矩阵计算

李宇峰

liyf@nju.edu.cn

人工智能学院

矩阵和向量的应用实例

模式识别与机器学习中向量的相似比较

- 聚类（clustering）和分类（classification）是模式识别和机器学习等统计数据分析中的重要技术

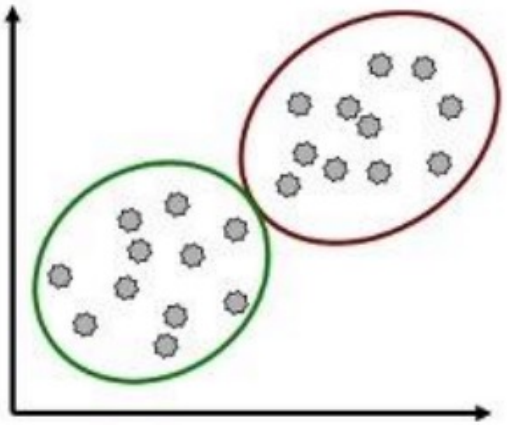
  - 聚类，就是将一给定的大数据集聚为几个小的子数据集，并且每个子集（目标类）内的数据要求各自具有共同或者相似的特征

  - 分类，就是已知有若干个目标类别，现要将一个（或者多个）未知类属的数据或特征值划分到具有最接近特征的某个已知目标类别中

聚类(clustering)和分类(classification)

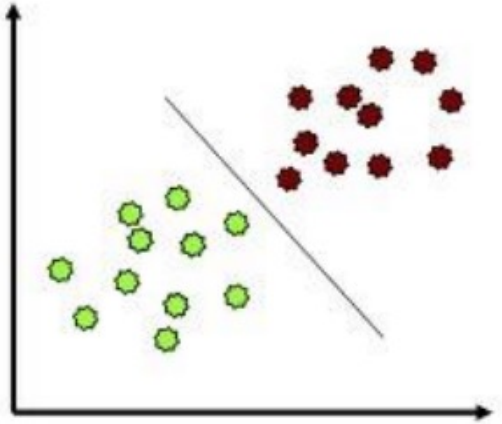
CLUSTERING

- Data is not labeled
- Group points that are "close" to each other
- Identify structure or patterns in data
- Unsupervised learning



CLASSIFICATION

- Labeled data points
- Want a "rule" that assigns labels to new points
- Supervised learning



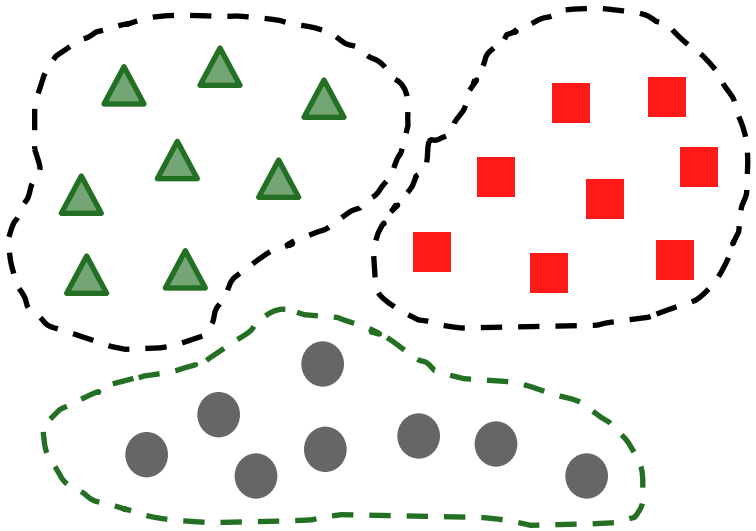
聚类 (Clustering)

在“无监督学习”任务中研究最多、应用最广

目标：将数据样本划分为若干个通常不相交的“簇”(cluster)

即可找寻数据内在的分布结构

也作为分类学习任务中提取特征、判断类别的重要支撑



课外知识：
不相交的簇=硬聚类；
可相交的簇=软聚类；

k-means（k-均值聚类）

每个簇中心以该簇中所有样本点的“均值”表示

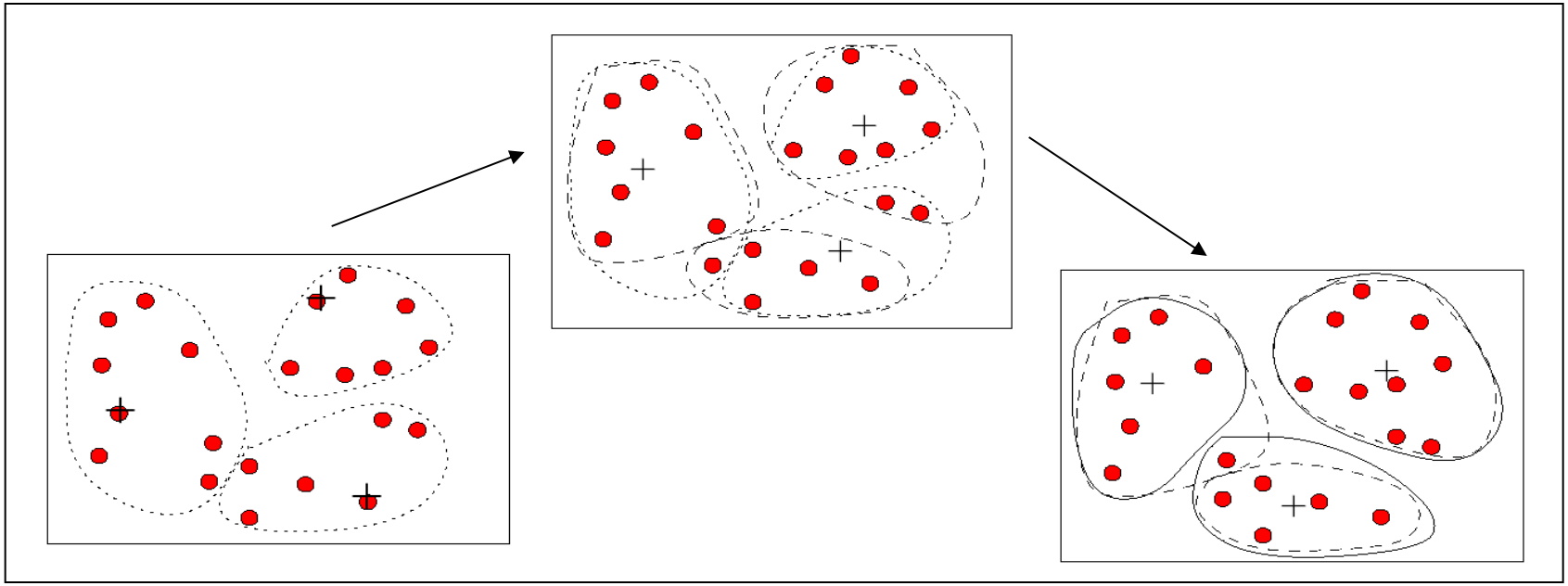
Step1: 随机选取k个样本点作为簇中心

Step2: 将其他样本点根据其与簇中心的距离，划分给最近的簇

Step3: 更新各簇的均值向量，将其作为新的簇中心

Step4: 若所有簇中心未发生改变，则停止；否则执行 Step 2

若不以均值向量为原型，而是以距离它最近的样本点为原型，则得到 k-medoids算法



k-means（k-均值聚类）

K-Means聚类算法等价于求解下式

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} dist(x, \mu_i)^2 = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2$$

$$J(c^{(1)}, c^{(2)}, \ldots, c^{(m)}; \mu_{(1)}, \ldots, \mu_{(k)}) = \frac{1}{m} \sum_{i=1}^{m} \left\| x^{(i)} - \mu_{c^{(i)}} \right\|^2$$

关键： k 值选取；距离计算

k 近邻学习器

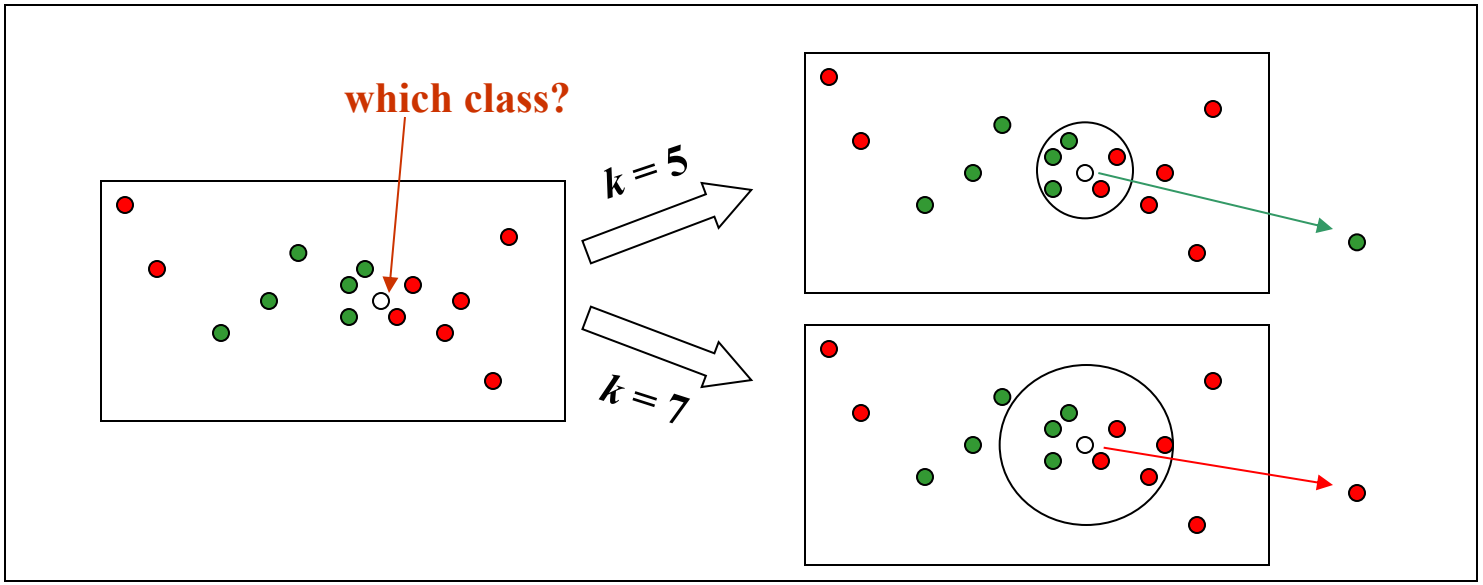
课外知识：
懒惰学习 = 事先没有分类器，见到测试样本再开始准备分类器

k 近邻 (k-Nearest Neighbor, kNN)

懒惰学习 (lazy learning) 的代表

基本思路：
近朱者赤，近墨者黑

(投票法；平均法)



关键： k 值选取；距离计算

维数灾难

但是在真实的应用中，我们是否能够准确的找到k近邻呢？

密采样(dense sampling)假设：样本的每个\eps-邻域都有近邻

如果近邻的距离阈值设为10^{-3}

假定维度为20，如果样本需要满足密采样条件需要的样本数量近10^{60}

想象一下：一张并不是很清晰的图像：70余万维，我们为了找到恰当的近邻，需要多少样本？

流形学习 - ISOMAP

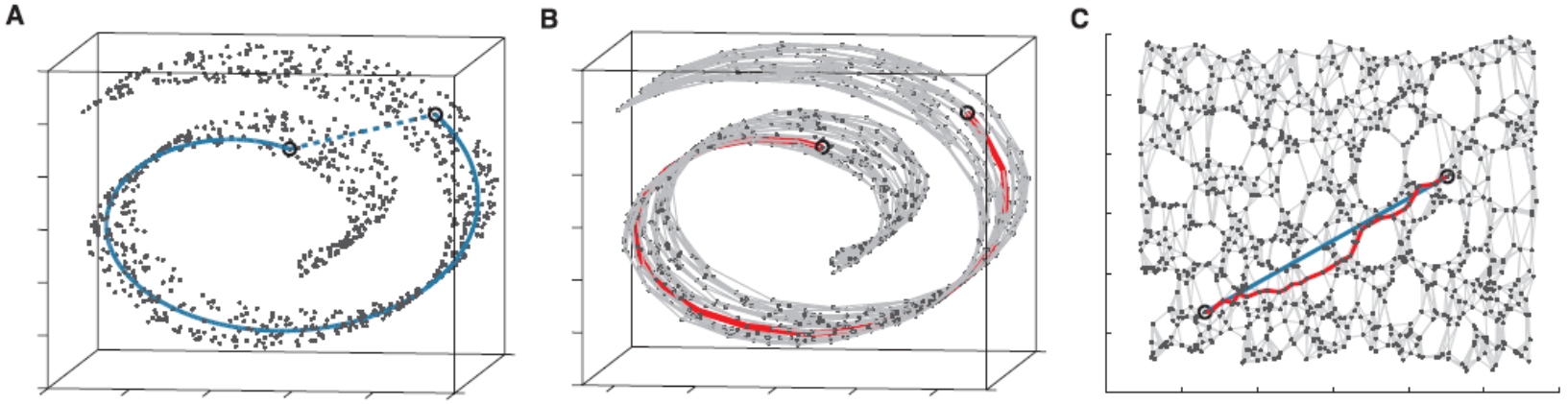


Fig. 3. The "Swiss roll" data set, illustrating how Isomap exploits geodesic paths for nonlinear dimensionality reduction. (**A**) For two arbitrary points (circled) on a nonlinear manifold, their Euclidean distance in the high-dimensional input space (length of dashed line) may not accurately reflect their intrinsic similarity, as measured by geodesic distance along the low-dimensional manifold (length of solid curve). (**B**) The neighborhood graph G constructed in step one of Isomap (with $K = 7$ and $N =$ 1000 data points) allows an approximation (red segments) to the true geodesic path to be computed efficiently in step two, as the shortest path in G. (**C**) The two-dimensional embedding recovered by Isomap in step three, which best preserves the shortest path distances in the neighborhood graph (overlaid). Straight lines in the embedding (blue) now represent simpler and cleaner approximations to the true geodesic paths than do the corresponding graph paths (red).

www.sciencemag.org SCIENCE VOL 290 22 DECEMBER 2000

基本步骤：

关键思路：测地线距离（近似）、保距

- 构造近邻图
- 基于最短路径算法近似任意两点之间的测地线(geodesic)距离
- 基于距离矩阵通过MDS获得低维嵌入

距离测度

- 在实现聚类和分类的过程中，其中一个重要环节是度量向量与向量、 向量与集合的相似程度。

- 如何来度量两个向量之间的相似程度？在聚类或分类中：用距离测度来度量两个未知向量间的相似度

- 距离测度：

  常用符号 $D(p \,||\, q)$ 表示向量p到向量q的距离。

距离测度

两个向量之间的距离$D(p \,||\, q)$, 必是满足下列条件的非负实数：

(1) 非负性和正定性：即对$\forall p, q, D(p \,||\, q) \geq 0$, 而$D(p \,||\, q) = 0$的充要条件是$p = q$；

(2) 对称性：即向量p到q的距离与q到p的距离相等，

$$D(p \,||\, q) = D(q \,||\, p);$$

(3) 三角不等式：两点之间的直线距离小于折线距离，即

$$\forall p, g, z, D(p \,||\, z) \leq D(p \,||\, g) + D(g \,||\, z)$$

这就是定义距离测度的三条公理。

定义了距离的线性空间就称为距离空间。

距离测度

- 距离是相异度（dissimilarity）的度量：
  - 相异度越小的两个向量越相似

令 $D(x, s_i)$ 表示未知模式向量x与已知模式向量 s_i 之间的距离。

以x与s_1, s_2的距离为例，若$D(x, s_1) \leq D(x, s_2)$, 那么就意味着未知模式向量x与样本模式向量s_1更相似。

距离度量的种类

选哪个距离度量呢？能否学出来一个最好的距离度量？——距离度量学习

- 欧式距离

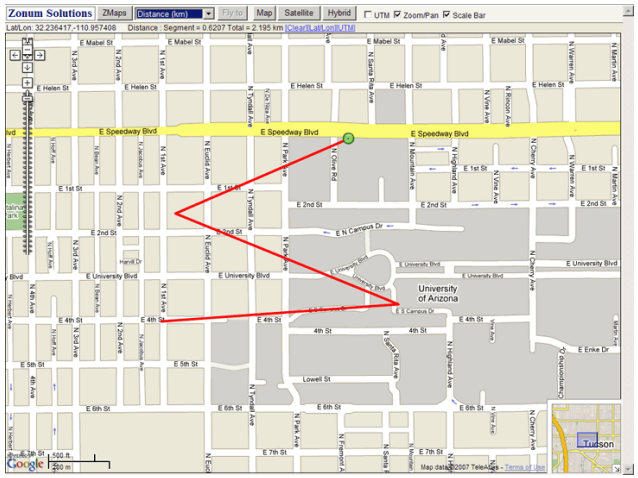
$$D_E(x, s_i) = \|x - s_i\|_2 = \sqrt{(x - s_i)^T (x - s_i)}\,.$$

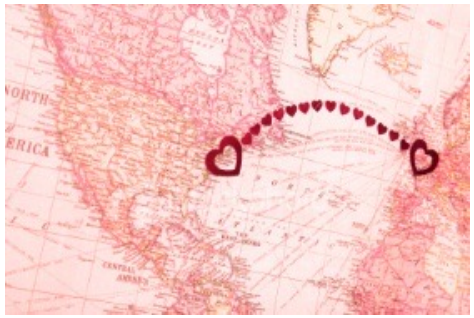
- 曼哈顿距离

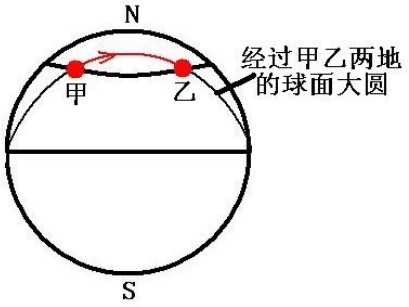
$$\mathrm{dist}_{\mathrm{mk}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left(\sum_{u=1}^{n} |x_{iu} - x_{ju}|^p \right)^{\frac{1}{p}}$$

p = 2: 欧氏距离(Euclidean distance)

p = 1: 曼哈顿距离(Manhattan distance)



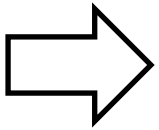




- 测地距离（Geodesic Distance）

距离度量学习 (distance metric learning)

希望找到一个“合适”低维空间下的距离度量



能否直接“学出”合适的距离？

首先，要能够通过 “参数化”来学习距离度量

马氏距离 (Mahalanobis distance) 是一个很好的选择：

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

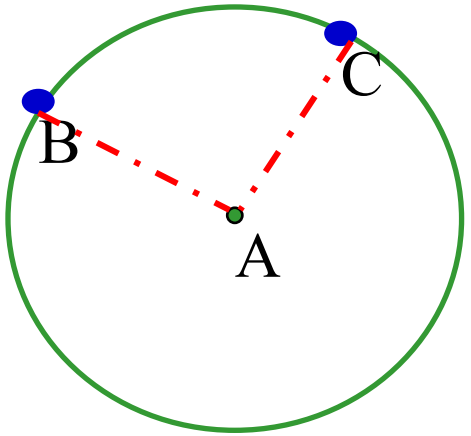
其中 $\mathbf{M}$ 称为“度量矩阵”，数学上它是一个半正定对称矩阵

距离度量学习就是要对 $\mathbf{M}$ 进行学习

距离度量学习 (distance metric learning)

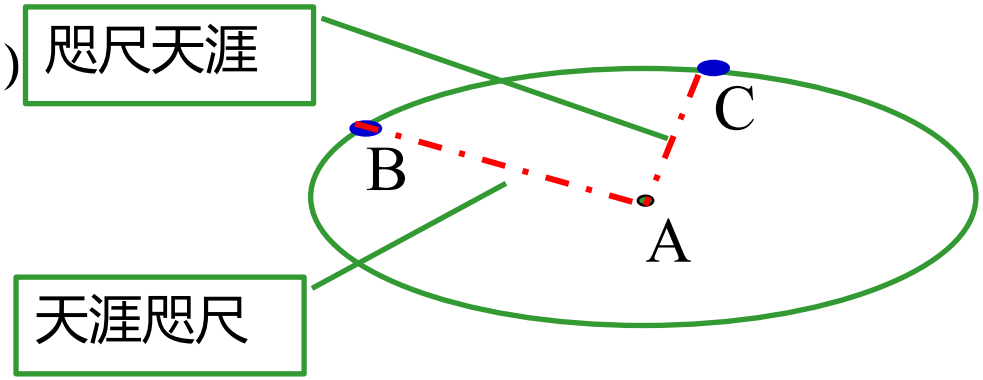
为什么是马氏距离？

欧氏距离的缺陷——各个方向同等重要



但是：

- 有缘千里来相会
  (表面欧氏距离大但实际距离小)
- 无缘对面手难牵
  (表面欧式距离小但实际距离大)
- 马氏距离应运而生



Example (MATLAB)

```
X = mvnrnd([0;0],[1 .9;.9 1],100); % zero-mean correlated data
Y = [1 1;1 -1;-1 1;-1 -1]; % four points in 2D
d1 = mahal(Y,X)      % Mahalanobis distance
d2 = sum((Y-repmat(mean(X),4,1)).^2, 2)  % Euclidean distance
scatter(X(:,1),X(:,2))
hold on
scatter(Y(:,1),Y(:,2),100,d1,'*','LineWidth',2)
hb = colorbar;
ylabel(hb,'Mahalanobis Distance')
legend('X','Y','Location','NW')
```

马氏距离

d1=
0.6288
19.3520
21.1384
0.9409

d2=
1.6170
1.9334
2.1094
2.4258

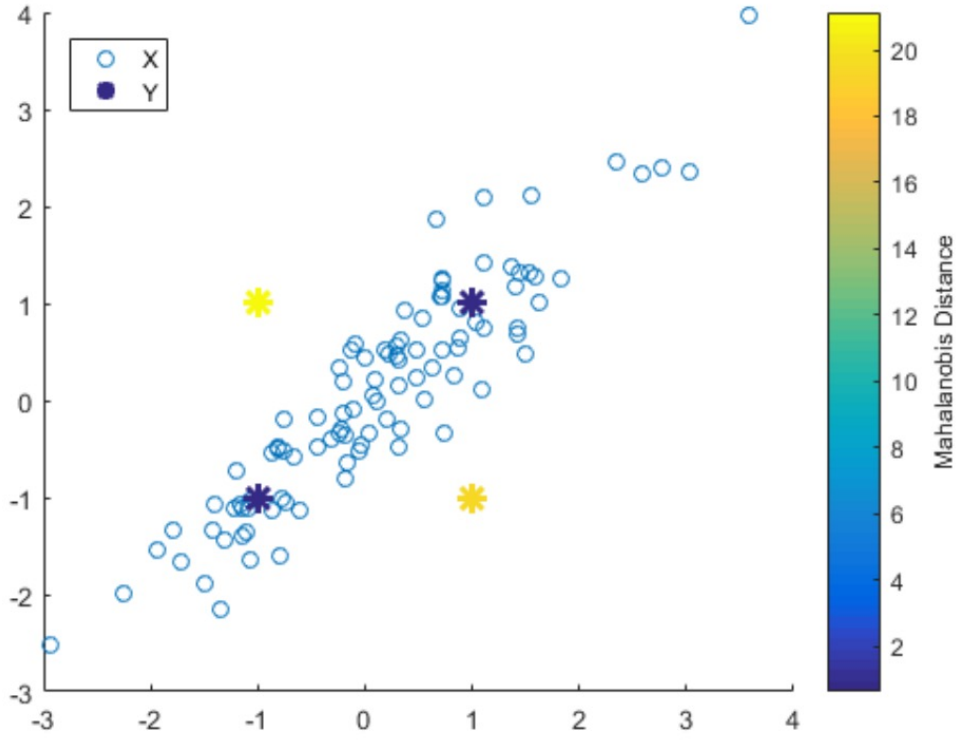


Fig. 2. Mahalanobis distance and Euclidean distance of four points (Y) to a set of zero-mean correlated data (X)

距离度量学习 (distance metric learning)

M怎么学习？要有个合适的目标函数

- 目标1：结合具体分类器的性能

  例如，以近邻分类器的性能为目标，则得到经典的NCA算法

- 目标2：结合领域知识

  例如，若已知“必连”(must-link) 约束集合$\mathcal{M}$ 与“勿连”(cannot-link) 约束集合$\mathcal{C}$，则可通过求解下述凸优化问题得到 $\mathbf{M}$:

$$\min_{\mathbf{M}} \sum_{(\boldsymbol{x}_i,\boldsymbol{x}_j)\in\mathcal{M}} \|\boldsymbol{x}_i-\boldsymbol{x}_j\|_{\mathbf{M}}^2$$

$$\text{s.t.} \sum_{(\boldsymbol{x}_i,\boldsymbol{x}_k)\in\mathcal{C}} \|\boldsymbol{x}_i-\boldsymbol{x}_k\|_{\mathbf{M}}^2 \geqslant 1 ,$$

$$\mathbf{M} \succeq 0 ,$$

余弦距离与应用

向量之间的相异度的测度不一定局限于距离函数。两个向量之间的锐角夹角的余弦函数

$$D(x, s_i) = cos(\theta_i) = \frac{x^T s_i}{\|x\|_2 \|s_i\|_2}$$

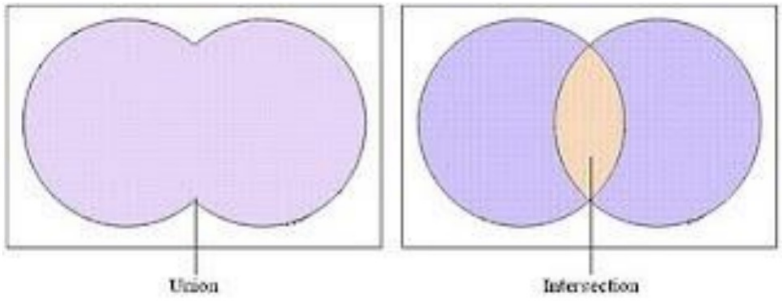
也是一种相异度的有效测度。若$cos(\theta_i) > cos(\theta_j)$, 对$\forall j \neq i$成立，则认为未知模式向量$x$与样本向量$s_i$最相似。

应用：新闻分类

Jaccard 距离与应用

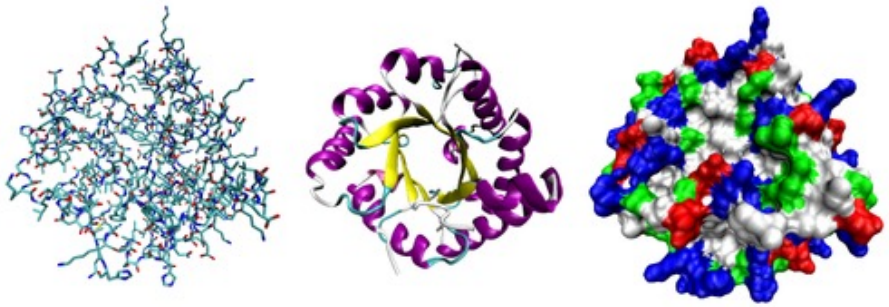
Tanimoto Coefficient（Jaccard Coefficient）主要用于计算布尔值度量的个体间的相似度。只关心个体间共同具有的特征是否一致。其值等于两个用户共同关联（不管喜欢还是不喜欢）的物品数量除于两个用户分别关联的所有物品数量。

$$D(x, s_i) = \frac{|X \cap Y|}{|X \cup Y|}$$



或者 $D(x, s_i) = \frac{x^T s_i}{x^T x + s_i^T s_i - x^T s_i}$

x和s_i两个向量都为bit sequence



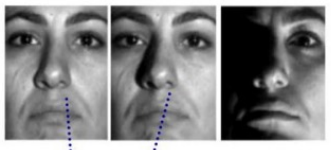
应用：疾病诊断

人脸识别的稀疏表示

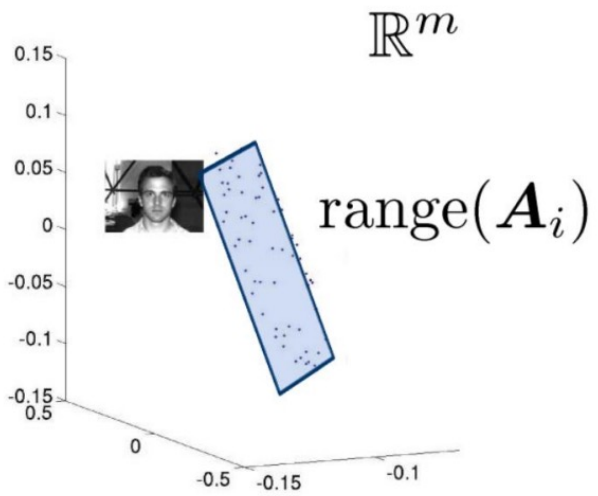
如果一个向量的分量中大多数是零元素那就称该向量是稀疏向量（sparse vector）或者稀疏矩阵（sparse matrix）。仅需少量的基本信号的线性组合就能表示一个目标信号，这就称为信号的稀疏表示。

稀疏表示是机器学习和模式识别等领域近几年的研究和应用的热点，也是大数据问题中中需要研究的重要课题之一。

Face Recognition



$$\boldsymbol{A}_i = [\,|\,|\,|\,\ldots\,] \in \mathbb{R}^{m \times n_i}$$



$$\boldsymbol{y} \approx x_{i,1} + x_{i,2} + \ldots + x_{i,n} = \boldsymbol{A}_i \boldsymbol{x}_i$$

人脸识别的稀疏表示

人脸设别问题：假定共有c类目标（人）,每类目标的脸部的每一幅训练图像的矩阵表示结果已经向量化表示成m维列向量(其中$m = R_1 \times R_2$为一幅图像的采样样本数目，例如m =512×512)，并且每个列向量的元素都已经归一化，使得每个列向量的*Euclidean*范数等于1。

第i类目标的脸部在不同照度下拍摄的N_i个训练图像，可以表示成$m \times N_i$数据矩阵$D_i = [d_{i,1}, d_{i,2}, \ldots, d_{i,N_i}] \in R^{m \times N_i}$。

人脸识别的稀疏表示

为了保证人脸识别的分辨率，假定每一个训练集D_i都比较大，第i个实验对象在另一照度下拍摄的新图像y即可以表示成已知训练图像的一线性组合$y \approx D_i\alpha_i$, 其中$\alpha_i \in R^{N_i}$为系数向量，决定未知人脸的目标类别i，问题是：

在实际应用中，往往不知道新的实验样本的具体目标属性，因而线性方程组$y \approx D_i\alpha_i$无法直接得到，导致系数向量α_i无法求解得到。

Face Recognition

如果我们大致知道或者猜测到新的测试样本是c类目标中的某类目标的信号，就可以将这c类目标的训练集合写成一个训练数据矩阵

$$D = [D_1, D_2, \dots, D_c]$$

$$= [d_{1,1}, d_{1,2}, \dots, d_{1,N_1}, \dots, d_{c,1}, d_{c,2}, \dots, d_{c,N_c}] \in R^{m \times n}$$

其中$N = \sum_{i=1}^{c} N_i$表示所有c类目标的训练图像的总个数。

Face Recognition

于是待设别的人脸图像y可以表示成线性组合

$$y = D\alpha_0 = [d_{1,1}, d_{1,2}, \dots, d_{1,N_1}, \dots, d_{c,1}, d_{c,2}, \dots, d_{c,N_c}] \begin{bmatrix} 0_{N_1} \\ \vdots \\ 0_{N_{i-1}} \\ \alpha_i \\ 0_{N_{i+1}} \\ \vdots \\ 0_{N_c} \end{bmatrix}$$

其中$0_{N_k}(k = 1, \dots, i-1, i+1, \dots, c)$为$N_k$维零向量，并且$\alpha_0$是一个非零向量。

Face Recognition

这样一来，人脸识别便变成一个线性方程组的求解问题

或者线性求逆问题：

已知 数据向量 y 和 数据矩阵 D

求线性方程组$y = D\alpha$的稀疏解向量α_0.

Face Recognition

需要注意的是：通常$m < N$, 故$y = D\alpha_0$为亚定线性方程组，它存在无穷多组解。其中最稀疏的解向量α_0（它只有α_i非零，其他部分全部为零）才是我们感兴趣的解。

由于解向量必须是稀疏向量，所以人脸识别问题可以描述成一个约束优化问题：在$y = D\alpha_0$的约束下，使得解向量 L_0“范数”（非零元素的个数）最小化：

$$min\|\alpha_0\|_0 \quad subject\ to \quad y = D\alpha_0$$

Face Recognition

代数与矩阵基础

- 向量，矩阵
- 线性空间，线性子空间
  - 零空间，像空间
- 线性相关，线性无关
  - 空间的维数与基底
- 矩阵的基本运算
  - 转置，共轭转置
  - 相乘，求逆
- 矩阵的数值特征
  - 行列式，二次型，特征值，trace，rank
- 内积与范数
  - 向量范数
  - 矩阵范数
- 应用
  - 马氏距离，欧氏距离
  - Tanimoto coefficient
  - 聚类、k近邻，度量学习
  - 人脸图像的稀疏表示