独立成分分析（Independent Component Analysis）

JerryLead
csxulijie@gmail.com

1. 问题：

1、上节提到的 PCA 是一种数据降维的方法，但是只对符合高斯分布的样本点比较有效，那么对于其他分布的样本，有没有主元分解的方法呢？

2、经典的鸡尾酒宴会问题（cocktail party problem）。假设在 party 中有 n 个人，他们可以同时说话，我们也在房间中一些角落里共放置了 n 个声音接收器（Microphone）用来记录声音。宴会过后，我们从 n 个麦克风中得到了一组数据$\{x^{(i)}(x_1^{(i)},x_2^{(i)},\ldots,x_n^{(i)}); i=1,\ldots,m\}$，i 表示采样的时间顺序，也就是说共得到了 m 组采样，每一组采样都是 n 维的。我们的目标是单单从这 m 组采样数据中分辨出每个人说话的信号。

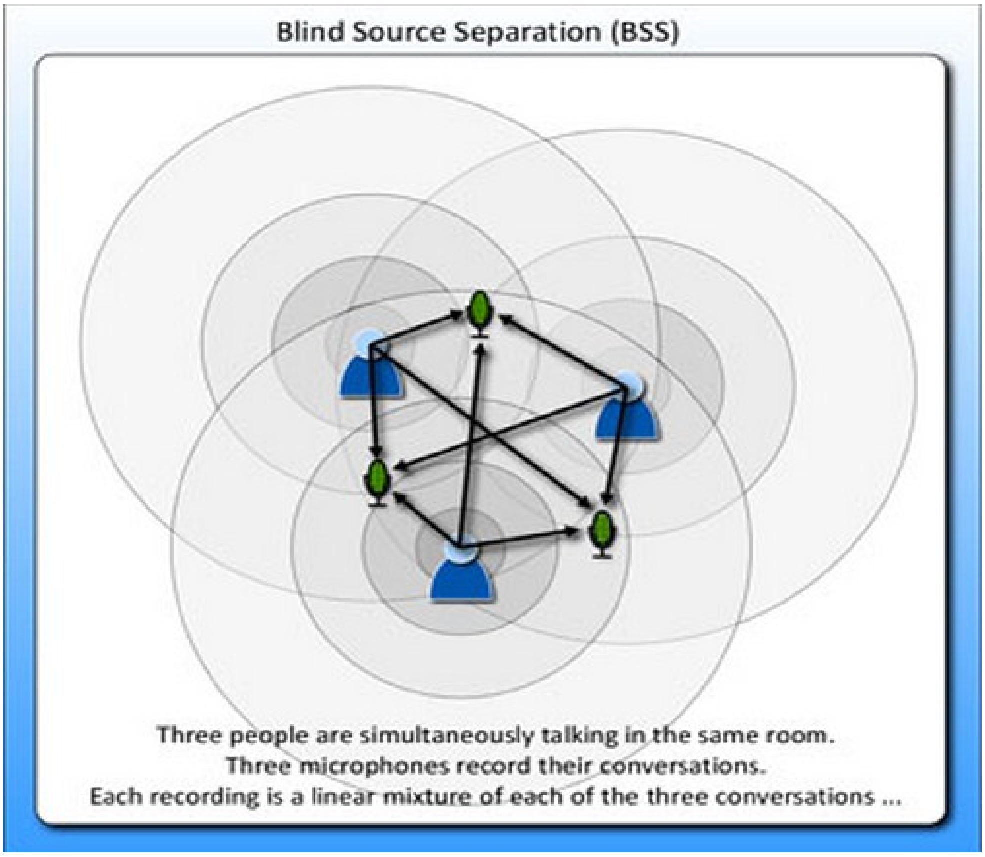
将第二个问题细化一下，有 n 个信号源$s\,(s_1,s_2,\ldots,s_n)^T$，$s\in\mathbb{R}^n$，每一维都是一个人的声音信号，每个人发出的声音信号独立。A 是一个未知的混合矩阵（mixing matrix），用来组合叠加信号 s，那么

$$x = As$$

x 的意义在上文解释过，这里的 x 不是一个向量，是一个矩阵。其中每个列向量是$x^{(i)}$，

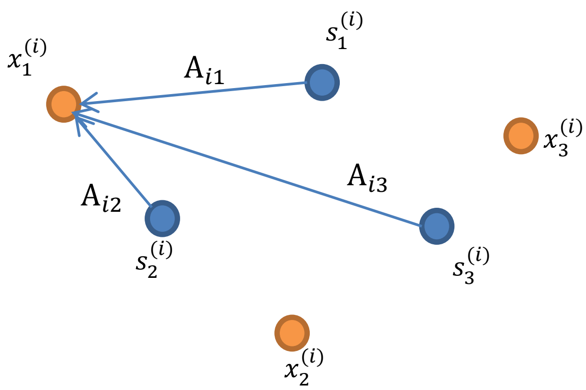
$$x^{(i)} = As^{(i)}$$

表示成图就是



这张图来自

http://amouraux.webnode.com/research-interests/research-interests-erp-analysis/blind-source-separation-bss-of-erps-using-independent-component-analysis-ica/



$x^{(i)}$的每个分量都由$s^{(i)}$的分量线性表示。A 和 s 都是未知的，x 是已知的，我们要想办法根据 x 来推出 s。这个过程也称作为盲信号分离。

令$W = A^{-1}$，那么

$$s^{(i)} = \mathrm{A}^{-1}x^{(i)} = Wx^{(i)}$$

将 W 表示成

$$W = \begin{bmatrix} — w_1^T — \\ \vdots \\ — w_n^T — \end{bmatrix}.$$

其中$w_i \in \mathbb{R}^n$，其实就是将w_i写成行向量形式。那么得到：

$$s_j^{(i)} = w_j^T x^{(i)}$$

2. ICA 的不确定性（ICA ambiguities）

由于 w 和 s 都不确定，那么在没有先验知识的情况下，无法同时确定这两个相关参数。比如上面的公式 s=wx。当 w 扩大两倍时，s 只需要同时扩大两倍即可，等式仍然满足，因此无法得到唯一的 s。同时如果将人的编号打乱，变成另外一个顺序，如上图的蓝色节点的编号变为 3,2,1，那么只需要调换 A 的列向量顺序即可，因此也无法单独确定 s。这两种情况称为原信号不确定。

还有一种 ICA 不适用的情况，那就是信号不能是高斯分布的。假设只有两个人发出的声音信号符合多值正态分布，$s \sim N(0, I)$，I 是 2*2 的单位矩阵，s 的概率密度函数就不用说了吧，以均值 0 为中心，投影面是椭圆的山峰状（参见多值高斯分布）。因为$x = As$，因此，x 也是高斯分布的，均值为 0，协方差为$E[xx^T] = E[Ass^TA^T] = AA^T$。

令 R 是正交阵($RR^T = R^TR = I$)，$A' = AR$。如果将 A 替换成 A'。那么$x' = A's$。s 分布没变，因此 x'仍然是均值为 0，协方差$E[x'(x')^T] = E[A'ss^T(A')^T] = E[ARss^T(AR)^T] = ARR^TA^T = AA^T$。

因此，不管混合矩阵是 A 还是 A'，x 的分布情况是一样的，那么就无法确定混合矩阵，也就无法确定原信号。

3. 密度函数和线性变换

在讨论 ICA 具体算法之前，我们先来回顾一下概率和线性代数里的知识。

假设我们的随机变量 s 有概率密度函数$p_s(s)$（连续值是概率密度函数，离散值是概率）。为了简单，我们再假设 s 是实数，还有一个随机变量 x=As，A 和 x 都是实数。令p_x是 x 的概率密度，那么怎么求p_x？

令$W = A^{-1}$，首先将式子变换成$s = Wx$，然后得到$p_x(x) = p_s(Ws)$，求解完毕。可惜这种方法是错误的。比如 s 符合均匀分布的话（s~Uniform[0,1]），那么 s 的概率密度是$p_s(s) = 1\{0 \le s \le 1\}$，现在令 A=2，即 x=2s，也就是说 x 在[0,2]上均匀分布，可知$p_x(x) = 0.5$。然而，前面的推导会得到$p_x(x) = p_s(0.5s) = 1$。正确的公式应该是

$$p_x(x) = p_s(Wx)|W|$$

推导方法

$$F_X(x) = P(X \le x) = P(AS \le x) = P(S \le Wx) = F_S(Wx)$$
$$p_x(x) = F_X'(x) = F_S'(Wx) = p_s(Wx)|W|$$

更一般地，如果 s 是向量，A 可逆的方阵，那么上式子仍然成立。

4. ICA 算法

ICA 算法归功于 Bell 和 Sejnowski，这里使用最大似然估计来解释算法，原始的论文中使用的是一个复杂的方法 Infomax principal。

我们假定每个s_i有概率密度p_s，那么给定时刻原信号的联合分布就是

$$\mathrm{p(s)} = \prod_{i=1}^{n} p_s(s_i)$$

这个公式代表一个假设前提：每个人发出的声音信号各自独立。有了 p(s)，我们可以求得 p(x)

$$\mathrm{p(x)} = \mathrm{p}_s(Wx)|\mathrm{W}| = |\mathrm{W}| \prod_{i=1}^{n} p_s(w_i{}^T x)$$

左边是每个采样信号 x（n 维向量）的概率，右边是每个原信号概率的乘积的|W|倍。

前面提到过，如果没有先验知识，我们无法求得 W 和 s。因此我们需要知道$p_s(s_i)$，我们打算选取一个概率密度函数赋给 s，但是我们不能选取高斯分布的密度函数。在概率论里我们知道密度函数 p(x)由累计分布函数（cdf）F(x)求导得到。F(x)要满足两个性质是：单调递增和在[0,1]。我们发现 sigmoid 函数很适合，定义域负无穷到正无穷，值域 0 到 1，缓慢递增。我们假定 s 的累积分布函数符合 sigmoid 函数

$$\mathrm{g(s)} = \frac{1}{1+e^{-s}}$$

求导后

$$p_s(s) = g'(s) = \frac{e^s}{(1+e^s)^2}$$

这就是 s 的密度函数。这里 s 是实数。

如果我们预先知道 s 的分布函数，那就不用假设了，但是在缺失的情况下，sigmoid 函数能够在大多数问题上取得不错的效果。由于上式中$p_s(s)$是个对称函数，因此 E[s]=0（s 的均值为 0），那么 E[x]=E[As]=0，x 的均值也是 0。

知道了$p_s(s)$，就剩下 W 了。给定采样后的训练样本$\{\mathrm{x}^{(i)}(x_1^{(i)}, x_2^{(i)}, \dots, x_n^{(i)}); i = 1, \dots, m\}$，样本对数似然估计如下：

使用前面得到的 x 的概率密度函数，得

$$\ell(W) = \sum_{i=1}^{m} \left(\sum_{j=1}^{n} \log g'(w_j^T x^{(i)}) + \log |W| \right).$$

大括号里面是$\mathrm{p}(\mathrm{x}^{(i)})$。

接下来就是对 W 求导了，这里牵涉一个问题是对行列式|W|进行求导的方法，属于矩阵微积分。这里先给出结果，在文章最后再给出推导公式。

$$\nabla_W |W| = |W|(W^{-1})^T$$

最终得到的求导后公式如下，$\log g'(s)$的导数为$1-2g(s)$（可以自己验证）：

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

其中α是梯度上升速率，人为指定。

当迭代求出 W 后，便可得到$s^{(i)} = Wx^{(i)}$来还原出原始信号。

注意：我们计算最大似然估计时，假设了$x^{(i)}$与$x^{(j)}$之间是独立的，然而对于语音信号或者其他具有时间连续依赖特性（比如温度）上，这个假设不能成立。但是在数据足够多时，假设独立对效果影响不大，同时如果事先打乱样例，并运行随机梯度上升算法，那么能够加快收敛速度。

回顾一下鸡尾酒宴会问题，s 是人发出的信号，是连续值，不同时间点的 s 不同，每个人发出的信号之间独立（s_i和s_j之间独立）。s 的累计概率分布函数是 sigmoid 函数，但是所有人发出声音信号都符合这个分布。A（W 的逆阵）代表了 s 相对于 x 的位置变化，x 是 s 和 A 变化后的结果。

5. 实例

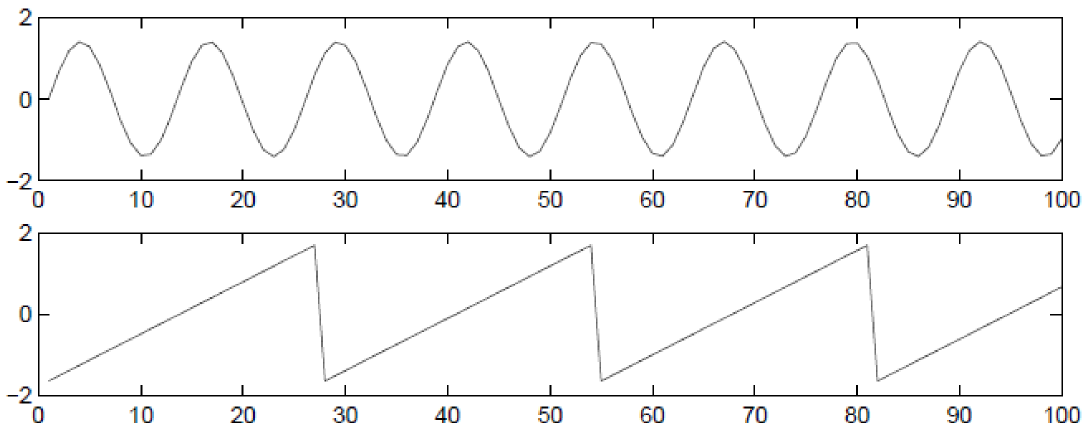


Figure 1: The original signals.

s=2 时的原始信号

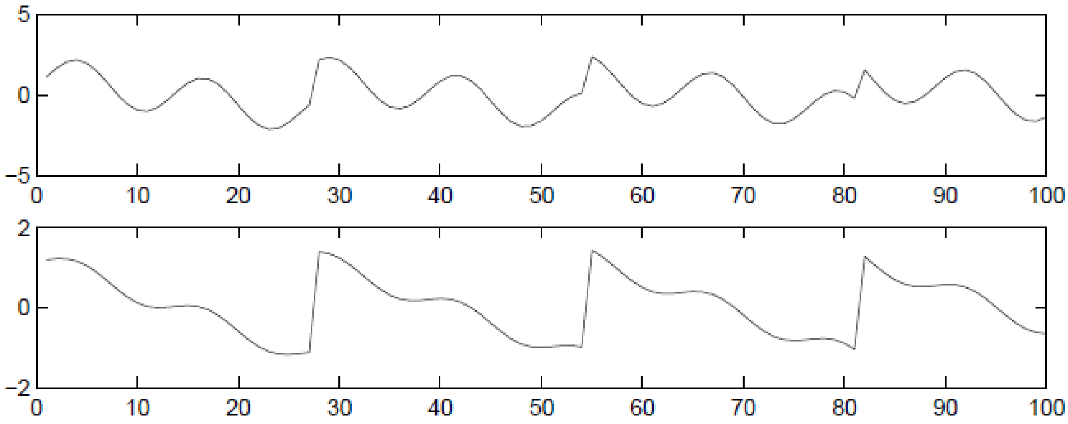
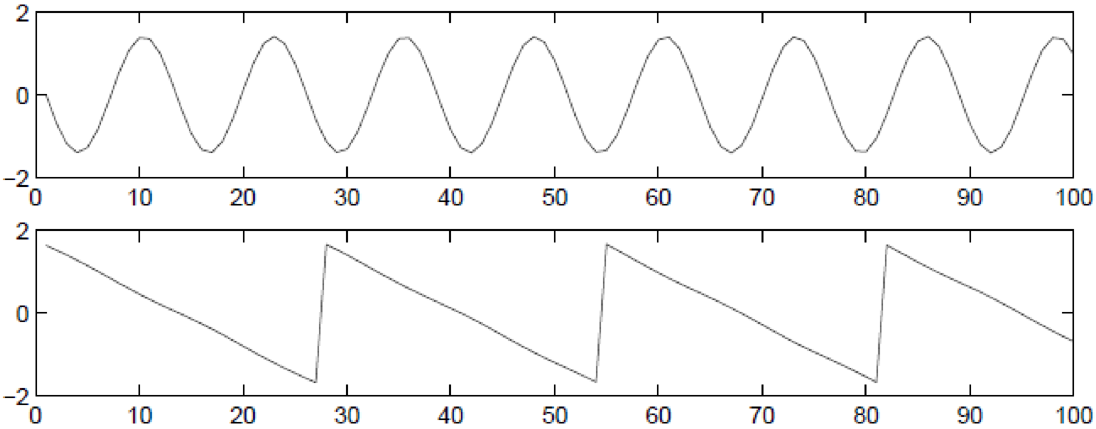


Figure 2: The observed mixtures of the source signals in Fig. 1.

观察到的 x 信号



使用 ICA 还原后的 s 信号

6. 行列式的梯度

对行列式求导，设矩阵 A 是 n×n 的，我们知道行列式与代数余子式有关，

$$|A| = \sum_{i=1}^{n} (-1)^{i+j} A_{ij} |A_{\backslash i, \backslash j}| \qquad (\text{for any } j \in 1, \ldots, n)$$

$A_{\backslash i, \backslash j}$是去掉第 i 行第 j 列后的余子式，那么对$A_{k,l}$求导得

$$\frac{\partial}{\partial A_{k\ell}} |A| = \frac{\partial}{\partial A_{k\ell}} \sum_{i=1}^{n} (-1)^{i+j} A_{ij} |A_{\backslash i, \backslash j}| = (-1)^{k+\ell} |A_{\backslash k, \backslash \ell}| = (\text{adj}(A))_{\ell k}.$$

adj(A)跟我们线性代数中学的A*是一个意思，因此

$$\nabla_A |A| = (\text{adj}(A))^T = |A| A^{-T}.$$

7. ICA 算法扩展描述

上面介绍的内容基本上是讲义上的，与我看的另一篇《Independent Component Analysis: Algorithms and Applications》（Aapo Hyvärinen and Erkki Oja）有点出入。下面总结一下这篇文章里提到的一些内容（有些我也没看明白）。

首先里面提到了一个与“独立”相似的概念“不相关（uncorrelated）”。Uncorrelated 属于部分独立，而不是完全独立，怎么刻画呢？

如果随机变量y_1和y_2是独立的，当且仅当$p(y_1,y_2)=p(y_1)p(y_2)$。

如果随机变量y_1和y_2是不相关的，当且仅当$E(y_1,y_2)=E(y_1)E(y_2)$

第二个不相关的条件要比第一个独立的条件“松”一些。因为独立能推出不相关，不相关推不出独立。

证明如下：

$$p_1(y_1)=\int p(y_1,y_2)dy_2,$$

$$p(y_1,y_2)=p_1(y_1)p_2(y_2).$$

$$\begin{aligned}E\{h_1(y_1)h_2(y_2)\} &= \int\int h_1(y_1)h_2(y_2)p(y_1,y_2)dy_1dy_2\\ &= \int\int h_1(y_1)p_1(y_1)h_2(y_2)p_2(y_2)dy_1dy_2 = \int h_1(y_1)p_1(y_1)dy_1 \int h_2(y_2)p_2(y_2)dy_2\\ &= E\{h_1(y_1)\}E\{h_2(y_2)\}.\end{aligned}$$

反过来不能推出。

比如，y_1和y_2的联合分布如下(0,1)，(0,-1)，(1,0)，(-1,0)。

$$E(y_1,y_2)=E(y_1)E(y_2)=0$$

因此y_1和y_2不相关，但是

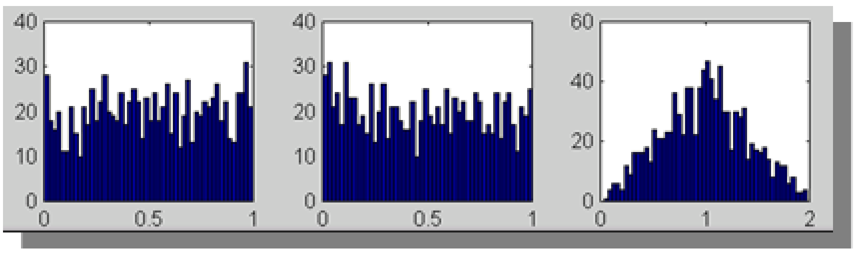
$$\mathrm{E}(y_1^2y_1^2)=0\neq\frac{1}{4}=E(y_1^2)E(y_2^2)$$

因此y_1和y_2不满足上面的积分公式，y_1和y_2不是独立的。

上面提到过，如果$s^{(i)}$是高斯分布的，A 是正交的，那么$x^{(i)}$也是高斯分布的，且$x^{(i)}$与$x^{(j)}$之间是独立的。那么无法确定 A，因为任何正交变换都可以让$x^{(i)}$达到同分布的效果。但是如果$s^{(i)}$中只有一个分量是高斯分布的，仍然可以使用 ICA。

那么 ICA 要解决的问题变为：如何从 x 中推出 s，使得 s 最不可能满足高斯分布？

中心极限定理告诉我们：大量独立同分布随机变量之和满足高斯分布。



$f(s_1)$ $f(s_2)$ $f(x_1)=f(s_1+s_2)$

我们一直假设的是$x^{(i)}$是由独立同分布的主元$s^{(i)}$经过混合矩阵 A 生成。那么为了求$s^{(i)}$，我们需要计算$s^{(i)}$的每个分量$y_j^{(i)} = w_j^T x^{(i)}$。定义$z_j = A^T w_j$，那么$y_j^{(i)} = w_j^T x^{(i)} = w_j^T A s^{(i)} = z_j^T s^{(i)}$，之所以这么麻烦再定义 z 是想说明一个关系，我们想通过整出一个w_j来对$x^{(i)}$进行线性组合，得出 y。而我们不知道得出的 y 是否是真正的 s 的分量，但我们知道 y 是 s 的真正分量的线性组合。由于我们不能使 s 的分量成为高斯分布，因此我们的目标求是让 y（也就是$w_j^T x^{(i)}$）最不可能是高斯分布时的 w。

那么问题递归到如何度量 y 是否是高斯分布的了。

一种度量方法是 kurtosis 方法，公式如下：

$$\operatorname{kurt}(y) = E\{y^4\} - 3(E\{y^2\})^2$$

如果 y 是高斯分布，那么该函数值为 0，否则绝大多数情况下值不为 0。

但这种度量方法不怎么好，有很多问题。看下一种方法：

负熵（Negentropy）度量方法。

我们在信息论里面知道对于离散的随机变量 Y，其熵是

$$H(Y) = -\sum_i P(Y = a_i)\log P(Y = a_i)$$

连续值时是

$$H(\mathbf{y}) = -\int f(\mathbf{y})\log f(\mathbf{y})d\mathbf{y}.$$

在信息论里有一个强有力的结论是：高斯分布的随机变量是同方差分布中熵最大的。也就是说对于一个随机变量来说，满足高斯分布时，最随机。

定义负熵的计算公式如下：

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y})$$

也就是随机变量 y 相对于高斯分布时的熵差，这个公式的问题就是直接计算时较为复杂，一般采用逼近策略。

$$J(y) \approx \frac{1}{12}E\{y^3\}^2 + \frac{1}{48}\operatorname{kurt}(y)^2$$

这种逼近策略不够好，作者提出了基于最大熵的更优的公式：

$$J(y) \approx \sum_{i=1}^{p} k_i[E\{G_i(y)\} - E\{G_i(\nu)\}]^2,$$

之后的 FastICA 就基于这个公式。

另外一种度量方法是最小互信息方法：

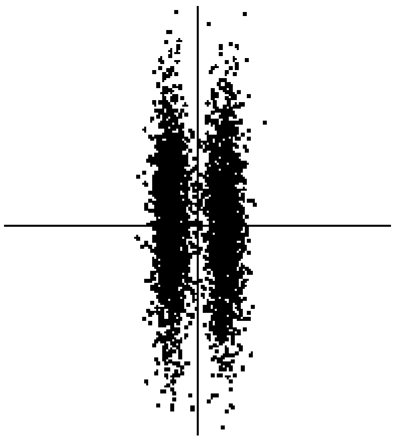
$$I(y_1, y_2, \ldots, y_m) = \sum_{i=1}^{m} H(y_i) - H(\mathbf{y}).$$

这个公式可以这样解释，前一个 H 是y_i的编码长度（以信息编码的方式理解），第二个 H 是 y 成为随机变量时的平均编码长度。之后的内容包括 FastICA 就不再介绍了，我也没看懂。

8. ICA 的投影追踪解释（Projection Pursuit）

投影追踪在统计学中的意思是去寻找多维数据的“interesting”投影。这些投影可用在数据可视化、密度估计和回归中。比如在一维的投影追踪中，我们寻找一条直线，使得所有的数据点投影到直线上后，能够反映出数据的分布。然而我们最不想要的是高斯分布，最不像高斯分布的数据点最 interesting。这个与我们的 ICA 思想是一直的，寻找独立的最不可能是高斯分布的 s。

在下图中，主元是纵轴，拥有最大的方差，但最 interesting 的是横轴，因为它可以将两个类分开（信号分离）。



9. ICA 算法的前处理步骤

1、**中心化**：也就是求 x 均值，然后让所有 x 减去均值，这一步与 PCA 一致。

2、**漂白**：目的是将 x 乘以一个矩阵变成$\tilde{x}$，使得$\tilde{x}$的协方差矩阵是I。解释一下吧，原始的向量是 x。转换后的是$\tilde{x}$。

$\tilde{x}$的协方差矩阵是I，即

$$E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\} = \mathbf{I}.$$

我们只需用下面的变换，就可以从 x 得到想要的$\tilde{x}$。

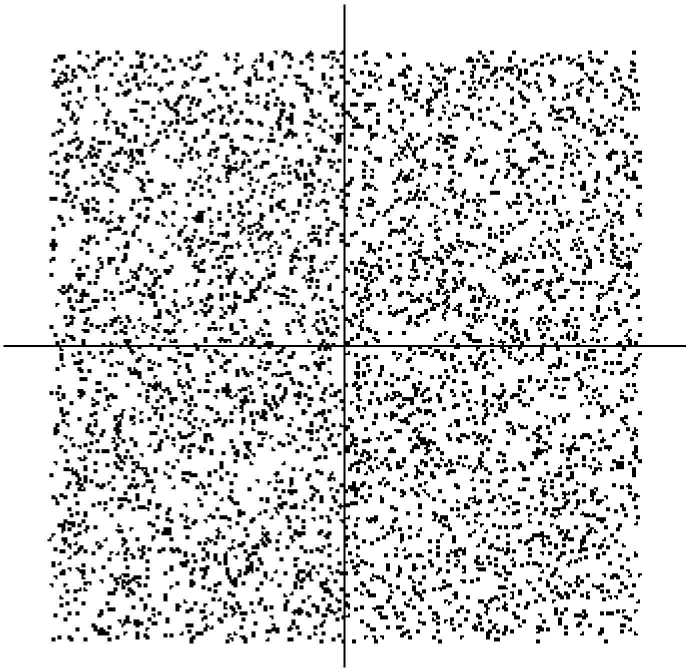
$$\tilde{\mathbf{x}} = \mathbf{E}\mathbf{D}^{-1/2}\mathbf{E}^T\mathbf{x}$$

其中使用特征值分解来得到 E（特征向量矩阵）和 D（特征值对角矩阵），计算公式为

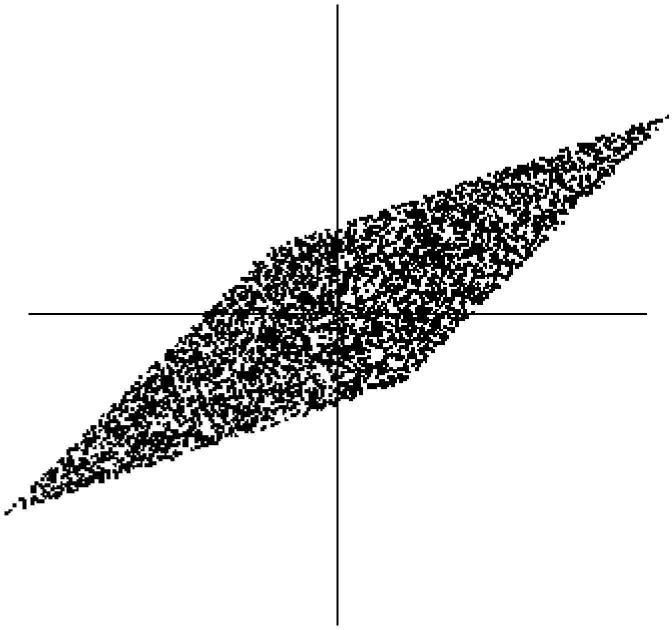
$$E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{E}\mathbf{D}\mathbf{E}^T$$

下面用个图来直观描述一下：

假设信号源 s1 和 s2 是独立的，比如下图横轴是 s1，纵轴是 s2，根据 s1 得不到 s2。

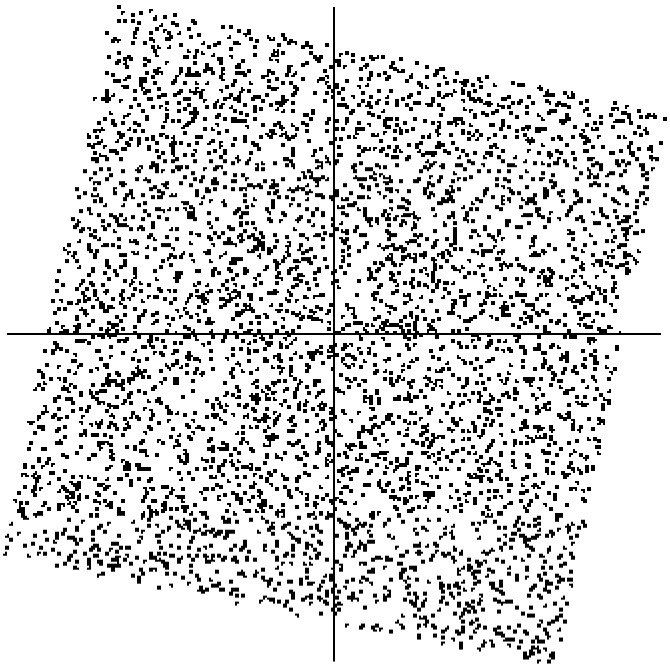


我们只知道他们合成后的信号 x，如下



此时 x1 和 x2 不是独立的（比如看最上面的尖角，知道了 x1 就知道了 x2）。那么直接代入我们之前的极大似然概率估计会有问题，因为我们假定 x 是独立的。

因此，漂白这一步为了让 x 独立。漂白结果如下：



可以看到数据变成了方阵，在x的维度上已经达到了独立。

然而这时 x 分布很好的情况下能够这样转换，当有噪音时怎么办呢？可以先使用前面提到的 PCA 方法来对数据进行降维，滤去噪声信号，得到 k 维的正交向量，然后再使用 ICA。

10. 小结

ICA 的盲信号分析领域的一个强有力方法，也是求非高斯分布数据隐含因子的方法。从之前我们熟悉的样本-特征角度看，我们使用 ICA 的前提条件是，认为样本数据由独立非高斯分布的隐含因子产生，隐含因子个数等于特征数。而 PCA 认为特征是由 k 个正交的特征（也可看作是隐含因子）生成的。同是因子分析，一个用来更适合用来还原信号（因为信号比较有规律，经常不是高斯分布的），一个更适合用来降维（用那么多特征干嘛，k 个正交的即可）。有时候也需要组合两者一起使用。这段是我的个人理解，仅供参考。