Author: 陆星宇 Date: 2021/7/26

第16章 主成分分析

一、交流讨论

问题1：我们用什么来衡量降维过程中的信息损失?

由于方差和不变，主成分方差和与原方差和只差可以用来衡量信息损失。

问题2：负荷量的实际含义是什么?

原变量某一特征在新变量某一特征中的占比。

二、内容概要

2.1 基本思想

主成分分析中，首先对给定数据进行规范化，使得数据每一变量的平均值为 0 ，方差为 1 。之后对数据进行正交变换，原来由线性相关变量表示的数据，通过正交变换变成由若干个线性无关的新变量表示的数据。主成分分析选择方差最大的方向(第一主成分)作为新坐标系的第一坐标轴，之后选择与第一坐标轴正交，且方差次之的方向(第二主成 分)作为新坐标系的第二坐标轴，以此类推，不断拓展主成分。

2.2 定义

给定一个线性变换，如果它们满足下列条件:

1. 系数向量α_i^T是单位向量，即 $\alpha_i^T \alpha_i = 1, i = 1, 2, \ldots, m$;
2. 变量 y_i 与 y_j 互不相关，即 $cov(y_i,\ y_j) = 0 (i \neq j)$;
3. 变量 y_1 的 x 的所有线性变换中方差最大的， y_2 是与 y_1 不相关的 x 的所有线性变换中方差最大的；一般地 y_i 是与 $y_1, y_2, \ldots, y_{i-1} (i = 1, 2, \ldots, m)$ 都不相关的 x 的所有线性变换中方差最大的；这时分别称 $y_1, y_2, \ldots, y_m$ 为 x 的第一主成分、 第二主成分、... 、第 m 主成分。

> 随机变量不相关的定义：
>
> 设 y_1 与 y_2 不相关，则 $cov(y_1, y_2) = 0$

2.3 性质

定理1（总体主成分与协方差矩阵的关系）

设 x 是 m 维随机变量，Σ 是 x 的协方差矩阵，Σ 的特征值分别是 $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m \geq 0$，特征值对应的单位特征向量分别是 $\alpha_1, \alpha_2, \ldots, \alpha_m$，则 x 的第 k 主成分是

$$y_k = \alpha_k^T x = \alpha_{1k} x_1 + \alpha_{2k} x_1 + \ldots + \alpha_{mk} x_m,\ \ k = 1, 2, \ldots, m$$

x 的第 k 主成分的方差是

$$var(y_k) = \alpha_k^T \Sigma \alpha_k = \lambda_k,\ k = 1, 2, \ldots, m$$

即协方差矩阵 Σ 的第 k 个特征值。

总体主成分的性质

1. $cov(y) = diag(\lambda_1, \lambda_2, \ldots, \lambda_m)$
2. $\sum_{i=1}^m \lambda_i = \sum_{i=1}^m \sigma_{ii}$，其中 σ_{ii} 是 x_i 的方差

3. 因子负荷量 $\rho(y_k, x_i) = \frac{\sqrt{\lambda_k}\alpha_{ik}}{\sqrt{\sigma_{ii}}}$，$k, i = 1, 2, \ldots, m$
4. $\sum_{i=1}^{m} \sigma_{ii}\rho^2(y_k, x_i) = \lambda_k$
5. $\sum_{k=1}^{m} \rho^2(y_k, x_i) = 1$

2.4 主成分个数

一般认为方差越大则随机变量所含信息越大，若要选择 k 个主成分，则选前 k 个主成分最优，因其方差和最大。

2.5 规范化变量

总体主成分

$$x_i^* = \frac{x_i - E(x_i)}{\sqrt{var(x_i)}},\ \ i = 1, 2, \ldots, m$$

样本主成分

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_j$$

$$S = [s_{ij}]_{m\times m}$$

$$s_{ij} = \frac{1}{n-1}\sum_{k=1}^{n}(x_{ik} - \overline{x}_i)(x_{jk} - \overline{x}_j),\ \ i, j = 1, 2, \ldots, m$$

$$x_i^* = \frac{x_i - \overline{x}_i}{\sqrt{s_{ii}}}$$

三、算法实现

```
import numpy as np

def pca(X0, k):
    X1 = np.array(X0)
    n = X1.shape(0)
    m = X1.shape(1)
    avg = np.mean(X1, axis = 0)
    s = np.var(X1, axis = 0, ddof = 1)
    X1 = (X1 - avg)/np.sqrt(s)
    lam, A = np.linalg.eig(X1)
    tmp = lam.copy()
    order = [-1 for i in range(len(tmp))]
    for i in range(len(tmp)):
        j = tmp.index(max(tmp))
        order[i] = j
        tmp[j] = -1
    lamk = [lam[i] for i in order ]
    Ak = [A[i] for i in order]
    return lamk, Ak
```

四、下周计划

阅读《统计学习方法》第十七章