Artificial Intelligence
Unsupervised Learning
and Causation

LÉON BOTTOU
FACEBOOK AI RESEARCH & NEW YORK UNIVERSITY
Facebook AI Research

- Mission: advancing the field of machine intelligence.
- Open research: we publish papers, we
- More than a hundred permanents and growing.
- New York, Menlo Park, Paris, Montreal, Seattle, ...

https://research.fb.com/category/facebook-ai-research-fair/
https://github.com/facebookresearch
Summary

1. What works in AI?
2. What does not work?
3. A detective guesswork?
4. Causation hints
5. Causal direction
6. Causation and unsupervised learning
7. Wasserstein GANs
8. The geometry of weak Integral Probability Metrics

Conclusion
1- What works in AI?

WHY ARE PEOPLE SO EXCITED?
Object recognition in images

Top5 error rate of the annual winner of the ImageNet image classification challenge. CNNs break through in 2012.
Object recognition in images

Such systems are really being used on a large scale.
Object recognition in images

Detectron

Detectron is Facebook AI Research's software system that implements state-of-the-art object detection algorithms, including Mask R-CNN. It is written in Python and powered by the Caffe2 deep learning framework.

At FAIR, Detectron has enabled numerous research projects, including: Feature Pyramid Networks for Object Detection, Mask R-CNN, Detecting and Recognizing Human-Object Interactions, Focal Loss for Dense Object Detection, Non-local Neural Networks, Learning to Segment Every Thing, and Data Distillation: Towards Omni-Supervised Learning.
Speech recognition

Comparison (2012) of the word error rates achieved by traditional GMMs and DNNs, reported by three different research groups on three different benchmark.
Speech recognition

wav2letter

wav2letter is a simple and efficient end-to-end Automatic Speech Recognition (ASR) system from Facebook AI Research. The original authors of this implementation are Ronan Collobert, Christian Puhrsch, Gabriel Synnaeve, Neil Zeghidour, and Vitaliy Liptchinsky.

wav2letter implements the architecture proposed in Wav2Letter: an End-to-End ConvNet-based Speech Recognition System and Letter-Based Speech Recognition with Gated ConvNets.

If you want to get started transcribing speech right away, we provide pre-trained models for the Librispeech dataset.

Papers

Our approach is detailed in two scientific contributions:
Machine translation

Although it is far from perfect (more on this later), Wikipedia says:

Usage [edit]

Introduction

FAIR Sequence-to-Sequence Toolkit (PyTorch)

This is a PyTorch version of fairseq, a sequence-to-sequence learning toolkit from Facebook AI Research. The original authors of this reimplementation are (in no particular order) Sergey Edunov, Myle Ott, and Sam Gross. The toolkit implements the fully convolutional model described in Convolutional Sequence to Sequence Learning and features GPU training on a single machine as well as fast beam search generation on both CPU and GPU. We provide pre-trained models for English to French and English to German translation.
Reinforcement learning in games

- TD-Gammon (Tesauro, 1992-1995)

  - Trained by self play.
  - Arguably the best backgammon player in the world.
Reinforcement learning in games

- AlphaGo (Deepmind)
  - Trained with self-play.
  - Arguably the best Go player in the world.
  - Go is more complex than backgammon.
  - Go still is a full information game.
  - Go games can be simulated at high speed (unlike self-driving cars.)
Reinforcement learning in games

- DeepStack (Moravčík et al., 2017)

THE FIRST COMPUTER PROGRAM TO OUTPLAY HUMAN PROFESSIONALS AT HEADS-UP NO-LIMIT HOLD’EM POKER

In a study completed December 2016 and involving 44,000 hands of poker, DeepStack defeated 11 professional poker players with only one outside the margin of statistical significance. Over all games played, DeepStack won 49 big blinds/100 (always folding would only lose 75 bb/100), over four standard deviations from zero, making it the first computer program to beat professional poker players in heads-up no-limit Texas hold'em poker.

- No longer a full information game.
- Still can be simulated at high speed.
2- What doesn't work?

WHY ARE PEOPLE TOO EXCITED?
Training demands too much data

- Locating and recognizing objects in images
  after training on more images than a human can see.

- Translating natural languages (somehow)
  after training on more bi-text than a human can read.

- Playing Atari games
  after playing more games than any teenager can endure.

- Playing Go (famously)
  after playing more grandmaster level games than mankind.
Training demands too much data

- Yann LeCun’s chocolate cake
  - In reinforcement learning, the learning algorithm focuses on the reward signal.
  - In supervised learning, the learning algorithm focuses on the manually annotated class labels.
  - But there may be a lot of signal in the patterns themselves.

**Reinforcement Learning (cherry)**
- The machine predicts a scalar reward given once in a while.
- A few bits for some samples

**Supervised Learning (icing)**
- The machine predicts a category or a few numbers for each input
- 10→10,000 bits per sample

**Unsupervised Learning (cake)**
- The machine predicts any part of its input for any observed part.
- Predicts future frames in videos
- Millions of bits per sample
The statistical problem is only a proxy

Example: detection of the action “giving a phone call”

(Oquab et al., CVPR 2014)
~70% correct (SOTA)
The statistical problem is only a proxy

Example: detection of the action “giving a phone call”
The statistical problem is only a proxy

Example: detection of the action “giving a phone call”

The learning algorithm is statistically correct!

In a typical image dataset, when an image shows a person near a phone, chances are that the person is giving a phone call.
The statistical problem is only a proxy

- Strong statistical biases in large datasets often mask the semantics
- Another example: Visual Question Answering

What color is the jacket?
- Red and blue.
- Yellow.
- Black.
- Orange.

How many cars are parked?
- Four.
- Three.
- Five.
- Six.

What event is this?
- A wedding.
- Graduation.
- A funeral.
- A picnic.

When is this scene taking place?
- Day time.
- Night time.
- Evening.
- Morning.

What is covering the ground?
- Snow
- Candies
- Grass
- Refuse
Structure does not help our systems

- **Structure in computer vision**
  - Scenes are made of objects, objects are made of parts,
  - Objects interacts through their parts, ...

- **Structure in natural language**
  - Sentences have a recursive grammatical structure,
  - This structure is associated to meaning.

  Structure allows us to reason.
  This is an important component of our human experience.
Structure does not help our systems

“Adding structural knowledge to machine learning systems should improve the performance!”

- This does not go very far in practice
  - Earlier techniques in computer vision used to recognize objects from their parts, only to be outperformed by convolutional neural networks.
  - For many natural language processing tasks such as document classification, sentiment analysis,* or text tagging, smartly using bags of bigrams give state-of-the-art performance. Their order does not seem to matter!
  - Earlier techniques in machine translation used to leverage the grammatical structure of the sentences, only to be displaced by neural models that only use the sequence of words.

*See for instance (Scheible & Schütze, 2013) https://arxiv.org/abs/1301.2811
Structure does not help our systems

“Adding structural knowledge to machine learning systems should improve the performance!”

- This does not go very far in practice
- Earlier techniques in computer vision used to recognize objects from their parts, only to be outperformed by convolutional neural networks.
- For many natural language processing tasks such as document classification, sentiment analysis,* or text tagging, smartly using bags of bigrams give state-of-the-art performance. Their order does not seem to matter!
- Earlier techniques in machine translation used to leverage the grammatical structure of the sentences, only to be displaced by neural models that only use the sequence of words.

*See for instance (Scheible & Schütze, 2013) https://arxiv.org/abs/1301.2811

“Every time I fire a linguist, the performance of the speech recognizer goes up”
Attributed to F. Jelinek, around 1988.
The FastText story

FastText is an open-source, free, lightweight library that allows users to learn text representations and text classifiers. It works on standard, generic hardware. Models can later be reduced in size to even fit on mobile devices.
What is structure for, exactly?

- This may not be a problem with structure, but a problem with our benchmarking methods.

We usually report an average performance measured on a testing set.

- The average performance emphasizes understanding frequent sentences...

  “How do you do?”

- ...and places little weight on understanding rarer sentences.

  “The bank was about to close when the four masked men showed up.”
What is structure for, exactly?

What is the purpose of the grammatical structure of language

- to help describing the distribution of *existing sentences*?
- to help constructing *new sentences* that describe new situations?
Statistics ≠ Semantics

(Example from D. Hofstader, 2018.)
3- Detective guesswork
The pinboard (1)

Need too much training data

Unexploited signal in data?

Unsupervised learning
The pinboard (2)

- Need too much training data
- Unexploited signal in data?
- Unsupervised learning
- Dataset collection is always biased
- Statistics ≠ semantics
- Structure does seem to help
- What about reasoning?
The pinboard (3)

- Need too much training data
- Unexploited signal in data?
- Unsupervised learning
- Dataset collection is always biased
- Structure helps making sense of things not seen in the datasets?
- Structure does seem to help
- What about reasoning?
- Statistics ≠ semantics

What about reasoning?
Causation

- Need too much training data
- Dataset collection is always biased
- Unexploited signal in data?
- Unsupervised learning
- Structure does seem to help
- What causes what?
- Structure helps making sense of things not seen in the datasets?
- Dataset collection is always biased
- Statistics ≠ semantics
- What about reasoning?
What about these arrows?

- Need too much training data
- Unexploited signal in data?
- Unsupervised learning
- Structure helps making sense of things not seen in the datasets?
- Structure does seem to help
- Dataset collection is always biased
- Statistics ≠ semantics
- What causes what?
- What about reasoning?
4- Causation hints

(LOPEZ-PAZ, NISHIHARA, CHINTALA, SCHÖLKOPF, & BOTTOU - CVPR17)
Causal information in the data distribution?

Causation ≠ Correlations

\[ X = aZ + \mathcal{U}(-s_1, s_1) \quad Y = bZ + cX + \mathcal{U}(-s_2, s_2) \]

\( X \sim \text{Bernoulli}, p = \frac{1}{2} \)
Causal footprints in the XY-scatterplot!
More scatterplots

The Hertzsprung–Russell diagram shows the relationship between the stars’ absolute magnitudes or luminosities versus their stellar classifications or effective temperatures.

Scientists clearly draw causal conclusions from a scatterplot, even when interventions are impossible.
Causal information in the data distribution?

- Observation can lead to causal intuitions.
- We can then apply the scientific method.

How to build an unsupervised learning machine that gets causal intuitions?
5- Causal direction

(LOPEZ-PAZ, NISHIHARA, CHINTALA, SCHÖLKOPF, & BOTTU - CVPR17)
Causal problems with two variables

Given two observed variables $X, Y$

I. Either $X$ causes $Y$,
II. or $Y$ causes $X$,
III. or $X$ and $Y$ have unobserved common causes,
IV. or $X$ and $Y$ are independent.

Let’s focus on causal direction detection (I and II)

Reichenbach
potentially confounding
How does causal direction look like?

In this scatter plot
- X is altitude.
- Y is average temperature.

Does the scatter plot reveal whether
- X causes Y
- or Y causes X?
Footprint example 1 – additive noise

\[ Y = \alpha X + \beta + \text{Noise} \]

Sometimes the high moments (the corners) reveal something.

(PETERS ET AL., 14)
Footprint example 2 -- coincidences

(JANZING ET AL., 2011)
From scatterplot to causation direction

Detecting causation direction at scale

- We could build a long list of causal footprint examples, then decide which example is most appropriate for a given scatterplot, etc.
- Or we can construct a classifier...

(LOPEZ-PAZ, ET AL., 2015)
Featurizing a scatterplot

High moments?
- \( F_{rs} = \frac{1}{m} \sum_{j=1}^{m} x_j^r y_j^s \) for well chosen \( r \) and \( s \).

Reproducing Kernel Hilbert space?
- \( F = \frac{1}{m} \sum_{j=1}^{\infty} \phi(x_j, y_j) \in \mathcal{H}_K \) with \( \langle \phi(.), \phi(.) \rangle_K = K(.,.) \)

Learning the features and the classifier
- \( F_w = \frac{1}{m} \sum_{j=1}^{m} \phi_w(x_j, y_j) \)
Neural Causation Classifier
Training NCC

We do not have access to large causal direction datasets but we can generate artificial scatterplots.

\[ Y = f(X) + v(X)\varepsilon \]

**Step 1 - draw distribution on X**
- Draw \( k \sim \mathcal{U}\{1,2,3,4,5\} \) \( r, s \sim \mathcal{U}[0,5] \)
- Take a mixture of \( k \) Gaussians with \( \mu \sim \mathcal{N}(0, r) \) and \( \sigma \sim \mathcal{N}(0, s) \)
Training NCC

Step 2 - draw mechanism $f$
- Cubic spline with random number of random knots...

Step 3 - draw noise
- Noise $\varepsilon$ is Gaussian with random variance $\sim U[0,5]$
- Function $\nu(X)$ is another cubic spline with random knots.

Step 4 – generate causal scatter plot $X \rightarrow Y$
- Draw $x_j, \varepsilon_j$ then compute $y_j = f(x_j) + \nu(x_j)\varepsilon_j$
- Rescale $x_j, y_j$ to enforce marginal mean 0 and sdev 1
Training NCC

Step 6 – Generate training examples

- Scatterplot \( \{(x_j, y_j)\} \) is associated with target label 1
- Scatterplot \( \{(y_j, x_j)\} \) is associated with target label 0

Repeat 100000 to generate a training set. Train the neural network classifier with the usual bag of tricks. (dropout regularization, rmsprop, cross-validation, etc.)
Sanity check

- After training on artificial data, NCC achieves state-of-the-art [79%] performance on the *Tübingen cause-effect dataset*, which contains 100 cause-effect pairs (https://webdav.tuebingen.mpg.de/cause-effect)

<table>
<thead>
<tr>
<th>Pair</th>
<th>Variable 1</th>
<th>Variable 2</th>
<th>Dataset</th>
<th>Ground Truth</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>pair0001</td>
<td>Altitude</td>
<td>Temperature</td>
<td>D1</td>
<td>→</td>
<td>1/6</td>
</tr>
<tr>
<td>pair0002</td>
<td>Altitude</td>
<td>Precipitation</td>
<td>D1</td>
<td>→</td>
<td>1/6</td>
</tr>
<tr>
<td>pair0003</td>
<td>Longitude</td>
<td>Temperature</td>
<td>D1</td>
<td>→</td>
<td>1/6</td>
</tr>
<tr>
<td>pair0004</td>
<td>Altitude</td>
<td>Sunshine hours</td>
<td>D1</td>
<td>→</td>
<td>1/6</td>
</tr>
<tr>
<td>pair0005</td>
<td>Age</td>
<td>Length</td>
<td>D2</td>
<td>→</td>
<td>1/7</td>
</tr>
<tr>
<td>pair0006</td>
<td>Age</td>
<td>Shell weight</td>
<td>D2</td>
<td>→</td>
<td>1/7</td>
</tr>
<tr>
<td>pair0007</td>
<td>Age</td>
<td>Diameter</td>
<td>D2</td>
<td>→</td>
<td>1/7</td>
</tr>
<tr>
<td>pair0008</td>
<td>Age</td>
<td>Height</td>
<td>D2</td>
<td>→</td>
<td>1/7</td>
</tr>
<tr>
<td>pair0009</td>
<td>Age</td>
<td>Whole weight</td>
<td>D2</td>
<td>→</td>
<td>1/7</td>
</tr>
<tr>
<td>pair0010</td>
<td>Age</td>
<td>Shucked weight</td>
<td>D2</td>
<td>→</td>
<td>1/7</td>
</tr>
<tr>
<td>pair0011</td>
<td>Age</td>
<td>Viscera weight</td>
<td>D2</td>
<td>→</td>
<td>1/7</td>
</tr>
<tr>
<td>pair0012</td>
<td>Age</td>
<td>Wage per hour</td>
<td>D3</td>
<td>→</td>
<td>1/2</td>
</tr>
<tr>
<td>pair0013</td>
<td>Displacement</td>
<td>Fuel consumption</td>
<td>D4</td>
<td>→</td>
<td>1/4</td>
</tr>
<tr>
<td>pair0014</td>
<td>Horse power</td>
<td>Fuel consumption</td>
<td>D4</td>
<td>→</td>
<td>1/4</td>
</tr>
</tbody>
</table>
Counterfactual on images

Asymmetric relation

- How would this image would have looked like if one had removed the cars?
- How would this image would have looked like if one had removed the bridge?

Can we use image datasets to identify the *causal dispositions* of object categories?

How to validate a result?
Causal and anti-causal features

For each object category, we can also define two sets of scene features

- The **causal features** are those that cause the presence of the object of interest. *If the object of interest had not been present in the image, these feature would still have appeared.*

- The **anticausal features** are those that are caused by the presence of the object of interest. *If the object of interest had not been present in the image, these feature would not have appeared.*
Proxy variables & shadow footprints

Assume there is a causal footprint in the distribution of variables that represent the presence of an object or a feature.

We apply NCC to these scores reveals to find out which features are causal or anticausal for each object category.

Same pre-trained NCC for all categories!
Object features and context features

In computer vision, one is often interested in another distinction:

- The **object features** “belong” to the object and are most often activated inside the object bounding box.  
  *Example: car wheels, person eyes, etc.*

- The **context features** are those most often activated outside the bounding box.  
  *Example: road under a car, car shadow*
Results

- Top anticausal features have higher object scores for all twenty categories.
- The probability that this happens for all 20 classes out of chance is $2^{-20} \approx 10^{-6}$. 
Hypotheses

Hypothesis 1. Image datasets carry an observable statistical signal revealing the asymmetric relationship between object categories.

Hypothesis 2. There exists an observable statistical dependence between object features and anticausal features.
More information

- The effect disappears completely if we replace NCC by the correlation coefficient (or its absolute value) between the feature and the category.
- The effect appears to be robust to many details of the experiment such as the precise composition of the NCC data, the precise computation of object/context scores, the methods we use to determine a continuous proxy for the categories, etc.
6- Causation and unsupervised learning

(WITH MARTIN ARJOVSKY, DAVID LOPEZ-PAZ AND MAXIME OQUAB)
The mythical unsupervised learning

What is inside the cake?

- Yann says “predictive modeling” and speaks about multimodal distributions. “when the pen falls, you do not know exactly where it will fall, but you know that the floor will stop it.”

- Without labels, everything is in $P(X)$. Statisticians say “density estimation”.
  
  See (Hastie et al., 2009) chapter 14.
The mythical unsupervised learning

What is inside the cake?

- What about “discovering affordances”? 
  - what can I do with a new toy?
  - what can others do with it?
  - what will be the result?

- This entails “discovering causal mechanisms”
  - whoever knows the distribution can reproduce what was demonstrated in the training data.
  - whoever knows the causal mechanism can play with the new toy in new ways.
Traditional unsupervised learning

Models engineered to resemble the true data distribution.

- Most systems train $\theta$ using the Maximum Likelihood Principle.
- If any observation has likelihood zero, the likelihood of the whole data set is zero, and its maximization is meaningless.
- Therefore one must model everything.
- This is why the models are highly engineered to resemble the true distribution enough.
- Complex models.

Any distance $D(Q, P_\theta)$ goes!
Simple models for a complex world
When $Q$ is far from the optimal $P_\theta$, optimizing different distances $D(Q, P_\theta)$ yields different solutions.

- Minimizing a distance $D(Q, P_\theta)$ sensitive to causation hints will select a model $P_\theta$ that possesses the same hints as the target distribution $Q$,

- ... and hopefully reveal causal phenomena.

Simple causal models, unrealistic data distributions

Distance sensitive to causal footprints

Alternative approach
Alternative approach

- When $Q$ is far from the optimal $P_\theta$, optimizing different distances $D(Q, P_\theta)$ yields different solutions.
- Minimizing a distance sensitive to causation hints will select a model $P_\theta$ that possesses the same hints as the target distribution $Q$, and hopefully reveal causal phenomena.

Simple causal models, unrealistic data distributions

Distance sensitive to causal footprints
Distances

We do not know how to construct a distance that is sensitive to causal hints.

Let’s start by looking at the known probabilistic distances.

Implicit distribution models are amenable to many distances.

The popular generative adversarial networks are a good example of implicit modeling.
Implicit modeling

Observed data
\[ X \sim Q \quad \text{(unknown)} \]

Generated data
\[ G_\theta(Z) \sim P_\theta \quad \text{(parametric)} \]

To be compared

Typically low dim

Low dim support
\( \rightarrow \text{cliff shaped "density"} \)
Comparing distributions

- The *Total Variation (TV) distance*

\[ \delta(\mathbb{P}_r, \mathbb{P}_g) = \sup_{A \in \Sigma} |\mathbb{P}_r(A) - \mathbb{P}_g(A)|. \]

- The *Kullback-Leibler (KL) divergence*

\[ KL(\mathbb{P}_r \| \mathbb{P}_g) = \int \log \left( \frac{P_r(x)}{P_g(x)} \right) P_r(x) d\mu(x), \]

requires densities, asymmetric, possibly infinite
Comparing distributions

- The *Jensen-Shannon* (JS) divergence
  \[
  JS(\mathbb{P}_r, \mathbb{P}_g) = KL(\mathbb{P}_r \| \mathbb{P}_m) + KL(\mathbb{P}_g \| \mathbb{P}_m),
  \]
  symmetric, does not require densities, \(0 \leq JS \leq \log(2)\)

- The *Earth-Mover* (EM) distance or *Wasserstein-1*
  \[
  W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y) \sim \gamma} \left[ \|x - y\| \right],
  \]
  always defined, involves metric on underlying space.
Generative adversarial network

\[ L(\phi, \theta) = \mathbb{E}_{x \sim P_r}[\log D_\phi(x)] + \mathbb{E}_{z \sim P_z}[\log(1 - D_\phi(g_\theta(z)))] \]
Findings (work in progress)

1. We should prefer topologically weak distances.
2. Optimizing a Wasserstein-like distance makes GANS work more reliably.
3. There are other topologically weak distances with better statistical properties and better optimization algorithms than the Wasserstein distance. But these distances also impose strict geometry constraints that may
   a. make it harder to minimize the nonconvex landscape,
   b. be incompatible with the idea of a distance sensitive to causal hints.

• (1) (2) : (Arkovsky et al., “Wasserstein GANS”, ICML 2017)
• (3a) : (Bottou et al., “Geometrical Insights for Implicit Generative Modeling”, ArXiV:1712.07822, 2017)
Findings (work in progress)

1. We should prefer topologically weak distances

2. Optimizing a Wasserstein-like distance makes GANS work more reliably.

3. There are other topologically weak distances with better statistical properties and better optimization algorithms than the Wasserstein distance. But these distances also impose strict geometry constraints that may
   a. make it harder to minimize the nonconvex landscape,
   b. be incompatible with the idea of a distance sensitive to causal hints.

• (1) (2) : (Arkovsky et al., “Wasserstein GANS”, ICML 2017)
• (3a) : (Bottou et al., “Geometrical Insights for Implicit Generative Modeling”, ArXiv:1712.07822, 2017)
7- Wasserstein GANs

(ARJOVSKY, CHINTALA, BOTTOU, 2017)
Generative adversarial network

$G_{\theta}$

$Z \sim P_z$

$G_{\theta}(Z) \sim P_g$

$D_{\phi}$

$X \sim P_r$

$P_r$ or $P_g$?

Discriminator maximizes and generator minimizes

$L(\phi, \theta) = \mathbb{E}_{x \sim P_r} [\log D_{\phi}(x)] + \mathbb{E}_{z \sim P_z} [\log (1 - D_{\phi}(g_{\theta}(z)))]$
Generative adversarial network

Discriminator maximizes and generator minimizes

\[ L(\phi, \theta) = \mathbb{E}_{x \sim P_r}[\log D_\phi(x)] + \mathbb{E}_{z \sim P_Z}[\log(1 - D_\phi(g_\theta(z)))] \]

- Keeping the discriminator optimal: \( \min_\theta L(\phi^*(\theta), \theta) \) minimizes \( JS(P_r, P_g) \)
- Keeping the generator optimal: \( \max_\phi L(\phi, \theta^*(\phi)) \) yields garbage

Nasty saddle point problem
Problem with GAN training

If one trains the discriminator thoroughly, the generator receives no gradient...
Alternate GAN training

Alternate update that has less vanishing gradients

\[ \Delta \theta \alpha \mathbb{E}_{z \sim p_z} [\nabla_\theta \log(D_{\phi}(g_\theta(z)))] \]

Under optimality optimizes

\[ KL(\mathbb{P}_\theta || \mathbb{P}_r) - 2JS(D_r || \mathbb{P}_\theta) \]

Problems: JSD with the wrong sign, reverse KL has high mode dropping. Still unstable when D is good.
Distributions with low dimensional support

Let $P_0$ and $P_{\theta}$ be two uniform distributions supported by parallel line segments separated by distance $\theta$.

- $W(P_0, P_{\theta}) = |\theta|$, 
- $JS(P_0, P_{\theta}) = \begin{cases} \log 2 & \text{if } \theta \neq 0, \\ 0 & \text{if } \theta = 0, \end{cases}$
- $KL(P_{\theta}||P_0) = KL(P_0||P_{\theta}) = \begin{cases} +\infty & \text{if } \theta \neq 0, \\ 0 & \text{if } \theta = 0, \end{cases}$
- $\delta(P_0, P_{\theta}) = \begin{cases} 1 & \text{if } \theta \neq 0, \\ 0 & \text{if } \theta = 0. \end{cases}$

Continuous in $\theta$
Optimizing a Wasserstein(ish) distance

Wasserstein-1 has a simple dual formulation (Kantorovich)

\[ W(P_r, P_\theta) = \max_{\|f\|_L \leq 1} \mathbb{E}_{x \sim P_r}[f(x)] - \mathbb{E}_{x \sim P_\theta}[f(x)] \]

- Parametrize \( f(x) \), for instance with a neural network.
- Enforce Lipschitz constraint, for instance by aggressively clipping the weights.
- Maintain \( f(x) \) well trained, and train \( G_\theta(z) \) by back-prop through \( f(x) \).
- No vanishing gradients!
No vanishing gradients
WGANG loss correlates with sample quality
Normal GAN loss does not correlate with sample quality
WGAN is less sensitive to modeling choices

Figure 5: Algorithms trained with a DCGAN generator. Left: WGAN algorithm. Right: standard GAN formulation. Both algorithms produce high quality samples.
WGAN is less sensitive to modeling choices

Figure 6: Algorithms trained with a generator without batch normalization and constant number of filters at every layer (as opposed to duplicating them every time as in [18]). Aside from taking out batch normalization, the number of parameters is therefore reduced by a bit more than an order of magnitude. Left: WGAN algorithm. Right: standard GAN formulation. As we can see the standard GAN failed to learn while the WGAN still was able to produce samples.
WGAN is less sensitive to modeling choices.

Figure 7: Algorithms trained with an MLP generator with 4 layers and 512 units with ReLU nonlinearities. The number of parameters is similar to that of a DCGAN, but it lacks a strong inductive bias for image generation. Left: WGAN algorithm. Right: standard GAN formulation. The WGAN method still was able to produce samples, lower quality than the DCGAN, and of higher quality than the MLP of the standard GAN. Note the significant degree of mode collapse in the GAN MLP.
8- The geometry of weak Integral Probability Metrics
Implicit modeling

Let $z$ be a random variable with known distribution $\mu_z$ defined on a suitable probability space $\mathcal{Z}$ and let $G_\theta$ be a measurable function, called the generator, parametrized by $\theta \in \mathbb{R}^d$,

$$G_\theta : \ z \in \mathcal{Z} \mapsto G_\theta(z) \in \mathcal{X}.$$  

The random variable $G_\theta(Z) \in \mathcal{X}$ follows the push-forward distribution

$$G_\theta(z) \# \mu_z(z) : \ A \in \mathcal{A} \mapsto \mu_z(G_\theta^{-1}(A)).$$

By varying the parameter $\theta$ of the generator $G_\theta$, we can change this push-forward distribution and hopefully make it close to the data distribution $Q$ according to the criterion of interest.
Implicit modeling

Let $z$ be a random variable with known distribution $\mu_z$ defined on a suitable probability space $\mathcal{Z}$ and let $G_\theta$ be a measurable map, called the generator, parametrized by $\theta \in \mathbb{R}^d$.

The random variable $z \mapsto \mu_{G_\theta}(z) = \mu_z(G_\theta^{-1}(A))$.

By varying the parameter $\theta$ of the generator $G_\theta$, we can change this push-forward distribution and hopefully make it close to the data distribution $Q$ according to the criterion of interest.
Learning

- Let $Q$ be the training data distribution (the complex world)
- Let $P_\theta \in \mathcal{F}$ be the (implicit) parametric models (the simple models)
- Learn by minimizing $\min_{P_\theta \in \mathcal{F}} D(Q, P_\theta)$

When $Q$ is far from $\mathcal{F}$, the choice of a distance matters!
Probability comparison criteria for implicit models

\[ D(Q, P) = \sup_{(f_Q, f_P) \in Q} \mathbb{E}_Q[f_Q(x)] - \mathbb{E}_P[f_P(x)] \]

Gives a broad family of probability distances by changing the set of pairs \((f_Q, f_P)\) considered in the supremum.

\[ \min_\theta \left\{ C(\theta) \triangleq \max_{(f_Q, f_P) \in Q} \mathbb{E}_{x \sim Q}[f_Q(x)] - \mathbb{E}_{z \sim \mu_z}[f_P(G_\theta(z))] \right\}. \] (4)
Theorem 3.1. Let $C$ be the cost function defined in (4) and let $\theta_0$ be a specific value of the generator parameter. Under the following assumptions,

a. there is $(f_Q^*, f_P^*) \in \mathcal{Q}$ such that $C(\theta_0) = \mathbb{E}_Q[f_Q^*(x)] - \mathbb{E}_{\mu_z}[f_P^*(G_{\theta_0}(z))],$

b. the function $C$ is differentiable in $\theta_0,$

c. the functions $h_z = \theta \mapsto f_P^*(G_\theta(z))$ are $\mu_z$-almost surely differentiable in $\theta_0,$

d. and there exists an open neighborhood $\mathcal{V}$ of $\theta_0$ and a $\mu_z$-integrable function $D(z)$ such that $\forall \theta \in \mathcal{V}, |h_z(h_z(\theta_0))| \leq D(z)\|\theta - \theta_0\|,$

we have the equality $\text{grad}_{\theta} C(\theta_0) = -\mathbb{E}_{z \sim \mu_z}[\text{grad}_{\theta} h_z(\theta_0)].$

(Arjovsky et al., ICML 2017)
Algorithmic ideas

The ideal world

- After optimizing $(f_Q, f_P)$ we can get unbiased estimates of the gradient of $C(\theta)$.
- Such gradient estimates can be used for stochastic gradient descent on $\theta$.

The real world

- We cannot really optimize $(f_Q, f_P)$ -- too slow, too hard, not enough data...
- In practice interleave (many) stochastic ascent steps on $(f_Q, f_P)$ and (relatively few) stochastic descent steps on $\theta$.
- This can be slow and tricky. Lots of room for improvement. Search for “xxxGAN”.
Algorithmic ideas

The ideal world

- After optimizing \((f_Q, f_P)\) we can get unbiased gradient estimates, \(\nabla C(\theta)\).
- Such gradient estimates can be used for stochastic gradient descent on \(C(\theta)\).

The real world

- We cannot really optimize \((f_Q, f_P)\) -- too slow, too hard, not enough data...
- In practice interleave (many) stochastic ascent steps on \((f_Q, f_P)\) and (relatively few) stochastic descent steps on \(\theta\).
- This can be slow and tricky. Lots of room for improvement. Search for “xxxGAN”.

Let’s assume that this works. What about the distances?
Integral Probability Metrics (IPM)

\[ D(Q, P) = \sup_{f \in Q} \mathbb{E}_Q[f(X)] - \mathbb{E}_P[f(X)] \]

where \( Q \) satisfies \( \forall f \in Q, -f \in Q \).

Proposition 3.4. Any integral probability metric \( D \), (5) is a pseudodistance.

\[ \forall x, y, z \in \mathcal{X} \]
\[  \begin{cases} 
  (o) & d(x, x) = 0 \\
  (i) & x \neq y \implies d(x, y) > 0 \\
  (ii) & d(x, y) = d(y, x) \\
  (iii) & d(x, y) \leq d(x, z) + d(z, y) 
\end{cases} \]

(zero) (separation) (symmetry) (triangular inequality)
f-divergences

\[ D_f(Q, P) \triangleq \int f\left(\frac{q(x)}{p(x)}\right) p(x) \, d\mu(x) \]  

**Proposition 3.5 ([52,53] (informal)).** Usually, \(^9\)

\[ D_f(Q, P) = \sup_{g \text{ bounded, measurable}} \mathbb{E}_Q[g(x)] - \mathbb{E}_P[f^*(g(x))] \]

<table>
<thead>
<tr>
<th>f(t)</th>
<th>dom(f*)</th>
<th>f*(u)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\frac{1}{2}</td>
<td>t - 1</td>
<td>)</td>
</tr>
<tr>
<td>(t \log(t))</td>
<td>(\mathbb{R})</td>
<td>(\exp(u - 1))</td>
</tr>
<tr>
<td>(-\log(t))</td>
<td>(\mathbb{R}_-)</td>
<td>(-1 - \log(-u))</td>
</tr>
<tr>
<td>(t \log(t) - (t + 1) \log(t + 1))</td>
<td>(\mathbb{R}_-)</td>
<td>(-\log(1 - \exp(u)))</td>
</tr>
</tbody>
</table>

\(\text{(Nguyen et al., IEEE Trans Inf Theory 2010)}\)  \(\text{(Nowozin et al., NIPS 2016)}\)
Wasserstein distances

\[ \forall Q, P \in \mathcal{P}_X^P \quad W_p(Q, P)^p \overset{\Delta}{=} \inf_{\pi \in \Pi(Q, P)} \mathbb{E}_{(x, y) \sim \pi} [d(x, y)^p], \quad (8) \]

Image stolen from Gabriel Peyré slides: “An introduction to Optimal Transport”

See also Cedric Villani “Optimal Transport Old and New” (2009)
Wasserstein distances

Kantorovich duality

\[ \forall Q, P \in P_\mathcal{X}^p, \quad W_p(Q, P)^p = \sup_{(f_Q, f_P) \in Q_c} \mathbb{E}_Q[f_Q(x)] - \mathbb{E}_P[f_P(x)], \quad (13) \]

\[ \forall Q, P \in P_\mathcal{X}^1, \quad W_1(Q, P) = \sup_{f \in \text{Lip}_1} \mathbb{E}_Q[f(x)] - \mathbb{E}_P[f(x)]. \quad (14) \]
Energy distance - Euclidean case

\[ \mathcal{E}(Q, P)^2 \triangleq 2\mathbb{E}_{x \sim Q, y \sim P}[\|x - y\|] - \mathbb{E}_{x \sim Q}[\|x - x'\|] - \mathbb{E}_{y \sim P}[\|y - y'\|], \quad (15) \]

This seems weird but it turns out that:

\[ \mathcal{E}(Q, P)^2 = \frac{1}{c_d} \int_{\mathbb{R}^d} \frac{|q(t) - \hat{p}(t)|^2}{\|t\|^{d+1}} dt \quad \text{with} \quad c_d = \frac{\pi^{\frac{d}{2}}}{\Gamma\left(\frac{d+1}{2}\right)}. \quad (16) \]
Energy distance – Generalized

\[ E_d(Q, P)^2 = 2\mathbb{E}_{x \sim Q}[d(x, y)] - \mathbb{E}_{x \sim Q}[d(x, x')] - \mathbb{E}_{y \sim P}[d(y, y')] \]. \hspace{1cm} (17)

Is this positive?
Is this a distance?
Energy distance – Generalized

**Theorem 3.12 ([69]).** The right hand side of definition (17) is:

i) nonnegative for all $P, Q$ in $P^1_X$ if and only if the symmetric function $d$ is a negative definite kernel, that is,

$$\forall n \in \mathbb{N} \ \forall x_1 \ldots x_n \in X \ \forall c_1 \ldots c_n \in \mathbb{R} \ \sum_{i=1}^{n} c_i = 0 \ \implies \ \sum_{i=1}^{n} \sum_{j=1}^{n} d(x_i, x_j) c_i c_j \leq 0 . \ (18)$$

ii) strictly positive for all $P \neq Q$ in $P^1_X$ if and only if the function $d$ is a strongly negative definite kernel, that is, a negative definite kernel such that, for any probability measure $\mu \in P^1_X$ and any $\mu$-integrable real-valued function $h$ such that $E_\mu[h(x)] = 0$,

$$E_{x \sim \mu}[d(x, y)h(x)h(y)] = 0 \ \implies \ h(x) = 0 \ \mu\text{-almost everywhere}.$$
Energy distance – Surprise

\[ K_d(x, y) \overset{\Delta}{=} \frac{1}{2} (d(x, x_0) + d(y, x_0) - d(x, y)) \].

(19)

**Proposition 3.15.** The function \( d \) is a negative definite kernel if and only if \( K_d \) is a positive definite kernel, that is,

\[
\forall n \in \mathbb{N} \quad \forall x_1 \ldots x_n \in \mathcal{X} \quad \forall c_1 \ldots c_n \in \mathbb{R} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j K_d(x_i, x_j) \geq 0.
\]

Then, thanks to RKHS theory...

\[
\forall Q, P \in \mathcal{P}_X^1 \quad \mathcal{E}_d(Q, P) = \| \mathbb{E}_{x \sim Q} [\Phi_x] - \mathbb{E}_{y \sim P} [\Phi_y] \|_{\mathcal{H}}.
\]

(Sejdinovic et al., 2013) (Rachev et al., 2013)
Energy distance
= Maximum Mean Discrepancy (MMD)

\[
\mathcal{E}_d(Q, P) = \| \mathbb{E}_Q[\Phi_x] - \mathbb{E}_P[\Phi_x] \|_{\mathcal{H}} \\
= \sup_{\|f\|_{\mathcal{H}} \leq 1} \langle f, \mathbb{E}_P[\Phi_x] - \mathbb{E}_Q[\Phi_x] \rangle \\
= \sup_{\|f\|_{\mathcal{H}} \leq 1} \mathbb{E}_P[\langle f, \Phi_x \rangle] - \mathbb{E}_Q[\langle f, \Phi_x \rangle] \\
= \sup_{\|f\|_{\mathcal{H}} \leq 1} \mathbb{E}_P[f(x)] - \mathbb{E}_Q[f(x)] .
\]  

(Gretton et al., 2012)
Strong topology vs weak topology

\[
\begin{align*}
W(P_0, P_\theta) &= |\theta|, \\
JS(P_0, P_\theta) &= \begin{cases} 
\log 2 & \text{if } \theta \neq 0, \\
0 & \text{if } \theta = 0,
\end{cases} \\
KL(P_\theta || P_0) &= KL(P_0 || P_\theta) = \begin{cases} 
+\infty & \text{if } \theta \neq 0, \\
0 & \text{if } \theta = 0,
\end{cases} \\
\delta(P_0, P_\theta) &= \begin{cases} 
1 & \text{if } \theta \neq 0, \\
0 & \text{if } \theta = 0.
\end{cases}
\end{align*}
\]

(Arjovsky et al., ICML 2017)
How different are WD and MMD?

$$W_1(Q, P) = \sup_{f \in \text{Lip}_1} E_Q[f(x)] - E_P[f(x)],$$

$$\mathcal{E}_d(Q, P) = \sup_{\|f\|_\infty \leq 1} E_P[f(x)] - E_Q[f(x)].$$
How different are WD and MMD?

Theorem 4.3. Let $Q$ be a probability distributions on $\mathcal{X}$. Let $x_1 \ldots x_n$ be $n$ independent $Q$-distributed random variables, and let $Q_n = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i}$ be the corresponding empirical probability distribution.

\[
Q \in \mathcal{P}^1_{\mathcal{X}} \quad \mathbb{E}_{x_1 \ldots x_n \sim Q} \left[ \mathcal{E}_d(Q, Q)^2 \right] = \frac{1}{n} \mathbb{E}_{x, x' \sim Q} [d(x, x')] = \mathcal{O}(n^{-1}).
\]

\[
Q \in \mathcal{P}^2_{\mathbb{R}^d} \quad \mathbb{E}_{x_1 \ldots x_n \sim Q} \left[ W_1(Q_n, Q) \right] = \mathcal{O}(n^{-1/d}).
\]

This is reached (Sanjeev’s sphere)

Wasserstein seem hopeless
How different are WD and MMD?

Things look different in practice

- ED/MMD training of low dim implicit models works nicely.
- ED/MMD training of high dim implicit models often gets stuck.
- whereas “WD” training of the same high dim implicit models can give results.

WD-like.  

Just the opposite of what one would expect!
3– Geodesic geometry for probability measures
Theorem 5.1. Let $\gamma : [a, b] \rightarrow \mathcal{X}$ be a curve joining two points $\gamma_a, \gamma_b$ such that $d(\gamma_a, \gamma_b) < \infty$. This curve is a minimal geodesic of length $d(\gamma_a, \gamma_b)$ if and only if $\forall a \leq t \leq t' \leq b$, $d(\gamma_a, \gamma_t) + d(\gamma_t, \gamma_{t'}) + d(\gamma_{t'}, \gamma_b) = d(\gamma_a, \gamma_b)$. 

$\mathbb{R}^2$ equipped with the $L_1$ distance.
Corollary 5.2. Let $\gamma : [0, 1] \to X$ be a curve joining two points $\gamma_0, \gamma_1 \in X$ such that $d(\gamma_0, \gamma_1) < \infty$. The following three assertions are equivalent:

a) The curve $\gamma$ is a constant speed minimal geodesic of length $d(\gamma_0, \gamma_1)$.

b) $\forall t, t' \in [0, 1], \quad d(\gamma_t, \gamma_{t'}) = |t - t'| d(\gamma_0, \gamma_1)$.

c) $\forall t, t' \in [0, 1], \quad d(\gamma_t, \gamma_{t'}) \leq |t - t'| d(\gamma_0, \gamma_1)$. 

Mixture geodesics

\[ \forall t \in [0, 1] \quad P_t = (1-t)P_0 + tP_1 \]  
(26)

**Theorem 6.1.** Let \( \mathcal{P}_X \) be equipped with a distance \( D \) that belongs to the IPM family (5). Any mixture curve (26) joining two distributions \( P_0, P_1 \in \mathcal{P}_X \) such that \( D(P_0, P_1) < \infty \) is a constant speed minimal geodesic.

**Theorem 6.3.** Let \( K \) be a characteristic kernel and let \( \mathcal{P}_X \) be equipped with the MMD distance \( \mathcal{E}_{d_K} \). Then any two probability measures \( P_0, P_1 \in \mathcal{P}_X \) such that \( \mathcal{E}_{d_K}(P_0, P_1) < \infty \) are joined by exactly one constant speed minimal geodesic, the mixture geodesic (26).
Let $\mathcal{X}$ be a Euclidean space.

Let $\mathcal{P}_\mathcal{X}$ be equipped with the $p$-Wasserstein distance (8).

Let $P_0, P_1 \in \mathcal{P}_\mathcal{X}$ be two distributions with optimal transport plan $\pi$.

For all $t \in [0, 1]$, $P_t = ((1-t)x + ty) \# \pi(x, y)$. 
Displacement geodesics (General)

Definition 6.7 (Displacement geodesic). Let $\mathcal{X}$ be a strictly intrinsic Polish metric space and let $\mathcal{P}_\mathcal{X}^p$ be equipped with the $p$-Wasserstein distance $W_p$. The curve $t \in [0,1] \mapsto P_t \in \mathcal{P}_\mathcal{X}^p$ is called a displacement geodesic if, for all $0 \leq t \leq t' \leq 1$, there is a distribution $\pi_4 \in \mathcal{P}_\mathcal{X}^4$ such that

i) The four marginals of $\pi_4$ are respectively equal to $P_0$, $P_t$, $P_{t'}$, $P_1$.

ii) The pairwise marginal $(x,z) \neq \pi_4(x,u,v,z)$ is an optimal transport plan

$$W_p(P_0,P_1)^p = \mathbb{E}_{(x,u,v,z) \sim \pi_4}[d(x,z)^p].$$

iii) The following relations hold $\pi_4(x,u,v,z)$-almost surely:

$$d(x,u) = t \cdot d(x,z), \quad d(u,v) = (t' - t) \cdot d(x,z), \quad d(v,z) = (1-t') \cdot d(x,z).$$

Essentially the same thing
The geodesics of WD and MMD

- With the Energy Distance $\mathcal{E}_d$ or the Maximum Mean Discrepancy $\mathcal{E}_{d_K}$, the sole shortest path is the mixture geodesic (Theorem 6.3.)

- With the $p$-Wasserstein distance $\mathcal{W}_p$, for $p > 1$, the sole shortest paths are displacement geodesics (Corollary 6.9.)

- With the 1-Wasserstein distance $\mathcal{W}_1$, there are many shortest paths, including the mixture geodesic, all the displacement geodesics, and all kinds of hybrid curves (Corollary 6.10.)
Families of curves

We now assume that \( \mathcal{P}_\mathcal{X} \) is a strictly intrinsic Polish space equipped with a distance \( D \). Let \( \mathcal{C} \) be a family of smooth constant speed curves in \( \mathcal{P}_\mathcal{X} \). Although these curves need not be minimal geodesics, the focus of this section is limited to three families of curves defined in Section 6:

- the family \( \mathcal{C}_g(D) \) of all minimal geodesics in \((\mathcal{P}_\mathcal{X}, D)\).
- the family \( \mathcal{C}_d(W_p) \) of the displacement geodesics in \((\mathcal{P}_\mathcal{X}^p, W_p)\).
- the family \( \mathcal{C}_m \) of the mixture curves in \( \mathcal{P}_\mathcal{X} \).
Convexity w.r.t. a family of curves

**Definition 7.1.** Let $\mathcal{P}_X$ be a strictly intrinsic Polish space. A closed subset $\mathcal{F} \subset \mathcal{P}_X$ is called convex with respect to the family of curves $C$ when $C$ contains a curve $t \in [0, 1] \mapsto P_t \mathcal{X}$ connecting $P_0$ and $P_1$ whose graph is contained in $\mathcal{F}$, that is, $P_t \in \mathcal{F}$ for all $t \in [0, 1]$.

**Definition 7.2.** Let $\mathcal{P}_X$ be a strictly intrinsic Polish space. A real valued function $f$ defined on $\mathcal{P}_X$ is called convex with respect to the family of constant speed curves $C$ when, for every curve $t \in [0, 1] \mapsto P_t \in \mathcal{P}_X$ in $C$, the function $t \in [0, 1] \mapsto f(P_t) \in \mathbb{R}$ is convex.

For brevity we also say that $\mathcal{F}$ or $f$ is geodesically convex when $C = C_g(D)$, mixture convex when $C = C_m$, and displacement convex when $C = C_d(W_p)$. 
Convex optimization à-la-carte

Theorem 7.3 (Convex optimization à-la-carte). Let $\mathcal{P}_X$ be a strictly intrinsic Polish space equipped with a distance $D$. Let the closed subset $\mathcal{F} \subseteq \mathcal{P}_X$ and the cost function $f : X \mapsto \mathbb{R}$ be both convex with respect to a same family $\mathcal{C}$ of constant speed curves. Then, for all $M \geq \min_{\mathcal{F}}(f)$,

i) the level set $L(f, \mathcal{F}, M) = \{ P \in \mathcal{F} : f(P) \leq M \}$ is connected,

ii) for all $P_0 \in \mathcal{F}$ such that $f(P_0) > M$ and all $\epsilon > 0$, there exists $P \in \mathcal{F}$ such that $D(P, P_0) = O(\epsilon)$ and $f(P) \leq f(P_0) - \epsilon(f(P_0) - M)$.

A descent algorithm will find the global minimum. Even with a nonconvex parametrization of $P_\theta \in \mathcal{F}$. 
The convexity of implicit model families

Short story

- An implicit model family cannot be mixture convex while having a nice smooth generator $G_\theta(z)$.
- It is relatively easy to make implicit model families that are displacement convex.
Example 7.5. Let $\mu_x$ be the uniform distribution on $\{-1, +1\}$. Let the parameter $\theta$ be constrained to the square $[-1, 1]^2 \subset \mathbb{R}^2$ and let the generator function be

$$G_\theta : z \in \{-1, 1\} \mapsto G_\theta(z) = z\theta.$$ 

The corresponding model family is

$$\mathcal{F} = \left\{ P_\theta = \frac{1}{2}(\delta_\theta + \delta_{-\theta}) : \theta \in [-1, 1] \times [-1, 1] \right\}.$$ 

It is easy to see that this model family is displacement convex but not mixture convex. Figure 5 shows the level sets for both criteria $\mathcal{E}(Q, P_\theta)$ and $W_1(Q, P_\theta)$ for the target distribution $Q = P_{(2,2)} \notin \mathcal{F}$. Both criteria have the same global minima in $(1,1)$ and $(-1,-1)$. However the energy distance has spurious local minima in $(-1,1)$ and $(1,-1)$ with a relatively high value of the cost function.
How things can go wrong
The convexity of distance functions

- Learn by minimizing $\min_{P_\theta \in F} D(Q, P_\theta)$

When is $D(Q, P_\theta)$ geodesically convex?
Mixture-convexity

Proposition 7.6. Let $\mathcal{P}_X$ be equipped with a distance $D$ that belongs to the IPM family (5). Then $D$ is mixture convex.

Therefore

- $\mathcal{E}_d(Q,P)$ is mixture convex and geodesically convex.
- $W_1(Q,P)$ is mixture convex.
The Wasserstein distance is not displacement convex
The Wasserstein distance is not displacement convex

Wait. There is still hope.
Proposition 7.8. Let $\mathcal{X}$ be a strictly intrinsic Polish space equipped with a geodesically convex distance $d$ and let $\mathcal{P}_\mathcal{X}$ be equipped with the $1$-Wasserstein distance $W_1$. For all $Q \in \mathcal{P}_\mathcal{X}$ and all displacement geodesics $t \in [0, 1] \mapsto P_t$,

$$
\forall t \in [0, 1] \quad W_1(Q, P_t) \leq (1-t)W_1(Q, P_0) + tW_1(Q, P_1) + 2t(1-t)K(Q, P_0, P_1)
$$

with $K(Q, P_0, P_1) \leq 2 \min_{u_0 \in \mathcal{X}} \mathbb{E}_{u \sim Q}[d(u, u_0)]$. 

Almost convexity

Bound the convexity violation

Gross bound
Almost convexity

Fig. 8. The construction of $\pi \in \mathcal{P}_{X^6}$ in the proof of Proposition 7.8.
Descent works until it gets too close

**Theorem 7.9.** Let $\mathcal{X}$ be a strictly intrinsic Polish space equipped with a geodesically convex distance $d$ and let $\mathcal{P}_\mathcal{X}^1$ be equipped with the 1-Wasserstein distance $W_1$. Let $\mathcal{F} \subset \mathcal{P}_\mathcal{X}^1$ be displacement convex and let $Q \in \mathcal{P}_\mathcal{X}^1$ have expected diameter

$$D = 2 \min_{u_0 \in \mathcal{X}} \mathbb{E}_{u \sim Q}[d(u, u_0)].$$

Then the level set $L(Q, \mathcal{F}, M) = \{P_\theta \in \mathcal{F} : W_1(Q, P_\theta) \leq M\}$ is connected if

$$M > \inf_{P_\theta \in \mathcal{F}} W_1(Q, P_\theta) + 2D.$$
Related works

Many authors went for this kind of results

- Amari – Information geometry
- Freeman & Bruna, 2017 – connectivity of level sets in relu networks
- Aufflinger & Ben Arous 2013 – random functions on the sphere
- More?

Critical difference

All these work look at the intrinsic geometry of a family of models. We look at the geometry of the entire family of probability measures, then introduce convexity concepts for the parametrized families of interest.
Conclusion
Economically meaningful technological successes have triggered a new wave of hope (and hype) about Artificial Intelligence.

Statistics ≠ semantics
⇒ machine learning alone cannot crack AI
⇒ we need a couple conceptual breakthroughs.

➔ Causation seems to play an central role (detective guesswork.)
➔ Even static image datasets contain hints about causal relations (experimental results.)
➔ Using the right probability distances could help (some theoretical and experimental results.)