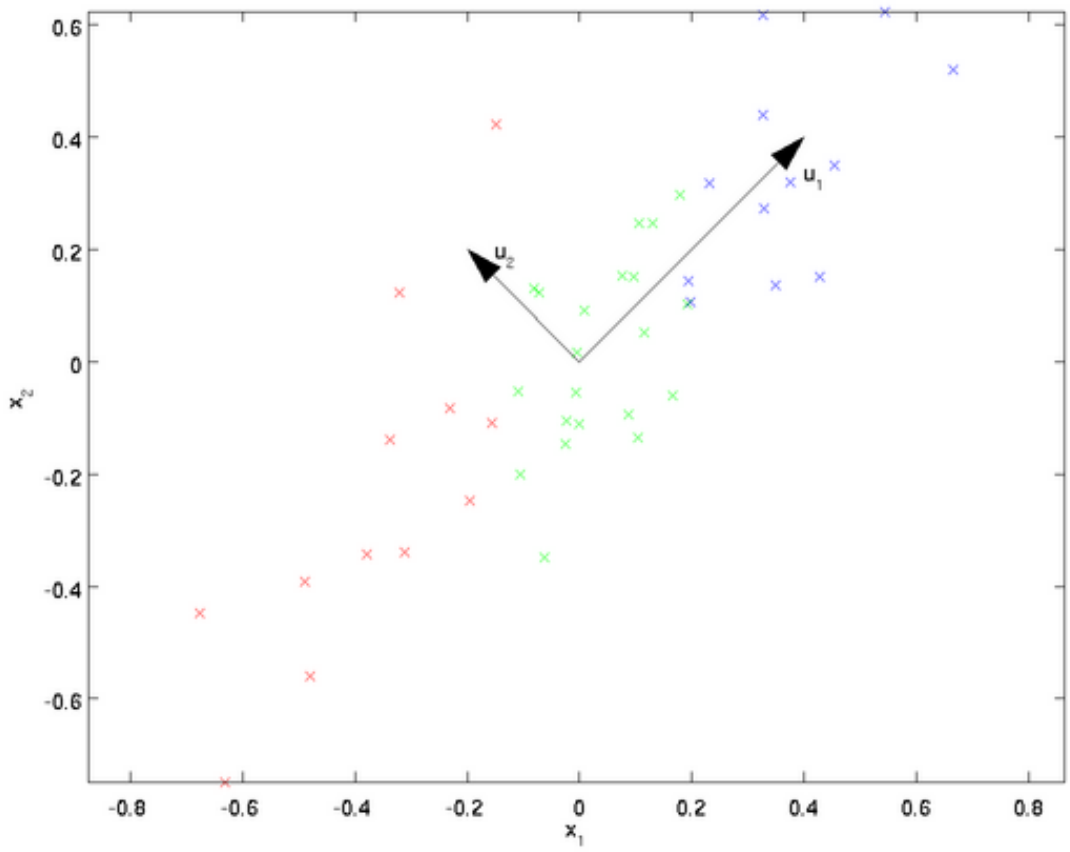
主成分分析(Principal components analysis, PCA)是最重要的降维方法之一，在数据压缩、消除冗余和数据噪音消除等方面有广泛的应用。通常我们提到降维算法，最先想到的就是PCA，下面我们对PCA原理进行介绍。

1. PCA思想

PCA就是找出数据中最主要的方面，用数据中最重要的方面来代替原始数据。假如我们的数据集是n维的，共有m个数据(x1,x2,...,xm)，我们将这m个数据从n维降到r维，希望这m个r的数据集尽可能的代表原始数据集。

我们知道从n维降到r维肯定会有损失，但是希望损失尽可能的小，那么如何让这r维的数据尽可能表示原来的数据呢？首先来看最简单的情况，即将二维数据降到一维，也就是n=2,r=1。数据如下图所示，我们希望找到某一个维度方向，它可以代表这两个维度的数据。图中列了两个向量，也就是u1和u2，那么哪个向量可以更好的代表原始数据集呢?



直观上看u1比u2更好，为什么呢？可以有两种解释，第一种解释是样本点在这个直线上的投影尽可能的分开，第二种解释是样本点到这个直线的距离足够近。假如我们把r从1维推广到任意维，则我们希望降维的标准为样本点在这个超平面上的投影尽可能分开，或者说样本点到这个超平面的距离足够近。基于上面的两种标准，我们可以得到PCA的两种等价推导。

2. PCA推导:基于最大投影方差

2.1 基变换

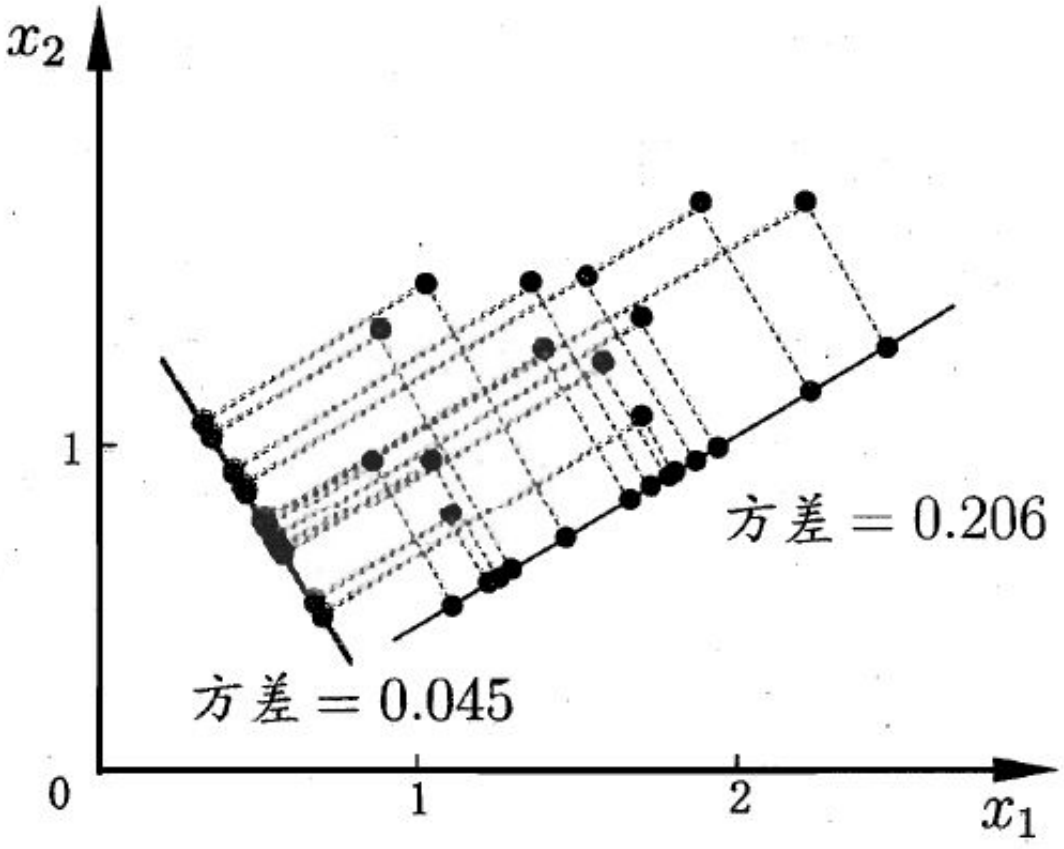
一般来说，想要获得原始数据的表示空间，最简单的方式是对原始数据进行线性变换(基变换)，即Y=PX。其中Y是样本在新空间的表达，P是基向量，X是原始样本。我们可知选择不同的基能够对一组数据给出不同的表示，同时当基的数量少于原始样本本身的维数时，则可以达到降维的效果，矩阵表示如下

$$\begin{pmatrix} p_1 \\ p_2 \\ \cdot \\ \cdot \\ \cdot \\ p_r \end{pmatrix} \begin{pmatrix} x^{(1)} & x^{(2)} & \dots & x^{(m)} \end{pmatrix} = \begin{pmatrix} p_1 x^{(1)} & p_1 x^{(2)} & \dots & p_1 x^{(m)} \\ p_2 x^{(1)} & p_2 x^{(2)} & \dots & p_2 x^{(m)} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ p_r x^{(1)} & p_r x^{(2)} & \dots & p_r x^{(m)} \end{pmatrix}$$

其中$p_i \in \{p_1, p_2, \dots, p_r\}$， $p_i \in \mathbb{R}^{1*n}$是一个行向量，表示第i个基。

$x^{(j)} \in \{x^{(1)}, x^{(2)}, \dots x^{(m)}\}$，$x^{(j)} \in \mathbb{R}^{n*1}$是一个列向量，表示第j个原始数据。我们可以将一个n维数据变换到更低维度的空间中去，即从原来的n维降到r维，变换后的维度取决于基的数量。

2.2 方差



那么考虑，如何选择一个方向或者基才是最优的呢? 观察上图，我们将所有的点分别向两条直线做投影，基于前面PCA最大可分的思想，我们要找的方向是降维后损失最小，可以理解为投影后的数据尽可能的分开。那么这种分散程度可以用数学上的方差进行表示，方差越大数据越分散，方差公式如下所示

$$Var(x) = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu)^2$$

对数据进行中心化后得到

$$Var(x) = \frac{1}{m}\sum_{i=1}^{m}(x_i)^2$$

现在我们知道以下几点

- 对原始数据进行基变换可以对原始数据给出不同表示。
- 基的维度小于数据的维度可以起到降维的效果。
- 对基变换后新的样本进行求方差，选择使其方差最大的基。

2.3 协方差

基于上面提到的几点，我们来探讨如何寻找计算方案。从上面可以得到，二维降到一维可以使用方差最大，来选出能使基变换后数据分散最大的方向(基)，但如果遇到高维的变换，怎么办呢?

针对高维情况，数学上采用协方差来表示

$$Cov(X,Y) = E((X-\mu)(Y-\nu)) = E(X\cdot Y) - \mu\nu$$

例如，二维已中心化数据$(x^{(1)}, x^{(2)})$的协方差为

$$Cov(x^{(1)}, x^{(2)}) = E(x^{(1)}\cdot x^{(2)}) = \frac{1}{m}\sum_{i=1}^{m} x_i^{(1)} x_i^{(2)}$$

当$Cov(x^{(1)}, x^{(2)}) = 0$时，表示两个字段完全独立，这也就是我们的**优化目标**。

2.4 协方差矩阵

假如只有$(x^{(1)}, x^{(2)})$两组数据，那么我们将$(x^{(1)}, x^{(2)})$按行组成矩阵X，表示如下

$$X = \begin{pmatrix} x_1^{(1)} & x_2^{(1)} & \dots & x_m^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \dots & x_m^{(2)} \end{pmatrix}$$

然后我们用X乘以X的转置，并乘上系数$\frac{1}{m}$，可以得到

$$\frac{1}{m}XX^T = \begin{pmatrix} \frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(1)} & \frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(2)} \\ \frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(2)} & \frac{1}{m}\sum_{i=1}^{m} x^{(2)}x^{(2)} \end{pmatrix}$$

可见，协方差矩阵是一个对称的矩阵，而且主对角线是各个维度的方差，其他元素是$x^{(1)}$和$x^{(2)}$的协方差。

2.5 协方差矩阵对角化

我们的目标是使$\frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(2)} = 0$，根据上面推导，可以得到优化目标$C = \frac{1}{m}XX^T$等价于协方差矩阵对角化。即除对角线外的其他元素(如$\frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(2)} = 0$)化为0，并且在对角线上将元素按照大小从上到下排列，这样就达到了优化的目的。

我们来看看原数据协方差矩阵和通过基变换后的协方差矩阵之间的关系。设原数据协方差矩阵为C，P是一组基按行组成的矩阵，设Y=PX，则Y为X对P做基变换后的数据。设Y的协方差矩阵为D，我们来推导一下D和C的关系

$$
\begin{aligned}
D &= \frac{1}{m}YY^T \\
&= \frac{1}{m}(PX)(PX)^T \\
&= \frac{1}{m}PXX^TP^T \\
&= P(\frac{1}{m}XX^T)P^T \\
&= PCP^T \\
&= P\begin{pmatrix} \frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(1)} & \frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(2)} \\ \frac{1}{m}\sum_{i=1}^{m} x^{(1)}x^{(2)} & \frac{1}{m}\sum_{i=1}^{m} x^{(2)}x^{(2)} \end{pmatrix}P^T
\end{aligned}
$$

可以看出，我们的目标是寻找能够让原始协方差矩阵对角化的P。换句话说，优化目标变成了寻找一个矩阵P，满足PCP^T是一个对角矩阵。并且对角元素按照从大到小依次排列，那么P的前k行就是要寻找的基，用P的前k行组成的矩阵乘以X就使得X从n维降到了r维。

我们希望投影后的方差最大化，于是优化目标为

$$
\underbrace{argmax}_{P}\ tr(PCP^T)\quad s.t.\ PP^T = I
$$

其中tr表示矩阵的迹，利用拉格朗日函数可以得到

$$
J(P) = tr(PCP^T) + \lambda(PP^T - I)
$$

对P进行求导，整理得到

$$
\begin{aligned}
CP^T + \lambda P^T &= 0 \\
CP^T &= (-\lambda)P^T
\end{aligned}
$$

于是，只需要对协方差矩阵C进行特征分解，对求得的特征值进行排序，再对$P^T = (p_1, p_2, \ldots, p_r)$取前k列组成的矩阵乘以原始数据矩阵X，就得到我们需要的降维后的数据矩阵Y。

3. PCA推导:基于最小投影距离

假设m个n维数据$(x^{(1)}, x^{(2)}, \ldots, x^{(m)})$都已经进行中心化，即$\sum_{i=1}^{m} x^{(i)} = 0$。经过投影变化后得到的坐标系为$\{p_1, p_2, \ldots, p_n\}$，降维后的坐标系为$\{p_1, p_2, \ldots, p_r\}$。样本点$x^{(i)}$在r维坐标系中的投影为$y^{(i)}$，$y^{(i)} = (y_1^{(i)}, y_2^{(i)}, \ldots, y_r^{(i)})$，其中$y_j^{(i)}$是$x^{(i)}$在低维坐标系中第j维的坐标，$y_j^{(i)} = p_j x^{(i)}$，即Y=PX。

如果我们用$y^{(i)}$来恢复到原始数据$x^{(i)}$，则得到的恢复数据$\bar{x}^i = \sum_{j=1}^{r} y_j^{(i)} p_j^T = P^T y^{(i)}$，其中P为标准正交基组成的矩阵。现在我们考虑整个样本集，我们希望所有的样本到这个超平面的矩阵足够近，即最小化下式

$$\sum_{i=1}^{m}||\bar{x}^{(i)}-x^{(i)}||_2^2$$

$$=\sum_{i=1}^{m}||P^Ty^{(i)}-x^{(i)}||_2^2$$

$$=\sum_{i=1}^{m}(P^Ty^{(i)})^T(P^Ty^{(i)})-2\sum_{i=1}^{m}(P^Ty^{(i)})^Tx^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)}$$

$$=\sum_{i=1}^{m}y^{(i)T}y^{(i)}-2\sum_{i=1}^{m}y^{(i)T}Px^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)}$$

$$=\sum_{i=1}^{m}y^{(i)T}y^{(i)}-2\sum_{i=1}^{m}y^{(i)T}y^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)}$$

$$=-\sum_{i=1}^{m}y^{(i)T}y^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)}$$

$$=-tr(P(\sum_{i=1}^{m}x^{(i)}x^{(i)T})P^T)+\sum_{i=1}^{m}x^{(i)T}x^{(i)}$$

$$=-tr(PXX^TP^T)+\sum_{i=1}^{m}x^{(i)T}x^{(i)}$$

注意到$\sum_{i=1}^{m}x^{(i)}x^{(i)T}$是数据集的协方差矩阵，设为C。最小化上式等价于

$$\underbrace{argmin}_{W}-tr(PXX^TP^T)\quad s.t.\,PP^T=I$$

$$\underbrace{argmin}_{W}-tr(PCP^T)\quad s.t.\,PP^T=I$$

可以发现，和第二节基于最大投影方差的优化目标完全一样。只是上述计算的是加负号的最小化，现在计算的是无负号最大化。然后利用拉格朗日函数可以得到

$$J(P)=-tr(PCP^T)+\lambda(PP^T-I)$$

对P求导有

$$-CP^T+\lambda P^T=0$$
$$CP^T=\lambda P^T$$

然后，只需要对协方差矩阵C进行特征分解，对求得的特征值进行排序，再对$P^T=(p_1,p_2,\dots,p_r)$取前k列组成的矩阵乘以原始数据矩阵X，就得到我们需要的降维后的数据矩阵Y。

4. PCA算法流程

从上面可以看出，样本x_i的r维主成分其实就是求样本集的协方差矩阵$\frac{1}{m}XX^T$的前r个特征值对应特征向量矩阵P。然后对每个样本x_i，做$y_i=Px_i$变化，即达到PCA的目的。下面我们看看具体的算法流程

输入：n维样本集$X=(x^{(1)},x^{(2)},\dots x^{(m)})$，要降维到的维数为r。

输出：降温后的样本集Y。

- 对所有样本进行中心化$x^{(i)}=x^{(i)}-\sum_{j=1}^{m}x^{(j)}$。
- 计算样本的协方差矩阵$C=\frac{1}{m}XX^T$。

- 求出协方差矩阵的特征值及对应的特征向量。
- 将特征向量按对应特征值大小从上到下按行排列成矩阵，取前k行组成矩阵P。
- Y=PX即为降维到k维后的数据。

注意：有时候，我们不指定降维后r的值，而是换种方式，指定一个降维到的主成分比重阈值t，t在[0,1]之间。假如我们的n个特征值为$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$，则r可以通过下式得到

$$\frac{\sum_{i=1}^{r} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \geq t$$

5. 核主成分分析KPCA

在上面的PCA算法中，我们假设存在一个线性的超平面，可以让我们对数据进行投影。但是有些时候，数据不是线性的，不能直接进行PCA降维。这里便需要利用和支持向量机一样的核函数思想，先把数据集从n维映射到线性可分的高维N，其中N>n，然后再从N维降维到一个低维度r，这里维度之间满足r<n<N。

使用核函数的主成分分析称为核主成分分析(Kernelized PCA, KPCA)。假设高维空间的数据由n维空间的数据通过映射ϕ产生。则对于n维空间的特征分解

$$\sum_{i=1}^{m} x^{(i)} x^{(i)T} P = \lambda P$$

映射为

$$\sum_{i=1}^{m} \phi(x^{(i)}) \phi(x^{(i)})^T P = \lambda P$$

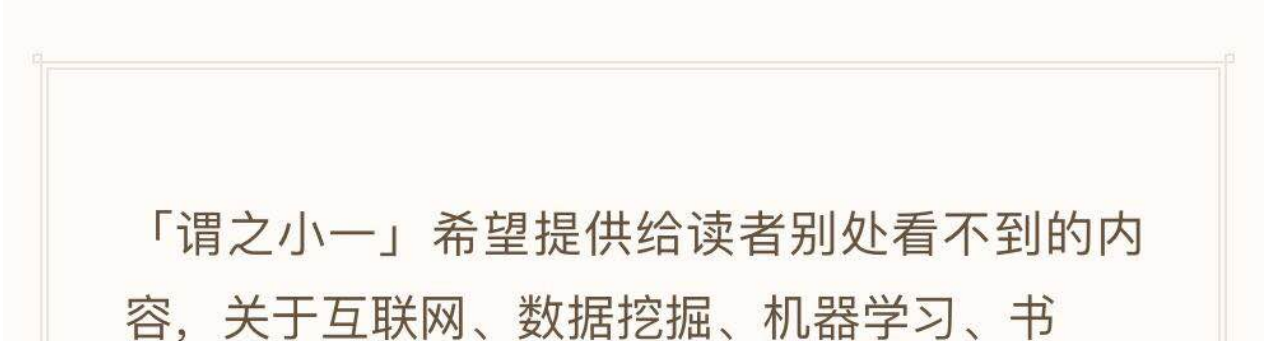
通过在高维空间进行协方差的特征值分解，然后用和PCA一样的方法进行降维。一般来说，映射ϕ不用显式的计算，而是在需要计算的时候通过核函数完成。由于KPCA需要核函数的运算，因此它的计算量要比PCA大很多。

6. PCA算法总结

作为一个非监督学习的降维方法，PCA只需要特征值分解，就可以对数据进行压缩，去噪。因此在实际场景应用很广泛。为了克服PCA的一些缺点，出现了很多PCA的变种，比如解决非线性降维的KPCA，还有解决内存限制的增量PCA方法Incremental PCA，以及解决稀疏数据降维的PCA方法Sparse PCA等。

7. 推广

更多内容请关注公众号**谓之小一**，若有疑问可在公众号后台提问，随时回答，欢迎关注，内容转载请注明出处。



籍、生活……

- 知乎：@谓之小一
- 公众号：@谓之小一
- GitHub：@weizhixiaoyi
- 技术博客：https://weizhixiaoyi.com

长按关注微信公众号

由锤子便签发送 via Smartisan Notes

引用

> [刘建平Pinard-主成分分析(PCA)原理总结]
>
> [鱼遇雨欲语与余-PCA主成分分析学习总结]