

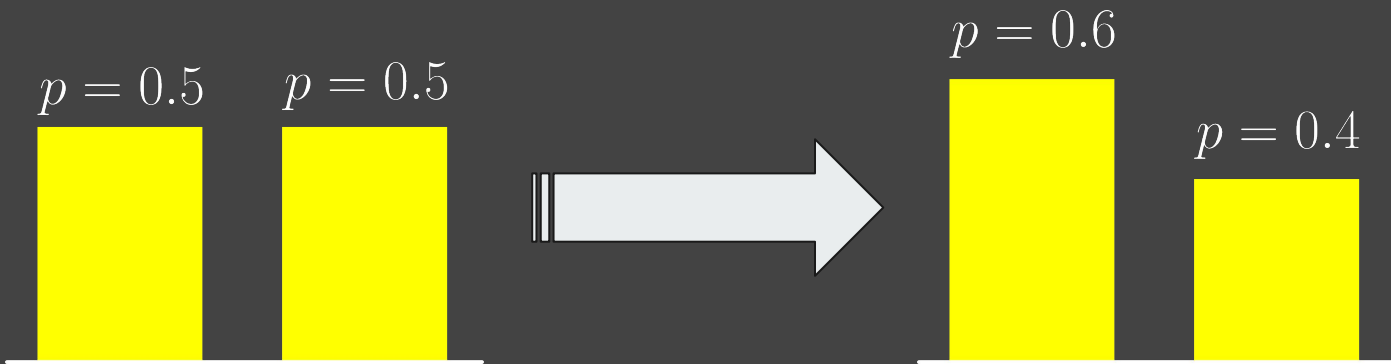


# **A Flow Approach for Learning a Conditional Probability Distribution**



# SAMPLING

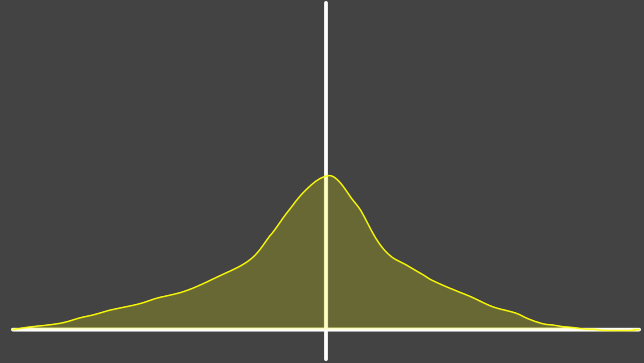
- Given a probability distribution,
- get samples from that distribution,
- assuming we are able to sample some fixed distribution.



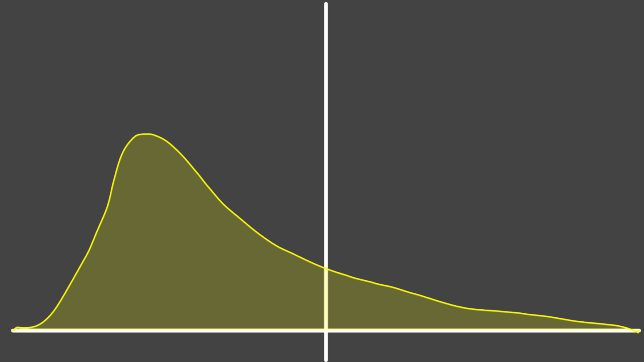
# SAMPLING - EXAMPLE



We **can** sample:

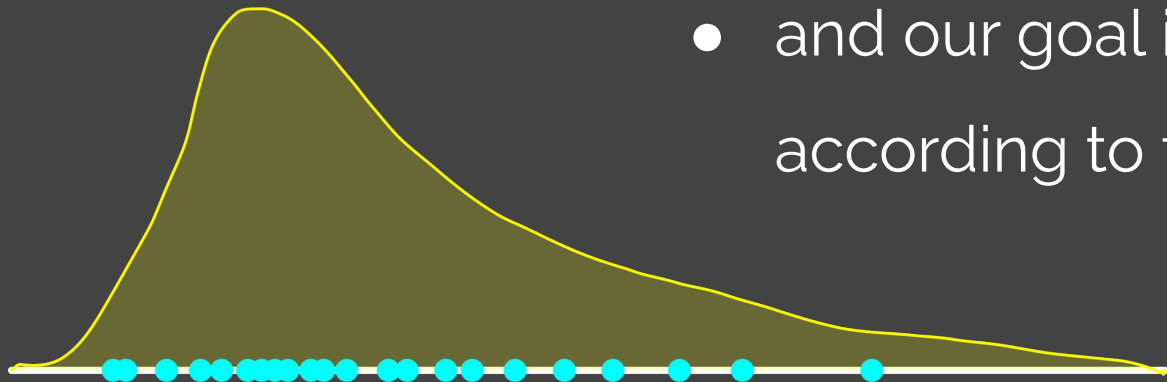


We **want to** sample:



# LEARNING

- instead of being given a distribution,  
we are given ***samples*** from that distribution

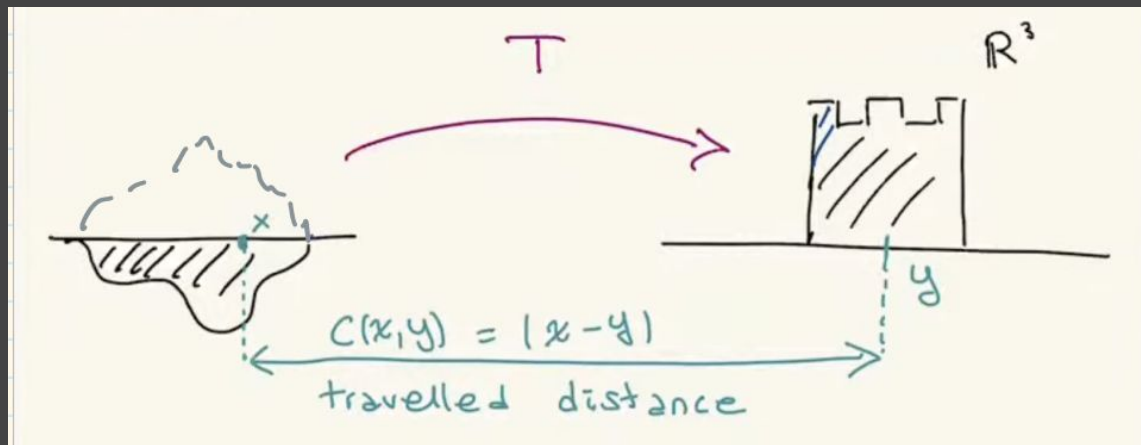


- and our goal is to create samples  
according to the same distribution

# Optimal transport framework

Monge, 1781

Q: assume one extracts soil from the ground to build fortifications, which is the the cheapest possible way to transport the soil?



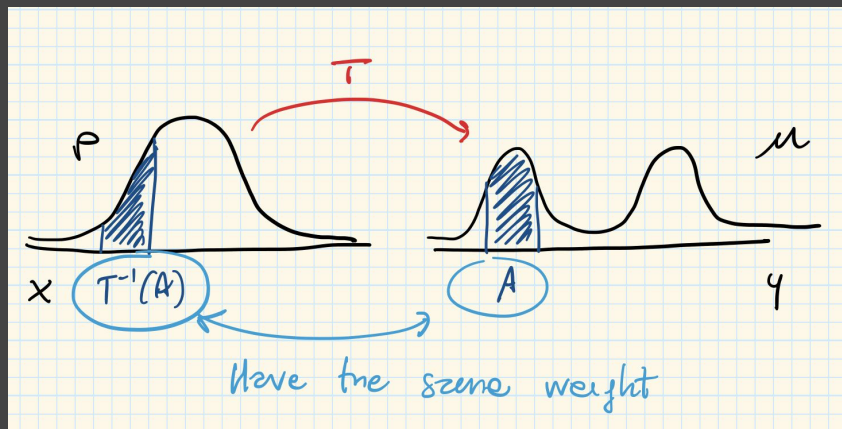
# Math formulation of OT (Monge)

Given two probability densities,  $\rho(z)$  and  $\mu(z)$ ,  $z \in \mathbb{R}^d$ , find a 1-to-1 map  $T: \mathbb{R}^d \rightarrow \mathbb{R}^d$  such that  $T\#\rho = \mu$  and that minimizes the functional:

$$M[T] = \int_{\mathbb{R}^d} c(\mathbf{z}, T(\mathbf{z})) \rho(\mathbf{z}) d\mathbf{z}$$

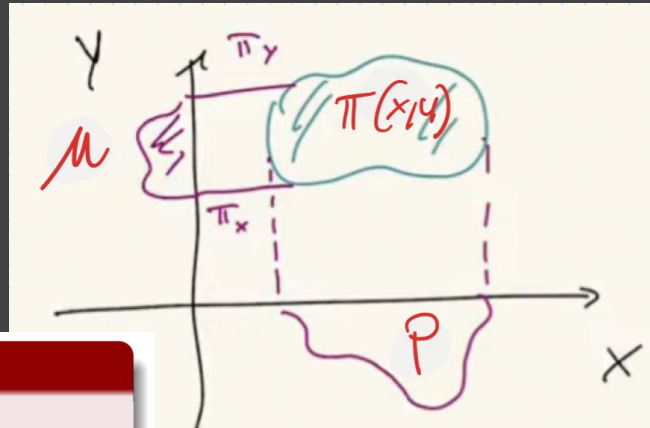
The minimizer  $T$  is called an **OT map**

But what does  $T\#\rho = \mu$  mean?



# Math formulation of OT (Kantorovich) ≡

Relaxation where the mass reaching each point  $y$  may come from various points  $x$  and, conversely, the mass from each point  $x$  may be split into various destinations  $y$ .



## Problem

Minimize the functional

$$F(\pi) = \int c(\mathbf{z}, \mathbf{z}') \pi(\mathbf{z}, \mathbf{z}') d\mathbf{z} d\mathbf{z}'$$

over all the joint probability distributions satisfying

$$\rho(\mathbf{z}) = \int \pi(\mathbf{z}, \mathbf{z}') d\mathbf{z}' \quad \mu(\mathbf{z}') = \int \pi(\mathbf{z}, \mathbf{z}') d\mathbf{z}$$

The minimizer  $\pi$  is called an **optimal plan**

# Results of OT



Under mild assumptions (finite second moments and a convex cost), the Monge and the Kantorovich problem have the same unique solution, in the sense that:

$$\min M(T) = \min F(\pi)$$

$$M(T) = \int c(z, T(z)) \rho(z) dz$$
$$F(\pi) = \int c(z, z') \pi(z, z') dz dz'$$

This is a primal approach to the problem, however it is easier to solve it through its dual:



# Dual Formation



A simple analogy, on the purple side we have a function to **maximize** with respect to a set of constraints, while on the black side we have to **minimize** a function with respect to another set of constraints.

The crucial observation is that both of them agree on the optimised i.e minimal and maximal.

Similarly we want to reconstruct our primal problem which agrees the primal problem on optimized solution, that is the dual problem.

# Dual Formation

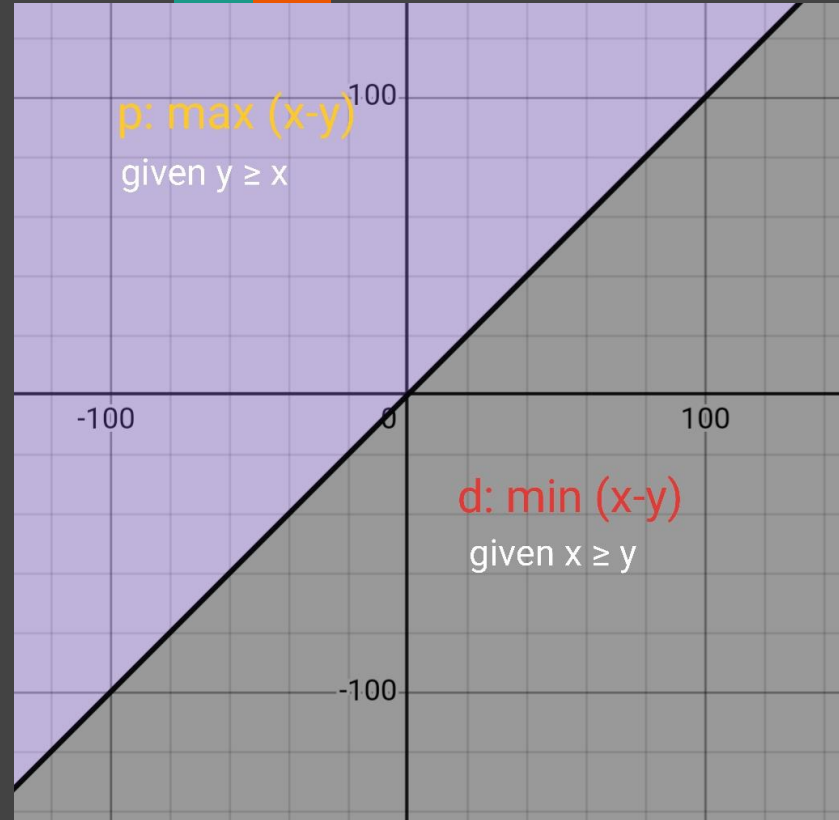
Primal : **max (f)**  
**Constraints C**

Dual : **min (g)**  
**Constraints D**

Strong duality says that  
**max (f) = min (g)**

p: **max (x-y)**  
given  $y \geq x$

d: **min (x-y)**  
given  $x \geq y$



$$\left\{ \begin{array}{l} \min_{\pi} F(\pi) \\ \int \pi(\mathbf{z}, \mathbf{z}') d\mathbf{z} = \mu(\mathbf{z}') \\ \int \pi(\mathbf{z}, \mathbf{z}') d\mathbf{z}' = \rho(\mathbf{z}) \end{array} \right. \longleftrightarrow \left\{ \begin{array}{l} \min_{\pi} \sum_{i,j} \pi_{ij} c_{ij} \\ \sum_i \pi_{ij} = \mu_j \\ \sum_j \pi_{ij} = \rho_i \end{array} \right.$$

$$\Downarrow \qquad \qquad \qquad \Downarrow$$

$$\left\{ \begin{array}{l} \max_{u,v} L(u, v) \\ u(x) + v(y) \leq c(x, y) \end{array} \right. \longleftrightarrow \left\{ \begin{array}{l} \max_{u,v} \sum u_i x_i + \sum v_j y_j \\ u_i + v_j \leq c_{ij} \end{array} \right.$$

where

$$F(\pi) = \int c(\mathbf{z}, \mathbf{z}') \pi(\mathbf{z}, \mathbf{z}') d\mathbf{z} d\mathbf{z}'$$

$$L(u, v) = \int u(\mathbf{z}) \rho(\mathbf{z}) d\mathbf{z} + \int v(\mathbf{z}') \mu(\mathbf{z}') d\mathbf{z}'$$

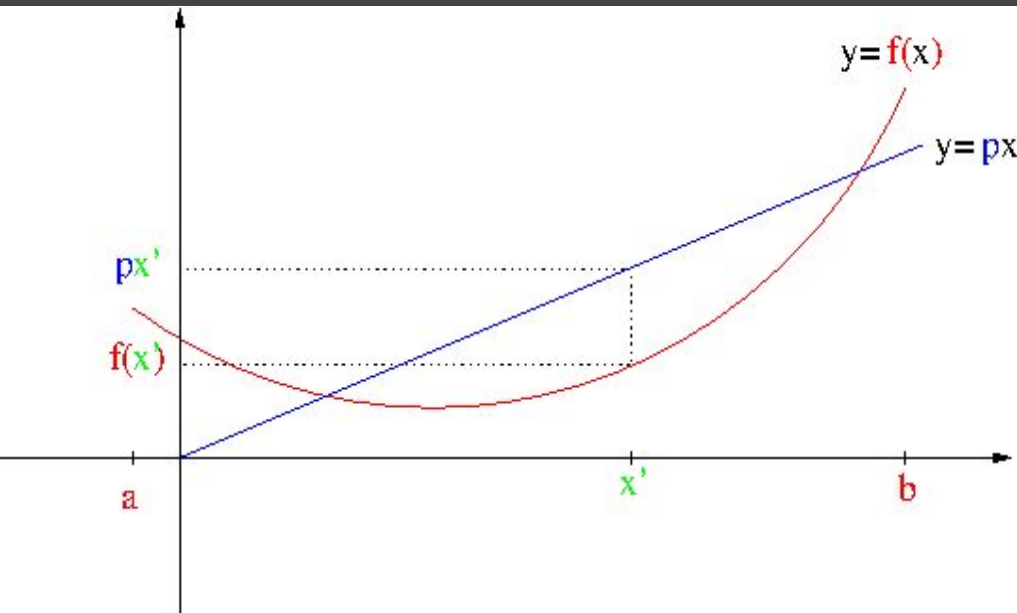


$\pi_{ij} = \pi(x_i, y_j)$  = Amount of sand gone to  $j^{\text{th}}$  Bucket from  $i^{\text{th}}$  heap.

$$\text{Cost} = c_{ij}$$

Easy to observe:  $\sum_j \pi_{ij} = x_i$ ,  $\sum_i \pi_{i,j} = y_j$

# Legendre-Fenchel Transform



$$f^*(p) = \max_x (px - f(x))$$

L-F Transform is for the convex functions which basically catches the maximum linear difference between a secant of slope  $p$  and the curve  $f(x)$ .

For our optimization we are interested in pair of functions  $(u, v)$  such that  $u^* = v$  and  $v^* = u$ .

**Such a pair is termed as Legendre Duals.**

The reason of this will be clear next slide!!

$$u(x) \rightarrow \frac{\|x\|^2}{2} - u(x)$$

$$v(y) \rightarrow \frac{\|y\|^2}{2} - v(y)$$

If  $c(\mathbf{z}, \mathbf{z}') = \|\mathbf{z} - \mathbf{z}'\|^2$  the problem can be rewritten as

$$\begin{cases} \min_{u,v} \int u(\mathbf{z})\rho(\mathbf{z})d\mathbf{z} + \int v(\mathbf{z}')\mu(\mathbf{z}')d\mathbf{z}' \\ u(\mathbf{z}) + v(\mathbf{z}') \geq \mathbf{z} \cdot \mathbf{z}' \end{cases}$$

The functional  $L(u, v)$  admits a unique minimizer  $(\bar{u}, \bar{v})$  which are convex conjugates

$$\begin{cases} \bar{u}(\mathbf{z}) = \max_{\mathbf{z}' \in \mathbb{R}^d} (\mathbf{z} \cdot \mathbf{z}' - v(\mathbf{z}')) \\ \bar{v}(\mathbf{z}') = \max_{\mathbf{z} \in \mathbb{R}^d} (\mathbf{z} \cdot \mathbf{z}' - u(\mathbf{z})) \end{cases}$$

## Theorem (Gangbo, McCann, Chartrand et al. )

*For smooth  $\rho$  and  $\mu$ , the functional*

$$\tilde{L}[u] = \int u(\mathbf{z})\rho(\mathbf{z})d\mathbf{z} + \int u^*(\mathbf{z}')\mu(\mathbf{z}')d\mathbf{z}'$$

*has a unique, convex minimizer  $\bar{u}$  and its gradient represents the solution of the Monge-Kantorovich problem.*

This is saying that if  $\rho$  and  $\mu$  are regular enough the optimal plan  $\pi$  on the Kantorovich problem reduces to a (optimal) map.

# Gist Of Using Optimal Transport

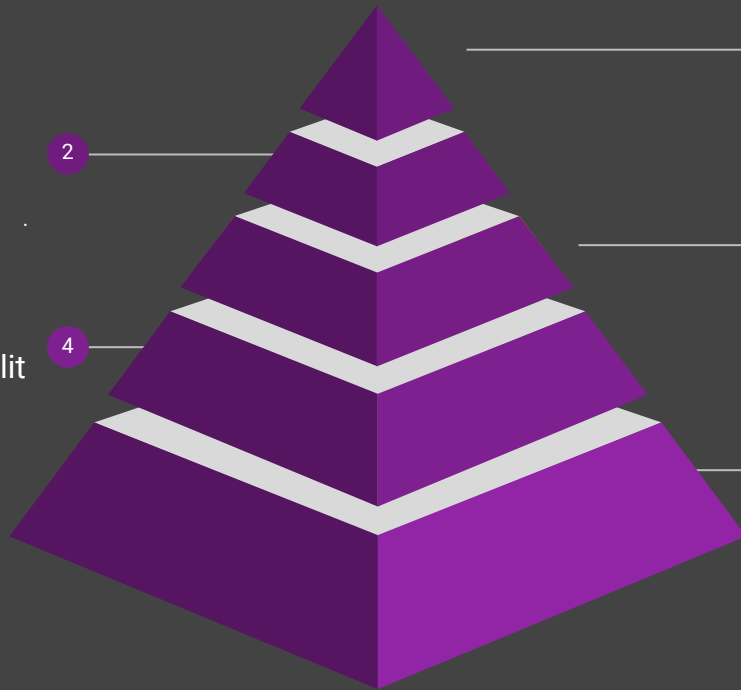


## Dual Problem

We construct the dual problem to our primal.

## Kantorovich Formation

We allowed our mass to split



## Refining

We use Legendre Transform to eliminate a large possible solutions and just focus on convex duals

## Primal Problem

We construct our desired problem

## Monge Formation

We had restriction that we cannot split the mass.





# Our Concrete Problem

**Motivation:** Evolving states over time, intricate likelihood models

We need to update our belief for state  $x$  as we sequentially collect observations  $y_1, y_2, \dots$

(i.e. transition from a prior  $\rho(x_t | y_{1:t-1})$  to posterior distribution  $\rho(x_t | y_{1:t})$ )

**Goal:** Given samples  $(x^i, y^i)$  drawn from the unknown  $\rho(x, y)$  we want samples of  $\rho(x|y)$

**Solution:** Triangular maps within the realm of OT



## What is a Triangular Map?

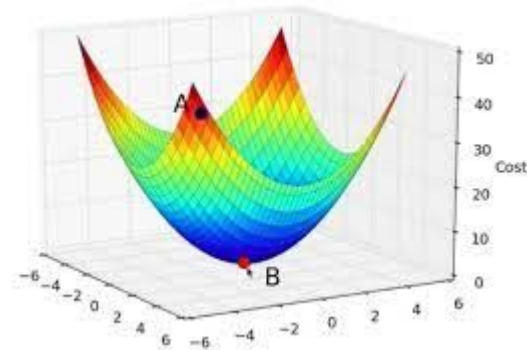
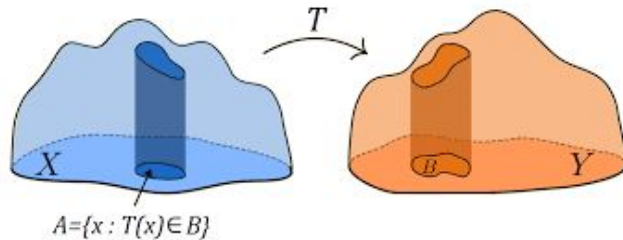
$$\mathcal{T} : \mathbb{R}^{d+m} \rightarrow \mathbb{R}^{d+m} \quad \mathcal{T}(\mathbf{y}, \mathbf{x}) = \begin{bmatrix} T_{\mathbf{Y}}(\mathbf{y}) \\ T_{\mathbf{X}}(\mathbf{y}, \mathbf{x}) \end{bmatrix}$$

$\mathbf{x} \rightarrow T_{\mathbf{X}}(\mathbf{y}, \mathbf{x})$  maps  $p_{\mathbf{X}}(\mathbf{x})$  into  $p(\mathbf{x}|\mathbf{y})$  for each  $\mathbf{y}$ .

# Triangular Map is a Limiting Solution to OT

Weighted  $L^2$  Cost Function:

$$c((x, y), (x', y')) = \frac{1}{2}(\lambda ||y - y'||^2 + ||x - x'||^2)$$



As  $\lambda \rightarrow \infty$ ,  $y$  values are mapped with the lowest cost (i.e. closest values) while  $x$  are transported based on both  $x$  and mapped  $y$  values.

## Resulting Generative Flow

- Composition of simple elementary maps in the block- triangular form
- Defines transformation  $T = T_K \circ \dots \circ T_2 \circ T_1$   
from  $p(y, x) = \mu(y)\mu(x)$  to  $\mu(y, x) = \mu(y)\mu(x|y)$
- Satisfies the push-forward condition  $T_{\#}p(y, x) = \mu(y, x)$
- Used to sample any conditional of the target measure

**Theorem 2.4.** Consider a reference  $\eta = \eta_{\mathcal{W}} \otimes \eta_{\mathcal{V}} \in \mathbb{P}(\mathcal{S})$  and a target  $\nu \in \mathbb{P}(\mathcal{Z})$  and let  $T$  be a block triangular map of the form (2.1) satisfying  $T_{\#}\eta = \nu$ . Then for  $F_{\#}\eta_{\mathcal{W}}$ -a.e.  $y$  it holds that  $G(y, \cdot)_{\#}\eta_{\mathcal{V}} = \nu(\cdot|y)$ .



# Maximization Process

$$\max_u \tilde{L}$$

## First Step

Gradient ascent using functional derivative

$$u^*(z') = \max_z (z * z' - u(z))$$

## Dual Problem Review

Goal: maximize over  $u$

$$u_{t+1} = u_t + \alpha \nabla_u \tilde{L}, \quad 0 < \alpha < 1$$

## Issues

Problematic Legendre Transform—can't always take gradient.

# Gradient Flow

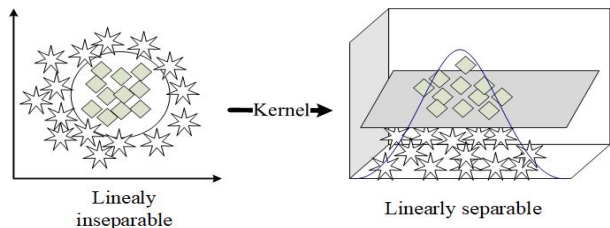
## Workaround

- Take Gradient WRT  $z$  on both sides—indirectly maximizing the objective
- Can be shown that derivative can be expressed as difference of probability measures
- But we don't know the measures pointwise, so need to approximate these gradients

$$z_{t+1} = z_t - \alpha (\nabla h)^{-1} \nabla_z \frac{\tilde{L}_t}{\delta u} | u_t$$

$$u_t = \frac{\|z\|}{2} \forall t : z_{t+1} = z_t - \alpha \nabla_z (\rho_t - \mu)(z_t)$$

# Kernel and RKHS Projection



img: Sushilkumar Yadav (Medium)

Kernels project points into higher dimension

Idea: use Reproducing Kernel Hilbert Spaces  
Why **RKHS**? It's a nice type of kernel that allows us to work in finite dimension

By projecting our data into a *Reproducing Kernel Hilbert Space* **k**, we can use kernel density estimation to approximate the gradients using samples

## 01 | Gradient of measures to gradient of kernels

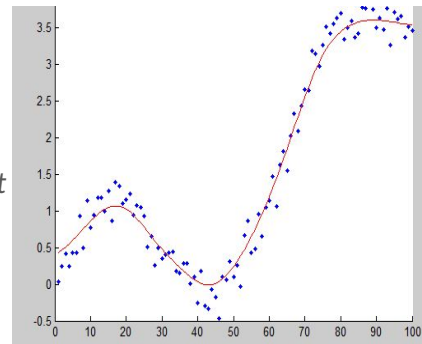
$$\nabla_z(\rho_t - \mu) \approx \int \nabla_z k(z, z')(\rho_t - \mu)(z') dz'$$

## 02 | Expected Value Interpretation

$$E_{\rho_t}(\nabla_z k(z, z')) - E_{\mu}(\nabla_z k(z, z'))$$

## 03 | Monte Carlo Approximation

$$\frac{1}{N} \sum_{i=1}^N \nabla_z \psi(z, z_t^i) + \frac{1}{M} \sum_{j=1}^M \nabla_z k(z, z'^j)$$



$$z_{t+1} = z_t - \alpha(LL^k)$$

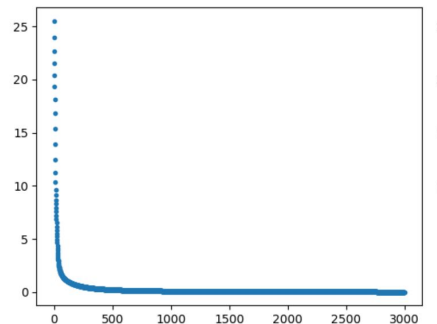
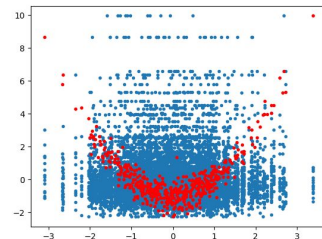
# Implementation Algorithm

**Input:** samples  $\{(y_i, x_i)\}_{i=1}^N$  from joint distribution  $\mu(y, x)$

1. Generate  $p(y, x) = \mu(y)\mu(x)$  and target  $\mu(y, x)$  samples
2. Loop until iteration limit reached or gradient shrinks close to 0
  - a. Compute new  $LL^k$  (approximated gradient. Depends on kernel)
  - b. Multiply gradient by the matrix  $\begin{bmatrix} 1 & 0 \\ 0 & 0.0001 \end{bmatrix}$  to discourage movement in y-direction (enforce triangular map)
  - c. Move points using triangular gradient, which now favors x-dir movement
  - d. Calculate magnitude of gradient (how much pts must move). If small enough, leave loop

$$p(y, x) = \mu(y)\mu(x)$$

target  $\mu(x, y)$



Norm of gradient over time

$$LL^k \equiv \frac{1}{N} \sum_{i=1}^N \nabla k(\mathbf{z}, \mathbf{z}_t^i) - \frac{1}{M} \sum_{j=1}^M \nabla k(\mathbf{z}, \mathbf{z}'^j).$$

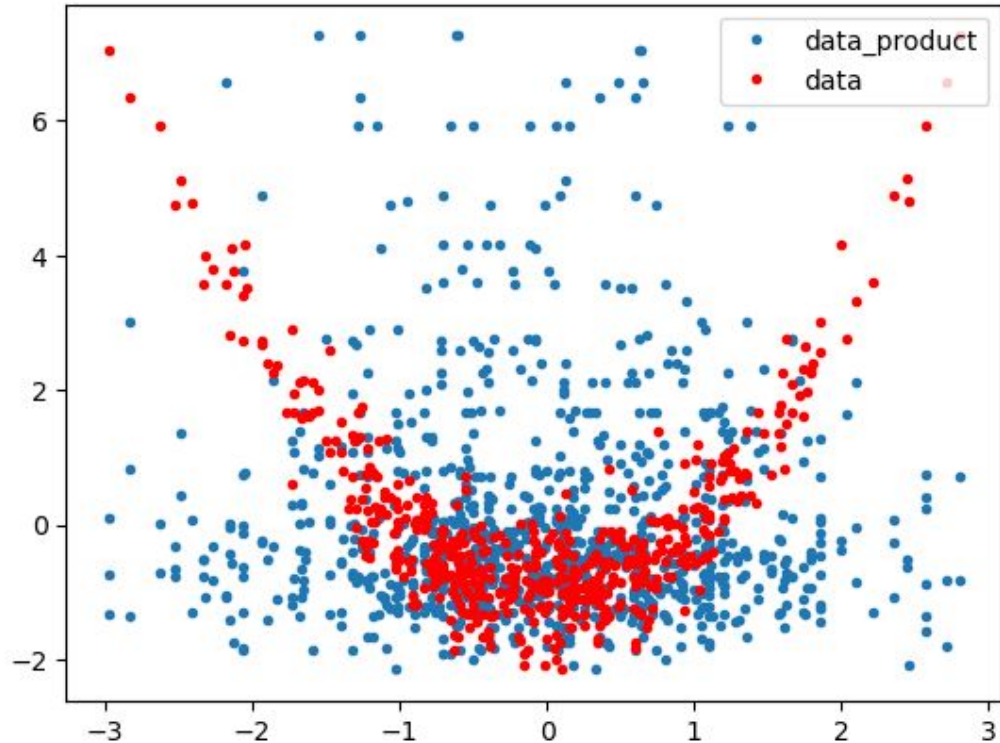
$$\begin{bmatrix} 1 & 0 \\ 0 & 0.0001 \end{bmatrix} * LL^k = \text{amt by which to move points}$$



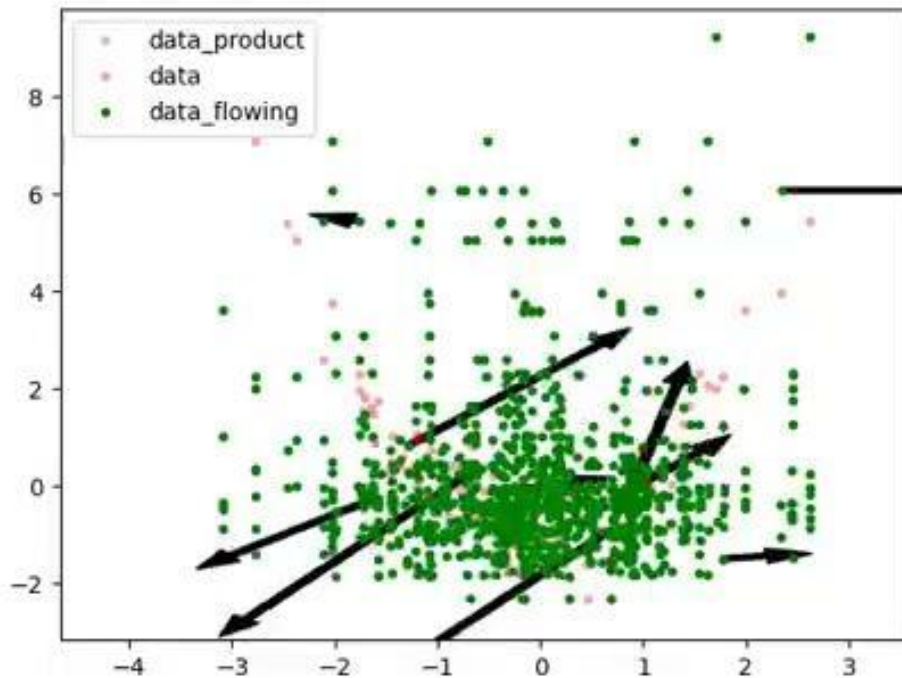
# Numerical Example

$X \sim \text{Gaussian}(\dots)$

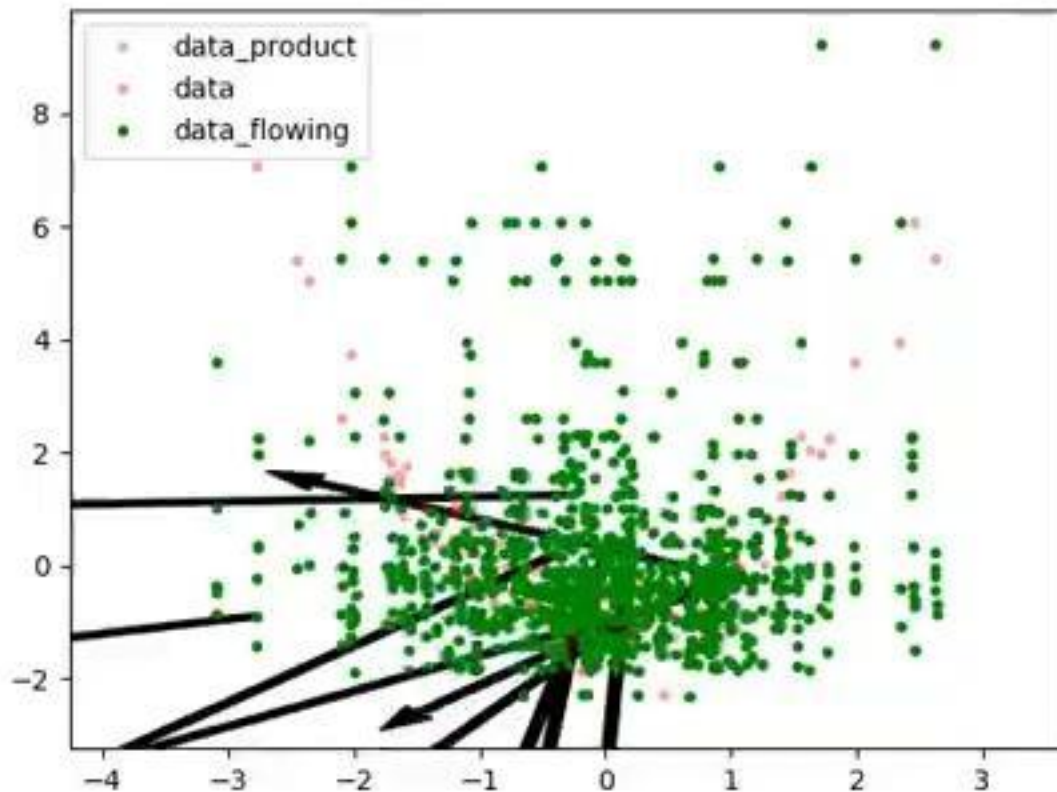
$Y \sim X^2 - 1 + \text{Gaussian}(\dots)$



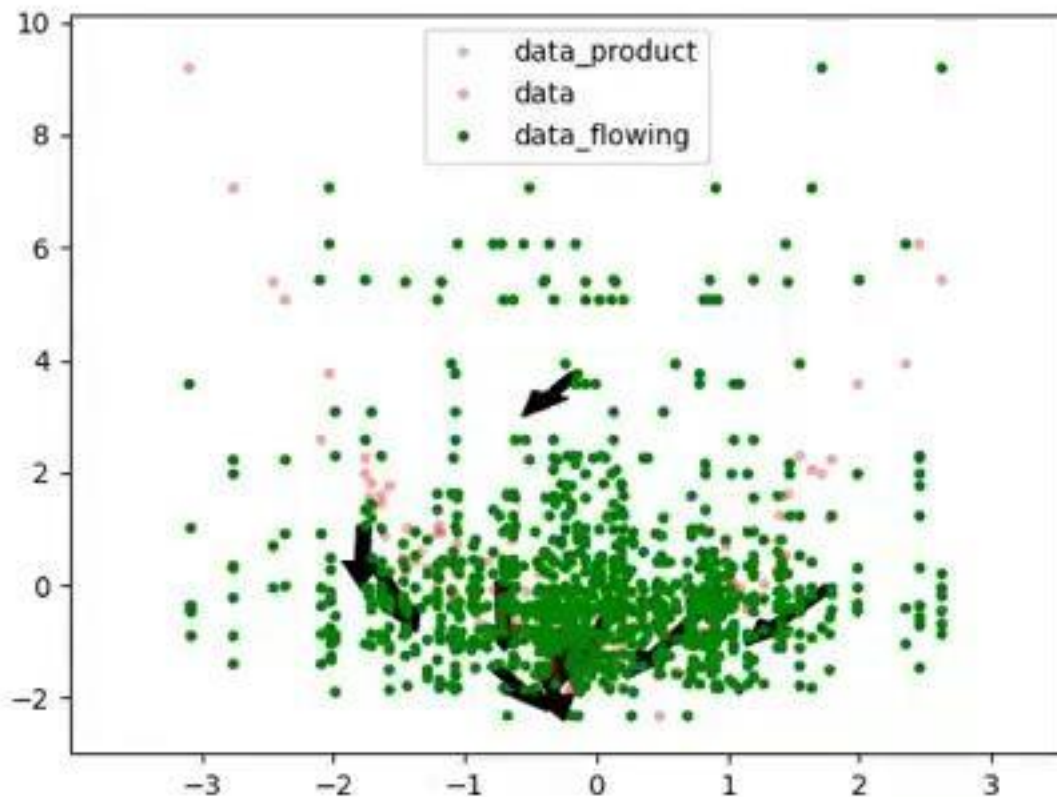
$$k(x, y) = \exp \left( -\frac{\|x - y\|^2}{0.05} \right)$$




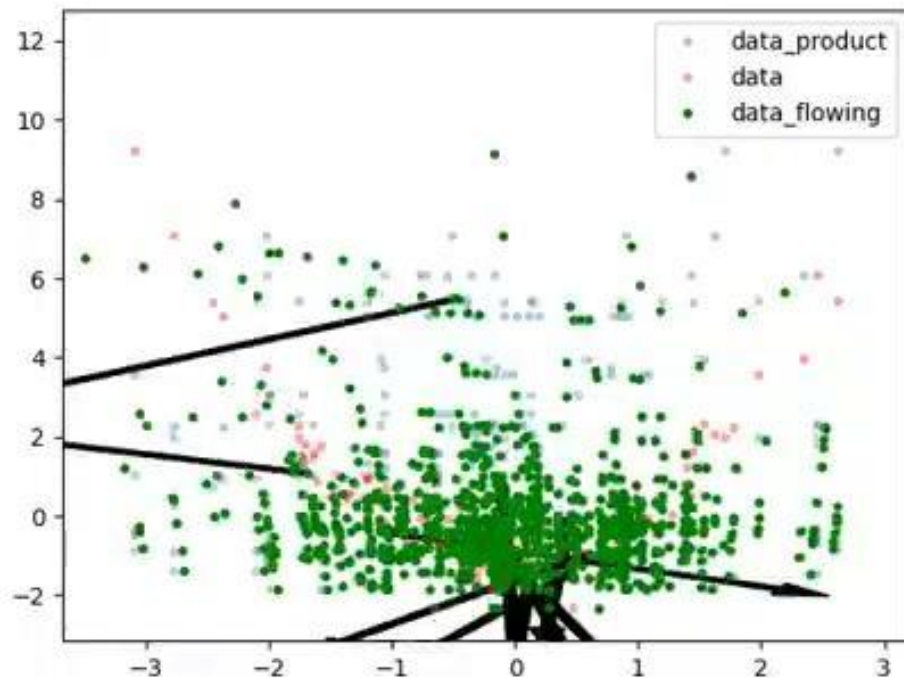
$$k(x, y) = \exp \left( -\frac{\|x - y\|^2}{1} \right)$$



$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{20}\right)$$

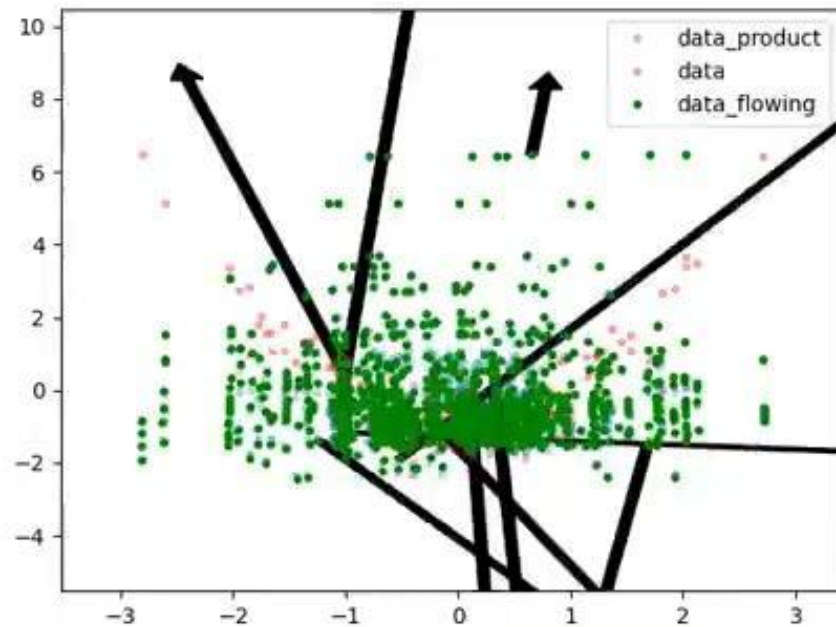



$$k(x, y) = (0.1 \cdot x \cdot y + 1)^4$$



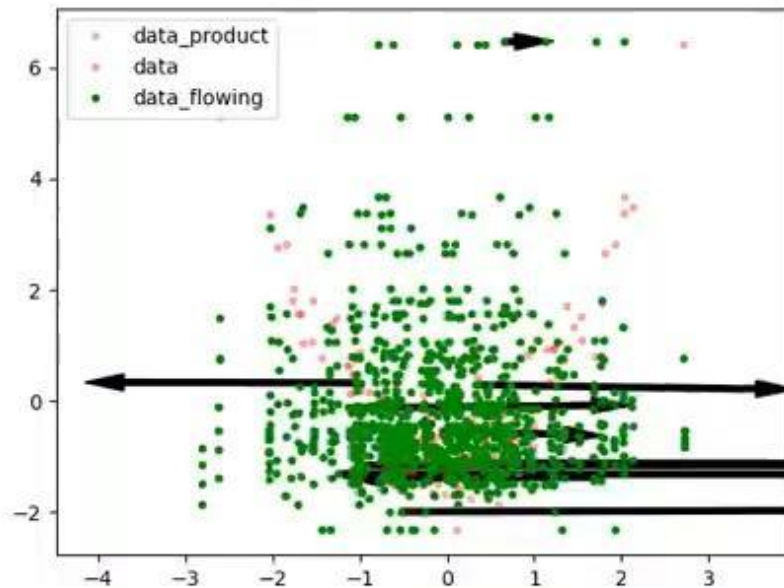


low lambda





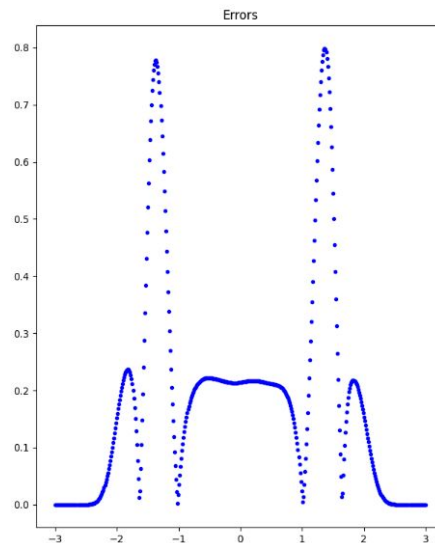
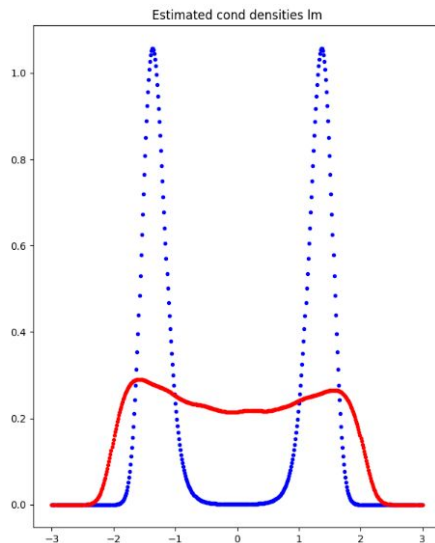
high lambda



# Errors

Predicted density  
from our  
triangular map

True density



Difference

Goal: flatten this  
curve by  
implementing  
RKHS as the  
kernel in our  
algorithm





# Thank you.

Questions?

