机器学习

08 降维与度量学习

李祎

liyi@dlut.edu.cn

大连理工大学 人工智能学院

School of Artificial Intelligence, Dalian University of Technology

目录

kNN

- k近邻(kNN)学习是一种常用的监督学习方法：
  - 确定训练样本，以及某种距离度量。
  - 对于某个给定的测试样本，找到**训练集中距离最近的k个样本**，对于分类问题使用“投票法”获得预测结果，对于回归问题使用“平均法”获得预测结果。还可基于距离远近进行加权平均或加权投票，距离越近的样本权重越大。

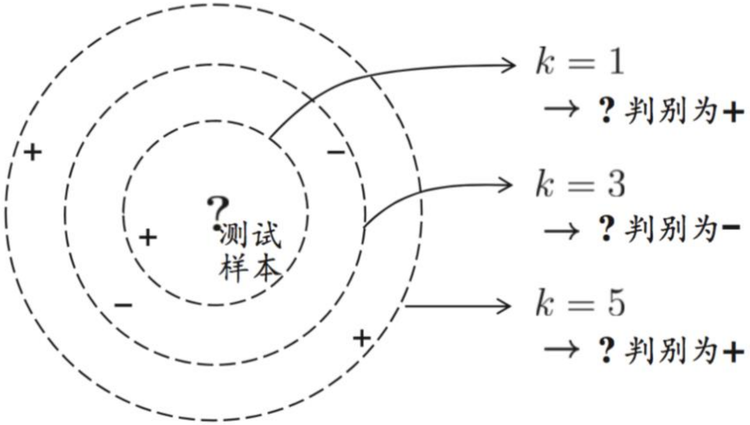


图 10.1 k 近邻分类器示意图. 虚线显示出等距线; 测试样本在 $k=1$ 或 $k=5$ 时被判别为正例, $k=3$ 时被判别为反例.

“懒惰学习”与“急切学习”

KNN没有显式的训练过程，属于“懒惰学习”

- “懒惰学习” (lazy learning): 此类学习技术在训练阶段仅仅是把样本保存起来，训练时间开销为零，待收到测试样本后再进行处理。
- “急切学习” (eager learning): 在训练阶段就对样本进行学习处理的方法。

kNN

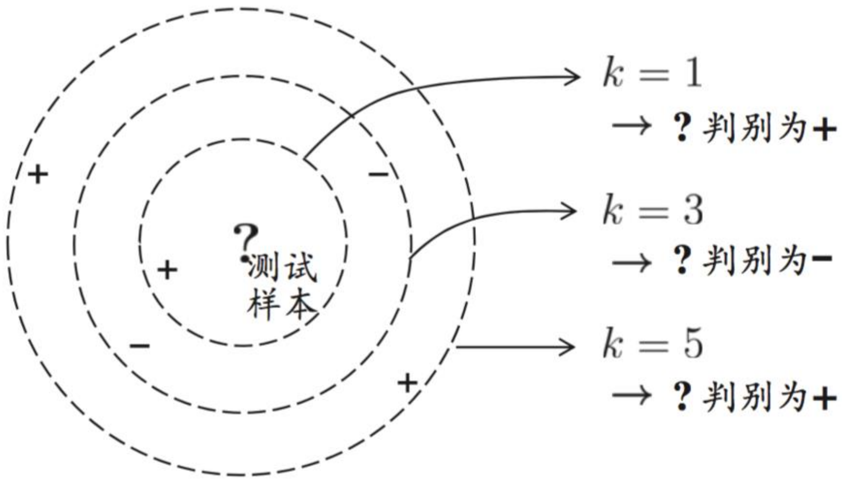


图 10.1 k 近邻分类器示意图. 虚线显示出等距线; 测试样本在 $k=1$ 或 $k=5$ 时被判别为正例, $k=3$ 时被判别为反例.

- k近邻分类器中的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。

kNN

- 分析1NN二分类错误率
- 暂且假设距离计算是“恰当”的，即能够恰当地找出k个近邻，我们来对“最近邻分类器”（1NN，即k=1）在二分类问题上的性能做一个简单的讨论。
- 给定测试样本$\boldsymbol{x}$，若其最近邻样本为$\boldsymbol{z}$，则最近邻出错的概率就是$\boldsymbol{x}$与$\boldsymbol{z}$类别标记不同的概率，即

$$P(err) = 1 - \sum_{c \in \mathcal{Y}} P(c|\boldsymbol{x})P(c|\boldsymbol{z})$$

kNN

- 假设样本独立同分布，且对任意$\boldsymbol{x}$和任意小正整数δ，在$\boldsymbol{x}$附近δ距离范围内总能找到一个训练样本；换言之，对任意测试样本，总能在任意近的范围内找到 $P(err) = 1 - \sum_{c \in \mathcal{Y}} P(c|\boldsymbol{x})P(c|\boldsymbol{z})$ 中的训练样本z。

- 令 $c^* = \arg\max_{c \in \mathcal{Y}} P(c|\boldsymbol{x})$ 表示贝叶斯最优分类器的结果，有

$$
\begin{aligned}
P(err) &= 1 - \sum_{c \in \mathcal{Y}} P(c|\boldsymbol{x})P(c|\boldsymbol{z}) \simeq 1 - \sum_{c \in \mathcal{Y}} P^2(c|\boldsymbol{x}) \\
&\leq 1 - P^2(c^*|\boldsymbol{x}) = (1 + P(c^*|\boldsymbol{x}))(1 - P(c^*|\boldsymbol{x})) \\
&\leq 2 \times (1 - P(c^*|\boldsymbol{x})).
\end{aligned}
$$

最近邻分类虽简单，但它的泛化错误率不超过
贝叶斯最优分类器错误率的两倍！

维数灾难

- 上述讨论基于一个重要的假设：任意测试样本$\boldsymbol{x}$附近的任意小的δ距离范围内总能找到一个训练样本，即训练样本的采样密度足够大，或称为“密采样”。然而，这个假设在现实任务中通常很难满足：
  - 若属性维数为1，当δ=0.001，仅考虑单个属性，则仅需1000个样本点平均分布在归一化后的属性取值范围内。
  - 若属性维数为20，若样本满足密采样条件，则至少需要$(10^3)^{20}=10^{60}$个样本。
- 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”。

线性降维方法：

主成分分析（PCA）

线性降维方法

- 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$\mathbf{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m) \in \mathbb{R}^{d \times m}$，变换之后得到$d' \leq d$维空间中的样本

$$\mathbf{Z} = \mathbf{W}^{\mathrm{T}}\mathbf{X},$$

其中$\mathbf{W} \in \mathbb{R}^{d \times d'}$是变换矩阵，$\mathbf{Z} \in \mathbb{R}^{d' \times m}$是样本在新空间中的表达。

- 变换矩阵$\mathbf{W}$可视为d'个d维属性向量。换言之，$\boldsymbol{z}_i$是原属性向量$\boldsymbol{x}_i$在新坐标系$\{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'}\}$中的坐标向量。若$\boldsymbol{w}_i$与$\boldsymbol{w}_j (i \neq j)$正交，则新坐标系是一个正交坐标系，此时$\mathbf{W}$为正交变换。显然，新空间中的属性是原空间中的属性的线性组合。

线性降维方法

- 基于线性变换来进行降维的方法称为线性降维方法，对低维子空间性质的不同要求可通过对 **W** 施加不同的约束来实现。
- 分类：
  - 线性有监督： LDA
  - 线性无监督：PCA
- 优点：
  - 1.对线性结构分布的数据集有较好的降维效果；
  - 2.在压缩、降噪以及数据可视化等方面非常有效的。
  - 3.计算简单，易于理解
- 缺点：
  - 对呈现出结构非线性或属性强相关性的数据集，无法发现复杂的非线性数据的内在本质结构。

主成分分析

- 对于正交属性空间中的样本点，如何用一个超平面对所有样本进行恰当的表达？

- 容易想到，若存在这样的超平面，那么它大概应具有这样的性质：
  - 最近重构性：样本点到这个超平面的距离都足够近；
  - 最大可分性：样本点在这个超平面上的投影能尽可能分开。

- 基于最近重构性和最大可分性，能分别得到主成分分析的两种等价推导。

主成分分析：最近重构性

- 对样本进行中心化，$\sum_i \boldsymbol{x}_i = \mathbf{0}$，再假定投影变换后得到的新坐标系为 $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_d\}$ ，其中 $\boldsymbol{w}_i$是标准正交基向量，

$$||\boldsymbol{w}_i||_2 = 1, \boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{w}_j = 0 (i \neq j).$$

- 若丢弃新坐标系中的部分坐标，即将维度降低到 $d' < d$ ，则样本点在低维坐标系中的投影是 $\boldsymbol{z}_i = (z_{i1}; z_{i2}; \ldots; z_{id'})$ ，而 $z_{ij} = \boldsymbol{w}_j^{\mathrm{T}} \boldsymbol{x}_i$ 是 $\boldsymbol{x}_i$ 在低维坐标下第 j 维的坐标，若基于$\boldsymbol{z}_i$来重构$\boldsymbol{x}_i$，则会得到

$$\hat{\boldsymbol{x}}_i = \sum_{j=1}^{d'} z_{ij} \boldsymbol{w}_j.$$

主成分分析：最近重构性

- 考虑整个训练集，原样本点 $\boldsymbol{x}_i$与基于投影重构的样本点 $\hat{\boldsymbol{x}}_i$之间的距离为

$$\sum_{i=1}^{m}\left\|\sum_{j=1}^{d'} z_{ij}\boldsymbol{w}_j - \boldsymbol{x}_i\right\|_2^2 = \sum_{i=1}^{m}\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{z}_i - 2\sum_{i=1}^{m}\boldsymbol{z}_i^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\boldsymbol{x}_i + \text{const}$$

$$\propto -\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}}\left(\sum_{i=1}^{m}\boldsymbol{x}_i\boldsymbol{x}_i^{\mathrm{T}}\right)\mathbf{W}\right).$$

- 根据最近重构性应最小化上式。考虑到 $\boldsymbol{w}_j$ 是标准正交基，$\sum_i \boldsymbol{x}_i\boldsymbol{x}_i^{\mathrm{T}}$ 是协方差矩阵，有

$$\min_{\mathbf{W}} \quad -\operatorname{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W})$$
$$\text{s.t.} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}.$$

PCA的优化目标

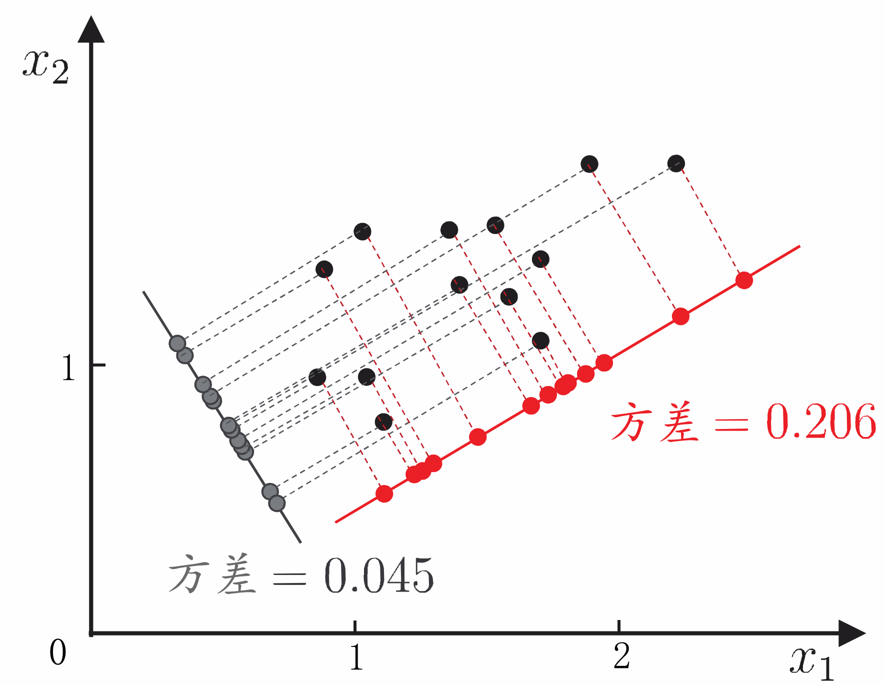
主成分分析：最大可分性

- 样本点$\boldsymbol{x}_i$在新空间中超平面上的投影是$\mathbf{W}^{\mathrm{T}}\boldsymbol{x}_i$，若所有样本点的投影能尽可能分开，则应该使得投影后样本点的方差最大化。若投影后样本点的方差是$\sum_i \mathbf{W}^{\mathrm{T}}\boldsymbol{x}_i\boldsymbol{x}_i^{\mathrm{T}}\mathbf{W}$，于是优化目标可写为

$$\max_{\mathbf{W}} \quad \mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W})$$
$$\text{s.t.} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}.$$

等价

$$\min_{\mathbf{W}} \quad -\mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W})$$
$$\text{s.t.} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}.$$



主成分分析：求解

- 对优化式使用拉格朗日乘子法可得

$$\max_{\mathbf{W}} \quad \mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}) \qquad \text{s.t.} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}.$$

$$\Leftrightarrow$$

$$\min_{\mathbf{W}} \quad -\mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}) \qquad \text{s.t.} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}.$$

$$\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} = \lambda\mathbf{W}.$$

只需对协方差矩阵 $\mathbf{X}\mathbf{X}^{\mathrm{T}}$进行特征值分解，并将求得的特征值排序：$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$ ，再取前 d'个特征值对应的特征向量构成 $\mathbf{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'})$，这就是主成分分析的解。

详细推导：https://zhuanlan.zhihu.com/p/77151308

主成分分析

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 $\mathbf{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'})$.

图 10.5 PCA 算法

主成分分析

- 降维后低维空间的维数d'通常是由用户事先指定，或通过在d'值不同的低维空间中对k近邻分类器（或其它开销较小的学习器）进行交叉验证来选取较好的d'值。对PCA，还可从重构的角度设置一个重构阈值，例如$t=95\%$，然后选取使下式成立的最小d'值：

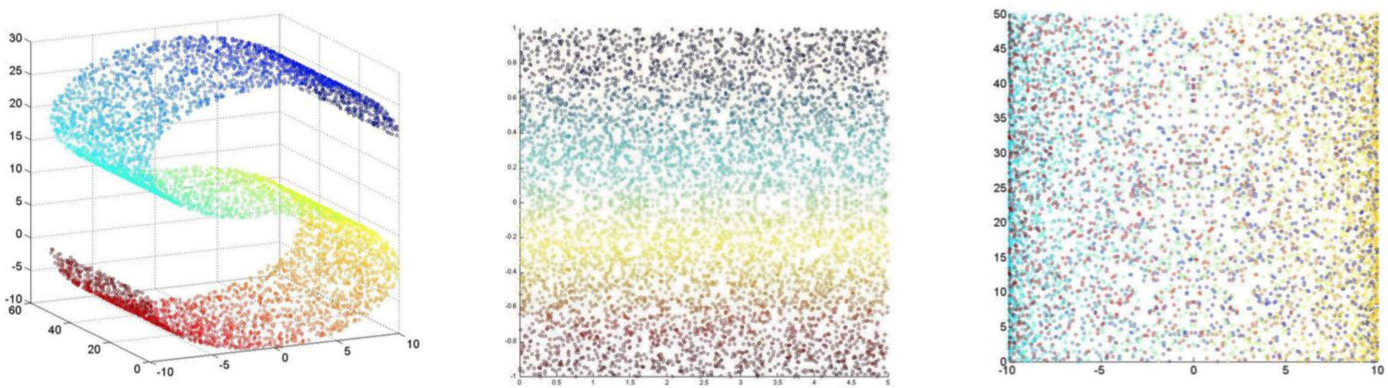
$$\frac{\sum_{i=1}^{d'}\lambda_i}{\sum_{i=1}^{d}\lambda_i} \geq t.$$

- 降维虽然会导致信息的损失，但一方面舍弃这些信息后能使得样本的采样密度增大，另一方面，当数据受到噪声影响时，最小的特征值所对应的特征向量往往与噪声有关，舍弃可以起到去噪效果。

核化线性降维：KPCA

核化线性降维

- 线性降维方法假设从高维空间到低维空间的函数映射是线性的，然而，在不少现实任务中，可能需要非线性映射才能找到恰当的低维嵌入：



(a) 三维空间中的观察　　(b) 本真二维结构　　(c) PCA 降维结果

图 10.6　三维空间中观察到的 3000 个样本点，是从本真二维空间中矩形区域采样后以 S 形曲面嵌入，此情形下线性降维会丢失低维结构．图中数据点的染色显示出低维空间的结构．

KPCA

- 非线性降维的一种常用方法，是基于核技巧对线性降维方法进行“核化”（kernelized)。
- 假定我们将在高维特征空间中把数据投影到由 $\mathbf{W}$ 确定的超平面上，即PCA欲求解

$$\left(\sum_{i=1}^{m} \boldsymbol{z}_i \boldsymbol{z}_i^{\mathrm{T}}\right) \mathbf{W} = \lambda \mathbf{W}.$$

- 其中$\boldsymbol{z}_i$ 是样本点 $\boldsymbol{x}_i$ 在高维特征空间中的像。令 $\boldsymbol{\alpha}_i = \frac{1}{\lambda} \boldsymbol{z}_i^{\mathrm{T}} \mathbf{W}$,

$$\mathbf{W} = \frac{1}{\lambda}\left(\sum_{i=1}^{m} \boldsymbol{z}_i \boldsymbol{z}_i^{\mathrm{T}}\right) \mathbf{W} = \sum_{i=1}^{m} \boldsymbol{z}_i \frac{\boldsymbol{z}_i^{\mathrm{T}} \mathbf{W}}{\lambda} = \sum_{i=1}^{m} \boldsymbol{z}_i \boldsymbol{\alpha}_i .$$

KPCA

- 假定$\boldsymbol{z}_i$是由原始属性空间中的样本点$\boldsymbol{x}_i$通过映射ϕ产生，即

$$\boldsymbol{z}_i = \phi(\boldsymbol{x}_i),\ i = 1, 2, \ldots, m.$$

$$\mathbf{W} = \sum_{i=1}^{m} \boldsymbol{z}_i \boldsymbol{\alpha}_i$$

- 若ϕ能被显式表达出来，则通过它将样本映射至高维空间，再在特征空间中实施PCA即可，即有

$$\left(\sum_{i=1}^{m} \phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^{\mathrm{T}}\right)\mathbf{W} = \lambda\mathbf{W}.$$

并且

$$\mathbf{W} = \sum_{i=1}^{m} \phi(\boldsymbol{x}_i)\boldsymbol{\alpha}_i.$$

KPCA

- 一般情形下，我们不清楚ϕ 的具体形式，于是引入核函数

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^{\mathrm{T}}\phi(\boldsymbol{x}_j).$$

- 又由 $\mathbf{W} = \sum_{i=1}^{m} \phi(\boldsymbol{x}_i)\boldsymbol{\alpha}_i$，代入优化式 $\left(\sum_{i=1}^{m} \phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^{\mathrm{T}}\right)\mathbf{W} = \lambda\mathbf{W}$，有

$$\mathbf{KA} = \lambda\mathbf{A}.$$

其中$\mathbf{K}$为κ对应的核矩阵, $(\mathbf{K})_{ij} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $\mathbf{A} = (\boldsymbol{\alpha}_1; \boldsymbol{\alpha}_2; \ldots; \boldsymbol{\alpha}_m)$.

- 上式为特征值分解问题，取$\mathbf{K}$最大的d'个特征值对应的特征向量得到解。

KPCA

- 对新样本$\boldsymbol{x}$，其投影后的第j $(j = 1, 2, \ldots, d')$维坐标为

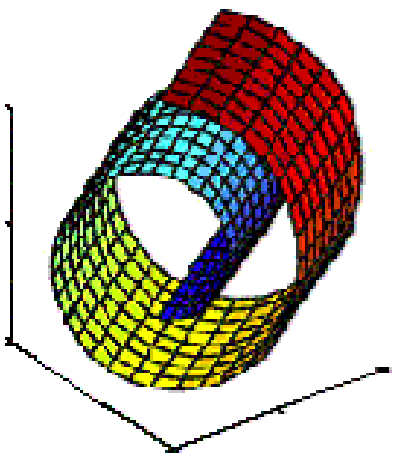
$$
\begin{aligned}
z_j = \boldsymbol{w}_j^{\mathrm{T}} \phi(\boldsymbol{x}) &= \sum_{i=1}^{m} \alpha_i^j \phi(\boldsymbol{x}_i)^{\mathrm{T}} \phi(\boldsymbol{x}) \\
&= \sum_{i=1}^{m} \alpha_i^j \kappa(\boldsymbol{x}_i, \boldsymbol{x}).
\end{aligned}
$$

其中$\boldsymbol{\alpha}_i$已经过规范化，α_i^j是$\boldsymbol{\alpha}_i$的第j个分量。由该式可知，为获得投影后的坐标，KPCA需对所有样本求和，因此它的计算开销较大。

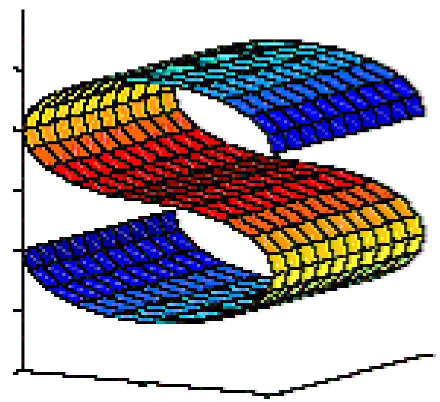
流形学习

流形学习

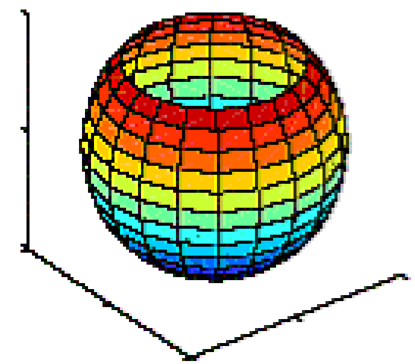
- “流形” 是线性子空间的一种非线性推广
- 拓扑学角度：局部区域线性，与低维欧式空间拓扑同胚
  - 在局部具有欧氏空间的性质，能用欧氏距离来进行距离计算



Swiss-roll



S-curve



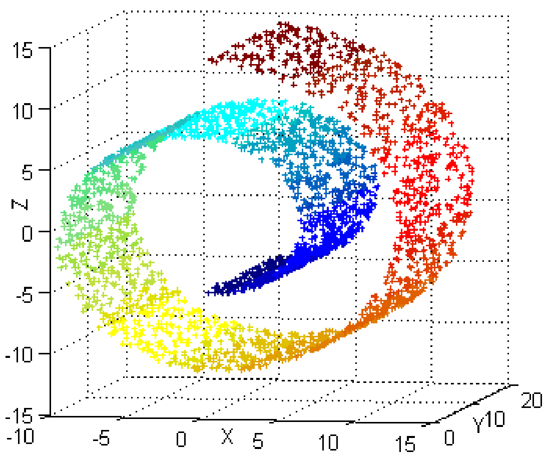
Fishbow

流形学习

- 流形学习(**Manifold Learning**)，**2000**年科学杂志**Science**首次提出。用于从高维采样数据恢复低维流形结构，*是一种非线性降维方法*。
- Seung HS, Lee DD. The manifold ways of perception. Science, 2000.
- 流形是感知的基础，人类的视觉记忆是以一种稳定的流形形式存贮在大脑中，人类具有捕获流形结构的能力；
- 流形学习可能是人类认知中一种自然的行为方式。

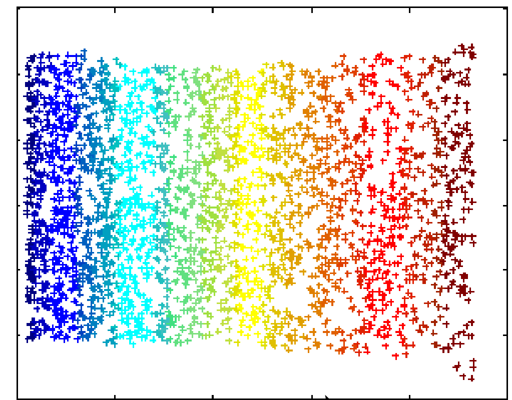
- 当维数被降至二维或三维时，能对数据进行可视化展示，因此流形学习也可被用于可视化。

流形学习



非线性降维

保持一定几何拓扑关系，如测地距离/邻域线性重构关系



高维数据空间
Data / Observation Space

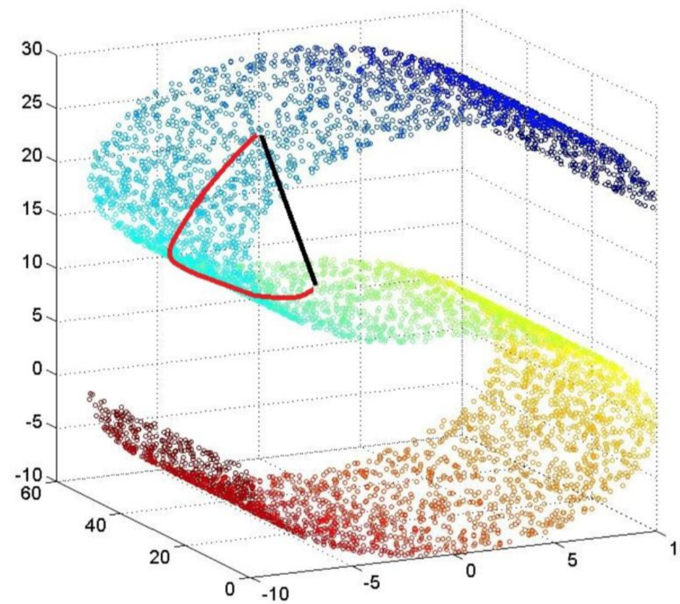
低维嵌入空间
Embedding / Coordinate Space

- 全局特性保持方法：Isomap
- 局部特性保持方法：LLE

Isomap

- 等度量映射(Isometric Mapping，Isomap)

- 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

Isomap

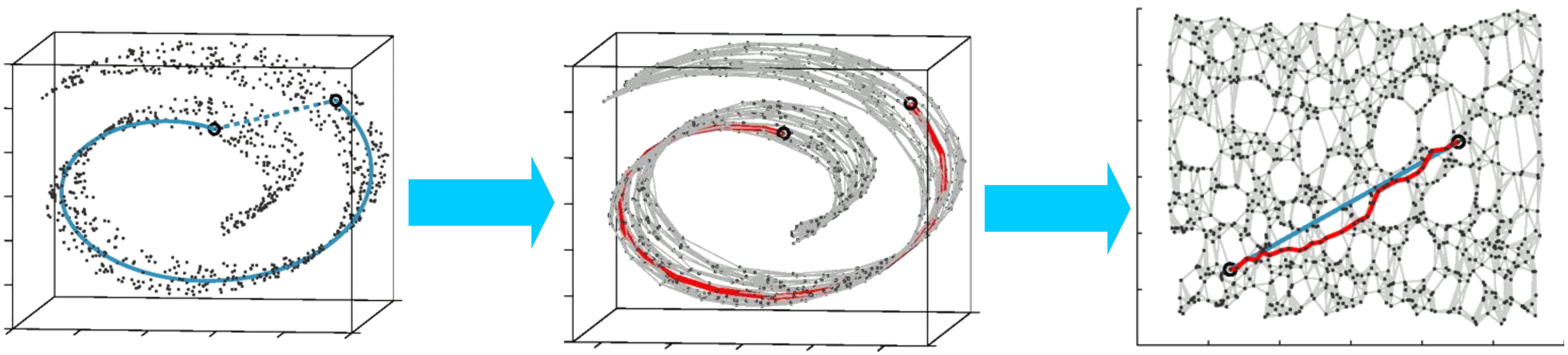
- 测地距离
  - ➢较近点对之间的测地距离用欧式距离代替
  - ➢较远点对之间的测地距离用最短路径来逼近

- 计算过程：
  - 利用流形在局部上与欧氏空间同胚这个性质，对每个点基于欧氏距离找出其近邻点
  - 建立一个近邻连接图，图种近邻点之间存在连接，而非近邻点之间不存在连接
  - 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。

- 最短路径的计算可通过Dijkstra算法或Floyd算法实现。得到距离后可通过多维缩放方法（MDS）获得样本点在低维空间中的坐标。

Isomap

- 测地距离反映数据在流形上的真实距离差异



欧式距离 vs.
测地距离

最短路径近似
测地距离

降维嵌入空间

Isomap

- 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：

for $i = 1, 2, \ldots, m$ **do**
    确定 $\boldsymbol{x}_i$ 的 k 近邻;
    $\boldsymbol{x}_i$ 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;
end for
调用最短路径算法计算任意两样本点之间的距离 $\mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$;
将 $\mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ 作为 MDS 算法的输入;
return MDS 算法的输出

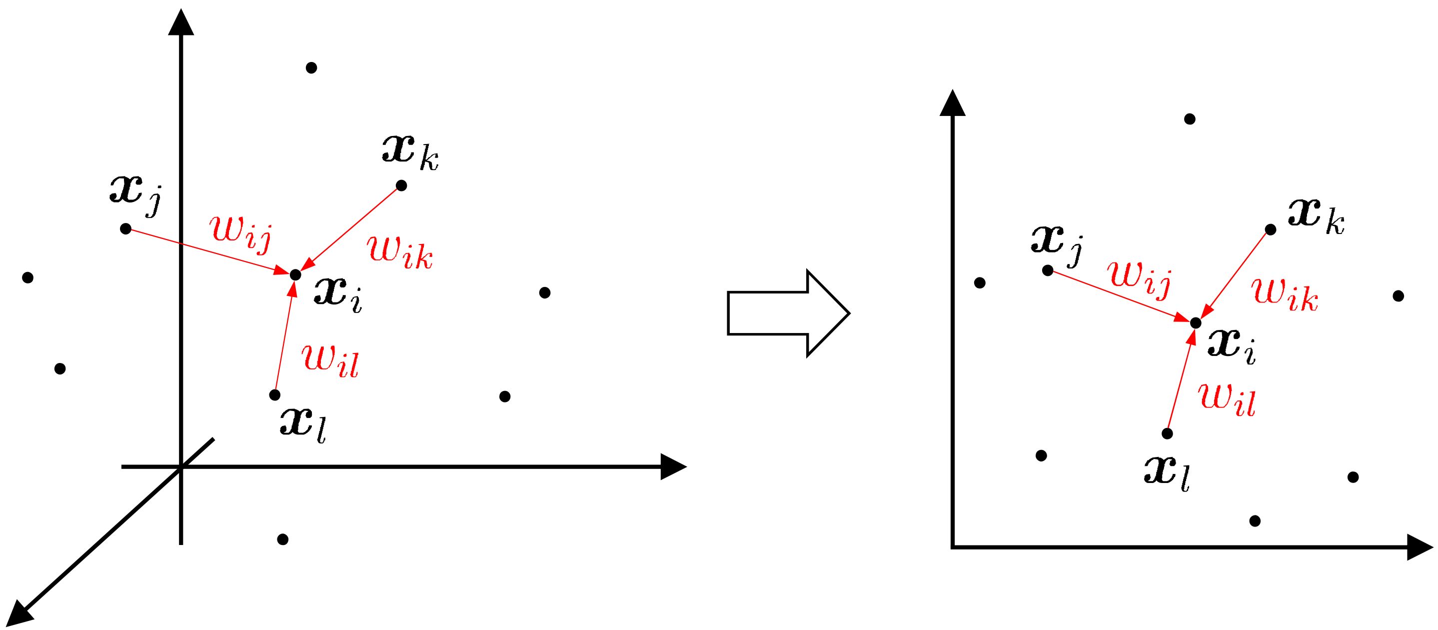
输出： 样本集 D 在低维空间的投影 $Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_m\}$.

图 10.8 Isomap 算法

LLE

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

- 局部线性嵌入试图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。



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

LLE

- LLE先为每个样本 $\boldsymbol{x}_i$找到其近邻下标集合 Q_i，然后计算出基于 Q_i 的中的样本点对 $\boldsymbol{x}_i$进行线性重构的系数$\boldsymbol{w}_i$：

$$\min_{\boldsymbol{w}_1,\boldsymbol{w}_2,\ldots,\boldsymbol{w}_m} \sum_{i=1}^{m} \left\| \boldsymbol{x}_i - \sum_{j \in Q_i} w_{ij} \boldsymbol{x}_j \right\|_2^2$$

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

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

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

LLE

- LLE在低维空间中保持$\boldsymbol{w}_i$不变，于是$\boldsymbol{x}_i$对应的低维空间坐标$\boldsymbol{z}_i$可通过下式求解：

$$\min_{\boldsymbol{w}_1,\boldsymbol{w}_2,\ldots,\boldsymbol{w}_m} \sum_{i=1}^{m} \left\| \boldsymbol{z}_i - \sum_{j\in Q_i} w_{ij}\boldsymbol{z}_j \right\|_2^2$$

- 令 $\mathbf{Z} = (\mathrm{z}_1, \mathrm{z}_2, \ldots, \mathrm{z}_m) \in \mathbb{R}^{d' \times m}, (\mathbf{W})_{ij} = w_{ij},$

$$\mathbf{M} = (\mathbf{I} - \mathbf{W})^{\mathrm{T}}(\mathbf{I} - \mathbf{W}),$$

- 则优化式可重写为右式，并通过特征值分解求解。

$$\min_{\mathbf{Z}} \operatorname{tr}(\mathbf{Z}\mathbf{M}\mathbf{Z}^{\mathrm{T}})$$
$$\text{s.t. } \mathbf{Z}\mathbf{Z}^{\mathrm{T}} = \mathbf{I}.$$

LLE

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

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：

```
for i = 1, 2, ..., m do
    确定 x_i 的 k 近邻;
    从式(10.27)求得 w_ij, j ∈ Q_i;
    对于 j ∉ Q_i, 令 w_ij = 0;
end for
从式(10.30)得到 M;
对 M 进行特征值分解;
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 $Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_m\}$.

图 10.10　LLE 算法

度量学习

度量学习

- 在机器学习中，对高维数据进行降维的主要目的是希望找到一个合适的低维空间，在此空间中进行学习能比原始空间性能更好。事实上，每个空间对应了在样本属性上定义的一个距离度量，而寻找合适的空间，实质上就是在寻找一个合适的距离度量。那么，为何不直接尝试"学习"出一个合适的距离度量呢？

度量学习

- 欲对距离度量进行学习，必须有一个便于学习的距离度量表达形式。对两个d维样本$\boldsymbol{x}_i$和$\boldsymbol{x}_j$，它们之间的平方欧氏距离可写为

$$\mathrm{dist}_{\mathrm{ed}}^2(\boldsymbol{x}_i,\boldsymbol{x}_j)=||\boldsymbol{x}_i-\boldsymbol{x}_j||_2^2=dist_{ij,1}^2+dist_{ij,2}^2+\cdots+dist_{ij,d}^2,$$

- 其中$dist_{ij,k}$表示$\boldsymbol{x}_i$与$\boldsymbol{x}_j$在第k维上的距离。若假定不同属性的重要性不同，则可引入属性权重$\boldsymbol{w}$，得到

$$\begin{aligned}\mathrm{dist}_{\mathrm{wed}}^2(\boldsymbol{x}_i,\boldsymbol{x}_j)&=||\boldsymbol{x}_i-\boldsymbol{x}_j||_2^2=w_1\cdot dist_{ij,1}^2+w_2\cdot dist_{ij,2}^2+\cdots+w_d\cdot dist_{ij,d}^2\\&=(\boldsymbol{x}_i-\boldsymbol{x}_j)^{\mathrm{T}}\mathbf{W}(\boldsymbol{x}_i-\boldsymbol{x}_j),\end{aligned}$$

- 其中$w_i\geq 0$，$\mathbf{W}=\mathrm{diag}(\boldsymbol{w})$是一个对角矩阵，$(\mathbf{W})_{ii}=w_i$，可通过学习确定。

度量学习

- $\mathbf{W}$的非对角元素均为零，这意味着坐标轴是正交的，即属性之间无关；但现实问题中往往不是这样，例如考虑西瓜的“重量”和“体积”这两个属性，它们显然是正相关的，其对应的坐标轴不再正交。为此将$\mathbf{W}$替换为一个普通的半正定对称矩阵$\mathbf{M}$，于是就得到了马氏距离(Mahalanobis distance)。

$$\mathrm{dist}_{\mathrm{mah}}^2(\boldsymbol{x}_i,\boldsymbol{x}_j)=(\boldsymbol{x}_i-\boldsymbol{x}_j)^{\mathrm{T}}\mathbf{W}(\boldsymbol{x}_i-\boldsymbol{x}_j)=||\boldsymbol{x}_i-\boldsymbol{x}_j||_{\mathbf{M}}^2,$$

其中$\mathbf{M}$亦称“度量矩阵”，而度量学习则是对$\mathbf{M}$进行学习。注意到为了保持距离非负且对称，$\mathbf{M}$必须是（半）正定对称矩阵。

度量学习

- 对 $\mathbf{M}$ 进行学习当然要设置一个目标。假定我们是希望提高近邻分类器的性能，则可将 $\mathbf{M}$ 直接嵌入到近邻分类器的评价指标中去，通过优化该性能指标相应地求得 $\mathbf{M}$。

- 不同的度量学习方法针对不同目标获得“好”的半正定对称距离度量矩阵 $\mathbf{M}$，若 $\mathbf{M}$ 是一个低秩矩阵，则通过对 $\mathbf{M}$ 进行特征值分解，总能找到一组正交基，其正交基数目为矩阵 $\mathbf{M}$ 的秩 $\mathrm{rank}(\mathbf{M})$，小于原属性数 d。于是，度量学习学得的结果可衍生出一个降维矩阵 $\mathbf{P} \in \mathbb{R}^{d \times \mathrm{rank}(\mathbf{M})}$，能用于降维之目的。

总结

- kNN算法：原理，分析，密采样
- 线性降维方法：主成分分析（PCA）
- 核化线性降维：KPCA
- 流形学习：流形，ISOMAP，LLE
- 度量学习：马氏距离