Author：Liu Jian

Time：2021-06-28

机器学习II-特征工程与稀疏表示

0 特征工程概述

事实上，我们都是在特征空间中进行模型的学习。若问题比较简单，输入数据的原始空间，也就是输入空间就可作为特征空间。但更多时候，我们在得到输入数据后，还需对输入数据进行处理，将其映射到合适的特征空间中以便进行机器学习，数据的这一转换过程被称为特征工程。

特征工程包括：特征降维和特征选择。特征降维就是对原有的一组特征进行数学变换，变换成数量更少的一组特征。而特征选择就是从众多特征中剔除不重要的特征并保留重要特征，虽然简单粗暴，但是极其容易实现，并易于使用，所以在工程中特征选择的使用频率要高于特征降维。

特征工程的目的：(1) 减少特征维数，使计算开销更小，采样密度变大；(2) 使转换后的低维特征更具特征性，使学习器学习起来更容易。

1 特征降维

人们观测或收集到的数据样本虽是高维的，但与学习任务密切相关的也许仅是某个低维分布，即高维空间中的一个低维嵌入 (embedding)，在此低维空间中进行学习要比在原始空间中的性能更好。显然，降维会舍弃掉一部分信息，但当数据受到噪声影响时，最小特征值所对应的特征向量往往与噪声有关，将它们舍弃能在一定程度上起到去噪的效果。降维效果的评估通常是比较降维前后学习器的性能，若将维数降至二维或三维，则可通过可视化技术来直观地判断降维效果。

原始空间维度：d

低维空间维度：d'

样本容量：N

原始样本数据：$X_{d\times N}=\langle \boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_N\rangle$，其中第 i 个样本点 $\boldsymbol{x}_i=\langle x_i^{(1)},x_i^{(2)},\cdots,x_i^{(d)}\rangle^T$

降维后样本数据：$Z_{d'\times N}=\langle \boldsymbol{z}_1,\boldsymbol{z}_2,\cdots,\boldsymbol{z}_N\rangle$，其中第 i 个样本点 $\boldsymbol{z}_i=\langle z_i^{(1)},z_i^{(2)},\cdots,z_i^{(d')}\rangle^T$

线性降维：基于线性变换来进行降维的方法称为线性降维方法，其形式均为 $Z=W^TX$，其中 $W_{d\times d'}=\langle \boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_{d'}\rangle$ 是变换矩阵，不同之处在于对低维子空间性质的要求不同，即对 W 施加了不同的约束。事实上，我们有 $\boldsymbol{z}_i=W^T\boldsymbol{x}_i,\ i=1,2,\cdots,N$，可以看到：(1) 新空间中的属性是原空间中属性的线性组合，第 i 维的组合方式由 $\boldsymbol{w}_i,\ i=1,2,\cdots,d'$ 给出；(2) $\boldsymbol{z}_i$ 是原属性向量 $\boldsymbol{x}_i$ 在新坐标系 $=\langle \boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_{d'}\rangle$ 中的坐标向量，一般地，我们会令 $\|\boldsymbol{w}_i\|_2=1$，此外，若 $\boldsymbol{w}_i,\boldsymbol{w}_j$ ($i\neq j$) 正交，则新坐标系是一个正交坐标系。

1.1 主成分分析 (Principal Component Analysis, PCA)

> 1. 知乎 - PCA 的四重境界
> 2. PCA - 最小均方误差/最小二乘 PPT
> 3. A Tutorial on Principal Component Analysis - Jonathon Shlens：PCA 的 tutorial，版本还在不断更新，最新是 3.02 版，不过这个 tutorial 写得并不好，从历次更新的版本来看，有些问题作者自己都没有想清楚，对 PCA 的理解并不通透，参考价值不大。

PCA 属于**无监督线性降维**，从如下四种思路出发均可推得 PCA：

1. **最近重构性**：重构的样本 (降维后的样本映射回原空间即为重构的样本) 与原样本的距离都足够的近；
2. **最大可分性**：降维后的样本点尽可能分开，也就是方差越大越好。在信号处理中，信号的方差通常较大，而噪声的方差通常较小，信噪比 (signal-to-noise ratio, SNR) 指信号与噪声的方差之比 $\sigma_{signal}^2/\sigma_{noise}^2$，信噪比越大说明数据的质量越好。**可见方差越大，越有可能是我们所想要的信号而不是噪声；或者说方差越大，数据所含的有效信息就越多。因此，PCA 认为降维后的样本方差应该尽可能大，更具体一点就是样本数据在我们所选择的相互正交坐标轴上的方差之和应该尽可能大。**
3. **最小均方误差/总最小二乘：详细内容参见笔记《6-矩阵论》。PCA 在寻找一个超曲面 $S:\ \boldsymbol{x}-WW^T\boldsymbol{x}=\boldsymbol{0}$，其中 W 为待定矩阵，使得样本点离超曲面欧式距离的平方和或者说均方误差尽可能小。可以看到，PCA 实际上就是在用超曲面对样本数据进行总最小二乘 (TLS；注意，不是我们常接触的普通最小二乘，即 OLS) 拟合。基于上面的论述，我们可以得到最近重构性和最大可分性比较形象的解释：比较好的拟合应该是样本点与其在超曲面上的投影点应尽可能靠近，即最近重构性；而对于最大可分性，我们可以想象使用一条直线对二维样本点进行总最小二乘拟合，改变直线的角度，可以看到当直线对样本拟合得最好时，这条直线应趋于穿过所有的样本点，此时，样本点在直线上的投影是最发散的，即最大可分性。这也就非正式地证明了最近重构性、最大可分性和最小均方误差三者是等价的。**

4. **逐一选取方差最大方向**：上述最大可分性是选取方差之和最大的 d' 个相互正交的方向作为新的坐标轴，而逐一选取方差最大方向则类似于**贪心算法**：每次只选取一个坐标轴，使得原始数据在该轴上的投影方差最大，且原始数据在各坐标轴上的投影之间是不相关的 (从下面的推导可以看到，这一条件实际上就要求选取的各坐标轴要相互正交)，依次进行 d' 次选取即得降维后的坐标系。

我们指出，**分别基于最近重构性、最大可分性和最小均方误差进行推导，所得问题的形式相同；而逐一选取方差最大方向所解决问题的形式虽然与前面三者不同，但得到结果相同。**

1.1.1 最大可分性、最近重构性、最小均方误差

由最小均方误差推导 PCA 可参见笔记《6-矩阵论》；这里只给出基于最大可分性和最近重构性的推导。

假设原始样本数据 $X_{d\times N}=\langle \boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_N\rangle$ 已被中心化，即 $\sum_{i=1}^{N}\boldsymbol{x}_i=\boldsymbol{0}$；变换矩阵 $W_{d\times d'}=\langle \boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_{d'}\rangle$ 中各列为单位正交向量，即 $\boldsymbol{w}_i^T\boldsymbol{w}_j=\delta_{ij}$。我们可得降维后的样本点：$\boldsymbol{z}_i=W^T\boldsymbol{x}_i$，基于 $\boldsymbol{z}_i$ 重构 $\boldsymbol{x}_i$ 得到重构后的样本点：$\hat{\boldsymbol{x}}_i=W\boldsymbol{z}_i$，则由**最近重构性**可得如下的约束优化问题：

> Frobenius范数，简称 F-范数，是一种矩阵范数，简单来说就是矩阵的每个元素的平方和的开方，对于向量而言就是 L2 范数。 $\|A_{m\times n}\|_F^2=tr(AA^T)=tr(A^TA=)\|A_{m\times n}^T\|_F^2$。

$$\begin{aligned}
&\min_W\|\hat{X}-X\|_F^2=\min_W\sum_{i=1}^{N}\|\hat{\boldsymbol{x}}_i-\boldsymbol{x}_i\|_2^2=\min_W-\sum_{i=1}^{N}\boldsymbol{z}_i^T\boldsymbol{z}_i+\sum_{i=1}^{N}\boldsymbol{x}_i^T\boldsymbol{x}_i\\
&\Leftrightarrow\min_W-\sum_{i=1}^{N}\boldsymbol{z}_i^T\boldsymbol{z}_i=\min_W-\sum_{i=1}^{N}tr\left(\boldsymbol{z}_i\boldsymbol{z}_i^T\right)=\min_W-tr\left(\sum_{i=1}^{N}\boldsymbol{z}_i\boldsymbol{z}_i^T\right)\\
&=\min_W-tr\left(W^T\left(\sum_{i=1}^{N}\boldsymbol{x}_i\boldsymbol{x}_i^T\right)W\right)=\min_W-tr\left(W^TXX^TW\right)\\
&s.t.\ \ {W^T}_{d'\times d}W_{d\times d'}=I_{d'\times d'}
\end{aligned}$$

而根据**最大可分性**，降维后的样本点应尽可能分开，即使它们的方差最大化。而我们现在处理的是多维数据，可得协方差矩阵：

$$\begin{aligned}
&[\mathbb{COV}_{ij}]_{d'\times d'}\\
&=\frac{1}{N-1}\sum_{k=1}^{N}(\boldsymbol{z}_k-\overline{\boldsymbol{z}})(\boldsymbol{z}_k^T-\overline{\boldsymbol{z}}^T)=\frac{1}{N-1}\sum_{k=1}^{N}\boldsymbol{z}_k\boldsymbol{z}_k^T=\frac{1}{N-1}ZZ^T\\
&\text{where }\ \overline{\boldsymbol{z}}=\sum_{k=1}^{N}\boldsymbol{z}_k=\sum_{k=1}^{N}W^T\boldsymbol{x}_k=W^T\sum_{k=1}^{N}\boldsymbol{x}_k=\boldsymbol{0}
\end{aligned}$$

其对角线元素为各维的方差，由此我们的优化目标为：

$$\begin{aligned}
&\max_W tr\left(\frac{1}{N-1}ZZ^T\right)\Leftrightarrow\max_W tr\left(ZZ^T\right)\\
&=\max_W tr\left(W^TXX^TW\right)\\
&s.t.\ \ {W^T}_{d'\times d}W_{d\times d'}=I_{d'\times d'}
\end{aligned}$$

与最近重构性所推得的问题相同。

接下来，我们**采用拉格朗日乘数法求解上述约束优化问题。**

问题的约束条件是一个矩阵形式，这意味着矩阵的每一个元素都要满足对应的条件，对应有 $d'\times d'$ 个等式约束，即：

$$\|\boldsymbol{w}_i\|_2 = 1,\ \ i = 1, 2, \cdots, d'$$
$$\boldsymbol{w}_i^T \boldsymbol{w}_j = 0,\ \ (i \neq j)$$

我们可以使用拉格朗日乘数矩阵 $\Theta_{d' \times d'}$ 引入约束条件：

$$\mathcal{L} = tr\left(W^T X X^T W\right) + tr\left(\Theta^T (W^T W - I)\right)$$

但事实上，我们先只考虑约束条件：

$$\|\boldsymbol{w}_i\|_2 = 1,\ \ i = 1, 2, \cdots, d'$$

对应地拉格朗日乘数矩阵 $\Lambda = diag(\lambda_1, \lambda_2, \cdots, \lambda_{d'})$，则
$\mathcal{L}(W, \Lambda) = tr\left(W^T X X^T W\right) + tr\left(\Lambda^T (W^T W - I)\right)$，我们令其对 W 的偏导为 $\mathbf{0}$，可得：

$$\frac{\partial \mathcal{L}(W, \Lambda)}{\partial W} = -2XX^T W + 2W\Lambda = \mathbf{0}$$
$$\left(\text{求导公式：}\ \frac{\partial tr(X^T A X)}{\partial X} = AX + A^T X,\ \ \frac{\partial tr(AX^T X)}{\partial X} = XA + XA^T\right)$$
$$\Updownarrow$$
$$XX^T \boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i,\ \ i = 1, 2, \cdots, d'$$

显然，此式为矩阵特征值和特征向量的定义式，其中 $\lambda_i, \boldsymbol{w}_i$ 分别表示矩阵 XX^T 的特征值和特征向量。可以看到，XX^T 是一个实对称矩阵，**实对称矩阵的不同特征值所对应的特征向量之间相互正交，同一特征值的不同特征向量可以通过施密特正交化使其变得正交**，所以通过上式求得的 $\boldsymbol{w}_i$ 可以同时满足约束条件 $\|\boldsymbol{w}_i\|_2 = 1, \boldsymbol{w}_i^T \boldsymbol{w}_j = 0,\ \ (i \neq j)$。根据拉格朗日乘子法的原理可知，此时求得的结果仅是最优解的必要条件，我们还需从 d 个特征向量里找出 d' 个能使得目标函数达到最优值的特征向量作为最优解。我们将 $XX^T \boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i$ 代入到目标函数中：

$$\max_W tr(W^T X X^T W) = \max_W \sum_{i=1}^{d'} \boldsymbol{w}_i^T X X^T \boldsymbol{w}_i = \max_W \sum_{i=1}^{d'} \boldsymbol{w}_i^T \lambda_i \boldsymbol{w}_i = \sum_{i=1}^{d'} \lambda_i$$

显然，此时只需要令 $\lambda_1, \lambda_2, \cdots, \lambda_{d'}$ 和 $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_{d'}$ 分别为矩阵 XX^T 的前 d' 个最大特征值和单位正交特征向量即可。

综上，**PCA 的步骤**：

1. 数据中心化：$\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i - \frac{1}{N}\sum_{i=1}^N \boldsymbol{x}_i$
2. 计算样本协方差矩阵 XX^T，并对其特征分解；
3. 取 d' 个最大特征值和单位正交特征向量，得到投影矩阵：$W = \langle \boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_{d'} \rangle$。其中，$d'$ 人为给定，使得 $\sum_{i=1}^{d'} \lambda_i / \sum_{i=1}^{d} \lambda_i$ 大于某一个给定的阈值如 95%。

1.1.2 逐一选取方差最大方向

> 知乎 - PCA 的最大方差推导

如前所述，采用贪心算法构建新空间的坐标系的步骤如下：

1. 寻找一个 d 维单位向量 $\boldsymbol{w}_1$，使样本数据 $X_{d \times N}$ 投影 $\boldsymbol{w}_1^T X$ 的方差最大；
2. 寻找另一个 d 维单位向量 $\boldsymbol{w}_2$，使样本数据 $X_{d \times N}$ 在其上的投影后 $\boldsymbol{w}_2^T X$ 与 $\boldsymbol{w}_1^T X$ 不相关，并且 $\boldsymbol{w}_2^T X$ 的方差最大；
3. 重复上述步骤，直到 d' 个坐标轴构建完毕。

基于上述步骤，我们的推导如下：

1. 我们首先选取第一个坐标轴 $\boldsymbol{w}_1$。$\boldsymbol{w}_1^T X$ 的方差为 $(\boldsymbol{w}_1^T X - \mathbf{0})(\boldsymbol{w}_1^T X - \mathbf{0})^T / (N-1) = \boldsymbol{w}_1^T X X^T \boldsymbol{w}_1$，约束条件是 $\boldsymbol{w}_1^T \boldsymbol{w}_1 = 1$，问题即为：

$$\max_{\boldsymbol{w}_1} \boldsymbol{w}_1^T X X^T \boldsymbol{w}_1$$
$$s.t.\ \ \boldsymbol{w}_1^T \boldsymbol{w}_1 = 1$$

使用拉格朗日乘数法，令拉格朗日函数对 $\boldsymbol{w}_1$ 的偏导为 $\mathbf{0}$ 得：

$$XX^T \boldsymbol{w}_1 = \lambda_1 \boldsymbol{w}_1$$

则 $\boldsymbol{w}_1$ 就是 XX^T 的单位特征向量，拉格朗日乘子 λ_1 为对应的特征值。但特征值和特征向量那么多，我们该如何选取呢？我们回到原式，发现：

$$\boldsymbol{w}_1^T X X^T \boldsymbol{w}_1 = \boldsymbol{w}_1^T \lambda_1 \boldsymbol{w}_1 = \lambda_1$$

则选取最大的特征值对应的特征向量即可。

2. 接着，我们选取第二个坐标轴 $\boldsymbol{w}_2$ 。 $\boldsymbol{w}_1^T X$ 与 $\boldsymbol{w}_2^T X$ 不相关：

$$\mathbb{COV}(\boldsymbol{w}_1^T X, \boldsymbol{w}_2^T X) = 0$$
$$\frac{1}{N-1}(\boldsymbol{w}_1^T X - \mathbf{0})(\boldsymbol{w}_2^T X - \mathbf{0})^T = 0$$
$$\boldsymbol{w}_2^T X X^T \boldsymbol{w}_1 = \lambda_1 \boldsymbol{w}_2^T \boldsymbol{w}_1 = 0$$

可以看到 $\boldsymbol{w}_1^T X$ 与 $\boldsymbol{w}_2^T X$ 不相关就意味着两个坐标轴正交。此时，问题即为：

$$\max_{\boldsymbol{w}_2} \boldsymbol{w}_2^T X X^T \boldsymbol{w}_2$$
$$s.t.\ \ \boldsymbol{w}_2^T \boldsymbol{w}_2 = 1,\ \boldsymbol{w}_2^T \boldsymbol{w}_1 = 0$$

使用拉格朗日乘数法，令拉格朗日函数对 $\boldsymbol{w}_2$ 的偏导为 $\mathbf{0}$ 可得：

$$XX^T \boldsymbol{w}_2 - \lambda_2 \boldsymbol{w}_2 - \mu_1 \boldsymbol{w}_1 = \mathbf{0}$$

对上式左乘 $\boldsymbol{w}_1^T$ 可得：

$$\boldsymbol{w}_1^T X X^T \boldsymbol{w}_2 - \lambda_2 \boldsymbol{w}_1^T \boldsymbol{w}_2 - \mu_1 \boldsymbol{w}_1^T \boldsymbol{w}_1 = 0 - 0 - \mu_1 = 0$$

则 $\mu_1 = 0$ ，则：

$$XX^T \boldsymbol{w}_2 = \lambda_2 \boldsymbol{w}_2$$

同样地， $\boldsymbol{w}_2$ 就是 XX^T 的单位特征向量，拉格朗日乘子 λ_2 为对应的特征值；又 $\boldsymbol{w}_2^T X X^T \boldsymbol{w}_2 = \lambda_2$ ，则 λ_2 为 XX^T 第二大的特征值。此外， XX^T 为实对称矩阵，则当 $\lambda_2 \neq \lambda_1$ 时， $\boldsymbol{w}_2, \boldsymbol{w}_1$ 正交；而当 $\lambda_2 = \lambda_1$ 也就是特征值出现重根的时候，我们总可以通过施密特正交化使得 $\boldsymbol{w}_2, \boldsymbol{w}_1$ 正交使得约束条件满足。

3. 重复上述步骤，我们可以看到，**这样的贪心算法与直接选取方差之和最大的 d' 个相互正交的坐标轴所得结果是相同的。**

1.1.3 有关 PCA 的进一步评述

1. 我们知道奇异值分解 (SVD) $A_{m\times n} = U_{m\times m}\Sigma_{m\times n}V_{n\times n}^T$ 中右奇异向量满足： $AA^T \boldsymbol{u}_i = \lambda_i \boldsymbol{u}_i$ 。可以看到， $\boldsymbol{w}_i$ 实际上就是对 $X_{d\times N}$ 进行奇异值分解所得到的右奇异向量。考虑到奇异值分解存在着比较高效的算法，因此，**实践中常通过对 X 进行奇异值分解来代替协方差矩阵的特征值分解。**有关奇异值分解的更多内容参见笔记《6-矩阵论》。
2. **看到均方误差，我们就应该想到高斯分布。但仔细分析我们可以看到 PCA 并没有假设或潜在地认为样本数据服从高斯分布，但误差最好服从高斯分布，参见笔记《6-矩阵论》。对于下文将要介绍的 LDA，可以看到其潜在地假设降维后的同类样本服从高斯分布，高斯分布在 LDA 中的痕迹比 PCA 更重。**

3. **我们提到样本方差是指某个维度数据的方差，而样本协方差是指两个维度数据间的协方差。某个维度与 X 中的某一行对应，可看做某个随机变量或某个信号源 (在 PCA 中被称为主成分)，这一行中的具体数据就是这个随机变量依次采样的结果或者说信号源依次发射的信号。**在 PCA 中，降维样本 W^TX：

$$W^TX = \begin{bmatrix} \boldsymbol{w}_1^TX \\ \vdots \\ \boldsymbol{w}_{d'}^TX \end{bmatrix}$$

其中每一维样本 $\boldsymbol{w}_i^TX$ 就是随机变量 $\boldsymbol{w}_i^T\boldsymbol{x}$ 采样所得数据，或则说信号源 $\boldsymbol{w}_i^T\boldsymbol{x}$ 发射的信号；而 X 就是随机向量 $\boldsymbol{x}$ 采样所得数据，或者说信号源 $\boldsymbol{x}$ 发射的信号。**显然，我们找到的 W 只能使随机变量 $\boldsymbol{w}_i^T\boldsymbol{x}, \boldsymbol{w}_j^T\boldsymbol{x}$ 的协方差的估计量为 $\mathbf{0}$，即 $\mathbb{COV}(\boldsymbol{w}_i^T\boldsymbol{x}, \boldsymbol{w}_j^T\boldsymbol{x})$ 的估计量 $\mathbb{COV}(\boldsymbol{w}_1^TX, \boldsymbol{w}_2^TX) = 0$，严格地讲，我们并不能说 $\mathbb{COV}(\boldsymbol{w}_i^T\boldsymbol{x}, \boldsymbol{w}_j^T\boldsymbol{x}) = 0$，即随机变量 $\boldsymbol{w}_i^T\boldsymbol{x}, \boldsymbol{w}_j^T\boldsymbol{x}$ 线性无关。而对于具体的数据，不存在线性无关，相互独立等说法，因为这些概念是针对随机变量而定义的。同样地，ICA 中分离得到的每个方向都只是使样本数据的非高斯性达到最大，即随机变量或信号源非高斯性的估计量达到最大，而不是随机变量或信号源的非高斯性达到最大。类似地，还有样本数据中心化等数据预处理操作，样本数据中心化只是使随机变量均值的估计量为 0，严格地讲并不能保证随机变量的均值就是 0。尽管存在上述区别，但我们一般还是不加区分地使用，不过我们对此还是应该做到心中有数。因此，这里我们说 PCA 的各主成分间线性无关。**

若不仅误差服从高斯分布，而且整个数据都服从高斯分布，则 PCA 得到的各主成分不仅线性无关，而且相互独立。因为对于高斯分布而言，线性无关等价于相互独立。

3. 随机向量的各分量表示的物理量可能不同，比如某个维度的分量记录的是重量信息，而另一个维度的分量记录的则是长度信息等，而一个物理量可取不同的单位，比如重量可取千克、克，长度可取米、毫米。**对于同一组样本数据，选择不同量级的单位，会使各个维度间数据的相对大小产生变化，由此 PCA 给出的结果也会不同，这是总最小二乘均会面临的问题。**我们通常会对各个维度的数据进行中心化和标准化处理，以消除量纲的影响，但这么做也并非完全没有问题，因为这可能会放大某个次要分量或减小某个重要分量的影响。
4. PCA 为**无参数方法**，无法引入先验信息；此外，PCA 假设通过线性变换就可以揭示样本数据的主要或者说特征结构，但显然有时这一假设并不成立，我们需要借助非线性变换才能较好地处理问题。**对于非线性降维，我们可以使用 KPCA，且 KPCA 是一种参数方法。**

1.2 线性判别分析 (Linear Discriminant Analysis, LDA)

> 博客园-线性判别分析LDA原理总结

首先，自然语言处理领域的 LDA 是指隐含狄利克雷分布 (Latent Dirichlet Allocation，简称LDA)，是一种处理文档的主题模型，而这里讨论的 LDA 是指线性判别分析，是一种**监督线性降维技术。**LDA 的想法是：**投影后类内方差尽量小，类间方差尽量大**，即希望投影后在低维空间中，同类别的数据尽可能集中，而类别不同数据的中心之间的距离尽可能大。可以看到，**LDA 选择分类性能最好的投影方向，而 PCA 选择投影具有最大方差的方向。**因此，在样本分类信息依赖均值而不是方差的时候，LDA 比 PCA 更优；而在样本分类信息依赖方差而不是均值的时候，PCA 比 LDA 更优。

因为 LDA 为监督降维，因此对每个样本点 $\boldsymbol{x}_i$，我们还需考察其对应标记 $y_i \in \{c_1, c_2, \cdots, c_t\}$。记 N_j 为第 j 类样本的个数，$\boldsymbol{\mu}_j$ 为第 j 类样本的均值，$\boldsymbol{\mu}$ 为所有样本的均值：

$$\boldsymbol{\mu}_j = \frac{1}{N_j}\sum_{y_i=c_j}\boldsymbol{x}_i, \quad \boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_i$$

此外，**LDA 降维时，能达到的最大维数为样本类别数减一，即** $d' \leqslant t-1$，但 PCA 没有这个限制。

我们需要说明的是，**LDA 除了用于降维外，还可以用于分类。**一个常见的 LDA 分类思路是：假设各个类别的样本数据符合高斯分布，这样利用 LDA 进行投影后，可以利用极大似然估计计算各个类别投影数据的均值和方差，进而得到该类别高斯分布的概率密度函数。当一个新的样本到来后，我们可以将它投影，然后将投影后的样本点分别代入各个类别的高斯分布概率密度函数中，计算它属于这个类别的概率，最大的概率对应的类别即为预测类别。西瓜书上还说，二分类任务中，两类数据满足高斯分布且方差相同时，LDA 产生贝叶斯最优分类器。**我们这里主要讨论 LDA 用于降维的情况。**

1.2.1 二类 LDA

此时 $y_i \in \{c_1, c_2\} = \{0, 1\}$，因此我们可以将 d 维样本降到 1 维，投影矩阵 $W_{d\times 1}$ 实际上是一个向量。

- 不同类别数据的中心之间的距离可以用欧式距离的平方来衡量：

$$\|W^T\boldsymbol{\mu}_1 - W^T\boldsymbol{\mu}_2\|_2^2 = W^T(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T W$$

我们定义类间散度矩阵 $S_b = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T$ 。

- 同一种类别数据的投影点尽可能集中，也就是同类样本投影后方差尽可能小，可基于协方差矩阵进行量化。记 Σ_1 、 Σ_2 ：

$$\Sigma_1 = \sum_{y_i=c_1}(\boldsymbol{x}_i - \boldsymbol{\mu}_1)(\boldsymbol{x}_i - \boldsymbol{\mu}_1)^T$$
$$\Sigma_2 = \sum_{y_i=c_2}(\boldsymbol{x}_i - \boldsymbol{\mu}_2)(\boldsymbol{x}_i - \boldsymbol{\mu}_2)^T$$

为投影前两类样本不计算分母的协方差矩阵，均为 $d \times d$ 阶。则投影后样本不计算分母的协方差矩阵为 $W^T\Sigma_1 W$ 和 $W^T\Sigma_2 W$ ，二者此时已经为标量，它们的和为：

$$W^T\Sigma_1 W + W^T\Sigma_2 W = W^T(\Sigma_1 + \Sigma_2)W$$

我们定义类内散度矩阵 $S_w = \Sigma_1 + \Sigma_2$ 。

- 由此，转化为如下的优化问题：

$$\arg\max_W \frac{W^T S_b W}{W^T S_w W}$$

即最大化广义瑞利商 $R(w; S_b, S_w)$ ，可知 W 为 $S_w^{-1}S_b$ 的最大特征值对应的特征向量。

对于此问题，我们还可做进一步推导：首先，我们有 $S_w^{-1}S_b W = \lambda_{\max} W$ ，而 $S_b W = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T W = C(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ ，不妨令常数 $C = \lambda_{max}$ ，则可得 $W = S_w^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ 。

1.2.2 多类 LDA

对于多分类问题，我们将 d 维样本投影到一个 d' 维空间而不是一条直线上。类似地，我们定义：

类间散度矩阵： $S_b = \sum_{i=1}^{t} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$

类内散度矩阵： $S_w = \sum_{j=1}^{t}\Sigma_i = \sum_{j=1}^{t}\sum_{y_i=c_j}(\boldsymbol{x}_i - \boldsymbol{\mu}_j)(\boldsymbol{x}_i - \boldsymbol{\mu}_j)^T$

事实上，我们还有全局散度矩阵 $S_g = \sum_{i=1}^{N}(\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T$ ，三者之间的关系为： $S_g = S_b + S_w$ ，均为 $d \times d$ 的矩阵。

使用 $W_{d\times d'}$ 进行降维后，对应地有 $W^T S_b W, W^T S_w W, W^T S_g W$ ，但它们现在不是标量，而是 $d' \times d'$ 的矩阵，因而无法直接对 $W^T S_b W / W^T S_w W$ 进行优化。通常，我们会选择进行如下的优化：

$$\arg\max_{W}\frac{\prod_{diag}(W^TS_bW)}{\prod_{diag}(W^TS_wW)}=\frac{\prod_{i=1}^{d'}(\boldsymbol{w}_i^TS_b\boldsymbol{w}_i)}{\prod_{i=1}^{d'}(\boldsymbol{w}_i^TS_w\boldsymbol{w}_i)}=\prod_{i=1}^{d'}\frac{\boldsymbol{w}_i^TS_b\boldsymbol{w}_i}{\boldsymbol{w}_i^TS_w\boldsymbol{w}_i}$$

其中，记号 $\prod_{diag}A$ 表示 A 主对角线元素的乘积。**可以看到，上式最右即为广义瑞利商累积的形式，则其最大值就是矩阵 $S_w^{-1}S_b$ 最大的 d' 个非零特征值的乘积，转换矩阵 W 就由对应的 d' 个特征向量组成。**

S_b 中每个 $\boldsymbol{\mu}_i-\boldsymbol{\mu}$ 的秩为 1，因此相加后秩最大为 t (矩阵的秩小于等于各个相加矩阵秩的和)，但是由于最后一个 $\boldsymbol{\mu}_t-\boldsymbol{\mu}$ 可由前 $t-1$ 个 $\boldsymbol{\mu}_i-\boldsymbol{\mu}$ 线性表示，因此 S_b 的秩最大为 $t-1$。**这意味着 S_b 最多存在 $t-1$ 个非零特征值，也就是说降维样本的维数 d' 最多为 $t-1$。**

1.3 核化线性降维 (Kernelized PCA, KPCA)

线性降维时高维空间到低维空间的映射是线性函数，即我们假设通过线性变换就能很好地暴露样本数据的特征或者说主要结构，但显然有时候这一假设并不成立，有些问题需要借助非线性映射才能被较好地处理。非线性降维的一中常用方法是基于核技巧对线性降维方法进行核化 (kernelized)，这里介绍的核化线性降维就是一个例子。

我们在 SVM 里介绍，核技巧通过将低维空间里的样本映射到合适的高维空间，在高维空间中暴露问题的本质，降低问题的难度。因此，**KPCA 的思路就是先将原始样本 $X_{d\times N}$ 非线性映射到高维空间中得到高维样本 $U_{d^*\times N}=\phi(X)$，即 $\boldsymbol{u}_i=\phi(\boldsymbol{x}_i),\ i=1,2,\cdots,N$，再在高维空间中对数据进行 PCA 降维 $Z_{d'\times N}=W^TU=W^T\phi(X)$。**记核矩阵
$K_{N\times N}=U^TU,\ K_{ij}=\boldsymbol{u}_i^T\boldsymbol{u}_j=\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)=\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)$，由上一节的内容：

$$UU^T\boldsymbol{w}_j=\left(\sum_{i=1}^{N}\boldsymbol{u}_i\boldsymbol{u}_i^T\right)\boldsymbol{w}_j=\lambda_j\boldsymbol{w}_j$$

注意，$\{UU^T\}_{d^*\times d^*}\neq K_{N\times N}$ 并不是核矩阵，$\lambda_j,\boldsymbol{w}_j$ 分别为其第 j 大的特征值 (考虑重根) 和对应的单位正交特征向量。在使用核方法时，我们所面临的主要问题是计算问题，因为我们只知道高维内积函数 $\kappa(\cdot,\cdot)=\phi(\cdot)^T\phi(\cdot)$ 的计算公式，而不知道高维映射 $\phi(\cdot)$ 的计算公式。可以看到，这里我们是无法直接计算 $UU^T=\sum_{i=1}^{N}\boldsymbol{u}_i\boldsymbol{u}_i^T=\sum_{i=1}^{N}\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T$ 的，也就无法直接得到 $\lambda_j,\boldsymbol{w}_j$。我们首先明确，KPCA 的输出是什么？首先我们要得数据 Z，因为我们需要基于 Z 进行模型的训练；其次，给定一个新的样本点 $\boldsymbol{x}$，我们要知道如何将其映射到特征空间中，即如何计算 $\boldsymbol{z}$。可以看到，若能解决后一个问题，我们就能采用同样的方法得到训练数据 Z，因此，问题的关键是 $\boldsymbol{z}=W\phi(\boldsymbol{x})$ 的计算。
我们可以看到，$\boldsymbol{z}$ 中第 j ($j=1,2,\cdots,d'$) 维坐标分量为：

$$z^{(j)}=\boldsymbol{w}_j^T\phi(\boldsymbol{x})$$

显然我们需要对 $\boldsymbol{w}_j^T$ (或者说 $\boldsymbol{w}_j$) 进行恒等变形，使得变形后最右边出现 $\phi(\cdot)^T$ (对应地为最左边出现 $\phi(\cdot)$) 的形式，从而和上式最右端的 $\phi(\boldsymbol{x})$ 组成高维内积的形式，使计算能够进行。

- 我们手头只有关系式 $\left(\sum_{i=1}^{N}\boldsymbol{u}_i\boldsymbol{u}_i^T\right)\boldsymbol{w}_j=\lambda_j\boldsymbol{w}_j$，因而只能由它入手来进行恒等变形：

$$\left(\sum_{i=1}^{N}\boldsymbol{u}_i\boldsymbol{u}_i^T\right)\boldsymbol{w}_j=\left(\sum_{i=1}^{N}\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T\boldsymbol{w}_j\right)=\lambda_j\boldsymbol{w}_j$$

$$\Updownarrow$$

$$\boldsymbol{w}_j=\left(\sum_{i=1}^{N}\phi(\boldsymbol{x}_i)\frac{\phi(\boldsymbol{x}_i)^T\boldsymbol{w}_j}{\lambda_j}\right)$$

可以看到，上面 $\boldsymbol{w}_j$ 的表达式中最左端已经出现了 $\phi(\cdot)$ 的形式，为了便于推导，我们将除了 $\phi(\cdot)$ 之外的右端标量部分打包，记为 α_i^j，则 α_i^j 为第 i 个高维样本点与第 j 个特征向量 $\boldsymbol{w}_j$ 的内积除以特征值 λ_j：

$$\alpha_i^j = \frac{\boldsymbol{u}_i^T \boldsymbol{w}_j}{\lambda_j}$$

$$\boldsymbol{\alpha}^j = \langle \alpha_1^j, \alpha_2^j, \cdots, \alpha_N^j \rangle^T = \frac{U\boldsymbol{w}_j}{\lambda_j}$$

回代可得：

$$\boldsymbol{w}_j = \sum_{i=1}^{N} \left(\boldsymbol{u}_i \alpha_i^j \right) = U\boldsymbol{\alpha}^j$$

- 通过上面对 $\boldsymbol{w}_j$ 的恒等变形，我们可以看到此时 $z^{(j)}$ 就可以不依赖 $\phi(\cdot)$ 而根据 $\kappa(\cdot,\cdot)$ 进行计算了：

$$z^{(j)} = \boldsymbol{w}_j^T \phi(\boldsymbol{x}) = \sum_{i=1}^{N} \left(\alpha_i^j \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}) \right) = \sum_{i=1}^{N} \alpha_i^j \kappa(\boldsymbol{x}_i, \boldsymbol{x})$$

至此，问题转化为对 α_i^j 的计算。由 SVD 中的内容， $\boldsymbol{w}_j$ 为 UU^T 的单位正交特征向量，即 U 的右奇异向量，则 $\frac{U\boldsymbol{w}_j}{\sqrt{\lambda_j}} = \sqrt{\lambda_j}\boldsymbol{\alpha}^j \triangleq \boldsymbol{\beta}^j$ 为核矩阵 $U^TU = K$ 对应的单位正交特征向量，即 U 的左奇异向量。那么我们只需对 K 进行特征值分解，其第 j 大 (考虑重根) 的特征值 λ_j 对应的单位正交特征向量即为 $\boldsymbol{\beta}^j$ ，再由 $\boldsymbol{\alpha}^j = \boldsymbol{\beta}^j / \sqrt{\lambda_j}$ 即可求解。

KPCA 的步骤如下：

1. 由 X 计算核矩阵：K；
2. 对 K 进行特征值分解，取前 d' 个最大的特征值 λ_j (考虑重根) 和对应的单位正交特征向量 $\boldsymbol{\beta}^j$；
3. 由如下的计算公式计算训练数据 Z 和预测点的映射：

$$\boldsymbol{\alpha}^j = \boldsymbol{\beta}^j / \sqrt{\lambda_j} z^{(j)} = \sum_{i=1}^{N} \alpha_i^j \kappa(\boldsymbol{x}_i, \boldsymbol{x})$$

最后，我们需要说明的是，根据 PCA 的要求， $U = \phi(X)$ 需是中心化的数据，虽然我们无法计算 U ，但若 U 是中心化的数据，那么 $Z = W^TU$ 也是中心化的数据，因此个人认为我们可以通过检查 Z 是否中心化来判断 U 是否符合要求。

1.4 多维缩放 (Multiple Dimensional Scaling, MDS)

MDS 要求**原始空间中样本之间的欧式距离在低维空间中得以保持**，思路如下：

1. 由距离矩阵 D 求内积矩阵 B：

- 记距离矩阵 $D_{N\times N} = [dist_{ij}]_{N\times N}$ ，其中 $dist_{ij}$ 为样本点 $\boldsymbol{x}_i$ 到 $\boldsymbol{x}_j$ 的距离，我们要求 $dist_{ij} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\| = \|\boldsymbol{z}_i - \boldsymbol{z}_j\|$ 。
- 我们记降维后样本的内积矩阵为 $B_{N\times N} = Z^TZ = [b_{ij}]_{N\times N}$ ，其中 $b_{ij} = \boldsymbol{z}_i^T\boldsymbol{z}_j$ ，则可由内积矩阵表示两个样本点之间的距离：

$$\begin{aligned} dist_{ij}^2 &= \|\boldsymbol{z}_i - \boldsymbol{z}_j\|^2 = \|\boldsymbol{z}_i\|^2 + \|\boldsymbol{z}_j\|^2 - 2\boldsymbol{z}_i^T\boldsymbol{z}_j \\ &= b_{ii} + b_{jj} - 2b_{ij} \end{aligned}$$

- 不妨令降维后的样本 Z 为中心化的样本，即 $\sum_{i=1}^{N} \boldsymbol{z}_i = \boldsymbol{0}$ ，则 $\sum_{i=1}^{N} b_{ij} = \sum_{j=1}^{N} b_{ij} = \boldsymbol{0}$ 。内积矩阵 B 可由距离矩阵 D 表示如下：

$$b_{ij} = -\frac{1}{2}(dist_{ij}^2 - dist_{i\cdot}^2 - dist_{\cdot j}^2 + dist_{\cdot\cdot}^2)$$

$$dist_{i\cdot}^2 = \frac{1}{N}\sum_{j=1}^{N} dist_{ij}^2$$

$$dist_{\cdot j}^2 = \frac{1}{N}\sum_{i=1}^{N} dist_{ij}^2$$

$$dist_{\cdot\cdot}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} dist_{ij}^2$$

2. 对内积矩阵 B 进行特征值分解：$B = V_{N\times d}\Lambda_{d\times d}{V^T}_{d\times N}$，其中 $\Lambda = diag(\lambda_1, \lambda_2, \cdots, \lambda_d)$，$\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_d$，含有 d^* 个非零的特征值，由这 d^* 个非零的特征值 ${\Lambda^*}_{d^*\times d^*}$ 和对应的特征向量 ${V^*}_{N\times d^*}$ 可得降维后的数据：

$$Z = {\Lambda^*}^{1/2}{V^*}^T$$

现实应用中，为了有效降维，仅需降维后的距离与原始空间中的距离相近，不必严格相等，此时可取 $d'(\ll d^*)$ 个最大特征值 ${\Lambda^*}_{d^*\times d^*}$ 和相应的特征向量 ${V^*}_{N\times d^*}$，得到降维后的数据：

$$Z = {\Lambda'}^{1/2}{V'}^T$$

问题：

1. 为什么对内积矩阵进行分解后就可以得到降维数据 Z？

解答：内积矩阵与样本数据的关系如下：$B = Z^TZ$。而内积矩阵 B 为对称矩阵，能被分解为如下的形式：$B = V\Lambda^{1/2}\Lambda^{1/2}V^T = (\Lambda^{1/2}V^T)^T(\Lambda^{1/2}V^T)$，则可令 $Z = \Lambda^{1/2}V^T$。而通过舍弃较小的特征值而只保留最大的 d' 个特征值则可以在降维的同时，最大限度地保持原内积矩阵，从而使原始空间中样本之间的欧式距离在低维空间中得到最大限度的保持。

2. 降维数据 Z 与原始数据 X 的关系是什么？对于新的样本点，如何将其映射到低维空间中？

解答：降维后的数据 Z 与原始数据 X 存在一一对应的关系，即 $\boldsymbol{x}_i \sim \boldsymbol{z}_i$。当由最大的 d' 个特征值组成的矩阵 ${\Lambda'}^{1/2}_{d'\times d'}$ 中对角元素的排列顺序发生变化时，对应地 ${V'}_{N\times d'}$ 中每列的顺序也会发生变化，得到的样本数据 $Z_{d'\times N} = {\Lambda'}^{1/2}{V'}^T$ 显然也会发生变化。但可以看到，这样的变化只是在调换 $Z_{d'\times N}$ 中各行的顺序，也就是属性的顺序，各列之间也就是各样本点间的顺序是没有发生交换的，因此与原数据的对应关系也不会发生变化。

MDS 应该**不属于**线性降维，根据流形学习中 Isomap 的内容，个人推想，我们也是将高维空间坐标作为输入，低维空间坐标作为输出，训练回归学习器，以将新的数据点映射到低维空间中。

1.5 流形学习 (Manifold Learning)

> 参考文献：
>
> 1. 知乎-流形学习的基本思想
> 2. 浙江大学何晓飞-流形学习报告
> 3. pluskid大神博客-浅谈流形学习：将流形引入到机器学习领域来主要有两种用途：一是将原来在欧氏空间中适用的算法加以改造，使得它工作在流形上，直接或间接地对流形的结构和性质加以利用 (比如，graph regularized semi-supervised learning)；二是直接分析流形的结构，并试图将其映射到一个欧氏空间中，再在得到的结果上运用以前适用于欧氏空间的算法来进行学习 (本小节介绍的就是这个层面上的流形学习，比如 Isomap、LLE)。

首先我们大概地了解一下什么是流形：球面就是一个嵌入到三维空间中的二维流形，虽然球面上的任意一点都可以用三维坐标唯一地表示，但事实上，站在这个二维流形上看，我们使用经度和纬度两个坐标就可以唯一地表示球面上的任何一点。此外，直接使用欧式距离作为流形上两点间的距离显然是不合理的，我们会选择**测地线距离**也就是依附在流形上的两点间最短路径的长度作为流形上两点间的距离。但是，流形是在局部与欧式空间同胚的空间，也就是说它在局部具有欧式空间的性质，比如能用欧氏距离来进行距离计算。对于球面来说，我们可以将球面上很小的一块区域看做是一个平面，此时可用欧式距离近似地表示这块局部区域上两点间的距离。

流形学习就是假设数据存在一定的低维内在结构，即位于一个低维流形上，因此我们可以对数据进行降维。比如，瑞士卷也是一个嵌入到三维空间中的二维流形，我们对它进行降维，就相当于将其拉直铺平。在降维时，我们会试图保持样本数据的局部关系不变。因为是局部关系，所以可以采用欧式空间中的方法去描述，比如使用欧式距离去描述两点间的距离。这样，通过对局部的不断近似，我们就能在保持数据流形特征的同时完成对数据流形的全局降维，而选择保持不同的局部关系就会得到不同的流形学习方法。

流形学习欲有效进行领域保持则需样本密采样，而这恰是高维情形下面临的重大障碍，因此流形学习方法在实践中的降维性能往往没有预期的好。

1.5.1 等度量映射 (Isometric Mapping, Isomap)

Isomap 试图保持近邻样本点之间的距离，从而保持任意两点间的测地线距离，具体做法如下：

1. 对于每个样本点基于欧氏距离找出其近邻点，建立一个近邻连接图 (比如， k 近邻图或 ϵ 近邻图)，图中近邻点之间存在连接，而非近邻点之间不存在连接；
2. 使用 Dijkstra 算法或 Floyd 算法计算近邻连接图上任意两点之间的最短路径，从而得到任意两点间的测地线距离，即流形上任意两点间的距离；
3. 在得到任意两点的距离之后就可通过 MDS 来获得样本点在低维空间中的坐标。

对于新样本的映射问题，可将高维空间坐标作为输入，低维空间坐标作为输出，训练回归学习器来对新样本的低维空间坐标进行预测。

1.5.2 局部线性嵌入 (Locally Linear Embedding, LLE)

LLE 试图保持邻域内样本之间的线性关系，比如样本点 $\boldsymbol{x}_i$ 能通过它的邻域样本 $\boldsymbol{x}_j, \boldsymbol{x}_k, \boldsymbol{x}_l$ 的坐标通过线性组合而重构出来，即：

$$\boldsymbol{x}_i = \beta_{ij}\boldsymbol{x}_j + \beta_{ik}\boldsymbol{x}_k + \beta_{il}\boldsymbol{x}_l$$

那么，LLE 就希望上述关系在低维空间中得以保持，其做法如下：

1. 首先为每个样本点构建起近邻下标集合 Q_i ，然后计算出基于 Q_i 中的样本点对 $\boldsymbol{x}_i$ 进行重构的系数 $\boldsymbol{\beta}_i$ ：

$$\begin{aligned} &\min_{\boldsymbol{\beta}_1,\boldsymbol{\beta}_2,\cdots,\boldsymbol{\beta}_N} \sum_{i=1}^{N} \left\| \boldsymbol{x}_i - \sum_{j\in Q_i} \beta_{ij}\boldsymbol{x}_j \right\|_2^2 \\ &s.t. \quad \sum_{j\in Q_i} \beta_{ij} = 1,\ i = 1, 2, \cdots, N \end{aligned}$$

可解得：

$$\beta_{ij} = \frac{\sum_{k\in Q_i} C_{jk}^{-1}}{\sum_{l,s\in Q_i} C_{ls}^{-1}}$$

其中， $C_{jk} = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T(\boldsymbol{x}_i - \boldsymbol{x}_k)$ 。

低维空间中 $\boldsymbol{\beta}_i$ 不变，对于 $\boldsymbol{x}_i$ 对应的低维空间坐标 $\boldsymbol{z}_i$ 可通过下式求解：

$$\min_{\beta_1,\beta_2,\cdots,\beta_N}\sum_{i=1}^{N}\left\|\boldsymbol{z}_i-\sum_{j\in Q_i}\beta_{ij}\boldsymbol{z}_j\right\|_2^2$$

$$\Updownarrow$$

$$\min_{Z} tr(ZMZ^T)$$

$$s.t.\ \ ZZ^T=I$$

其中，$M_{N\times N}=(I_{N\times N}-\mathrm{B}_{N\times N})^T(I-\mathrm{B})$，$\mathrm{B}=[\mathrm{B}_{ij}]_{N\times N}$ 为权重矩阵，$\mathrm{B}_{ij}=\beta_{ij}$，当 $j\notin Q_i$ 时，$\beta_{ij}=0$。我们添加约束条件 $ZZ^T=I$ 是为了得到标准化 (标准正交空间) 的低维数据。上式可通过特征值分解求解：M **最小的** d' **个特征值对应的特征向量组成的矩阵即为** ${Z^T}_{N\times d'}$。当然，对于新样本点的映射，个人猜想和 Isomap 的做法一样。

1.6 度量学习 (Metric Learning)

事实上，每个空间对应了在样本属性上定义的一个距离度量，而寻找合适的空间，实际上就是在寻找一个合适的距离度量。度量学习就是尝试去直接学习出一个合适的距离度量。

我们对欧式距离的平方作推广，得到马氏距离 (Mahalanobis distance)：

$$dist_{mah}^2(\boldsymbol{x}_i,\boldsymbol{x}_j)=(\boldsymbol{x}_i-\boldsymbol{x}_j)^TM(\boldsymbol{x}_i-\boldsymbol{x}_j)=\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_M^2$$

其中，M 为度量矩阵，是我们学习的目标。为了保证距离非负且对称，M 需为为半正定对称矩阵，由此存在正交基 P 使得 $M=PP^T$。$W=I_{d\times d}$ 时马氏距离即为欧式距离。

度量矩阵的学习可以嵌入到学习器的学习中，西瓜书上以近邻成分分析 (Neighbourhood Component Analysis, NCA) 为例进行了说明，这里不再赘述。若求得的 M 是一个低秩矩阵，则通过对 M 进行特征值分解，总能找到一组正交基，其正交基数目为矩阵 M 的秩 $rank(M)$，小于原属性数 d。于是，度量学习学得的结果可衍生出一个降维矩阵 $P_{d\times rank(M)}$，能用于降维之目的。

2 特征选择

> 1. 博客园-特征工程之特征选择

2.1 特征选择的一般思路

- 特征选择的目标是从初始的特征集合中选取一个包含了所有重要信息的特征子集。因为存在组合爆炸的问题，我们一般不会去遍历所有可能的子集，而是根据某种策略对子集进行生成和搜索，并结合相应的子集评价指标对子集进行判断。也就是说**为了进行特征选择，我们需要解决如下两个问题：(1) 子集生成与搜索策略；(2) 子集评价指标。**可以看到，特征选择的过程与 BSPCE 的构建算法很像，事实上，我们可以认为基于 KIC 进行模型选择就是在进行特征选择，BSPCE 中的多项式基函数就对应特征选择中的特征。**BSPCE 使用** L_2 **正则化进行嵌入式特征选择，并基于 KIC 进行包裹式特征选择。**
- **子集生成与搜索策略一般都是贪心的，主要有如下几种：(1) 前向 (forward) 搜索；(2) 后向 (backward) 搜索；(3) 双向 (bidirectional) 搜索。**前向搜索每一轮识别出一个相关特征，后一轮在前一轮的基础上进行，直到添加剩余的任何一个特征都无法使特征子集的性能更优为止；后向搜索则与之相反，它每一轮从特征集合中剔除一个无关特征，直至剔除剩余的任何一个特征都无法使特征子集的性能更优为止；双向搜索则是前向搜索和后向搜索的结合，每一轮增加相关特征的同时减少无关特征，识别为相关特征的特征在后续轮中将不会被去除，而识别为无关特征的特征在后续轮中则不会再出现。
- 接下来，我们来讨论子集评价的问题。我们将数据集记为 D，某个属性子集 $A=\{a_1,a_2,\cdots,a_n\}$，属性 a_i 可能的取值个数为 v_i，即 a_i 的可能取值为 $\{a_i^1,a_i^2,\cdots,a_i^{v_i}\}$，那么特征子集 A 的可能取值个数为 $\prod_{i=1}^n v_i$。样本标记为 y，取值空间为 $\mathcal{Y}=\{y_1,y_2,\cdots,y_m\}$，即样本标记的可能取值有 $m=|\mathcal{Y}|$ 个。

特征子集 A 可视为输入向量，含有 n 个分量，而标记 y 可视为输出。我们可以分别基于输入 A 的取值和输出 y 的取值对样本数据 D 进行划分。可以看到，基于 y 的划分是我们所想要的划分，因为在这个划分中，相同标记的样本都被划分到一起。**因此，基于 A 的划分越接近基于 y 的划分就说明特征子集 A 越优，我们可以通过估计这两个划分的差异来对特征子集 A 进行评价。可见，许多"多样性度量"如不合度量、相关系数等，稍加调整即可用于特征子集评价。**

一种子集评价指标是信息增益，特征集 A 的信息增益如下：

$$Gain(D, A) = H_D(y) - H_D(y|A)$$

即在数据集 D 中，标记 y 的信息熵减去给定属性集 A 后 y 的条件熵，反映了给定属性集 A 后标记 y 取值不确定性下降的程度，或者说纯度上升的程度，因此信息增益越大说明特征集越好。若此时子集生成与搜索策略为前向搜索，则特征选择的过程与 ID3 决策树学习算法非常相似。事实上，决策树可用于特征选择，树结点的划分属性所组成的集合就是选择出的特征子集。**其他的特征选择方法未必像决策树特征选择这么明显，但它们本质上都是显示或隐式地结合了某种 (或多种) 子集搜索机制和子集评价机制。**

- 在子集的生成与搜索方面，可引入很多人工智能搜索技术，如分支限界法、浮动搜索法等；在子集评价方面，除了上述提到的信息熵，还可以使用 AIC、KIC 等指标。

2.2 特征选择的常用方法

常见的特征选择方法大致可分为三类：(1) 过滤式 (filter)，比如针对二分类问题的 Relief (Relevant features) 及其针对多分类问题的扩展变体 Relief-F；(2) 包裹式 (wrapper)，比如 LVW (Las Vegas wrapper)；(3) 嵌入式 (embedding)，比如 L_1, L_2 正则化。**过滤式方法先对数据集进行特性选择，然后再训练学习器，特征选择过程与后续学习器无关**，相当于先用特征选择过程对初始特征进行过滤，再用过滤后的特征来训练学习器。与过滤式方法不考虑后续学习器不同，**包裹式特征选择直接把最终将要使用的学习器的性能作为特征子集的评价标准**，因此，从最终学习器性能来看，包裹式特征选择比过滤式特征选择更好，但计算开销通常却大得多。过滤式和包裹式方法中，特征选择过程与学习器训练过程有明显的分别，而**嵌入式特征选择将特征选择过程与学习器训练过程融为一体，两者在同一个优化过程中完成**，即在学习器训练过程中自动进行了特征选择。

1. Relief 针对二分类问题，使用"相关统计量"度量特征的重要性，并通过选择相关统计量最大的 k 个特征或相关统计量大于某个阈值的那些特征完成特征的选择。第 j, $(j = 1, \cdots, n)$ 个属性的相关统计量的计算公式如下：

$$\delta^j = \sum_{i=1}^{N} -\mathrm{diff}(\boldsymbol{x}_i^j, \boldsymbol{x}_{i,nh}^j)^2 + \mathrm{diff}(\boldsymbol{x}_i^j, \boldsymbol{x}_{i,nm}^j)^2$$

其中， N 为样本容量； $\boldsymbol{x}_i^j$ 为第 i 个样本点在第 j 个属性上的取值； $\boldsymbol{x}_{i,nh}$ 表示与 $\boldsymbol{x}_i$ 标记相同的最近样本点，被称为"猜中近邻" (near-hit)，而 $\boldsymbol{x}_{i,nm}$ 表示与 $\boldsymbol{x}_i$ 标记不同的最近样本点，被称为"猜错近邻" (near-miss)； $\mathrm{diff}(\cdot, \star)$ 衡量了两个属性值 $\cdot, \star$ 之间的差异：

- 对于离散属性，若 $\cdot = \star$，则 $\mathrm{diff}(\cdot, \star) = 0$；若 $\cdot \neq \star$，则 $\mathrm{diff}(\cdot, \star) = 1$；
- 对于连续属性，我们首先将属性取值范围规范化到区间 $[0, 1]$，则 $\mathrm{diff}(\cdot, \star) = |\cdot - \star|$

从相关统计量的计算公式中可以看到，若 $\boldsymbol{x}_i$ 与其猜中近邻 $\boldsymbol{x}_{i,nh}$ 在第 j 个属性上的距离小于 $\boldsymbol{x}_i$ 与其猜错近邻 $\boldsymbol{x}_{i,nm}$ 在第 j 个属性上的距离，则说明第 j 个属性有助于区分同类与异类样本，对应地其相关统计量的值就大。

一般地，我们不会基于整个样本集 D 计算属性的相关统计量，而是先对样本集 D 进行采样，得到一些样本子集 $D_1, D_2, \cdots$，再分别基于这些样本子集计算属性的相关统计量，最后取平均，得到各属性最终的相关统计量。

Relief-F 是 Relief 在多分类问题上的推广，相关统计量的计算公式如下

$$\delta^j=\sum_{i=1}^N\left(-\mathrm{diff}(\boldsymbol{x}_i^j,\boldsymbol{x}_{i,nh}^j)^2+\sum_{l\neq k}p_l\times\mathrm{diff}(\boldsymbol{x}_i^j,\boldsymbol{x}_{i,l,nm}^j)^2\right)$$

其中，样本点 $\boldsymbol{x}_i$ 的标记为 k； p_l 为 $l\ (l\neq k)$ 类样本在数据集中所占比例 (个人觉得西瓜书上这里有误，应该是第 $l\ (l\neq k)$ 类样本占所有非 k 类样本的比例，这样才能与 Relief 统一)， $\boldsymbol{x}_{i,l,nm}^j$ 为标记为 l 的样本中与 $\boldsymbol{x}_i$ 最近的样本点。

可以看到，Relief 类方法的缺点是并没有考虑属性之间的相关作用。

2. LVW 在拉斯维加斯方法 (Las Vegas method) 框架下使用随机策略生成特征子集，然后基于得到的特征子集训练学习器，并使用交叉验证计算泛化误差，我们选择泛化误差较小的特征子集，或者在泛化误差相等的情况下选择属性较少的特征子集。

> 拉斯维加斯方法和蒙特卡罗方法是两个以著名赌城名字命名的随机化方法，两者的主要区别是：若有时间限制，则拉斯维加斯方法或者给出满足要求的解，或者不给出解，而蒙特卡罗方法一定会给出解，虽然给出的解未必满足要求；若无时间限制，则两者都能给出满足要求的解。

3. L_1,L_2 正则化不再赘述。我们指出，**对于基于 L_1 范数的优化问题，可使用近端梯度下降 (proximal gradient descent, PGD) 快速求解。**基于 LASSO (L_1 正则化)，还有考虑特征分组结构的 Group LASSO (感觉类似于 BSPCE 中将相互依赖的自变量打包成一个 Group 的操作)，考虑特征序结构的 Fused LASSO 等变体。由于凸性不严格，LASSO 类方法可能产生多个解，该问题可通过弹性网 (elastic net) 予以解决。

最小角回归 (Least angle regression, LARS) 也是一种嵌入式特征选择方法，而 LASSO 可通过对 LARS 稍加修改而实现。

3 稀疏表示

若样本矩阵 $X_{d\times N}=(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_N)$ 是稀疏的，即存在很多的 0 (当然不是整行整列地为 0)，机器学习的难度会大大地降低。比如，文本数据在使用字频表示后具有高度的稀疏性，从而变得线性可分，使用线性支持向量机就能很好地处理。此外，稀疏矩阵已有很多高效的存储方法，因此并不会造成存储上的巨大负担。

自然地，对于稠密样本，我们也希望将其稀疏化，得到其稀疏表示后再基于稀疏样本进行机器学习。我们可以通过求解如下的优化问题实现这一过程：

$$\min_{W,\boldsymbol{z}_i}\sum_{i=1}^N\|\boldsymbol{x}_i-W^T\boldsymbol{z}_i\|_2^2+\lambda\sum_{i=1}^N\|\boldsymbol{z}_i\|_1$$

其中， $W_{d\times d'}=(\boldsymbol{w}_1,\cdots,\boldsymbol{w}_{d'})$， W^T 被称为字典矩阵或码书矩阵， d' 为字典的词汇量， $\boldsymbol{z}_i$ 为样本的稀疏表示。**可以看到，上述优化问题和 PCA 一样也是一个最小二乘问题，只不过多了一个正则化项，即我们试图寻找一组基 $W_{d\times d'}=(\boldsymbol{w}_1,\cdots,\boldsymbol{w}_{d'})$ 和相应的坐标 $\boldsymbol{z}_i$，在很好地近似或者说重构原始样本 $\boldsymbol{x}_i$ 的同时，坐标 $\boldsymbol{z}_i$ 应尽可能地稀疏。**由此，我们就得到了原样本 $\boldsymbol{x}_i$ 在这组基上的稀疏表示 $\boldsymbol{z}_i$，而这些基 $\boldsymbol{w}_i$ 也被称为稀疏基。**但和 PCA 不同的是，PCA 旨在降维，因此 $d'<d$，而这里的目标是将原始的稠密样本映射为稀疏样本，因此通常 $d'\geqslant d$，而我们也常通过设置词汇量 d' 的大小来控制字典的规模，从而影响稀疏的程度。**可以看到，**稀疏表示的思路类似于使用核方法将样本映射到高维空间，从而降低学习的难度，虽然稀疏表示的维度相比原样本不一定会升高，但显然维度越高 (即 d' 越大) 表示会越稀疏。**除了通过控制码书规模影响稀疏性，有时还希望控制码书的"结构"，例如假设码书具有"分组结构"，即同一个分组内的变量或同为非零，或同为零 (以汉语文档为例，一个概念可能由多个字词来表达，这些字词就构成了一个分组；若这个概念在文档中没有出现，则这整个分组所对应的变量都将为零)。这样的性质称为"分组稀疏性" (group sparsity)，相应的稀疏编码方法称为分组稀疏编码 (group sparse coding)。

> 我们作一个关于概念的说明：得到字典或码书 W^T 的过程被称为字典学习 (dictionary learning) 或码书学习 (codebook learning)；而得到样本稀疏表示 $\boldsymbol{z}_i$ 的过程被称为稀疏编码 (sparse coding)；可以看到，二者通常在同一个优化求解过程中完成，因此常不加区分而笼统地称为码书学习。

接下来就是如何求解上述优化问题了。存在两个优化变量，我们采用**类似于**坐标上升的方法进行求解：

1. 固定码书 W^T ，优化稀疏表示 $Z=(\boldsymbol{z}_1,\cdots,\boldsymbol{z}_N)$ ，此时：

$$\min_{\boldsymbol{z}_i}\sum_{i=1}^{N}\|\boldsymbol{x}_i-W^T\boldsymbol{z}_i\|_2^2+\lambda\sum_{i=1}^{N}\|\boldsymbol{z}_i\|_1$$

$$\Updownarrow$$

$$\min_{\boldsymbol{z}_i}\|\boldsymbol{x}_i-W^T\boldsymbol{z}_i\|_2^2+\lambda\|\boldsymbol{z}_i\|_1,\ \ i=1,\cdots,N$$

此时 $\boldsymbol{z}_i$ 存在解析解。

2. 以 Z 为初值，优化 W 和 Z (不是单纯地固定 Z 优化 W ，因此我们前面强调是“类似于坐标上升的方法”)：

$$\text{以上一步求得的}Z\text{为初值优化}W\text{，则问题即为：}$$

$$\min_{W}\sum_{i=1}^{N}\|\boldsymbol{x}_i-W^T\boldsymbol{z}_i\|_2^2+\lambda\sum_{i=1}^{N}\|\boldsymbol{z}_i\|_1=\min_{W}\|X-W^TZ\|_F^2$$

$$\text{但我们使用基于逐列更新策略的 KSVD 求解上述问题时对}Z\text{也进行了更新，}$$

$$\text{记}W\text{，}Z\text{的行向量分别为}\hat{\boldsymbol{w}}_i\text{，}\hat{\boldsymbol{z}}_i\ (i=1,\cdots,d')\text{，则：}$$

$$\|X-W^TZ\|_F^2=\left\|X-\sum_{j=1}^{d'}\hat{\boldsymbol{w}}_j^T\hat{\boldsymbol{z}}_j\right\|_F^2$$

$$=\left\|\left(X-\sum_{j\neq i}\hat{\boldsymbol{w}}_j^T\hat{\boldsymbol{z}}_j\right)-\hat{\boldsymbol{w}}_i^T\hat{\boldsymbol{z}}_i\right\|_F^2=\left\|E_i-\hat{\boldsymbol{w}}_i^T\hat{\boldsymbol{z}}_i\right\|_F^2$$

我们每次在更新码书 W^T 的第 i 列 $\hat{\boldsymbol{w}}_i^T$ 的同时更新稀疏表示 Z 的第 i 行 $\hat{\boldsymbol{z}}_i$ ，而 W^T,Z 的其他部分保持不变，即 E_i 保持不变，则问题即为：

$$\min_{\hat{\boldsymbol{w}}_i^T,\hat{\boldsymbol{z}}_i}\left\|E_i-\hat{\boldsymbol{w}}_i^T\hat{\boldsymbol{z}}_i\right\|_F^2$$

我们基于奇异值分解求解上述问题，但直接对 E_i 进行奇异值分解，再令 $\hat{\boldsymbol{w}}_i,\hat{\boldsymbol{z}}_i$ 为最大奇异值对应的正交向量可能会破坏 Z 的稀疏性。因此，我们仅保留 $\hat{\boldsymbol{z}}_i$ 的非零元素， E_i 则仅保留 $\hat{\boldsymbol{w}}_i$ 与 $\hat{\boldsymbol{z}}_i$ 非零元素的乘积项，然后进行奇异值分解。

4 附录

4.1 协方差矩阵

1. 方差：用来衡量单个随机变量的离散程度。

- 总体/随机变量 X ，总体均值 $\mu=\mathbb{E}X$ ，**总体方差** $\sigma^2=\mathbb{V}[X]=\mathbb{E}[(X-\mathbb{E}X)^2]=\mathbb{E}[X^2]-(\mathbb{E}X)^2=\int(X-\mu)^2p(X)dX$ 。
- 样本 $(x_1,x_2,\cdots,x_N)$ ，样本均值 $\overline{x}=\sum_{i=1}^{N}x_i$ ，**样本方差** $S^2=\dfrac{1}{N-1}\sum_{i=1}^{N}(x_i-\overline{x})^2$ 。

样本方差公式中除以 $N-1$ 而不是 N 的原因：事实上，样本方差应该为：$\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2$，但关键是我们不知道总体均值 μ 为多少，因此用样本均值 $\overline{x}$ 替代，得到近似值 $\frac{1}{N}\sum_{i=1}^{N}(x_i-\overline{x})^2$，替代之后我们需要对此进行修正，即乘以 $\frac{N}{N-1}$，从而得到样本方差的计算公式 $S^2=\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\overline{x})^2$，这样样本方差才是总体方差的无偏估计，即 $\mathbb{E}(S^2)=\sigma^2$。进一步地，对于函数 $f(t)=\frac{1}{N}\sum_{i=1}^{N}(x_i-t)^2$，当 $t=\overline{x}$ 时函数取得最小值，因此 $\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2\geqslant\frac{1}{N}\sum_{i=1}^{N}(x_i-\overline{x})^2$，所以我们修正时称了一个比 1 大的数 $\frac{N}{N-1}$。事实上，我们有：

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}(x_i-\overline{x})^2\right]=\sigma^2-\mathbb{E}\left[(\overline{x}-\mu)^2\right]=\sigma^2-\frac{1}{N}\sigma^2$$

2. 协方差：两个随机变量的协方差除以它们的标准差之积即得相关系数，相关系数刻画了这两个随机变量的线性相关程度。

随机变量 $X^{(1)}$ 和 $X^{(2)}$，二者的协方差：
$\mathbb{COV}(X^{(1)},X^{(2)})=\mathbb{E}[(X^{(1)}-\mathbb{E}X^{(1)})(X^{(2)}-\mathbb{E}X^{(2)})]=\mathbb{E}[X^{(1)}X^{(2)}]-\mathbb{E}X^{(1)}\mathbb{E}X^{(2)}$，当 $\mathbb{COV}(X^{(1)},X^{(2)})=0$ 时表明 $X^{(1)},X^{(2)}$ (线性) 不相关，$\mathbb{COV}(X^{(1)},X^{(1)})=\mathbb{V}[X^{(1)}]$。

样本的协方差：$\frac{1}{N-1}\sum_{k=1}^{N}(x_k^{(1)}-\overline{x^{(1)}})(x_k^{(2)}-\overline{x^{(2)}})$。

可以看到，**随机变量 $X^{(1)}$ 和 $X^{(2)}$ 的采样应该具有对应关系，若样本点间不存在对应关系，那么协方差的定义也就没有意义了。**比如 $(x_1^{(1)},x_2^{(1)})=(1,3)$，$(x_1^{(2)},x_2^{(2)})=(1,4)$，对应的 $\mathbb{COV}(X^{(1)},X^{(2)})=4$，而若改变 $X^{(1)}$ 的样本顺序 $(x_1^{(1)},x_2^{(1)})=(3,1)$，对应的 $\mathbb{COV}(X^{(1)},X^{(2)})=-4$。**事实上，随机变量 $X^{(1)}$ 和 $X^{(2)}$ 可以看做是样本点的两个分量。**

3. 协方差矩阵：

n 个随机变量 $X^{(1)},X^{(2)},\cdots,X^{(n)}$，协方差矩阵为：$[\mathbb{COV}_{ij}]_{n\times n}$，其中 $\mathbb{COV}_{ij}=\mathbb{COV}(X^{(i)},X^{(j)})=\mathbb{E}[(X^{(i)}-\mathbb{E}X^{(i)})(X^{(j)}-\mathbb{E}X^{(j)})]$。**我们可以将这 n 个随机变量打包成一个样本点，也就是随机向量 $\boldsymbol{x}=\langle X^{(1)},X^{(2)},\cdots,X^{(n)}\rangle^T$，$[\mathbb{COV}_{ij}]_{n\times n}$ 也就是随机向量各维度间的协方差矩阵 (注意，我们所说的协方差是不同维度之间的协方差，不存在不同样本点之间的协方差一说)。**进一步地，我们推得**随机向量协方差矩阵**的向量表示形式：

$$[\mathbb{COV}_{ij}]_{n\times n}=[\mathbb{E}[(X^{(i)}-\mathbb{E}X^{(i)})(X^{(j)}-\mathbb{E}X^{(j)})]]_{n\times n}$$
$$=\mathbb{E}[(X^{(i)}-\mathbb{E}X^{(i)})(X^{(j)}-\mathbb{E}X^{(j)})]_{n\times n}$$

其中，i指定了行坐标，j指定了列坐标，可得：

$$[(X^{(i)}-\mathbb{E}X^{(i)})(X^{(j)}-\mathbb{E}X^{(j)})]_{n\times n}=(\boldsymbol{x}-\mathbb{E}\boldsymbol{x})(\boldsymbol{x}^T-\mathbb{E}\boldsymbol{x}^T)$$

则随机向量的协方差矩阵：

$$\mathbb{COV}=\mathbb{E}[(\boldsymbol{x}-\mathbb{E}\boldsymbol{x})(\boldsymbol{x}^T-\mathbb{E}\boldsymbol{x}^T)]$$

对于**样本协方差矩阵**，$\mathbb{COV}_{ij}=\frac{1}{N-1}\sum_{k=1}^{N}(x_k^{(i)}-\overline{x^{(i)}})(x_k^{(j)}-\overline{x^{(j)}})$，其中，第 k 个样本点 $\boldsymbol{x}_k=\langle x_k^{(1)},x_k^{(2)},\cdots,x_k^{(n)}\rangle^T$，我们有：

$$
\begin{aligned}
&[\mathbb{COV}_{ij}]_{n\times n} = \\
&\left[\frac{1}{N-1}\sum_{k=1}^{N}(x_k^{(i)}-\overline{x^{(i)}})(x_k^{(j)}-\overline{x^{(j)}})\right]_{n\times n} \\
&=\frac{1}{N-1}\sum_{k=1}^{N}\left[(x_k^{(i)}-\overline{x^{(i)}})(x_k^{(j)}-\overline{x^{(j)}})\right]_{n\times n} \\
&=\frac{1}{N-1}\sum_{k=1}^{N}\left[(x_k^{(i)}-\overline{x^{(i)}})(x_k^{(j)}-\overline{x^{(j)}})\right]_{n\times n}
\end{aligned}
$$

其中，i指定了行坐标，j指定了列坐标，可得：

$$
\left[(x_k^{(i)}-\overline{x^{(i)}})(x_k^{(j)}-\overline{x^{(j)}})\right]_{n\times n} = (\boldsymbol{x}_k-\overline{\boldsymbol{x}})(x_k^T-\overline{\boldsymbol{x}}^T)
$$

则样本的协方差矩阵：

$$
\mathbb{COV} = \frac{1}{N-1}\sum_{k=1}^{N}(\boldsymbol{x}_k-\overline{\boldsymbol{x}})(x_k^T-\overline{\boldsymbol{x}}^T)
$$

4.2 瑞利商 (Rayleigh quotient) 与广义瑞利商 (generalized Rayleigh quotient)

- **瑞利商**是指这样的函数：

$$
R(\boldsymbol{x};A) = \frac{\boldsymbol{x}^H A\boldsymbol{x}}{\boldsymbol{x}^H\boldsymbol{x}}
$$

其中，$\boldsymbol{x}_{n\times 1}$ 为非零向量，$A_{n\times x}$ 为 Hermitan 矩阵，即自共轭矩阵，满足 $A^H = A$ (若 A 为实矩阵，则 $A^T = A$，Hermitan 矩阵即为对称矩阵)。**瑞利商的最大值等于矩阵 A 的最大特征值，最小值等于矩阵 A 的最小特征值**：

$$
\lambda_{\min} \leqslant R(\boldsymbol{x};A) \leqslant \lambda_{\max}
$$

证明：不妨令 $\boldsymbol{x}^H\boldsymbol{x} = 1$，问题转化为：

$$
\begin{aligned}
&\min_{\boldsymbol{x}} R(\boldsymbol{x};A) = \boldsymbol{x}^H A\boldsymbol{x} \\
&s.t.\ \ \boldsymbol{x}^H\boldsymbol{x} = 1
\end{aligned}
$$

由拉格朗日乘数法可得：

$$
\begin{aligned}
A\boldsymbol{x} &= \lambda\boldsymbol{x} \\
\boldsymbol{x}^H\boldsymbol{x} &= 1
\end{aligned}
$$

即为特征值和特征向量的定义，其中，$\boldsymbol{x}$ 为单位特征向量 (对于原问题，$\boldsymbol{x}$ 为特征向量即可，不需单位化)。此时，$R(\boldsymbol{x};A) = \boldsymbol{x}^H A\boldsymbol{x} = \boldsymbol{x}^H\lambda\boldsymbol{x} = \lambda$，取最小特征值即可，求最大值的情况类似。

当 $\boldsymbol{x}$ 为标准基向量即 $\boldsymbol{x}^H\boldsymbol{x} = 1$ 时，瑞利商退化为 $R(\boldsymbol{x};A) = \boldsymbol{x}^H A\boldsymbol{x}$ 。

- **广义瑞利商**是指这样的函数：

$$
R(\boldsymbol{x};A,B) = \frac{\boldsymbol{x}^H A\boldsymbol{x}}{\boldsymbol{x}^H B\boldsymbol{x}}
$$

其中，$\boldsymbol{x}_{n\times 1}$ 为非零向量，$A_{n\times x}, B_{n\times x}$ 均为 Hermitan 矩阵，且 B 正定。

对于广义瑞利商的最值问题，有三种解法：

1. 和上面类似，**将问题转化为约束优化问题，使用拉格朗日乘数法求解。**不妨令 $\boldsymbol{x}^H B\boldsymbol{x} = 1$，问题可转化为：

$$\min_{\boldsymbol{x}} \boldsymbol{x}^H A\boldsymbol{x}$$
$$s.t.\ \ \boldsymbol{x}^H B\boldsymbol{x} = 1$$

可得：

$$A\boldsymbol{x} = \lambda B\boldsymbol{x} \Leftrightarrow B^{-1}A\boldsymbol{x} = \lambda\boldsymbol{x}$$
$$\boldsymbol{x}^H B\boldsymbol{x} = 1$$

即为求矩阵 $B^{-1}A$ 的特征值和特征向量的问题，又 $\boldsymbol{x}^H A\boldsymbol{x} = \boldsymbol{x}^H \lambda B\boldsymbol{x} = \lambda$，取最小的特征值即可，$\boldsymbol{x}$ 为对应的特征向量，其各元素间成比例变化，对于约束优化问题取刚好使 $\boldsymbol{x}^H B\boldsymbol{x} = 1$ 成立的情况即可，但对于原问题，$\boldsymbol{x}$ 为 $B^{-1}A$ 的特征向量即可，没有 $\boldsymbol{x}^H B\boldsymbol{x} = 1$ 的限制。

2. 我们进行**变量代换** $\boldsymbol{x}' = B^{1/2}\boldsymbol{x}$，就可以将广义瑞利商 $R(\boldsymbol{x}; A, B)$ 转化为瑞利商 $R(\boldsymbol{x}'; B^{-1/2}AB^{-1/2})$，则广义瑞利商的最值问题转化为矩阵 $B^{-1/2}AB^{-1/2}$ 的特征值问题。又 $B^{-1/2}AB^{-1/2}$ 和 $B^{-1}A$ 这两个 n 阶矩阵相似，即 $B^{-1}A = B^{-1/2}(B^{-1/2}AB^{-1/2})B^{1/2}$，因此二者的特征值相同，问题最终转化为矩阵 $B^{-1}A$ 的特征值问题。
3. 求导法。$R(\boldsymbol{x}; A, B) = \boldsymbol{x}^T A\boldsymbol{x}(\boldsymbol{x}^T B\boldsymbol{x})^{-1}$，直接对 $\boldsymbol{x}$ 求导：

$$\frac{\partial R}{\partial \boldsymbol{x}} = 2A\boldsymbol{x}(\boldsymbol{x}^T B\boldsymbol{x})^{-1} - 2\boldsymbol{x}^T A\boldsymbol{x}(\boldsymbol{x}^T B\boldsymbol{x})^{-2}B\boldsymbol{x} = \boldsymbol{0}$$
$$\Rightarrow A\boldsymbol{x}(\boldsymbol{x}^T B\boldsymbol{x})_{1\times 1} = (\boldsymbol{x}^T A\boldsymbol{x})_{1\times 1}B\boldsymbol{x}$$
$$\Rightarrow \boldsymbol{x} \propto B^{-1}A\boldsymbol{x}$$

显然，瑞利商作为广义瑞利商的特殊情况 (此时，B 取单位矩阵 I) 也可以采用上述求导法求解。

4.3 样本重构的原理

左乘表示对作用的矩阵进行行变换 (此处就是对各属性进行线性组合)，各列之间不会发生交叉作用 (此处就是各样本点间相互独立，彼此之间不会产生影响)。

在 PCA 中，我们使用变换矩阵 $W_{d\times d'}$ 对 d 维样本点 $\boldsymbol{x}_i$ 进行降维得到 d' 维样本点 $\boldsymbol{z}_i = W^T\boldsymbol{x}_i$，再对其进行重构也就是升维，得到 $\hat{\boldsymbol{x}}_i = W\boldsymbol{z}_i$，那么这样进行重构的原理是什么呢？**我们首先可以从矩阵乘法的角度去解读上述重构过程，窥见一斑。**我们知道，矩阵乘法对应着一个线性变换，即将空间中的一个点线性映射为空间中的另一个点，而连续乘法则可看做是连续或者说复合线性变换。因为变换矩阵 W 中的列向量为相互正交的单位向量，当 $d' = d$ 时，W 就为正交矩阵。此时，样本重构只需乘以正交矩阵的转置即可：$\hat{\boldsymbol{x}}_i = W_{d\times d}{W^T}_{d\times d}\boldsymbol{x}_i = I_{d\times d}\boldsymbol{x}_i = \boldsymbol{x}_i$，因为 W 为正交矩阵时，连续两个线性映射 WW^T 复合起来就是 I，即样本依次经过两个线性映射后得到的是自己本身。但 PCA 中 $d' < d$，我们可以将此时的 W 看做是正交矩阵丢失了某些基向量的情况，类比正交矩阵，我们仍使用公式 $\hat{\boldsymbol{x}}_i = W_{d\times d'}{W^T}_{d'\times d}\boldsymbol{x}_i$ 进行重构，但此时 $W_{d\times d'}{W^T}_{d'\times d} \neq I_{d\times d}$，样本信息会存在损失。**接下来，我们来讨论重构的严格证明。**

显然，重构原理的严格证明实际上是一个优化问题：降维：$\boldsymbol{z}_i = W^T\boldsymbol{x}_i$，$W$ 的列向量为单位正交向量；重构：$\hat{\boldsymbol{x}}_i = P_{d\times d'}\boldsymbol{z}_i$，与 W 一样，我们**令 P 的列向量为单位正交向量**，即 $P^TP = I_{d'\times d'}$ (注意不是行向量，$d' < d$，不存在 d 个相互正交的 d' 维单位行向量)；那么**重构数据 $\hat{\boldsymbol{x}}_i$ 与原始数据 $\boldsymbol{x}_i$ 应尽可能靠近，即我们选取的 P 要最小化二者间的差距，原始样本与重构样本的差距可用欧氏距离的平方来衡量**：

$$\|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\|_2^2 = \boldsymbol{x}_i^T\boldsymbol{x}_i - 2\boldsymbol{x}_i\hat{\boldsymbol{x}}_i + \hat{\boldsymbol{x}}_i^T\hat{\boldsymbol{x}}_i$$
$$= \boldsymbol{x}_i^T\boldsymbol{x}_i - 2\boldsymbol{x}_iP\boldsymbol{z}_i + \boldsymbol{z}_i^TP^TP\boldsymbol{z}_i$$
$$= \boldsymbol{x}_i^T\boldsymbol{x}_i - 2\boldsymbol{x}_i^TP\boldsymbol{z}_i + \boldsymbol{z}_i^T\boldsymbol{z}_i$$

其中，$\boldsymbol{x}_i$ 为样本数据。根据上面的表述，**我们会很自然地将问题设定为：(1) 给定 W 后，寻找使上式最小的 P，即**

$$\arg\max_P \boldsymbol{x}_i^T P\boldsymbol{z}_i = \arg\max_P \boldsymbol{x}_i^T PW^T\boldsymbol{x}_i =$$

$$\arg\max_P (P^T\boldsymbol{x}_i)^T(W^T\boldsymbol{x}_i) = \arg\max_P \langle P^T\boldsymbol{x}_i, W^T\boldsymbol{x}_i\rangle$$

$$s.t.\ \ P^TP = I_{d'\times d'}$$

即要使**变化的向量** $P^T\boldsymbol{x}_i$ **与固定的向量** $W^T\boldsymbol{x}_i$ 的内积最大。若 $P^T\boldsymbol{x}_i$ 的长度固定，不随 P 的变化而变化，那么显然两个向量共线且同向时内积最大，此时 $P=W$，又 W 的列向量也单位正交，因而 P 的约束条件此时也满足。但很可惜，$PP^T \neq I_{d\times d}$，那么 $\langle P^T\boldsymbol{x}_i, P^T\boldsymbol{x}_i\rangle = \boldsymbol{x}_i^T PP^T\boldsymbol{x}_i$ 并不为常数，所以我们不能采用上述思路求解，只能老老实实地使用拉格朗日乘数法。**显然，对于上述约束优化问题，** P **的选取和** W **与** $\boldsymbol{x}_i$ **的值相关，那么为什么重构时取** $P=W$ **而与** $\boldsymbol{x}_i$ **无关呢？至此，问题陷入困境，不过我们可以先看一下知乎上对重构原理的推导，见链接 知乎- 重构的推导。在此之前，我们提醒大家注意，这里若考虑所有的样本点，而不是只最小化一个样本点** $\boldsymbol{x}_i$ **的重构差距，所得问题的形式就和 PCA 所求解的约束优化问题很像了，这是一个伏笔，我们稍后再谈。**

前面，我们以 W 为给定，而以 P 为变量最小化重构差距，但知乎上给出的推导思路则恰好相反，认为：(2) P 是给定的，而 W 为变量 (也就是以 $\boldsymbol{z}_i$ 为变量) 最小化重构差距：

$$\arg\min_{\boldsymbol{z}_i} -2\boldsymbol{x}_i^T P\boldsymbol{z}_i + \boldsymbol{z}_i^T\boldsymbol{z}_i$$

$$s.t.\ \ W^TW = I_{d'\times d'}$$

令上式对 $\boldsymbol{z}_i$ 的导数为零：

$$2\boldsymbol{z}_i - 2P^T\boldsymbol{x}_i = \boldsymbol{0}$$

$$\Updownarrow$$

$$\boldsymbol{z}_i = P^T\boldsymbol{x}_i$$

我们找到了变量 $\boldsymbol{z}_i$ **与** $P, \boldsymbol{x}_i$ **之间的关系，而我们又有** $\boldsymbol{z}_i = W^T\boldsymbol{x}_i$，**则** $P=W$，**又** P **的列向量为相互正交的单位向量，这使得** W **的约束条件** $W^TW=I$ **自然满足。**

可以看到，(1)、(2) 的目标均是最小化重构差距，但二者得到的结果却**看似不同**：虽然我们没有求解出 (1) 中 P 的最优点，但显然其值除了与 W 相关外，还与样本点 $\boldsymbol{x}_i$ 的值相关；而 (2) 中，我们却直接得到了 $W=P$ 这样一个结论。**这看似矛盾的两点实际上是统一的，事实上，我们处理的本质上是一个"二元函数"的优化问题，变量有两个，即** $W_{d\times d'}, P_{d\times d'}$，**我们要求** $W^TW = P^TP = I_{d'\times d'}$：

$$\min_{W,P} \|\boldsymbol{x}_i - PW^T\boldsymbol{x}_i\|_2^2$$

$$s.t.\ \ W^TW = P^TP = I_{d'\times d'}$$

我们知道，对于二元函数 $f(x,y)$ **的最值问题，我们通过** $\dfrac{\partial f}{\partial x}=0, \dfrac{\partial f}{\partial y}=0$ **来求解。其中，前一个式子可看作是以** x **为变量** y **为固定值最小化函数** f **得到；而后一个式子可看作是以** y **为变量** x **为固定值最小化函数** f **得到，联立二者求解即可得** x, y **的具体数值。虽然我们在构建方程的时候只研究一个变量，而将令一个变量视为给定，但我们要明确的是，我们得到的方程中** x, y **均为变量。这里也是一样的，我们联立 (1) 和 (2) 就可计算出** P, W **的最终数值，它们最终的结果只与样本** $\boldsymbol{x}_i$ **有关，因此 (1)、(2) 中的结果并不矛盾。事实上，我们由 (2) 得到** $P=W$ **这一结论，将其代入到 (1) 中可以看到，此时问题的形式和前面 PCA 中的一样，只不过处理的样本点的个数不同，PCA 中的样本有** N **个数据点，而 (1) 中的样本只有一个数据点，这也就解释了我们前文伏笔中的内容，因为 (1) 和 PCA 中的问题本质上是同一个问题。**

总而言之，最小化重构差距本质上是一个二元函数的优化问题，因此实际上对应存在两个子问题，最终结果由这两个子问题的结果联合给出。前文 PCA 的推导中只关注了其中的一个子问题，而直接使用了另一个子问题的结论，即取 $P=W$，**这就会令人感到困惑，即为什么重构的时候用降维矩阵** W^T **的转置左乘降维后的样本即** $W\boldsymbol{z}_i$ **就可以了呢。事实上取** $P=W$ **也是最小化重构差距这一问题结果的一部分。**

4.4 西瓜书上逐一选取最大方差方向描述的证明

PCA 将样本从原始坐标系映射到新坐标系下，我们这里指出，**通过逐一选取方差最大的方向作为新的坐标轴，我们能得到和 PCA 相同的结果**：新坐标系的第一个坐标轴是原始数据投影到该轴后方差最大的方向；新坐标系的第二个坐标轴和第一个坐标轴正交，而且使得原始数据在该轴上的投影方差最大，以此类推，进行 d 次选取得到 d 个新的坐标轴，其中，大部分方差都包含在前面几个新坐标轴中，我们通过选取前 d' 个坐标轴，而忽略剩下的坐标轴，就对数据进行了降维处理。**上述过程在数学上的操作如下**：先求协方差矩阵 XX^T 的特征值和特征向量，取最大特征值对应的特征向量 $\boldsymbol{w}_1$；再求 $XX^T-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T$ 的特征值和特征向量，取最大特征值对应的特征向量 $\boldsymbol{w}_2$，以此类推。接下来，我们要解决两个问题：第一个问题是为什么逐一选取方差最大的方向在数学上为上述操作；第二个问题是上述操作得到的结果为什么和 PCA 的结果相同 (事实上，这两个问题是对"为什么逐一选取方差最大的方向与 PCA 的结果相同"这一问题的分解)。

我们首先使用数学归纳法解决第一个问题：

(i) 原始数据为 $X_{d\times N}$，我们选择投影方向为 $\boldsymbol{u}_1$，是一个 d 维单位列向量，则投影后的数据为 $\boldsymbol{u}_1^TX$。原始数据 X 是中心化的数据，则可推得投影后的数据也是中心化的。因此，投影后数据的方差，也就是原始数据在 $\boldsymbol{u}_1$ 方向上的方差为 $\boldsymbol{u}_1^TX(\boldsymbol{u}_1^TX)^T/(N-1)=\boldsymbol{u}_1^TXX^T\boldsymbol{u}_1/(N-1)$，则问题为：

$$\max_{\boldsymbol{u}_1}\boldsymbol{u}_1^TXX^T\boldsymbol{u}_1$$
$$s.t.\ \ \boldsymbol{u}_1^T\boldsymbol{u}_1=1$$

为**瑞利商的最值问题，取 $\boldsymbol{u}_1$ 为 XX^T 的最大特征值 λ_1 对应的单位特征向量 $\boldsymbol{w}_1$ 即可。**

(ii) 接下来我们求第二个坐标轴 $\boldsymbol{u}_2$。类似于施密特正交化，对于每个样本点 $\boldsymbol{x}_i$，我们首先减去它在 $\boldsymbol{u}_1$ 也就是 $\boldsymbol{w}_1$ 方向上的投影，以保证我们在与第一个坐标轴垂直的空间中寻找第二个坐标轴，处理后的第 i 个样本点为 $\boldsymbol{x}_i-\boldsymbol{w}_1\boldsymbol{w}_1^T\boldsymbol{x}_i$，由此可得数据为 $X-\boldsymbol{w}_1\boldsymbol{w}_1^TX$，在 $\boldsymbol{u}_2$ 上的投影为 $\boldsymbol{u}_2^T(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)$，方差为

$$\frac{1}{N-1}\left([\boldsymbol{u}_2^T(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)][\boldsymbol{u}_2^T(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)]^T\right)$$
$$=\frac{1}{N-1}\left(\boldsymbol{u}_2^T[(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)^T]\boldsymbol{u}_2\right)$$

问题即为：

$$\max\boldsymbol{u}_2^T[(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)^T]\boldsymbol{u}_2$$
$$s.t.\ \ \boldsymbol{u}_2^T\boldsymbol{u}_2=1$$

依然是瑞利商的最大值问题，则 $\boldsymbol{u}_2$ 即为矩阵 $(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)^T$ 最大特征值对应的单位特征向量。接下来我们只需证明 $(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)^T=XX^T-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T$ 即可。为了便于推导，我们不妨记 $\boldsymbol{w}_1\boldsymbol{w}_1^T=A_1$。我们有：

$$A_1A_1=A_1,\ \ A_1^T=A_1$$
$$XX^T\boldsymbol{w}_1=\lambda_1\boldsymbol{w}_1\Rightarrow XX^T\boldsymbol{w}_1\boldsymbol{w}_1^T=\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T\Leftrightarrow XX^TA_1=\lambda_1A_1\Leftrightarrow A_1XX^T=\lambda_1A_1$$

而

$$(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)(X-\boldsymbol{w}_1\boldsymbol{w}_1^TX)^T=(X-A_1X)(X^T-X^TA_1^T)$$
$$=(X-A_1X)X^T-(X-A_1X)(X^TA_1^T)=XX^T-\lambda_1A_1=XX^T-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T$$
$$\text{with}$$
$$(X-A_1X)X^T=XX^T-A_1XX^T=XX^T-\lambda_1A_1$$
$$(X-A_1X)(X^TA_1^T)=XX^TA_1^T-A_1XX^TA_1^T=\lambda_1A_1-\lambda_1A_1A_1=0$$

因此，**我们取 $XX^T - \lambda_1 \boldsymbol{w}_1 \boldsymbol{w}_1^T$ 最大特征值 (记为 λ_2) 对应的与 $\boldsymbol{w}_1$ 正交的单位特征向量 $\boldsymbol{w}_2$ 为 $\boldsymbol{u}_2$ 即可。**

(iii) 对于第 $t+1$ 个坐标轴的选取，我们要证明：

$$
\begin{aligned}
&(X - A_1X - A_2X - \cdots - A_tX)(X^T - X^TA_1^T - X^TA_2^T - \cdots - X^TA_t^T) \\
&= (X - \sum_{i=1}^{t} A_iX)(X^T - \sum_{i=1}^{t} X^TA_i^T) \\
&= X - \sum_{i=1}^{t} \lambda_i A_i \\
&\text{with} \\
&\boldsymbol{w}_i^T \boldsymbol{w}_j = \delta_{ij} \\
&A_i = \boldsymbol{w}_i \boldsymbol{w}_i^T \\
&A_iA_j = A_i\delta_{ij}, \;\; A_i^T = A_i \\
&XX^TA_i = \lambda_i A_i, \;\; A_iXX^T = \lambda_i A_i
\end{aligned}
$$

则

$$
\begin{aligned}
&(X - \sum_{i=1}^{t} A_iX)(X^T - \sum_{i=1}^{t} X^TA_i^T) \\
&= (X - \sum_{i=1}^{t} A_iX)X^T - \\
&(X - \sum_{i=1}^{t} A_iX) \sum_{i=1}^{t} X^TA_i^T
\end{aligned}
$$

第一项

$$
(X - \sum_{i=1}^{t} A_iX)X^T = XX^T - \sum_{i=1}^{t} A_iXX^T = X - \sum_{i=1}^{t} \lambda_i A_i
$$

第二项

$$
\begin{aligned}
&(X - \sum_{i=1}^{t} A_iX) \sum_{i=1}^{t} X^TA_i^T = \sum_{i=1}^{t} \lambda_i A_i - \sum_{i=1}^{t} \lambda_i A_i = 0 \\
&\text{with} \\
&XX^TA_i^T = \lambda_i A_i \\
&A_iXX^TA_j = \lambda_i A_iA_j = \lambda_i A_i\delta_{ij}
\end{aligned}
$$

则**第 $t+1$ 个坐标轴 $\boldsymbol{u}_{t+1}$ 就是选择 $X^TX - \sum_{i=1}^{t} \lambda_i A_i$ 最大特征值 (记为 λ_{t+1}) 对应的与 $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_t$ 正交的单位特征向量 $\boldsymbol{w}_{t+1}$ 。**

至此，我们解决了第一个问题。

对于第二个问题，事实上， XX^T 为 d 阶半正定实对称矩阵，可以对其进行如下的特征值分解：
$XX^T = \sum_{i=1}^{d} \lambda_i \boldsymbol{w}_i \boldsymbol{w}_i^T$ ，其中 $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_d \geqslant 0$ ， $\boldsymbol{w}_i \boldsymbol{w}_j = \delta_{ij}$ 。可以看到，PCA 就是选择前 d' 个特征值 $\lambda_1, \lambda_2, \cdots, \lambda_{d'}$ 对应的特征向量 $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_{d'}$ ，与依次取
$XX^T - \sum_{i=1}^{t} \lambda_i \boldsymbol{w}_i \boldsymbol{w}_i^T, \; (t = 0, 1 \cdots, d'-1)$ 的最大的特征值对应的特征向量结果相同，比如对于第 $t+1$ 个坐标轴的选取：

$$XX^T - \sum_{i=1}^{t} \lambda_i \boldsymbol{w}_i \boldsymbol{w}_i^T = \sum_{i=t+1}^{d} \lambda_i \boldsymbol{w}_i \boldsymbol{w}_i^T$$

对于矩阵 $\sum_{i=t+1}^{d} \lambda_i \boldsymbol{w}_i \boldsymbol{w}_i^T$ 而言，易知其最大的特征值为 λ_{t+1}，对应的特征向量为 $\boldsymbol{w}_i$。

至此，我们证明了逐一选取方差最大的方向与 PCA 等价，这也体现了方差越大的方向所含的信息也就越多这一观点。

4.5 独立成分分析 (Independent Component Analysis, ICA)

> 参考文献：
>
> 1. Independent Component Analysis: Algorithms and Applications - Aapo Hyvärinen and Erkki Oja - 2000

独立成分分析最典型的例子就是鸡尾酒会问题 (the cocktail-party problem)，我们试图从不同麦克风记录的混合声音中分离出每个人的声音。

4.5.1 ICA 模型

ICA 假设：

$$\boldsymbol{x} = A\boldsymbol{s}$$

$$\boldsymbol{x} = \sum_{i=1}^{n} \boldsymbol{a}_i s^{(i)}$$

$$x^{(i)} = a_{i1} s^{(1)} + a_{i2} s^{(2)} + \cdots + a_{in} s^{(n)},\ \ i = 1, \cdots, n$$

其中，$s^{(i)}$ 为 n 个独立的信号源，我们假设为 n **个相互独立均值为 0 方差为 1 的非高斯分布随机变量**；$x^{(i)}$ 为 n 个独立信号源的线性组合，因此也是零均值的，一般假设与信号源个数相同，即也为 n 个。独立成分 $s^{(i)}$ 为隐变量，无法被直接观测，混合矩阵 A 也是未知的，我们能观测到的只有随机向量 $\boldsymbol{x}$，独立成分分析就是试图通过 $\boldsymbol{x}$ 得到 A 和 $\boldsymbol{s}$。

我们作如下说明：

- 可以看到，ICA 假设输入 $\boldsymbol{s}$ 相互独立，而输出 $\boldsymbol{x}$ 为输入的线性组合，我们只能观测到输出 $\boldsymbol{x}$，而 **输入 $\boldsymbol{s}$ 无法观测，是个隐变量**。ICA 可视为一种盲信号分离 (blind source separation or blind signal separation, BSS) 方法。
- 统计独立的含义。两个随机变量 y_1, y_2 相互独立，意味着 $p(y_1, y_2) = p(y_1)p(y_2)$，$p(\cdot)$ 表示概率密度函数。一个推论是，两个相互独立的随机变量对任意函数 $h_1(\cdot), h_2(\cdot)$ 总有：

$$\mathbb{E}[h_1(y_1)h_2(y_2)] = \mathbb{E}[h_1(y_1)]\mathbb{E}[h_2(y_2)]$$

而不相关的充要条件是 $\mathbb{E}[y_1, y_2] = \mathbb{E}[y_1]\mathbb{E}[y_2]$，可见不相关只是部分独立 (partly independent)。举个例子，(y_1, y_2) 取 $(0,1), (0,-1), (1,0), (-1,0)$ 的概率均为 $1/4$，可以看到 y_1, y_2 线性无关，而

$$\mathbb{E}[y_1^2 y_2^2] = 0 \neq \mathbb{E}[y_1^2]\mathbb{E}[y_2^2] = \frac{1}{4}$$

表明 y_1, y_2 并不独立。

- $s^{(i)}$ 方差为 1。因为 A 和 $\boldsymbol{s}$ 均未知，因此信号源的方差无法被唯一地确定。因为对于信号源 $s^{(i)}$，我们总可以乘以一个常数，然后对对应混合系数除以同一个常数而保证输出 $\boldsymbol{x}$ 不变。因此，我们不妨令独立信号源的方差均为 1。**但尽管如此，信号源仍无法被唯一地确定，因为还存在符号的问题，即若 $\boldsymbol{s}$ 为信号源 (混合矩阵为 A)，那么 $-\boldsymbol{s}$ 也可作为信号源 (混合矩阵为 $-A$)，但好在符号问题一般并不重要。**

- **信号源服从非高斯分布。**若 $s^{(i)}$ 为相互独立的高斯变量，则对任意正交矩阵 B， Bs 中各分量也是相互独立的高斯变量。因此，此时我们无法确认 ICA 分离得到的到底是 $\boldsymbol{s}$ 还是 $B\boldsymbol{s}$，ICA 失效。但可以看到的是，如果只有一个信号源服从高斯分布，那么 ICA 仍然可以进行。
- 为了简便起见，我们假设未知的**混合矩阵 A 是方阵，但这个假设是可以放松的，可见 4.5.6 节投影寻踪**；
- 在上述模型中加入噪声项会更加符合实际情况，但为了简便起见，我们还是采用无噪声模型，因为无噪声模型的估计就已经比较困难了，而且对很多应用来说也足够了。**若模型存在高斯噪声，则可先使用 PCA 去噪，再进行 ICA。**
- ICA 的优化指标称为 contrast functions.

4.5.2 ICA 之最大化非高斯性 (Nongaussianity)

根据中心极限定理，相互独立的随机变量之和趋于高斯分布。因此，两个相互独立的随机变量之和通常比其中一个更趋于高斯分布。我们假设 $y^{(i)}$ 为 $\boldsymbol{x}$ 的线性组合： $y^{(i)}=\boldsymbol{w}_i^T\boldsymbol{x}=\boldsymbol{w}_i^TA\boldsymbol{s}\triangleq\boldsymbol{v}_i\boldsymbol{s}$ 。**当 $\boldsymbol{w}_i$ 使 $y^{(i)}$ 的非高斯性达到极大时， $\boldsymbol{v}_i$ 就应该只存在一个元素非零，而其他元素均为 0，此时我们也就分离得到了一个独立信号源 $y^{(i)}$** 。上述优化问题的极大值点不考虑正负号共有 n 个 (考虑正负号则有 $2n$ 个，信号源为 $s^{(i)}$ 及 $-s^{(i)}$ 均可)。为了找到这 n 个不同的信号源，我们需要找到所有的局部极大值点，这并不困难。考虑到这些不同的独立成分是线性无关的，我们总能将搜索限制在与前面已得的独立成分线性无关的空间中 (**易推知 $\boldsymbol{w}_i$ 间正交，我们相当于在寻找 n 个相互垂直的使非高斯性最大的方向 $\boldsymbol{w}_i$，这与 PCA 有些类似，但不同的是这些方向对应的负熵虽然存在大小之分，但我们认为它们均是重要的，而不像 PCA 中那样，会舍弃奇异值较小的方向**)，从而在每次搜索时只需寻找当前子空间下的某个局部极大值。可以看到，**上述思路是相当启发式的 (rather heuristic)，但由下一小节可知，这仍然是有严格证明的。**

非高斯性的度量 (Measures of nongaussianity):

1. 假设检验中，偏度、**峰度**可用于检验一个分布是否为高斯分布。很自然地，我们也可以由此入手构造非高斯性的度量。均值为 0 方差为 1 的随机变量 y 的峰度 (kurtosis or the fourth-order cumulant) 为:

$$kurt(y)=\mathbb{E}[y^4]-3(\mathbb{E}[y^2])^2=\mathbb{E}[y^4]-3$$

若 y 为高斯分布，则 $kurt(y)=0$；对于大部分非高斯分布， $kurt(y)\neq0$ 。**因此，可以用峰度的绝对值 (或平方等) 作为非高斯性的度量，也就是 ICA 的优化指标。**

峰度可为正，可为负。峰度为正的随机变量被称作是 supergaussian or leptokurtic，峰度为负的随机变量被称作是 subgaussian or platykurtic。Supergaussian 随机变量的概率密度函数的形状比较尖，尾部也比较大，即随机变量在 0 处和取值较大处概率密度相对较大，而在中间处的概率密度相对较小，典型的就是拉普拉斯分布。Subgaussian 随机变量有一个相对平坦的概率密度函数，典型地就是均匀分布。

优点：理论和计算均很简单。**缺点**：峰度对异常点 (outliers) 非常敏感 (Its value may depend on only a few observations in the tails of the distribution, which may be erroneous or irrelevant observations)，这也就意味着峰度并不是一个鲁棒 (robust) 的非高斯性度量。Below we shall consider negentropy whose properties are rather opposite to those of kurtosis, and finally introduce approximations of negentropy that more or less combine the good properties of both measures.

2. **负熵 (Negentropy)**

随机向量 $\boldsymbol{y}$ 的信息熵:

$$H(y)=-\int p(\boldsymbol{y})\log p(\boldsymbol{y})dy$$

方差相等时，高斯分布的信息熵最大 (A fundamental result of information theory is that a gaussian variable has the largest entropy among all random variables of equal variance；若去掉方差相等的限制，在同等情况下，显然是均匀分布的信息熵最大)。**这意味着信息熵可用于衡量非高斯性，为了得到非高斯性的非负度量，且满足当且仅当分布为高斯分布时该度量为 0，我们引入负熵**:

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

其中， $\boldsymbol{y}_{gauss}$ 为高斯随机向量，其协方差矩阵与 $\boldsymbol{y}$ 相同。负熵一个有趣的性质是，**对于可逆线性变换，负熵保持不变。**

负熵的**优点**是理论基础扎实。就统计性质而言，在某种意义上，负熵是非高斯性最优的估计。负熵的**缺点**是直接计算非常困难，因此我们常采用近似的方法计算负熵。

负熵的近似计算：

- 高阶矩近似：

$$J(y) \approx \frac{1}{12}(\mathbb{E}[y^3])^2 - \frac{1}{48}(kurt(y))^2$$

其中， y 均值为 0 方差为 1 。和峰度一样，这一近似的鲁棒性不好。

- 基于最大熵原则 (the maximum-entropy principle) 的近似：

$$J(y) \approx \sum_{i=1}^{t} k_i(\mathbb{E}[G_i(y)] - \mathbb{E}[G_i(z)])^2$$

其中， k_i 为正的常数； z 为标准高斯变量； y 均值为 0 方差为 1 ； $G_i(\cdot)\ (i = 1, \cdots, t)$ 是一些非二次的函数 (nonquadratic functions)。

若我们只使用一种函数，则：

$$J(y) \propto (\mathbb{E}[G(y)] - \mathbb{E}[G(z)])^2$$

若 y 是对称的，上式显然是高阶矩近似 (取 $G(y) = y^4$) 的推广。当我们选择合适的 $G(\cdot)$ 时，我们能得到比高阶矩近似更好的对负熵的近似。比如，我们可以选择变化缓慢的 $G(\cdot)$ ，这样得到的结果就更加鲁棒。常用的 $G(\cdot)$ 有：

$$G(y) = \frac{1}{\alpha}\log\cosh\alpha y$$
$$G(y) = -\exp\left(-y^2/2\right)$$

其中， α 为 $[1, 2]$ 间的常数。

4.5.3 ICA 之最小化互信息 (Mutual information)

1. **互信息的定义。**

n 个随机变量 $y^{(i)}\ (i = 1, \cdots, n)$ 间的互信息为联合概率密度 $p_{12\cdots n}(y^{(1)}, \cdots, y^{(n)})$ 对边缘密度累积 $\prod_{i=1}^{n} p_i(y^{(i)})$ 的 KL 散度：

$$I(y^{(1)}, \cdots, y^{(n)}) = KL\left(p_{12\cdots n}(y^{(1)}, \cdots, y^{(n)}) \| \prod_{i=1}^{n} p_i(y^{(i)})\right)$$

$$= \int p_{1\cdots n}\log\frac{p_{1\cdots n}}{\prod_{i=1}^{n} p_i} = -\int p_{1\cdots n}\sum_{i=1}^{n}\log p_i + \int p_{1\cdots n}\log p_{1\cdots n}$$

$$= -\sum_{i=1}^{n}\int p_{1\cdots n}\log p_i - H(\boldsymbol{y}) = -\sum_{i=1}^{n}\int p_i\log p_i - H(\boldsymbol{y})$$

$$= \sum_{i=1}^{n} H(y^{(i)}) - H(\boldsymbol{y})$$

其中， $\int p_{1\cdots n}\log p_i d\boldsymbol{y} = \int p_i\log p_i dy^{(i)}$

可以看到，**互信息是随机变量独立性最本质的度量**，而不是像相关系数那样只考察了部分 (也就是一阶) 独立性。

2. **互信息与负熵之间的关系。**

互信息一个重要的性质是，对**可逆**线性变换 $\boldsymbol{y}=W\boldsymbol{x}$：

$$I(\boldsymbol{y})=\sum_{i=1}^{n}H(y^{(i)})-H(\boldsymbol{x})-\log|\det(W)|\cdots\cdots(1)$$

由此，我们可推得互信息与负熵之间的关系：

- **若 $y^{(i)}$ 线性无关且方差为 1**，即协方差矩阵为单位矩阵，则：

$$\mathbb{E}[\boldsymbol{y}\boldsymbol{y}^T]=W\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T]W^T=I$$
$$\det(I)=1=\det(W)\det\left(\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T]\right)\det\left(W^T\right)$$

这意味着此时 $\det(W)$ 为常数。因为 $y^{(i)}$ 的方差为 1，则其信息熵与负熵只相差一个常数和符号，最终我们可推得**互信息和负熵的关系如下**：

$$I(\boldsymbol{y})=C-\sum_{i=1}^{n}J(y^{(i)})\cdots\cdots(2)$$

其中，C 为一个**不依赖**于 W 的常数。

- 我们可以看到，**当限制 $y^{(i)}$ 线性无关时，寻找一个可逆矩阵 W 使得 $\boldsymbol{y}$ 各分量的互信息最小等价于最大化估计量 $y^{(i)}$ 的非高斯性之和。**事实上，线性无关的条件不是必须的，但其存在可以大大地简化计算，因为我们可以直接使用更简单的 (2) 式而不是复杂的 (1) 式。(Rigorously speaking, (2) shows that ICA estimation by minimization of mutual information is equivalent to maximizing the sum of nongaussianities of the estimates, when the estimates are constrained to be uncorrelated. The constraint of uncorrelatedness is in fact not necessary, but simplifies the computations considerably, as one can then use the simpler form in (2) instead of the more complicated form in (2).) 可以看到，**基于最小化互信息的 ICA 给出了上一小节寻找非高斯性最大的方向 $\boldsymbol{w}_i$ 这样一种启发式方法的严格证明。**

4.5.4 ICA 的其他方法

1. **极大似然估计**

可以基于极大似然估计对 ICA 模型进行求解，似然函数见下文。

A very popular approach for estimating the ICA model is maximum likelihood estimation, which is closely connected to the infomax principle. It is possible to formulate directly the likelihood in the noise-free ICA model, which was done in (Pham et al., 1992), and then estimate the model by a maximum likelihood method. Denoting by $W=(\boldsymbol{w}_1,\cdots,\boldsymbol{w}_n)$ the matrix A^{-1}, the log-likelihood takes the form:

$$\mathcal{L}=\sum_{j=1}^{N}\sum_{i=1}^{n}\log p_{s^{(i)}}(\boldsymbol{w}_i^T\boldsymbol{x}_j)+N\log|\det(W)|$$

其中，N 为样本容量，$\boldsymbol{x}_j$ 为第 j 个样本，含有 n 个分量；$p_{s^{(i)}}(\cdot)$ 为 $s^{(i)}$ 的概率密度函数，这里假设是已知的。

> 推导似然函数时会用到的结论：In general, for any random vector $\boldsymbol{x}$ with density $p_x(\cdot)$ and for any matrix W, the density of $\boldsymbol{y}=W\boldsymbol{x}$ is given by $p_x(W\boldsymbol{x})|\det(W)|$.

我们指出，**it (基于极大似然估计对 ICA 模型进行求解) is essentially equivalent to minimization of mutual information。**我们考察对数似然函数的期望：

$$\frac{1}{N}\mathbb{E}[\mathcal{L}] = \sum_{i=1}^{n}\mathbb{E}[\log p_{s^{(i)}}(\boldsymbol{w}_i^T\boldsymbol{x}_j)] + \log|\det(W)|$$

可见，当 $p_{s^{(i)}}(\cdot)$ 等于 $\boldsymbol{w}_i^T\boldsymbol{x}_j$ 的真实概率密度时，上式右边第一项 $\sum_{i=1}^{n}\mathbb{E}[\log p_{s^{(i)}}(\boldsymbol{w}_i^T\boldsymbol{x}_j)] = -\sum_{i=1}^{n}H(\boldsymbol{w}_i^T\boldsymbol{x}_j)$，则似然函数与互信息的计算公式 (1) 只相差一个常数和符号。而在基于 MLE 进行 ICA 的实际过程中， 因为我们并不知道独立成分 $s^{(i)}$ 的概率密度，因此我们常先对 $\boldsymbol{w}_i^T\boldsymbol{x}$ 的概率分布进行极大似然估计，并以此作为 $s^{(i)}$ 的概率密度。因此，**在实际计算层面，使用 MLE 和互信息进行 ICA 是等价的。**可以看到，基于 MLE 的 ICA 需正确估计出 $p_{s^{(i)}}(\cdot)$，但使用合理的非高斯性度量则无需考虑这个问题。

2. **The infomax principle (i.e. the principle of network entropy maximization)**

The infomax principle 就是要最大化神经网络输出的熵，即最大化下面给出的 L_2。而通过合适地选取 L_2 中的非线性函数 $\phi_i(\cdot)$，上述原则也可用于 ICA 模型的估计。事实上，当 $\phi_i'(\cdot) = p_{s^{(i)}}(\cdot)$ 即 $\phi_i(\cdot)$ 为独立成分的累积分布函数时，The infomax principle 等价于上面介绍的极大似然估计法。

Another related contrast function was derived from a neural network viewpoint in (Bell and Sejnowski, 1995; Nadal and Parga, 1994). This was based on maximizing the output entropy (or information flow) of a neural network with non-linear outputs. Assume that $\boldsymbol{x}$ is the input to the neural network whose outputs are of the form $\phi_i(\boldsymbol{w}_i^T\boldsymbol{x})$, where the $\phi_i(\cdot)$ are some non-linear scalar functions, and the $\boldsymbol{w}_i$ are the weight vectors of the neurons. One then wants to **maximize the entropy of the outputs**:

$$L_2 = H(\phi_1(\boldsymbol{w}_1^T\boldsymbol{x}), \cdots, \phi_n(\boldsymbol{w}_n^T\boldsymbol{x}))$$

If the $\phi_i(\cdot)$ are well chosen, this framework also enables the estimation of the ICA model. Indeed, several authors, e.g., (Cardoso, 1997; Pearlmutter and Parra, 1997), proved the surprising result that the principle of network entropy maximization, or "infomax", is equivalent to maximum likelihood estimation. This equivalence requires that the non-linearities $\phi_i(\cdot)$ used in the neural network are chosen as the cumulative distribution functions corresponding to the densities $p_{s^{(i)}}(\cdot)$, i.e., $\phi_i'(\cdot) = p_{s^{(i)}}(\cdot)$.

4.5.5 ICA 具体算法之 FastICA

本小节介绍 ICA 的一个具体算法 FastICA，分为两个部分，首先是数据预处理，接着是算法主体部分。

4.5.5.1 数据预处理

(1) **中心化 (centering)。**将观测数据 $(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N)$ 中心化：

$$\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i - \frac{1}{N}\sum_{j=1}^{N}\boldsymbol{x}_j, \quad \text{for all i}$$

由此， $\boldsymbol{s}$ 也是零均值的。

(2) **漂白 (whitening)，就是指对样本进行线性变换，使得变换后样本的协方差矩阵为单位矩阵 I。**可以看到，相比使方差为 1， 漂白还要求变换后样本各维度间的协方差为 0， 要求更高。

观察数据 $X = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N)$ 的协方差矩阵：

$$\frac{1}{N-1}\sum_{i=1}^{N}\boldsymbol{x}_i\boldsymbol{x}_i^T = \frac{1}{N-1}XX^T$$

类似地，经过线性变换 $B_{n\times n}$ 后协方差矩阵：

$$\frac{1}{N-1}\sum_{i=1}^{N}B\boldsymbol{x}_i(B\boldsymbol{x}_i)^T = \frac{1}{N-1}BXX^TB^T$$

对 XX^T 进行特征值分解：$XX^T = U\Lambda U^T$，则我们取 $B = \sqrt{N-1}U\Lambda^{-1/2}U^T$，代入上式可令协方差矩阵为 I。从而，**线性变换矩阵 $B = \sqrt{N-1}U\Lambda^{-1/2}U^T$，漂白操作为**：

$$\boldsymbol{x}_i \leftarrow B\boldsymbol{x}_i, \text{ for all i}$$

可以看到，**因为 XX^T 总可以进行特征值分解，因此漂白总可以进行。此外，在漂白前我们还可以对样本进行降维 (比如 PCA 降维) 以去噪，从而避免过拟合。**

漂白的好处：若 $X = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N)$ 已漂白，又 $\boldsymbol{x} = A\boldsymbol{s}$，其中 $s^{(1)}$ 相互独立，方差为 1，则随机变量 $\boldsymbol{x}$ 的协方差矩阵：

$$\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T] = I = A\mathbb{E}[\boldsymbol{s}\boldsymbol{s}^T]A^T = AA^T$$

可知，A 此时为正交矩阵。而正交矩阵的自由度为 $n(n-1)/2$，相比于一般情况下的自由度 n^2，减少了一半还多，大大降低了求解难度，提高了计算效率。

(3) 其他

If the data consists of time-signals, some band-pass filtering may be very useful. Note that if we filter linearly the observed signals to obtain new signals, the ICA model still holds for the new signals, with the same mixing matrix.

4.5.5.2 FastICA 算法主体

经过上述预处理过程后，接下来 FastICA 处理的就是**已中心化和漂白的数据。**我们先介绍单个独立信号源的寻找，在此基础上再介绍多个独立信号源的寻找，其中多个独立信号源的寻找存在着两种方法。

1. **单个独立信号源的寻找**

我们选择负熵作为非高斯性的度量，由前文可知 $J(y) \propto (\mathbb{E}[G(y)] - \mathbb{E}[G(z)])^2$，常用的 $G(\cdot)$ 函数及其导数 $g(\cdot)$ 为：

$$G(y) = \frac{1}{\alpha}\log\cosh\alpha y, \quad g(y) = \tanh(\alpha y)$$
$$G(y) = -\exp\left(-y^2/2\right), \quad g(y) = y\exp\left(-y^2/2\right)$$

其中，$1 \leqslant \alpha \leqslant 2$ 且常取 1。**We recall that 我们限制独立信号源 $y = \boldsymbol{w}^T\boldsymbol{x}$ 的方差为 1，对于漂白后的数据来说这相当于限制 $\boldsymbol{w}$ 的范数为 1（$\mathbb{E}[(\boldsymbol{w}^T\boldsymbol{x})^2] = \|\boldsymbol{w}\|^2 = 1$）。**

FastICA 基于不定点迭代法 (a fixed-point iteration scheme) 寻找最优的向量 $\boldsymbol{w}$ 使 $\boldsymbol{w}^T\boldsymbol{x}$ 的非高斯性最大，该方法可被推导为近似牛顿迭代 (approximative Newton iteration) 法。**算法的步骤如下**：

(1) 随机初始化向量 $\boldsymbol{w}$；

(2) 令 $\boldsymbol{w}^* = \mathbb{E}[\boldsymbol{x}g(\boldsymbol{w}^T\boldsymbol{x})] - \mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})]\boldsymbol{w}$；

(3) 单位化，令 $\boldsymbol{w} = \boldsymbol{w}^*/\|\boldsymbol{w}^*\|$；

(4) 若还未收敛，则回到步骤 (2)。

上述算法中的期望由相应的样本估计量给出，但考虑到计算效率，我们通常只使用全部样本的一部分来计算期望，且每次迭代时使用不同的样本子集，若计算不收敛，则可增大样本子集的容量。

上述算法的推导如下：首先，最大化 $\boldsymbol{w}^T\boldsymbol{x}$ 的负熵等价于 $\mathbb{E}[G(\boldsymbol{w}^T\boldsymbol{x})]$ 取极值，考虑到约束 $\|\boldsymbol{w}\|^2 = 1$，由 KKT 条件，我们有：

$$\mathcal{F}(\boldsymbol{w}) \triangleq \mathbb{E}[\boldsymbol{x}g(\boldsymbol{w}^T\boldsymbol{x})] - \lambda\boldsymbol{w} = \boldsymbol{0}$$

我们使用牛顿迭代法求解上述方程。记函数 $\mathcal{F}(\boldsymbol{w})$ 的雅克比矩阵为 $\mathcal{J}(\boldsymbol{w})$，则：

$$\begin{aligned}\mathcal{J}(\boldsymbol{w}) &= \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T g'(\boldsymbol{w}^T\boldsymbol{x})] - \lambda I \\ &\approx \mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})]I - \lambda I\end{aligned}$$

其中，考虑到数据是球形的 (Since the data is sphered)，我们有 $\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T g'(\boldsymbol{w}^T\boldsymbol{x})] \approx \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T]\mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})] = \mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})]I$ 。由此，**雅克比矩阵被近似为很容易求逆的对角矩阵，从而可得如下的近似牛顿迭代公式**：

$$\boldsymbol{w}^* = \boldsymbol{w} - \frac{\mathbb{E}[\boldsymbol{x}g(\boldsymbol{w}^T\boldsymbol{x})] - \lambda\boldsymbol{w}}{\mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})] - \lambda}$$

化简可得：

$$(-\mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})] + \lambda)\boldsymbol{w}^* = -\mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})]\boldsymbol{w} + \mathbb{E}[\boldsymbol{x}g(\boldsymbol{w}^T\boldsymbol{x})]$$

我们只需得到向量 $\boldsymbol{w}^*$ 的方向再单位化即可，因此不妨令上式左端 $\boldsymbol{w}^* \leftarrow (-\mathbb{E}[g'(\boldsymbol{w}^T\boldsymbol{x})] + \lambda)\boldsymbol{w}^*$ ，这样就得到了步骤 (2)，然后在第三步中单位化 $\boldsymbol{w}^*$ 。

2. **多个独立信号源的寻找**

信号源 $\boldsymbol{w}_1^T\boldsymbol{x}, \cdots, \boldsymbol{w}_n^T\boldsymbol{x}$ 相互独立，**可知 $\boldsymbol{w}_i$ 除了是单位向量外，彼此之间还正交。**

方法一：一个接一个地寻找独立信号源。假设我们已经找到了 q 个独立成分，即已经得到了 q 个向量 $\boldsymbol{w}_1, \cdots, \boldsymbol{w}_q$ ，**我们仍然使用单个独立信号源算法去寻找第 $q+1$ 个向量，只不过为了不让其收敛为已找到的向量，在每个迭代步中，我们均需调整 $\boldsymbol{w}_{q+1}$ 使其正交于已找到的各个向量。**我们只需对单个独立信号源算法的步骤 (3) 稍作修改即可，具体如下：

(1) 随机初始化向量 $\boldsymbol{w}_{q+1}$；

(2) 令 $\boldsymbol{w}_{q+1}^* = \mathbb{E}[\boldsymbol{x}g(\boldsymbol{w}_{q+1}^T\boldsymbol{x})] - \mathbb{E}[g'(\boldsymbol{w}_{q+1}^T\boldsymbol{x})]\boldsymbol{w}_{q+1}$；

(3) 令 $\boldsymbol{w}'_{q+1} = \boldsymbol{w}^*_{q+1} - \sum_{i=1}^{q}\boldsymbol{w}^*_{q+1}\boldsymbol{w}_i\boldsymbol{w}_i$；令 $\boldsymbol{w}_{q+1} = \boldsymbol{w}'_{q+1}/\|\boldsymbol{w}'_{q+1}\|_2$

(4) 若还未收敛，则回到步骤 (2)。

易知，上述一个接一个的寻找算法每次都找的是当前子空间中负熵的某个极值点，并不是最值点，因此，我们依次得到的方向 $\boldsymbol{w}_1, \cdots, \boldsymbol{w}_n$ 对应负熵的大小关系是不定的，并不一定是降序排列。显然，我们总可以对找到的方向 $\boldsymbol{w}_1, \cdots, \boldsymbol{w}_n$ 按其对应的负熵大小进行排序，但这并没有必要，ICA 不会像 PCA 中按 $\boldsymbol{w}_i$ 的"重要性"对其进行比较排序，ICA 认为得到的每个 $\boldsymbol{w}_i$ 都是重要的，尽管它们对应的负熵存在大小之分。

方法二：同时寻找多个独立信号源。类似于单个独立信号源的寻找，寻找多个独立信号源 $\boldsymbol{y} = \langle y^{(1)}, \cdots, y^{(n)}\rangle^T = \langle \boldsymbol{w}_1^T\boldsymbol{x}, \cdots, \boldsymbol{w}_n^T\boldsymbol{x}\rangle^T = W\boldsymbol{x}$ 时的**迭代公式为**：

$$W^* = W + \Gamma(\mathrm{B} + \mathbb{E}[g(\boldsymbol{y})\boldsymbol{y}^T])W$$

其中，$\mathrm{B} = diag(\beta_1, \cdots, \beta_n),\ \beta_i = -\mathbb{E}[y^{(i)}g(y^{(i)})]$ ，$\Gamma = diag(\gamma_1, \cdots, \gamma_n),\ \gamma_i = -1/(\beta_i - \mathbb{E}[g'(y^{(i)})])$ 。**每一步迭代中，矩阵 W 均需正交化，有两种方法**：

- 直接法

$$W = (WW^T)^{-1/2}W$$

其中，$(WW^T)^{-1/2} = U\Lambda^{-1/2}U^T$ ， $U\Lambda U^T$ 为 WW^T 特征值分解的结果。

- 迭代法：

$$1.\ \text{Let}\ \ W = W/\sqrt{\|WW^T\|}$$

$$\text{Repeat 2. until convergence:}$$

$$2.\ \text{Let}\ \ W = \frac{3}{2}W - \frac{1}{2}WW^TW$$

其中，The norm in step 1 can be almost any ordinary matrix norm, e.g., the 2-norm or the largest absolute row (or column) sum (**but not the Frobenius norm**).

最后，我们指出：

- FastICA can be considered as a fixed-point algorithm for maximum likelihood estimation of the ICA data model, for details, see (Hyvärinen, 1999b). In FastICA, convergence speed is optimized by the choice of the matrices Γ and B. Another advantage of FastICA is that it can estimate both sub- and super-gaussian independent components, which is in contrast to ordinary ML algorithms, which only work for a given class of distributions.
- The independent components can be estimated one by one, which is roughly equivalent to doing projection pursuit. This is useful in exploratory data analysis, and decreases the computational load of the method in cases where only some of the independent components need to be estimated.

4.5.6 ICA 与 投影寻踪 (Projection Pursuit)

前面我们提到，**ICA 可视为一种盲信号分离 (BSS) 方法，二者的目的相同，因此对所得结果的认识也相同。**这里我们来阐述 ICA 与投影寻踪间密切的联系，我们可以看到，**ICA 也是一种投影寻踪方法，区别在于对所得结果的认识上。因为 ICA 基于 ICA 模型 (ICA 模型的假设成立)，因此认为我们所得的结果是一个个独立的信号源，但投影寻踪没有要求 ICA 模型中的假设成立，当 ICA 模型成立时所得结果可认为是一个个独立的信号源，但 ICA 模型中的假设不成立时也是可以进行投影寻踪的，这个时候所得的结果就只是一个个我们感兴趣的投影方向，而没有独立信号源的含义了。**

投影寻踪的目的：Projection pursuit is a technique developed in statistics for finding “interesting” projections of multidimensional data. Such projections can then be used for optimal visualization of the data, and for such purposes as density estimation and regression. In basic (1-D) projection pursuit, we try to find directions such that the projections of the data in those directions have interesting distributions, i.e., display some structure.

投影寻踪的原则：It has been argued by Huber (Huber, 1985) and by Jones and Sibson (Jones and Sibson, 1987) that the Gaussian distribution is the least interesting one, and that the most interesting directions are those that show the least Gaussian distribution. This is exactly what we do to estimate the ICA model.

投影寻踪与 ICA 的关系：

- (相同点) Thus, in the general formulation, ICA can be considered a variant of projection pursuit. All the nongaussianity measures and the corresponding ICA algorithms presented here could also be called projection pursuit “indices” and algorithms. In particular, the projection pursuit allows us to tackle the situation where there are less independent components $s^{(i)}$ than original variables $x^{(i)}$ is. (**投影寻踪还能处理独立成分个数少于观测数据维度的情况，对应的就是 ICA 放松对 A 的限制，可以不是方阵的情况**，步骤如下：) Assuming that those dimensions of the space that are not spanned by the independent components are filled by gaussian noise, we see that computing the nongaussian projection pursuit directions, we effectively estimate the independent components. When all the nongaussian directions have been found, all the independent components have been estimated. Such a procedure can be interpreted as a hybrid of projection pursuit and ICA. (对于存在高斯噪声的情况，个人认为可以先使用 PCA 去噪，再进行 ICA)

- (不同点) However, it should be noted that in the formulation of projection pursuit, no data model or assumption about independent components is made. If the ICA model holds, optimizing the ICA nongaussianity measures produce independent components; if the model does not hold, then what we get are the projection pursuit directions. 也就是说，不管前面假设的 ICA 模型 (ICA 模型假设存在独立成分，可观察到的输出是独立成分的线性组合) 是否成立，我们总能进行投影追踪。若 ICA 模型成立，则我们得到的就是独立成分；若 ICA 模型不成立，那么我们得到的仅仅只是一个我们感兴趣的方向，没有独立成分的含义。

4.5.7 ICA 与 PCA 的比较

- **我们明确"PCA 假设数据服从高斯分布，因此比较适合处理高斯分布的数据；ICA 则只能处理非高斯分布的数据" 这一论断是错误的。**首先，PCA 只是假设误差服从高斯分布，而没有对数据的整体分布作任何假设。此外，ICA 也能处理高斯分布的数据，它可以将联合高斯分布分解为一些相互独立的高斯分布，也就是相互独立的信号源，但此时分解并不唯一，对真实信号源 $\boldsymbol{s}$ 作任意正交变换所得的结果均能满足要求，因此存在无穷多种解，因此这对于旨在确定真实信号源 $\boldsymbol{s}$ 的 ICA 来说是不够的。当真实信号源 $\boldsymbol{s}$ 中最多只含有一个高斯信号源时，ICA 的分解唯一，能得到真实信号源 $\boldsymbol{s}$。
- **PCA 得到的是线性无关的信号源，而 ICA 得到的是相互独立的信号源。PCA 会对所得结果按方差大小进行排序，方差越大的信号源就越重要；而 ICA 旨在对信号进行解混，认为其所得的信号源都是重要的，不会也无法 ("无法"是指没有一个评判标准) 对信号源进行排序，因此 ICA 无法用于降维。**(PCA attempts to find uncorrelated sources, whereas ICA attempts to find independent sources. PCA can also rank each source. ICA does not have this property, which makes it a poor tool for dimensionality reduction.) 当数据服从高斯分布时，线性无关与相互独立等价，因而此时 PCA 得到的也是相互独立的信号源，但正如前面所说，PCA 会对这些独立的高斯信号按其重要性进行排序，方差越大的就越重要。**可以看到，对于高斯分布的数据，ICA 有无穷多种解，而 PCA 因存在对方差的约束，解仍然是唯一的。**
- **PCA 是一种特征降维方法，旨在揭示样本的主要特征，降低学习难度；而 ICA 则是一种机器学习算法，旨在对信号进行解混。除了中心化和漂白，在进行 ICA 前，我们通常还会对数据进行 PCA 来去除高斯噪声并进行降维，以降低接下来 ICA 的求解难度。**

4.6 压缩感知

> 1. 知乎 - 压缩感知简介

前文提到的码书学习旨在对原信号进行处理以获得其稀疏表示，除了码书学习外，为了获得稀疏表示，还可以使用傅里叶变换、小波变换等方法。**压缩感知 (compressed sensing, compressive sensing, compressed sampling, CS) 则旨在根据对信号少量的观测比较精确地复原出或者说重构出原信号，前提条件是原信号本身是稀疏的 (存在很多个 0)，或原信号存在稀疏表示 (即原信号可以用稀疏基展开，稀疏基的系数就是原信号的稀疏表示；类似于对一个连续函数进行傅里叶展开，所得的离散的基函数系数就是该函数的稀疏表示)。**

> 根据奈奎斯特采样定理，要想让采样之后的数字信号完整保留原始信号中的信息，即可以精确重构原始信号，那么采样频率必须达到原始信号中最高频率的 2 倍。但压缩感知却突破了这一限制，即，在信号采样的过程中，用很少的采样点，实现了和全采样一样的效果。当然这是有条件的，即上面提到的稀疏性 (当然在具体操作时，采样方式也要发生变化，不能再是奈奎斯特采样定理中的等间距采样，而是不等间距的随机亚采样)。此外，压缩感知直接催生了人脸识别的鲁棒主成分分析和基于矩阵补全的协同过滤。

事实上，压缩感知根据少量采样恢复的都是原信号的稀疏表示，又稀疏基已知，则我们可以进一步恢复原信号；但若原信号本身就是稀疏的，那么得到的稀疏表示就是原信号，而无需使用稀疏基作进一步映射。我们可以用如下的公式来说明：

$$\boldsymbol{y} = \Phi\Psi\boldsymbol{s}$$

$$\boldsymbol{x} = \Psi\boldsymbol{s}$$

其中， m 维列向量 $\boldsymbol{y}$ 为我们对原信号进行少量 (m 次) 采样得到的采样信号； $\Phi_{m\times n}$ 为观测矩阵，显示了我们如何对长度为 n （ $n \gg m$) 的原信号 $\boldsymbol{x}$ 进行采样，以及如何组成采样后的信号，也就是随机亚采样方式； $\Psi_{n\times n}$ 为稀疏矩阵 (每一行相当于一个稀疏基)， $\boldsymbol{s}$ 为原信号 $\boldsymbol{x}$ 在稀疏基上的表示或者说坐标。此外，我们也会称 $A = \Phi\Psi$ 为传感矩阵。**压缩感知的过程就是已知 $\boldsymbol{y}, \Phi, \Psi$ ，求 $\boldsymbol{s}$ ；再由 $\boldsymbol{s}, \Psi$ 得到原信号 $\boldsymbol{x} = \Psi\boldsymbol{s}$ 。但若原信号本身就是稀疏的，则我们可以直接得到 $\boldsymbol{x}$ ，即 $\boldsymbol{x} = \boldsymbol{s}$ 。**

为了实现对原信号的重构，只有稀疏性显然不够，我们还需要对采样方式提出要求，即观测矩阵 Φ 需满足一定的条件，这个条件就是约束等距性条件 (Restricted Isometry Property, RIP)。RIP 的等价条件是观测矩阵 Φ 和稀疏基 Ψ 不相关 (incoherent，不是指我们常说的线性相关)。我们指出，独立同分布的高斯随机测量矩阵可以成为普适的压缩感知测量矩阵。

有了稀疏性和约束等距性条件，已知 $\boldsymbol{y}, \Phi, \Psi$ ，求 $\boldsymbol{s}$ 的压缩感知问题就可转化为如下的优化问题:

$$\min_{\boldsymbol{s}} \|\boldsymbol{s}\|_0$$
$$s.t. \quad \boldsymbol{y} = \Phi\Psi\boldsymbol{s}$$

上述优化问题涉及 L_0 范数最小化，是个 NP 难问题，但好在一定条件下 L_0 范数最小化与 L_0 范数最小化问题共解。由此，问题即为:

$$\min_{\boldsymbol{s}} \|\boldsymbol{s}\|_1$$
$$s.t. \quad \boldsymbol{y} = \Phi\Psi\boldsymbol{s}$$

上述优化问题的一种解法就是将其转化为 LASSO 的等价形式再通过近端梯度下降法求解。

西瓜书上还给出了一个基于部分信息恢复全部信息的例子，**那个例子中原信号本身就具有稀疏性**，而我们已知原信号中部分元素的值。通过求解一个矩阵补全 (matrix completion) 问题我们就能直接恢复原信号。
