1 Notes on Optimal Transport

1.1 最优传输问题

最优运输问题的目标就是以**最小的成本**将**一个概率分布**转换为**另一个概率分布**。即把概率分布 $\mathbf{c}$ 以最小的成本转换到概率分布 $\mathbf{r}$，此时就要获得一个分配方案 $P \in \mathbb{R}_{>0}^{n\times m}$，满足 $P\mathbf{1}_m = \mathbf{c}$ 和 $P^T\mathbf{1}_n = \mathbf{r}$。同时又要考虑运输成本：一个已知的成本矩阵 M。于是最优传输问题可以表示为：

$$d_M(\mathbf{r},\mathbf{c}) = \min_{P\in U(\mathbf{r},\mathbf{c})} \sum_{i,j} P_{i,j} M_{i,j}$$

这里 $U(\mathbf{r},\mathbf{c}) \triangleq \{P \in \mathbb{R}_{>0}^{n\times m} | P\mathbf{1}_m = \mathbf{c}, P^T\mathbf{1}_n = \mathbf{r}\}$。此时 $d_M(\mathbf{r},\mathbf{c})$ 被称为 *Wasserstein metric* 或者为 *earth mover distance* (EMD) 代价函数。

1.2 Sinkhorn 距离和 Sinkhorn 算法

Sinkhorn 距离是对前者 *Wasserstein/EMD* 距离的改进，引入了熵 (Entropy) 正则项：

$$d_M^\lambda(\mathbf{r},\mathbf{c}) = \min_{P\in U(\mathbf{r},\mathbf{c})} \left(\sum_{i,j} P_{i,j} M_{i,j} - \frac{1}{\lambda} h(P) \right)$$

这里 $h(P) = -\sum_{i,j} P_{i,j} \log P_{i,j}$ 是 P 的信息熵，λ 是正则项对应的超参。此时，求解这个优化问题：

$$\begin{aligned}
&\mathcal{L} = \sum_{i,j} P_{i,j} M_{i,j} + \frac{1}{\lambda} \sum_{i,j} P_{i,j} \log P_{i,j} \\
\Rightarrow\quad & 0 = \frac{\partial \mathcal{L}}{\partial P_{i,j}} = M_{ij} + \frac{1}{\lambda}(P_{i,j} + 1) \\
\Rightarrow\quad & P_{i,j} = e^{-\lambda M_{ij} - 1}
\end{aligned}$$

这是在无约束条件下求得的 P 的解，考虑约束条件的话，引入缩放因子 α_i 和 β_j 用来让行和列满足约束：

$$P_{i,j} = \alpha_i \beta_j e^{-\lambda M_{ij}}$$

最终的满足约束的 P 的解可以通过下面这个伪代码和算法获得：

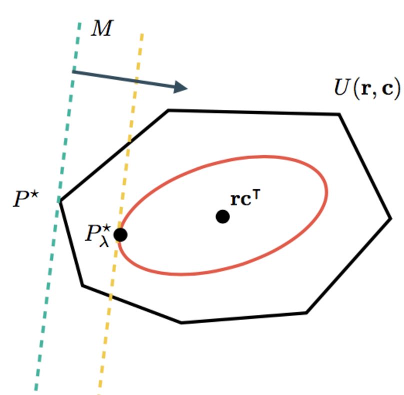
Algorithm 1 Sinkhorn algorithm

Input: M, $\mathbf{c}$, $\mathbf{r}$, λ.
Initialize: $P_\lambda = e^{-\lambda M}$
repeat
  scale the rows such that the row sums match $\mathbf{r}$
  scale the columns such that the column sums match $\mathbf{c}$
until convergence
Return Matrix P_λ

```python
def compute_optimal_transport(M, r, c, lam, epsilon=1e-8):
    """
    Computes the optimal transport matrix and Sinkhorn distance using
    the Sinkhorn-Knopp algorithm
    Inputs:
        - M : cost matrix (n, m)
        - r : vector of marginals (n, )
        - c : vector of marginals (m, )
        - lam : strength of the entropic regularization
        - epsilon : convergence parameter
    Outputs:
        - P : optimal transport matrix (n, m)
        - dist : Sinkhorn distance
    """
    n, m = M.shape
    # initialize: P = e^{-\lambda * M} (P_{ij} = e^{-\lambda * M_{ij}})
    P = np.exp(- lam * M)
    P /= P.sum()
    u = np.zeros(n)
    # normalize this matrix
    while np.max(np.abs(u - P.sum(1))) > epsilon:
        # scale the rows such that the row sums match r
        u = P.sum(1)
        P *= (r / u).reshape((-1, 1))
        # scale the columns such that the column sums match c
        P *= (c / P.sum(0)).reshape((1, -1))
    return P, np.sum(P * M)
```

1.3 几何解释

最优传输问题，无论有没有熵正则化，都有一个漂亮的几何解释。



成本矩阵决定了分布的优劣方向。集合 $U(\mathbf{r},\mathbf{c})$ 包含所有可行的分布。在未正则化

的情况下，最优 $P^\star$ 通常在这样一个集合的某个角上。在添加熵正则因子时，我们将自己限制在熵最小的分布上，即位于平滑的红色曲线内。由于我们不必再处理 $U(\mathbf{r},\mathbf{c})$ 的尖角，因此更容易找到最优值。特殊情况下，当 $\lambda\to\infty$ 时，$P_\lambda^\star$ 将变得越来越接近 $P^\star$（直到算法遇到数值困难）。另一方面，对于 $\lambda\to 0$ 的情况，只考虑熵项，$P_\lambda^\star=\mathbf{r}\mathbf{c}^T$（这是一种均匀分布，均匀分布信息熵最小）。