EMPCA

EM Algorithms for PCA

夏山川 2011150140

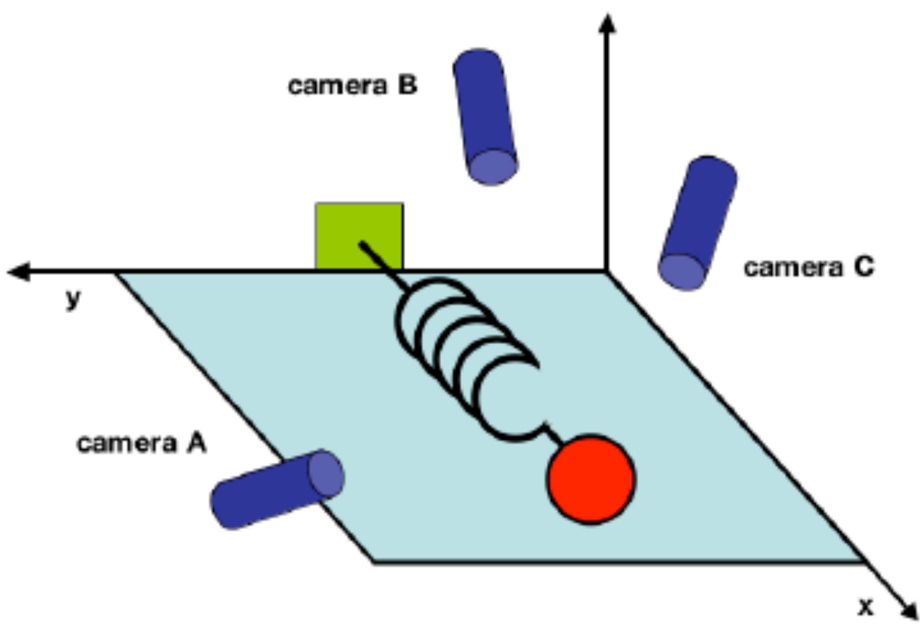
PCA算法

PCA是Principal component analysis的缩写，中文翻译为主元分析。

这种方法可以有效的找出数据中最“主要”的元素和结构，去除噪音和冗余，将原有的复杂数据降维，揭示隐藏在复杂数据背后的简单结构。

它的优点是简单，而且无参数限制，可以方便的应用与各个场合。因此应用极其广泛，从神经科学到计算机图形学都有它的用武之地。

简单模型



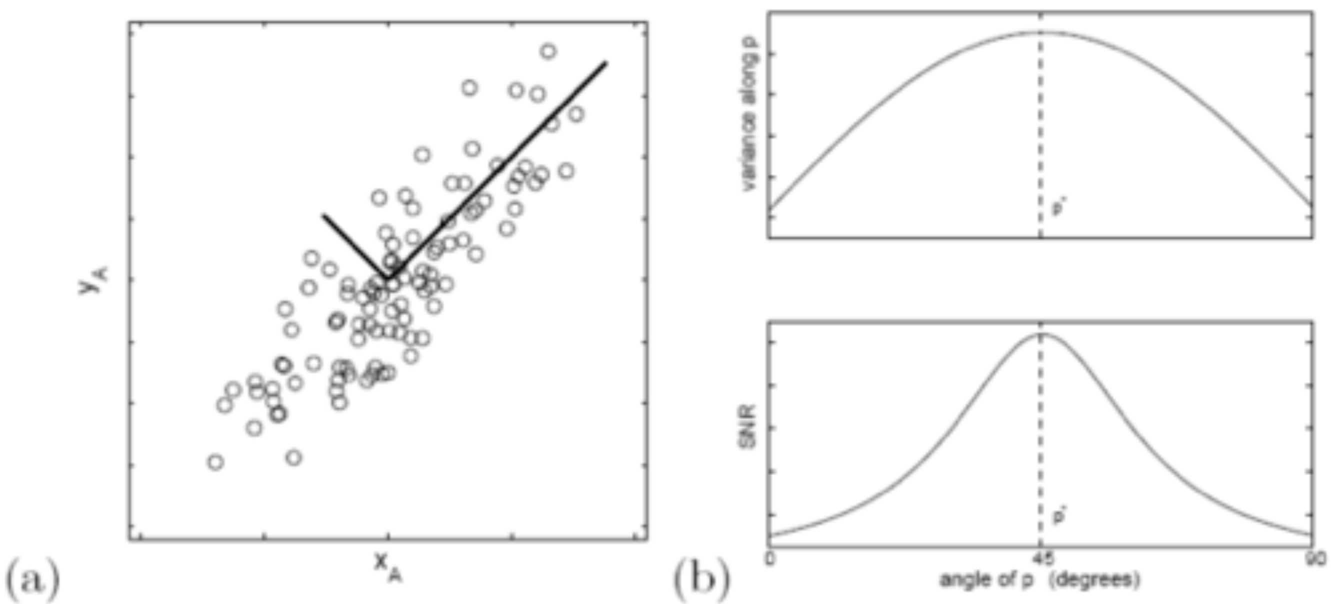
图表 1

矩阵说明

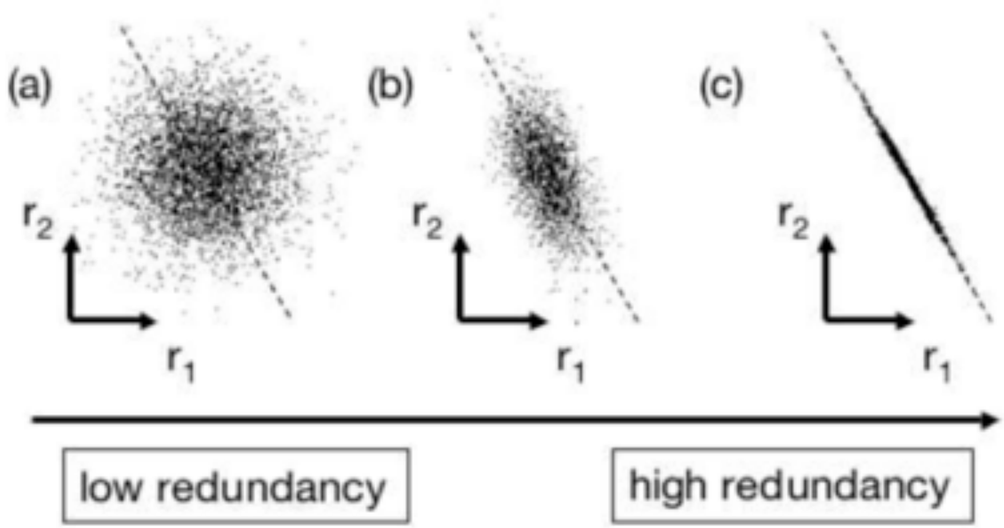
$$\text{数据维数}\left\{ \overbrace{\begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix}}^{\text{样本数量}} \right.$$

一个样本

噪音



冗余



图表 3：可能冗余数据的频谱图表示。r_1 和 r_2 分别是两个不同的观测变量。

（比如例子中的 x_A，y_B）。最佳拟合线 $r_2 = kr_1$ 用虚线表示。

PCA算法过程

1. 对于一个训练集，100个sample(i=1,2,3,...,100)，特征Xi是20维.[Xi1,Xi2,Xi3,...Xij,...,Xi20](j=1,2,..,20)，那么它可以建立一个20*100的样本矩阵M。

2. 紧接着我们开始求这个样本的协方差矩阵，得到一个20*20的协方差矩阵，计算过程如下：

- 先求解出Xi的平均Xav=(∑xi)/20;
- 对每一个Xi,计算Xi-Xav,即Mi(第 i 行)变为 Mi-Xav,记为Mn;
- 则容易得到协方差矩阵Z为Mn*Mn'(' 表示转置)

PCA算法过程

3. 求出这个协方差矩阵Z20x20的特征值和特征向量，一般情况下应该有20个特征值和特征向量，现在根据特征值的大小，取出较大的特征值以及其所对应的特征向量，（假设提取的特征值为较大的5个特征值），那么这5个特征向量就会构成一个20*5的矩阵V，这个矩阵就是我们要求的特征矩阵。

4. 用Mn'去乘以V，得到一个base矩阵（*），大小为100x5。

5. 任取一个样本1x100,乘上这个100*5的特征矩阵，就得到了一个1*5的新的样本，显然每个sample的维数下降了，然后再用这个1x5向量去比较相似性。

EM算法

- 最大期望算法(Expectation-maximization algorithm)
- 在统计中被用于寻找，依赖于不可观察的隐性变量的概率模型中，参数的最大似然估计。

EM算法流程

最大期望算法经过两个步骤交替进行计算：

第一步是计算期望（E），利用对隐藏变量的现有估计值，计算其最大似然估计值；
第二步是最大化（M），最大化在 E 步上求得的最大似然值来计算参数的值。

M 步上找到的参数估计值被用于下一个 E 步计算中，这个过程不断交替进行。

1.初始化分布参数

2.重复直到收敛：

E步骤：估计未知参数的期望值，给出当前的参数估计。

M步骤：重新估计分布参数，以使得数据的似然性最大，给出未知变量的期望估计。

EM for PCA的需要性

- PCA计算中需要计算样本的协方差矩阵来求特征值，计算量很大，效率低下，对于高维度数据处理效果欠佳。（主要原因）
- PCA在处理数据丢失时，需要使用特定的插值函数进行补全数据，或者直接扔掉数据。
- PCA为线性投影，只保留了数据之间的欧式距离，即原理欧式距离大的两点，降维后的空间中距离也大，保证方差大。

EMPCA优点

- 在处理大量高维度数据点，计算少量的特征向量和特征值
- 允许计算中存在数据丢失情况
- …

EM for PCA

- 简单来说，EM for PCA所做的就是推导设计出适合PCA算法的E步骤 和 M步骤的迭代函数

EMPCA算法简述

在 EMPCA算法中 ,PCA 方法可看作是一种有限情形下的 线性高斯模型特殊类。这种线性高斯模型假设变量 y 是由 k 维变 量 x 和附加的高斯噪声 v 构成。
因此 ,该模型可写成 :
y=Cx+v x~N(0,I) v~N(0,R)

其中矩阵 C 为 p∗k 维矩阵 ,v 为具有协方差矩阵 R 的 p 维噪声。 由于 x,v 是独立分布的 ,所以我们可以将变量 y 写成 :
y~N(0,CC’ +R)

EMPCA算法简述

- 在上面的线性高斯模型的时候 ,有 2 个中心问题是我们要关 注的。第一个问题是压缩:给定固定的模型参数C,R ,如何确定 观察值 y 的隐状态 x ?
- 由于数据点是独立的 ,我们关注单个隐状态给相应单个观察 量的后验概率 P(x | y)。这个可以很容易的通过线性矩阵投影计算 得到 :

$$P(x|y)=\frac{P(y|x)P(x)}{P(y)}=\frac{N(Cx,R)|_y\, N(0,I)|_x}{N(0,CC^T+R)|_y}$$

$$P(x|y)=N(\beta y, I-\beta C)|_x \text{ ，} \beta=C^T(CC^T+R)^{-1}$$

EMPCA算法简述

- 从以上我们可以得到隐状态βy和I−βC ,同时也可以得出如何用x重构y
- P(y|x)=N(Cx,R)|y
- 最终 ,任何数据点 y 的概率均可通过
- y~N(0,CC’ +R) 得到

EMPCA算法简述

- 第二个问题是参数设置这包括如何确定矩阵C,R ,使得由该模型构造的观察数据似然性最大。有很多 EM 算法去做这个过程, 但是不同的算法都有相似的结构:用公式

$$P(x \mid y) = N(\beta y, I - \beta C)\big|_x \ ， \ \beta = C^T (CC^T + R)^{-1}$$

- 在 E-step 中估计未知状态,然后在M-step中选择C,R目的是使估计值x和观察值 y 之间的概率最大.

EMPCA算法流程

- 变量：n为数据维数，N为数据量，Iter为迭代次数，K为要保留的主成分个数，x_mean为样本均值，X为样本矩阵，Y为估计矩阵
- 初始化C为 n*K的矩阵，可由rand()生成
- 经过N此迭代求C
  - E-step： $Y = (C^T C)^{-1} C^T X$
  - M-step: $C = XY^T (YY^T)^{-1}$

EMPCA算法流程

- 求出C的标准正交基，作为新的C值,即 C’C=I
- 求C’X的特征值和特征向量，假设为λ和V，并大小顺序排序，则最终的特征向量 U=C*V
- 假定x为一个n维样本，则EMPCA之后的新向量y为 y=U’ (x-x_mean)

- Thanks

原论文：

S.Roweis.EM.algorithms.for.PCA.and.SPCA.In.
M.I.Jordan,M.J.Kearns,and.S.A.Solla,editors,Advances.
in.Neural.Information.Processing.Systems,volume.10.MIT. Presss,1998.