北京航空航天大學
COLLEGE OF SOFTWARE BEIHANG UNIVERSITY 软件学院

人工智能

第8讲：机器学习-无监督学习I

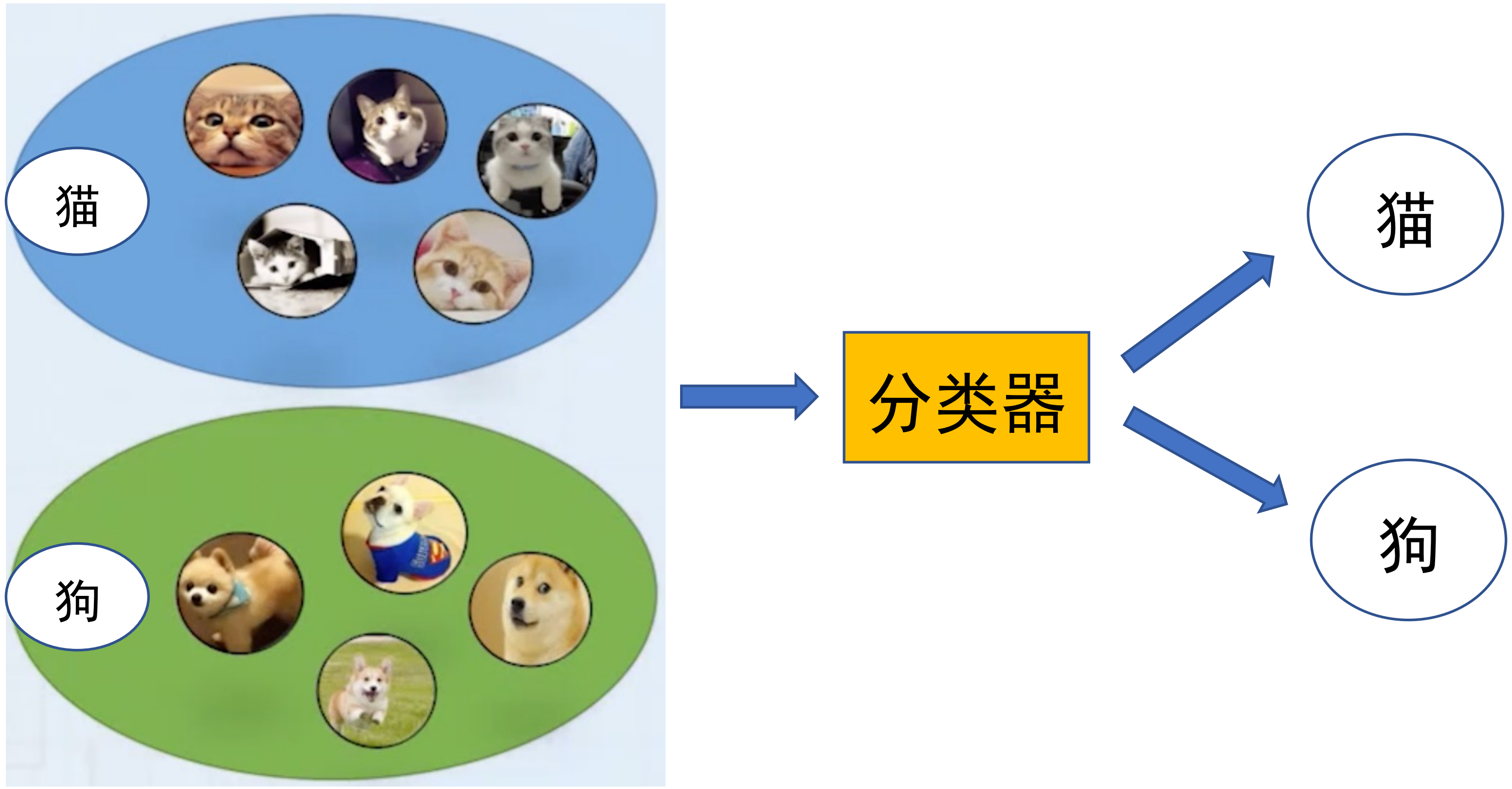
张晶

2023年春季

- 参考教材： 吴飞，《人工智能导论：模型与算法》，高等教育出版社
- 在线课程： https://www.icourse163.org/course/ZJU-1003377027?from=searchPage

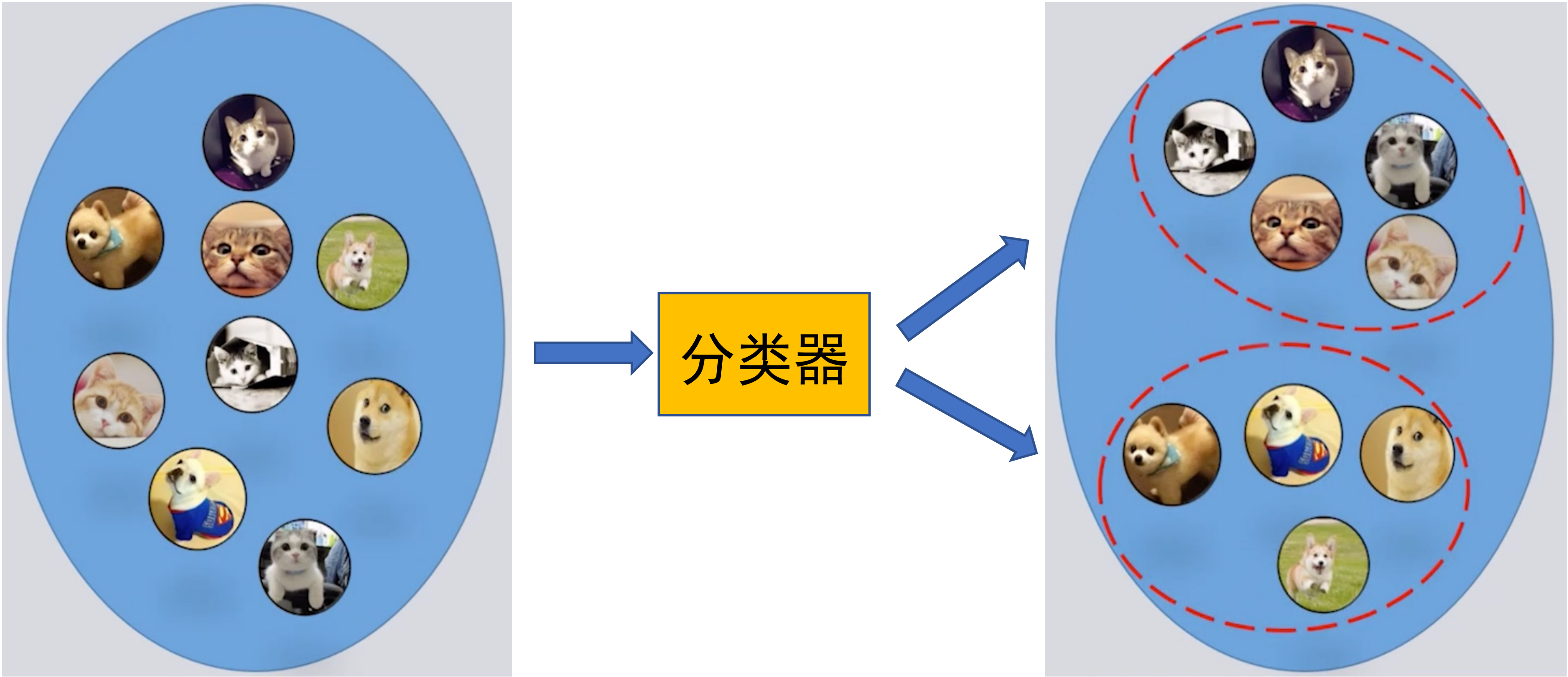
机器学习的分类（按数据标注情况分类）

- 有监督学习（supervised learning）

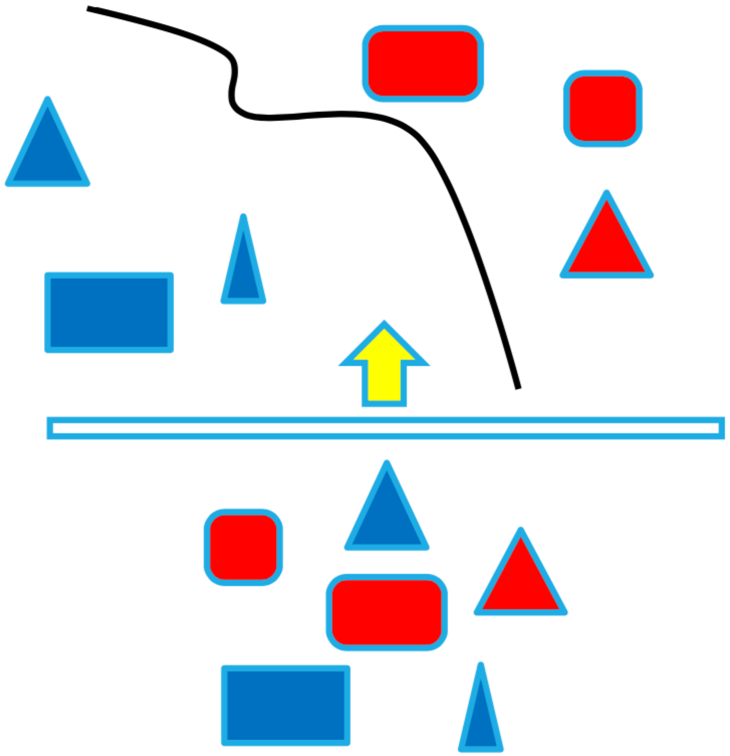


机器学习的分类（按数据标注情况分类）

- 无监督学习（unsupervised learning）



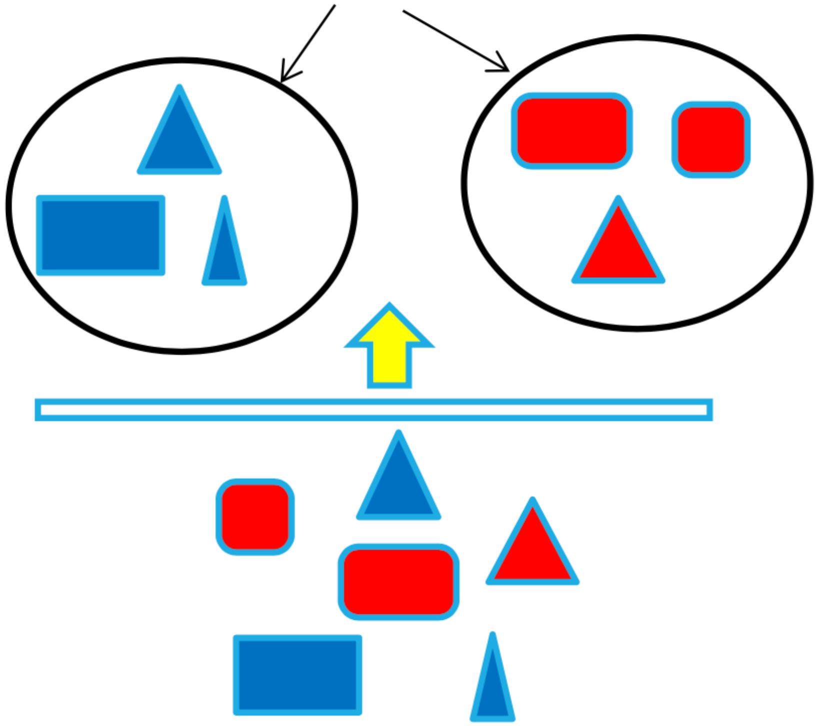
监督学习 versus 无监督学习



红色：汽车　蓝色：飞机

左：监督学习

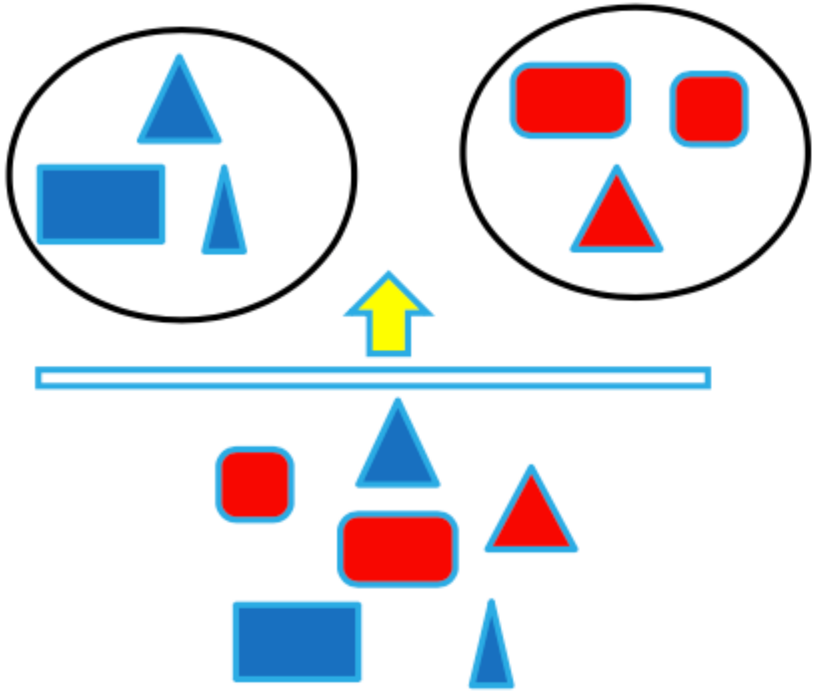
它们是相似的
数据的语义标签并不知道



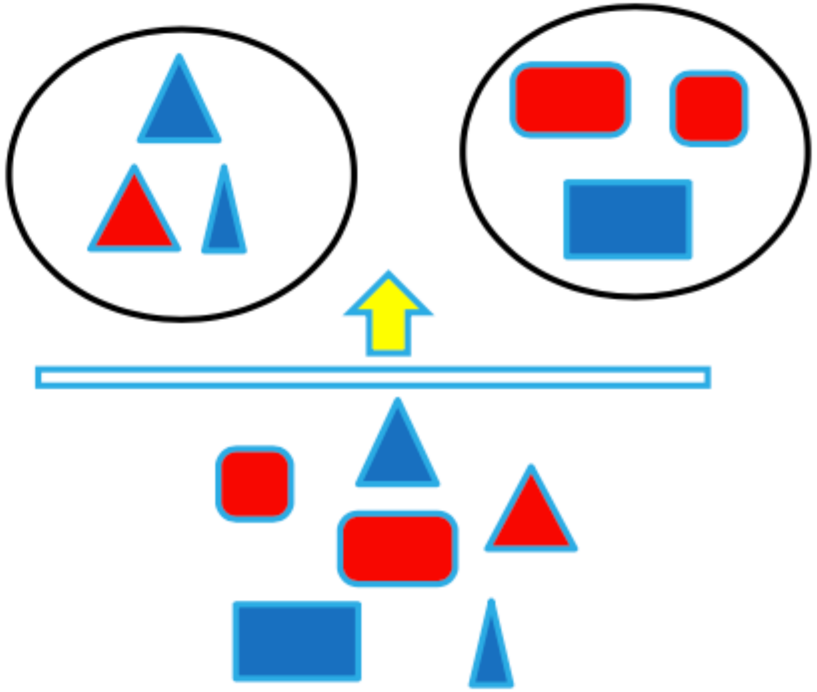
右：无监督学习

无监督学习：数据特征和相似度函数都很重要

相似度函数：颜色相似



相似度函数：形状相似



无监督学习

提纲

K均值聚类 (K-means 聚类)

- 物以类聚，人以群分(《战国策·齐策三》)
- 输入：$\boldsymbol{n}$个数据（无任何标注信息）
- 输出：$\boldsymbol{k}$个聚类结果
- 目的：将$\boldsymbol{n}$个数据聚类到$\boldsymbol{k}$个集合（也称为类簇）

K均值聚类 (K-means 聚类)

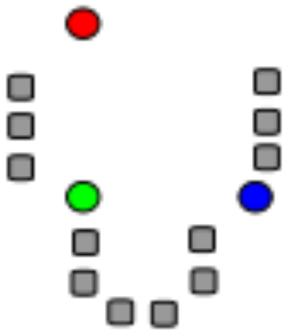
K均值聚类算法描述

- 若干定义：
  - n个m−维数据$\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n\}$，$\boldsymbol{x}_i \in R^m (1 \le i \le n)$
  - 两个m维数据之间的欧氏距离为

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sqrt{(\boldsymbol{x}_{i1} - \boldsymbol{x}_{j1})^2 + (\boldsymbol{x}_{i2} - \boldsymbol{x}_{j2})^2 + \cdots + (\boldsymbol{x}_{im} - \boldsymbol{x}_{jm})^2}$$

  - $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$值越小，表示$\boldsymbol{x}_i$和$\boldsymbol{x}_j$越相似；反之越不相似
  - 聚类集合数目k
- 问题：如何将n个数据依据其相似度大小将它们分别聚类到k个集合，使得每个数据仅属于一个聚类集合。

K均值聚类算法：初始化



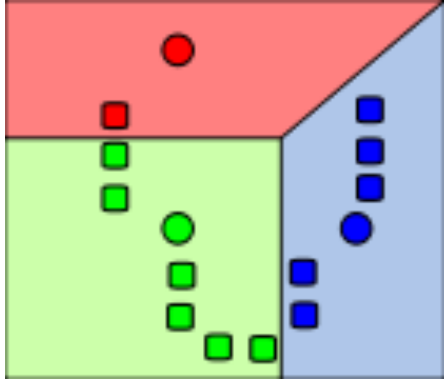
■ 第一步：初始化聚类质心

- 初始化k个聚类质心

$$C=\{\boldsymbol{c}_1,\boldsymbol{c}_2,\ldots,\boldsymbol{c}_k\},\boldsymbol{c}_j\in R^m(1\leq j\leq k)$$

- 聚类质心可以从数据$\{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n\}$，$\boldsymbol{x}_i\in R^m(1\leq i\leq n)$中采样得到。
- 每个聚类质心$\boldsymbol{c}_j$所在集合记为G_j。

K均值聚类算法：对数据进行聚类



■ 第二步：将每个待聚类数据放入唯一一个聚类集合中

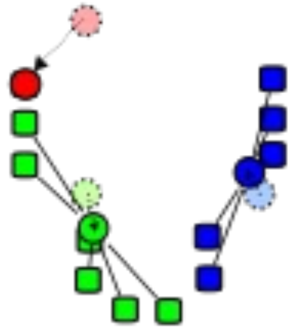
- 计算待聚类数据x_i和质心c_j之间的欧氏距离

$$d(x_i, c_j)\ (1 \le i \le n, 1 \le j \le k)$$

- 将每个x_i放入与之距离最近聚类质心所在聚类集合中，即

$$\underset{c_j \in C}{argmin}\, d(x_i, c_j)$$

K均值聚类算法：更新聚类质心



■ 第三步：根据聚类结果、更新聚类质心

- 根据每个聚类集合中所包含的数据，更新该聚类集合质心值，即：

$$c_j = \frac{1}{|G_j|}\sum_{x_i \in G_j} x_i$$

K均值聚类算法：继续迭代

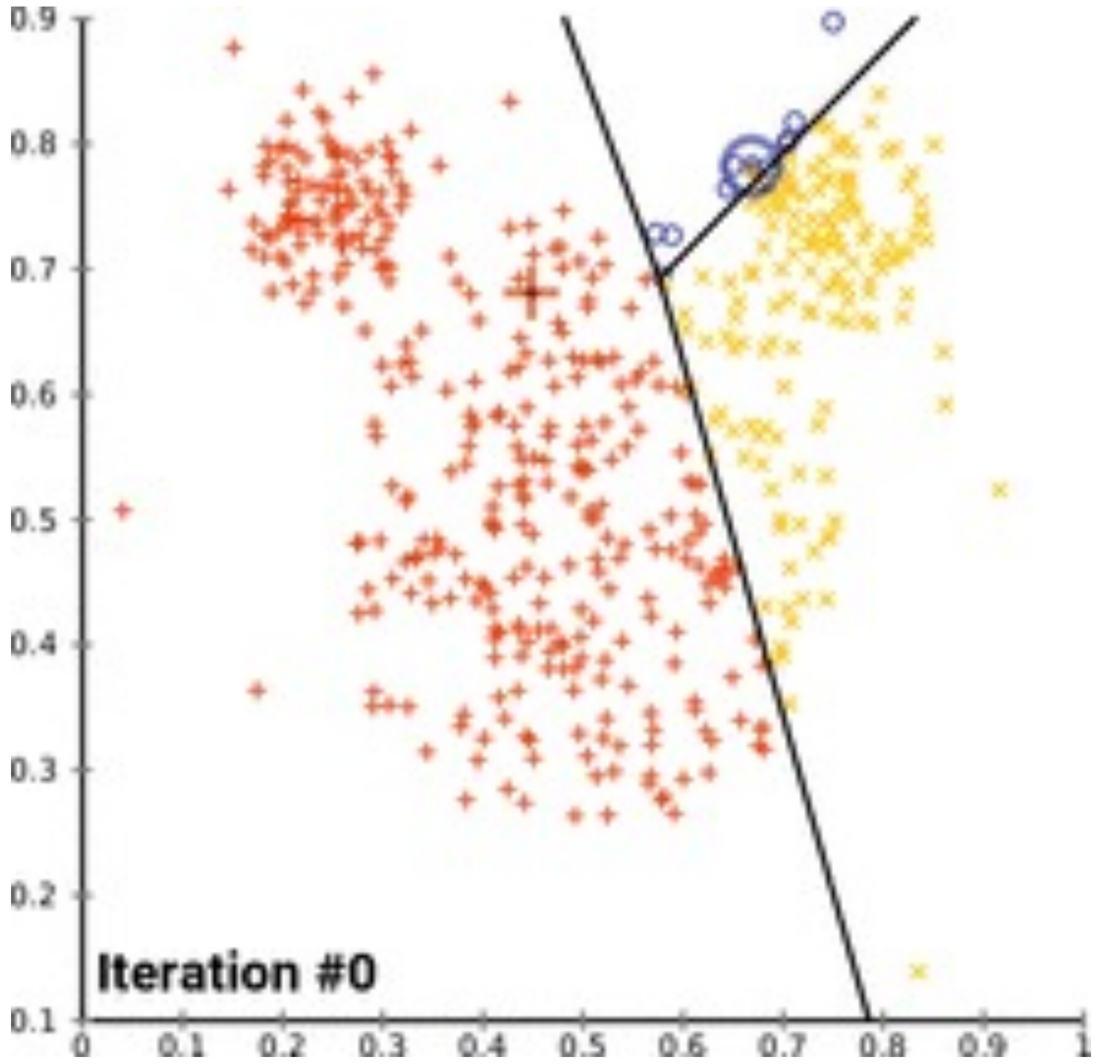
■ 第四步：算法循环迭代，直到满足条件

- 在新聚类质心基础上，根据欧氏距离大小，将每个待聚类数据放入唯一一个聚类集合中
- 根据新的聚类结果、更新聚类质心

聚类迭代满足如下任意一个条件，则聚类停止：

- 已经达到了迭代次数上限
- 前后两次迭代中，聚类质心基本保持不变

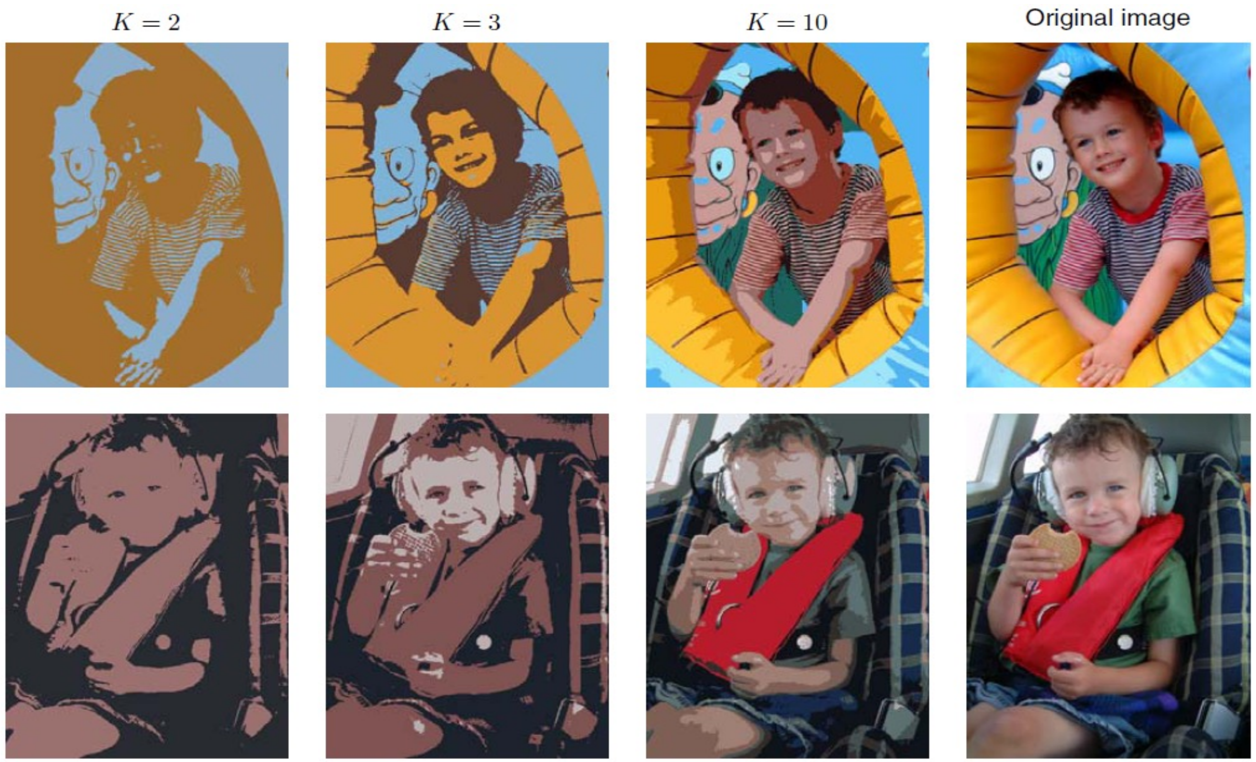
K均值聚类算：收敛过程



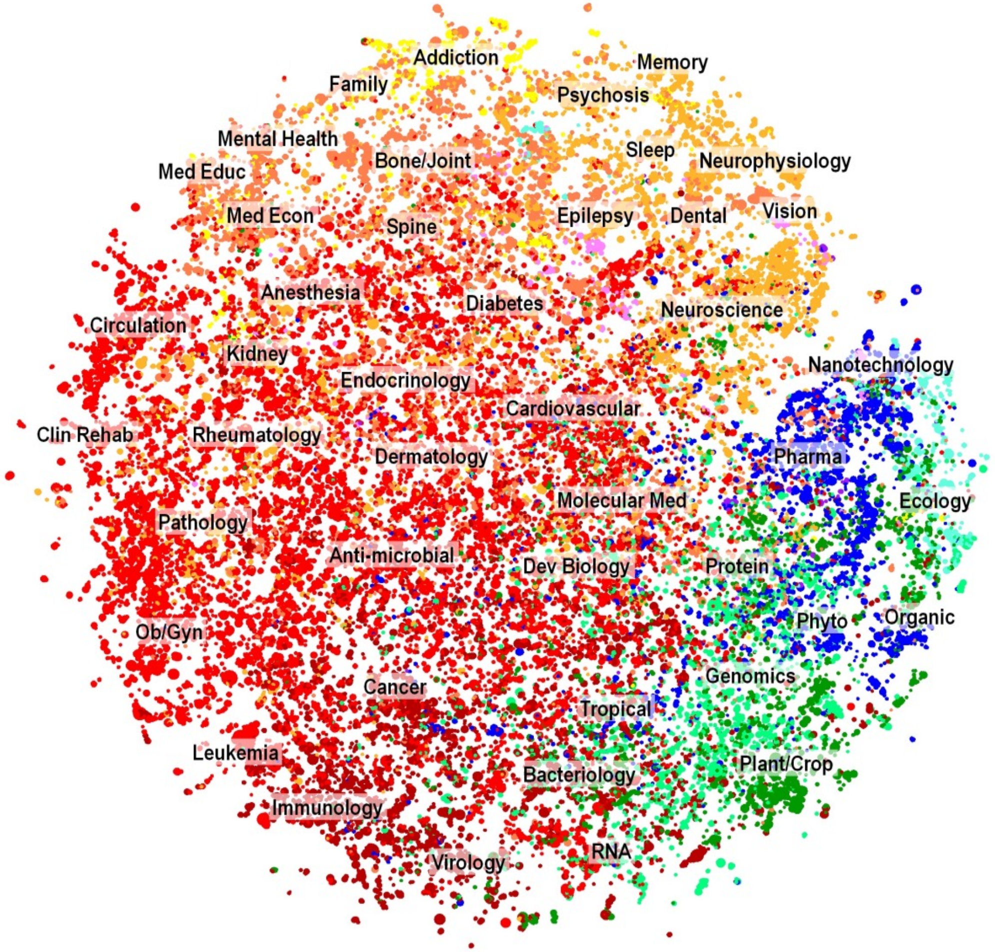
K均值聚类算法的不足

- 需要事先确定聚类数目，很多时候我们并不知道数据应被聚类的数目
- 需要初始化聚类质心，初始化聚类中心对聚类结果有较大的影响
- 算法是迭代执行，时间开销非常大
- 欧氏距离假设数据每个维度之间的重要性是一样的

K均值聚类算法的应用



图像分割



文本分类：将200多万篇论文聚类到29,000个类别，包括化学、工程、生物、传染疾病、生物信息、脑科学、社会科学、计算机科学等及给出了每个类别中的代表单词

提纲

一、K均值聚类

二、主成分分析

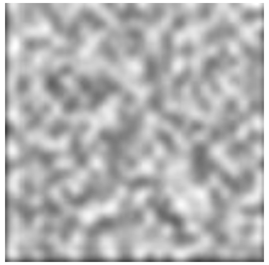
三、特征人脸方法

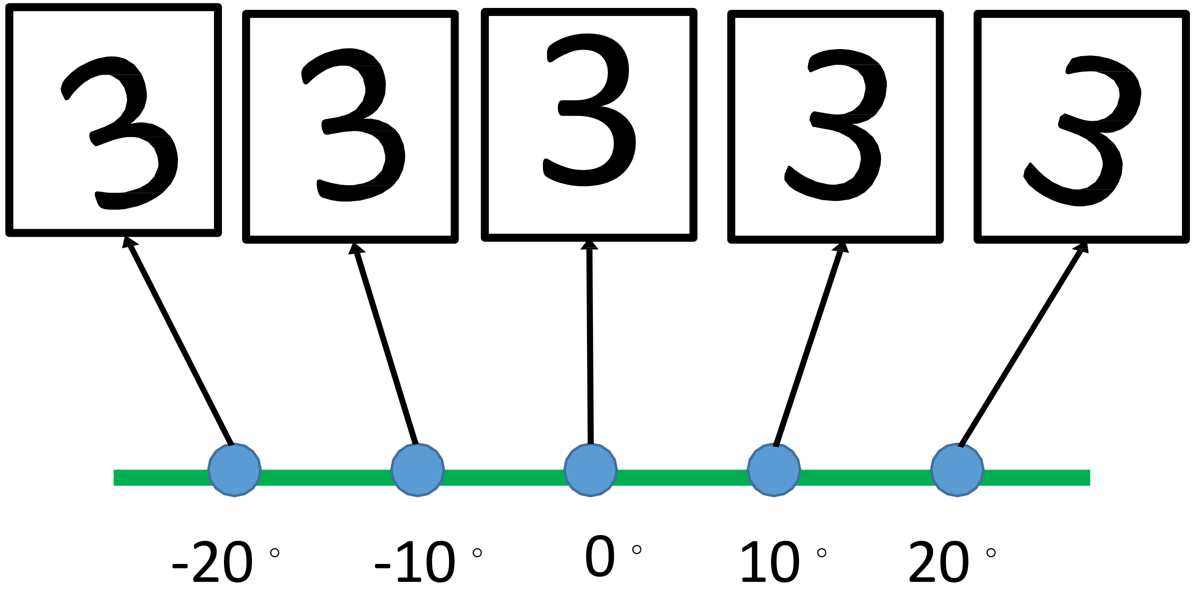
四、潜在语义分析

五、期望最大化算法

主成分分析: Principle Component Analysis （PCA）

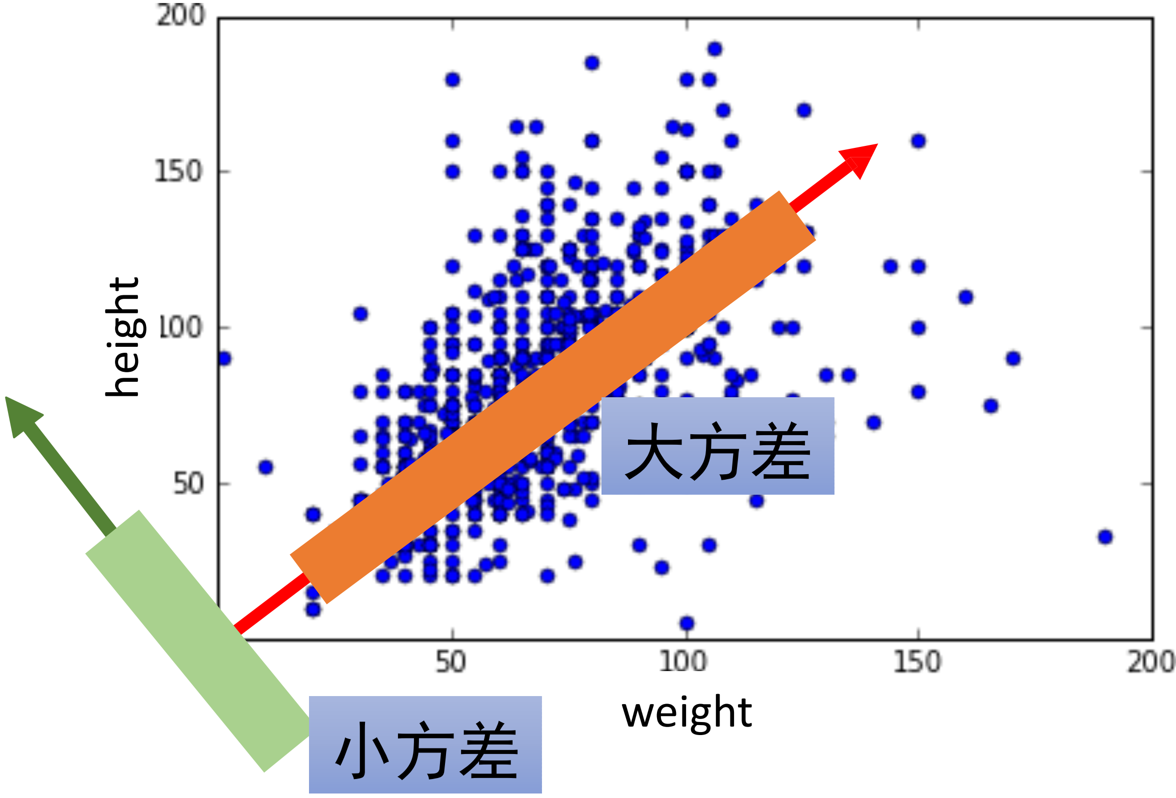
- 主成分分析是一种特征降维方法。人类在认知过程中会主动“化繁为简”
- 奥卡姆剃刀定律（Occam’s Razor）：“如无必要，勿增实体”，即“简单有效原理”





主成分分析: 算法动机

- 在数理统计中，方差被经常用来度量数据和其数学期望（即均值）之间偏离程度，这个偏离程度反映了数据分布结构。
- 在许多实际问题中，研究数据和其均值之间的偏离程度有着很重要的意义。
- 在降维之中，需要尽可能将数据向方差最大方向进行投影，使得数据所蕴含信息没有丢失，彰显个性。
- 如右图所示，向红色方向比绿色方向投影结果在降维这个意义上好。



主成分分析: 算法动机

- 主成分分析思想是将$\boldsymbol{n}$维特征数据映射到$\boldsymbol{l}$维空间（$\boldsymbol{n} \gg \boldsymbol{l}$），去除原始数据之间的冗余性（通过去除相关性手段达到这一目的）。
- 将原始数据向这些数据方差最大的方向进行投影。一旦发现了方差最大的投影方向，则继续寻找保持方差第二的方向且进行投影。
- 将每个数据从$\boldsymbol{n}$维高维空间映射到$\boldsymbol{l}$维低维空间，每个数据所得到最好的$\boldsymbol{k}$维特征就是使得每一维上样本方差都尽可能大。

主成分分析: 若干概念-方差与协方差

数据样本的方差 variance

- 假设有n个数据，记为$X = \{\boldsymbol{x}_i\}$ $(i = 1, \ldots, n)$
  - 方差等于各个数据与样本均值之差的平方和之平均数
  - 方差描述了样本数据的波动程度

$$Var(X) = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{x}_i - E(X))^2$$

  - 其中$E(X)$是样本均值， $E(X) = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i$

主成分分析: 若干概念-方差与协方差

数据样本的协方差 covariance

- 假设有n个两维变量数据，记为$(X, Y) = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}\ (i = 1, \ldots, n)$
  - 衡量两个变量之间的相关度

$$cov(X, Y) = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{x}_i - E(X))(\boldsymbol{y}_i - E(Y))$$

  - 其中$E(X)$和$E(Y)$分别是X和Y的样本均值，分别定义如下

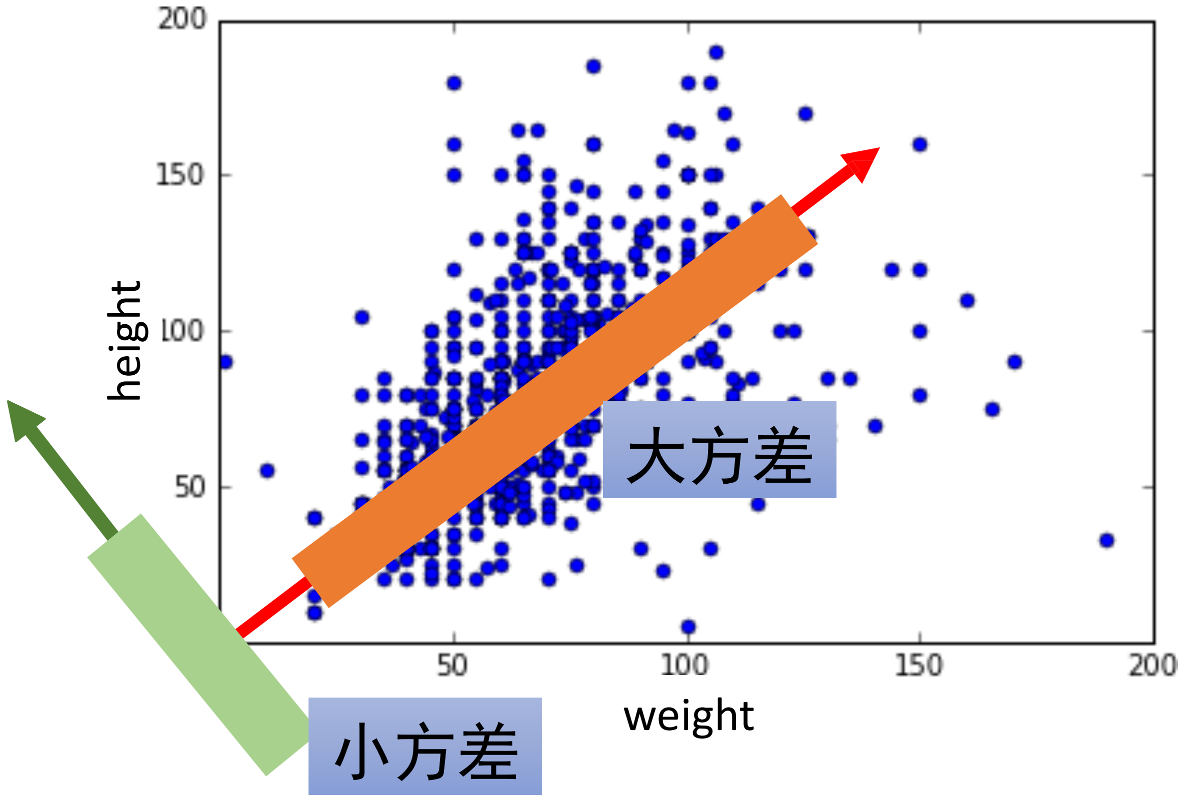
$$E(X) = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i, \qquad E(Y) = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{y}_i$$

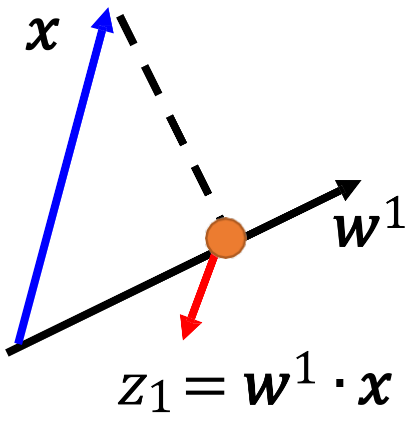
主成分分析：目标函数

$$z = W\boldsymbol{x}$$

降维到1维：

$$z_1 = (\boldsymbol{w}^1)^T \cdot \boldsymbol{x}$$





- 将所有数据$\boldsymbol{x}$映射到$\boldsymbol{w}^1$, 得到一组新的特征z_1
- 希望映射后的数据z_1的方差尽可能大

$$Var(z_1) = \frac{1}{N}\sum_{z_1}(z_1 - \bar{z_1})^2 \qquad \|\boldsymbol{w}^1\|_2 = 1$$

主成分分析：目标函数

$$z = W\boldsymbol{x}$$

降维到多维：

$$z_1 = (\boldsymbol{w}^1)^T \cdot \boldsymbol{x}$$

$$z_2 = (\boldsymbol{w}^2)^T \cdot \boldsymbol{x}$$

$$W = \begin{bmatrix} (w^1)^T \\ (w^2)^T \\ \vdots \end{bmatrix}$$

正交矩阵

将所有数据$\boldsymbol{x}$ 映射到 $\boldsymbol{w}^1$, 得到一组新的特征 z_1

希望映射后的数据z_1的方差尽可能大

$$Var(z_1) = \frac{1}{N}\sum_{z_1}(z_1 - \bar{z_1})^2 \qquad \|\boldsymbol{w}^1\|_2 = 1$$

希望映射后的数据z_2的方差尽可能大

$$Var(z_2) = \frac{1}{N}\sum_{z_2}(z_2 - \bar{z_2})^2 \qquad \|\boldsymbol{w}^2\|_2 = 1$$

$$(\boldsymbol{w}^1)^T\boldsymbol{w}^2 = 0$$

主成分分析：推导

- 根据$z_1 = (\boldsymbol{w}^1)^T\boldsymbol{x}$

  以及$\bar{z}_1 = \frac{1}{N}\sum z_1 = \frac{1}{N}\sum(\boldsymbol{w}^1)^T\boldsymbol{x} = (\boldsymbol{w}^1)^T\frac{1}{N}\sum\boldsymbol{x} = (\boldsymbol{w}^1)^T\overline{\boldsymbol{x}}$

- 得到$Var(z_1) = \frac{1}{N}\sum(z_1 - \bar{z}_1)^2 = \frac{1}{N}\sum((\boldsymbol{w}^1)^T\boldsymbol{x} - (\boldsymbol{w}^1)^T\overline{\boldsymbol{x}})^2$

  $= \frac{1}{N}\sum((\boldsymbol{w}^1)^T(\boldsymbol{x} - \overline{\boldsymbol{x}}))^2 = (\boldsymbol{w}^1)^T\frac{1}{N}\sum(\boldsymbol{x} - \overline{\boldsymbol{x}})(\boldsymbol{x} - \overline{\boldsymbol{x}})^T\boldsymbol{w}^1$

  $= (\boldsymbol{w}^1)^T Cov(\boldsymbol{x})\boldsymbol{w}^1 = (\boldsymbol{w}^1)^T S\boldsymbol{w}^1$

- 因此，主成分分析的目标函数为

$$\max(\boldsymbol{w}^1)^T S\boldsymbol{w}^1$$

$$s.t.\ \|\boldsymbol{w}^1\|_2 = (\boldsymbol{w}^1)^T\boldsymbol{w}^1 = 1$$

主成分分析：推导

$$\max (\boldsymbol{w}^1)^T S \boldsymbol{w}^1 \quad s.t. \|\boldsymbol{w}^1\|_2 = (\boldsymbol{w}^1)^T \boldsymbol{w}^1 = 1$$

- 根据拉格朗日乘子法[Bishop, Appendix E]

$$g(\boldsymbol{w}^1) = (\boldsymbol{w}^1)^T S \boldsymbol{w}^1 - \alpha((\boldsymbol{w}^1)^T \boldsymbol{w}^1 - 1)$$

$$\left.\begin{aligned} &\partial g(\boldsymbol{w}^1)/\partial \boldsymbol{w}_1^1 = 0 \\ &\partial g(\boldsymbol{w}^1)/\partial \boldsymbol{w}_2^1 = 0 \\ &\qquad \vdots \end{aligned}\right\}$$

$$S\boldsymbol{w}^1 - \alpha \boldsymbol{w}^1 = 0$$

$$S\boldsymbol{w}^1 = \alpha \boldsymbol{w}^1$$

w^1 :特征向量

$$(\boldsymbol{w}^1)^T S \boldsymbol{w}^1 = \alpha (\boldsymbol{w}^1)^T \boldsymbol{w}^1$$

$$= \alpha$$

选择最大的特征值

$\boldsymbol{w}^1$ 是协方差矩阵S的一个特征向量

对应于最大特征值λ_1

主成分分析：推导

$$\max(\boldsymbol{w}^2)^T S\boldsymbol{w}^2 \quad s.t.(\boldsymbol{w}^2)^T\boldsymbol{w}^2 = 1 \qquad (\boldsymbol{w}^2)^T\boldsymbol{w}^1 = 0$$

$$g(\boldsymbol{w}^2) = (\boldsymbol{w}^2)^T S\boldsymbol{w}^2 - \alpha((\boldsymbol{w}^2)^T\boldsymbol{w}^2 - 1) - \beta((\boldsymbol{w}^2)^T\boldsymbol{w}^1 - 0)$$

$$\left.\begin{array}{l}\partial g(\boldsymbol{w}^2)/\partial \boldsymbol{w}_1^2 = 0 \\ \partial g(\boldsymbol{w}^2)/\partial \boldsymbol{w}_2^2 = 0 \\ \vdots\end{array}\right\} \quad S\boldsymbol{w}^2 - \alpha\boldsymbol{w}^2 - \beta\boldsymbol{w}^1 = 0$$

$$0 - \alpha \; 0 - \beta \; 1 = 0$$

$$= \left((\boldsymbol{w}^1)^T S\boldsymbol{w}^2\right)^T = (\boldsymbol{w}^2)^T S^T\boldsymbol{w}^1$$

$$= (\boldsymbol{w}^2)^T S\boldsymbol{w}^1 = \lambda_1(\boldsymbol{w}^2)^T\boldsymbol{w}^1 = 0$$

$$S\boldsymbol{w}^1 = \lambda_1\boldsymbol{w}^1$$

$$\beta = 0: \quad S\boldsymbol{w}^2 - \alpha\boldsymbol{w}^2 = 0 \qquad S\boldsymbol{w}^2 = \alpha\boldsymbol{w}^2$$

$\boldsymbol{w}^2$ 是协方差矩阵S的一个特征向量

对应于第二大特征值λ_2

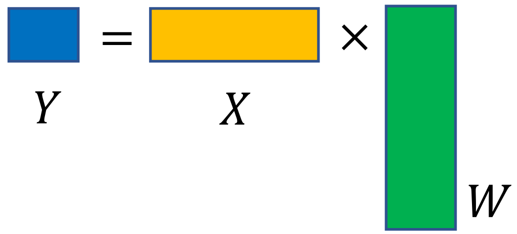
主成分分析: 算法描述

- 假设有n个d维样本数据所构成的集合$X^{org}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n\}$，其中$\boldsymbol{x}_i(1\le i\le n)\in R^d$
- 主成分分析的主要步骤：
  - 数据预处理：$X=\frac{X^{org}-\mu}{\sigma}$
  - 计算协方差矩阵：$\Sigma=\frac{1}{n-1}X^TX$
  - 求得协方差矩阵Σ的特征向量和特征根
  - 取前l个最大特征根所对应的特征向量组成映射矩阵W
  - 给定一个样本$\boldsymbol{x}_i$，可将$\boldsymbol{x}_i$从d维空间如下映射到l维空间：$(\boldsymbol{x}_i)_{1\times d}(W)_{d\times l}$
  - 将所有降维后数据用Y表示，有$\underbrace{Y}_{\text{降维结果}}=\underbrace{X}_{\text{原始数据}}\underbrace{W}_{\text{映射矩阵}}$

$Y\in R^{n\times l}$

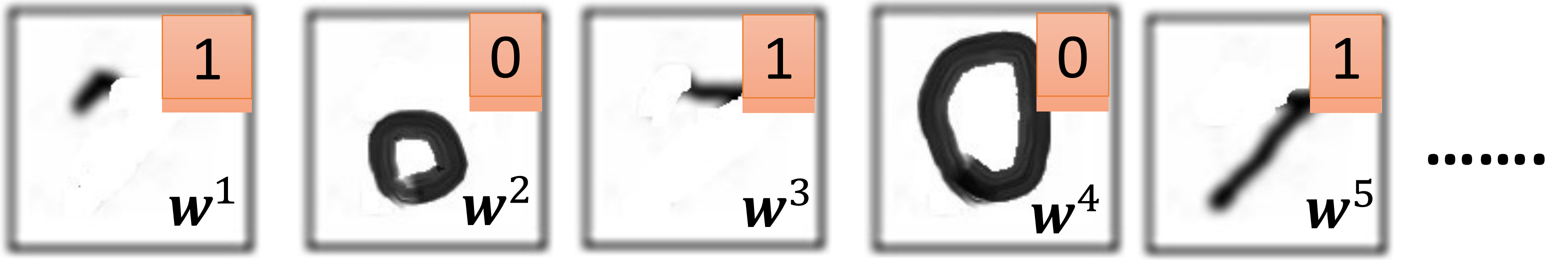
$X\in R^{n\times d}$

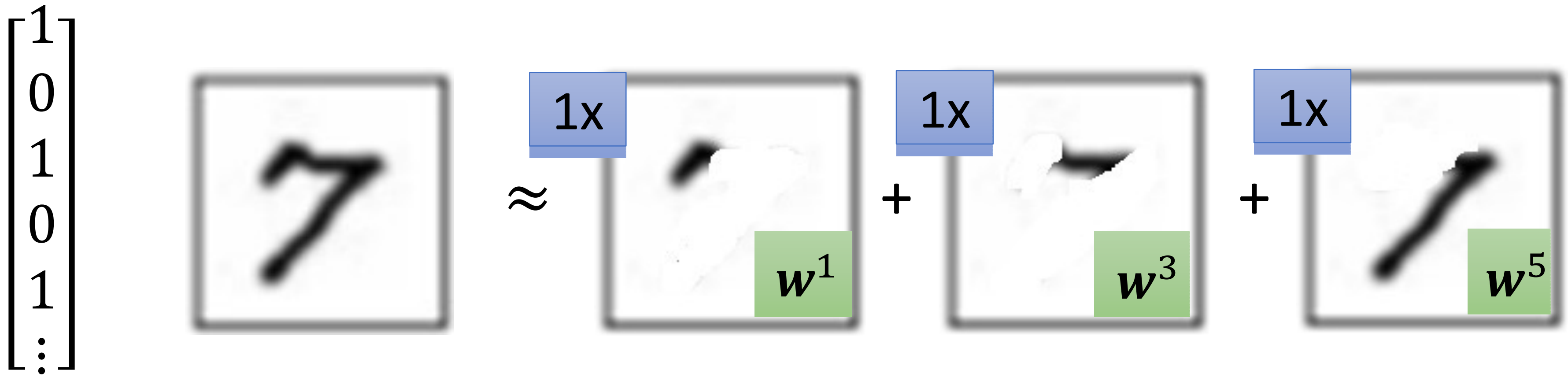
$W\in R^{d\times l}$



主成分分析: 另一个角度

主成分:





$$\boldsymbol{x} \approx y_1\boldsymbol{w}^1 + y_2\boldsymbol{w}^2 + \cdots + y_K\boldsymbol{w}^K + \bar{\boldsymbol{x}}$$

手写数字图像的像素值表达

主成分

$$\begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_K \end{bmatrix}$$

手写数字图像新特征表达

主成分分析: 另一个角度

$$\boldsymbol{x}-\overline{\boldsymbol{x}} \approx y_1\boldsymbol{w}^1+y_2\boldsymbol{w}^2+\cdots+y_K\boldsymbol{w}^K=\widehat{\boldsymbol{x}}$$

重构误差：$\|(\boldsymbol{x}-\overline{\boldsymbol{x}})-\widehat{\boldsymbol{x}}\|_2$

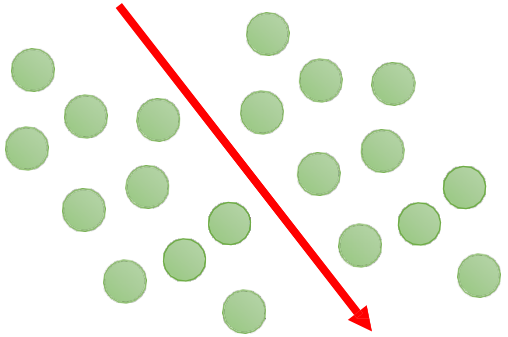
优化问题：通过最小化L得到$\boldsymbol{w}$，

$$L=\min_{\{\boldsymbol{w}^1,\ldots,\boldsymbol{w}^K\}}\sum\left\|(\boldsymbol{x}-\overline{\boldsymbol{x}})-\left(\sum_{k=1}^{K}y_k\boldsymbol{w}^k\right)\right\|_2$$

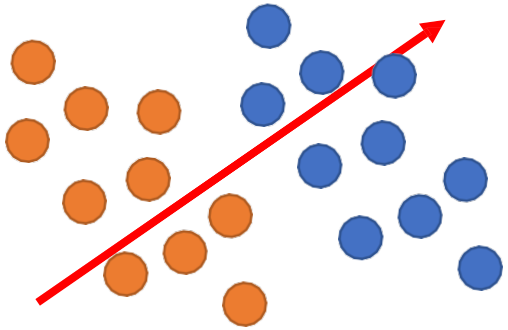
PCA的缺点

- 无监督

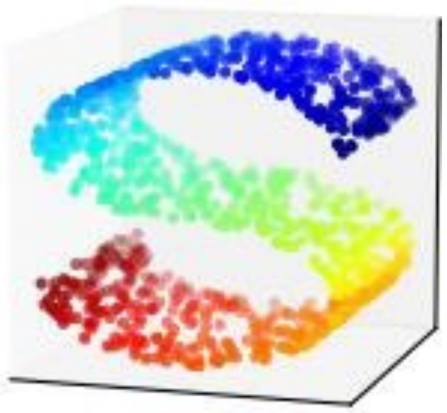
PCA

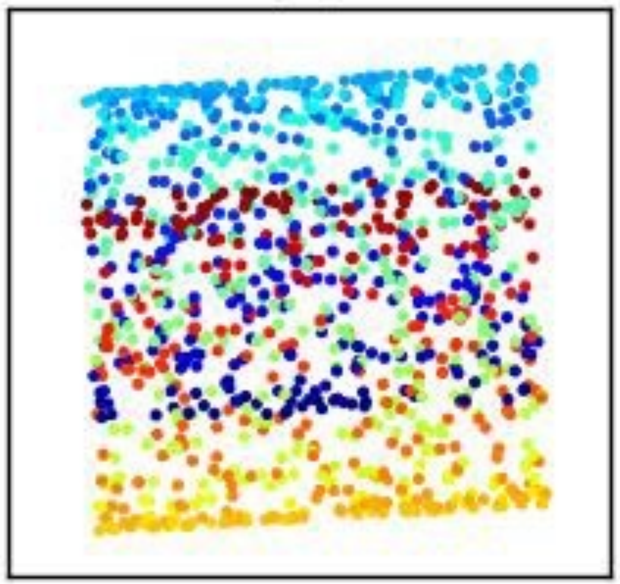


LDA



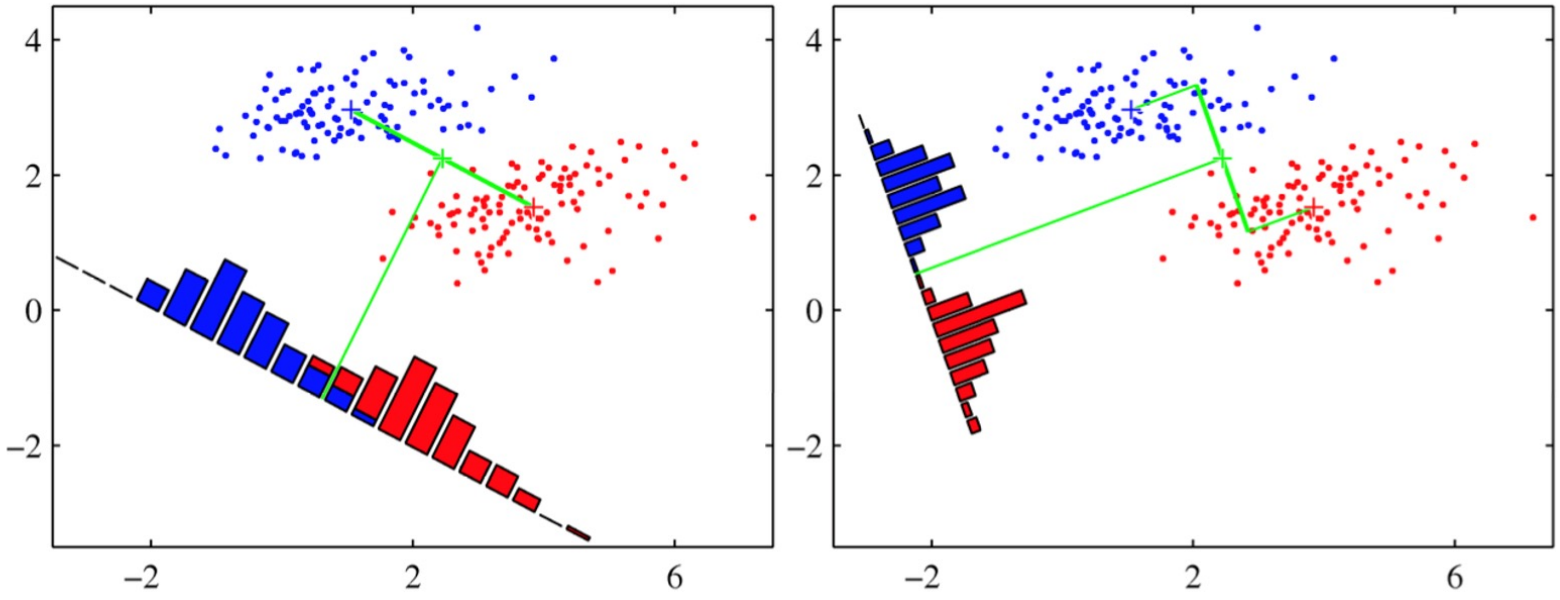
- 线性





http://www.astroml.org/book_figures/chapter7/fig_S_manifold_PCA.html

回顾：线性判别分析



- Fisher准则
  - 在要求类间差异尽量大的同时，要求类内数据在投影以后尽量集中。

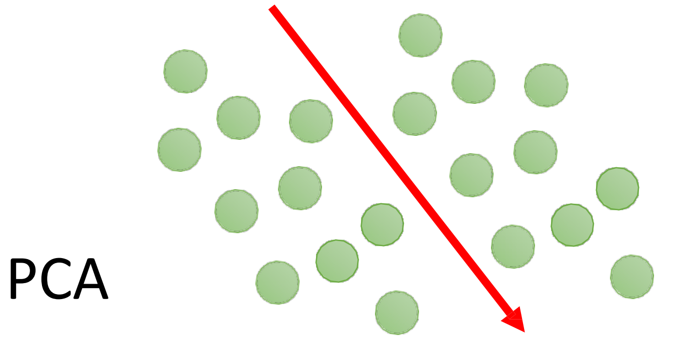
$$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^T S_b \boldsymbol{w}}{\boldsymbol{w}^T S_w \boldsymbol{w}}$$

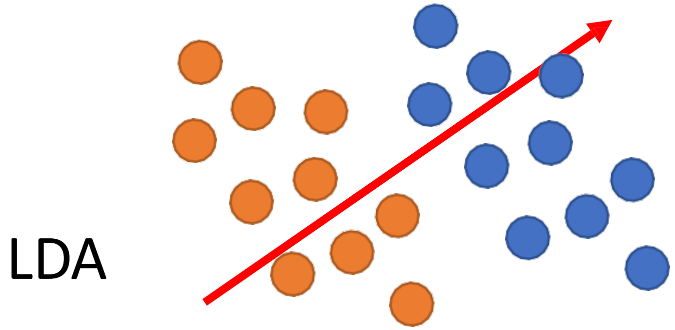
主成分分析：与线性判别分析法的异同

	Fisher线性判别分析	主成分分析
是否需要样本标签	监督学习	无监督学习
降维方法	优化寻找特征向量$\boldsymbol{w}$	优化寻找特征向量$\boldsymbol{w}$
目标	类内方差小、类间距离大	寻找投影后数据之间方差最大的投影方向
维度	LDA降维后所得到维度是与数据样本的类别个数K有关	PCA对高维数据降维后的维数通常远小于原始数据特征维度即可

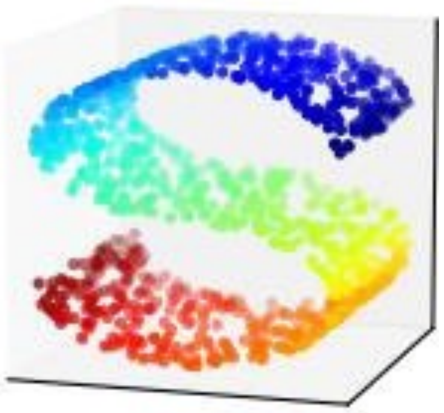
主成分分析的缺点

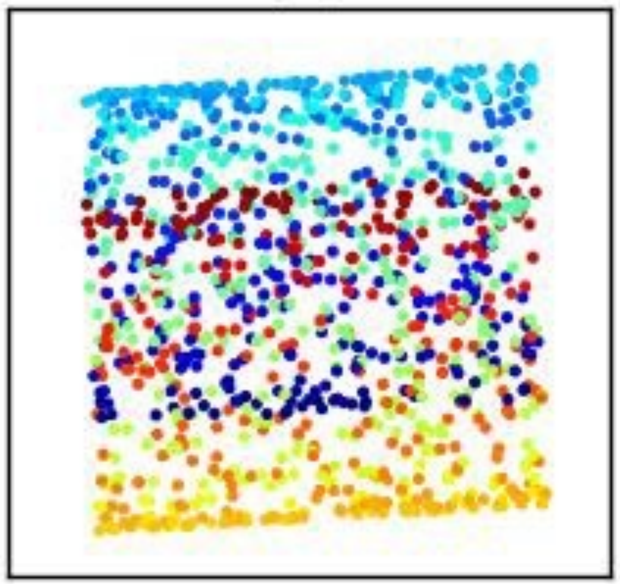
- 无监督





- 线性





http://www.astroml.org/book_figures/chapter7/fig_S_manifold_PCA.html

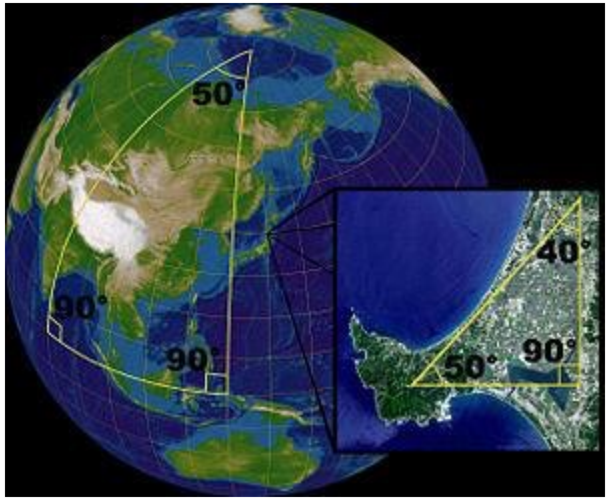
其他常用降维方法

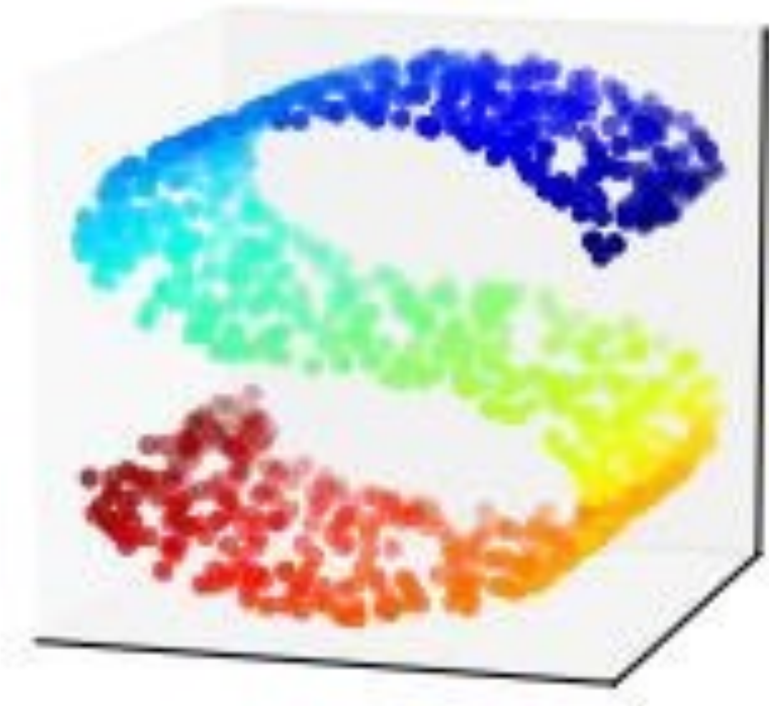
- 非负矩阵分解（non-negative matrix factorization, NMF）
- 多维尺度法(Metric multidimensional scaling, MDS)
- 流形学习：局部线性嵌入（Locally Linear Embedding，LLE）

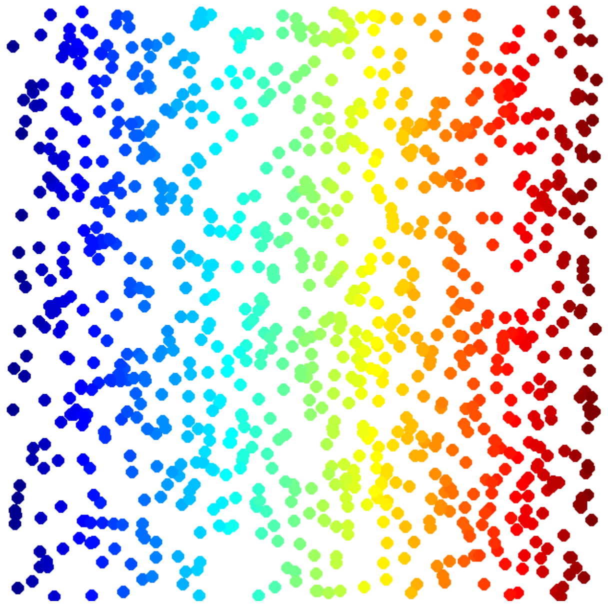
局部线性嵌入（Locally Linear Embedding，LLE）

- PCA、NMF和MDS都属于线性降维方法，LLE是一种非线性降维方法。
- LLE的基本假设是：一个流形的局部可以近似于一个欧式空间，每个样本均可以利用其邻居进行线性重构。换句话说，如果假设数据是局部线性的（即使数据的原始高维空间是非线性流形嵌入），于是可以用和邻居数据的线性关系进行局部重构。
- 局部线性嵌入保留了每个数据的局部性质，利用局部线性来逼近全局非线性，通过相互重叠的局部邻域提供全局结构信息，最终保持数据整体几何信息。

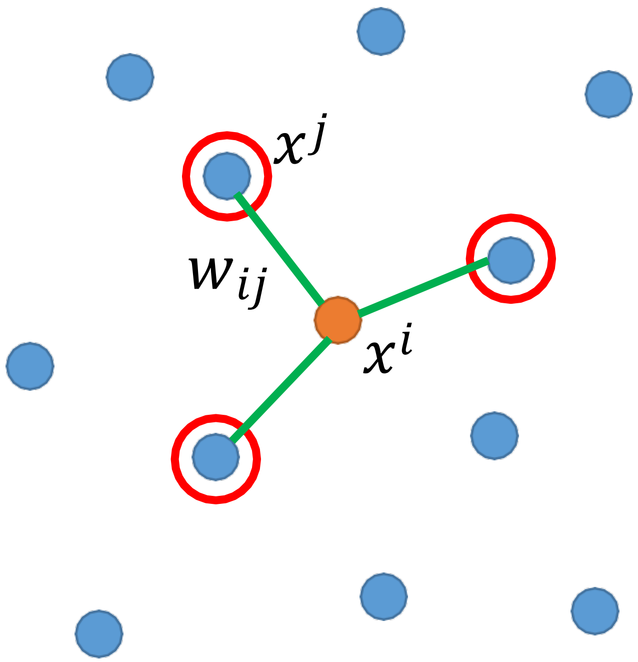
局部线性嵌入（Locally Linear Embedding， LLE）







局部线性嵌入（Locally Linear Embedding，LLE）



w_{ij} 表示x^i 和 x^j之 间 的 关 系

找到一组w_{ij} 以最小化

$$\sum_i \left\| x^i - \sum_j w_{ij} x^j \right\|_2$$

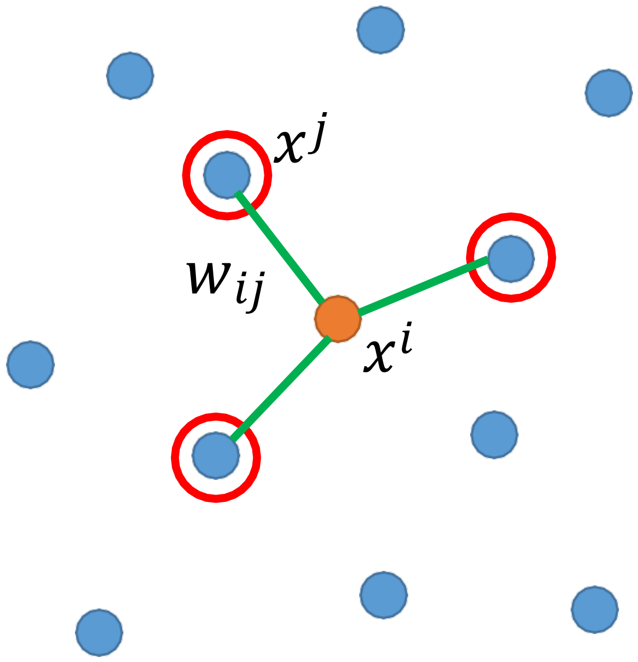
然后基于 w_{ij} 找出降维后的结果z^i 和 z^j

Lawrence K. Saul, Sam T. Roweis, “Think Globally, Fit Locally:Unsupervised Learning of Low Dimensional Manifolds”, JMLR, 2013

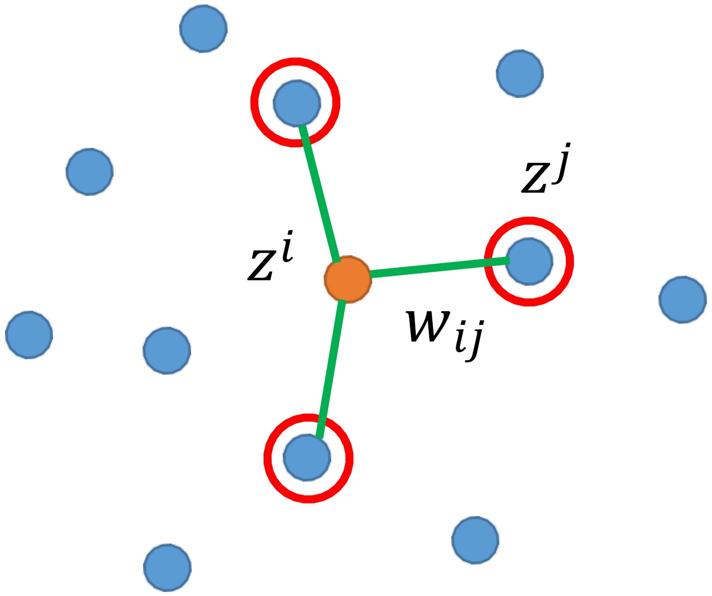
局部线性嵌入（Locally Linear Embedding，LLE）

保持 w_{ij} 不变

找出一组z^i 以最小化： $\sum_i \left\| z^i - \sum_j w_{ij} z^j \right\|_2$

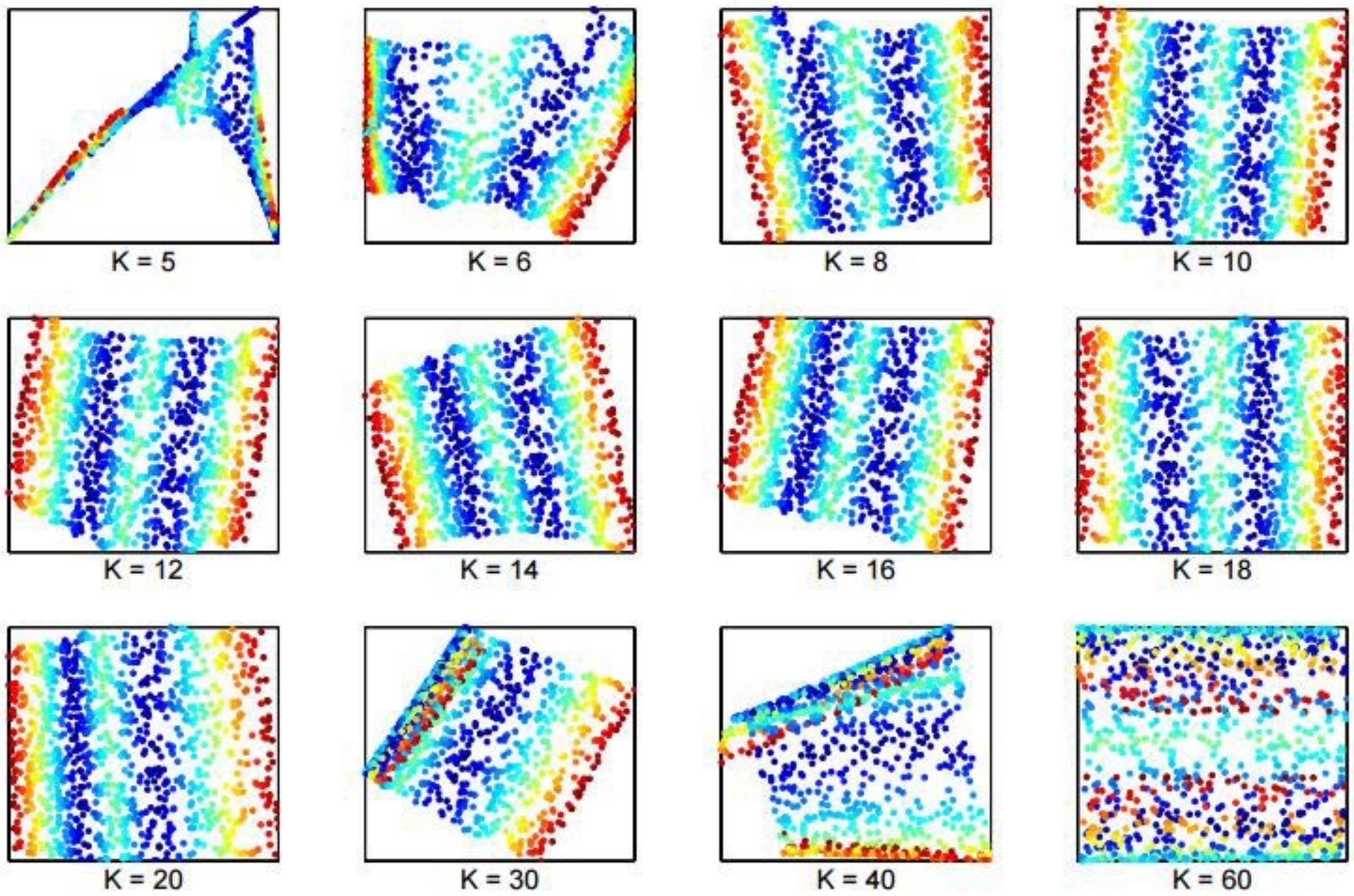


原始空间



降维后的低维空间

局部线性嵌入（Locally Linear Embedding，LLE）



Lawrence K. Saul, Sam T. Roweis, “Think Globally, Fit Locally:Unsupervised Learning of Low Dimensional Manifolds”, JMLR, 2013
https://cs.nyu.edu/~roweis/lle/

更多非线性降维方法

- 局部线性嵌入 Locally Linear Embedding (LLE): [Alpaydin, Chapter 6.11]
- 拉普拉斯特征图 Laplacian Eigenmaps: [Alpaydin, Chapter 6.12]
- t-SNE
  - Laurens van der Maaten, Geoffrey Hinton,“Visualizing Data using t-SNE”, JMLR, 2008
  - Excellent tutorial: https://github.com/oreillymedia/t-SNE-tutorial

提纲

一、K均值聚类

二、主成分分析

三、特征人脸方法

四、潜在语义分析

五、期望最大化算法

特征人脸方法: 动机

- 特征人脸方法是一种应用主成份分析来实现人脸图像降维的方法，其本质是用一种称为“特征人脸(eigenface)”的特征向量按照线性组合形式来表达每一张原始人脸图像，进而实现人脸识别。
- 由此可见，这一方法的关键之处在于如何得到特征人脸。

用（特征）人脸表示人脸，而非用像素点表示人脸

特征人脸方法: 算法描述

$$\begin{bmatrix}45 & \cdots & 68\\ \cdots & \cdots & \cdots\\ 36 & \cdots & 86\end{bmatrix}_{32\times32=1024} \longrightarrow \begin{bmatrix}45\\ \cdots\\ 68\\ \cdots\\ \cdots\\ 36\\ \cdots\\ 86\end{bmatrix}_{1024\times1}$$

- 将每幅人脸图像转换成列向量
- 如将一幅**32×32**的人脸图像转成**1024×1**的列向量

特征人脸: 算法描述

$Y \in R^{n\times l}$ $X \in R^{n\times d}$ $W \in R^{d\times l}$

- 输入： n个1024维人脸样本数据所构成的矩阵X，降维后的维数l
- 输出：映射矩阵$W = \{w_1, w_2, \dots, w_l\}$（其中每个$w_j (1 \le j \le l)$是一个特征人脸）
- 算法步骤：

1：对于每个人脸样本数据x_i进行中心化处理： $\boldsymbol{x}_i = \boldsymbol{x}_i - \boldsymbol{\mu}$， $\boldsymbol{\mu} = \frac{1}{n}\sum_{j=1}^{n} \boldsymbol{x}_j$

2：计算原始人脸样本数据的协方差矩阵：$\Sigma = \frac{1}{n-1} X^T X$

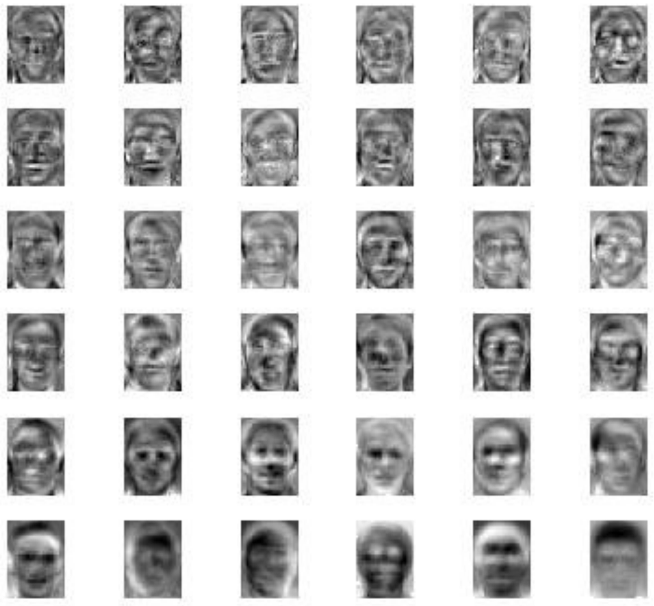
3：对协方差矩阵Σ进行特征值分解，对所得特征根从到小排序$\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_d$

4：取前l个最大特征根所对应特征向量$\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_l$组成映射矩阵W

5：将每个人脸图像x_i按照如下方法降维：$(\boldsymbol{x}_i)_{1\times d} (W)_{d\times l} = 1\times l$

特征人脸: 算法描述

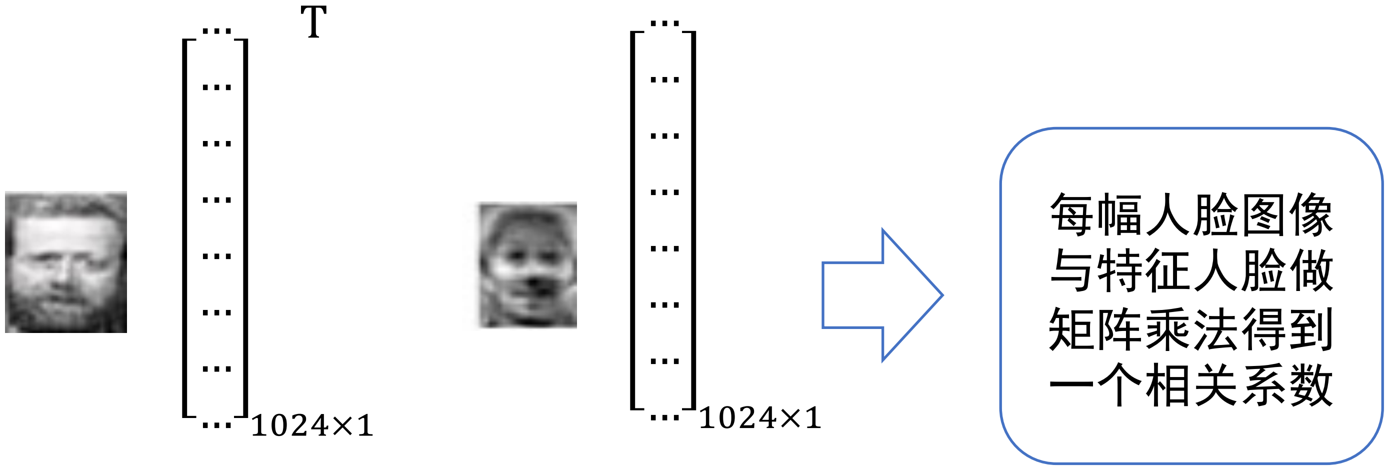
- 每个人脸特征向量$\boldsymbol{w}_i$与原始人脸数据$\boldsymbol{x}_i$的维数是一样的，均为1024。
- 可将每个特征向量还原为32×32的人脸图像，称之为特征人脸，因此可得到l个特征人脸。



400个人脸（左）和与之对应的36个特征人脸

基于特征人脸的降维

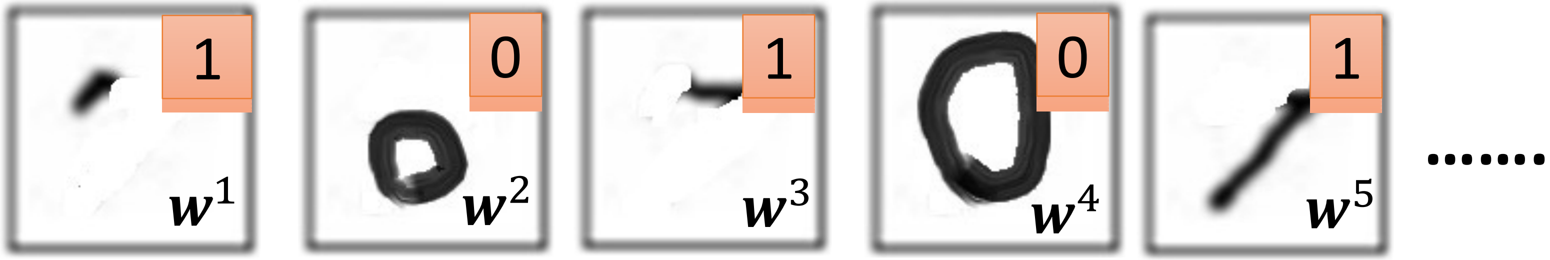
- 将每幅人脸分别与每个特征人脸做矩阵乘法，得到一个相关系数
- 每幅人脸得到l个相关系数⇒每幅人脸从1024维约减到l维



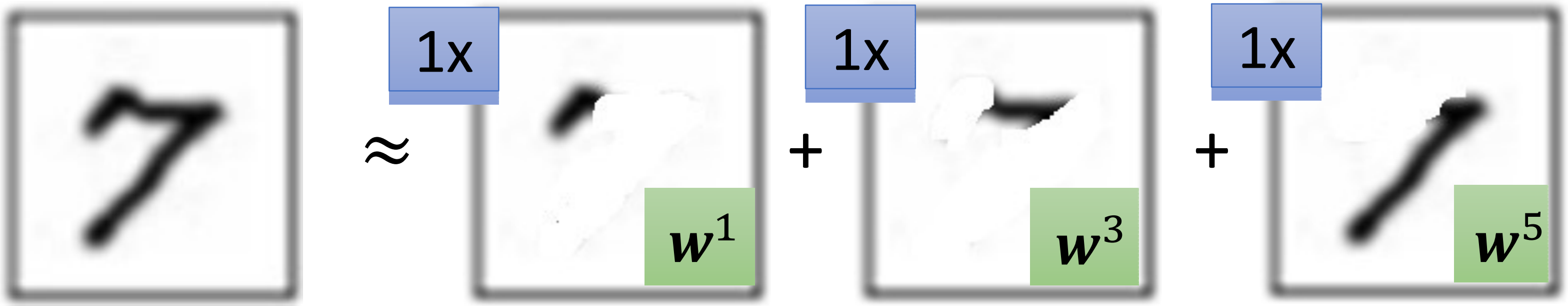
主成分分析: 另一个角度

$$\hat{x} = \sum_{k-1}^{K} y_k \boldsymbol{w}^k \Leftrightarrow (\boldsymbol{x} - \overline{\boldsymbol{x}}) \qquad y_k = (\boldsymbol{x} - \overline{\boldsymbol{x}}) \cdot \boldsymbol{w}^k$$

主成分:



$$\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ \vdots \end{bmatrix}$$



$$\boldsymbol{x} \approx y_1 \boldsymbol{w}^1 + y_2 \boldsymbol{w}^2 + \cdots + y_K \boldsymbol{w}^K + \overline{\boldsymbol{x}}$$

手写数字图像的像素值表达

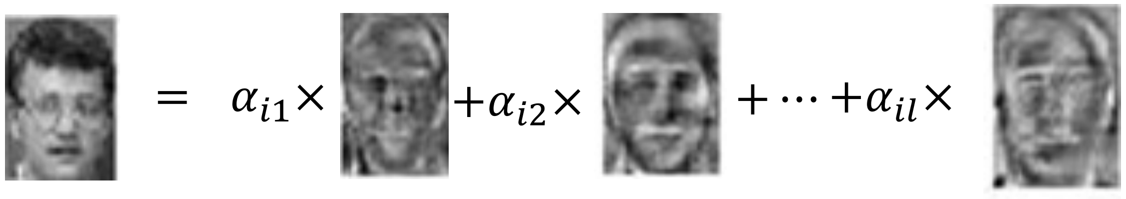
主成分

$$\begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_K \end{bmatrix}$$

手写数字图像新特征表达

基于特征人脸的降维

- 由于每幅人脸是所有特征人脸的线性组合，因此就实现人脸从“像素点表达”到“特征人脸表达”的转变。每幅人脸从1024维约减到l维。



$(\alpha_{i1},\ldots\ \alpha_{il})$

$\boldsymbol{x}_i$

使用l个特征人脸的线性组合来表达原始人脸数据$\boldsymbol{x}_i$

$\boldsymbol{x}_i$的像素点空间表达 32×32

$\boldsymbol{x}_i$的人脸子空间的l个系数表达

在后续人脸识别分类中，就使用这l个系数来表示原始人脸图像。即计算两张人脸是否相似，不是去计算两个32×32矩阵是否相似，而是计算两个人脸所对应的l个系数是否相似

人脸表达的方法对比：聚类、主成份分析、非负矩阵分解

$\boldsymbol{x}_i$

聚类表示：用待表示人脸最相似的聚类质心来表示

$\boldsymbol{x}_i$

$= \alpha_{i1}\times$  $+\alpha_{i2}\times$  $+\cdots+\alpha_{il}\times$ 

特征人脸(主成分分析）表示：使用l个特征人脸的线性组合来表达原始人脸数据$\boldsymbol{x}_i$

$\boldsymbol{x}_i$

非负矩阵人脸分解方法表示：通过若干个特征人脸的线性组合来表达原始人脸数据$\boldsymbol{x}_i$，体现了“部分组成整体”

Daniel D. Lee & H. Sebastian Seung, Learning the parts of objects by non-negative matrix factorization, 1999, Nature

人脸表达后的分析与处理