

主成分分析PCA

PCA作用

1. 无用的特征

前面讲过一节模型选择和正则化，那一节中我们解决了这个问题，去除无用特征

2. 有多余的特征

再举一个例子，有一个数据集，其中的 $x_1(i)$ 指代的是飞行员 i 的飞行技能的度量，而 $x_2(i)$ 指代的是该飞行员对飞行的喜爱程度。无线电遥控直升机是很难操作的，只有那些非常投入，并且特别热爱飞行的学生，才能成为好的飞行员。所以，上面这两个属性 x_1 和 x_2 之间的相关性是非常强的，所以我们可以认为在数据中沿着对角线方向（也就是下图中的 u_1 方向）表征了一个人对飞行投入程度的内在“源动力（karma）”，只有少量的噪音脱离这个对角线方向。如下图所示，我们怎么自动去计算出 u_1 的方向呢？

3. 特征数>样本数

上述的第一个问题已经解决，而第二第三个问题可以通过PCA来解决。

PCA就是用来进行特征降维的，减少噪音和冗余，减少过度拟合的可能性。

PCA 的思想是将 n 维特征映射到 k 维上 ($k < n$)，这 k 维是全新的正交特征。这 k 维特征称为主元，是重新构造出来的 k 维特征，而不是简单地从 n 维特征中去掉其余 $n - k$ 维特征。

PCA算法过程详解

摘自puluotianyi的中文笔记（10）主成分分析，大佬写的非常棒
网址：<https://github.com/zhouc97/CS229/tree/master/4.%E7%A8%AC94%E8%AEB0/puluotianyi%E7%9A%84%E4%B8%AD%E6%96%87%E7%AC%94%E8%AEB0>

假设我们得到了二维数据集，共10个数据，每个数据由两个特征

| x | y |
|-----|-----|
| 2.5 | 2.4 |
| 0.5 | 0.7 |
| 2.2 | 2.9 |
| 1.9 | 2.2 |
| 3.1 | 3.0 |
| 2.3 | 2.7 |
| 2 | 1.6 |
| 1 | 1.1 |
| 1.5 | 1.6 |
| 1.1 | 0.9 |

我们可以认为x是10篇文档中“learn”的TF-IDF值，y是“study”的TF-IDF值。或者可以认为x是10辆汽车的“千米/小时”速度，y是“英里/小时”速度等等。总之x和y强相关，并且是作用极其类似的两个特征，可以用PCA将其降维

将均值设为0.即求出均值，然后每个数据减去均值。这里 x 的均值是 1.81，y 的均值是 1.91

| x | y |
|-------|-------|
| .69 | .49 |
| -1.31 | -1.21 |
| .39 | .99 |
| .09 | .29 |
| 1.29 | 1.09 |
| .49 | .79 |
| .19 | -.31 |
| -.81 | -.81 |
| -.31 | -.31 |
| -.71 | -1.01 |

这里应该还有一步求方差，为了省事，我们忽略这一步

求出特征的协方差矩阵

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

分别为 $cov(x, x)$ $cov(x, y)$
 $cov(y, x)$ $cov(y, y)$

协方差大于0表示x,y其中一个增加，则另一个也增加

求出协方差的特征值、特征向量

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

0.49特征值对应特征向量为(-.735, -.677)T

将特征值按大到小排序，选出最大的k个，以及这k个特征值对应的特征向量作为列向量组成的特征矩阵

我们这里选择1.284以及特征向量(.677873399, -.735178656)

将样本点投影到选取的特征向量上（即降维步骤）

把原n个特征降维到k个新特征

$$FinalData(m * k) = DataAdjust(m * n) \times EigenVectors(n * k)$$

这里是
 $FinalData[10*1] = DataAdjust[10*2] \times$ 特征向量 $(-.735178656, .677873399)^T$
得到结果是

| Transformed Data (Single eigenvector) |
|---------------------------------------|
| -.827970186 |
| 1.77758033 |
| -.992197494 |
| -.274210416 |
| -.167580142 |
| -.912949103 |
| .0991094375 |
| 1.14457216 |
| .438046137 |
| 1.22382056 |

这样就得到了“learn”和“study”融合起来的一个新的特征

上述过程图示

如果k=2

为什么求协方差的特征向量就是最理想的 k 维向量？下面PCA算法给出解释

Principal components analysis

PCA算法

1. 预处理（正则化）

(1) (2) 步设数据的均值为0，对于本身均值就为0的数据可以省略掉这两步

(3) (4) 步对坐标进行缩放使之具有单位方差 (unit variance)，以确保不同的属性可以在相同的“规模”上进行处理

一种方法是找出一个单位向量 (unit vector) u ，并使得数据在 u 上的投影的方差最大

比如数据集中有5个数据

假设u的朝向如图，我们可以看到投影后的数据仍然有很大的方差

反例如下（方差很小）

我们想要找到使得投影后的数据方差很大的单位向量 u

2. 预处理完毕，接下来应该要找出数据大致的朝向，也就是变化的主轴 (main axis of variation) u 。

注意：给定一个向量 u 和数据 x ， x 在 u 上的投影的长度可以用 $x^T u$ 来表示

也就是说，若 $x^{(i)}$ 是我们数据集的一个点，则这个点在 u 上的投影（肯定也是一个点）就是从原点到 $x^T u$ 的距离

因此，要最大化投影的方差（使点的投影离原点最远），就要最大化下列公式

$$\frac{1}{m} \sum_{i=1}^m (x^{(i)T} u)^2 = \frac{1}{m} \sum_{i=1}^m u^T x^{(i)} x^{(i)T} u$$
$$= u^T \left(\frac{1}{m} \sum_{i=1}^m x^{(i)} x^{(i)T} \right) u$$

注：两个概念

主特征值：矩阵的特征值中，模最大的那个（实数矩阵中就是绝对值最大的那个）

主特征向量：主特征值对应的特征向量

用 λ 来表示 $\frac{1}{m} \sum_{i=1}^m (x^{(i)T} u)^2$ ， Σ 表示 $\frac{1}{m} \sum_{i=1}^m x^{(i)} x^{(i)T}$

我们将上述公式记为 $\lambda = u^T \Sigma u$

因为 u 是单位向量，所以 $\|u\| = 1$ ，即 $u^T u = uu^T = 1$

上述公式两边左乘 u

$$u \lambda = \lambda u = u u^T \Sigma u = \Sigma u$$

即 $\Sigma u = \lambda u$

我们发现，原始公式左边那一坨就是 u 的特征值，而右边括号内的一坨就是 u 的特征向量。

我们的问题就变成了，找出协方差矩阵 Σ 的最大特征向量！（主特征向量）

简单总结，如果我们要找1维度的子空间来近似拟合数据，就要选择 Σ 表示 $\frac{1}{m} \sum_{i=1}^m x^{(i)} x^{(i)T}$ 的主向量。如果要将数据投影到 k 维度的子空间中，就要找前 k 大的特征值对应的特征向量组成数据新的正交基

如果用新的正交基来表示 $x^{(i)}$ ，只需要计算

$$y^{(i)} = \begin{bmatrix} u_1^T x^{(i)} \\ u_2^T x^{(i)} \\ \vdots \\ u_k^T x^{(i)} \end{bmatrix} \in \mathbb{R}^k$$

单位向量 u_1, \dots, u_k 叫做数据集的前 k 个主成分 (principal components)