Optimal Transport

Yanhua Huang

Apr 2018

In generative models, with an assumption on the latent variable space, e.g., Gaussian distribution, we want to find a mapping from the latent space to the data space such that the mapped distribution can fit the raw data. However, following MLE, it is difficult to calculate the KL divergence because supports before and after mapping are mismatched. One approach to address this issue are using invertible mappings such as normalization flows. Optimal transport gives another perspective with metrics on different supports. In this blog, we will give a brief introduction to optimal transport as well as one of its well-known applications, Wasserstein distance, based on `http://www.damtp.cam.ac.uk//user/mt748/Notes.pdf`.

Formulations

Definition of Transport The map $T : X \to Y$ that transports $\mu \in \mathcal{P}(X)$ to $\nu \in \mathcal{P}(Y)$ is a transport map if $\nu(\mathcal{B}) = \mu(T^{-1}(\mathcal{B}))$ for all ν-measurable sets $\mathcal{B}$. As shorthand we write $\nu = T_{\#}\mu$.

Given a cost function $c : X \times Y \to [0, +\infty]$ that measures the cost of transporting one unit of mass from X to Y. The optimal transport problem is how to transport μ to ν whilst minimizing the cost. One of the formulations of optimal transport problem is Monge.

Monge's formulation Given $\mu \in \mathcal{P}(X)$ to $\nu \in \mathcal{P}(Y)$, Monge's optimal transport problem is to find a μ-measurable map $T : X \to Y$ that minimizes $\int_X c(x, T(x))\mathrm{d}\mu(x)$ subject to $\nu = T_{\#}\mu$.

Note that masses are not split in Monge formulation. There is another general formulation defined as follows.

Kantorovich's formulation Given $\mu \in \mathcal{P}(X)$ to $\nu \in \mathcal{P}(Y)$, Kantorovich's optimal transport problem is to find a measurable map π over $\mathcal{P}(X \times Y)$ that minimizes $\int_{X\times Y} c(x, y)\mathrm{d}\pi(x, y)$.

Special case: convex cost function

Assume F and G are CDFs of μ and ν respectively, i.e., $F(x) = \int_{-\infty}^{x} \mathrm{d}\mu = \mu((-\infty, x])$. We define the generalised inverse of F as $F^{-1}(t) = \inf_{x\in\mathbb{R}} F(x) >$

t. Assume the cost function $c(x, y) = d(x - y)$ where d is convex and continuous. We have the following theorem.

Theorem 1. *Let π be the measure on $\mathbb{R}^2$ with CDF $H(x, y) = \min(F(x), G(y))$. Then π is the solution for Kantorovich's optimal transport problem with cost function c. Moreover the optimal transport cost is $\int_0^1 d(F^{-1}(t) - G^{-1}(t))\mathrm{d}t$.*

Notice that for such cost function, the Kantorovich's problem is convex. In particular, consider discrete measures $\mu = \sum_{i=1}^m \alpha_i \theta_{x_i}$ and $\nu = \sum_{j=1}^n \beta_j \theta_{y_j}$, where $1 = \sum_{i=1}^m \alpha_i = \sum_{j=1}^n \beta_j$. Then the Kantorovich problem is just to solve the linear programme

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c(x_i, y_j)\pi(x_i, y_j), \text{ s.t.} \beta_j = \sum_{i=1}^{m} \pi(x_i, y_j), \alpha_i = \sum_{j=1}^{n} \pi(x_i, y_j). \quad (1)$$

Special case: 0-1 loss function

Assume the cost function is the 0-1 loss, i.e., $c(x, y) = \mathbb{I}\_x \neq y$. Then the Kantorovich's optimal transport problem coincides with the total variation distance between measures.

Theorem 2. *Let $\mu, \nu \in \mathcal{P}(X)$ where X is a Polish space. Then the optimal transport cost of for Kantorovich's problem is $\sup_{A \subset X} |\mu(A) - \nu(A)|$.*

Wasserstein distance

Wasserstein distance, also known as the earth movers distance, is a widely used metric from optimal transport theory. Consider cost function $c(x, y) = |x - y|^p$ on $X \subset \mathbb{R}^d$ where $p \geq 1$. The p-Wasserstein distance is defined as

$$W_p(\mu, \nu) = \min_{\pi \in \Pi(\mu, \nu)} \left(\int_{X \times X} |x - y|^p \mathrm{d}\pi(x, y) \right)^{\frac{1}{p}} \quad (2)$$

where

$$\mu, \nu \in \mu | \int_X |x|^p \mathrm{d}\mu(x) < +\infty. \quad (3)$$

Sinkhorn iteration is known as an approximating method for calculating the Wasserstein distance. Here is an PyTorch implementation.

```
""" Sinkhorn iteration
Reference:
Cuturi M. Sinkhorn Distances: Lightspeed Computation of Optimal
    Transport[C].
Neural Information Processing Systems, 2013: 2292-2300.
"""
import torch
```

```
def main_iter(r, c, max_iter, threshold, lamda):
    """ Main iteration of Sinkhorn
    Args:
        r (Tensor): probability vectors with size (d, )
        c (Tensor): probability vectors with size (d, )
        max_iter (int): max iteration of Sinkhorn algorithm
        threshold (float): stop threshold
        lamda (float): `- 1 / lamda` is the Lagrange multiplier for
            the entropy
    """
    assert r.dim() == c.dim() == 1 and r.size(0) == c.size(0)
    d = r.size(0)
    m = (r.unsqueeze(0) - c.unsqueeze(1)).pow(2) # cost matrix
    k = (-lamda * m).exp()
    u = torch.ones(d, 1).float() / d
    k_tilde = torch.mm(torch.eye(d) / d, k)
    for i in range(max_iter):
        pre_u = u
        u = 1.0 / torch.mm(k_tilde, c / torch.mm(k_tilde, u))
        if (u - pre_u).abs().sum(-1).mean().item() < threshold:
            break
    v = c / torch.mm(k_tilde, u)
    return u * torch.mm(k * m, v)


if __name__ == '__main__':
    print(main_iter(torch.rand(3), torch.rand(3), 3, 0.1,
        0.2).size())
```