接下来我们要讲的就是独立成分分析（Independent Components Analysis，ICA）算法。这个方法和主成分分析（PCA）类似，也是要找到一组新的**基向量**（basis）来**表征**（represent）样本数据。然而，这两个方法的目的却截然不同。

我们还以 “鸡尾酒会问题” 为例。在一个聚会中，有 n 个人同时说话，并且放置在房间内的任何话筒仅记录这 n 个人叠加在一起的声音。如果假设我们也有 n 个不同的话筒安装在屋子里，并且这些话筒与每个说话人的距离都各自不同，那么记录的也就是不同组合形式的叠加声音。使用这样布置的 n 个话筒来录音，能不能区分开原始的 n 个说话者每个人的声音信号呢?

把这个问题用方程的形式来表示，我们需先假设时刻 i 有某样本数据 $s^{(i)}=(s_1^{(i)},s_2^{(i)},\dots,s_n^{(i)})^T\in R^n$，这个数据是**由 n 个独立的源（independent sources）在时刻 i 生成**的，例如 $s_j^{(i)}$ 是第 j 个说话者在时刻 i 发出的声音。总样本数据为 $s=(s^{(1)},s^{(2)},\dots,s^{(m)})\in R^{n\times m}$，表示 n 个说话者在 m 个时刻中生成的总数据。我们观察到的则为:

$$
\begin{gathered}
x^{(i)}=As^{(i)}\\
x=As
\end{gathered}
$$

上面式子中的 A 是一个未知的方形矩阵，叫做混合矩阵（mixing matrix）。通过重复的观察，我们就得到了训练样本 $x=\{x^{(i)};\ i=1,\dots,m\}\in R^{n\times m}$，其中 $x^{(i)}$ 类似于 $s^{(i)}$，例如 $x_j^{(i)}$ 是第 j 各个话筒在时刻 i 记录的声音。我们的目的是恢复出这些样本 $x^{(i)}=As^{(i)}$ 的原始声音源 $s^{(i)}$。

$$
x=\begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \cdots & x^{(m)} \\ | & | & & | \end{bmatrix}=\begin{bmatrix} | & | & & | \\ As^{(1)} & As^{(2)} & \cdots & As^{(m)} \\ | & | & & | \end{bmatrix}=A\begin{bmatrix} s_1^{(1)} & s_1^{(2)} & \cdots & s_1^{(m)} \\ s_2^{(1)} & s_2^{(2)} & \cdots & s_2^{(m)} \\ \vdots & \vdots & \ddots & \vdots \\ s_n^{(1)} & s_n^{(2)} & \cdots & s_n^{(m)} \end{bmatrix}=As
$$

令 $W=A^{-1}$，称之为还原矩阵（unmixing matrix）。我们的目标就是找到 W，这样针对给定的 $x^{(i)}$，就可以通过计算 $s^{(i)}=Wx^{(i)}$ 来还原出声音源。为了方便，我们用 w_j^T 来表示 W 的第 j 行，如下:

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

其中，$w_j\in R^n$，然后就可以通过计算 $s_j^{(i)}=w_j^Tx_j^{(i)}$ 恢复出第 j 个声源了。

1. 独立成分分析的不确定性（ICA ambiguities）

下面是两个独立成分分析的不确定性:

第一个，由于 W 和 s 都不确定，那么在没有先验证知识的情况下，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 只需要扩大两倍即可，等式仍然成立，因此无法得到唯一的 s。

第二个，如果将 s 的顺序打乱，变成另外一个顺序，那么只需要调整 A 的列向量顺序即可，因此也无法单独确定 s 。

另外，还有一种独立成分分析不适用的情况，那就是信号不能是**高斯分布**的。当源信号是高斯分布的时候，可以由不同的混合矩阵 A 乘以 s，得到相同的 x，一样无法确定源信号。

2. 密度函数和线性变换

在继续推导独立成分分析（ICA）算法之前，我们先来简要讲一讲线性变换对密度函数的影响（effect）。

假设我们有一个概率密度函数为 $p_s(s)$ 的随机变量 s。简单起见，我们把 s 当做一个实数，即 $s \in R$。然后，设又有一随机变量 x，x 定义为 $x = As$。那么 x 的概率密度函数 $p_x(x)$ 是多少呢？在计算之前，我们先讨论一下概率分布函数 $F(x) = P(X \leq x)$ 与概率密度函数 $f(x) = p(x)$ 之间的关系吧，事实上，概率分布函数是概率密度函数的积分，概率密度函数是概率分布函数的导数，二者关系如下：

$$
\begin{aligned}
F(x) &= \int_{-\infty}^{x} f(x)dx \\
f(x) &= F^{'}(x)
\end{aligned}
$$

因此，关于 $p_x(x)$ 的推导如下：

$$
\begin{aligned}
F_x(x) &= P(X \leq x) = P(AS \leq x) = P(S \leq Wx) = F_s(Wx) \\
p_x(x) &= F_x^{'}(x) = F_s^{'}(Wx) = p_s(Wx)|W|
\end{aligned}
$$

即，$p_x(x) = p_s(Wx)|W|$（$|W|$ 其实也可以分步求导解释）

3. 独立成分分析算法

现在就可以推导独立成分分析（ICA）算法了。我们这里描述的算法来自于 Bell 和 Sejnowski，对算法的解释使用**最大似然估计**的方法。我们假设每个声源的分布 s_i 的概率密度函数为 p_s，那么联合分布 s 则为：

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

这里要注意，由于我们假设每个声源都是独立的，故可以在建模中将联合分布直接写成边界分布的乘积。利用上一节推导的结论，这就表明 $x = As = W^{-1}s$ 的密度函数为：

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

剩下的就是为单个源 p_s 指定一个密度。回忆一下，给定一个实数值的随机变量 z，其累积分布函数（即概率分布函数）F 的定义为：$F(z_0) = P(z \leq z_0) = \int_{-\infty}^{z_0} p_z(z)dz$。然后对这个累积分布函数求导，就能得到 z 的密度函数：$p(_z z) = F^{'}(z)$。

因此，要确定 s_i 的密度函数，首先要做的就是确定其累积分布函数。根据我们之前的讨论，这里不能选用高斯分布的累积分布函数，因为独立成分分析不适用于高斯分布的数据。这里我们选择一个保证从 0 到 1 单调递增函数即可，比如 s 形函数（sigmoid function）$g(s) = 1/(1+e^{-s})$。这样就有：$p_s(s) = g^{'}(s)$。

模型中的参数 W 是一个正方形矩阵。给定一个训练集合 $\{x^{(i)}; i = 1, \cdots, m\}$，对数似然函数则为：

$$l(W) = \sum_{i=1}^{m} \Big(\sum_{j=1}^{n} \log g^{'}(w_j^T x^{(i)}) + \log |W| \Big)$$

我们要做的就是选择合适的 W 最大化上式。通过求导和前面讲义中的定理：$\nabla_W |W| = |W| (W^{-1})^T$，就可以很容易推导出**随机梯度上升**的学习规则。对一个给定的训练样本 $x^{(i)}$，这个更新规则为：

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

上式中的 α 是学习速率。在算法收敛之后，就能计算出 $s^{(i)} = W x^{(i)}$，这样就能恢复出原始的音源了。