主成份分析PCA

# 降维的必要性

1. 多重共线性--预测变量之间相互关联。多重共线性会导致解空间的不稳定，从而可能导致结果的不连贯。
2. 高维空间本身具有稀疏性。一维正态分布有68%的值落于正负标准差之间，而在十维空间上只有0.02%。
3. 过多的变量会妨碍查找规律的建立。
4. 仅在变量层面上分析可能会忽略变量之间的潜在联系。例如几个预测变量可能落入仅反映数据某一方面特征的一个组内。

# 降维的目的

1. 减少预测变量的个数
2. 确保这些变量是相互独立的
3. 提供一个框架来解释结果

降维的方法有：主成分分析、因子分析、用户自定义复合等。

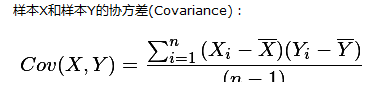
# PCA（Principal Component Analysis）

PCA（Principal Component Analysis）不仅仅是对高维数据进行降维，更重要的是经过降维去除了噪声，发现了数据中的模式。

PCA把原先的n个特征用数目更少的m个特征取代，新特征是旧特征的线性组合，这些线性组合最大化样本方差，尽量使新的m个特征互不相关。从旧特征到新特征的映射捕获数据中的固有变异性。

# 准备知识

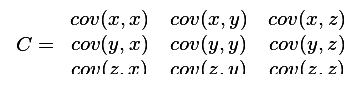
## 协方差



协方差为正时说明X和Y是正相关关系，协方差为负时X和Y是负相关关系，协方差为0时X和Y不是线性相关

Cov(X,X)就是X的方差(Variance).

当样本是n维数据时，它们的协方差实际上是协方差矩阵（对称方阵）。比如对于3维数据(x,y,z)，计算它的协方差就是：



## 特征值与特征向量

设是A是一个n阶方阵，λ是一个数，如果方程

AX = λX （1）

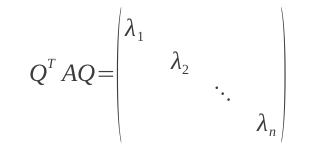
即

（A - λE）X = 0 (2)

存在非零解向量，则称为λ为A的一个特征值，相应的非零解向量X称为属于特征值λ的特征向量。

实际上可以这样理解：矩阵A作用在它的特征向量X上，仅仅使得X的长度发生了变化，缩放比例就是相应的特征值

当A是n阶可逆矩阵时，A与P-1Ap相似，相似矩阵具有相同的特征值。特别地，当A是对称矩阵时，A的奇异值等于A的特征值，存在正交矩阵Q（Q-1=QT），使得：



对A进行[奇异值分解](http://www.cnblogs.com/zhangchaoyang/articles/2575948.html)就能求出所有特征值和Q矩阵

AQ = QD

D是由特征值组成的对角矩阵, 由特征值和特征向量的定义知，Q的列向量就是A的特征向量

# **PCA的基本原理**

PCA的原理就是将原来的样本数据投影到一个新的空间中，相当于我们在矩阵分析里面学习的将一组矩阵映射到另外的坐标系下。通过一个转换坐标，也可以理解成把一组坐标转换到另外一组坐标系下，但是在新的坐标系下，表示原来的原本不需要那么多的变量，只需要原来样本的最大的一个线性无关组的特征值对应的空间的坐标即可。

一般来说，PCA降维后的每个样本的特征的维数，不会超过训练样本的个数，因为超出的特征是没有意义的。

# **PCA过程**

1. 特征中心化,即将原始矩阵A的每一维的数据都减去该维的均值，得到矩阵B。

这里的“维”指的就是一个特征（或属性），变换之后每一维的均值都变成了0。

1. 计算B的协方差矩阵C
2. 计算协方差矩阵C的特征值和特征向量
3. 选取大的特征值对应的特征向量，得到新的数据集。

---- 得到转换矩阵，同时用于测试集、验证集

举例：

1. 对于一个训练集，100个对象模板，特征是10维，那么它可以建立一个100\*10的矩阵，作为样本。
2. 求这个样本的协方差矩阵，得到一个10\*10的协方差矩阵，然后求出这个协方差矩阵的特征值和特征向量，应该有10个特征值和特征向量，
3. 我们根据特征值的大小，取前四个特征值所对应的特征向量，构成一个10\*4的矩阵，这个矩阵就是我们要求的特征矩阵，
4. 100\*10的样本矩阵乘以这个10\*4的特征矩阵，就得到了一个100\*4的新的降维之后的样本矩阵，每个特征的维数下降了。
5. 当给定一个测试的特征集之后，比如1\*10维的特征，乘以上面得到的10\*4的特征矩阵，便可以得到一个1\*4的特征，用这个特征去分类。

# R语言计算

library("e1071")

#读取数据

wineData=read.table("E:\\research in progress\\百度云同步盘\\blog\\特征值和特征向量\\data.csv",header=T,sep=",");

#计算协方差阵

covariance = cov(wineData[2:14])

#计算特征值和特征向量

eigenResult=eigen(covariance)

#选取6个主成分,并计算这6个主成分解释的方差总和

PC\_NUM = 6

varSum=sum(eigenResult$values[1:PC\_NUM])/sum(eigenResult$values)

#降维后的样本

ruduceData= data.matrix(wineData[2:14])%\*%eigenResult$vectors[,1:PC\_NUM]

#加入分类标签

#finalData=cbind(wineData$class,ruduceData)

#给finalData添加列名

#colnames(finalDat) =c("calss","pc1","pc2","pc3","pc4","pc5","pc6")

#训练样本--主成分分析后的样本作为训练样本

y=wineData$class;

x1=ruduceData;

model1 <- svm(x1, y,cross=10)

pred1 <- predict(model1, x1)

#pred1 <- fitted(model1)

table(pred1, y) #使用table来查看预测结果

#训练样本--原数据作为训练样本

x2=wineData[2:14]

model2 <- svm(x2, y,cross=10)

#pred2 <- predict(model2, x2)

pred2 <- fitted(model2)

table(pred2, y) #使用table来查看预测结果