特征工程

1、数据降维

2、特征提取

3、特征选择

一、数据降维

一、特征工程

特征工程是一个很大的概念，实在找不到合适的词，语句来描述特征工程。为了直观的学习特征工程，还是从特征工程处理的流程来窥视特征工程为何物？

1、数据的生成，这部分严格意义上说不属于特征工程范畴。因为数据都没有的话，谈何学习，但是数据即是特征，如何生成数据也是特征生成的过程。在工程上，数据的生成是非常重要的部分，也称特征构建。

2、数据预处理，缺损值，特征编码，归一化/标准化，数据清洗（异常点）

3、特征提取和选择

一般来说，特征工程之后的特征作为模型学习的输入。在特征工程中，特征提取和特征选择一般通过模型去学习，所以特征工程本身就涉及到模型。这里，笔者理解特征工程为特征表示，是对数据的一种表示。其中，数据生成和数据预处理比较泛，没有相对严格的处理方式，经验性较强。特征提取和特征选择则有非常多成熟的方法，一般来讲特征提取和特征选择是一个数据降维的过程。

二、数据降维

数据降维有以下几点好处：

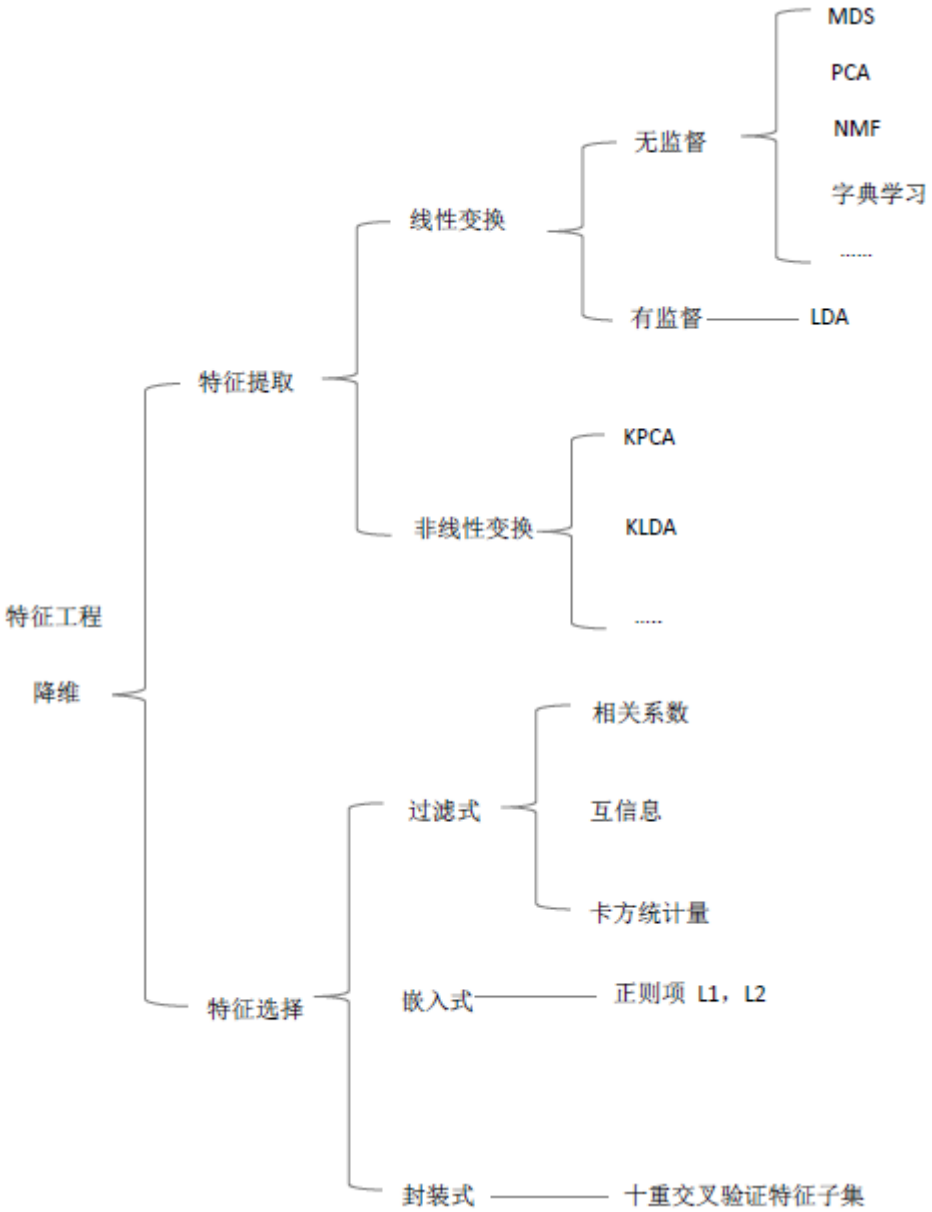
1、避免维度灾难，导致算法失效，或者时间复杂度高

2、避免高维数据中引入的噪声，防止过拟合

3、压缩存储，可视化分析

数据降维的方法有**特征提取**和**特征选择**两种方式。特征提取理论上是一种坐标变换，将原始数据特征上进行线性非线性变换到目标空间进行表示；而特征选择则是直接在原始数据特征上进行选择，选出的特征集是原始特征集的子集。特征提取的降维方法可以根据**线性**和**非线性**进行划分（非线性降维一般是在线性降维方法上加上核技巧）。特征选择的方法可以分为**过滤式**和**封装式**两种，过滤式特征选择是采用一些特征重要性的度量方式来对特征进行选择，过滤掉一些不重要的特征。封装式特征选择是采用一些优化搜索策略随机选择一些特征子集根据算法最终的性能进行特征选择。另外有一种**嵌入**在学习算法中的特征选择方法采用正则化来进行稀疏，如L1，L2范数进行正则化约束，当然正则化项的最终目标不是降维，而是使得解稀疏，也可以达到数据降维效果。

不同的数据降维方法除了实现降维目标的作用，同时具有各自的特点，比如主成分分析，降维后的各个特征在坐标上是正交；非负矩阵分解，因为在一些文本，图像领域数据要求非负性，非负矩阵分解在降维的同时保证降维后的数据均非负；字典学习，可以基于任意基向量表示，特征之间不再是独立，或者非负；局部线性嵌入，是一种典型的流型学习方法，具有在一定邻域保证样本之间的距离不变性。



一般来说，特征提取只适合数值型数据，无法直接处理非数值型属性。而特征选择则二者皆适合，比如熵就就非常适合非数值型属性，而且在树模型中，数值型属性还需要离散化处理，以便划分决策。

二、特征提取

特征提取是一个特征空间上的变换（映射），可以是线性和非线性的。所以特征提取与特征选择的不同之处在于，特征提取之后的特征已经不是原始特征了，而特征选择则是在原始的特征中选择出有价值的特征。既然，特征提取是一个空间上的映射，那么特征提取的问题就变成了选择合适的投影方向。选择合适的投影的方向就是必然有一个目标（保持损失最小），即目标函数。特征提取的方法很多，每一种方法都有因各自的目标不同而有不同的特性，下面笔者目标的形式将特征提取分为三类：**成对保持**，**单点保持** ，

一、MDS（成对保持）：

与其说Multiple Dimensional Scaling (MDS)是一种降维方法，不如理解为一种特征提取的思想。其特征提取的思想是成对保持（相似性，距离），如ISOMAP多维尺度分析与等距映射，谱哈希等。都是成对的保持数据在原始空间的关系，将数据映射到一个低维的空间。因此，为了描述样本点在原始空间的成对关系，我们需要一个相似性度量矩阵$S \in R^{m \times m}$，其中m为样本数据集大小，样本数据集$X \in R^{r \times m}$。为了使得映射到低维的空间后相似性度量矩阵的损失最小，我们可以用下式表示：

$$\min \sum_{ij} ||S_{ij} - S(Z_i, Z_j)||^2$$
$$s.t.\, ZZ^T = mI$$

其中$Z_i \in R^{d \times 1}$表示样本X_i在低维空间的表示，其中约束条件为规范化约束。$S(Z_i, Z_j)$表示样本在低维空间的相似性度量，这里我们采用简单的度量$S(Z_i, Z_j) = Z_i^T Z_j$。那么，目标函数可以简化为：

$$\min \sum_{ij} ||S_{ij} - Z_i^T Z_j||^2 \Leftrightarrow \max SZ^T Z$$
$$s.t.\, ZZ^T = mI$$

到此，目标函数已非常清晰，类似于谱聚类，把Z看作由一维一维的向量构成，那么$SZ^T ZZ^T = \lambda Z^T$，目标函数问题就是S的前d个最大特征值，Z即为S的最大特征值对应的特征向量构成的矩阵。

由SVD分解，$S = V \wedge V^T$，其中$\wedge_{ii}$为S对应的前d个最大特征值，为V特征值对应的特征向量构成的矩阵，那么有：

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

在确定投影空间的向量表示Z之后，我们在回过头来求投影向量W有：

$$Z = W^T X$$

其中$W \in R^{r \times d}$，可见MDS是一个线性变换。在成对保持的思想下，换用不同的相似性度量可以导出不同的方法，另外添加不同的约束条件，也可以导出不同的方法，如哈希学习。

二、PCA（单点保持）

PCA是最常用的一种无监督数据降维方法，既可以进行特征提取，更一般用于数据可视化分析。众所周知，PCA有两种解释，最大方差解释，最小重构损失解释，最大方差解释是一种较合理性解释，即寻找方差最大的方法最能反映样本的区分度，信息量最大（相对而言，如果数据在某一维度都等于3，那么自然方差为0，信息量无意义）。但是在机器学习中，我们最常用的目标还是损失最小，所以个人觉得第二种解释才是PCA的根本，而最大方差解释更像是形象化解释。所以，我们从最小重构损失解释推导出最大方差解释：

假设有样本数据集$X \in r \times m$，投影方向为$W = \{w_1, w_2, .w_i., w_d\}, w_i \in R^{r \times d}$，其中$w_i$是一组标准正交基，即$||w_i||^2 = 1, w_i^T w_j = 0 \Rightarrow W^T W = I$，那么在$W$坐标系下，样本的数据集表示为$Z \in R^{d \times m}$由重构损失最小化有如下目标函数：

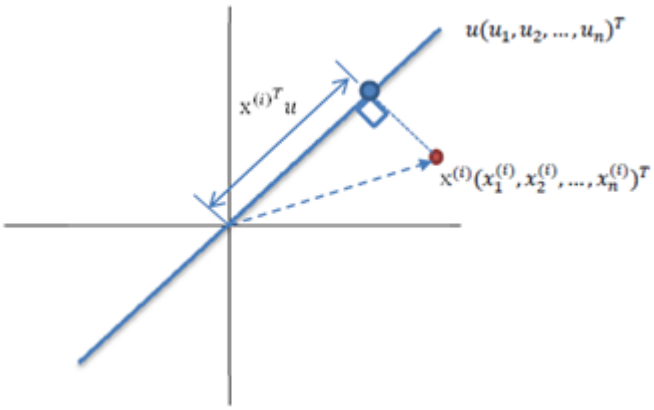
$$\sum_{i=1}^{m} ||Wz_i - x_i||^2 = \sum_{i=1}^{m} \left(||z_i^T W^T W z_i|| - 2||z_i^T W^T x_i|| + const \right)$$
$$W^T W = I, z_i = W^T x_i \Rightarrow \sum_{i=1}^{m} -||x_i^T W W^T x_i|| = -tr(X^T W W^T X)$$
$$= -tr(W^T X X^T W) = -tr(X X^T W W^T)$$

由上，目标函数又非常清晰了，把XX^T看作是度量矩阵S，那么$SWW^TW=\lambda W$，目标函数问题就是S的前d个最大特征值，W即为S的最大特征值对应的特征向量构成的矩阵。

注意：从目标函数我们发现似乎还少了什么？对，回忆线性回归，我们都会有一个b来表示偏置项。而PCA目标函数中，没有这一项，而这一项正好对应于X在各个方向上的均值（偏置项），所以在构造上诉目标函数时，需要对X去均值。

再回过头看目标函数，我们发现$\max tr(W^TXX^TW)$，其中XX^T在去均值之后就对应于样本的协方差矩阵，所以，我们从最大协方差角度来看，就是求解协方差最大的几个方向构成样本在原始空间的主成分。

结合下图，我们发现在标准正交基的约束下，最小重构损失和最大协方差方向是一致的：



上图中，样本分布于高维空间，我们选择一个超平面来表示数据，从最小重构损失，其中u超平面为主方向，与u正交的方向是我们丢弃的，那么重构损失最小，即与投影方向距离最小的（损失最小），投影方向距离最大（方差最大）。

一般而言，PCA在进行协方差矩阵求解特征值之前需要做两件事，其一去均值，其二标准化，标准化不改变原始数据的分布信息，只是对数据在方差上进行一个归一化，使得方差分布在$[0,1]$：

1. Let $\mu=\frac{1}{m}\sum_{i=1}^{m}x^{(i)}$.
2. Replace each $x^{(i)}$ with $x^{(i)}-\mu$.
3. Let $\sigma_j^2=\frac{1}{m}\sum_i(x_j^{(i)})^2$
4. Replace each $x_j^{(i)}$ with $x_j^{(i)}/\sigma_j$.

在PCA中，对协方差矩阵XX^T进行特征值求解，首先需要计算XX^T，这里的计算复杂度就是rm^2，当样本数很大时，复杂度较高。考虑到协方差的特殊性（XX^T），往往采用SVD分解来进行PCA中主成分（特征向量）的求解。

下面接着PCA来介绍SVD

奇异值分解中，若$X\in R^{r\times m}$，X的若干个正奇异值为$\sigma_1\geq\sigma_2,..,\sigma_d\geq 0.\sigma_r$，则存在酉矩阵$U\in R^{r\times r},U^TU=I$和$V\in R^{m\times m},V^TV=I$矩阵，满足：

$$X^{r\times m}=U^{r\times r}D^{r\times m}V_{m\times m}^T=U\begin{bmatrix}\Delta & 0\\ 0 & 0\end{bmatrix}V^T\approx U^{r\times d}D^{d\times d}V_{m\times d}^T$$

那么有 协方差矩阵可以表示为：

$$
\begin{aligned}
XX^T &= \left(UDV^T\right)\left(UDV^T\right)^T \\
&= UDV^TVD^TU^T \\
&= UDD^TU^T \\
\Rightarrow XX^TU &= UDD^T = DD^TU
\end{aligned}
$$

$$
XX^T \cdot \begin{bmatrix} U_1 & U_2 & U_d \end{bmatrix}_{r\times d} = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \sigma_d & \\ & & & \sigma_r \end{bmatrix}_{r\times r} \begin{bmatrix} U_1 & U_2 & U_d \end{bmatrix}_{r\times d}
$$

可知DD^T的主对角元素为协方差矩阵XX^T的特征值，左奇异矩阵U是XX^T的特征向量。所以，PCA协方差矩阵特征分解问题就可以转化为SVD分解，从而减少了协方差XX^T的计算复杂度rm^2。

回到数据降维，PCA的关键思想是保持点损失最小，加上投影方向是一组标准正交基。在保持点损失最小，加上不同的约束，我们可以得到其他的数据降维方法，从其目标函数和约束项，我们能很好地理解这一类数据降维方法的共性

非负矩阵分解：

$$
\begin{aligned}
&\min_{W,H} \frac{1}{2}||X - WZ||^2 \\
&s.t. W \geq 0, Z \geq 0
\end{aligned}
$$

稀疏编码：

$$
\min_{W,Z} ||X - WZ||^2 + \lambda||Z||^1
$$

局部线性嵌入:

$$
\begin{aligned}
&\min_{W} \sum_{i}^{m} ||X_i - \sum_{l\in N(i)} w_{kl}X_l||^2 \\
&s.t. \sum_{l\in N(i)} w_{kl} = 1
\end{aligned}
$$

典型关联分析:

$$
\begin{aligned}
&\min_{a,b} ||\frac{X^Ta}{||X^Ta||} - \frac{Y^Ta}{||Y^Ta||}||^2 \\
\Leftrightarrow &\max_{a,b} a^TXY^Tb \\
&s.t. a^TXX^Ta = 1, b^TYY^Tb = 1
\end{aligned}
$$

以上，所有的降维方法都是无监督线性映射，一般而言非线性降维方法是在线性映射之前做一个核处理.

如KPCA: 结合核技巧，在进行PCA降维之前，先进行核变换（核函数），映射之后有，$x^* = \phi(x)$，同样由于这里有XX^T，那么对于核映射之后的协方差矩阵可以用一个正定矩阵来表示：

$$
K(x_i, x_j) = \phi_{x_i}\phi_{x_j}
$$

如果直接定义核矩阵，我们对核函数ϕ是无法求解，那么对于新的样本，我们就是不能映射到核空间，所以这里的核$K(x_i,x_j)$是通过核函数$\phi(x)$计算后而得的，核的大小也是可以选择的。有了核矩阵K，那么就是对K进行特征分解：

$$KW=\lambda W$$

求解K的前d个最大特征值对应的特征向量。

三、LDA：有监督

线性判别分析中无论是分类还是降维，关键点在于**类内距离**尽可能小（类紧致），**类间距离**尽可能大（类分离）。所以如何定义类内距离和类间距离是关键：

$$\tilde{m}_i=\frac{1}{n_i}\sum_{y_j\in X_i}^{i=1,2}y_j=\frac{1}{n_i}\sum_{y_j\in X_i}^{i=1,2}w^Tx_j=w^Tm_i$$

其中$\tilde{m}_i$表示降维后的类中心，m_i表示原始空间的类中心。

类内距离：

$$\begin{aligned}\tilde{S}_w&=\sum_{y_j\in X_i}^{i=1,2}(y_j-\tilde{m}_i)^2\\&=\sum_{x_j\in X_i}^{i=1,2}\left(w^Tx_j-\frac{1}{n_i}\sum_{x_j\in X_i}^{i}w^Tx_j\right)^2\\&=w^T\sum_{x_j\in X_i}^{i=1,2}\left(x_j-\frac{1}{n_i}\sum_{x_j\in X_i}^{i}x_j\right)^2w\\&=w_TS_ww\end{aligned}$$

其中$S_w=\sum_{x_j\in X_i}^{i=1,2}\left(x_j-m_i\right)^2$表示每类样本在原始空间的一个类内距离。

类间距离：

$$\begin{aligned}\tilde{S}_b&=(\tilde{m}_1-\tilde{m}_2)^2\\&=\left(\frac{1}{n_1}\sum_{x_j\in X_1}w^Tx_j-\frac{1}{n_2}\sum_{x_j\in X_2}w^Tx_j\right)^2\\&=w^T\left(\frac{1}{n_1}\sum_{x_j\in X_1}x_j-\frac{1}{n_2}\sum_{x_j\in X_2}x_j\right)^2w\\&=w^TS_bw\end{aligned}$$

其中$S_b=\sum_{x_j\in X_i}^{i=1,2}\left(m_1-m_2\right)^2$表示每类样本在原始空间的一个类间距离。

所以为了使类内紧致，类间分离，可以最大化如下目标函数：

$$\max J(w)=\frac{\tilde{S}_b}{\tilde{S}_w}=\frac{w^TS_bw}{w^TS_ww}$$

等价于$\max(w^TS_bw)s.t.w^TS_ww=c$,由拉格朗日乘子法有

$$L(w,\lambda) = w^T S_b w - \lambda(w^T S_w w - c)$$

可以看出目标函数是关于w的二次凸规划，极值在导数为0处取到，对上式求导有$S_b w^* - \lambda S_w w^* = 0$,如果$S_w$可逆的话，即

$$(S_b - \lambda S_w)w = 0$$
$$S_w^{-1} S_b w^* = \lambda w^*$$

也就是说w是$(S_b - \lambda S_w)$的特征向量。将$S_b = (m_1 - m_2)^2$带入有

$$S_w^{-1}(m_1 - m_2)(m_1 - m_2^T)w^* = \lambda w^*$$

又$(m_1 - m_2^T)w^*$是一个标量，所以$w^* = S_w^{-1}(m_1 - m_2)$.

这里，在分类问题中 w为一个向量，即一个投影方向，将数据映射到一维线上。当然w为矩阵的话，那么就不再是将数据映射到一维空间。

三、特征选择

以上，特征提取的方法实际上都是建立在一个空间变换上，所以一般只适合数值型属性，然而现实场景中有非常多的非数值型属性，对于这一类的属性是无法直接做特征提取的（一般可编码成数值型），所以特征选择的适用场景更广泛。

一、过滤式

过滤式选择是设计一个“**统计量**”（比如和标签的相关性）来度量特征的重要性，即作用在特征x_j上对标签的区分度是不是很高。常用的统计量有：**方差** ，**相关系数**，**卡方统计量**，**互信息**（熵）

方差：计算各个特征的方差，然后根据阈值，选择方差大于阈值的特征

相关系数：计算各个特征对目标值的相关系数以及相关系数的P值 ，选择较大

卡方统计：自变量有N种取值，因变量有M种取值，考虑自变量等于i且因变量等于j的样本频数的观察值与期望的差距，构建统计量，也是在计算特征与标签的相关性

互信息：经典的互信息也是评价定性自变量对定性因变量的相关性的

二、封装式

封装式选择是选择一些特征构成特征子集，验证学习器在不同的特征子集上的评价指标，选择最佳的特征子集作为特征选择。当然，在特征全集很多时，这种方法效率低，一种贪心的方法是，每次从剩余的子集选择一个最佳的特征加入到当前特征子集下，逐步贪心求解次优的特征子集。

三、嵌入式

嵌入式选择是一种嵌入在模型中的特征选择方法，比如L1范数的稀疏编码，就是将一些不重要的特征的权重规范为0，还有树模型中的RF,GBDT,XGBoost，在构建不同的子树时，我们都会得到一个误差值，那么误差值小的子树选择的特征就可以看作是最佳特征子集。