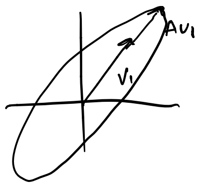
Eigen:

$Av = \lambda v$ (Markov)



$\Rightarrow$ Av do the same transformation like λv.

PCA:

- a statistical procedure: use orthogonal transformation to convert correlated var. into a set of linearly independent var.
  - smaller space
  - $\uparrow$ PC

Matric Diagonalization:

$A = V \Lambda V^{-1}$

$$\begin{bmatrix} | & & | \\ v_1 & \cdots & v_3 \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & & \\ & \lambda_2 & \\ & & \lambda_3 \end{bmatrix} \begin{bmatrix} | & & | \\ v_1 & & v_3 \\ | & & | \end{bmatrix}^{-1}$$

$\rightarrow$ only if A $\begin{cases} \text{is a squre matrix} \\ \text{has } n \text{ linearly indep. eigenvectors.} \end{cases}$

SVD. matrix factorization method.

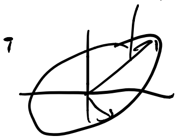
- Any A can be

$A = USV^T$, U.V orthogonal matrices with orthonormal eigenvec ($U \leftarrow AA^T$, $V \leftarrow A^TA$)

$S \rightarrow$ r. eigenvalues.

$= \underline{\sigma_1 u_1 v_1^T} + \cdots + \sigma_n u_n v_n^T$

$\sigma_1 u_1 v_1^T$



- extracts data in the directions with highest variances.

$C = A^T A/(n-1)$ cov. matrix. US: principal components.

$= VLV^T$

$V \rightarrow$ eigenvectors: pricipal axes.

henryWang

27 人赞同了该回答

先说PCA吧，PCA降维的大致思想就是：**挑选特征明显的、显得比较重要的信息保留下来。**

那么关键就是【特征明显的，重要的信息】如何选择？ 选择标准有两个：
1：同一个维度内的数据，方差大的比较明显，因为方差大表示自己和平均水平差异大，有个性，降维后也最可能分的开～
2：两个不同维度间关联度越小越好，因为关联度小表示这两个维度表征共同信息的量比较少，最理想就是两个维度不相关，相关度为0（**相关度可以用协方差cov(a,b)表示**），在线性空间内表现为两个维度正交～

协方差矩阵的主对角线和其余元素正好可以分别表示方差和协方差，而根据两条标准又很容易想到求特征值和特征向量，推导过程 @覃含章大大写的很清楚了

下面说一下为啥还有SVD，上面我们说PCA针对的是协方差矩阵C，但你得知道协方差矩阵是个方阵啊，难道不是方阵我们就不资瓷么？？ 所以就有了SVD～～

大概可以把SVD看作是对非方阵做PCA处理的一种方式啦，毕竟两者的套路都差不多，分解出特征值（SVD里是奇异值，数据XX'的特征值的平方根），挑比较大的特征值对应的特征向量构成投影矩阵，然后做线性变换（将数据X投影到低维空间）

编辑于 2018-11-21

方差最大的轴第一个奇异向量.

- PCA. SVD. two common ways to do dimension reduction.
- PCA can find out the principal axis, among which data will have the longest variance. So that we can keep most info in data with fewest axis/dimensions.
- SVD is a way to realize PCA. PCA has covariance matrix, use SVD. do the matrix factorization. → find eigenvector/singular value can represent the cov. matrix space
  ⇒ get principal axes and principal components.
  ↳ map the original data on new principal axes.