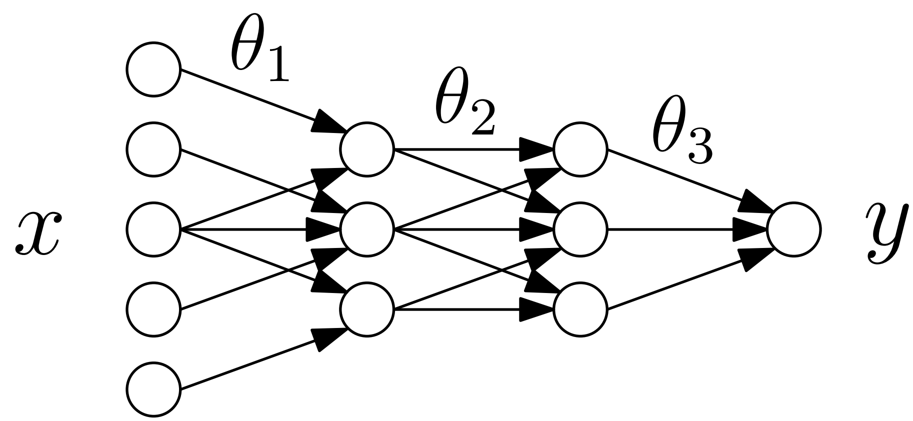
- **Multi-layer neural network** $h(x, \theta) = \theta_m^\top \sigma(\theta_{m-1}^\top \sigma(\cdots \theta_2^\top \sigma(\theta_1^\top x))$



- **Generalization guarantees**
  - See "MythBusters: A Deep Learning Edition" by Sasha Rakhlin
  - Bartlett et al. (2017); Golowich et al. (2018)

Theoretical analysis of deep learning

- **Multi-layer neural network** $h(x, \theta) = \theta_m^\top \sigma(\theta_{m-1}^\top \sigma(\cdots \theta_2^\top \sigma(\theta_1^\top x))$

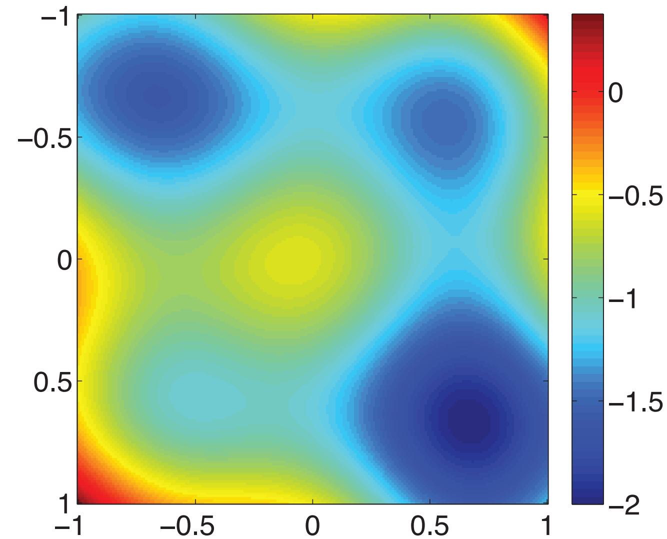


- **Generalization guarantees**
  - See "MythBusters: A Deep Learning Edition" by Sasha Rakhlin
  - Bartlett et al. (2017); Golowich et al. (2018)

- **Optimization**
  - Non-convex optimization problems

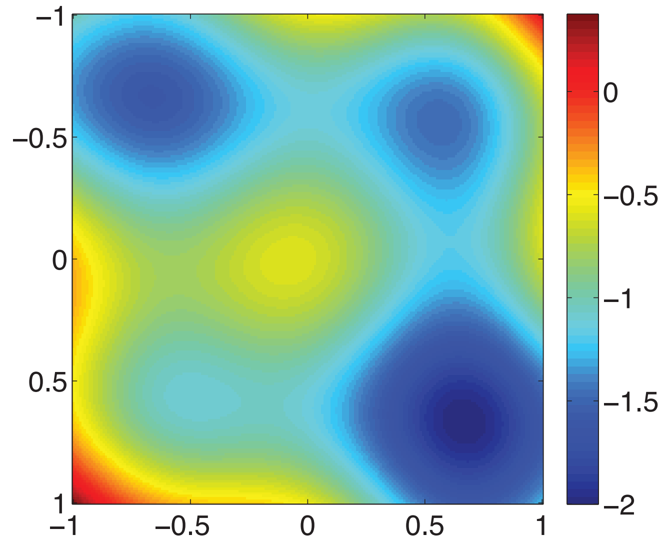
Optimization for multi-layer neural networks

- **What can go wrong with non-convex optimization problems?**
  - Local minima
  - Stationary points
  - Plateaux
  - Bad initialization
  - etc...



Optimization for multi-layer neural networks

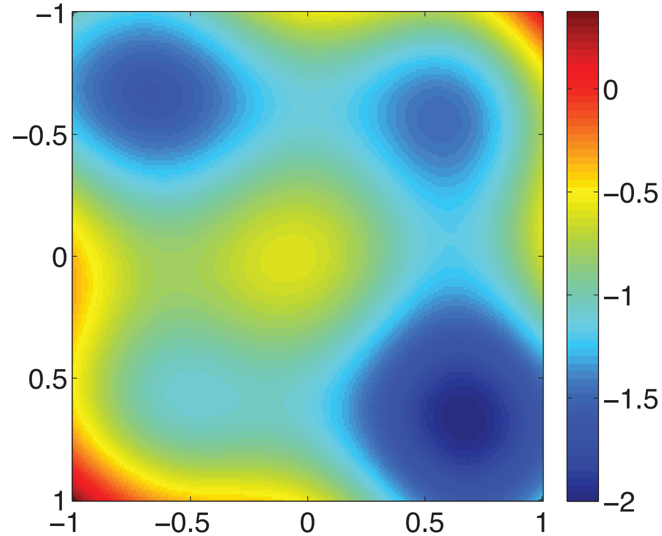
- **What can go wrong with non-convex optimization problems?**
  - Local minima
  - Stationary points
  - Plateaux
  - Bad initialization
  - etc...



- **Generic local theoretical guarantees**
  - Convergence to stationary points or local minima
  - See, e.g., Lee et al. (2016); Jin et al. (2017)

Optimization for multi-layer neural networks

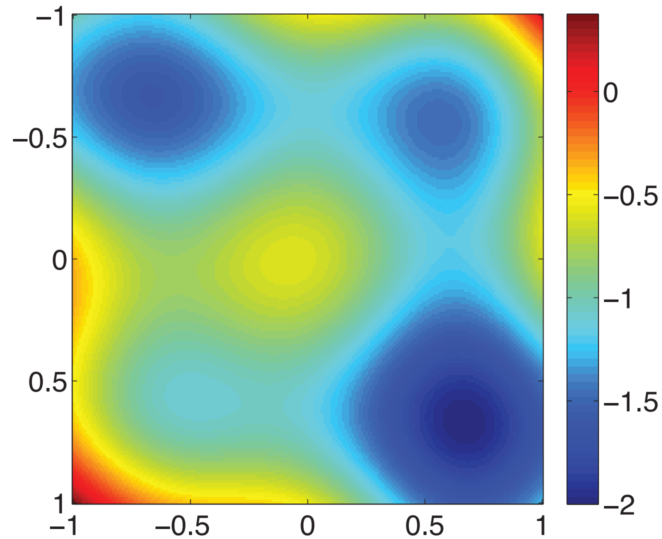
- **What can go wrong with non-convex optimization problems?**
  - Local minima
  - Stationary points
  - Plateaux
  - Bad initialization
  - etc...



- **General global performance guarantees impossible to obtain**

Optimization for multi-layer neural networks

- **What can go wrong with non-convex optimization problems?**



  – Local minima
  – Stationary points
  – Plateaux
  – Bad initialization
  – etc...

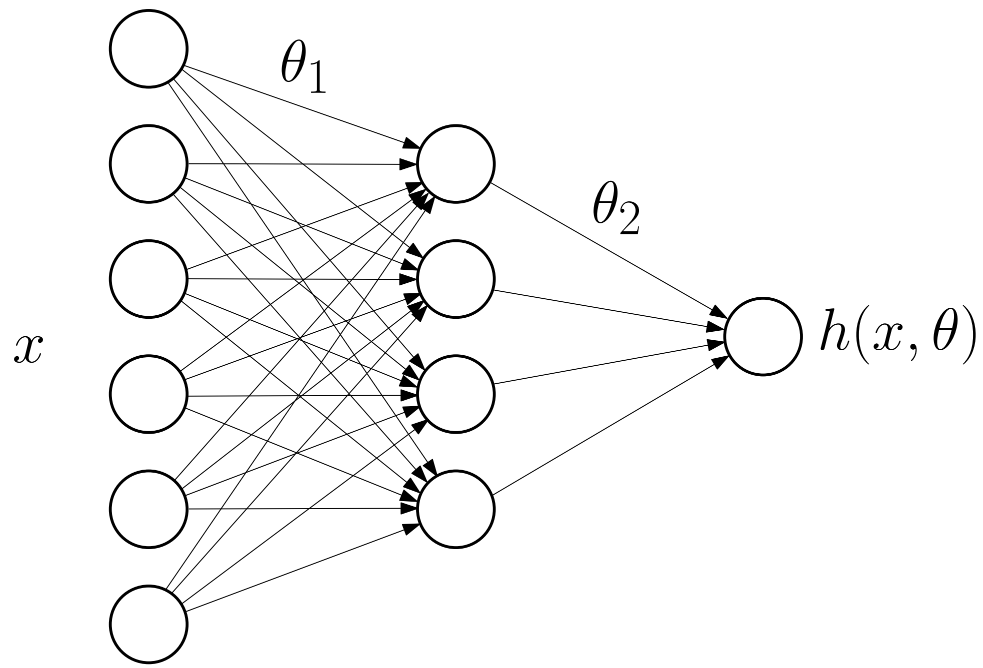
- **General global performance guarantees impossible to obtain**

- **Special case of (deep) neural networks**
  – Most local minima are equivalent (Choromanska et al., 2015)
  – No spurrious local minima (Soltanolkotabi et al., 2018)
  – NB: see Jain and Kar (2017) for guarantees in other contexts

Gradient descent for a single hidden layer

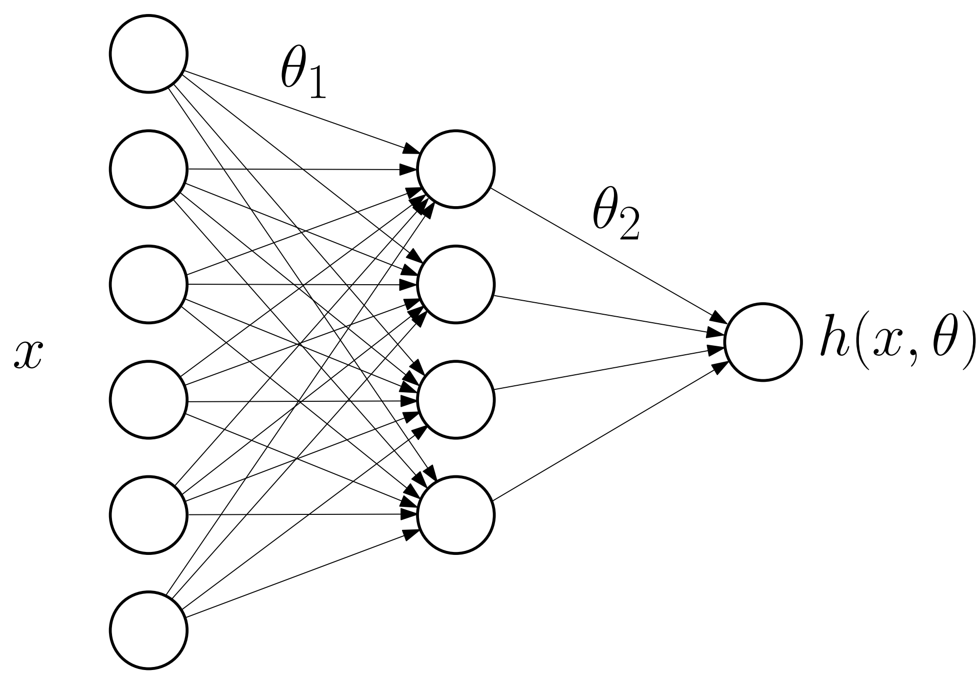
- **Predictor**: $h(x) = \theta_2^\top \sigma(\theta_1^\top x) = \sum_{i=1}^m \theta_2(i) \cdot \sigma\big[\theta_1(\cdot, i)^\top x\big]$

- **Goal**: minimize $R(h) = \mathbb{E}_{p(x,y)} \ell(y, h(x))$, with R convex



Gradient descent for a single hidden layer

- **Predictor**: $h(x) = \theta_2^\top \sigma(\theta_1^\top x) = \sum_{i=1}^m \theta_2(i) \cdot \sigma\big[\theta_1(\cdot,i)^\top x\big]$
  - Family: $h = \dfrac{1}{m}\displaystyle\sum_{i=1}^m \Psi(w_i)$ with $\Psi(w_i)(x) = m\theta_2(i) \cdot \sigma\big[\theta_1(\cdot,i)^\top x\big]$
- **Goal**: minimize $R(h) = \mathbb{E}_{p(x,y)} \ell(y, h(x))$, with R convex



Gradient descent for a single hidden layer

- **Predictor**: $h(x) = \theta_2^\top \sigma(\theta_1^\top x) = \sum_{i=1}^m \theta_2(i) \cdot \sigma\big[\theta_1(\cdot, i)^\top x\big]$
  - Family: $h = \dfrac{1}{m}\displaystyle\sum_{i=1}^m \Psi(w_i)$ with $\Psi(w_i)(x) = m\theta_2(i) \cdot \sigma\big[\theta_1(\cdot, i)^\top x\big]$
- **Goal**: minimize $R(h) = \mathbb{E}_{p(x,y)} \ell(y, h(x))$, with R convex
- **Main insight**
  - $h = \dfrac{1}{m}\displaystyle\sum_{i=1}^m \Psi(w_i) = \int_{\mathcal{W}} \Psi(w) d\mu(w)$ with $d\mu(w) = \dfrac{1}{m}\displaystyle\sum_{i=1}^m \delta_{w_i}$
  - Overparameterized models with m large $\approx$ measure μ with densities
  - Barron (1993); Kurkova and Sanguineti (2001); Bengio et al. (2006); Rosset et al. (2007); Bach (2014)

Optimization on measures

- **Minimize with respect to measure** μ: $R\Big(\int_{\mathcal{W}} \Psi(w) d\mu(w)\Big)$
  - Convex optimization problem on measures
  - Frank-Wolfe techniques for incremental learning
  - Non-tractable (Bach, 2014), not what is used in practice

Optimization on measures

- **Minimize with respect to measure** μ: $R\Big(\int_{\mathcal{W}} \Psi(w) d\mu(w)\Big)$
  - Convex optimization problem on measures
  - Frank-Wolfe techniques for incremental learning
  - Non-tractable (Bach, 2014), not what is used in practice

- **Represent** μ **by a finite set of "particles"** $\mu = \frac{1}{m}\sum_{i=1}^{m} \delta_{w_i}$
  - Backpropagation $=$ gradient descent on $(w_1, \dots, w_m)$

- **Two questions**:
  - Algorithm limit when number of particles m gets large

  - Global convergence

Optimization on measures

- **Minimize with respect to measure** μ: $R\Big(\int_{\mathcal{W}} \Psi(w) d\mu(w)\Big)$
  - Convex optimization problem on measures
  - Frank-Wolfe techniques for incremental learning
  - Non-tractable (Bach, 2014), not what is used in practice
- **Represent** μ **by a finite set of "particles"** $\mu = \frac{1}{m}\sum_{i=1}^{m} \delta_{w_i}$
  - Backpropagation $=$ gradient descent on $(w_1, \dots, w_m)$
- **Two questions**:
  - Algorithm limit when number of particles m gets large
    Wasserstein gradient flow (Nitanda and Suzuki, 2017)
  - Global convergence
    to the optimal measure μ (Chizat and Bach, 2018a)

Many particle limit and global convergence (Chizat and Bach, 2018a)

- **General framework**: minimize $F(\mu) = R\Big(\int_{\mathcal{W}} \Psi(w) d\mu(w) \Big)$
  - Minimizing $F_m(w_1, \dots, w_m) = R\Big(\frac{1}{m} \sum_{i=1}^{m} \Psi(w_i) \Big)$

Many particle limit and global convergence (Chizat and Bach, 2018a)

- **General framework**: minimize $F(\mu) = R\Big(\int_{\mathcal{W}} \Psi(w) d\mu(w)\Big)$
  - Minimizing $F_m(w_1, \dots, w_m) = R\Big(\frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)\Big)$
  - Gradient flow $\dot{W} = -m\nabla F_m(W)$, with $W = (w_1, \dots, w_m)$
  - Idealization of (stochastic) gradient descent

Many particle limit and global convergence (Chizat and Bach, 2018a)

- **General framework**: minimize $F(\mu) = R\Big(\int_{\mathcal{W}} \Psi(w) d\mu(w)\Big)$

  – Minimizing $F_m(w_1, \dots, w_m) = R\Big(\frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)\Big)$
  – Gradient flow $\dot{W} = -m\nabla F_m(W)$, with $W = (w_1, \dots, w_m)$
  – Idealization of (stochastic) gradient descent

- **Limit when m tends to infinity**

  – Wasserstein gradient flow (Nitanda and Suzuki, 2017; Chizat and Bach, 2018a; Mei, Montanari, and Nguyen, 2018; Sirignano and Spiliopoulos, 2018; Rotskoff and Vanden-Eijnden, 2018)

- NB: for more details on gradient flows, see Ambrosio et al. (2008)

(intuitive) link with Wasserstein gradient flows

- Gradient flow on Euclidean spaces, for smooth function $f : \mathcal{A} \to \mathbb{R}$
  - Given $a = a(t)$, $a(t + dt)$ is the minimizer of $f(b) + \dfrac{1}{2dt}\|b - a\|^2$
  - Optimality conditions: $\nabla f(b) + \dfrac{1}{dt}(b - a) = 0$

(intuitive) link with Wasserstein gradient flows

- Gradient flow on Euclidean spaces, for smooth function $f : \mathcal{A} \to \mathbb{R}$
  - Given $a = a(t)$, $a(t + dt)$ is the minimizer of $f(b) + \frac{1}{2dt}\|b - a\|^2$
  - Optimality conditions: $\nabla f(b) + \frac{1}{dt}(b - a) = 0$
  - For smooth f, $\nabla f(b) - \nabla f(a) = O(dt)$
  - Thus $a(t + dt) = b = a - (dt)\nabla f(a) = a(t) - (dt)\nabla f(a(t))$
  - Equivalent to regular ODE: $\dot{a} = -\nabla f(a)$

(intuitive) link with Wasserstein gradient flows

- Given measure $\mu = \mu(t)$, $\nu = \mu(t + dt)$ defined as the minimizer of

$$F(\nu) + \frac{W_2^2(\mu,\nu)}{2dt} = R\Big(\int \Psi(v) d\nu(v) \Big) + \frac{1}{2dt} \inf_{\gamma \in \Pi(\mu,\nu)} \int \|v - w\|^2 d\gamma(w,v)$$

  - $\Pi(\mu,\nu)$ set of joint distributions with marginals μ and ν

(intuitive) link with Wasserstein gradient flows

- Given measure $\mu = \mu(t)$, $\nu = \mu(t + dt)$ defined as the minimizer of

$$F(\nu) + \frac{W_2^2(\mu, \nu)}{2dt} = R\Big(\int \Psi(v) d\nu(v)\Big) + \frac{1}{2dt} \inf_{\gamma \in \Pi(\mu,\nu)} \int \|v - w\|^2 d\gamma(w, v)$$

$$\approx \Big\langle \nabla R\Big(\int \Psi d\mu\Big), \int \Psi(v) d\nu(v) \Big\rangle + \inf_{\gamma \in \Pi(\mu,\nu)} \int \frac{\|v - w\|^2}{2dt} d\gamma(w, v) + \text{cst}$$

(intuitive) link with Wasserstein gradient flows

- Given measure $\mu = \mu(t)$, $\nu = \mu(t + dt)$ defined as the minimizer of

$$
\begin{aligned}
F(\nu) + \frac{W_2^2(\mu,\nu)}{2dt} &= R\Big(\int \Psi(v) d\nu(v) \Big) + \frac{1}{2dt} \inf_{\gamma \in \Pi(\mu,\nu)} \int \|v - w\|^2 d\gamma(w,v) \\
&\approx \Big\langle \nabla R\Big(\int \Psi d\mu \Big), \int \Psi(v) d\nu(v) \Big\rangle + \inf_{\gamma \in \Pi(\mu,\nu)} \int \frac{\|v-w\|^2}{2dt} d\gamma(w,v) + \text{cst} \\
&\approx \inf_{\gamma \in \Pi(\mu,\nu)} \int \Big\{ \Big\langle \nabla R\Big(\int \Psi d\mu \Big), \Psi(v) \Big\rangle + \frac{\|v-w\|^2}{2dt} \Big\} d\gamma(w,v) + \text{cst}
\end{aligned}
$$

(intuitive) link with Wasserstein gradient flows

- Given measure $\mu = \mu(t)$, $\nu = \mu(t+dt)$ defined as the minimizer of

$$
\begin{aligned}
F(\nu) + \frac{W_2^2(\mu,\nu)}{2dt} &= R\Big(\int \Psi(v) d\nu(v)\Big) + \frac{1}{2dt} \inf_{\gamma \in \Pi(\mu,\nu)} \int \|v-w\|^2 d\gamma(w,v) \\
&\approx \Big\langle \nabla R\Big(\int \Psi d\mu\Big), \int \Psi(v) d\nu(v) \Big\rangle + \inf_{\gamma \in \Pi(\mu,\nu)} \int \frac{\|v-w\|^2}{2dt} d\gamma(w,v) + \text{cst} \\
&\approx \inf_{\gamma \in \Pi(\mu,\nu)} \int \Big\{ \Big\langle \nabla R\Big(\int \Psi d\mu\Big), \Psi(v) \Big\rangle + \frac{\|v-w\|^2}{2dt} \Big\} d\gamma(w,v) + \text{cst}
\end{aligned}
$$

  - Given $w \sim \mu$, $\nu(\cdot|w)$ is a Dirac at minimizer of $\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \Psi(v) \Big\rangle + \frac{\|v-w\|^2}{2dt}$, that is, at $v = w - dt \Big\langle \nabla R\Big(\int \Psi d\mu\Big), \frac{\partial \Psi}{\partial w}(w) \Big\rangle$

(intuitive) link with Wasserstein gradient flows

- Given measure $\mu = \mu(t)$, $\nu = \mu(t+dt)$ defined as the minimizer of

$$F(\nu) + \frac{W_2^2(\mu,\nu)}{2dt} = R\Big(\int \Psi(v)d\nu(v)\Big) + \frac{1}{2dt}\inf_{\gamma\in\Pi(\mu,\nu)}\int \|v-w\|^2 d\gamma(w,v)$$

$$\approx \Big\langle \nabla R\Big(\int \Psi d\mu\Big), \int \Psi(v)d\nu(v)\Big\rangle + \inf_{\gamma\in\Pi(\mu,\nu)}\int \frac{\|v-w\|^2}{2dt}d\gamma(w,v) + \text{cst}$$

$$\approx \inf_{\gamma\in\Pi(\mu,\nu)}\int \Big\{\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \Psi(v)\Big\rangle + \frac{\|v-w\|^2}{2dt}\Big\}d\gamma(w,v) + \text{cst}$$

  - Given $w \sim \mu$, $\nu(\cdot|w)$ is a Dirac at minimizer of $\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \Psi(v)\Big\rangle + \frac{\|v-w\|^2}{2dt}$, that is, at $v = w - dt\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \frac{\partial \Psi}{\partial w}(w)\Big\rangle$
  - If $\mu \approx \frac{1}{m}\sum_{i=1}^m \delta_{w_i}$, then $\mu_{t+dt} \approx \frac{1}{m}\sum_{i=1}^m \delta_{v_i}$ with $v_i = w_i - dt\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \frac{\partial \Psi}{\partial w}(w_i)\Big\rangle$

(intuitive) link with Wasserstein gradient flows

- Given measure $\mu = \mu(t)$, $\nu = \mu(t + dt)$ defined as the minimizer of

$$F(\nu) + \frac{W_2^2(\mu,\nu)}{2dt} = R\Big(\int \Psi(v)d\nu(v)\Big) + \frac{1}{2dt}\inf_{\gamma\in\Pi(\mu,\nu)}\int \|v-w\|^2 d\gamma(w,v)$$

$$\approx \Big\langle \nabla R\Big(\int \Psi d\mu\Big), \int \Psi(v)d\nu(v)\Big\rangle + \inf_{\gamma\in\Pi(\mu,\nu)}\int \frac{\|v-w\|^2}{2dt} d\gamma(w,v) + \text{cst}$$

$$\approx \inf_{\gamma\in\Pi(\mu,\nu)}\int \Big\{\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \Psi(v)\Big\rangle + \frac{\|v-w\|^2}{2dt}\Big\} d\gamma(w,v) + \text{cst}$$

  - Given $w \sim \mu$, $\nu(\cdot|w)$ is a Dirac at minimizer of $\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \Psi(v)\Big\rangle + \frac{\|v-w\|^2}{2dt}$, that is, at $v = w - dt\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \frac{\partial \Psi}{\partial w}(w)\Big\rangle$
  - If $\mu \approx \frac{1}{m}\sum_{i=1}^m \delta_{w_i}$, then $\mu_{t+dt} \approx \frac{1}{m}\sum_{i=1}^m \delta_{v_i}$ with $v_i = w_i - dt\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \frac{\partial \Psi}{\partial w}(w_i)\Big\rangle$
  - Evolution of particles as $w_i(t+dt) = w_i - dt\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \frac{\partial \Psi}{\partial w}(w_i)\Big\rangle$

(intuitive) link with gradient flows

- Evolution of particles as $w_i(t+dt) = w_i - dt \Big\langle \nabla R\Big(\int \Psi d\mu\Big), \frac{\partial \Psi}{\partial w}(w_i)\Big\rangle$

(intuitive) link with gradient flows

- Evolution of particles as $w_i(t+dt) = w_i - dt\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \dfrac{\partial \Psi}{\partial w}(w_i)\Big\rangle$

- Equivalence with gradient flow $\dot{W} = -m F_m(W)$
  - with $W = (w_1, \dots, w_m)$, for $F_m(w_1, \dots, w_m) = R\Big(\dfrac{1}{m}\sum_{i=1}^m \Psi(w_i)\Big)$
  - as $\dfrac{\partial F_m}{\partial w_i} = \Big\langle \nabla R\Big(\dfrac{1}{m}\sum_{i=1}^m \Psi(w_i)\Big), \dfrac{1}{m}\dfrac{\partial \Psi}{\partial w_i}\Big\rangle$

(intuitive) link with gradient flows

- Evolution of particles as $w_i(t+dt) = w_i - dt\Big\langle \nabla R\Big(\int \Psi d\mu\Big), \dfrac{\partial \Psi}{\partial w}(w_i)\Big\rangle$

- Equivalence with gradient flow $\dot{W} = -mF_m(W)$
  - with $W = (w_1, \dots, w_m)$, for $F_m(w_1, \dots, w_m) = R\Big(\dfrac{1}{m}\sum_{i=1}^{m}\Psi(w_i)\Big)$
  - as $\dfrac{\partial F_m}{\partial w_i} = \Big\langle \nabla R\Big(\dfrac{1}{m}\sum_{i=1}^{m}\Psi(w_i)\Big), \dfrac{1}{m}\dfrac{\partial \Psi}{\partial w_i}\Big\rangle$

- **Global convergence ?**
  - Difficulty 1: potentially many local minima and stationary points (even if R is convex)
  - Difficulty 2: globally optimal measure is often singular

Many particle limit and global convergence (Chizat and Bach, 2018a)

- **Two ingredients**: homogeneity and initialization

Many particle limit and global convergence (Chizat and Bach, 2018a)

- **Two ingredients**: homogeneity and initialization
- **Homogeneity** (see, e.g., Haeffele and Vidal, 2017; Bach et al., 2008)
  - Full or partial, e.g., $\Psi(w_i)(x) = m\theta_2(i) \cdot \sigma\big[\theta_1(\cdot, i)^\top x\big]$
  - Applies to rectified linear units (but also to sigmoid activations)
- **Sufficiently spread initial measure**
  - Needs to cover the entire sphere of directions

Many particle limit and global convergence (Chizat and Bach, 2018a)

- **Two ingredients**: homogeneity and initialization

- **Homogeneity** (see, e.g., Haeffele and Vidal, 2017; Bach et al., 2008)
  - Full or partial, e.g., $\Psi(w_i)(x) = m\theta_2(i) \cdot \sigma\big[\theta_1(\cdot, i)^\top x\big]$
  - Applies to rectified linear units (but also to sigmoid activations)

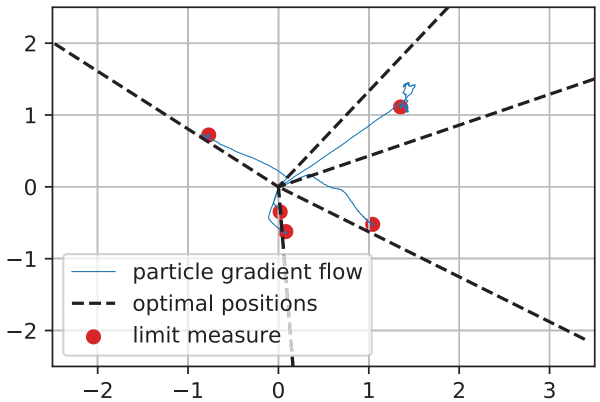
- **Sufficiently spread initial measure**
  - Needs to cover the entire sphere of directions

- NB 1 : see precise definitions and statement in paper

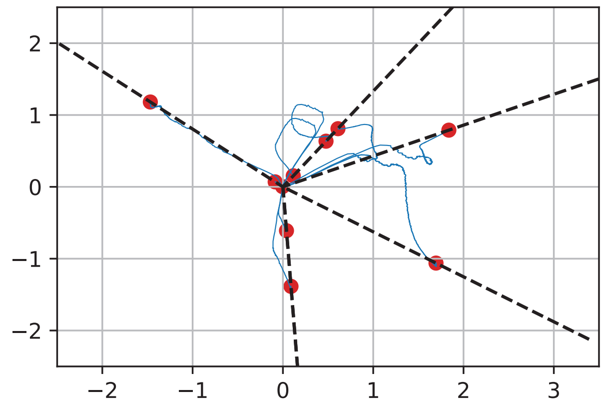
- NB 2 : also applies to spike deconvolution

Simple simulations with neural networks

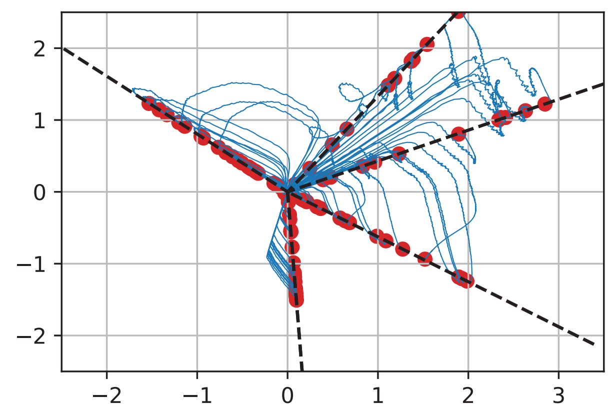
- ReLU units with $d = 2$ (optimal predictor has 5 neurons)



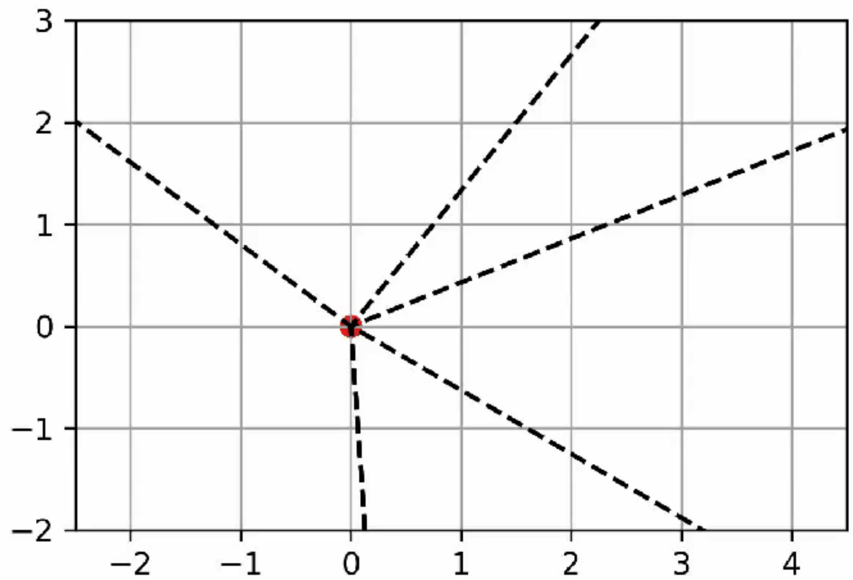
5 neurons



10 neurons

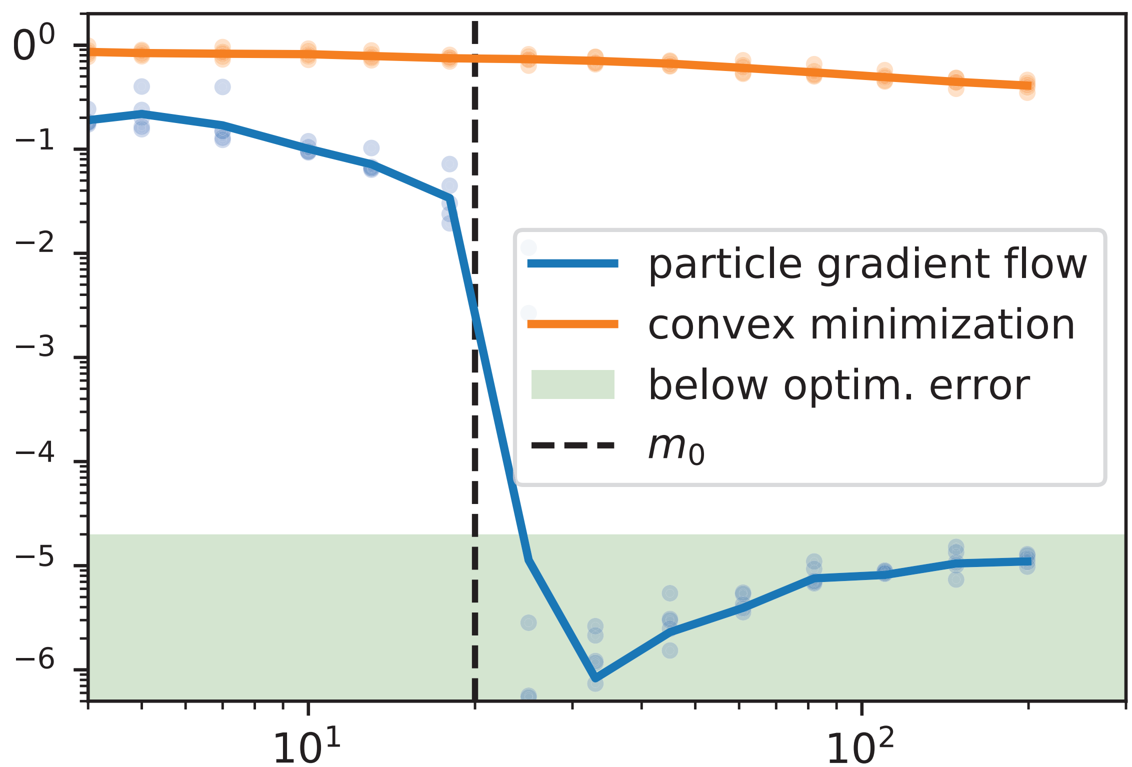


100 neurons



Simple simulations with neural networks

- ReLU units with $d = 100$ (optimal predictor has m_0 neurons)
  - Comparing gradient descent on particles with sampling (and reweighting by convex optimization) fixed particles
  - No quantitative analysis (yet)



From qualitative to quantitative results ?

- **Adding noise** (Mei, Montanari, and Nguyen, 2018)
  - On top of SGD "à la Langevin" $\Rightarrow$ convergence to a diffusion
  - Quantitative analysis of the needed number of neurons
  - Recent improvement (Mei, Misiakiewicz, and Montanari, 2019)

From qualitative to quantitative results ?

- **Adding noise** (Mei, Montanari, and Nguyen, 2018)
  - On top of SGD "à la Langevin" $\Rightarrow$ convergence to a diffusion
  - Quantitative analysis of the needed number of neurons
  - Recent improvement (Mei, Misiakiewicz, and Montanari, 2019)

- **Recent strong activity on ArXiv**
  - `https://arxiv.org/abs/1810.02054`
  - `https://arxiv.org/abs/1811.03804`
  - `https://arxiv.org/abs/1811.03962`
  - `https://arxiv.org/abs/1811.04918`
  - See also Jacot et al. (2018)

From qualitative to quantitative results ?

- **Adding noise** (Mei, Montanari, and Nguyen, 2018)
  - On top of SGD "à la Langevin" $\Rightarrow$ convergence to a diffusion
  - Quantitative analysis of the needed number of neurons
  - Recent improvement (Mei, Misiakiewicz, and Montanari, 2019)

- **Recent strong activity on ArXiv**
  - Global quantitative linear convergence of gradient descent
  - Zero training loss
  - Extends to deep architectures and skip connections

From qualitative to quantitative results ?

- **Mean-field limit**: $h(x) = \frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)$
  - With w_i initialized randomly (with variance independent of m)
  - Dynamics equivalent to Wasserstein gradient flow
  - Convergence to global minimum of $R(\int \Psi d\mu)$

From qualitative to quantitative results ?

- **Mean-field limit**: $h(x) = \frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)$
  - With w_i initialized randomly (with variance independent of m)
  - Dynamics equivalent to Wasserstein gradient flow
  - Convergence to global minimum of $R(\int \Psi d\mu)$
- **Recent strong activity on ArXiv**
  - Corresponds to initializing with weights which are $\sqrt{m}$ times larger
  - Where does it converge to?

From qualitative to quantitative results ?

- **Mean-field limit**: $h(x) = \frac{1}{m}\sum_{i=1}^{m}\Psi(w_i)$
  - With w_i initialized randomly (with variance independent of m)
  - Dynamics equivalent to Wasserstein gradient flow
  - Convergence to global minimum of $R(\int \Psi d\mu)$
- **Recent strong activity on ArXiv**
  - Corresponds to initializing with weights which are $\sqrt{m}$ times larger
  - Where does it converge to?
- **Equivalence to lazy training** (Chizat and Bach, 2018b)
  - Convergence to a positive-definite kernel method
  - Neurons move infinitesimally

Lazy training (Chizat and Bach, 2018b)

- **Generic criterion** $G(W) = R(h(W))$ **to minimize w.r.t.** W
  - Example: R loss, $h = \frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)$ prediction function
  - Introduce (large) scale factor $\alpha > 0$ and $G_\alpha(W) = G(\alpha h(W))/\alpha^2$
  - Initialize $W(0)$ such that $\alpha W(0)$ is bounded
    (using e.g., $\mathbb{E}\Psi(w_i) = 0$)

Lazy training (Chizat and Bach, 2018b)

- **Generic criterion** $G(W) = R(h(W))$ **to minimize w.r.t.** W
  - Example: R loss, $h = \frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)$ prediction function
  - Introduce (large) scale factor $\alpha > 0$ and $G_\alpha(W) = G(\alpha h(W))/\alpha^2$
  - Initialize $W(0)$ such that $\alpha W(0)$ is bounded
    (using e.g., $\mathbb{E}\Psi(w_i) = 0$)

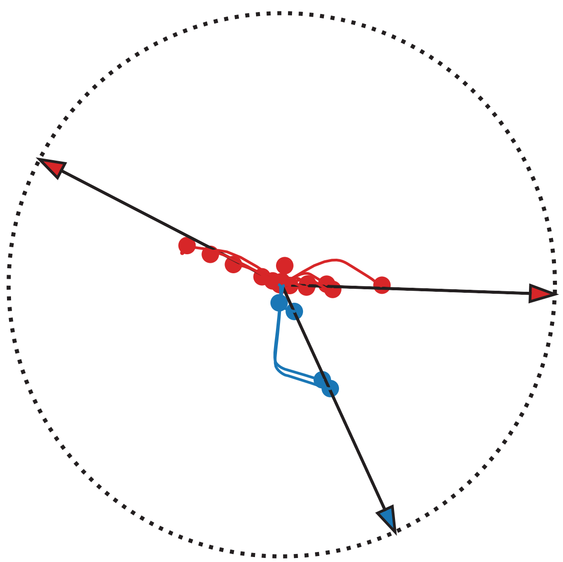
- **Proposition** (informal)
  - Assume differential of h at $W(0)$ is surjective
  - Gradient flow $\dot{W} = -\nabla G_\alpha(W)$ is such that

    $\|W(t) - W(0)\| = O(1/\alpha)$ and $\alpha h(W(t)) \to \arg\min_h R(h)$ "linearly"

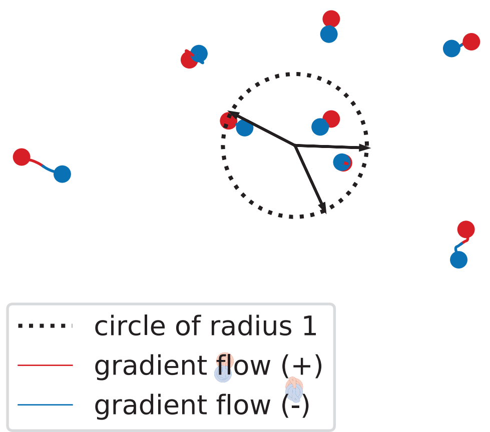
$\Rightarrow$ Equivalent to a linear model
$h(W) \approx h(W(0)) + (W - W(0))^\top \nabla h(W(0))$

Lazy training (Chizat and Bach, 2018b)

- **Generic criterion** $G(W) = R(h(W))$ **to minimize w.r.t.** W
  - Example: R loss, $h = \frac{1}{m}\sum_{i=1}^{m}\Psi(w_i)$ prediction function
  - Introduce (large) scale factor $\alpha > 0$ and $G_\alpha(W) = G(\alpha h(W))/\alpha^2$
  - Initialize $W(0)$ such that $\alpha W(0)$ is bounded
    (using e.g., $\mathbb{E}\Psi(w_i) = 0$)

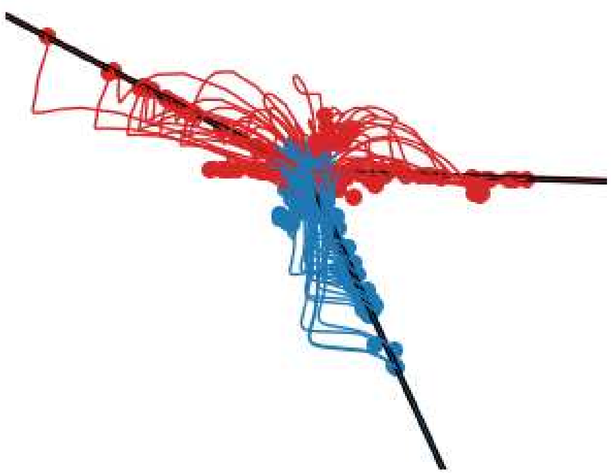


Lazy
$\Rightarrow$

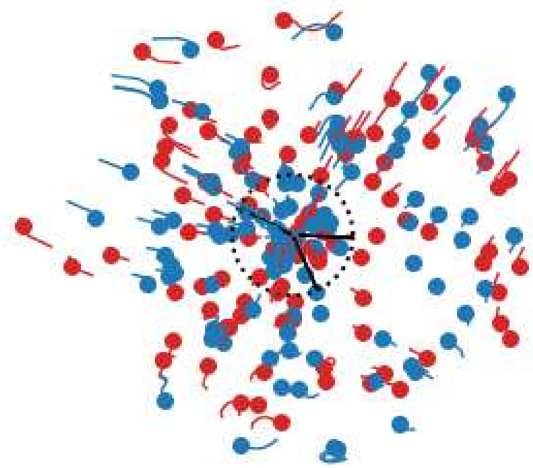


Lazy training (Chizat and Bach, 2018b)

- **Generic criterion** $G(W) = R(h(W))$ **to minimize w.r.t.** W
  - Example: R loss, $h = \frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)$ prediction function
  - Introduce (large) scale factor $\alpha > 0$ and $G_\alpha(W) = G(\alpha h(W))/\alpha^2$
  - Initialize $W(0)$ such that $\alpha W(0)$ is bounded
    (using e.g., $\mathbb{E}\Psi(w_i) = 0$)

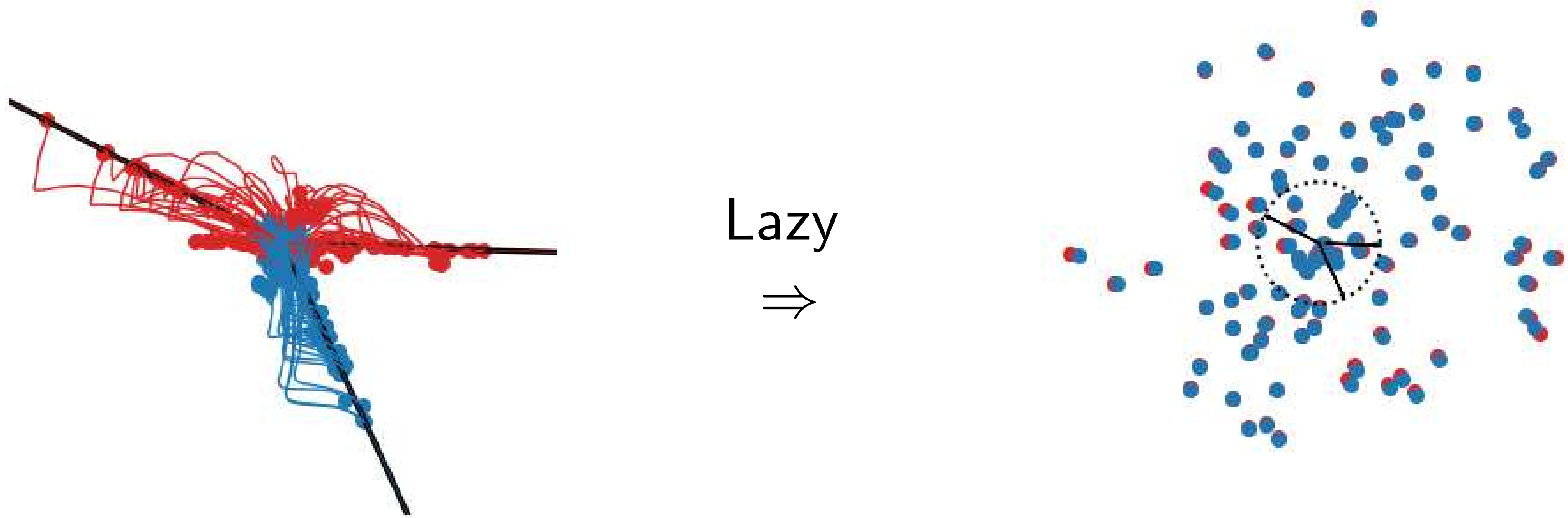


Lazy
$\Rightarrow$



Lazy training (Chizat and Bach, 2018b)

- **Generic criterion $G(W) = R(h(W))$ to minimize w.r.t. W**
  - Example: R loss, $h = \frac{1}{m}\sum_{i=1}^{m} \Psi(w_i)$ prediction function
  - Introduce (large) scale factor $\alpha > 0$ and $G_\alpha(W) = G(\alpha h(W))/\alpha^2$
  - Initialize $W(0)$ such that $\alpha W(0)$ is bounded
    (using e.g., $\mathbb{E}\Psi(w_i) = 0$)



Lazy training (Chizat and Bach, 2018b)

- **Equivalence to kernel methods**
  - Still non-parametric estimation
  - See details and additional experiments in preprint

- **Does this really "demistify" generalization in deep networks?**

Lazy training (Chizat and Bach, 2018b)

- **Equivalence to kernel methods**
  - Still non-parametric estimation
  - See details and additional experiments in preprint

- **Does this really “demistify” generalization in deep networks?**
  - (first!) Guarantees for deep networks
  - Deep neural networks = efficient kernel methods?
  - Neurons don’t move?

Lazy training (Chizat and Bach, 2018b)

- **Equivalence to kernel methods**
  - Still non-parametric estimation
  - See details and additional experiments in preprint

- **Does this really "demistify" generalization in deep networks?**
  - (first!) Guarantees for deep networks
  - Deep neural networks = efficient kernel methods?
  - Neurons don't move?

- **What is actually happening in practice?** (ongoing work)
  - Between mean field regime and lazy regime?
  - Empirical comparison for state-of-the-art networks

Healthy interactions between theory, applications, and hype?

Healthy interactions between theory, applications, and hype?

- **Empirical successes of deep learning cannot be ignored**

Healthy interactions between theory, applications, and hype?

- **Empirical successes of deep learning cannot be ignored**
- **Scientific standards should not be lowered**
  - Critics and limits of theoretical and empirical results
  - Rigor beyond mathematical guarantees

References

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2008.

Francis Bach. Breaking the curse of dimensionality with convex neural networks. Technical Report 1412.8690, arXiv, 2014.

Francis Bach, Julien Mairal, and Jean Ponce. Convex sparse matrix factorizations. Technical Report 0812.1869, arXiv, 2008.

A. R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, 1993.

Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, pages 6240–6249, 2017.

Y. Bengio, N. Le Roux, P. Vincent, O. Delalleau, and P. Marcotte. Convex neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2006.

Lénaïc Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. Technical Report 1805.09545, arXiv, 2018a.

Lenaic Chizat and Francis Bach. A note on lazy training in supervised differentiable programming. Technical Report To appear, ArXiv, 2018b.

Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, pages 192–204, 2015.

A. Defazio, J. Domke, and T. S. Caetano. Finito: A faster, permutable incremental gradient method for big data problems. In *Proc. ICML*, 2014a.

Aaron Defazio. A simple practical accelerated method for finite sums. In *Advances in Neural Information Processing Systems*, pages 676–684, 2016.

Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, 2014b.

Noah Golowich, Alexander Rakhlin, and Ohad Shamir. Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pages 297–299, 2018.

Benjamin D. Haeffele and René Vidal. Global optimality in neural network training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7331–7339, 2017.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8580–8589, 2018.

Prateek Jain and Purushottam Kar. Non-convex optimization for machine learning. *Foundations and Trends in Machine Learning*, 10(3-4):142–336, 2017.

Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M. Kakade, and Michael I. Jordan. How to escape saddle points efficiently. *arXiv preprint arXiv:1703.00887*, 2017.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, 2013.

V. Kurkova and M. Sanguineti. Bounds on rates of variable-basis and neural-network approximation.

IEEE Transactions on Information Theory, 47(6):2659–2665, Sep 2001.

G. Lan. An optimal randomized incremental gradient method. Technical Report 1507.02000, arXiv, 2015.

N. Le Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

Jason D. Lee, Max Simchowitz, Michael I. Jordan, and Benjamin Recht. Gradient descent only converges to minimizers. In *Conference on Learning Theory*, pages 1246–1257, 2016.

H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems (NIPS)*, 2015a.

Qihang Lin, Zhaosong Lu, and Lin Xiao. An accelerated randomized proximal coordinate gradient method and its application to regularized empirical risk minimization. *SIAM Journal on Optimization*, 25(4):2244–2273, 2015b.

J. Mairal. Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, 25(2):829–855, 2015.

Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layers neural networks. Technical Report 1804.06561, arXiv, 2018.

Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit. *arXiv preprint arXiv:1902.06015*, 2019.

A. S. Nemirovski and D. B. Yudin. *Problem complexity and method efficiency in optimization*. John Wiley, 1983.

Y. Nesterov. *Introductory lectures on convex optimization: a basic course*. Kluwer, 2004.

A. Nitanda. Stochastic proximal gradient descent with acceleration techniques. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

Atsushi Nitanda and Taiji Suzuki. Stochastic particle gradient descent for infinite ensembles. *arXiv preprint arXiv:1712.05438*, 2017.

S. Rosset, G. Swirszcz, N. Srebro, and J. Zhu. ℓ_1-regularization in infinite dimensional feature spaces. In *Proceedings of the Conference on Learning Theory (COLT)*, 2007.

Grant M. Rotskoff and Eric Vanden-Eijnden. Neural networks as interacting particle systems: Asymptotic convexity of the loss landscape and universal scaling of the approximation error. *arXiv preprint arXiv:1805.00915*, 2018.

S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb):567–599, 2013.

S. Shalev-Shwartz and T. Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In *Proc. ICML*, 2014.

Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks. *arXiv preprint arXiv:1805.01053*, 2018.

Mahdi Soltanolkotabi, Adel Javanmard, and Jason D Lee. Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Transactions on Information Theory*, 2018.

Blake E. Woodworth and Nati Srebro. Tight complexity bounds for optimizing composite objectives. In *Advances in neural information processing systems*, pages 3639–3647, 2016.

L. Zhang, M. Mahdavi, and R. Jin. Linear convergence with condition number independent access of full gradients. In *Advances in Neural Information Processing Systems*, 2013.