机器学习导论
第九章

王小航

维度的诅咒

- 许多机器学习问题涉及每个训练实例的成千上万甚至数百万个特征。
- 所有这些特征不仅使训练变得极其缓慢，而且还会使找到好的解决方案变得更加困难。这个问题通常称为维度的诅咒（the curse of dimensionality）。

降维

- 降维就是一种对高维度特征数据预处理方法。降维是将高维度的数据保留下最重要的一些特征，去除噪声和不重要的特征，从而实现提升数据处理速度的目的。在实际的生产和应用中，降维在一定的信息损失范围内，可以为我们节省大量的时间和成本。降维也成为应用非常广泛的数据预处理方法
- 降维具有如下一些优点：
  1. 使得数据集更易使用。
  2. 降低算法的计算开销。
  3. 去除噪声。
  4. 使得结果容易理解。

流行的降维技术

- 主成分分析（PCA）
- 内核主成分分析（Kernel PCA）
- 局部线性嵌入（LLE）
- 随机投影
- 多维缩放（MDS）
- Isomap
- t分布随机近邻嵌入（t-SNE）
- 线性判别分析（LDA）

主成分分析

- 在许多领域的研究与应用中，通常需要对含有多个变量的数据进行观测，收集大量数据后进行分析寻找规律。
- 多变量大数据集无疑会为研究和应用提供丰富的信息，但是也在一定程度上增加了数据采集的工作量。
- 更重要的是在很多情形下，许多变量之间可能存在相关性，从而增加了问题分析的复杂性。
- 如果分别对每个指标进行分析，分析往往是孤立的，不能完全利用数据中的信息，因此盲目减少指标会损失很多有用的信息，从而产生错误的结论。

主成分分析

- 需要找到一种合理的方法，在减少需要分析的指标同时，尽量减少原指标包含信息的损失，以达到对所收集数据进行全面分析的目的。
- 由于各变量之间存在一定的相关关系，因此可以考虑将关系紧密的变量变成尽可能少的新变量，使这些新变量是两两不相关的，那么就可以用较少的综合指标分别代表存在于各个变量中的各类信息。
- 主成分分析就属于这类降维算法。

概念

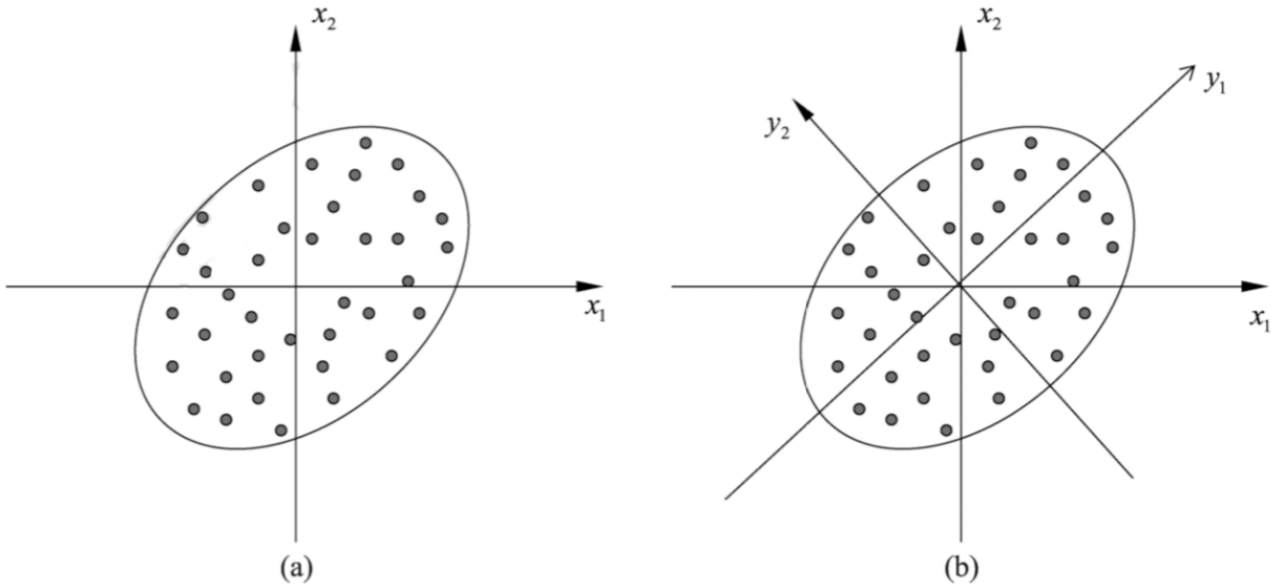
- PCA(Principal Component Analysis)，即主成分分析方法，是一种使用最广泛的数据降维算法。
- PCA的主要思想是将n维特征映射到k维上，这k维是全新的正交特征也被称为主成分，是在原有n维特征的基础上重新构造出来的k维特征。
- PCA的工作就是从原始的空间中顺序地找一组相互正交的坐标轴，新的坐标轴的选择与数据本身是密切相关的。

概念

- 第一个新坐标轴选择是原始数据中方差最大的方向，第二个新坐标轴选取是与第一个坐标轴正交的平面中使得方差最大的，第三个轴是与第1,2个轴正交的平面中方差最大的。依次类推，可以得到n个这样的坐标轴。
- 通过这种方式获得的新的坐标轴，我们发现，大部分方差都包含在前面k个坐标轴中，后面的坐标轴所含的方差几乎为0。
- 于是，我们可以忽略余下的坐标轴，只保留前面k个含有绝大部分方差的坐标轴。事实上，这相当于只保留包含绝大部分方差的维度特征，而忽略包含方差几乎为0的特征维度，实现对数据特征的降维处理。

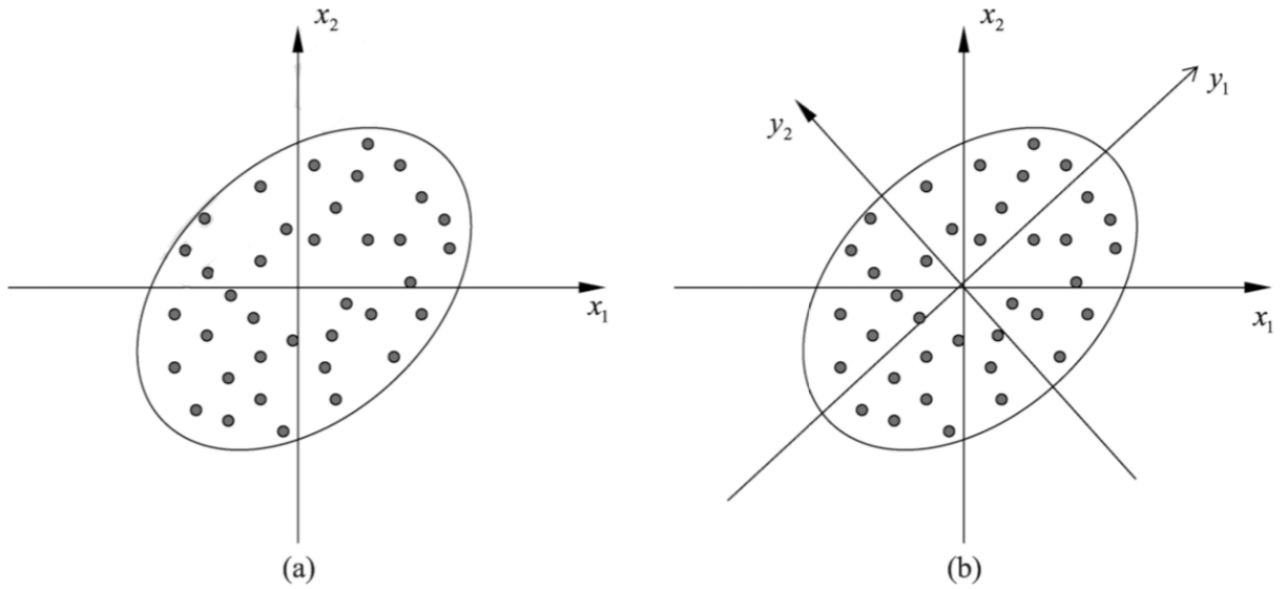
例

- 数据由线性相关的两个变量x1和x2表示
- 主成分分析对数据进行正交变换，对原坐标系进行旋转变换，并将数据在新坐标系表示



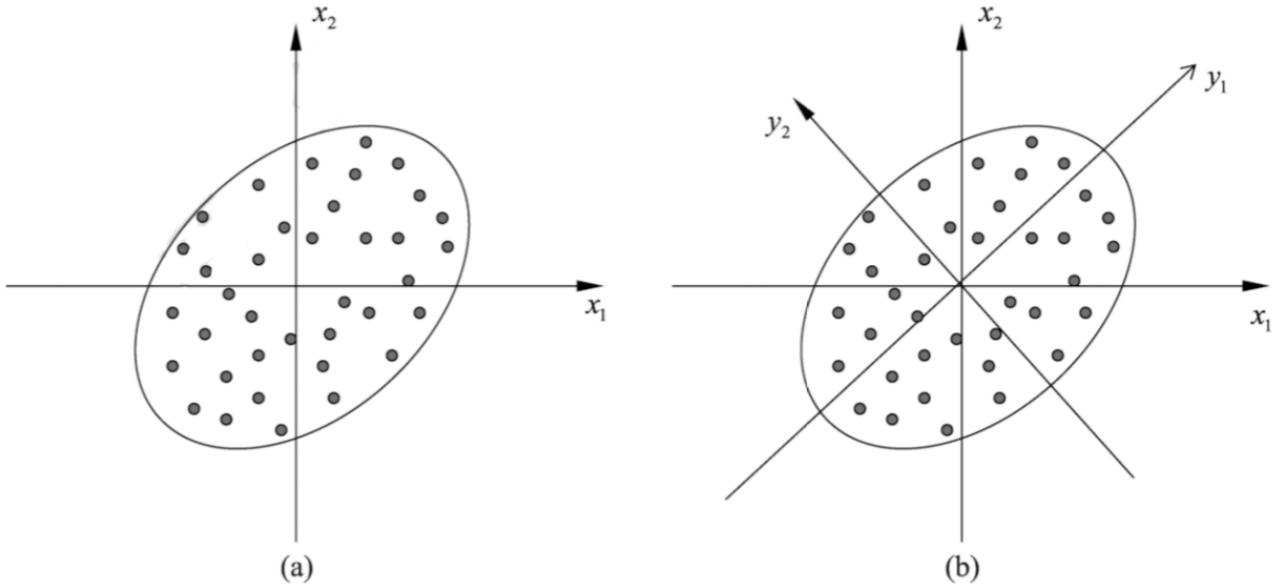
例

- 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，即y_1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，即y_2轴



例

- 在新坐标系里，数据中的变量y_1和y_2是线性无关的，当知道其中一个变量y_1的取值时，对另一个变量y_2的预测是完全随机的，反之亦然
- 如果主成分分析只取第一主成分，即新坐标系的y_1轴，那么等价于将数据投影在椭圆长轴上，用这个主轴表示数据，将二维空间的数据压缩到一维空间中。



概念

- 思考：我们如何得到这些包含最大差异性的主成分方向呢？
- 答案：事实上，通过计算数据矩阵的协方差矩阵，然后得到协方差矩阵的特征值特征向量，选择特征值最大(即方差最大)的k个特征所对应的特征向量组成的矩阵。这样就可以将数据矩阵转换到新的空间当中，实现数据特征的降维。

协方差

- 样本均值：

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{N} x_i$$

- 样本方差：

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

- 样本X和样本Y的协方差：

$$Cov(X,\mathrm{Y}) = \mathrm{E}[\mathrm{X} - \mathrm{E}(\mathrm{X})][\mathrm{Y} - \mathrm{E}(\mathrm{Y})] = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

协方差

- 由上面的公式，我们可以得到以下结论：

  (1) 方差的计算公式是针对一维特征，即针对同一特征不同样本的取值来进行计算得到；而协方差则必须要求至少满足二维特征；方差是协方差的特殊情况。

  (2) 方差和协方差的除数是n-1,这是为了得到方差和协方差的无偏估计。

- 协方差为正时，说明X和Y是正相关关系；协方差为负时，说明X和Y是负相关关系；协方差为0时，说明X和Y是相互独立。Cov(X,X)就是X的方差

协方差

- 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)。例如，对于3维数据(x,y,z)，计算它的协方差就是：

$$Cov(X,Y,Z) = \begin{bmatrix} Cov(x,x) & Cov(x,y) & Cov(x,z) \\ Cov(y,x) & Cov(y,y) & Cov(y,z) \\ Cov(z,x) & Cov(z,y) & Cov(z,z) \end{bmatrix}$$

特征值分解矩阵原理

- (1) 特征值与特征向量
- 如果一个向量v是矩阵A的特征向量，将一定可以表示成下面的形式：

$$Av = \lambda v$$

其中，λ是特征向量v对应的特征值，一个矩阵的一组特征向量是一组正交向量。

特征值分解矩阵原理

- (2) 特征值分解矩阵
- 协方差矩阵 是一个对称矩阵。
- 回想我们学过的线性代数内容，对于任意对称矩阵Σ，存在一个特征值分解(eigenvalue decomposition, EVD)：

$$\Sigma = Q\Lambda Q^{-1}$$

其中，的每一列都是相互正交的特征向量，且是单位向量，满足$Q^TQ = I$，Λ对角线上的元素是从大到小排列的特征值，非对角线上的元素均为0。

基于特征值分解协方差矩阵实现PCA算法

- 输入：数据集$X=\{x_1,x_2,\ldots,x_n\}$，需要降到k维。
  - 1) 去平均值(即去中心化)，即每一位特征减去各自的平均值。
  - 2) 计算协方差矩阵
  - 3) 用特征值分解方法求协方差矩阵的特征值与特征向量。
  - 4) 对特征值从大到小排序，选择其中最大的k个。然后将其对应的k个特征向量分别作为行向量组成特征向量矩阵P。
  - 5) 将数据转换到k个特征向量构建的新空间中，即Y=PX。

主成分个数的选择

- 第k主成分的方差贡献率为：

$$\frac{\lambda_k}{\sum_{i=1}^{m}\lambda_i}$$

- k个主成分的累计方差贡献率为：

$$\frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{m}\lambda_i}$$

- 选择k:给定一个比率，如90%，使得

$$\frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{m}\lambda_i} \geq 90\%$$

例

$$X = \begin{pmatrix} -1 & -1 & 0 & 2 & 0 \\ -2 & 0 & 0 & 1 & 1 \end{pmatrix}$$

- 以X为例，我们用PCA方法将这两行数据降到一行。

例

- 1)因为X矩阵的每行已经是零均值，所以不需要去平均值。
- 2)求协方差矩阵：

$$C = \frac{1}{n-1} XX^T = \begin{pmatrix} \frac{3}{2} & 1 \\ 1 & \frac{3}{2} \end{pmatrix}$$

例

- 3)求协方差矩阵的特征值与特征向量。
- 求解后的特征值为：

$$\lambda_1 = \frac{5}{2}, \lambda_2 = \frac{1}{2}$$

- 对应的特征向量为：

$$c_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, c_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

例

- 标准化后的特征向量为：

$$c_1 = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, c_2 = \begin{pmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

- 4)矩阵P为：

$$P = \begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

例

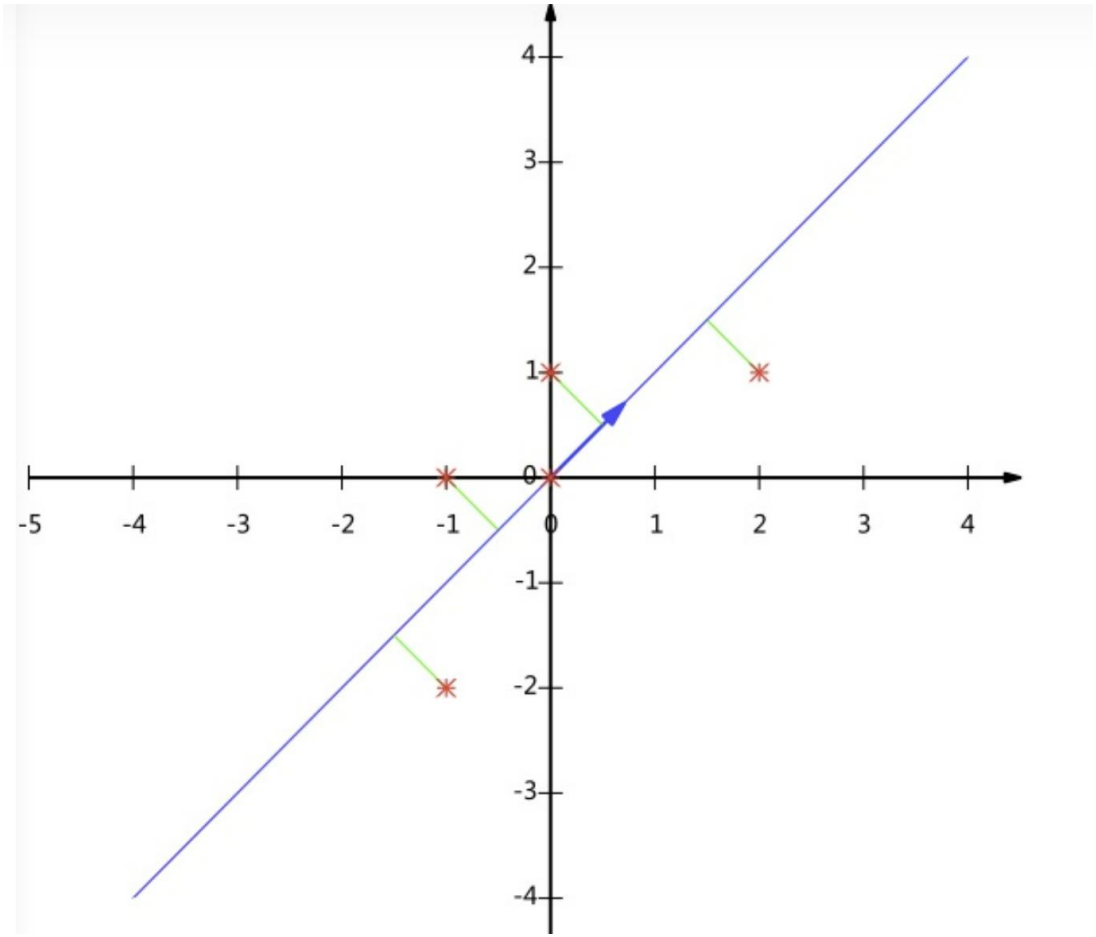
- 5)最后我们用P的第一列的转置乘以数据矩阵X，就得到了降维后的表示：

$$Y = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} -1 & -1 & 0 & 2 & 0 \\ -2 & 0 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} -\frac{3}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & \frac{3}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

- 样本主成分矩阵为$Y = P^T X$
- PCA逆变换，回到原始数量的维度：$X_{recovered} = PY$

例

- 投影结果：



PCA的Python实现

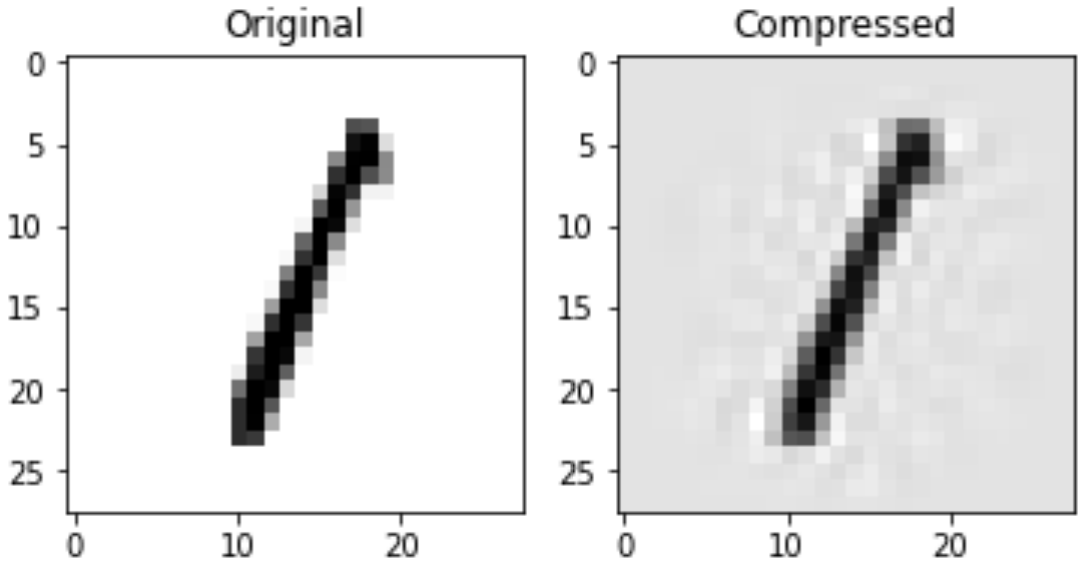
学号	语文	数学	物理	化学	英语	历史
1	84	65	61	72	79	81
2	64	77	77	76	55	70
3	65	67	63	49	57	67
4	74	80	69	75	63	74
5	84	74	70	80	74	82

基于sklearn的PCA

- from sklearn.decomposition import PCA
- import numpy as np
- X = np.array([[84,65,61,72,79,81],[64,77,77,76,55,70],[65,67,63,49,57,67],[74,80,69,75,63,74],[84,74,70,80,74,82]])
- pca=PCA(n_components=2)
- pca.fit(X)
- print(pca.transform(X))

用PCA处理MNIST数据

```
pca = PCA(n_components = 154)
X_reduced = pca.fit_transform(X_train)
X_recovered = pca.inverse_transform(X_reduced)
```



KPCA

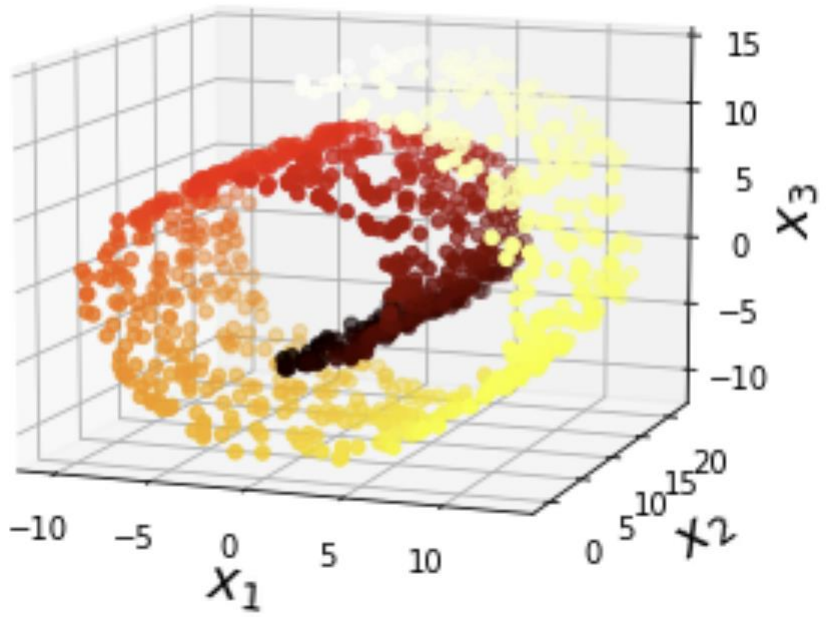
- 主成分分析(Principal Components Analysis, PCA)适用于数据的线性降维。而核主成分分析(Kernel PCA, KPCA)可实现数据的非线性降维，用于处理线性不可分的数据集。

- KPCA的大致思路是：对于输入空间(Input space)中的矩阵X ，我们先用一个非线性映射把 X中的所有样本映射到一个高维甚至是无穷维的空间(称为特征空间，Feature space)，(使其线性可分)，然后在这个高维空间进行PCA降维

KPCA

- “瑞士卷”数据

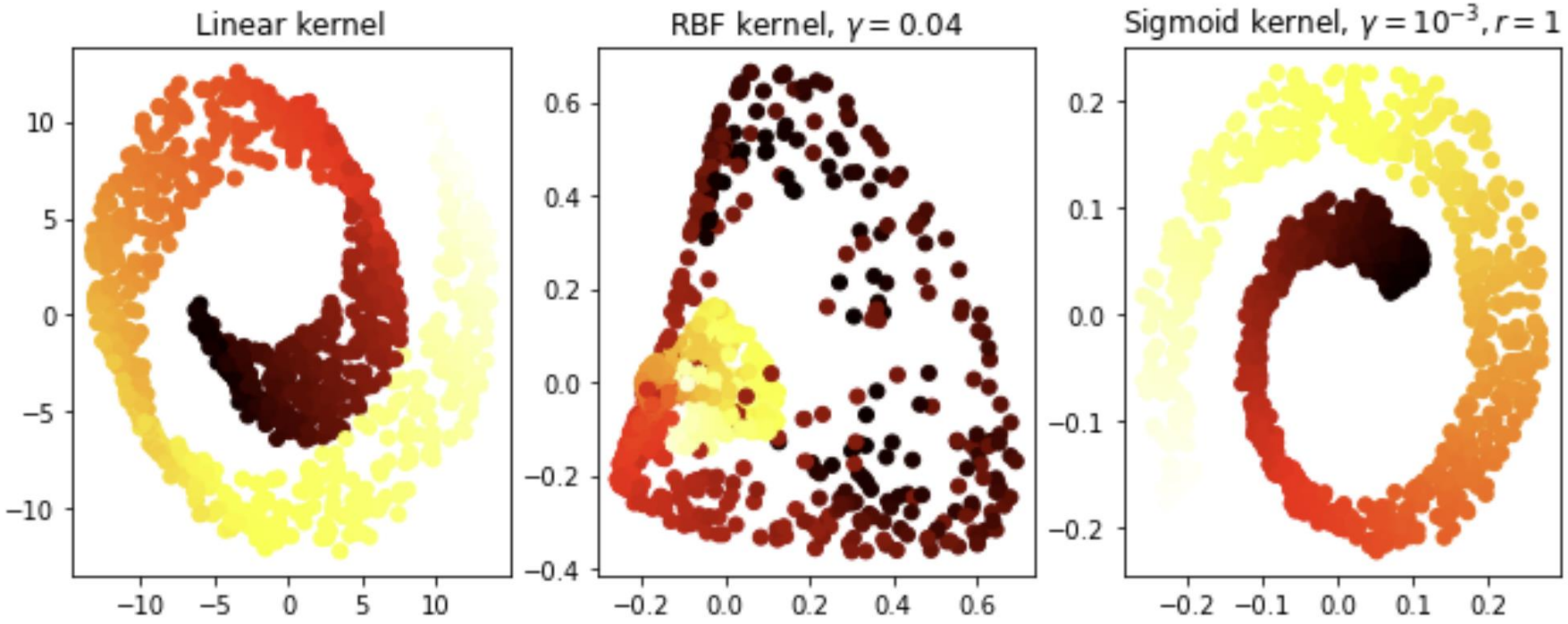
```
from sklearn.datasets import make_swiss_roll
X, t = make_swiss_roll(n_samples=1000, noise=0.2, random_state=42)
```



KPCA

- from sklearn.decomposition import KernelPCA
- lin_pca = KernelPCA(n_components = 2, kernel="linear")
- rbf_pca = KernelPCA(n_components = 2, kernel="rbf", gamma=0.0433)
- sig_pca = KernelPCA(n_components = 2, kernel="sigmoid", gamma=0.001, coef0=1)

KPCA


Linear kernel
RBF kernel, $\gamma = 0.04$
Sigmoid kernel, $\gamma = 10^{-3}, r = 1$
10
5
0
−5
−10
−10
−5
0
5
10
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.2
0.0
0.2
0.4
0.6
0.2
0.1
0.0
−0.1
−0.2
−0.2
−0.1
0.0
0.1
0.2