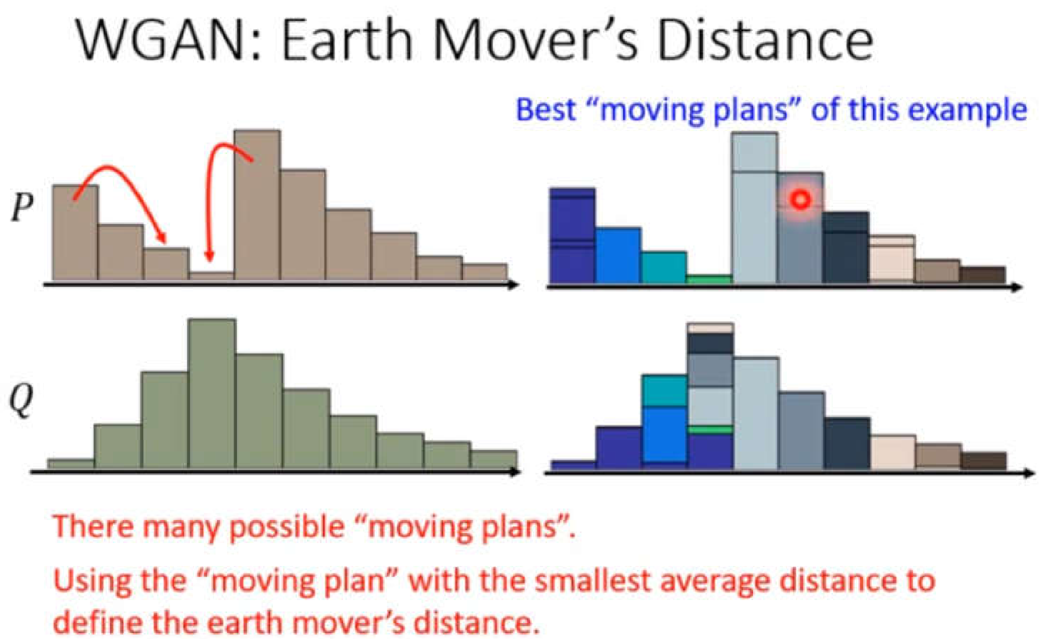
Earth Mover's Distance

Abstract

第一件要知道的事情就是 EMD 就是 Wasserstein 距离，Wasserstein 就是 EMD。

EMD 是一种用来评估在某些特征空间中两个多维分布之间差异的方法，EMD 最初是在图像检索背景下用于图像相似度度量的方法，由于其各种优点，逐渐用到其他方面的相似度度量。

目前我知道 EMD 的直接用法就是衡量两个直方图之间的相似度。衡量原理就是从直方图 P 变为直方图 Q 所需要消耗的最小能量，或者说把概率分布 q 转换为概率分布 q 的最小传输质量（这个我们待会也好好解释一下），也叫最优传输距离或者推土机距离。所谓推土机距离，就和 “推土机” 稍微有点联系。如果将分布看做空间中泥土的分布，那么这两个分布间的距离就是将这些泥土从一个分布改变到另一个分布所需要消耗的最小能量。这里的能量是泥土重量 (权值) 和移动距离的乘积。下面给出一个超级直观的图来理解 Earth Mover's Distance.

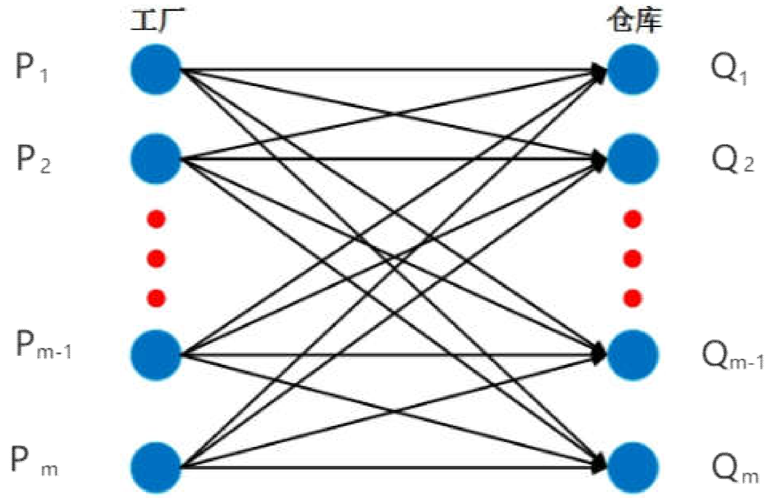


Earth Mover's Distance

Am example of 货物运输优化问题

这里我们举这么个例子来帮助你更好的理解这个所谓的推土机距离和所谓的由直方图（概率分布）p 变为直方图（概率分布）Q 的最小传输质量到底是个什么意思？

推土机距离实际上是线性规划中运输问题的最优解。推土机距离实际上是线性规划中运输问题的最优解。首先，简要描述一下问题。我们假设这个例子是从多个工厂运输货物到多个仓库。如下图所示：



$\mathrm{P}=[P_1,P_2,P_3...,P_m]$代表 m 个工厂，其中工厂$P_i$由重量为$\mathrm{wP}_i$的货物。

$\mathrm{Q}=[Q_1,Q_2,Q_3...,Q_n]$代表 n 个仓库，仓库$Q_j$最大容量为$\mathrm{w}Q_{\mathrm{j}}$。货物之间没有什么区别，都是同一类东西。每个仓库希望装尽可能多的货物。如何尽可能高效的把所有（当仓库总容量大于货物总重量）或部分货物（当仓库总容量小于货物总重量）从 P 运送到到 Q，就是运输问题的优化目标。在本例中，由于 P 和 Q 都是离散的，那么推土机距离可以用运输问题的匈牙利算法来求解分布之间的距离。

我们定义货物从工厂P_i运到仓库Q_j之间的距离为d_{ij}，运送货物的重量为f_{ij}这样一次运输需要的工作量为。显然，距离越远或货物越重，工作量就越大。运输可能是多对多的，即一个工厂运输货物到多个仓库，或者一个仓库可以接受多个工厂的货物。货物从工厂运到仓库需要很多次这样的运输，经过一些计算和优化，这时我们得到工作量总和的最小值 W：

$$\min W=\sum_{i=1}^{m}\sum_{j=1}^{n}d_{ij}\times f_{ij}$$

其中，距离d_{ij}是事先定义的，所以运输量f_{ij}是式中唯一的变量，对f_{ij}作如下约束：

约束一：

运输过程中从工厂P_i到仓库Q_j不能反向，即

$$f_{ij}\geq 0,\ \ 1\leq i\leq m,\ \ 1\leq j\leq n$$

约束二：

从工厂P_i运出去的所有货物重量的和不可能超过该工厂中所有货物的总重量wP_i，即

$$\sum_{j=1}^{n}f_{ij}\leq wP_i\ ,\ \ 1\leq i\leq m$$

约束三：

仓库 Q_j 接受的所有货物总重量不可能超过其总容量 $\mathrm{wQ_j}$，即

$$\sum_{i=1}^{m} f_{ij} \le wQ_j,\ 1 \le j \le n$$

约束四：

总运输量的上限是工厂中货物总重量、仓库总容量中的最小值。

$$\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min(\sum_{i=1}^{m} wP_i,\ \sum_{j=1}^{n} wQ_j)$$

假设仓库总容量和工厂货物总重量相等。且我们已经找到了阿最优解 f_{ij}^*，为了使推土机距离不会随着总运输量的变化而变化，每一次运输量还要除以总重量，以达到归一化的目的：

$$EMD(P,Q) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} \times f_{ij}^*}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}^*}$$

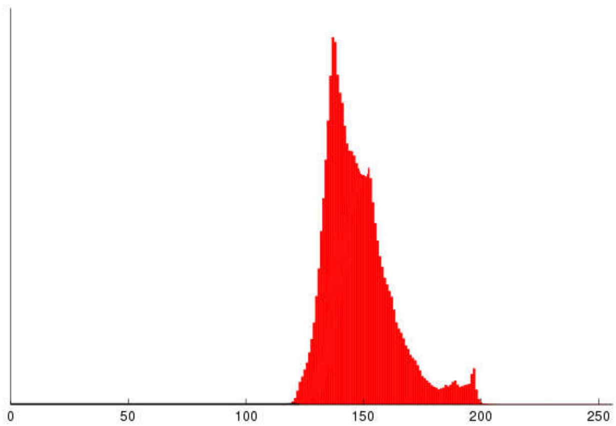
看到这里你应该知道所谓的**最小传输质量**是什么意思了吧。

承上启下的过渡

刚才讲解了 EMD 在货物运输的优化问题，现在来讲解一下 EMD 是怎么计算图像之间的相似度的，从上面的例子，我们可以感受到，EMD 肯定不是直接作用于一幅图像的，即图像这种矩阵的输入不太可能符合我们 EMD 的接口，所以这个图像肯定需要转换成可以用 EMD 计算的形式，所以下面我们来介绍一下 Signature 的概念，Signature 就是 EMD 可以接受的输入形式

Signature

1. 我们现在有一个图片（**假设只有一个通道!!**），我们把他转换成直方图来统计这个图片每个像素值出现的频率。



传统的直方图一半辉聚集在某些 bin（横坐标）上，过于浪费空间（很多 bin 都没有像素出

现）。所以我们可以用 signature 来表达直方图内容。

signature 定义为一系列重要特征，可以写作 $s=(m,w)$，其中 m 是某个特征，在这里 m 是指 bin。w 是特征的权重，在这里指的是 bin 的高度，也即某个 bin 中像素出现的次数。这样我们就可以得到一张图片的一个 signature 集合，如下所示是图片 P 的 signature 集合

$$P=\{(P_1,w_{P_1}),(P_2,w_{P_2}),(P_3,w_{P_3})\dots,(P_M,w_{P_M})\}$$

同样的我们还有一个图片 Q，也给出他的 signature 集合。

$$Q=\{(Q_1,w_{Q_1}),(Q_2,w_{Q_2}),(Q_3,w_{Q_3})\dots,(Q_N,w_{Q_N})\}$$

有了这两个集合，是不是就很像上面的货物运输优化问题了，我们想得到两个图片的相似度，就可以靠衡量从直方图 P 转化到直方图 Q 的最小做功，具体怎么做看小道儿我操作:

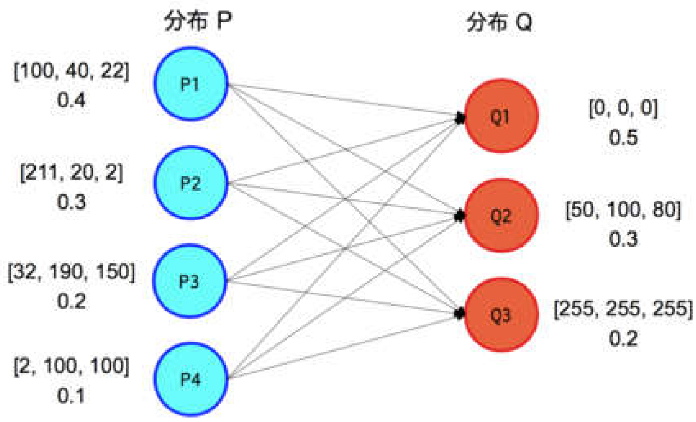
我们现在只需要把 P_i 当作 i 工厂的货物，Q_j 代表 j 仓库的容量，求

$$EMD(P,Q)=\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}d_{ij}\times f_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n}f_{ij}}$$

就可以了。

什么？ 看到这里还有小伙伴疑惑这个 d_{ij} 怎么求？最优 f_{ij} 怎么求？下面讲解一下

首先这个 d_{ij} 代表从 P_i 到 Q_j 的距离，假如，图像是 RGB 的，那么特征就是三维的即[红,绿,蓝]。下面分布 P 的 0.4 0.3 0.2 0.1 代表权重，反正他们的加和必须要是 1 。



这个 P_i 到 Q_j 就是这个三维特征之间的 L1 或者 L2 范数之类的向量距离（其他距离也行），这样我们就可以求整个距离矩阵

然后我们既然知道权重 w，利用前面四个约束条件，我们也就可以求得最佳得 f 矩阵。

EMD 的重点还是在于求这个 f 矩阵。有了 f 矩阵就可以算 EMD 了，算 EMD 都不是啥难事儿，关键就是这个线性规划求 f 矩阵！

代码演示

```
import numpy as np
import cv
#p、q是两个矩阵，第一列表示权值，后面三列表示直方图或数量
p=np.asarray([[0.4,100,40,22],
            [0.3,211,20,2],
            [0.2,32,190,150],
            [0.1,2,100,100]],np.float32)
q=np.array([[0.5,0,0,0],
            [0.3,50,100,80],
            [0.2,255,255,255]],np.float32)
pp=cv.fromarray(p)
qq=cv.fromarray(q)
emd=cv.CalcEMD2(pp,qq,cv.CV_DIST_L2)
print(emd)
```

p 矩阵第一列是权重，后面三个表示直方图的 bin，这个直方图是三维的，因为图片也是三维的。

reference:
https://www.cnblogs.com/aijianiula/p/9382641.html
http://blog.leanote.com/post/braveapple/%E6%8E%A8%E5%9C%9F%E6%9C%BA%E8%B7%9D%E7%A6%BB-Earth-Mover-Distance
这俩是好文章，通俗又不失细节的讲了货物运输优化问题来帮助理解 EMD 在图像中的部分工作原理。
https://ghost-hunter.net9.org/2019/02/01/evaluation.html
http://www.jeepxie.net/article/84505.html