



NYU

COURANT INSTITUTE OF  
MATHEMATICAL SCIENCES

# MATHEMATICS OF DEEP LEARNING

---

JOAN BRUNA , CIMS + CDS, NYU, SPRING'18

*Lecture 1: The Curse of Dimensionality*

# COURSE OVERVIEW

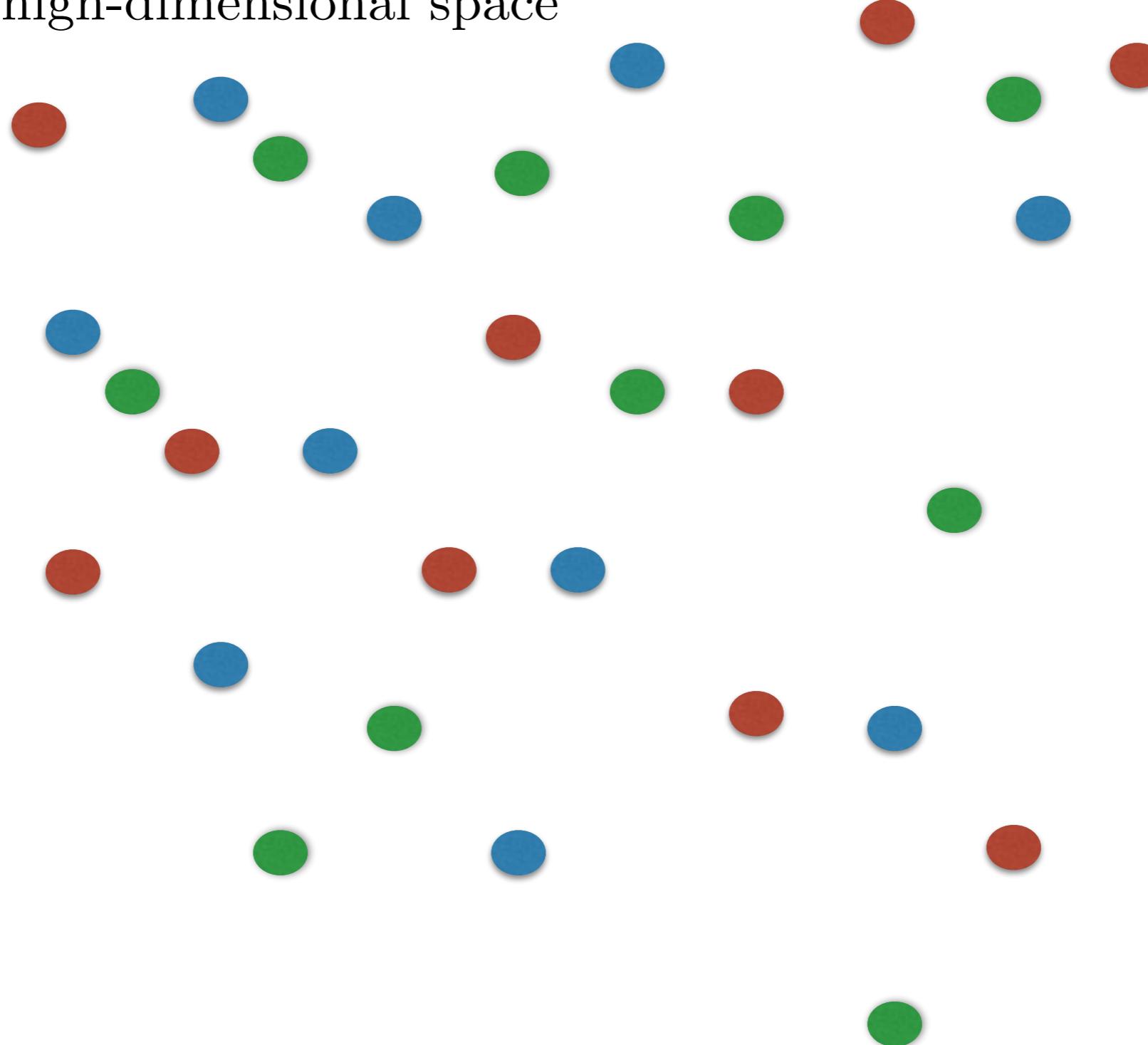
---

- Part I: Geometry of Data
- Part II: Geometry of Optimization and Generalization.
- Parallel Curricula [Optional, TA: Cinjon Resnik]
  - Focus on RL

# PART I: GEOMETRY OF DATA

---

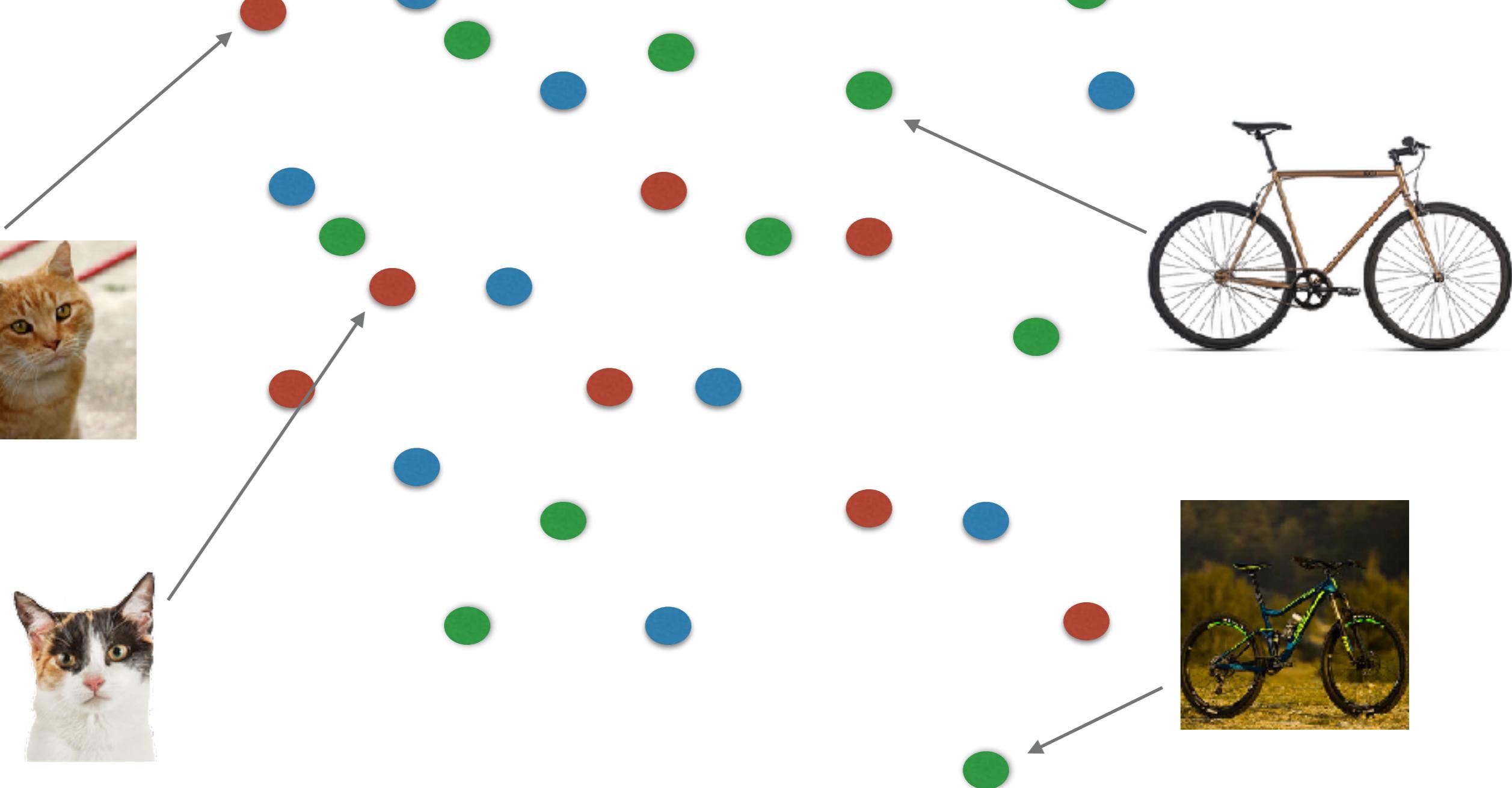
high-dimensional space



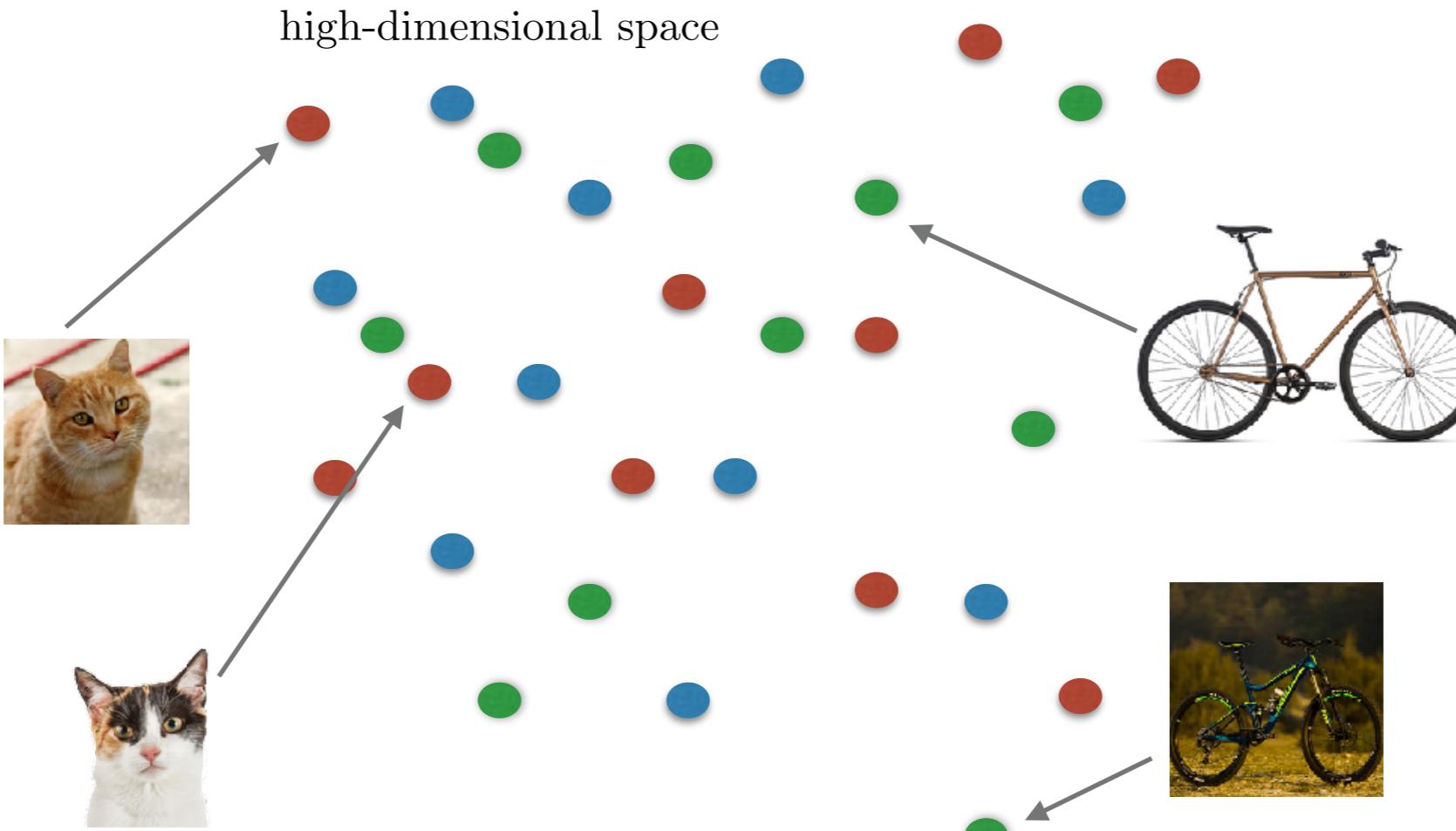
# PART I: GEOMETRY OF DATA

---

high-dimensional space



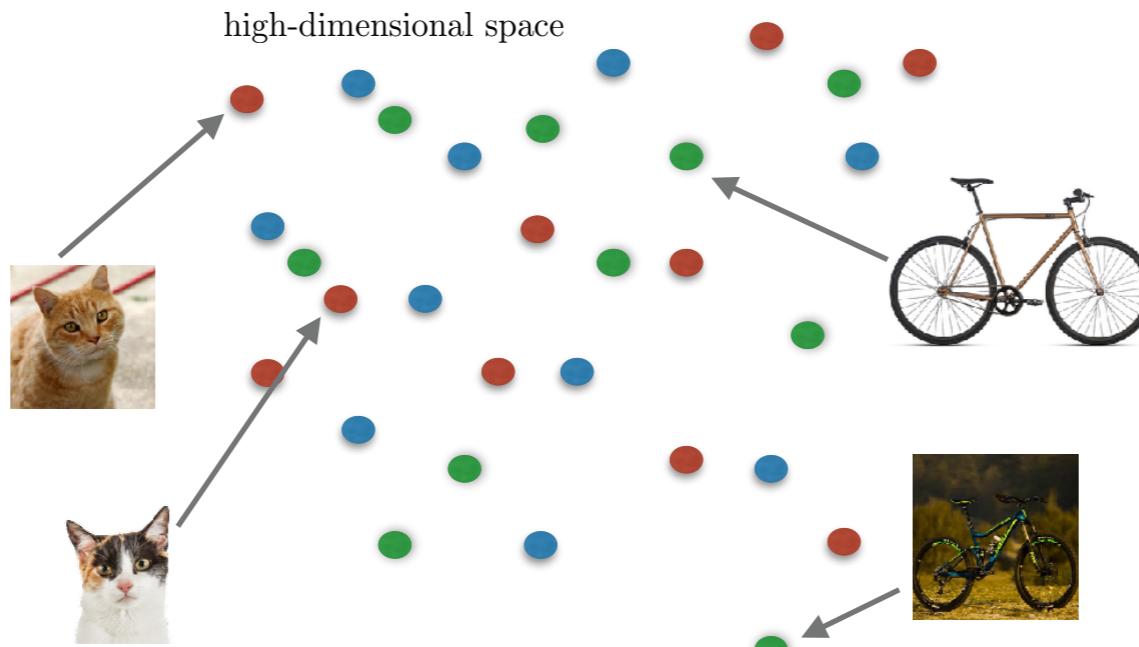
# PART I: GEOMETRY OF DATA



- Learning can be seen as a high-dimensional interpolation problem: estimate unknown function  $f : \mathcal{X} \subseteq \mathbb{R}^d \rightarrow \mathcal{Y}$  given its values on  $n$  training points
$$\{(x_1, f(x_1)), \dots, (x_n, f(x_n))\}$$

# PART I: GEOMETRY OF DATA

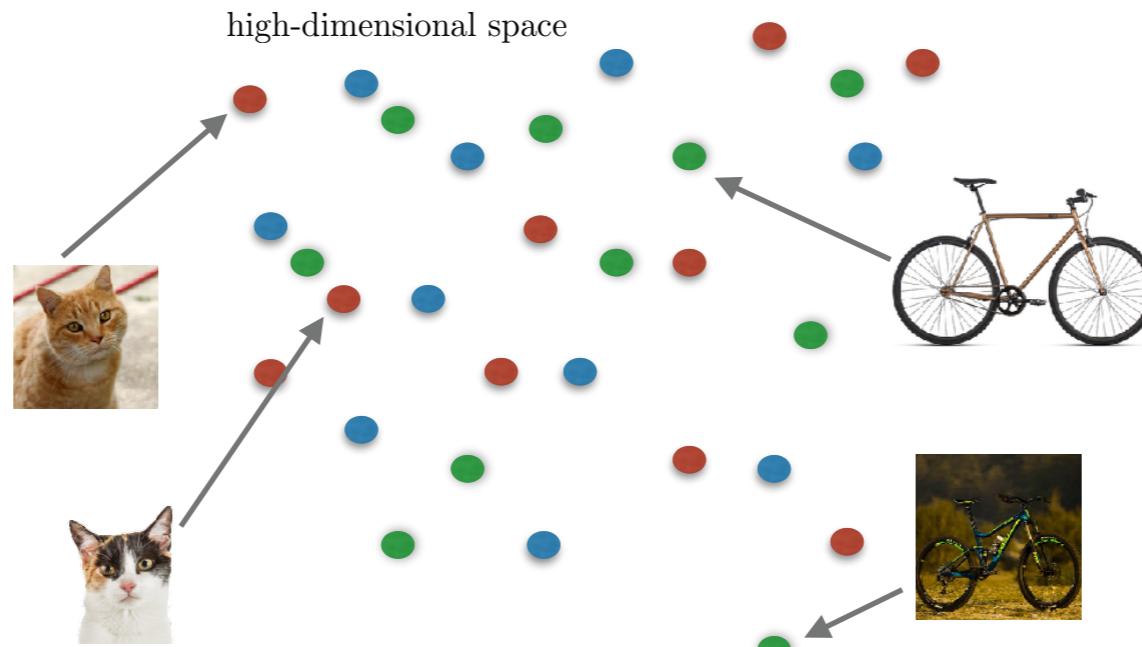
---



- Learning can be seen as a high-dimensional interpolation problem: estimate unknown function  $f : \mathcal{X} \subseteq \mathbb{R}^d \rightarrow \mathcal{Y}$  given its values on  $n$  training points  
$$\{(x_1, f(x_1)), \dots, (x_n, f(x_n))\}$$
- This problem is ill-defined without assumptions on  $f$ .

# PART I: GEOMETRY OF DATA

---



- Learning can be seen as a high-dimensional interpolation problem: estimate unknown function  $f : \mathcal{X} \subseteq \mathbb{R}^d \rightarrow \mathcal{Y}$  given its values on  $n$  training points  
 $\{(x_1, f(x_1)), \dots, (x_n, f(x_n))\}$
- This problem is ill-defined without assumptions on  $f$ .
  - *Which assumptions?*

# PART I: GEOMETRY OF DATA

---

- Regularity of  $f$  is defined relative to a distance in  $\mathcal{X}$  and  $\mathcal{Y}$ .
- E.g.,  $f$  is Lipschitz with respect to distances  $d_{\mathcal{X}}, d_{\mathcal{Y}}$  if

$$\exists C \text{ s.t } \forall x, x' , d_{\mathcal{Y}}(f(x), f(x')) \leq C d_{\mathcal{X}}(x, x') .$$

# PART I: GEOMETRY OF DATA

---

► Regularity of  $f$  is defined relative to a distance in  $\mathcal{X}$  and  $\mathcal{Y}$ .

► E.g.,  $f$  is Lipschitz with respect to distances  $d_{\mathcal{X}}, d_{\mathcal{Y}}$  if

$$\exists C \text{ s.t } \forall x, x' , d_{\mathcal{Y}}(f(x), f(x')) \leq C d_{\mathcal{X}}(x, x') .$$

► Take standard Euclidean metric and assume real outputs:

$$|f(x) - f(x')| \leq C \|x - x'\| .$$

► How many points  $N(\epsilon)$  do we need to observe to guarantee that

$$|\hat{f}(x) - f(x)| \leq \epsilon ?$$

# PART I: GEOMETRY OF DATA

---

- Regularity of  $f$  is defined relative to a distance in  $\mathcal{X}$  and  $\mathcal{Y}$ .
- E.g.,  $f$  is Lipschitz with respect to distances  $d_{\mathcal{X}}, d_{\mathcal{Y}}$  if
$$\exists C \text{ s.t } \forall x, x' , d_{\mathcal{Y}}(f(x), f(x')) \leq C d_{\mathcal{X}}(x, x') .$$
- Take standard Euclidean metric and assume real outputs:
$$|f(x) - f(x')| \leq C \|x - x'\| .$$
- How many points  $N(\epsilon)$  do we need to observe to guarantee that
$$|\hat{f}(x) - f(x)| \leq \epsilon ?$$
- This is equivalent to finding an “ $\epsilon$ -covering” of the data. If we simply assume compactness (e.g. finite energy), then  $N(\epsilon) \simeq \epsilon^{-d} .$

# PART I: GEOMETRY OF DATA

---

- This is an instance of the curse of dimensionality
  - more on that in today's lecture.
- Thus, we need further regularity priors on the class of functions (=tasks) we are interested in.

# PART I: GEOMETRY OF DATA

---

- This is an instance of the curse of dimensionality
  - more on that in today's lecture.
- Thus, we need further regularity priors on the class of functions (=tasks) we are interested in.
- In typical deep learning applications, input space  $\mathcal{X}$  is
  - Images / Videos:  $\mathcal{X} = L^2(\Omega)$  ,  $\Omega$  = Grid of 2d, 3d pixels.
  - Time-series:  $\mathcal{X} = L^2(\mathbb{R})$
  - Text:  $\mathcal{X}$  = sequences over discrete alphabet.
- Are there distances in  $\mathcal{X}$  that encode regularity of typical tasks  $f$  ?

# PART I: GEOMETRY OF DATA

---

- We will introduce the notion of geometric stability
- In a vector space, not a lot of operations defined (only linear operations in fact).
- In a functional space, we can exploit much more structure
  - for instance, we can consider changes of variable given by geometric transformations (eg translation, rotations).

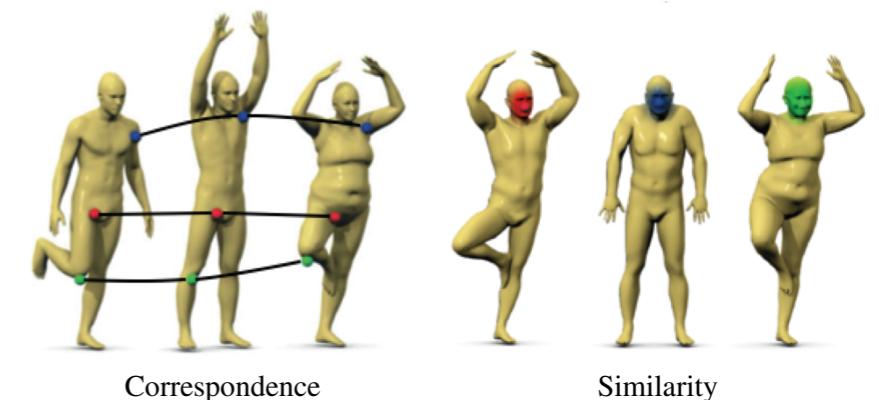
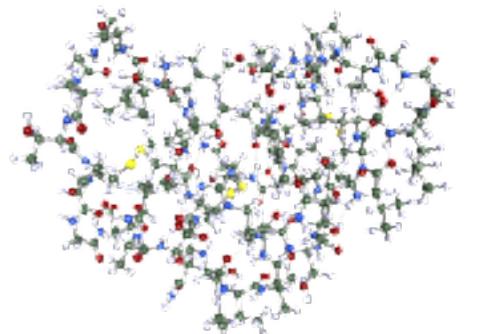
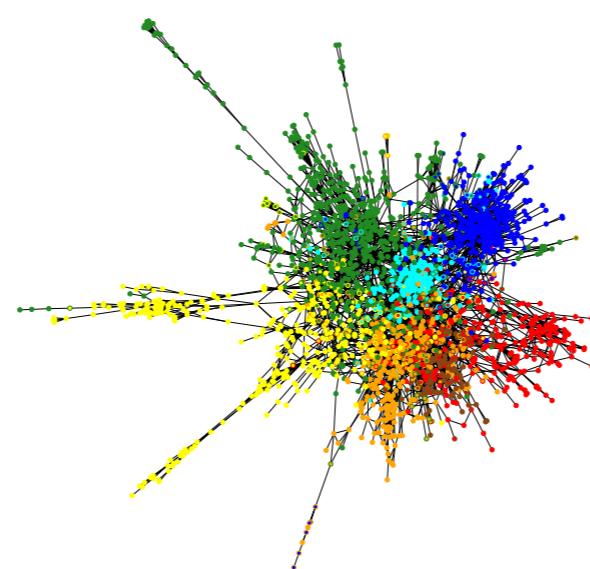


- it turns out these are really meaningful in many machine learning contexts.

# PART I: GEOMETRY OF DATA

---

- How to properly define geometric stability?
- How to exploit it with appropriate representations?
  - As it turns out, CNNs are one possibility.
- Supervised and unsupervised learning under geometric stability.
  
- How to generalize the notion of geometric stability to more general domains?
  - Non-Euclidean geometry
  - Graphs.
  - Applications



## PART II: GEOMETRY OF OPTIMIZATION AND GENERALIZATION

---

- Essentially all deep learning models are trained with some form of Stochastic Gradient Descent.
- Convergence properties of SGD and its variants (momentum, adam, adagrad,etc. )
- Can we learn something about SGD by studying continuous time optimization?
- Langevin Dynamics, Fokker-Plank, Stochastic Differential Equations.

## PART II: GEOMETRY OF OPTIMIZATION AND GENERALIZATION

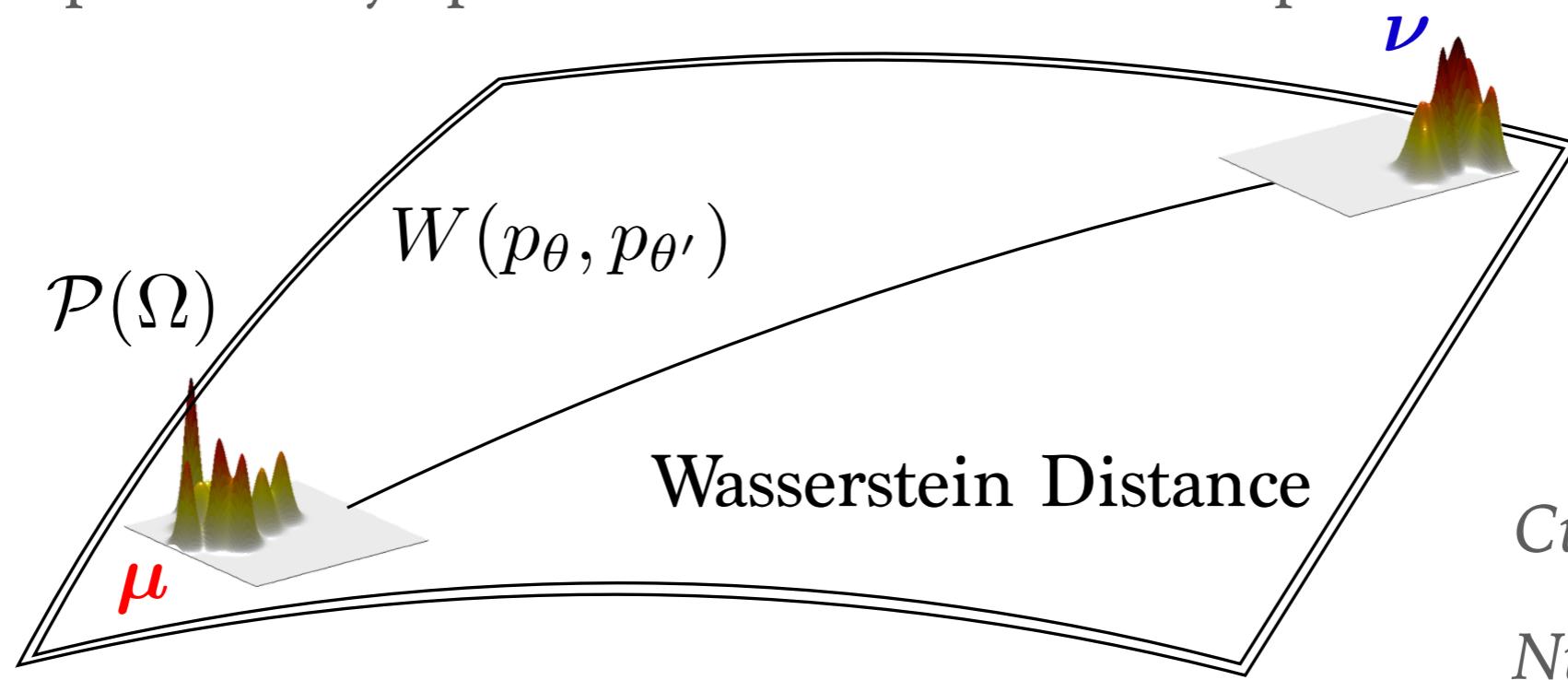
---

- Essentially all deep learning models are trained with some form of Stochastic Gradient Descent.
- Convergence properties of SGD and its variants (momentum, adam, adagrad, etc.)
  - Can we learn something about SGD by studying continuous time optimization?
  - Langevin Dynamics, Fokker-Plank, Stochastic Differential Equations.
- Gradients belong to tangent spaces.
  - In Euclidean spaces, we assimilate the tangent space at each point with the space itself.
  - What is the “right” space where to think about optimization/learning?

# PART II: GEOMETRY OF OPTIMIZATION AND GENERALIZATION

---

- Overview of Information Geometry [Amari, Rao, Kullback]
  - Describes the manifold of probability measure using Fisher metric.
- Reproducing Kernel Hilbert Spaces (RKHS).
  - Enable convex optimization machinery.
- Optimal Transport and Generalization
  - Studies probability spaces defined over metric spaces.

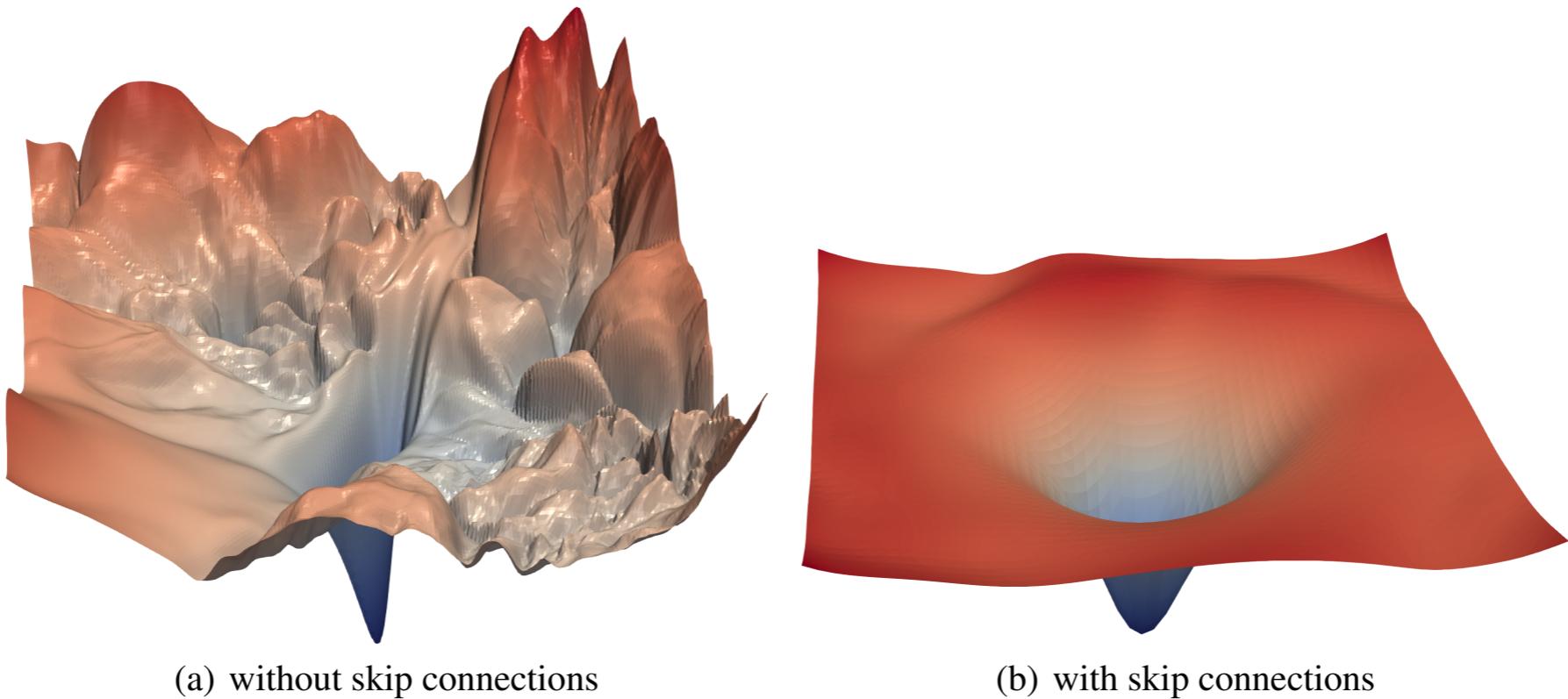


Cuturi & Solomon  
Nips'17 OT

# PART II: GEOMETRY OF OPTIMIZATION AND GENERALIZATION

---

- Optimization Landscape of neural networks.



[Li et al.'17]

- Full of poor local minima?
- Role of overparametrisation/architecture?
- Is this a problem in terms of generalization?

# PARALLEL CURRICULA

---

- Optional.
- Led by Cinjon Resnik.
- Fridays 11am-12:30pm WWH 101
- Mission: “Understand the impactful papers in Machine Learning by going back to their roots, building the tree of dependencies, and envisioning where the leaves will grow.”
  
- We will start with a curriculum targeted to AlphaGo [Deepmind, Nature’16].

# LOGISTICS

---

- Course website:  
<https://joanbruna.github.io/MathsDL-spring18/>
- Office Hours: Tuesdays, 9-11am. Office 612 60 5th ave.

# GRADING

---

- Midterm project checkpoint [25%]
  - Abstract Proposal detailing the specific topic.
  - Due: March 9th.
- Class Participation [25%]
- Final Project [50%]
- Parallel Curricula [+ 25% bonus]

# FINAL PROJECT

---

- In-depth survey article about a selected topic.
- By groups of {1,2} people.
- Project due May 1st, based on feedback from Proposal.
- Similar in depth as:
  - BAIR Research blog. [baseline]
  - Off the convex path/argmin [happy]
  - Distill.pub [enthusiastic]
- At the end of the semester, projects will be integrated into course website.
- Best projects will be encouraged to submit to Distill!

# LECTURE 1: THE CURSE OF DIMENSIONALITY IN ML

---

- Statistical Courses
  - In Supervised Learning.
  - In Unsupervised Learning.
  - Remarkable exception: Principal Component Analysis.
- Learning Courses
  - Gradient-free courses?
  - Gradient-descent failures.
- Conclusion: Learning without Priors?

# WHAT IS THE CURSE OF DIMENSIONALITY?

---

- It refers generally to the intrinsic statistical/computational drama arising when a quantity of interest scales exponentially with the input dimensionality.
  - so, always watch out for dimensions in the exponents!
  - Equivalently, this reflects a bound that depends roughly on the number of “distinct” instances of the problem.
- We will see some examples to familiarize ourselves with it.

# PRELIMINARIES: RATE OF CONVERGENCE/ RATE OF APPROXIMATION

---

- Rates in statistics: As a function of number of (iid) observations, how fast does the estimator approach the desired quantity? in which sense we measure convergence?
- Rates in optimization: How fast does my algorithm converge to a designated solution?
- Rates in approximation theory: approximation error as a function of number of basis elements.

# EMPIRICAL RISK MINIMIZATION: SUPERVISED LEARNING SETUP

---

- Data distribution:  
 $(X, Y) \sim \mathcal{D}$  ,  $X \in \mathbb{R}^d$ ,  $Y = \pm 1$  (binary classification).
- Loss  $\ell(y, f(x))$  (mean-square, hinge, logistic).
- Hypothesis class: Function space  $\mathcal{F} = \{f : \mathcal{X} \rightarrow \mathbb{R}\}$  .

# EMPIRICAL RISK MINIMIZATION: SUPERVISED LEARNING SETUP

---

- Data distribution:

$$(X, Y) \sim \mathcal{D}, \quad X \in \mathbb{R}^d, \quad Y = \pm 1 \text{ (binary classification).}$$

- Loss  $\ell(y, f(x))$  (mean-square, hinge, logistic).

- Hypothesis class: Function space  $\mathcal{F} = \{f : \mathcal{X} \rightarrow \mathbb{R}\}$ .

- Population and Empirical Risk:

$$\mathcal{R}(f) = \mathbb{E}_{\mathcal{D}}[\ell(Y, f(X))] \quad \widehat{\mathcal{D}} = \text{empirical distribution from } n \text{ iid samples.}$$

$$\widehat{\mathcal{R}}(f) = \mathbb{E}_{\widehat{\mathcal{D}}}[\ell(Y, f(X))] = \frac{1}{n} \sum_{i=1}^n \ell(y_i, f(x_i)),$$

- Empirical Risk Minimization:

$$\widehat{f}_\delta = \arg \min_{f \in \mathcal{F}; \|f\|_{\mathcal{F}} \leq \delta} \widehat{\mathcal{R}}(f).$$

# SUPERVISED LEARNING

---

- “Fundamental Theorem of ML”:

$$\mathcal{R}(f) = \underbrace{\widehat{\mathcal{R}}(f)}_{\text{empirical loss}} + \underbrace{[\mathcal{R}(f) - \widehat{\mathcal{R}}(f)]}_{\text{generalization gap}}$$

# SUPERVISED LEARNING

---

- “Fundamental Theorem of ML”:

$$\mathcal{R}(f) = \underbrace{\widehat{\mathcal{R}}(f)}_{\text{empirical loss}} + \underbrace{[\mathcal{R}(f) - \widehat{\mathcal{R}}(f)]}_{\text{generalization gap}}$$

- More precisely, suppose we have approximately solved the ERM:  
 $\hat{f}_\delta \text{ s.t } \widehat{\mathcal{R}}(\hat{f}_\delta) \leq \epsilon + \inf_{\|f\|_{\mathcal{F}} \leq \delta} \widehat{\mathcal{R}}(f)$ .

- Then [Shalev-Shwartz, Ben-David'14]:

$$\mathcal{R}(\hat{f}_\delta) - \inf_{f \in \mathcal{F}} \mathcal{R}(f) \leq \underbrace{\left[ \inf_{\|f\|_{\mathcal{F}} \leq \delta} \mathcal{R}(f) - \inf_{f \in \mathcal{F}} \mathcal{R}(f) \right]}_{\text{approximation}} + \underbrace{2 \sup_{\|f\|_{\mathcal{F}} \leq \delta} |\widehat{\mathcal{R}}(f) - \mathcal{R}(f)|}_{\text{statistical}} + \epsilon.$$

- Approximation error decreases by making  $\mathcal{F}$  larger, but statistical error increases.

# CURSES IN EMPIRICAL RISK MINIMIZATION

---

- Consider two extremes.
- No prior structure:  $\mathcal{F}$  = Lipschitz continuous functions.  
[von Luxburg, Bousquet'04, Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq n^{-\frac{1}{d+3}} \log n .$$

# CURSES IN EMPIRICAL RISK MINIMIZATION

---

- Consider two extremes.
- No prior structure:  $\mathcal{F}$  = Lipschitz continuous functions.  
[von Luxburg, Bousquet'04, Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq n^{-\frac{1}{d+3}} \log n .$$

- A lot of prior structure:  $\mathcal{F}$  = Affine functions.  
[Shalev-Swartz, Ben-David'04, Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq \sqrt{\frac{d}{n}} .$$

# CURSES IN EMPIRICAL RISK MINIMIZATION

---

- Consider two extremes.
- No prior structure:  $\mathcal{F} = \text{Lipschitz continuous functions}$ .  
[von Luxburg, Bousquet'04, Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq n^{-\frac{1}{d+3}} \log n .$$

- A lot of prior structure:  $\mathcal{F} = \text{Affine functions}$ .  
[Shalev-Swartz, Ben-David'04, Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq \sqrt{\frac{d}{n}} .$$

- Less structure:  $\mathcal{F} = \text{One-layer ReLU networks with regularization}$ :  
[Bach'17]
- Sample complexity to reach generalization gap  $\epsilon$ ?

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq k \sqrt{\frac{d}{n}} .$$

$k$ : size of hidden layer.

# CURSES IN EMPIRICAL RISK MINIMIZATION

---

- Consider two extremes.
- No prior structure:  $\mathcal{F} = \text{Lipschitz continuous functions}$ .  
[von Luxburg, Bousquet'04, Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq n^{-\frac{1}{d+3}} \log n .$$

- A lot of prior structure:  $\mathcal{F} = \text{Affine functions}$ .  
[Shalev-Swartz, Ben-David'04, Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq \sqrt{\frac{d}{n}} .$$

- Less structure:  $\mathcal{F} = \text{One-layer ReLU networks with regularization}$ :  
[Bach'17]

$$|\mathcal{R}(f) - \hat{\mathcal{R}}(f)| \simeq k \sqrt{\frac{d}{n}} .$$

$k$ : size of hidden layer.

- Sample complexity to reach generalization gap  $\epsilon$ ?  
 $\Omega(\epsilon^{-d})$  vs  $\Omega(d/\epsilon^2)$ !

# UNSUPERVISED LEARNING

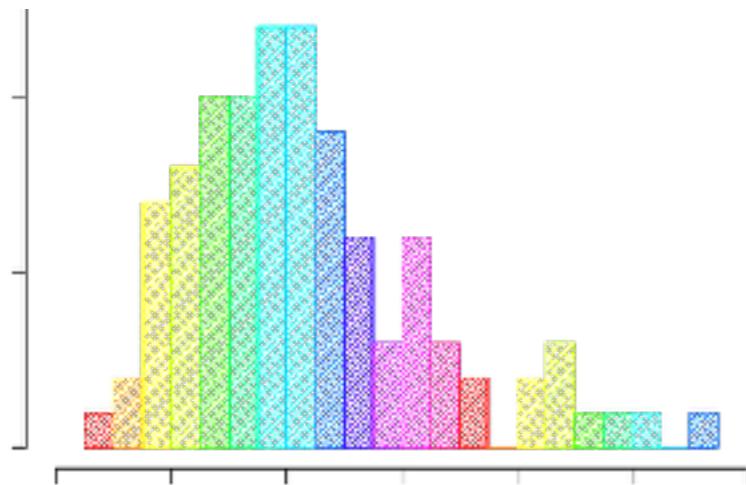
---

- Density estimation: Given *iid* samples  $x_1, \dots, x_n$  from some distribution  $\mathcal{D}$  in  $\mathbb{R}^d$ , estimate  $\mathcal{D}$ .

# UNSUPERVISED LEARNING

---

- Density estimation: Given *iid* samples  $x_1, \dots, x_n$  from some distribution  $\mathcal{D}$  in  $\mathbb{R}^d$ , estimate  $\mathcal{D}$ .
- Ex: Histogram estimation in 1D: If  $\mathcal{D}$  admits a density  $p(x)$



Kernel Density Estimator:

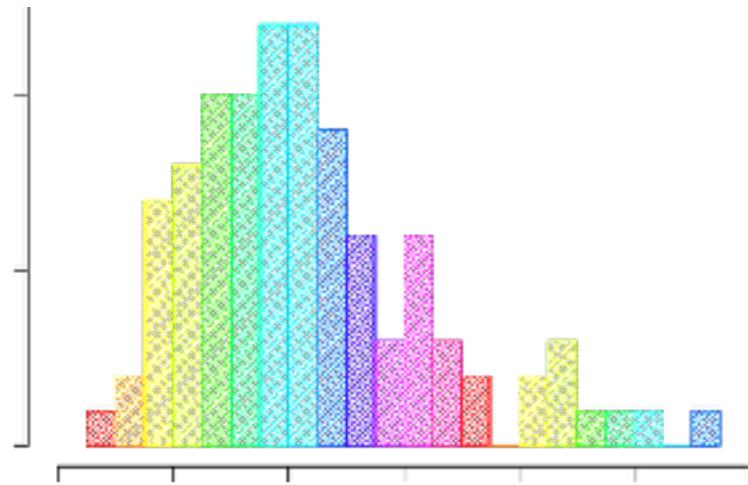
$$\hat{p}_\sigma(x) = \frac{1}{n} \sum_{i=1}^n \phi_\sigma(x - x_i)$$

$\sigma$ : bandwidth parameter

# UNSUPERVISED LEARNING

---

- Density estimation: Given *iid* samples  $x_1, \dots, x_n$  from some distribution  $\mathcal{D}$  in  $\mathbb{R}^d$ , estimate  $\mathcal{D}$ .
- Ex: Histogram estimation in 1D: If  $\mathcal{D}$  admits a density  $p(x)$



Kernel Density Estimator:

$$\hat{p}_\sigma(x) = \frac{1}{n} \sum_{i=1}^n \phi_\sigma(x - x_i)$$

$\sigma$ : bandwidth parameter

- $\sigma$  controls the bias-variance tradeoff of the estimator: typical convergence results are of the form

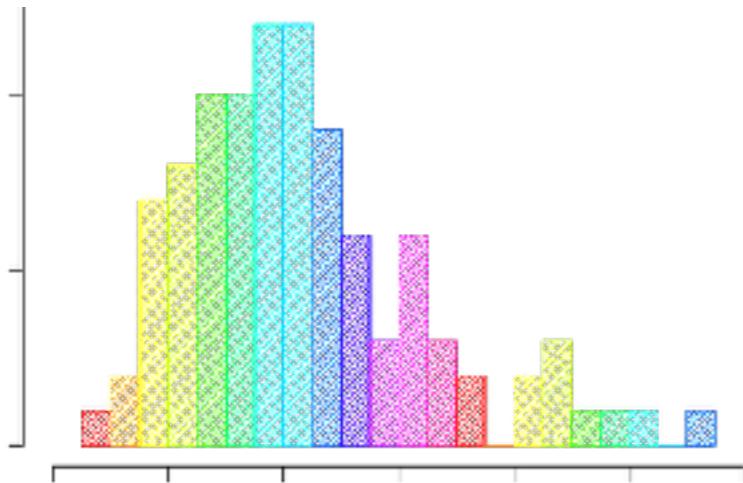
$$\sup_{x \in \mathbb{R}^d} |\hat{p}_\sigma(x) - p(x)| \lesssim \sigma^\alpha + \sqrt{\frac{\log n}{n\sigma^d}} . \text{[Tsybakov'08, Jiang'17]}$$

(assuming  $p$  is  $\alpha$ -Holder continuous).

# UNSUPERVISED LEARNING

---

- Density estimation: Given *iid* samples  $x_1, \dots, x_n$  from some distribution  $\mathcal{D}$  in  $\mathbb{R}^d$ , estimate  $\mathcal{D}$ .
- Ex: Histogram estimation in 1D: If  $\mathcal{D}$  admits a density  $p(x)$



Kernel Density Estimator:

$$\hat{p}_\sigma(x) = \frac{1}{n} \sum_{i=1}^n \phi_\sigma(x - x_i)$$

$\sigma$ : bandwidth parameter

- $\sigma$  controls the bias-variance tradeoff of the estimator: typical convergence results are of the form

$$\sup_{x \in \mathbb{R}^d} |\hat{p}_\sigma(x) - p(x)| \lesssim \sigma^\alpha + \sqrt{\frac{\log n}{n\sigma^d}} . \text{[Tsybakov'08, Jiang'17]}$$

(assuming  $p$  is  $\alpha$ -Holder continuous).

- Leads to a rate

$$|\hat{p} - p|_\infty \simeq n^{-\alpha/(2\alpha+d)}$$

# UNSUPERVISED LEARNING USING OPTIMAL TRANSPORT

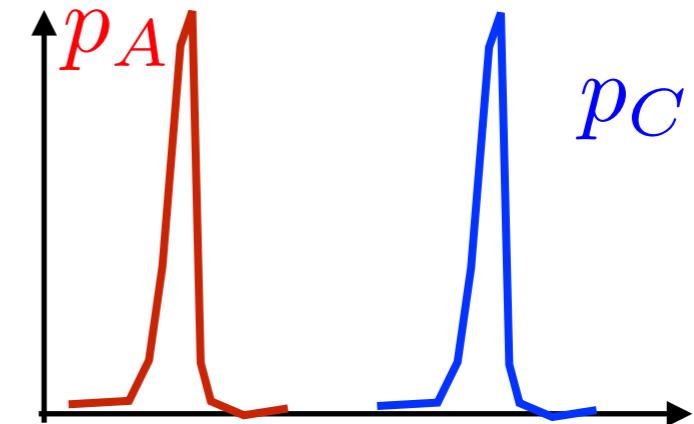
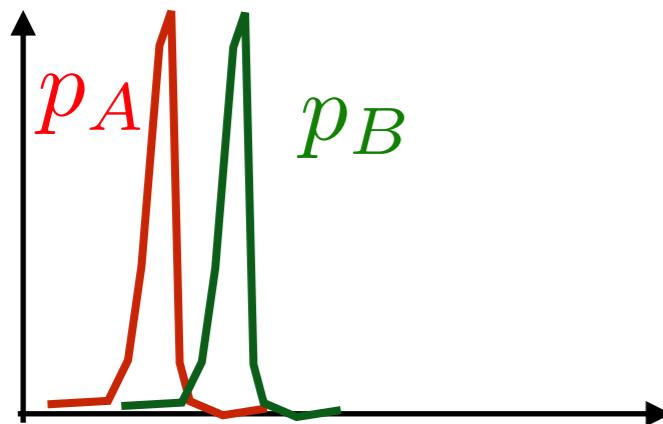
---

- Perhaps issue with metric? TV, MSE all produce similar rates.
- Recently, interest in doing unsupervised learning with *transportation distances*.
- Do they also suffer from curse of dimensionality?

# OPTIMAL TRANSPORT (SNEAK PREVIEW)

---

- Consider two measures  $\mu, \nu$  defined on a metric space  $\Omega$ .



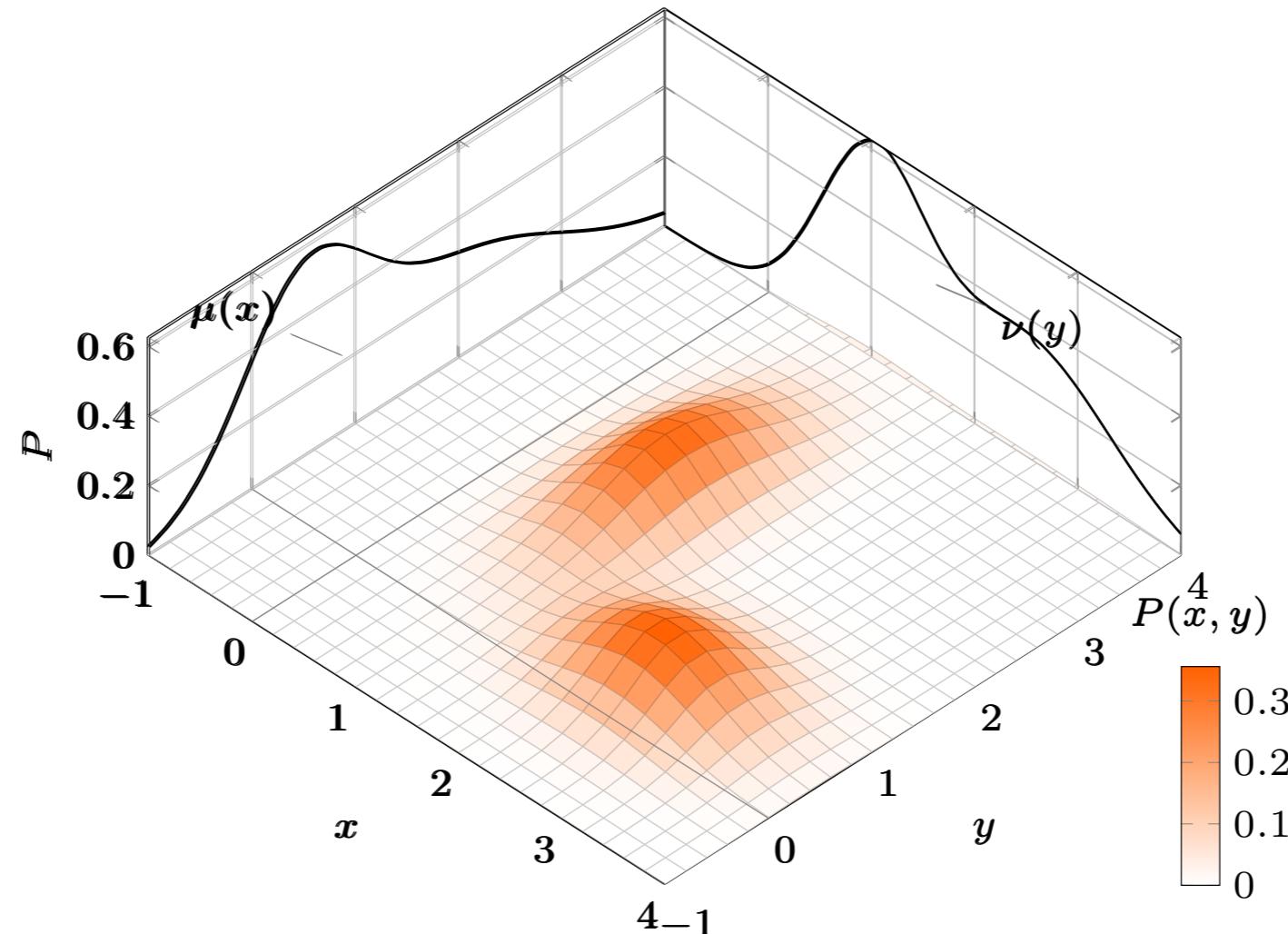
$$\text{MSE}(p_A, p_B) \approx \text{MSE}(p_A, p_C) = O(1)$$

- Can we find a distance between measures that takes the metric structure of  $\Omega$  into account?

# OPTIMAL TRANSPORT (SNEAK PREVIEW)

---

- Consider two measures  $\mu, \nu$  defined on a metric space  $\Omega$ .
- Consider  $\Pi(\mu, \nu) = \{P \in \mathcal{P}(\Omega \times \Omega) ; \forall A, B \subset \Omega,$   
 $P(A \times \Omega) = \mu(A), P(\Omega \times B) = \nu(B)\}$ .



[from NIPS'17 OT tutorial, Cuturi & Solomon]

# OPTIMAL TRANSPORT (SNEAK PREVIEW)

---

- Consider two measures  $\mu, \nu$  defined on a *metric space*  $\Omega$ .

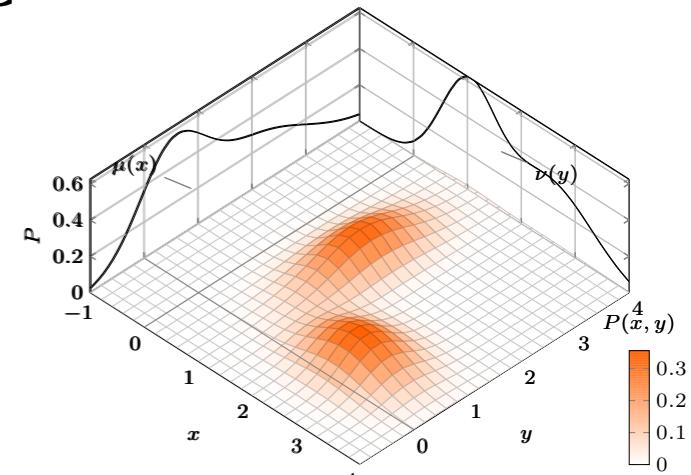
- Consider

$$\Pi(\mu, \nu) = \{P \in \mathcal{P}(\Omega \times \Omega) ; \forall A, B \subset \Omega,$$

$$P(A \times \Omega) = \mu(A), P(\Omega \times B) = \nu(B)\}$$
.

- The  $p$ -Wasserstein distance ( $p \geq 1$ ) is

$$W_p^p(\mu, \nu) = \inf_{P \in \Pi(\mu, \nu)} \mathbb{E}_{(X, Y) \sim P} \text{dist}_\Omega(X, Y)$$



# OPTIMAL TRANSPORT (SNEAK PREVIEW)

- Consider two measures  $\mu, \nu$  defined on a metric space  $\Omega$ .

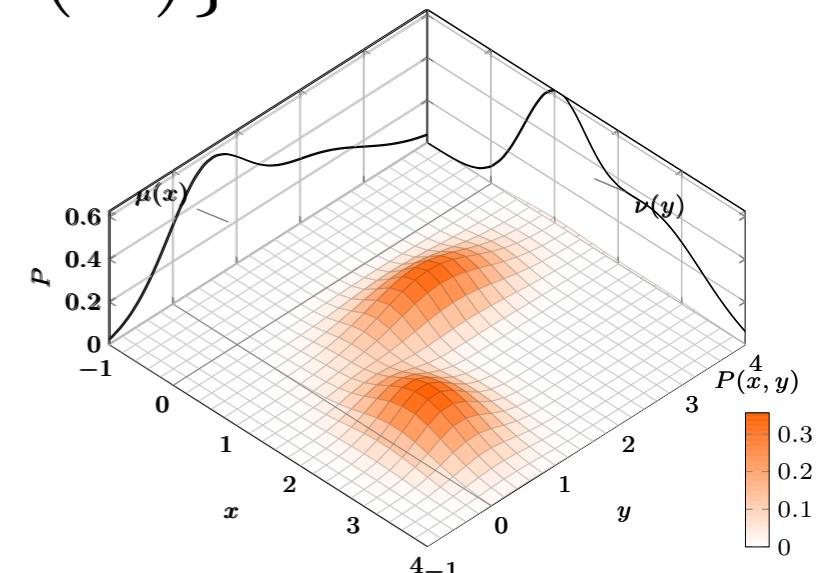
- Consider

$$\Pi(\mu, \nu) = \{P \in \mathcal{P}(\Omega \times \Omega) ; \forall A, B \subset \Omega,$$

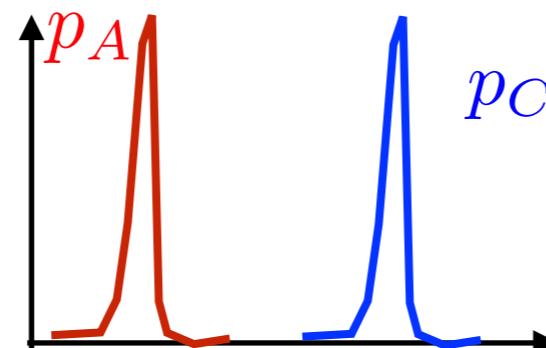
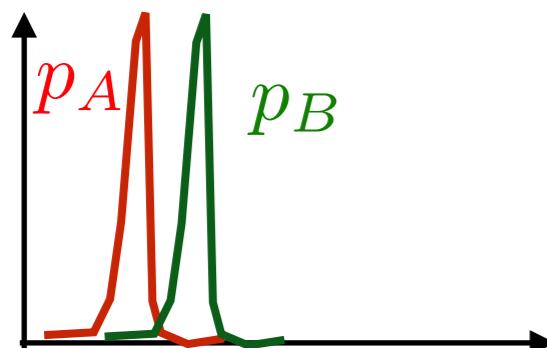
$$P(A \times \Omega) = \mu(A), P(\Omega \times B) = \nu(B)\}$$
.

- The  $p$ -Wasserstein distance ( $p \geq 1$ ) is

$$W_p^p(\mu, \nu) = \inf_{P \in \Pi(\mu, \nu)} \mathbb{E}_{(X, Y) \sim P} \text{dist}_\Omega(X, Y)$$



- When  $p=1$ , it admits a dual formulation.



$$W(p_A, p_B) \ll W(p_A, p_C)$$

# OPTIMAL TRANSPORT IN MACHINE LEARNING

---

- Given “true” measure  $\mu$  and a model  $\nu(\theta)$ ,  $\mu, \nu(\theta) \in \mathcal{P}(\Omega)$ , we would like to find parameters  $\theta$  such that

$$\inf_{\theta} W(\mu, \nu(\theta))$$

- But instead of  $\mu$ , we only have access to empirical measure

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n \delta_{x_i}, \quad x_i \text{ iid } \sim \mu.$$

# OPTIMAL TRANSPORT IN MACHINE LEARNING

---

- Given “true” measure  $\mu$  and a model  $\nu(\theta)$ ,  $\mu, \nu(\theta) \in \mathcal{P}(\Omega)$ , we would like to find parameters  $\theta$  such that

$$\inf_{\theta} W(\mu, \nu(\theta))$$

- But instead of  $\mu$ , we only have access to empirical measure

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n \delta_{x_i}, \quad x_i \text{ iid } \sim \mu.$$

- Again, we face a “bias-variance” tradeoff:

$$W(\mu, \nu(\theta)) \leq \underbrace{W(\mu, \hat{\mu}_n)}_{\text{statistical error}} + \underbrace{W(\hat{\mu}_n, \nu(\theta))}_{\text{approximation error}}.$$

# OPTIMAL TRANSPORT IN MACHINE LEARNING

---

- Bad news [Dudley'68]: If  $\mu$  is absolutely continuous wrt Lebesgue measure on  $\mathbb{R}^d$ , then

$$W_1(\mu, \hat{\mu}_n) \simeq n^{-1/d} .$$

# OPTIMAL TRANSPORT IN MACHINE LEARNING

---

- Bad news [Dudley'68]: If  $\mu$  is absolutely continuous wrt Lebesgue measure on  $\mathbb{R}^d$ , then

$$W_1(\mu, \hat{\mu}_n) \simeq n^{-1/\textcolor{red}{d}}.$$

- Slightly better if we assume that the intrinsic dimensionality of  $\mu$  is  $s \ll d$  [Weed & Bach'17]:

$$W_p(\mu, \hat{\mu}_n) \simeq n^{-1/\textcolor{blue}{s}}.$$

- Lesson: ambient dimension is not necessarily the driving force behind the curse of dimensionality.

# OPTIMAL TRANSPORT IN MACHINE LEARNING

---

- Bad news [Dudley'68]: If  $\mu$  is absolutely continuous wrt Lebesgue measure on  $\mathbb{R}^d$ , then

$$W_1(\mu, \hat{\mu}_n) \simeq n^{-1/\textcolor{red}{d}}.$$

- Slightly better if we assume that the intrinsic dimensionality of  $\mu$  is  $s \ll d$  [Weed & Bach'17]:

$$W_p(\mu, \hat{\mu}_n) \simeq n^{-1/\textcolor{blue}{s}}.$$

- Lesson: ambient dimension is not necessarily the driving force behind the curse of dimensionality.
- Possible alternative: Energy-distances [Sejdinovic et al'12].
- How often can this be assumed in practice?

# COVARIANCE ESTIMATION AND PCA

---

- Principal Component Analysis: ubiquitous data analysis tool for dimensionality reduction, data exploration, etc.
- Given  $\mu \in \mathcal{P}(\mathbb{R}^d)$  with finite energy ( $\mathbb{E}_{X \sim \mu} \|X\|^2 < \infty$ ), it consists in the spectral decomposition of its covariance operator:

$$\Sigma = \mathbb{E}_{X \sim \mu} (X - \mathbb{E}X)(X - \mathbb{E}X)^\top \in \mathbb{R}^{d \times d} .$$

$$\Sigma = U \Lambda U^\top , \quad U^\top U = \mathbf{I}, \quad \Lambda = \text{diag}(\lambda_1, \dots, \lambda_d) .$$

# COVARIANCE ESTIMATION AND PCA

---

- Principal Component Analysis: ubiquitous data analysis tool for dimensionality reduction, data exploration, etc.
- Given  $\mu \in \mathcal{P}(\mathbb{R}^d)$  with finite energy ( $\mathbb{E}_{X \sim \mu} \|X\|^2 < \infty$ ), it consists in the spectral decomposition of its covariance operator:

$$\Sigma = \mathbb{E}_{X \sim \mu} (X - \mathbb{E}X)(X - \mathbb{E}X)^\top \in \mathbb{R}^{d \times d} .$$

$$\Sigma = U \Lambda U^\top , \quad U^\top U = \mathbf{I}, \quad \Lambda = \text{diag}(\lambda_1, \dots, \lambda_d) .$$

- Empirical covariance operator:

$$\Sigma, \widehat{\Sigma}_n \succeq 0 .$$

$$\widehat{\Sigma}_n = \mathbb{E}_{X \sim \hat{\mu}_n} (X - \mathbb{E}X)(X - \mathbb{E}X)^\top = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^\top .$$

# ESTIMATING THE PRINCIPAL COMPONENTS

---

- How good is this estimator? Is it cursed?

# ESTIMATING THE PRINCIPAL COMPONENTS

---

- How good is this estimator? Is it cursed?

**Theorem [Vershynin]:** Let  $X$  be a random vector in  $\mathbb{R}^d$ . Assume for some  $C \geq 1$ ,  $\|X\|_2 \leq C\sqrt{\mathbb{E}\|X\|^2}$  a.s.

Then

$$\mathbb{E}\|\hat{\Sigma}_n - \Sigma\| \lesssim \left( \sqrt{\frac{d \log d}{n}} + \frac{d \log d}{n} \right) \|\Sigma\| .$$

- Sample Complexity: for a desired approximation  $\frac{\mathbb{E}\|\hat{\Sigma}_n - \Sigma\|}{\|\Sigma\|} \leq \epsilon$  we need  $n \sim \epsilon^{-2}d \log d$  samples.
- Lesson: Covariance estimation is legit in high dimensions!

# CURSES IN OPTIMIZATION

---

- The curse of dimensionality not only manifests itself in statistics.
- In complexity theory, it captures EXPTIME/EXPSPACE problems.
- How does it affect our ability to *learn*, ie optimize an objective function?

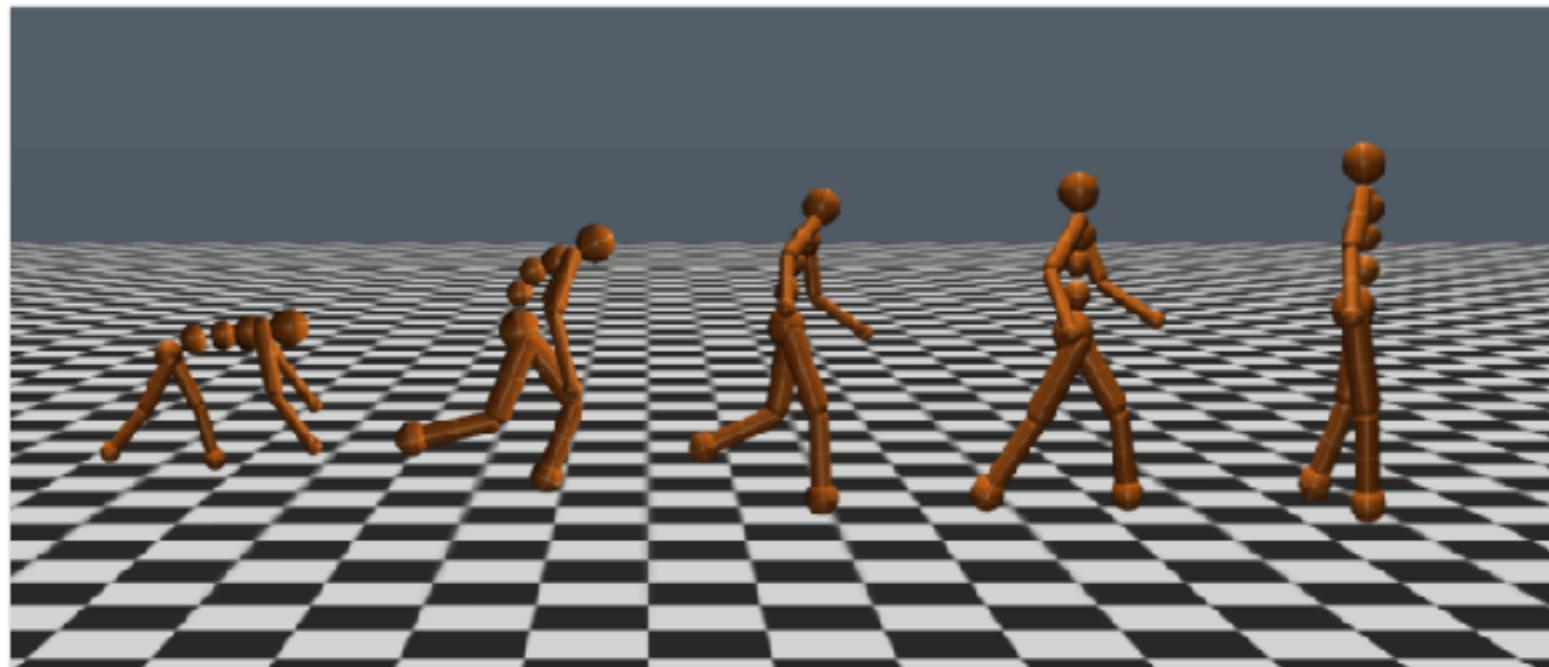
# GRADIENT-FREE OPTIMIZATION

---

## Welcoming the Era of Deep Neuroevolution

By Kenneth O. Stanley & Jeff Clune

December 18, 2017



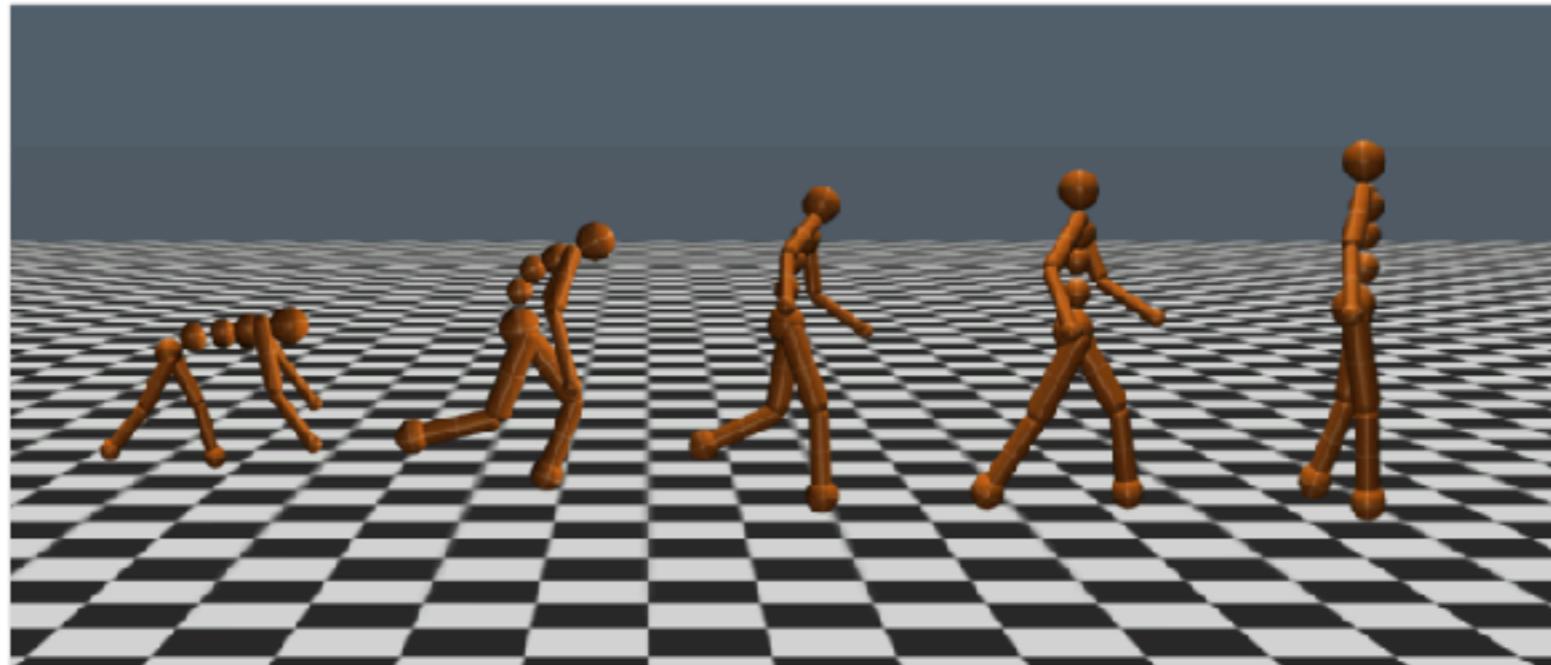
# GRADIENT-FREE OPTIMIZATION

---

## Welcoming the Era of Deep Neuroevolution

By Kenneth O. Stanley & Jeff Clune

December 18, 2017



- What is *Neuroevolution*?  $(F : \mathbb{R}^d \rightarrow \mathbb{R})$
- It is an iterative optimization scheme of the form

$$x_{t+1} = x_t + \frac{\alpha}{2\sigma} [F(x_t + \sigma\beta_t) - F(x_t - \sigma\beta_t)]\beta_t , \quad \beta_t \sim \mathcal{N}(0, I) .$$

# GRADIENT-FREE OPTIMIZATION

---

- It is an iterative optimization scheme of the form

$$x_{t+1} = x_t + \frac{\alpha}{2\sigma} [F(x_t + \sigma\beta_t) - F(x_t - \sigma\beta_t)]\beta_t , \quad \beta_t \sim \mathcal{N}(0, I) .$$

- If  $\sigma$  is small, then the update approximates a (random) directional derivative:

$$\lim_{\sigma \rightarrow 0} \frac{F(x + \sigma\beta) - F(x - \sigma\beta)}{2\sigma} = \langle \nabla F(x), \beta \rangle$$

# GRADIENT-FREE OPTIMIZATION

---

- It is an iterative optimization scheme of the form

$$x_{t+1} = x_t + \frac{\alpha}{2\sigma} [F(x_t + \sigma\beta_t) - F(x_t - \sigma\beta_t)]\beta_t , \quad \beta_t \sim \mathcal{N}(0, I) .$$

- If  $\sigma$  is small, then the update approximates a (random) directional derivative:

$$\lim_{\sigma \rightarrow 0} \frac{F(x + \sigma\beta) - F(x - \sigma\beta)}{2\sigma} = \langle \nabla F(x), \beta \rangle$$

- Thus

$$\mathbb{E}_{\beta_t} \left\{ \frac{1}{2\sigma} [F(x_t + \sigma\beta_t) - F(x_t - \sigma\beta_t)]\beta_t \right\} \rightarrow \mathbb{E} \{ \langle \nabla F(x_t), \beta_t \rangle \beta_t \} = \nabla F(x_t) .$$

- We are doing a stochastic (gradient-free) approximation of the gradient.

# GRADIENT-FREE OPTIMIZATION

---

- It is an iterative optimization scheme of the form

$$x_{t+1} = x_t + \frac{\alpha}{2\sigma} [F(x_t + \sigma\beta_t) - F(x_t - \sigma\beta_t)]\beta_t , \quad \beta_t \sim \mathcal{N}(0, I) .$$

- If  $\sigma$  is small, then the update approximates a (random) directional derivative:

$$\lim_{\sigma \rightarrow 0} \frac{F(x + \sigma\beta) - F(x - \sigma\beta)}{2\sigma} = \langle \nabla F(x), \beta \rangle$$

- Thus

$$\mathbb{E}_{\beta_t} \left\{ \frac{1}{2\sigma} [F(x_t + \sigma\beta_t) - F(x_t - \sigma\beta_t)]\beta_t \right\} \rightarrow \mathbb{E} \{ \langle \nabla F(x_t), \beta_t \rangle \beta_t \} = \nabla F(x_t) .$$

- We are doing a stochastic (gradient-free) approximation of the gradient.
- What is the price to pay of not using gradient information?

# GRADIENT-FREE IS NOT TOTALLY CURSED...

---

- It turns out [Nesterov, Spokoiny'10] that
  - Using directional derivatives ( $\sigma \rightarrow 0$ ) requires at most  $O(d)$  times more iterations than standard sub gradient method.
  - Using finite-differences, we require  $O(d^2)$  times more iterations in general.
- ... but in many DL tasks,  $d \sim 10^8$ !
- That said, such methods can be vastly parallelized.

# CURSE OF GRADIENT DESCENT

---

- Consider a supervised learning setting where labels  $y = h(x)$  are generated according to some hypothesis  $h \in \mathcal{H}$ .
- Gradient descent learning attempts to minimize

$$\min_{\theta} \mathcal{L}_h(\theta) = \mathbb{E}_x \{ \ell(\Phi_\theta(x), h(x)) \}$$

using gradient information  $\nabla_{\theta} \mathcal{L}_h(\theta)$  .

# CURSE OF GRADIENT DESCENT

---

- Consider a supervised learning setting where labels  $y = h(x)$  are generated according to some hypothesis  $h \in \mathcal{H}$ .
- Gradient descent learning attempts to minimize

$$\min_{\theta} \mathcal{L}_h(\theta) = \mathbb{E}_x \{\ell(\Phi_\theta(x), h(x))\}$$

using gradient information  $\nabla_{\theta} \mathcal{L}_h(\theta)$  .

- Consider the variance of that gradient wrt the hypothesis class:
$$\text{Var}(\mathcal{H}, \mathcal{L}, \theta) = \mathbb{E}_h \|\nabla \mathcal{L}_h(\theta) - \mathbb{E}_{h'} \nabla \mathcal{L}_{h'}(\theta)\|^2$$
- It measures how much information gradient conveys on target function  $h \in \mathcal{H}$ .

# CURSE OF GRADIENT DESCENT

---

- Theorem [Shalev-Shwartz et al.'17]: If
  - $\mathcal{H}$  consists of real functions  $\mathbb{E}_x \|h(x)\|^2 \leq 1$ , such that  $\mathbb{E}_x\{h(x)h'(x)\} = 0$  for  $h \neq h' \in \mathcal{H}$ .
  - $\Phi_\theta(x)$  satisfies  $\mathbb{E}_x\{\|\nabla_\theta \Phi_\theta(x)\|^2\} \leq G(\theta)^2$  for some scalar  $G(\theta)$ .
  - $\ell$  is either square loss or binary classification Lipschitz loss.

Then  $\text{Var}(\mathcal{H}, \mathcal{L}, \theta) \leq \frac{G(\theta)^2}{|\mathcal{H}|}$ .

# CURSE OF GRADIENT DESCENT

---

- Theorem [Shalev-Shwartz et al.'17]: If
  - $\mathcal{H}$  consists of real functions  $\mathbb{E}_x \|h(x)\|^2 \leq 1$ , such that  $\mathbb{E}_x\{h(x)h'(x)\} = 0$  for  $h \neq h' \in \mathcal{H}$ .
  - $\Phi_\theta(x)$  satisfies  $\mathbb{E}_x\{\|\nabla_\theta \Phi_\theta(x)\|^2\} \leq G(\theta)^2$  for some scalar  $G(\theta)$ .
  - $\ell$  is either square loss or binary classification Lipschitz loss.
- Then  $\text{Var}(\mathcal{H}, \mathcal{L}, \theta) \leq \frac{G(\theta)^2}{|\mathcal{H}|}$ .
- When learning from a large collection of uncorrelated functions, sensitivity of the gradient to the target function decreases linearly with  $|\mathcal{H}|$ .
- How does that relate to dimensionality of the problem?

# CURSE OF GRADIENT DESCENT

---

- Let's illustrate it with the example of learning *parity functions*:

$$\mathcal{H} = \{x \mapsto (-1)^{\langle x, v \rangle} ; x, v \in \{0, 1\}^d\} .$$

- $|\mathcal{H}| = 2^d$ , and
- $\mathbb{E}_x\{h(x)h'(x)\} = \mathbb{E}_x\{(-1)^{\langle x, v+v' \rangle}\} = \prod \mathbb{E}_{x_i}(-1)^{x_i} = \delta(v - v') .$
- It follows that  $\text{Var}(\mathcal{H}, \mathcal{L}, \theta) \leq \frac{G(\theta)^2}{2^d} \cdot \sum_{v_i \neq v'_i}$ .
- By Chebyshev inequality, the gradient at any point  $\theta$  will concentrate independently of  $v$ .
- This is regardless of which class of predictors.
- Hard to learn with gradient descent, but easy by solving a linear system! (albeit in  $\mathbb{Z}_2$ ).