Standford CS229 2022Fall，第15讲：ICA

独立成分分析

我们的下一个主题是独立成分分析（ICA)。与 PCA 类似，这将在一个新的基中表示我们的数据。然而，目标截然不同。

作为一个动机示例，考虑“鸡尾酒会问题”。在这里，d 位说话者同时在一个聚会上讲话，房间里的任何麦克风只记录下 d 位说话者声音的重叠组合。
但假设我们在房间里放置了 d 个不同的麦克风，由于每个麦克风与每位说话者的距离不同，它记录的是说话者声音的不同组合。利用这些麦克风录音，我们能否分离出原始的 d 位说话者的声音信号？

为了形式化这个问题，我们想象有一些数据 $s \in \mathbb{R}^d$ 是通过 d 个独立源生成的。我们观察到的是
$x = As,$
其中 A 是一个未知的方阵，称为混合矩阵。重复观测得到一个数据集 $\{x^{(i)}; i = 1, \ldots, n\}$，我们的目标是恢复生成我们数据的源 $s^{(i)}$（即 $x^{(i)} = As^{(i)}$)。

在我们的鸡尾酒会问题中，$s^{(i)}$ 是一个 d 维向量，$s_j^{(i)}$ 是说话者 j 在时间 i 发出的声音。同样，$x^{(i)}$ 是一个 d 维向量，$x_j^{(i)}$ 是麦克风 j 在时间 i 记录的声学读数。

令 $W = A^{-1}$ 为解混矩阵。我们的目标是找到 W，以便在给定麦克风录音 $x^{(i)}$ 的情况下，通过计算 $s^{(i)} = Wx^{(i)}$ 来恢复源。为了记号方便，我们还让 w_i^T 表示 W 的第 i 行，因此

$$W = \begin{bmatrix} —w_1^T— \\ \ldots \\ —w_d^T— \end{bmatrix}.$$

因此，$w_i \in \mathbb{R}^d$，第 j 个源可以通过 $s_j^{(i)} = w_j^T x^{(i)}$ 恢复。

1 ICA 的模糊性

在多大程度上可以恢复 $W = A^{-1}$？如果我们对源和混合矩阵没有先验知识，仅凭 $x^{(i)}$ 很容易看出 A 存在一些固有的、无法恢复的模糊性。

具体来说，令 P 为任意 $d \times d$ 置换矩阵。这意味着 P 的每一行和每一列都恰好有一个“1”。以下是置换矩阵的一些示例：

$$P = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad P = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

如果 z 是一个向量，则 Pz 是另一个包含 z 坐标排列版本的向量。仅凭 $x^{(i)}$，我们将无法区分 W 和 PW。具体来说，原始源的排列是模糊的，这不足为奇。幸运的是，这对大多数应用来说并不重要。

此外，我们无法恢复 w_i 的正确缩放比例。例如，如果 A 被替换为 $2A$，且每个 $s^{(i)}$ 被替换为 $(0.5)s^{(i)}$，那么我们观察到的 $x^{(i)} = 2A \cdot (0.5)s^{(i)}$ 仍然是相同的。更广泛地说，如果 A 的某一列被缩放因子 α 缩放，而相应的源被缩放因子 $1/\alpha$ 缩放，那么仅凭 $x^{(i)}$ 也无法确定发生了这种情况。因此，我们无法恢复源的“正确”缩放比例。然而，对于我们关心的应用——包括鸡尾酒会问题——这种模糊性也不重要。具体来说，将说话者语音信号 $s_j^{(i)}$ 缩放某个正因子 α 只会影响该说话者语音的音量。此外，符号变化无关紧要，因为 $s_j^{(i)}$ 和 $-s_j^{(i)}$ 在扬声器上播放时听起来相同。因此，如果算法找到的 w_i 被任何非零实数缩放，相应的恢复源 $s_i = w_i^T x$ 将被相同的因子缩放；但这通常无关紧要。（这些评论也适用于我们在课堂上讨论的大脑/MEG 数据的 ICA。）

这些是 ICA 中唯一的模糊性来源吗？事实证明，只要源 s_i 是非高斯分布的，它们就是唯一的模糊性来源。为了理解高斯数据的困难，考虑一个例子，其中 $n=2$，且 $s \sim N(0,I)$。这里，I 是 2×2 单位矩阵。请注意，标准正态分布 $N(0,I)$ 的密度轮廓是以原点为中心的圆，且密度具有旋转对称性。

现在，假设我们观察到一些 $x=As$，其中 A 是我们的混合矩阵。
那么，x 的分布将是高斯分布，$x \sim N(0,AA^T)$，因为
$E_{s\sim N(0,I)}[x]=E[As]=AE[s]=0$
$Cov[x]=E_{s\sim N(0,I)}[xx^T]=E[Ass^TA^T]=AE[ss^T]A^T=A\cdot Cov[s]\cdot A^T=AA^T$

现在，令 R 为任意正交矩阵（更通俗地说，是一个旋转/反射矩阵），使得 $RR^T=R^TR=I$，并令 $A'=AR$。那么，如果数据是根据 A' 而不是 A 混合的，我们将观察到 $x'=A's$。x' 的分布也是高斯分布，$x' \sim N(0,AA^T)$，因为
$E_{s\sim N(0,I)}[x'(x')^T]=E[A'ss^T(A')^T]=E[ARss^T(AR)^T]=ARR^TA^T=AA^T.$

因此，无论是使用 A 还是 A' 作为混合矩阵，我们都会观察到来自 $N(0,AA^T)$ 分布的数据。因此，无法判断源是用 A 还是 A' 混合的。混合矩阵中存在一个任意的旋转分量，无法从数据中确定，我们也无法恢复原始源。

我们上面的论证基于多元标准正态分布具有旋转对称性的事实。尽管这为高斯数据上的 ICA 描绘了一幅黯淡的画面，但事实证明，只要数据不是高斯分布的，就有可能在拥有足够数据的情况下恢复 d 个独立源。

2 密度与线性变换

在继续推导 ICA 算法之前，我们先简要讨论一下线性变换对密度的影响。

假设随机变量 s 根据某个密度 $p_s(s)$ 抽取。为简单起见，暂时假设 $s\in\mathbb{R}$ 是一个实数。现在，令随机变量 x 定义为 $x=As$（这里，$x\in\mathbb{R}$，$A\in\mathbb{R}$）。令 p_x 为 x 的密度。p_x 是什么？

令 $W=A^{-1}$。要计算 x 的特定值的“概率”，人们可能会倾向于计算 $s=Wx$，然后在该点评估 p_s，并得出结论“$p_x(x)=p_s(Wx)$”。然而，这是不正确的。例如，令 $s\sim Uniform[0,1]$，所以 $p_s(s)=1_{\{0\le s\le 1\}}$。现在，令 $A=2$，所以 $x=2s$。
显然，x 在区间 $[0,2]$ 上均匀分布。因此，它的密度由 $p_x(x)=(0.5)1_{\{0\le x\le 2\}}$ 给出。这不等于 $p_s(Wx)$，其中 $W=0.5=A^{-1}$。相反，正确的公式是 $p_x(x)=p_s(Wx)\cdot|W|$。

更一般地，如果 s 是一个具有密度 p_s 的向量值分布，且 $x=As$ 对于一个可逆方阵 A，则 x 的密度由下式给出：
$p_x(x)=p_s(Wx)\cdot|W|,$
其中 $W=A^{-1}$。

备注。如果你见过 A 将 $[0,1]^d$ 映射到体积为 $|A|$ 的集合的结果，那么这里还有另一种方式来记住上面给出的 p_x 公式，这也推广了我们之前的 1 维例子。具体来说，令 $A\in\mathbb{R}^{d\times d}$ 给定，且如常令 $W=A^{-1}$。同时令 $C_1=[0,1]^d$ 为 d 维超立方体，并定义 $C_2=\{As:s\in C_1\}\subseteq\mathbb{R}^d$ 为 $C1$ 在 A 映射下的像。那么，线性代数中的一个标准结果（实际上，也是定义行列式的一种方式）是 $C2$ 的体积由 $|A|$ 给出。现在，假设 s 在 $[0,1]^d$ 上均匀分布，所以其密度为 $p_s(s)=1_{\{s\in C_1\}}$。那么显然 x 将在 $C2$ 上均匀分布。其密度因此被发现为 $p_x(x)=1_{\{x\in C_2\}}/vol(C_2)$（因为它必须在 $C2$ 上积分到 1）。但利用矩阵逆的行列式只是行列式的逆这一事实，我们有
$1/vol(C_2)=1/|A|=|A^{-1}|=|W|$。因此，$p_x(x)=1_{\{x\in C_2\}}|W|=1_{\{Wx\in C_1\}}|W|=p_s(Wx)|W|$。

3 ICA 算法

我们现在准备推导一个 ICA 算法。我们描述 Bell 和 Sejnowski 的算法，并将其解释为一种最大似然估计方法。（这与他们最初的解释不同，后者涉及一个复杂的概念，称为信息最大化原理，但在现代对 ICA 的理解下已不再必要。）

我们假设每个源 s_j 的分布由密度 p_s 给出，且源 s 的联合分布由下式给出：
$p(s) = \prod_{j=1}^{d} p_s(s_j)$.
请注意，通过将联合分布建模为边缘分布的乘积，我们捕捉到了源是独立的假设。利用我们前一节的公式，这意味着对于 $x = As = W^{-1}s$，其密度为：
$p(x) = \prod_{j=1}^{d} p_s(w_j^T x) \cdot |W|$.
剩下的就是指定单个源 p_s 的密度。

回想一下，对于一个实值随机变量 z，其累积分布函数（cdf）F 定义为 $F(z_0) = P(z \leq z_0) = \int_{-\infty}^{z_0} p_z(z)dz$，而密度是 cdf 的导数：$p_z(z) = F'(z)$。

因此，要指定 s_i 的密度，我们只需要为其指定某个 cdf。cdf 必须是一个从零单调增加到一的函数。根据我们之前的讨论，我们不能选择高斯 cdf，因为 ICA 在高斯数据上不起作用。相反，我们将选择一个合理的“默认”cdf，它缓慢地从 0 增加到 1，即 sigmoid 函数 $g(s) = 1/(1+e^{-s})$。因此，$p_s(s) = g'(s)$。

方阵 W 是我们模型中的参数。给定训练集 $\{x^{(i)}; i = 1, \ldots, n\}$，对数似然函数为：
$\ell(W) = \sum_{i=1}^{n} \left(\sum_{j=1}^{d} \log g'(w_j^T x^{(i)}) + \log |W| \right)$.
我们希望最大化这个关于 W 的函数。通过对 W 求导并利用（第一组讲义中的）事实 $\nabla_W |W| = |W|(W^{-1})^T$，我们很容易推导出一个随机梯度上升学习规则。对于一个训练样本 $x^{(i)}$，更新规则是：

$$W := W + \alpha \left(\begin{bmatrix} 1 - 2g(w_1^T x^{(i)}) \\ 1 - 2g(w_2^T x^{(i)}) \\ \cdots \\ 1 - 2g(w_d^T x^{(i)}) \end{bmatrix} x^{(i)T} + (W^T)^{-1} \right),$$

其中 α 是学习率。

算法收敛后，我们再计算 $s^{(i)} = Wx^{(i)}$ 来恢复原始源。

备注。在写下数据的似然函数时，我们隐含地假设了 $x^{(i)}$ 之间是相互独立的（对于不同的 i 值；请注意，这个问题与 $x^{(i)}$ 的不同坐标是否独立是不同的），因此训练集的似然函数由 $\prod_i p(x^{(i)}; W)$ 给出。这个假设对于语音数据和其他时间序列显然是不正确的，因为 $x^{(i)}$ 是相关的，但可以证明，如果有足够的数据，相关训练样本不会影响算法的性能。然而，对于连续训练样本相关的场景，在实现随机梯度上升时，有时以随机打乱的顺序访问训练样本有助于加速收敛。（即，在训练集的随机洗牌副本上运行随机梯度上升。）