Wasserstein 1 Distance for Generative Models

Tristan Milne

March 12th, 2021

Introduction

I know what you're all here for...

I know what you're all here for...

celebrity quizzes

Celebrity Quiz

Figure: Can you name these A-list celebs?

[1] From Karras et. al. [6]

Celebrity Quiz

Figure: Can you name these A-list celebs?

Name them whatever you want, because they're **not real people**[1]

[1] From Karras et. al. [6]

Generative Modelling

Suppose $\Omega \subset \mathbb{R}^d$ is compact, and $\nu \in \mathcal{P}(\Omega)$ is a distribution we **want to sample**.

[2] Arjovsky et. al. [1]

Generative Modelling

Suppose $\Omega \subset \mathbb{R}^d$ is compact, and $\nu \in \mathcal{P}(\Omega)$ is a distribution we **want to sample**.

- e.g. pictures of celebrities, bank data, medical information for rare diseases ...

[2] Arjovsky et. al. [1]

Generative Modelling

Suppose $\Omega \subset \mathbb{R}^d$ is compact, and $\nu \in \mathcal{P}(\Omega)$ is a distribution we **want to sample**.

- e.g. pictures of celebrities, bank data, medical information for rare diseases ...
- have some samples of ν, but want more.

[2] Arjovsky et. al. [1]

Generative Modelling

Suppose $\Omega \subset \mathbb{R}^d$ is compact, and $\nu \in \mathcal{P}(\Omega)$ is a distribution we **want to sample**.

- e.g. pictures of celebrities, bank data, medical information for rare diseases ...
- have some samples of ν, but want more.

Let μ be a distribution **we can sample**

[2] Arjovsky et. al. [1]

Generative Modelling

Suppose $\Omega \subset \mathbb{R}^d$ is compact, and $\nu \in \mathcal{P}(\Omega)$ is a distribution we **want to sample**.

- e.g. pictures of celebrities, bank data, medical information for rare diseases ...
- have some samples of ν, but want more.

Let μ be a distribution **we can sample**

- e.g. $G_w : \mathbb{R}^m \to \mathbb{R}^d$ is a function (the "generator") with parameters w, $\zeta = \mathcal{N}(0, I_m)$,

$$\mu = (G_w)_{\#}\zeta$$

[2] Arjovsky et. al. [1]

Generative Modelling

Suppose $\Omega \subset \mathbb{R}^d$ is compact, and $\nu \in \mathcal{P}(\Omega)$ is a distribution we **want to sample**.

- e.g. pictures of celebrities, bank data, medical information for rare diseases ...
- have some samples of ν, but want more.

Let μ be a distribution **we can sample**

- e.g. $G_w : \mathbb{R}^m \to \mathbb{R}^d$ is a function (the "generator") with parameters w, $\zeta = \mathcal{N}(0, I_m)$,

$$\mu = (G_w)_{\#}\zeta$$

Want to choose w so that $\mu \approx \nu$.

[2] Arjovsky et. al. [1]

Generative Modelling

Suppose $\Omega \subset \mathbb{R}^d$ is compact, and $\nu \in \mathcal{P}(\Omega)$ is a distribution we **want to sample**.

- e.g. pictures of celebrities, bank data, medical information for rare diseases ...
- have some samples of ν, but want more.

Let μ be a distribution **we can sample**

- e.g. $G_w : \mathbb{R}^m \to \mathbb{R}^d$ is a function (the "generator") with parameters w, $\zeta = \mathcal{N}(0, I_m)$,

$$\mu = (G_w)_\# \zeta$$

Want to choose w so that $\mu \approx \nu$.

I'll explain how to do this using **Wasserstein Generative Adversarial Networks (WGANs)**[2]

[2] Arjovsky et. al. [1]

Testing if $\mu \approx \nu$

How do we test if $\mu \approx \nu$?

Testing if $\mu \approx \nu$

How do we test if $\mu \approx \nu$?

- We put a metric on $\mathcal{P}(\Omega)$.

Testing if $\mu \approx \nu$

How do we test if $\mu \approx \nu$?

- We put a metric on $\mathcal{P}(\Omega)$.
- The **Wasserstein distance** for the Euclidean cost is a convenient choice.

Background on OT

Monge's problem

Problem Data

- $\Omega \subset \mathbb{R}^d$ a compact set.
- $c : \Omega \times \Omega \to \mathbb{R}$ a cost function (e.g. $c(x, y) = |x - y|^p$, $p \geq 1$.)
- $\mu, \nu \in \mathcal{P}(\Omega)$ two probability measures.

Monge's problem

Problem Data

- $\Omega \subset \mathbb{R}^d$ a compact set.
- $c : \Omega \times \Omega \to \mathbb{R}$ a cost function (e.g. $c(x, y) = |x - y|^p$, $p \geq 1$.)
- $\mu, \nu \in \mathcal{P}(\Omega)$ two probability measures.

For a measurable map $T : \Omega \to \Omega$ the **pushforward measure** $T_\# \mu$ is

$$T_\# \mu(E) = \mu(T^{-1}(E))$$

Monge's problem

Problem Data

- $\Omega \subset \mathbb{R}^d$ a compact set.
- $c : \Omega \times \Omega \to \mathbb{R}$ a cost function (e.g. $c(x, y) = |x - y|^p$, $p \geq 1$.)
- $\mu, \nu \in \mathcal{P}(\Omega)$ two probability measures.

For a measurable map $T : \Omega \to \Omega$ the **pushforward measure** $T_\# \mu$ is

$$T_\# \mu(E) = \mu(T^{-1}(E))$$

Monge's Problem

$$\min_{T_\# \mu = \nu} \int_\Omega c(x, T(x)) d\mu.$$

Kantorovich Relaxation

Requiring a map T is quite strong.

Kantorovich Relaxation

Requiring a map T is quite strong.

- The admissible set $T_{\#}\mu = \nu$ is non-convex and possibly empty.

Kantorovich Relaxation

Requiring a map T is quite strong.

- The admissible set $T_{\#}\mu = \nu$ is non-convex and possibly empty.
- It requires that mass from each point x be sent to exactly one point y.

Kantorovich Relaxation

Requiring a map T is quite strong.

- The admissible set $T_{\#}\mu = \nu$ is non-convex and possibly empty.
- It requires that mass from each point x be sent to exactly one point y.

The **Kantorovich Relaxation** allows for mass at one point x to be sent to multiple points y.

Kantorovich Relaxation

Requiring a map T is quite strong.

- The admissible set $T_\# \mu = \nu$ is non-convex and possibly empty.
- It requires that mass from each point x be sent to exactly one point y.

The **Kantorovich Relaxation** allows for mass at one point x to be sent to multiple points y.

Kantorovich Problem

$$\min_{\gamma \in \Pi(\mu,\nu)} \int_\Omega c(x,y) d\gamma \quad \text{(KP)}$$

where $\Pi(\mu,\nu)$ is the set of **admissible plans**

$$\Pi(\mu,\nu) = \{\gamma \in \mathcal{P}(\Omega \times \Omega) \mid (\pi_x)_\# \gamma = \mu, (\pi_y)_\# \gamma = \nu\}.$$

About $\Pi(\mu, \nu)$

The set of admissible plans is **non-empty**

About $\Pi(\mu, \nu)$

The set of admissible plans is **non-empty**

- For example,

$$\gamma(E_1 \times E_2) = \mu(E_1)\nu(E_2).$$

About $\Pi(\mu, \nu)$

The set of admissible plans is **non-empty**

- For example,
$$\gamma(E_1 \times E_2) = \mu(E_1)\nu(E_2).$$
- **In general**, for $\gamma \in \Pi(\mu, \nu)$,
$$\gamma(E_1 \times E_2)$$
measures how much mass γ moves from E_1 to E_2.

Existence of optimal plan

Theorem

If Ω is compact and $c : \Omega \times \Omega \to \mathbb{R}$ is continuous, then (KP) admits a solution γ_0 which we call an optimal transport plan.

Existence of optimal plan

Theorem

If Ω is compact and $c : \Omega \times \Omega \to \mathbb{R}$ is continuous, then (KP) admits a solution γ_0 which we call an optimal transport plan.

Important Question: Is $\gamma_0 = (I, T_0)_{\#}\mu$ for some map T_0?

Existence of optimal plan

Theorem

If Ω is compact and $c : \Omega \times \Omega \to \mathbb{R}$ is continuous, then (KP) admits a solution γ_0 which we call an optimal transport plan.

Important Question: Is $\gamma_0 = (I, T_0)_{\#}\mu$ for some map T_0?

- Such a map is automatically optimal for **Monge's Problem**.

Kantorovich duality

Under **mild conditions**,

$$\min_{\gamma \in \Pi(\mu,\nu)} \int_\Omega c(x,y) d\gamma = \max_{\varphi,\psi \in C(\Omega), \varphi \oplus \psi \leq c} \int_\Omega \varphi d\mu + \int_\Omega \psi d\nu.$$

Maximizing (φ, ψ) are called **Kantorovich potentials**.

Kantorovich duality

Under **mild conditions**,

$$\min_{\gamma \in \Pi(\mu,\nu)} \int_\Omega c(x,y)d\gamma = \max_{\varphi,\psi \in C(\Omega), \varphi \oplus \psi \leq c} \int_\Omega \varphi d\mu + \int_\Omega \psi d\nu.$$

Maximizing (φ, ψ) are called **Kantorovich potentials**.

For c symmetric, **define** the c-transform

$$\varphi^c(y) = \inf_{x \in \Omega} c(x,y) - \varphi(x),$$

we have

$$\min_{\gamma \in \Pi(\mu,\nu)} \int_\Omega c(x,y)d\gamma = \max_{\varphi,\psi \in C(\Omega)} \int_\Omega \varphi d\mu + \int_\Omega \varphi^c d\nu$$

c-concave functions

We say φ is **c-concave** (or $\varphi \in c\text{-conc}(\Omega)$) if there exists ψ such that

$$\varphi(y) = \psi^c(y).$$

c-concave functions

We say φ is **c-concave** (or $\varphi \in c\text{-conc}(\Omega)$) if there exists ψ such that

$$\varphi(y) = \psi^c(y).$$

Lemma

For $\varphi : \Omega \to \mathbb{R}$,

$$\varphi^{cc} \geq \varphi, \quad \varphi^{ccc} = \varphi^c$$

c-concave functions

We say φ is **c-concave** (or $\varphi \in c\text{-conc}(\Omega)$) if there exists ψ such that

$$\varphi(y) = \psi^c(y).$$

Lemma

For $\varphi : \Omega \to \mathbb{R}$,

$$\varphi^{cc} \geq \varphi, \quad \varphi^{ccc} = \varphi^c$$

Means we can write

$$\min_{\gamma \in \Pi(\mu,\nu)} \int_\Omega c(x,y) d\gamma = \max_{\varphi \in c\text{-conc}(\Omega)} \int_\Omega \varphi d\mu + \int_\Omega \varphi^c d\nu$$

Relationship between φ and γ

Lemma

If $\gamma \in \Pi(\mu, \nu)$ *is an optimal plan and* φ *is a potential, then*

$$spt(\gamma) \subset \{(x, y) \in \Omega^2 \mid \varphi(x) + \varphi^c(y) = c(x, y)\}$$

Proof.

$\square$

Hint for constructing a map

If γ is optimal, it **must satisfy**

$$\varphi(x) + \varphi^c(y) = c(x, y)$$

for all $(x, y) \in \mathrm{spt}(\gamma)$.

Hint for constructing a map

If γ is optimal, it **must satisfy**

$$\varphi(x) + \varphi^c(y) = c(x, y)$$

for all $(x, y) \in \mathrm{spt}(\gamma)$.

Recalling the definition of φ^c,

$$c(x, y) - \varphi(x) = \varphi^c(y) = \min_z c(z, y) - \varphi(z).$$

Hint for constructing a map

If γ is optimal, it **must satisfy**

$$\varphi(x) + \varphi^c(y) = c(x, y)$$

for all $(x, y) \in \mathrm{spt}(\gamma)$.

Recalling the definition of φ^c,

$$c(x, y) - \varphi(x) = \varphi^c(y) = \min_z c(z, y) - \varphi(z).$$

Hence,

$$x \in \mathrm{argmin}_z c(z, y) - \varphi(z).$$

So if **the set of y for which x is in this argmin is a singleton** we have $T(x)$.

The choice of c

For the rest of this talk, take

$$c(x, y) = |x - y|^p \quad p \geq 1.$$

KP becomes

$$W_p^p(\mu, \nu) := \min_{\gamma \in \Pi(\mu,\nu)} \int_\Omega |x - y|^p d\gamma$$

- $p = 1$, measures **work**, (Optimal map found in 1999, 2001, 2002)
- $p = 2$, measures **kinetic energy**. (Optimal map found in 1987)

The choice of c affects the map

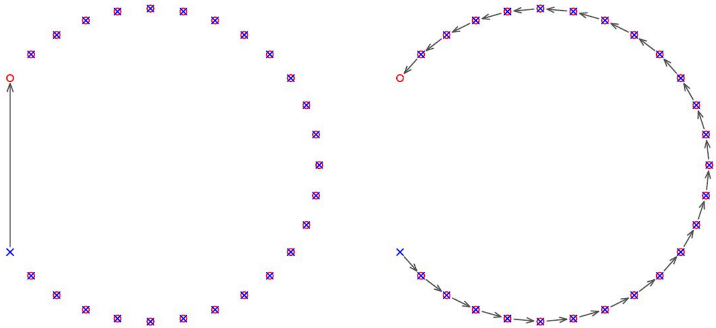


Figure: Each blue x is a point in $\mathrm{spt}(\mu)$, and red circle is a point in $\mathrm{spt}(\nu)$, all with equal mass. Left: the optimal map with $p = 1$. Right: the optimal map with $p = 2$.

3

[3] Figure taken from Hartmann and Schuhmacher [5]

Given φ, finding a map is easy with $p > 1$

Theorem

If $p > 1$, $\mu \ll \mathcal{L}$, $\mathcal{L}(\partial\Omega) = 0$, *and* φ *is a potential, then*

$$T(x) = x - (\nabla|\cdot|^p)^{-1}(\nabla\varphi(x))$$

is an optimal map for $W_p(\mu, \nu)$.

Given φ, finding a map is easy with $p > 1$

Theorem

If $p > 1$, $\mu \ll \mathcal{L}$, $\mathcal{L}(\partial\Omega) = 0$, and φ is a potential, then

$$T(x) = x - (\nabla|\cdot|^p)^{-1}(\nabla\varphi(x))$$

is an optimal map for $W_p(\mu, \nu)$.

No such theorem for $p = 1$

Given φ, finding a map is easy with $p > 1$

Theorem

If $p > 1$, $\mu \ll \mathcal{L}$, $\mathcal{L}(\partial\Omega) = 0$, *and* φ *is a potential, then*

$$T(x) = x - (\nabla|\cdot|^p)^{-1}(\nabla\varphi(x))$$

is an optimal map for $W_p(\mu, \nu)$.

No such theorem for $p = 1$

- But a potential u is **instrumental in constructing a map**.

Given φ, finding a map is easy with $p > 1$

Theorem

If $p > 1$, $\mu \ll \mathcal{L}$, $\mathcal{L}(\partial\Omega) = 0$, and φ is a potential, then

$$T(x) = x - (\nabla|\cdot|^p)^{-1}(\nabla\varphi(x))$$

is an optimal map for $W_p(\mu, \nu)$.

No such theorem for $p = 1$

- But a potential u is **instrumental in constructing a map**.
- Just no simple formula.

The c-transform for $p = 1$ is simple to compute

Lemma

If $c(x, y) = |x - y|$, *then*

$$\varphi^c(y) = \inf_{x \in \Omega} |x - y| - \varphi(x)$$

is 1*-Lipschitz.*

The c-transform for $p = 1$ is simple to compute

Lemma

If $c(x,y) = |x-y|$*, then*

$$\varphi^c(y) = \inf_{x \in \Omega} |x-y| - \varphi(x)$$

is 1*-Lipschitz.*

Lemma

If $c(x,y) = |x-y|$ *and* $\varphi \in 1\text{-}Lip(\Omega)$*, then*

$$\varphi^c = -\varphi.$$

Thus,

$$c\text{-}conc(\Omega) = 1\text{-}Lip(\Omega).$$

Computational complexity of $W_1(\mu, \nu)$ is lower than $p > 1$

Suppose we calculate $W_p(\mu, \nu)$ by the **dual**

$$W_p(\mu, \nu) = \max_{\varphi \in c\text{-conc}(\Omega)} \int_\Omega \varphi d\mu + \int_\Omega \varphi^c d\nu$$

If $p = 1$, this becomes

$$W_1(\mu, \nu) = \max_{u \in 1\text{-Lip}(\Omega)} \int_\Omega u d\mu - \int_\Omega u d\nu$$

Computational complexity of $W_1(\mu, \nu)$ is lower than $p > 1$

Suppose we calculate $W_p(\mu, \nu)$ by the **dual**

$$W_p(\mu, \nu) = \max_{\varphi \in c\text{-conc}(\Omega)} \int_\Omega \varphi d\mu + \int_\Omega \varphi^c d\nu$$

If $p = 1$, this becomes

$$W_1(\mu, \nu) = \max_{u \in 1\text{-Lip}(\Omega)} \int_\Omega u d\mu - \int_\Omega u d\nu$$

Compare: computing φ^c for $p = 2$ is equivalent to computing a **Legendre dual**

Computational complexity of $W_1(\mu,\nu)$ is lower than $p > 1$

Suppose we calculate $W_p(\mu,\nu)$ by the **dual**

$$W_p(\mu,\nu) = \max_{\varphi \in c\text{-conc}(\Omega)} \int_\Omega \varphi d\mu + \int_\Omega \varphi^c d\nu$$

If $p = 1$, this becomes

$$W_1(\mu,\nu) = \max_{u \in 1\text{-Lip}(\Omega)} \int_\Omega u d\mu - \int_\Omega u d\nu$$

Compare: computing φ^c for $p = 2$ is equivalent to computing a **Legendre dual**

- On a grid with n points per dimension, complexity of $O(n^d)$.

Summary

p	c-transform is easy	a potential gives a map
1	✓	X
> 1	X	✓

Obtaining an optimal map for $p = 1$

History of solutions

Theorem

If $\mu \ll \mathcal{L}$, there is an optimal transport map T for $W_1(\mu, \nu)$.

History of solutions

Theorem

If $\mu \ll \mathcal{L}$, there is an optimal transport map T for $W_1(\mu, \nu)$.

The first **partial solution** came from Sudakov in [8]

- Proof was discovered to have a gap by L. Ambrosio, fixed in 2003-04

History of solutions

Theorem

If $\mu \ll \mathcal{L}$, there is an optimal transport map T for $W_1(\mu, \nu)$.

The first **partial solution** came from Sudakov in [8]

- Proof was discovered to have a gap by L. Ambrosio, fixed in 2003-04
- First **correct proof** from Evans and Gangbo [3] with PDE methods for Lipschitz densities.

History of solutions

Theorem

If $\mu \ll \mathcal{L}$, there is an optimal transport map T for $W_1(\mu, \nu)$.

The first **partial solution** came from Sudakov in [8]

- Proof was discovered to have a gap by L. Ambrosio, fixed in 2003-04
- First **correct proof** from Evans and Gangbo [3] with PDE methods for Lipschitz densities.
- Proof for less regular densities from Caffarelli, Feldman, and McCann in [2] and Trudinger and Wang in [9].

History of solutions

Theorem

If $\mu \ll \mathcal{L}$, there is an optimal transport map T for $W_1(\mu, \nu)$.

The first **partial solution** came from Sudakov in [8]

- Proof was discovered to have a gap by L. Ambrosio, fixed in 2003-04
- First **correct proof** from Evans and Gangbo [3] with PDE methods for Lipschitz densities.
- Proof for less regular densities from Caffarelli, Feldman, and McCann in [2] and Trudinger and Wang in [9].

All methods use the **properties of a potential** u.

History of solutions

Theorem

If $\mu \ll \mathcal{L}$, there is an optimal transport map T for $W_1(\mu, \nu)$.

The first **partial solution** came from Sudakov in [8]

- Proof was discovered to have a gap by L. Ambrosio, fixed in 2003-04
- First **correct proof** from Evans and Gangbo [3] with PDE methods for Lipschitz densities.
- Proof for less regular densities from Caffarelli, Feldman, and McCann in [2] and Trudinger and Wang in [9].

All methods use the **properties of a potential** u.

The method I'll sketch here is that of [2] and [9].

A first observation

Lemma

If $\gamma \in \Pi(\mu, \nu)$ *is optimal for* $W_1(\mu, \nu)$*, and* $u \in 1\text{-}Lip(\Omega)$ *is a potential, then*

$$spt(\gamma) \subset \{(x, y) \in \Omega^2 \mid u(x) - u(y) = |x - y|\}$$

A first observation

Lemma

If $\gamma \in \Pi(\mu, \nu)$ is optimal for $W_1(\mu, \nu)$, and $u \in 1\text{-}Lip(\Omega)$ is a potential, then

$$spt(\gamma) \subset \{(x, y) \in \Omega^2 \mid u(x) - u(y) = |x - y|\}$$

This is just the theorem we had before **translated to the case $p = 1$**.

A first observation

Lemma

If $\gamma \in \Pi(\mu,\nu)$ is optimal for $W_1(\mu,\nu)$, and $u \in 1\text{-}Lip(\Omega)$ is a potential, then

$$spt(\gamma) \subset \{(x,y) \in \Omega^2 \mid u(x) - u(y) = |x-y|\}$$

This is just the theorem we had before **translated to the case $p = 1$**.

Let's **examine this set**!

u is affine on some segments

Lemma

If $u \in 1\text{-}Lip(\Omega)$ *and*

$$u(x) - u(y) = |x - y|,$$

then for all $z \in [x, y] := \{(1 - t)x + ty \mid t \in [0, 1]\}$,

$$u(x) - u(z) = |x - z|.$$

Proof.

□

Transport rays

Definition

We call a segment $[x, y]$ a **transport ray** if

$$u(x) - u(y) = |x - y|, \quad x \neq y$$

and $[x, y]$ is the largest such segment containing x and y.

Examples:

Transport rays are almost disjoint

Lemma

Let $[x, y]$ be a transport ray. Then for all $z \in]x, y[$, $\nabla u(z)$ exists and satisfies

$$\nabla u(z) = \frac{x - y}{|x - y|}.$$

As such, two transport rays can only intersect at their endpoints.

Proof.

$\square$

Ω decomposes into rays

Ω can be **decomposed**[4] **into transport rays** that only intersect at their endpoints.

[4] almost; what about the points in no transport ray?

Ω decomposes into rays

Ω can be **decomposed**[4] **into transport rays** that only intersect at their endpoints.

- By **Rademacher's Theorem**, the set of ray intersections have $\mathcal{L}$ measure 0.

[4] almost; what about the points in no transport ray?

Ω decomposes into rays

Ω can be **decomposed**[4] **into transport rays** that only intersect at their endpoints.

- By **Rademacher's Theorem**, the set of ray intersections have $\mathcal{L}$ measure 0.

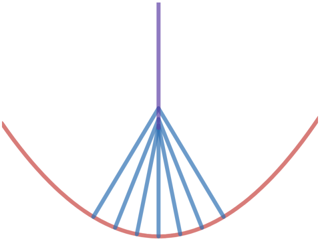


Figure: For u the distance to the parabola $y = x^2$, the blue lines are some transport rays, and the purple line together with the parabola is the set of ray ends.

[4]almost; what about the points in no transport ray?

Sufficient condition for optimality

Lemma

If T is a map satisfying $T_{\#}\mu = \nu$ and for all $x \in \Omega$,

$$u(x) - u(T(x)) = |x - T(x)|$$

then T is optimal.

Proof.

$\square$

Strategy for constructing T

Need to construct a map T such that

- T preserves transport rays

Strategy for constructing T

Need to construct a map T such that

- T preserves transport rays
- T balances mass on each ray (so that $T_{\#}\mu = \nu$) .

Strategy for constructing T

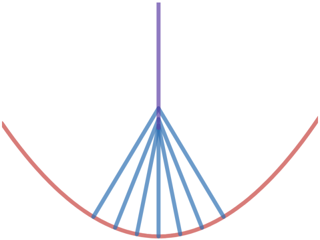
Need to construct a map T such that

- T preserves transport rays
- T balances mass on each ray (so that $T_{\#}\mu = \nu$) .

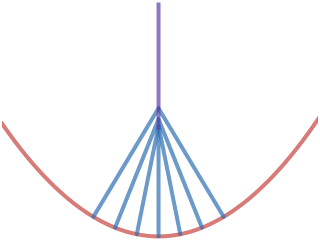
But mass balance is easy for **1-D problems with an AC source**

A clever change of variable

Reduction to proving that μ can be **disintegrated along T-rays** such that we get AC measures on each ray.



Reduction to proving that μ can be **disintegrated along T-rays** such that we get AC measures on each ray.



Using a **Lipschitz change of variable** that straightens rays, can get desired disintegration.

What does u give directly?

Let T be optimal for $W_1(\mu, \nu)$. Then if u is **differentiable** at x,

$$\nabla u(x) = \frac{x - T(x)}{|x - T(x)|}. \tag{1}$$

What does u give directly?

Let T be optimal for $W_1(\mu, \nu)$. Then if u is **differentiable** at x,

$$\nabla u(x) = \frac{x - T(x)}{|x - T(x)|}. \tag{1}$$

So ∇u gives **direction of transport**, not distance.

What does u give directly?

Let T be optimal for $W_1(\mu, \nu)$. Then if u is **differentiable** at x,

$$\nabla u(x) = \frac{x - T(x)}{|x - T(x)|}. \tag{1}$$

So ∇u gives **direction of transport**, not distance.

p	c-transform is easy	a potential gives a map
1	✓	X, but gives direction
> 1	X	✓

Applications of $W_1(\mu, \nu)$ for generative models

Measuring similarity of measures

How do we test if $(G_w)_\# \zeta \approx \nu$?

- Put a metric on $\mathcal{P}(\Omega)$

Measuring similarity of measures

How do we test if $(G_w)_{\#}\zeta \approx \nu$?

- Put a metric on $\mathcal{P}(\Omega)$
- $W_1(\mu, \nu)$ is a convenient choice.

Measuring similarity of measures

How do we test if $(G_w)_{\#}\zeta \approx \nu$?

- Put a metric on $\mathcal{P}(\Omega)$
- $W_1(\mu, \nu)$ is a convenient choice.

Questions

- How do we design G_w to have a hope of approximating ν?

Measuring similarity of measures

How do we test if $(G_w)_\#\zeta \approx \nu$?

- Put a metric on $\mathcal{P}(\Omega)$
- $W_1(\mu, \nu)$ is a convenient choice.

Questions

- How do we design G_w to have a hope of approximating ν?
- How do we compute $W_1((G_w)_\#\eta, \nu)$?

Measuring similarity of measures

How do we test if $(G_w)_\# \zeta \approx \nu$?

- Put a metric on $\mathcal{P}(\Omega)$
- $W_1(\mu, \nu)$ is a convenient choice.

Questions

- How do we design G_w to have a hope of approximating ν?
- How do we compute $W_1((G_w)_\# \eta, \nu)$?
- How do we find a good w?

Neural Networks

We will use **feedforward neural networks** for G_w and to calculate $W_1((G_w)_{\#}\eta, \nu)$

[5] See Leshno et. al. [7]

We will use **feedforward neural networks** for G_w and to calculate $W_1((G_w)_{\#}\eta, \nu)$

Feedforward neural networks are a broad class of **parametrized functions**.

[5] See Leshno et. al. [7]

Neural Networks

We will use **feedforward neural networks** for G_w and to calculate $W_1((G_w)_\#\eta, \nu)$

Feedforward neural networks are a broad class of **parametrized functions**.

- Constructed by **composing simple functions**, called "layers". Usually

$$f(x) = \sigma(Wx + b), \quad \sigma(z_1, \ldots, z_n) = (z_1^+, \ldots, z_n^+)$$

  (W, b) are the parameters of the layer, and the parameters for all layers make up w.

[5] See Leshno et. al. [7]

Neural Networks

We will use **feedforward neural networks** for G_w and to calculate $W_1((G_w)_\#\eta, \nu)$

Feedforward neural networks are a broad class of **parametrized functions**.

- Constructed by **composing simple functions**, called “layers”. Usually

$$f(x) = \sigma(Wx + b), \quad \sigma(z_1, \ldots, z_n) = (z_1^+, \ldots, z_n^+)$$

  (W, b) are the parameters of the layer, and the parameters for all layers make up w.
- Given enough parameters, they can **approximate any continuous function**[5].

[5] See Leshno et. al. [7]

Neural network facts cont.

Feedforward neural networks are a broad class of **parametrized functions**.

Neural network facts cont.

Feedforward neural networks are a broad class of **parametrized functions**.

- The process of finding good parameters w is called **training the network**; usually done by applying stochastic gradient descent to a loss function measuring performance.

Neural network facts cont.

Feedforward neural networks are a broad class of **parametrized functions**.

- The process of finding good parameters w is called **training the network**; usually done by applying stochastic gradient descent to a loss function measuring performance.
- A type of NN known as a **convolutional neural network (CNN)** excels at imaging tasks. For a CNN, general linear maps W are replaced by matrices associated with convolutions.

Neural network facts cont.

Feedforward neural networks are a broad class of **parametrized functions**.

- The process of finding good parameters w is called **training the network**; usually done by applying stochastic gradient descent to a loss function measuring performance.
- A type of NN known as a **convolutional neural network (CNN)** excels at imaging tasks. For a CNN, general linear maps W are replaced by matrices associated with convolutions.
- Huge amounts of **engineering** required in design; not a lot of good math explanations, but that's slowly changing.

Estimating $W_1((G_w)_\#\zeta, \nu)$

The distance $W_1(G_w)_\#\zeta, \nu)$ is estimated by **solving the dual problem**

$$W_1(\mu, \nu) = \sup_{u \in 1-\mathrm{Lip}(\Omega)} \int_\Omega u(d\mu - d\nu) \tag{2}$$

Note the importance of $p = 1$!

Estimating $W_1((G_w)_\#\zeta, \nu)$

The distance $W_1(G_w)_\#\zeta, \nu)$ is estimated by **solving the dual problem**

$$W_1(\mu, \nu) = \sup_{u \in 1-\mathrm{Lip}(\Omega)} \int_\Omega u(d\mu - d\nu) \tag{2}$$

Note the importance of $p = 1$!

Estimate by **parametrizing** $u = u_\theta$, a neural network.

$$\sup_{u \in 1-\mathrm{Lip}(\Omega)} \int_\Omega u(d(G_w)_\#\zeta - d\nu) \approx \sup_{\theta, u_\theta \in 1-\mathrm{Lip}(\Omega)} \int_\Omega u_\theta(d(G_w)_\#\zeta - d\nu). \tag{3}$$

Estimating $W_1((G_w)_\#\zeta, \nu)$

The distance $W_1(G_w)_\#\zeta, \nu)$ is estimated by **solving the dual problem**

$$W_1(\mu, \nu) = \sup_{u \in 1-\mathrm{Lip}(\Omega)} \int_\Omega u(d\mu - d\nu) \tag{2}$$

Note the importance of $p = 1$!

Estimate by **parametrizing** $u = u_\theta$, a neural network.

$$\sup_{u \in 1-\mathrm{Lip}(\Omega)} \int_\Omega u(d(G_w)_\#\zeta - d\nu) \approx \sup_{\theta, u_\theta \in 1-\mathrm{Lip}(\Omega)} \int_\Omega u_\theta(d(G_w)_\#\zeta - d\nu). \tag{3}$$

How is $u_\theta \in 1 - \mathbf{Lip}(\Omega)$ enforced? Researchers have found adding a regularizer works best.

$$\min_\theta \int_\Omega u_\theta(d\nu - d(G_w)_\#\zeta) + \lambda R[\nabla u_\theta] \tag{4}$$

Enforcing $u_\theta \in 1 - \mathrm{Lip}(\Omega)$

Various regularizers are used to **penalize large gradients of u_θ**.

[6] Gulrajani et. al., [4]

Enforcing $u_\theta \in 1 - \mathrm{Lip}(\Omega)$

Various regularizers are used to **penalize large gradients of u_θ.**

One idea[6]: For a suitably chosen distribution σ,

$$\lambda R[\nabla u_\theta] = \lambda \int_\Omega (|\nabla_x u_\theta(x)| - 1)^2 d\sigma(x) \tag{5}$$

[6] Gulrajani et. al., [4]

Enforcing $u_\theta \in 1 - \mathrm{Lip}(\Omega)$

Various regularizers are used to **penalize large gradients of u_θ**.

One idea[6]**:** For a suitably chosen distribution σ,

$$\lambda R[\nabla u_\theta] = \lambda \int_\Omega (|\nabla_x u_\theta(x)| - 1)^2 d\sigma(x) \tag{5}$$

We know $|\nabla u(x)| = 1$ on transport rays, so this regularization makes some sense.

[6] Gulrajani et. al., [4]

Training the potential u_θ

Given a current value of $w = w_0$ and $\theta = \theta_0$,

Training the potential u_θ

Given a current value of $w = w_0$ and $\theta = \theta_0$,

- **Generate fake samples** $\{x_i\}_{i=1}^N$, $x_i = G_{w_0}(z_i)$, $z_i \sim \zeta$.

Training the potential u_θ

Given a current value of $w = w_0$ and $\theta = \theta_0$,

- **Generate fake samples** $\{x_i\}_{i=1}^N$, $x_i = G_{w_0}(z_i)$, $z_i \sim \zeta$.
- **Take real samples** $\{y_i\}_{i=1}^N$ from ν.

Training the potential u_θ

Given a current value of $w = w_0$ and $\theta = \theta_0$,

- **Generate fake samples** $\{x_i\}_{i=1}^N$, $x_i = G_{w_0}(z_i)$, $z_i \sim \zeta$.
- **Take real samples** $\{y_i\}_{i=1}^N$ from ν.
- For each pair x_i, y_i, **sample** $t_i \sim U([0,1])$

Training the potential u_θ

Given a current value of $w = w_0$ and $\theta = \theta_0$,

- **Generate fake samples** $\{x_i\}_{i=1}^N$, $x_i = G_{w_0}(z_i)$, $z_i \sim \zeta$.
- **Take real samples** $\{y_i\}_{i=1}^N$ from ν.
- For each pair x_i, y_i, **sample** $t_i \sim U([0,1])$
- **Approximate**

$$
\begin{aligned}
&\int_\Omega u_{\theta_0}(d\nu - d(G_{w_0})_\# \zeta) + \lambda R[\nabla u_{\theta_0}], \\
&\qquad \approx \frac{1}{N}\sum_{i=1}^N u_{\theta_0}(y_i) - u_{\theta_0}(x_i) + \lambda(||\nabla u_{\theta_0}((1-t_i)x_i + t_i y_i)|| - 1)^2, \\
&\qquad =: \hat{L}(\theta_0)
\end{aligned}
$$

Training the potential u_θ

Given a current value of $w = w_0$ and $\theta = \theta_0$,

- **Generate fake samples** $\{x_i\}_{i=1}^N$, $x_i = G_{w_0}(z_i)$, $z_i \sim \zeta$.
- **Take real samples** $\{y_i\}_{i=1}^N$ from ν.
- For each pair x_i, y_i, **sample** $t_i \sim U([0,1])$
- **Approximate**

$$\int_\Omega u_{\theta_0}(d\nu - d(G_{w_0})_\# \zeta) + \lambda R[\nabla u_{\theta_0}],$$

$$\approx \frac{1}{N}\sum_{i=1}^N u_{\theta_0}(y_i) - u_{\theta_0}(x_i) + \lambda(||\nabla u_{\theta_0}((1-t_i)x_i + t_i y_i)|| - 1)^2,$$

$$=: \hat{L}(\theta_0)$$

- **Update** θ_0 by gradient descent

$$\theta_0^{\text{new}} = \theta_0 - \eta \nabla \hat{L}(\theta_0).$$

Training the potential u_θ

Given a current value of $w = w_0$ and $\theta = \theta_0$,

- **Generate fake samples** $\{x_i\}_{i=1}^N$, $x_i = G_{w_0}(z_i)$, $z_i \sim \zeta$.
- **Take real samples** $\{y_i\}_{i=1}^N$ from ν.
- For each pair x_i, y_i, **sample** $t_i \sim U([0,1])$
- **Approximate**

$$\int_\Omega u_{\theta_0}(d\nu - d(G_{w_0})_\# \zeta) + \lambda R[\nabla u_{\theta_0}],$$

$$\approx \frac{1}{N} \sum_{i=1}^N u_{\theta_0}(y_i) - u_{\theta_0}(x_i) + \lambda(||\nabla u_{\theta_0}((1-t_i)x_i + t_i y_i)|| - 1)^2,$$

$$=: \hat{L}(\theta_0)$$

- **Update** θ_0 by gradient descent

$$\theta_0^{\text{new}} = \theta_0 - \eta \nabla \hat{L}(\theta_0).$$

- **Repeat** until the value of $\hat{L}(\theta)$ stabilizes, or predetermined max iter.

Training the generator G_w

Given initial parameters w_0,

Training the generator G_w

Given initial parameters w_0,

- **Compute** u_{θ_0} using method from last slide.

Training the generator G_w

Given initial parameters w_0,

- **Compute** u_{θ_0} using method from last slide.
- **Generate** fake data $\{G_{w_0}(z_i)\}_{i=1}^N$, $z_i \sim \zeta$, and **sample** real data $\{y_j\}_{j=1}^N$, $y_j \sim \nu$.

Training the generator G_w

Given initial parameters w_0,

- **Compute** u_{θ_0} using method from last slide.
- **Generate** fake data $\{G_{w_0}(z_i)\}_{i=1}^N$, $z_i \sim \zeta$, and **sample** real data $\{y_j\}_{j=1}^N$, $y_j \sim \nu$.
- **Estimate** $W_1((G_{w_0})_\#\zeta, \nu)$ using u_{θ_0} and samples

$$W_1((G_{w_0})_\#\zeta, \nu) \approx \frac{1}{N}\sum_{i=1}^{N} u_{\theta_0}(G_{w_0}(z_i)) - u_{\theta_0}(y_i)$$

Training the generator G_w

Given initial parameters w_0,

- **Compute** u_{θ_0} using method from last slide.
- **Generate** fake data $\{G_{w_0}(z_i)\}_{i=1}^N$, $z_i \sim \zeta$, and **sample** real data $\{y_j\}_{j=1}^N$, $y_j \sim \nu$.
- **Estimate** $W_1((G_{w_0})_\# \zeta, \nu)$ using u_{θ_0} and samples

$$W_1((G_{w_0})_\# \zeta, \nu) \approx \frac{1}{N} \sum_{i=1}^{N} u_{\theta_0}(G_{w_0}(z_i)) - u_{\theta_0}(y_i)$$

- **Perform** gradient descent on estimate of Wasserstein distance

$$w_0^{\text{new}} = w_0 - \epsilon \nabla_w |_{w=w_0} \frac{1}{N} \sum_{i=1}^{N} u_{\theta_0}(G_w(z_i)) \tag{6}$$

Training the generator G_w

Given initial parameters w_0,

- **Compute** u_{θ_0} using method from last slide.
- **Generate** fake data $\{G_{w_0}(z_i)\}_{i=1}^N$, $z_i \sim \zeta$, and **sample** real data $\{y_j\}_{j=1}^N$, $y_j \sim \nu$.
- **Estimate** $W_1((G_{w_0})_\#\zeta, \nu)$ using u_{θ_0} and samples

$$W_1((G_{w_0})_\#\zeta, \nu) \approx \frac{1}{N}\sum_{i=1}^N u_{\theta_0}(G_{w_0}(z_i)) - u_{\theta_0}(y_i)$$

- **Perform** gradient descent on estimate of Wasserstein distance

$$w_0^{\text{new}} = w_0 - \epsilon \nabla_w|_{w=w_0} \frac{1}{N}\sum_{i=1}^N u_{\theta_0}(G_w(z_i)) \tag{6}$$

- **Repeat** until samples $\{G_{w_0}(z_i)\}_{i=1}^N$ are of sufficient visual quality.

More WGAN Results

Figure: More results from a different dataset.[7]

[7] from Karras et. al. [6]

Open questions for WGANs

Despite this empirical success, there are many theoretical mysteries:

Open questions for WGANs

Despite this empirical success, there are many theoretical mysteries:

- The optimization problems for finding w and θ are massively high dimensional and non-convex; **why does gradient descent with sampling (SGD) work so well?**

Open questions for WGANs

Despite this empirical success, there are many theoretical mysteries:

- The optimization problems for finding w and θ are massively high dimensional and non-convex; **why does gradient descent with sampling (SGD) work so well?**
- Does solving
$$\min_{\theta} \int_{\Omega} u_{\theta}(d\nu - d(G_{w_0})_{\#}\zeta) + \lambda R[\nabla u_{\theta}]$$
actually produce a Kantorovich potential?

Open questions for WGANs

Despite this empirical success, there are many theoretical mysteries:

- The optimization problems for finding w and θ are massively high dimensional and non-convex; **why does gradient descent with sampling (SGD) work so well?**
- Does solving
$$\min_{\theta} \int_{\Omega} u_{\theta}(d\nu - d(G_{w_0})_{\#}\zeta) + \lambda R[\nabla u_{\theta}]$$
actually produce a Kantorovich potential?
- The W_1 distance is known to have **horrible sample complexity**; how do we get good results despite this?

Open questions for WGANs

Despite this empirical success, there are many theoretical mysteries:

- The optimization problems for finding w and θ are massively high dimensional and non-convex; **why does gradient descent with sampling (SGD) work so well?**
- Does solving
$$\min_{\theta} \int_{\Omega} u_\theta (d\nu - d(G_{w_0})_{\#}\zeta) + \lambda R[\nabla u_\theta]$$
actually produce a Kantorovich potential?
- The W_1 distance is known to have **horrible sample complexity**; how do we get good results despite this?
- In reality we do not train θ to completion before updating w; how do the dynamics of these two descent schemes **affect each other**?

Open questions for WGANs

Despite this empirical success, there are many theoretical mysteries:

- The optimization problems for finding w and θ are massively high dimensional and non-convex; **why does gradient descent with sampling (SGD) work so well?**
- Does solving
$$\min_{\theta} \int_{\Omega} u_\theta (d\nu - d(G_{w_0})_{\#}\zeta) + \lambda R[\nabla u_\theta]$$
actually produce a Kantorovich potential?
- The W_1 distance is known to have **horrible sample complexity**; how do we get good results despite this?
- In reality we do not train θ to completion before updating w; how do the dynamics of these two descent schemes **affect each other**?

We should also consider the **ethical implications**.

Summary

We discussed how W_p for $p > 1$ **compares to** W_1

Summary

We discussed how W_p for $p > 1$ **compares to W_1**

- With $p > 1$, a potential gives an optimal map, whereas for $p = 1$ a potential gives **only direction of transport**.

Summary

We discussed how W_p for $p > 1$ **compares to** W_1

- With $p > 1$, a potential gives an optimal map, whereas for $p = 1$ a potential gives **only direction of transport**.
- With $p = 1$, the c-transform is easier to compute; this is why $p = 1$ is more **popular in ML**.

Summary

We discussed how W_p for $p > 1$ **compares to W_1**

- With $p > 1$, a potential gives an optimal map, whereas for $p = 1$ a potential gives **only direction of transport**.
- With $p = 1$, the c-transform is easier to compute; this is why $p = 1$ is more **popular in ML**.

We sketched a method for **constructing an optimal map** for $W_1(\mu, \nu)$

Summary

We discussed how W_p for $p > 1$ **compares to W_1**

- With $p > 1$, a potential gives an optimal map, whereas for $p = 1$ a potential gives **only direction of transport**.
- With $p = 1$, the c-transform is easier to compute; this is why $p = 1$ is more **popular in ML**.

We sketched a method for **constructing an optimal map** for $W_1(\mu, \nu)$

- We **decompose the space into transport rays**, and solve the resulting 1-D problems.

Summary

We discussed how W_p for $p > 1$ **compares to W_1**

- With $p > 1$, a potential gives an optimal map, whereas for $p = 1$ a potential gives **only direction of transport**.
- With $p = 1$, the c-transform is easier to compute; this is why $p = 1$ is more **popular in ML**.

We sketched a method for **constructing an optimal map** for $W_1(\mu, \nu)$

- We **decompose the space into transport rays**, and solve the resulting 1-D problems.

We went over the **algorithm for training WGANs**.

Summary

We discussed how W_p for $p > 1$ **compares to** W_1

- With $p > 1$, a potential gives an optimal map, whereas for $p = 1$ a potential gives **only direction of transport**.
- With $p = 1$, the c-transform is easier to compute; this is why $p = 1$ is more **popular in ML**.

We sketched a method for **constructing an optimal map** for $W_1(\mu, \nu)$

- We **decompose the space into transport rays**, and solve the resulting 1-D problems.

We went over the **algorithm for training WGANs**.

- Many **open questions**, and serious **ethical issues**.

M. Arjovsky, S. Chintala, and L. Bottou.
Wasserstein GAN.
arXiv preprint arXiv:1701.07875, 2017.

L. Caffarelli, M. Feldman, and R. McCann.
Constructing optimal maps for monge's transport problem as a limit of strictly convex costs.
Journal of the American Mathematical Society, 15(1):1–26, 2002.

L. C. Evans and W. Gangbo.
Differential equations methods for the Monge-Kantorovich mass transfer problem.
Number 653. American Mathematical Soc., 1999.

I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville.
Improved training of Wasserstein GANs.
In *Advances in neural information processing systems*, pages 5767–5777, 2017.

V. Hartmann and D. Schuhmacher.
Semi-discrete optimal transport: a solution procedure for the unsquared euclidean distance case.
Mathematical Methods of Operations Research, pages 1–31, 2020.

T. Karras, T. Aila, S. Laine, and J. Lehtinen.
Progressive growing of gans for improved quality, stability, and variation.
arXiv preprint arXiv:1710.10196, 2017.

M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken.
Multilayer feedforward networks with a nonpolynomial activation function can approximate any function.
Neural Networks, 1993.

V. N. Sudakov.
Geometric problems in the theory of infinite-dimensional probability distributions, volume 141.
American Mathematical Soc., 1979.

N. S. Trudinger and X.-J. Wang.
On the monge mass transfer problem.
Calculus of Variations and Partial Differential Equations, 13(1):19–31, 2001.