第七讲 维度归约

授课老师：郭 迟 教授

guochi@whu.edu.cn

武汉大学测绘学院

2021.11

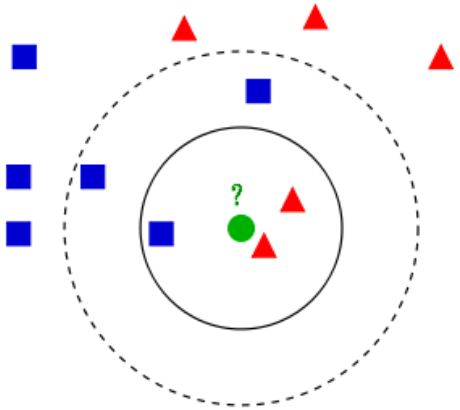
维度归约

1. K近邻学习（自学）
2. 低维嵌入与多维标定（MDS）
3. 主成分分析（PCA）
4. 流形学习

谁还记得与之相对应的概念是什么？

K近邻学习

- K近邻(k-Nearesr Neighbor，kNN)学习：经典的监督学习和基于样例的学习方法。算法思想是：给定某个测试样本，kNN基于某种**距离度量**在训练集中找出与其最近的k个带有真实标记的训练样本，然后基于这k个邻居的真实标记来进行预测



- 投票法：选择k个样本中出现最多的类别标记作为预测结果
- 平均法：将这k个样本的实值输出标记的平均值作为预测结果

如上图所示，有两类不同的样本数据，分别用蓝色的小正方形和红色的小三角形表示，而图正中间的那个绿色的圆所标示的数据则是待分类的数据。根据k近邻的思想来给绿色圆点进行分类。

- 如果k=3，绿色圆点的最邻近的3个点是2个红色小三角形和1个蓝色小正方形，少数从属于多数，基于统计的方法，判定绿色的这个待分类点属于红色的三角形一类
- 如果k=5，绿色圆点的最邻近的5个邻居是2个红色三角形和3个蓝色的正方形，还是少数从属于多数，基于统计的方法，判定绿色的这个待分类点属于蓝色的正方形一类

教材P110

第七讲 维度归约

K近邻学习

- kNN分类器中的k是一个重要参数，当k取不同值时，分类结果会有显著不同。另一方面，若采用不同的距离计算方式，则找出的“近邻”可能有显著差别，从而也会导致分类结果有显著不同
- “距离”可以是明氏距离的各种特例，也可以是自定义的有含义的距离

明氏（Minkowski） 距离

设特征空间 $\mathcal{X}$ 是 n 维实数向量空间 $\mathbf{R}^n$，$x_i, x_j \in \mathcal{X}$，$x_i = (x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(n)})^{\mathrm{T}}$，$x_j = (x_j^{(1)}, x_j^{(2)}, \cdots, x_j^{(n)})^{\mathrm{T}}$，$x_i, x_j$ 的 L_p 距离定义为

$$L_p(x_i, x_j) = \left(\sum_{l=1}^{n} | x_i^{(l)} - x_j^{(l)} |^p \right)^{\frac{1}{p}}$$

这里 $p \geqslant 1$．当 $p = 2$ 时，称为欧氏距离(Euclidean distance)，即

$$L_2(x_i, x_j) = \left(\sum_{l=1}^{n} | x_i^{(l)} - x_j^{(l)} |^2 \right)^{\frac{1}{2}}$$

当 $p = 1$ 时，称为曼哈顿距离（Manhattan distance），即

$$L_1(x_i, x_j) = \sum_{l=1}^{n} | x_i^{(l)} - x_j^{(l)} |$$

当 $p = \infty$ 时，它是各个坐标距离的最大值，即

$$L_\infty(x_i, x_j) = \max_l | x_i^{(l)} - x_j^{(l)} |$$

K近邻学习

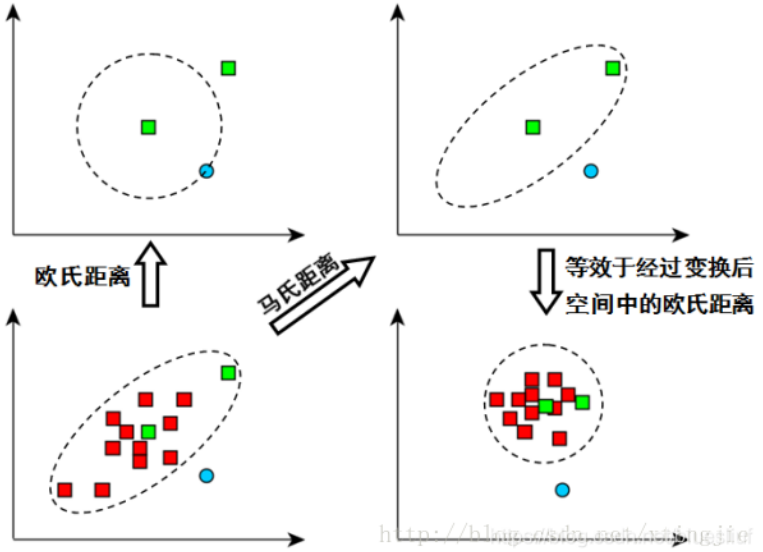
➢ 马氏距离（Mahalanobis Distance）

对于一个均值为$\mu = (\mu_1, \mu_2, \cdots, \mu_p)^T$，协方差矩阵为$S$的多变量$x = (x_1, x_2, \cdots, x_p)^T$，其马氏距离为：

$$D_M(x) = \sqrt{(x-\mu)^T S^{-1}(x-\mu)}$$

可以发现如果S^{-1}是单位矩阵的时候，马氏距离简化为欧式距离

- 马氏距离不受量纲的影响，两点之间的马氏距离与原始数据的测量单位无关；
- 由标准化数据和中心化数据（即原始数据与均值之差）计算出的二点之间的马氏距离相同；
- 马氏距离还可以排除变量之间的相关性的干扰

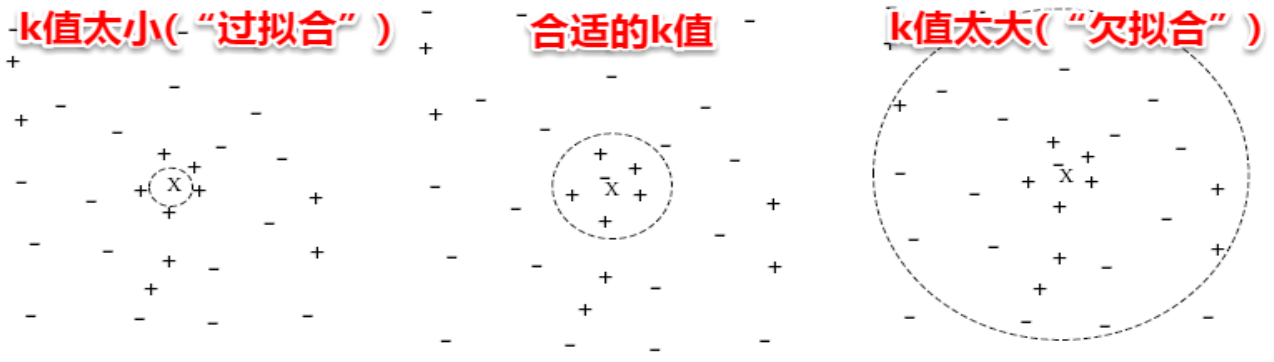


参考阅读

第七讲 维度归约

K近邻学习

- *k*NN是一种典型的懒惰学习(lazy learning)方法。 此类学习技术在训练阶段仅仅是把样本保存起来，而是当有新样本需要预测时，才来计算出最近的k个邻居，训练时间开销为零。这样的学习也称为急切学习(eager learning)，即在训练阶段就对样本进行学习处理的方法



(a) Neighborhood too small.　(b) Neighborhood just right.　(c) Neighborhood too large.

- *k*值选取太小，模型很容易受到噪声数据的干扰，例如：极端地取*k*=1，若待分类样本正好与一个噪声数据距离最近，就导致了分类错误；
- 若*k*值太大， 则在更大的邻域内进行投票，此时模型的预测能力大大减弱，例如：极端取*k*=训练样本数，就相当于模型根本没有学习，所有测试样本的预测结果都是一样的
- **一般地我们都通过交叉验证法来选取一个适当的*k*值**

K近邻学习

- 为了保证每个特征同等重要性，我们这里对每个特征进行归一化，否则在计算距离的时候，数值大的特征会起主要作用，掩盖数值小的特征
- *k*NN的方法虽然简单，但十分有效。其泛化错误率不超过贝叶斯最优分类器错误率的2倍

推导见西瓜书P226

思考：*k*NN算法降低了样本的特征维度吗？为什么很多ML教材都在“维度归约”的章节先讲*k*NN呢？

维度归约

1. K近邻分类器（自学）
2. 低维嵌入与多维标定（MDS）
3. 主成分分析（PCA）
4. 流形学习

第七讲 维度归约

低维嵌入与多维标定

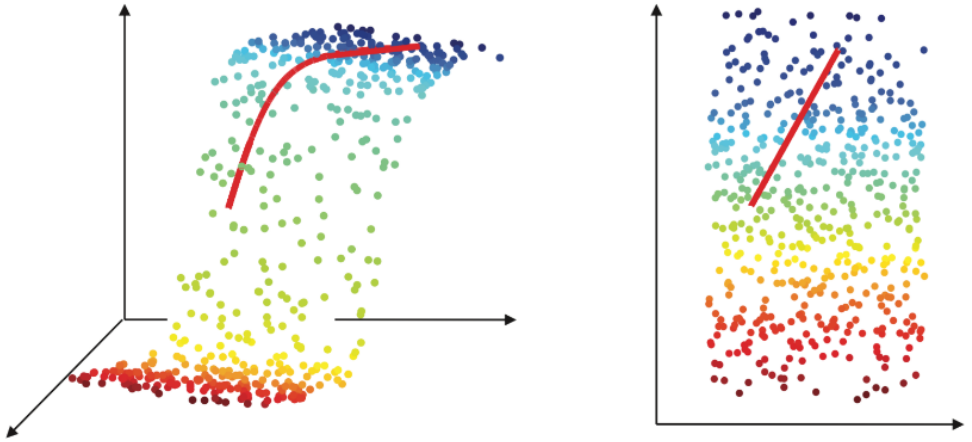
- 现实应用中属性维数经常成千上万，要满足密采样条件所需的样本数目是无法达到的天文数字。许多学习方法都涉及距离计算，而高维空间会给距离计算带来很大的麻烦，例如当维数很高时甚至连计算内积都不再容易。在高维情形下出现的数据样本稀疏、距离计算困难等问题，是所有机器学习方法共同面临的严重障碍，被称为“维数灾难”（curse of dimensionality）

- 特征选择（feature selection）：从d维特征中找出提供最多信息的k个，丢弃剩下的d-k个特征（归纳偏倚）
- 特征提取（feature extraction）：找出k个维的新特征集合，这些维是原来d个维的组合（$k<<d$），又称为维度归约(dimension reduction)或降维。即通过某种数学变换，将原始高维属性空间转变为一个低维“子空间”(subspace)，在这个子空间中样本密度大幅度提高，距离计算也变得更为容易

第七讲 维度归约

低维嵌入与多维标定

- 数据样本虽然是高维的，但与学习任务密切相关的也许仅是某个低维分布，即高维空间中的一个“低维嵌入”(embedding)，因而可以对数据进行有效的降维



(a) 三维空间中观察到的样本点　　(b) 二维空间中的曲面

- 比较降维前后学习器的性能，若性能有所提高则认为降维起到了作用；若维数降至2维或者3维，还可通过可视化的技术来直观地判断降维效果

- 多维标定（Multiple Dimensional Scaling, MDS）是一种经典的数据降维方法。

  该方法降维的核心思想是：寻找一个低维子空间，样本在此空间的距离与原始空间中样本间距离保持不变

教材P79

MDS

假定有m个样本，在原始空间中的距离矩阵为 $D \in \Re^{m \times m}$，其第i行j列的元素dist_{ij}为样本x_i到x_j的距离。目标是获得样本在$d'(d' \le d)$维空间中的表示$Z \in \Re^{d' \times m}$，样本的欧氏距离与原始空间中的距离相等，即

$$\|z_i - z_j\| = dist_{ij}$$

- 令$B = Z^T Z \in \Re^{m \times m}$，其中$B$为降维后的内积矩阵，$b_{ij} = z_i^T z_j$，有

$$\begin{aligned} dist_{ij}^2 &= \|z_i\|^2 + \|z_j\|^2 - 2z_i^T z_j \\ &= b_{ii} + b_{jj} - 2b_{ij} \end{aligned}$$

- 为便于讨论，令降维后的样本Z被中心化（矩阵所有的元素减去均值后的新矩阵，原点移至矩阵中心），即$\sum_{i=1}^{m} z_i = 0$。显然，矩阵$B = Z^T Z \in \Re^{m \times m}$的行与列之和均为零$\sum_{i=1}^{m} b_{ij} = \sum_{j=1}^{m} b_{ij} = 0$

$$B = \begin{bmatrix} z_1 \\ \dots \\ z_m \end{bmatrix} * \begin{bmatrix} z_1 & \dots & z_m \end{bmatrix} = \begin{bmatrix} z_1 z_1 & z_1 z_2 & \dots & z_1 z_m \\ z_2 z_1 & z_2 z_2 & \dots & z_2 z_m \\ \dots & \dots & \dots & \dots \\ z_m z_1 & z_m z_2 & \dots & z_m z_m \end{bmatrix}$$

和为零向量

和为零向量

- 定义矩阵B的迹（trace）为其主对角线元素之和，记为$tr(B) = \sum_{i=1}^{m} \|z_i\|^2$，有：

$$\left.\begin{aligned} dist_{ij}^2 &= b_{ii} + b_{jj} - 2b_{ij} \\ \sum_{i=1}^{m} b_{ij} &= \sum_{j=1}^{m} b_{ij} = 0 \end{aligned}\right\} \Rightarrow \left\{\begin{aligned} \sum_{i=1}^{m} dist_{ij}^2 &= tr(B) + m b_{jj}, \\ \sum_{j=1}^{m} dist_{ij}^2 &= tr(B) + m b_{ii}, \\ \sum_{i=1}^{m}\sum_{j=1}^{m} dist_{ij}^2 &= 2m\,tr(B) \end{aligned}\right.$$

第七讲 维度归约

低维嵌入与多维标定

- 令 $dist_{i\bullet} = \frac{1}{m}\sum_{j=1}^{m} dist_{ij}$

  $dist_{\bullet j} = \frac{1}{m}\sum_{i=1}^{m} dist_{ij}$

  $dist_{\bullet\bullet} = \frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{m} dist_{ij}$

- 由此即可通过D矩阵求取新样本Z的内积矩阵B，保持距离不变

$$b_{ij} = -\frac{1}{2}\left(dist_{ij}^2 - dist_{i\bullet}^2 - dist_{\bullet j}^2 + dist_{\bullet\bullet}^2\right)$$

- 对矩阵B做特征值分解，$B = V\Lambda V^{T}$，其中 $\Lambda = diag(\lambda_1, \lambda_2, \ldots \lambda_d)$ 为特征值构成的对角矩阵，V为特征向量矩阵。又 $B = Z^T Z \in \Re^{m\times m}$，则

$$\mathrm{B} = V\begin{pmatrix}\lambda_1 & & 0\\ & \ddots & \\ 0 & & \lambda_n\end{pmatrix}V' = V\begin{pmatrix}\sqrt{\lambda_1} & & 0\\ & \ddots & \\ 0 & & \sqrt{\lambda_n}\end{pmatrix}\underbrace{\begin{pmatrix}\sqrt{\lambda_1} & & 0\\ & \ddots & \\ 0 & & \sqrt{\lambda_n}\end{pmatrix}}_{Z}V'$$

第七讲 维度归约

低维嵌入与多维标定

- 令降维后的距离与原始空间中的距离尽可能接近，而不必严格相等。此时可取$d' << d$个最大特征值构成对角矩阵$\tilde{\Lambda} = diag(\lambda_1, \lambda_2, \ldots \lambda_{d'})$，令$\tilde{V}$表示相应的特征向量矩阵，则$Z$可表达为

$$Z = \tilde{\Lambda}^{1/2} \tilde{V}^T \in \Re^{d' \times m}$$

➢ 因此，MDS算法描述为：

输入： 距离矩阵 $\mathbf{D} \in \mathbb{R}^{m \times m}$，其元素 $dist_{ij}$ 为样本 $\boldsymbol{x}_i$ 到 $\boldsymbol{x}_j$ 的距离；
低维空间维数 d'.

过程：

1: 计算 $dist_{i\cdot}^2$, $dist_{\cdot j}^2$, $dist_{\cdot\cdot}^2$;
2: 计算矩阵 $\mathbf{B}$;
3: 对矩阵 $\mathbf{B}$ 做特征值分解;
4: 取 $\tilde{\mathbf{\Lambda}}$ 为 d' 个最大特征值所构成的对角矩阵，$\tilde{\mathbf{V}}$ 为相应的特征向量矩阵.

降维后样本Z的内积矩阵

输出： 矩阵 $\tilde{\mathbf{V}}\tilde{\mathbf{\Lambda}}^{1/2} \in \mathbb{R}^{m \times d'}$，每行是一个样本的低维坐标

维度归约

1. K近邻分类器（自学）
2. 低维嵌入与多维标定（MDS）
3. 主成分分析（PCA）
4. 流形学习

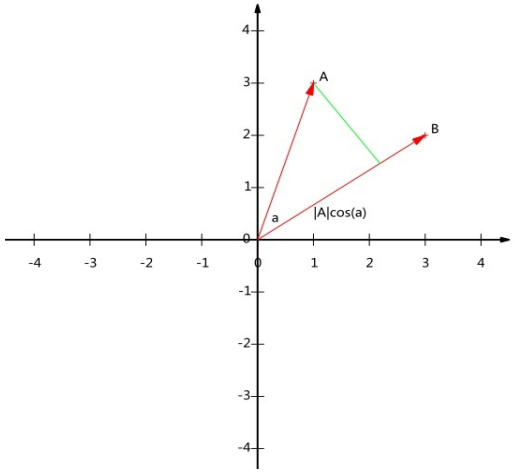
第七讲 维度归约

主成分分析

- 主成分分析(Principal Component Analysis, 简称PCA) 是指通过正交变换将一组高维的、可能存在相关性的变量转换为一组低维的、线性不相关的变量的方法。转换后的这组变量称为“主成分”，能最大程度反映原变量的特性

1. 内积和基变化

- 两个向量的 A 和 B 内积 $(a_1, a_2, \cdots, a_n) \cdot (b_1, b_2, \cdots, b_n)^{\mathsf{T}} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$ 等于 A 到 B 的投影长度乘以 B 的模



设 B 的模为 1，即让 $|B| = 1$，那么：

A 与 B 的内积值等于 A 向 B 所在直线投影的标量大小

第七讲 维度归约

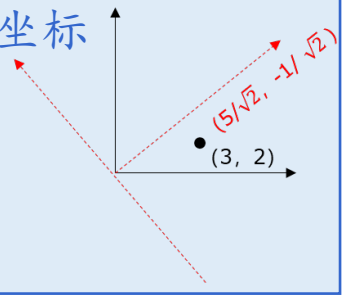
主成分分析

1. 内积和基变化

- 在线性代数中，基（也称为基底）是描述、刻画向量空间的基本工具。向量空间中任意一个元素，都可以唯一地表示成基向量的线性组合。要准确描述向量，首先要确定一组基，然后给出在基向量上的投影值。比如向量(3,2) = 3(1,0) +2(0,1)

e.g. 计算向量（3, 2）在 $(\frac{1}{\sqrt{2}},\frac{1}{\sqrt{2}}),(-\frac{1}{\sqrt{2}},\frac{1}{\sqrt{2}})$ 这组基下的坐标

解：基矩阵和向量求内积 $\begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}\begin{pmatrix} 3 \\ 2 \end{pmatrix}=\begin{pmatrix} 5/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix}$



- 将样本特征矩阵变换到一个新的空间中的通用方法：

基向量1 → p_1
基向量2 → p_2
…
基向量R → p_R

$$\begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_R \end{pmatrix}\begin{pmatrix} a_1 & a_2 & \cdots & a_M \end{pmatrix}=\begin{pmatrix} p_1a_1 & p_1a_2 & \cdots & p_1a_M \\ p_2a_1 & p_2a_2 & \cdots & p_2a_M \\ \vdots & \vdots & \ddots & \vdots \\ p_Ra_1 & p_Ra_2 & \cdots & p_Ra_M \end{pmatrix}$$

a_1：特征向量1（样本1）　a_M：特征向量M（样本M）

p_1a_1：新特征向量1（新样本1）　p_1a_M：新特征向量M（新样本M）

问：原样本和新样本各有多少维特征？

主成分分析

2. 线性降维

- 一般来说欲获得低维子空间，最简单的是对原始高维空间进行线性变换。给定d维空间中的样本$X=(x_1,x_2,\ldots,x_m)\in\Re^{d\times m}$，变换之后得到$d'\le d$维空间中的样本$Z=W^TX$，其中$W\in\Re^{d\times d'}$是变换矩阵，$Z\in\Re^{d'\times m}$是样本在新空间中的表达，即在一个线性超平面上的表达
- 变换矩阵W由新空间的d‘个d维向量作为基向量。若$\{w_1,w_2,\ldots,w_{d'}\}$这组基向量两两正交，则实现了PCA中将“高维的、可能存在相关性的变量转换为一组低维的、线性不相关的变量”的变换目的

提问：1. 什么叫向量正交？

2. 想一想正交投影相较于非正交投影用于降维的优点

主成分分析

- PCA方法可以从多个角度解释，最常见的推导角度有：
  - 最大可分性：样本点在一个线性超平面上的投影能尽可能分开
  - 最近重构性：样本点到一个线性超平面的距离都足够近；

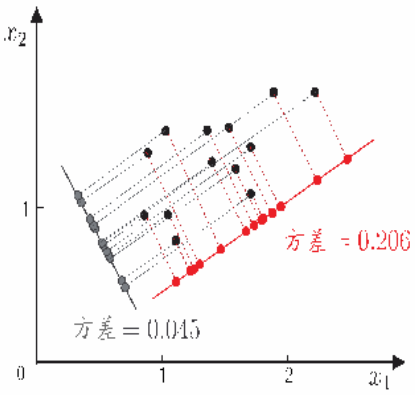
3. 最大可分性

PCA（1）

假定有m个样本，目标是获得样本在$d'(d' \leq d)$维空间中的表示 $Z \in \Re^{d' \times m}$，样本投影能尽可能分开

教材P70 - 71

- 样本点在一个线性超平面上的投影能尽可能分开，也就避免了样本在降维后“重叠”，保持了最大的信息熵。因此最大可分性可以理解为：投影后样本点的方差最大化
- 引入协方差矩阵 Σ求解这个过程。 Σ反映各变量的相关性



主成分分析

3. 最大可分性

- 对于一个中心化的矩阵$X=\begin{pmatrix} a_1 & a_2 & \cdots & a_m \\ b_1 & b_2 & \cdots & b_m \end{pmatrix}$，两个变量的协方差为：

$$Cov(a,b)=\frac{1}{m-1}\sum_{i=1}^{m}(a_i-\mu_a)(b_i-\mu_b)\approx\frac{1}{m}\sum_{i=1}^{m}a_ib_i$$

- X的协方差矩阵为：

协方差反映线性相关性

$$\Sigma=\frac{1}{m}XX^{\mathsf{T}}=\begin{pmatrix} \frac{1}{m}\sum_{i=1}^{m}a_i^2 & \frac{1}{m}\sum_{i=1}^{m}a_ib_i \\ \frac{1}{m}\sum_{i=1}^{m}a_ib_i & \frac{1}{m}\sum_{i=1}^{m}b_i^2 \end{pmatrix}=\begin{pmatrix} Cov(a,a) & Cov(a,b) \\ Cov(b,a) & Cov(b,b) \end{pmatrix}$$

主对角线：方差

- 因此PCA的目标有可以解释为：设原始数据矩阵 X 对应的协方差矩阵为 $\frac{1}{m}XX^{\mathsf{T}}$，而 P 是一组基按行组成的矩阵，设 $Z=PX$，则 Z 为 X 对 P 做基变换后的数据（投影数据）。Z的协方差矩阵为$\frac{1}{m}ZZ^{\mathsf{T}}$，我们希望$Z$的协方差矩阵是一个对角矩阵，且主对角线元素尽可能大

主成分分析

3. 最大可分性

- 推导可知：

$$\frac{1}{m}ZZ^{\mathrm{T}}=\frac{1}{\mathrm{m}}(\mathrm{P}X)\,(\mathrm{P}X)^{\mathrm{T}}=\frac{1}{\mathrm{m}}\mathrm{P}XX^{\mathrm{T}}\mathrm{P}^{\mathrm{T}}=\mathrm{P}\left(\frac{1}{\mathrm{m}}XX^{\mathrm{T}}\right)\mathrm{P}^{\mathrm{T}}$$

由于$C=\frac{1}{\mathrm{m}}XX^{\mathrm{T}}$是原始数据$X$的协方差矩阵，是一个实对称矩阵。实对称矩阵不同特征值对应的特征向量必然正交，且可以对角化、对角阵上的元素即为矩阵本身特征值。求X协方差矩阵其特征值、特征向量，将求得的特征值排序：$\lambda_1\geq\lambda_2\geq\cdots\geq\lambda_d$，再取前$d'$个特征值对于的特征向量$W^{\mathrm{T}}$即我们要找的这一组$d'$个正交基P

$$W^{T}CW=\Lambda=\begin{pmatrix}\lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n\end{pmatrix}$$

X投影后的方差就是协方差矩阵的特征值。我们要找到最大方差也就是协方差矩阵最大的特征值，最佳投影方向就是最大特征值所对应的特征向量，次佳就是第二大特征值对应的特征向量...

教材P72碎石图

主成分分析

3. 最大可分性

- 因此，PCA求解的过程可以用目标函数描述为：

$$\max_{W} \quad tr(W^T XX^T W)$$
$$s.t. \quad W^T W = I$$

用Lagrange乘数法求解： $XX^T w_i = \lambda_i w_i$

即对X协方差矩阵进行特征值分解，取最大d'个特征值对应的特征向量做基变换

输入： 样本集 $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$;
低维空间维数 d'.

过程：

1: 对所有样本进行中心化: $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i - \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i$;
2: 计算样本的协方差矩阵 $\mathbf{X}\mathbf{X}^{\mathrm{T}}$;
3: 对协方差矩阵 $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ 做特征值分解;
4: 取最大的 d' 个特征值所对应的特征向量 $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'}$.

输出： 投影矩阵 $\mathbf{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'})$.

主成分分析

- PCA方法可以从多个角度解释，最常见的推导角度有：
  - 最大可分性：样本点在一个线性超平面上的投影能尽可能分开
  - 最近重构性：样本点到一个线性超平面的距离都足够近；

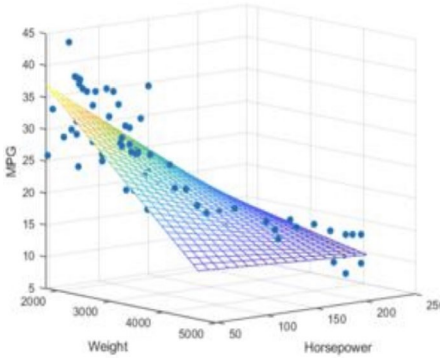
4. 最近重构性

PCA（2）

假定有m个样本，目标是获得样本在$d'(d' \leq d)$维空间中的表示 $Z \in \Re^{d' \times m}$，样本投影到一个线性超平面能尽可能近

教材P74

- 这是从回归分析的角度看待PCA，即求解一个线性函数更好地拟合样本点集合。这就使得我们的优化目标从方差最大转化为平方误差最小，因为映射距离越短，丢失的信息也会越小



主成分分析

4. 最近重构性

目标：未来使得样本点到超平面的距离足够近

- 假如数据集是n维的，共有m个数据$(x^{(1)}, x^{(2)}, \cdots, x^{(m)})$,原坐标系为$\{w_1, w_2, \cdots, w_n\}$，其中$w$是标准正交基，即$||w||_2 = 1$，$w_i^T w_j = 0$。
- 将数据从n维降到p维,则新的坐标系为$\{w_1, w_2, \cdots, w_p\}$，样本点$x^{(i)}$在$p$维坐标系中的投影为：$z^{(i)} = (z_1^{(i)}, z_2^{(i)}, \cdots, z_p^{(i)})^T$。其中$z_j^{(i)} = w_i^T x^{(i)}$，是$x^{(i)}$在低维坐标系里第$j$维的坐标

如果用$z^{(i)}$来恢复原始数据$x^{(i)}$，则得到的恢复数据$\bar{x}^{(i)} = \sum_{j=1}^{p} z_j^{(i)} w_j = W z^{(i)}$，其中，$W$为标准正交基组成的矩阵，维度为$n \times p$，可以理解为$p$个维度为$n$的基向量。我们希望得到所有样本带这个超平面的距离足够近，即最小化下式

主成分分析

4. 最近重构性

$$
\begin{aligned}
\sum_{i=1}^{m}\left\|\bar{x}^{(i)}-x^{(i)}\right\|_2^2 &= \sum_{i=1}^{m}\left\|Wz^{(i)}-x^{(i)}\right\|_2^2 \\
&= \sum_{i=1}^{m}\left(Wz^{(i)}\right)^T\left(Wz^{(i)}\right)-2\sum_{i=1}^{m}\left(Wz^{(i)}\right)^T x^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)} \\
&= \sum_{i=1}^{m}z^{(i)T}z^{(i)}-2\sum_{i=1}^{m}z^{(i)T}W^T x^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)} \\
&= \sum_{i=1}^{m}z^{(i)T}z^{(i)}-2\sum_{i=1}^{m}z^{(i)T}z^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)} \\
&= -\sum_{i=1}^{m}z^{(i)T}z^{(i)}+\sum_{i=1}^{m}x^{(i)T}x^{(i)} \\
&= -tr\left(W^T\left(\sum_{i=1}^{m}x^{(i)}x^{(i)T}\right)W\right)+\sum_{i=1}^{m}x^{(i)T}x^{(i)} \\
&= -tr\left(W^T X X^T W\right)+\sum_{i=1}^{m}x^{(i)T}x^{(i)}
\end{aligned}
$$

推导过程

主成分分析

4. 最近重构性

- 因此，PCA求解的过程可以用目标函数描述为：

$$\min_{W} -tr(W^T XX^T W)$$
$$s.t. \quad W^T W = I$$

和最大可分性的推导一致

PCA实例

数据：$\begin{pmatrix} -1 & -1 & 0 & 2 & 0 \\ -2 & 0 & 0 & 1 & 1 \end{pmatrix}$

协方差矩阵：$C = \frac{1}{5}\begin{pmatrix} -1 & -1 & 0 & 2 & 0 \\ -2 & 0 & 0 & 1 & 1 \end{pmatrix}\begin{pmatrix} -1 & -2 \\ -1 & 0 \\ 0 & 0 \\ 2 & 1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{6}{5} & \frac{4}{5} \\ \frac{4}{5} & \frac{6}{5} \end{pmatrix}$

特征值：$\lambda_1 = 2, \lambda_2 = 2/5$　　特征向量：$c_1\begin{pmatrix} 1 \\ 1 \end{pmatrix}, c_2\begin{pmatrix} -1 \\ 1 \end{pmatrix}$

对角化：$PCP^{\mathsf{T}} = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}\begin{pmatrix} 6/5 & 4/5 \\ 4/5 & 6/5 \end{pmatrix}\begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 2/5 \end{pmatrix}$

降维：$Y = (1/\sqrt{2} \quad 1/\sqrt{2})\begin{pmatrix} -1 & -1 & 0 & 2 & 0 \\ -2 & 0 & 0 & 1 & 1 \end{pmatrix} = (-3/\sqrt{2} \quad -1/\sqrt{2} \quad 0 \quad 3/\sqrt{2} \quad 1/\sqrt{2})$

一个PCA降维的例子

主成分分析

5. 在相关矩阵上做PCA

- 两个变量之间的相关系数定义为两个变量的协方差除以它们标准差的乘积

$$\rho_{X_i,X_j} = \frac{\sigma_{X_i,X_j}}{\sqrt{\sigma_{X_i,X_i}\sigma_{X_j,X_j}}}$$

- 是一种剔除了两个变量量纲影响、标准化后的特殊协方差。它消除了两个变量变化幅度的影响，而只是单纯反应两个变量每单位变化时的相似程度。很多ML工具软件如SPSS将相关系数矩阵替代协方差矩阵，来做PCA

教材P73

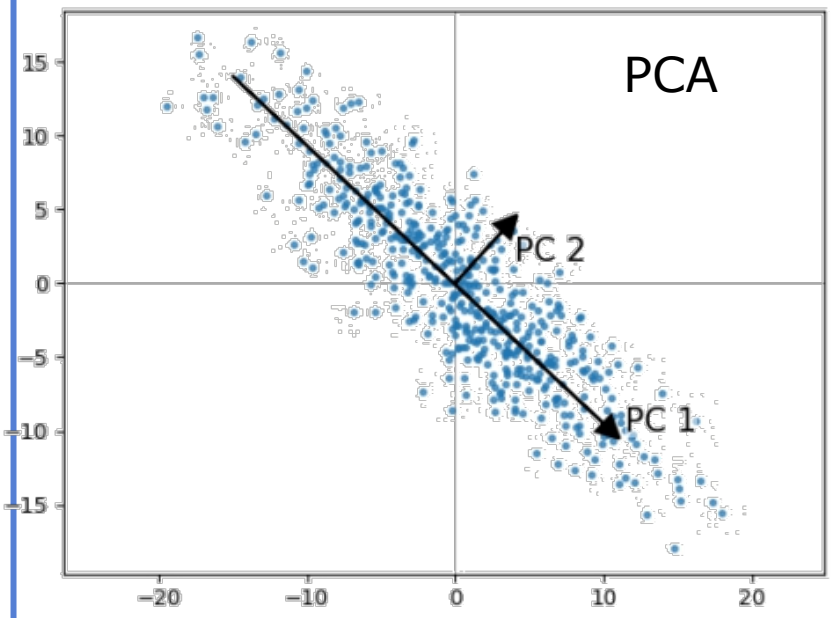
- 在相关矩阵而不是协方差矩阵上做PCA等价于用标准欧氏距离做MDS

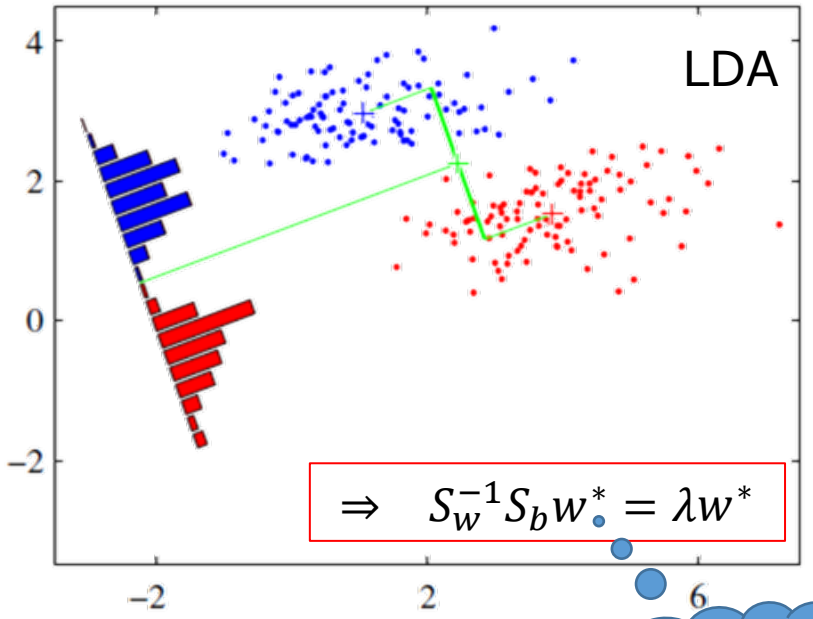
教材P81

第七讲 维度归约

主成分分析

6. PCA与LDA的比较





还记得这个式子吗?

(1) LDA是有监督的降维方法，保持了类信息。PCA是无监督的

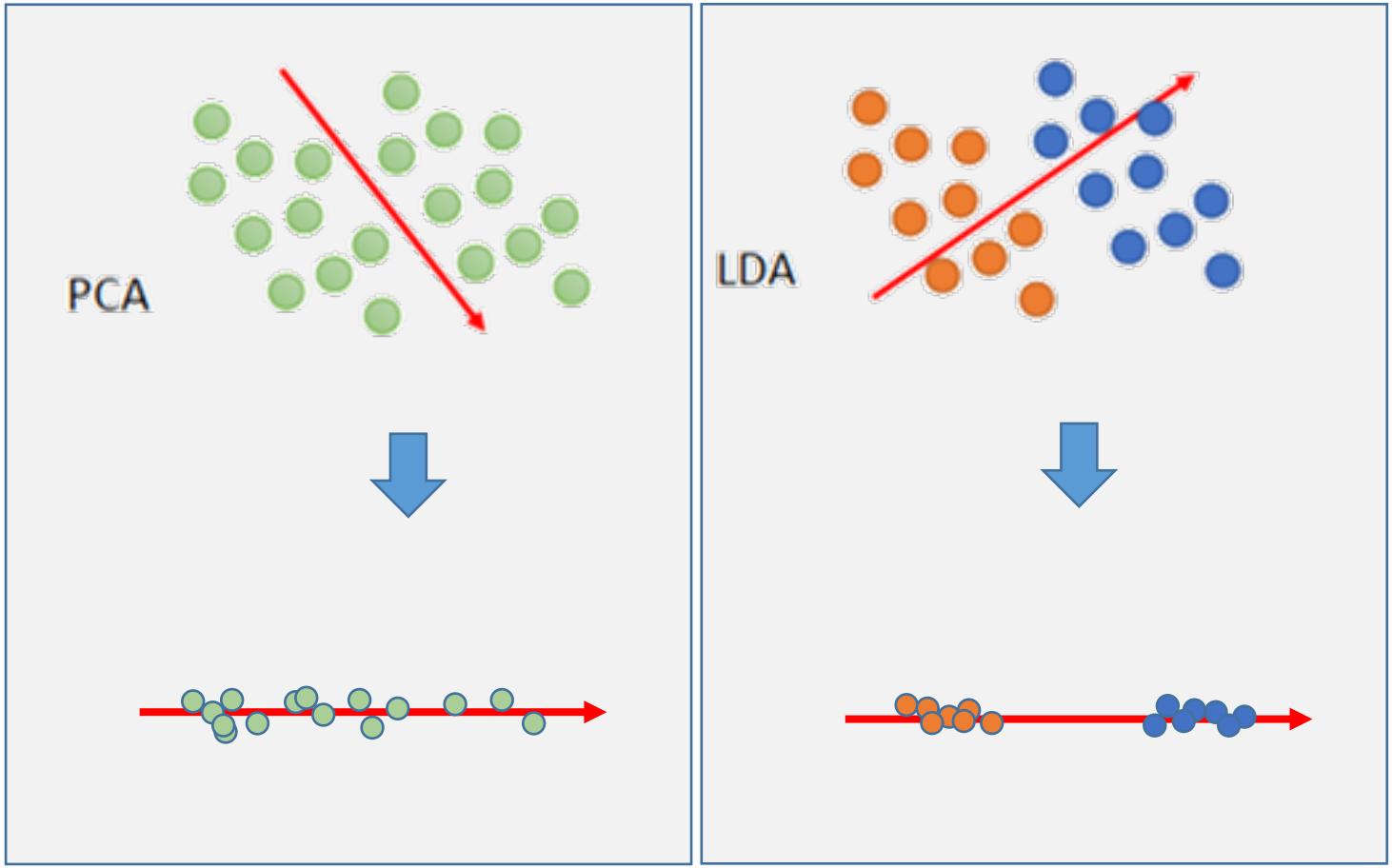
(2) LDA降维最多降到类别数K-1的维数，PCA没有这个限制

(3) LDA更依赖均值，如果样本信息更依赖方差的话，效果将没有PCA好

主成分分析

6. PCA与LDA的比较

教材P83

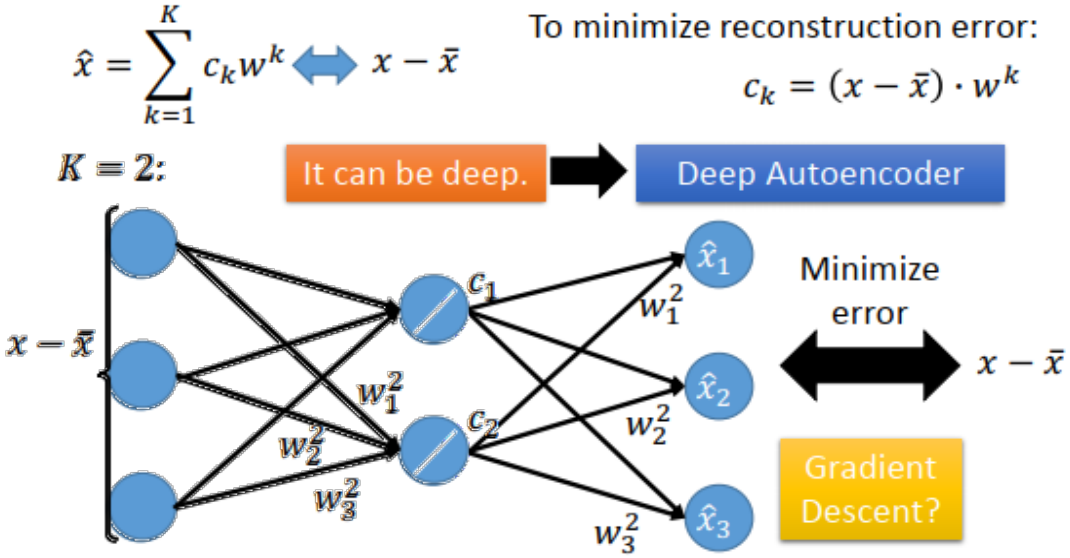


提问：绘制一个二维数据集示意图，让PCA和LDA投影效果都好，或者都不好

主成分分析

6. PCA与感知机

- PCA可以表示成一个神经网络，只有一个激活函数为线性的隐藏层，训练的目标是让输入和输出越接近越好。这样德神经网络又叫做自编码器（autoencoder），也是一种降维工具



PCA得到的基向量可以让重构误差最小，但是用神经网络不一定能找出来，不可能让重构误差比PCA找到的还要小。在线性情况下，使用PCA找w比较快，使用神经网络则比较麻烦。但是使用神经网络的好处是用多层感知机，这个就是deep autoencoder

主成分分析

课后练习题： 试用PCA变换做一维特征提取

$$\omega_1:\{(-5,-5)',(-5,-4)',(-4,-5)',(-5,-6)',(-6,-5)'\}$$

$$\omega_2:\{(5,5)',(5,6)',(6,5)',(5,4)',(4,5)'\}$$

解： (1) $\vec{m}=\frac{1}{5}\sum_{i=1}^{5}\vec{x}_i^{(1)}+\frac{1}{5}\sum_{i=1}^{5}\vec{x}_i^{(2)}=\vec{0}\quad\because \hat{P}(\omega_1)=\hat{P}(\omega_2)=5/10=1/2$

(2)

$$\therefore R=E[\vec{x}\vec{x}']=\sum_{i=1}^{2}\hat{P}(\omega_i)E[\vec{x}^{(i)}\vec{x}^{(i)}']=\frac{1}{2}[\frac{1}{5}\sum_{i=1}^{5}\vec{x}_i^{(1)}\vec{x}_i^{(1)}']+\frac{1}{2}[\frac{1}{5}\sum_{i=1}^{5}\vec{x}_i^{(2)}\vec{x}_i^{(2)}']$$

$$=\begin{pmatrix}25.4 & 25\\ 25 & 25.4\end{pmatrix}$$

(3) **求R的特征值、特征矢量**

$$|R-\lambda I|=(25.4-\lambda)^2-25^2=0\ \Rightarrow\ \lambda_1=50.4,\ \lambda_2=0.4$$

$$R\vec{t}_j=\lambda_j\vec{t}_j,j=1,2\ \Rightarrow\ \vec{t}_1=\frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix},\ \vec{t}_2=\frac{1}{\sqrt{2}}\begin{pmatrix}1\\-1\end{pmatrix}$$

主成分分析

(4) 选λ_1对应的 $\vec{t}_1$ 作为变换矩阵 $\longrightarrow$ $T=[\vec{t}_1]=\frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix}$

由 $y=T'\vec{x}$ 得变换后的一维模式特征为

$$y_1^{(1)}=T'\vec{x}_1^{(1)}=\frac{1}{\sqrt{2}}(1,1)\begin{pmatrix}-5\\-5\end{pmatrix}=-\frac{10}{\sqrt{2}}$$

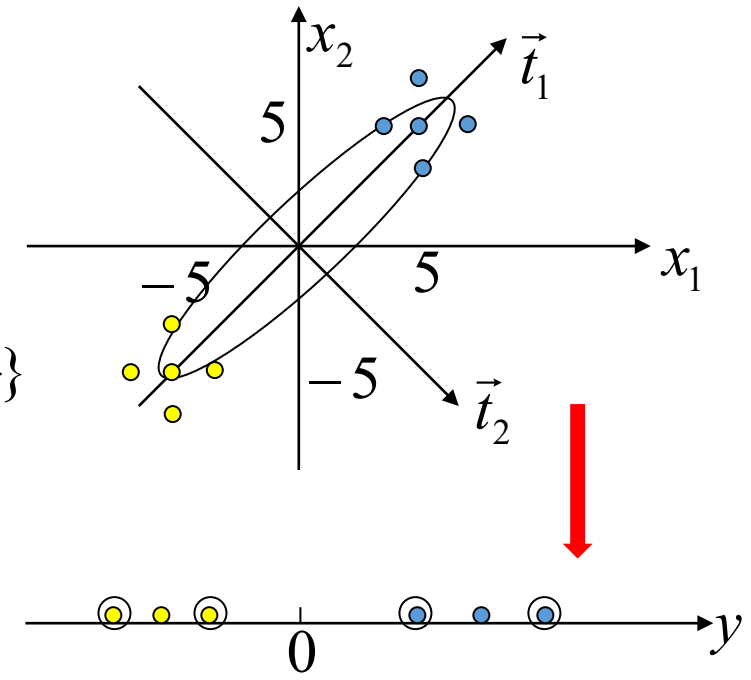
$$\vdots$$

$$y_5^{(1)}=T'\vec{x}_5^{(1)}=-\frac{11}{\sqrt{2}}$$

得

$$\omega_1:\{-\frac{10}{\sqrt{2}},-\frac{9}{\sqrt{2}},-\frac{9}{\sqrt{2}},-\frac{11}{\sqrt{2}},-\frac{11}{\sqrt{2}}\}$$

$$\omega_2:\{\frac{10}{\sqrt{2}},\frac{11}{\sqrt{2}},\frac{11}{\sqrt{2}},\frac{9}{\sqrt{2}},\frac{9}{\sqrt{2}}\}$$



从图中可得，用 $\vec{t}_1$ 作为变换矩阵，就是原模式在其上的投影。

维度归约

1. K近邻分类器（自学）
2. 低维嵌入与多维标定（MDS）
3. 主成分分析（PCA）
4. 流形学习

第七讲 维度归约

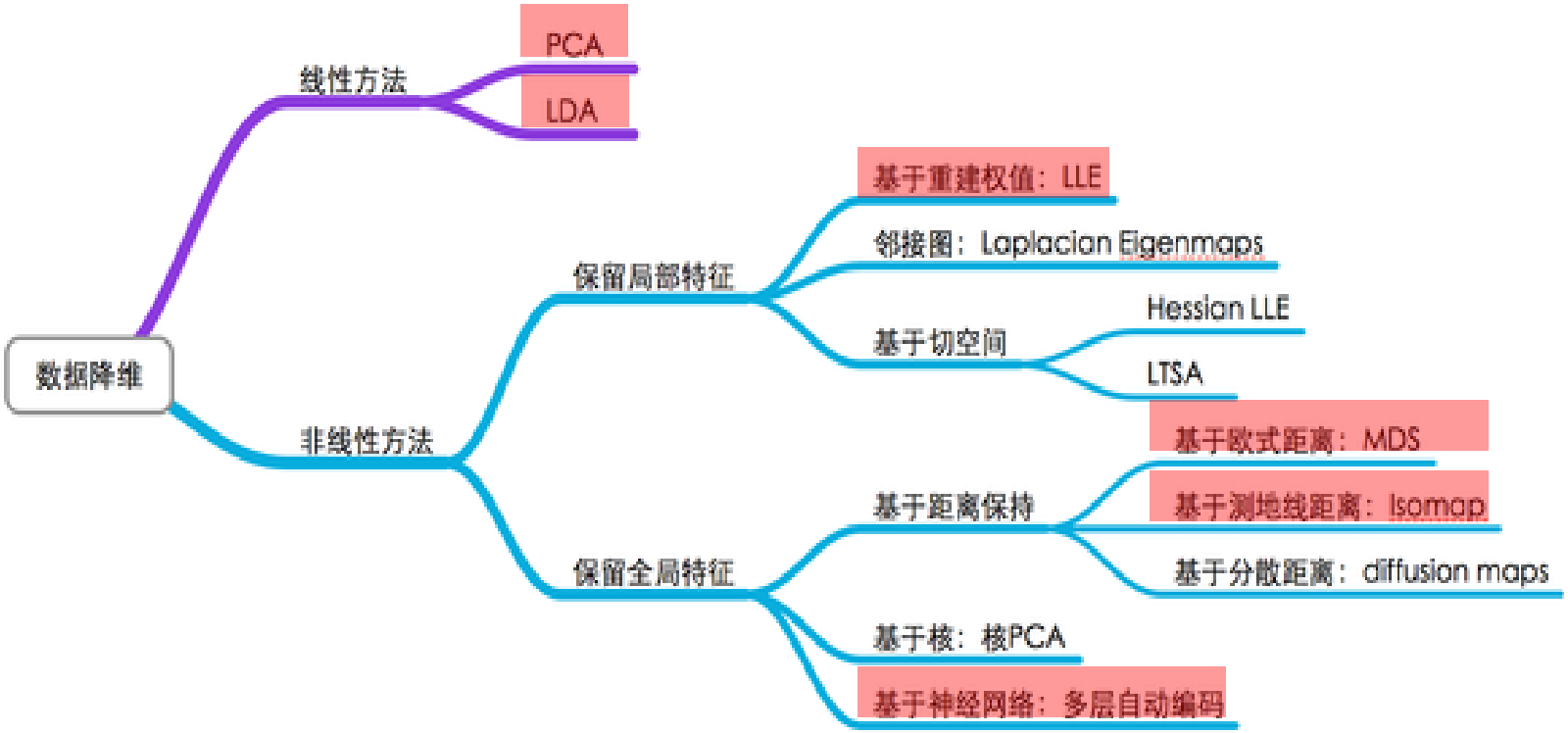
流形学习

- 流形学习方法(Manifold Learning)，简称流形学习。假设数据是均匀采样于一个高维欧氏空间中的低维流形，流形学习就是从高维采样数据中恢复低维流形结构，即找到高维空间中的低维流形，并求出相应的嵌入映射，以实现维数约简或者数据可视化
- “流形”是在局部与欧氏空间同胚的空间，换言之，它在局部具有欧氏空间的性质，能用欧氏距离来进行距离计算。若低维流形嵌入到高维空间中，则数据样本在高维空间的分布虽然看上去非常复杂，但在局部上仍具有欧氏空间的性质，因此，可以容易地在局部建立降维映射关系，然后再设法将局部映射关系推广到全局

第七讲 维度归约

流形学习

- 线性的流形学习方法：主成分分析（PCA），线性判别分析（LDA）；
- 非线性的流形学习方法：等距映射（Isomap）、拉普拉斯特征映射（Laplacian eigenmaps，LE）、局部线性嵌入(Locally-linear embedding，LLE)。



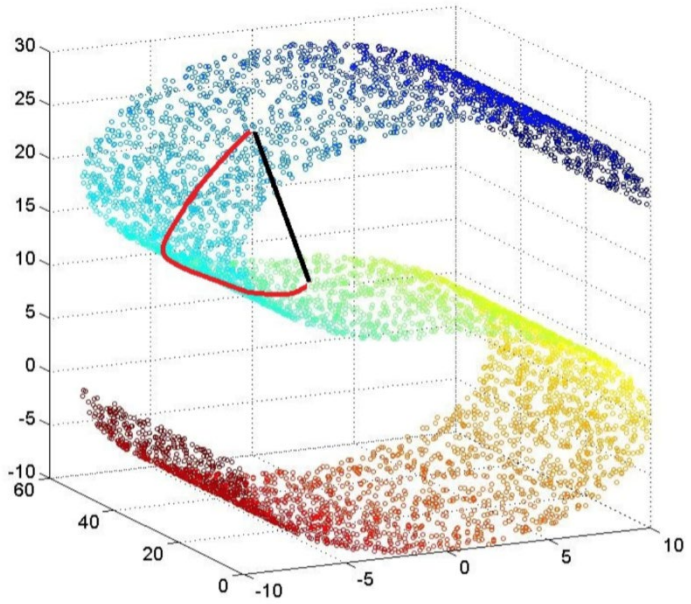
第七讲 维度归约

流形学习

1. 等度量映射(Isometric Mapping，Isomap)

教材P83

- 低维流形嵌入到高维空间之后，直接在高维空间中计算直线距离具有误导性，因为高维空间中的直线距离在低维嵌入流形上不可达。而低维嵌入流形上两点间的本真距离是“测地线”(geodesic)距离



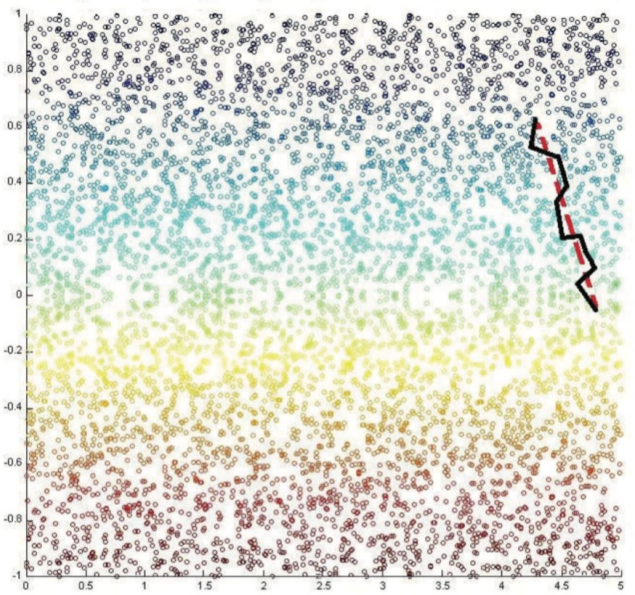
(a) 测地线距离与高维直线距离

- 它所采用的核心算法和MDS是一致的，区别在于原始空间中的距离矩阵的计算上：利用流形在局部上与欧氏空间同胚性，计算近邻距离

流形学习

1. 等度量映射(Isometric Mapping，Isomap)

- 对每个点基于欧氏距离找出其近邻点，然后就能建立一个近邻连接图，图中近邻点之间存在连接，而非近邻点之间不存在连接，于是，计算两点之间测地线距离的问题，就转变为计算近邻连接图上两点之间的最短路径问题



(b) 测地线距离与近邻距离

- 对于**近邻图**的构建，常用的有两种方法：
  - **一种是指定近邻点个数**，像kNN一样选取k个最近的邻居；
  - **另一种是指定邻域半径**，距离小于该阈值的被认为是它的近邻点

流形学习

1. 等度量映射(Isometric Mapping，Isomap)

输入： 样本集 $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$;
近邻参数 k;
低维空间维数 d'.

过程：

样本集形成一张可达图

1: **for** $i = 1, 2, \ldots, m$ **do**
2: 确定 $\boldsymbol{x}_i$ 的 k 近邻;
3: $\boldsymbol{x}_i$ 与 k 近邻点之间的距离设置为欧氏距离, 与其他点的距离设置为无穷大;
4: **end for**
5: 调用最短路径算法计算任意两样本点之间的距离 $\text{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$; 内积矩阵B
6: 将 $\text{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ 作为 MDS 算法的输入;
7: **return** MDS 算法的输出 降维后的低维坐标

输出： 样本集 D 在低维空间的投影 $Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_m\}$.

流形学习

1. 等度量映射(Isometric Mapping，Isomap)

- Isomap仅是得到了训练样本在低维空间的坐标，对于新样本，如何将其映射到低维空间呢?

解决方案：将训练样本高维空间坐标作为输入，低维空间坐标作为输出，训练一个回归学习器来对新样本的低维空间坐标进行预测。或者使用N+1个实例重新允许整个算法

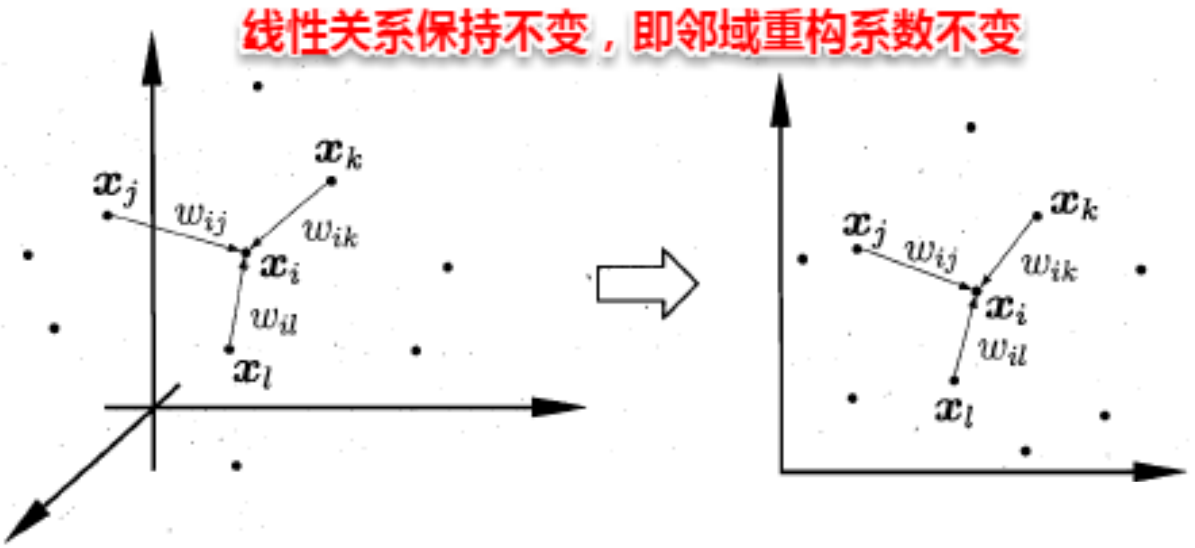
教材P87

流形学习

2. 局部线性嵌入 (Locally Linear Embedding，LLE)

- 不同于Isomap算法去保持邻域距离，LLE算法试图去保持邻域内的线性关系，并使得该线性关系在降维后的空间中继续保持
- 假定样本x_i的坐标可以通过它的邻域样本x_j，x_k，x_l线性表出：

$$\boldsymbol{x}_i = w_{ij}\boldsymbol{x}_j + w_{ik}\boldsymbol{x}_k + w_{il}\boldsymbol{x}_l$$



流形学习

2. 局部线性嵌入 (Locally Linear Embedding，LLE)

- LLE算法分为两步走，首先第一步根据近邻关系计算出所有样本的邻域重构系数w：

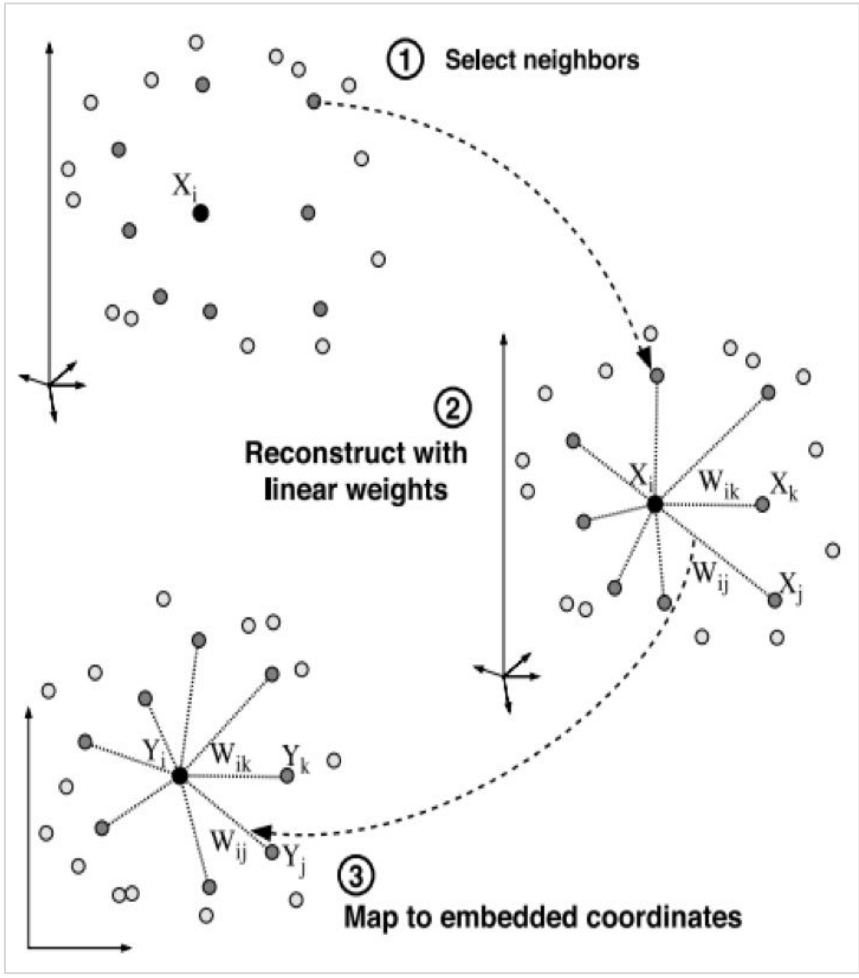
$$\min_{w_1,w_2,\ldots,w_m} \sum_{i=1}^{m} \left\| \boldsymbol{x}_i - \sum_{j\in Q_i} w_{ij}\boldsymbol{x}_j \right\|_2^2$$

$$\text{s.t.} \sum_{j\in Q_i} w_{ij} = 1,$$

$\Rightarrow$

令 $C_{jk} = (\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}}(\boldsymbol{x}_i - \boldsymbol{x}_k)$, w_{ij} 有闭式解

$$w_{ij} = \frac{\sum_{k\in Q_i} C_{jk}^{-1}}{\sum_{l,s\in Q_i} C_{ls}^{-1}}$$



流形学习

2. 局部线性嵌入 (Locally Linear Embedding，LLE)

- 接着根据邻域重构系数不变，去求解低维坐标

$$\min_{z_1,z_2,\ldots,z_m} \sum_{i=1}^{m} \left\| z_i - \sum_{j \in Q_i} w_{ij} z_j \right\|_2^2$$

$$令\ \mathbf{Z} = (z_1, z_2, \ldots, z_m) \in \mathbb{R}^{d' \times m},\ (\mathbf{W})_{ij} = w_{ij}$$

- 构造一个M矩阵 $\mathbf{M} = (\mathbf{I} - \mathbf{W})^{\mathrm{T}}(\mathbf{I} - \mathbf{W})$，则上述优化问题转化为：

$$\min_{\mathbf{Z}} \ \mathrm{tr}(\mathbf{Z}\mathbf{M}\mathbf{Z}^{\mathrm{T}})$$

$$\text{s.t.} \ \ \mathbf{Z}\mathbf{Z}^{\mathrm{T}} = \mathbf{I}$$

特征值分解又来了~

流形学习

2. 局部线性嵌入 (Locally Linear Embedding，LLE)

- M特征值分解后最小的d'个特征值对应的特征向量组成Z，LLE算法的具体流程如下图所示

输入： 样本集 $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$;
近邻参数 k;
低维空间维数 d'.

过程：

1: **for** $i = 1, 2, \ldots, m$ **do**
2: 确定 $\boldsymbol{x}_i$ 的 k 近邻;
3: 求得 w_{ij}, $j \in Q_i$;
4: 对于 $j \notin Q_i$, 令 $w_{ij} = 0$;
5: **end for**
6: 得到 M;
7: 对 M 进行特征值分解;
8: **return** M 的最小 d' 个特征值对应的特征向量

输出： 样本集 D 在低维空间的投影 $Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_m\}$.

局部(邻域)线性关系保持不变

和Isomap一样只得到了低维坐标

第七讲 维度归约

总结

- 降维是将原高维空间嵌入到一个合适的低维子空间中，接着在低维空间中进行学习任务
- 降维的方法可以是监督式的，也可以是非监督式的；有线性降维和非线性降维。它们都是特征提取的方法

——也许大家最后心存疑惑，那kNN呢，为什么一开头就说了kNN算法？正是因为在降维算法中，低维子空间的维数d'通常都由人为指定，因此我们需要使用一些低开销的学习器来选取合适的d'，kNN这家伙懒到家了根本无心学习，在训练阶段开销为零，测试阶段也只是遍历计算了距离，因此拿kNN来进行交叉验证就十分有优势了~同时降维后样本密度增大同时距离计算变易，更为kNN来展示它独特的十八般手艺提供了用武之地