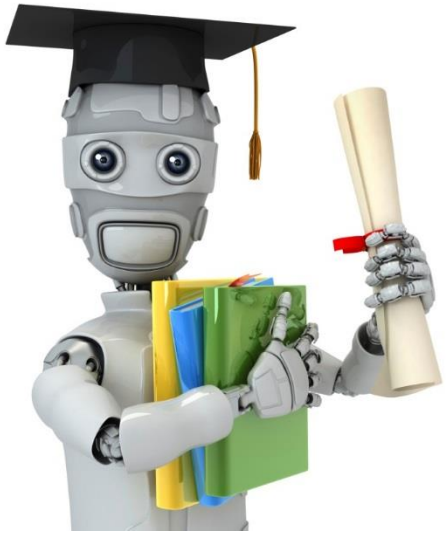


Machine Learning

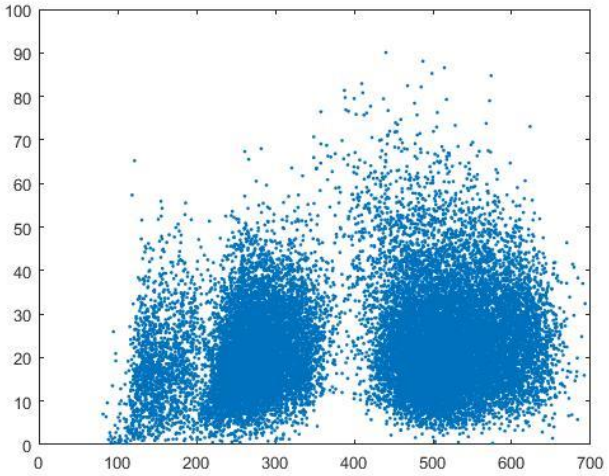
聚类
(Clustering)

梁毅雄

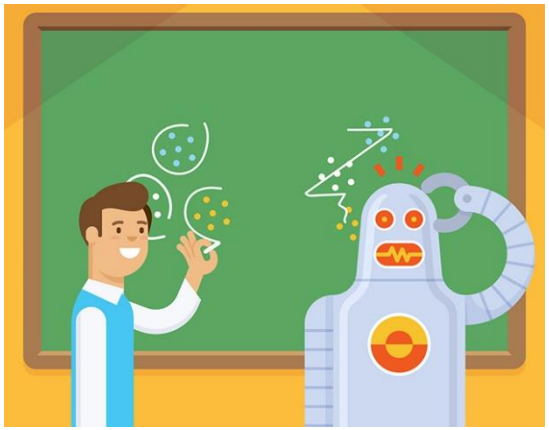
yxliang@csu.edu.cn

Some materials from Andrew Ng, Eric Xing, Andrew W. Moore and others

聚类(Clustering)



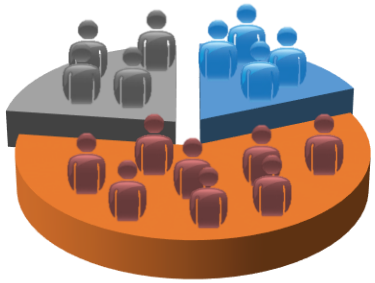
- What is clustering
  - Are there any “grouping” them ?
  - What is each group ?
  - How many ?
  - How to identify them?
  - …



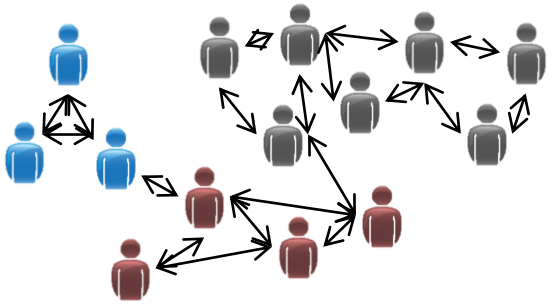
聚类(Clustering)

- 聚类：
  - the process of grouping a set of objects into classes of similar objects
  - 同一簇中的样本尽可能的相似，而不同簇中的样本尽可能不同
  - 高类内相似性(high intra-class similarity)
  - 低类间相似性(low inter-class similarity)
  - 属于最典型的非监督学习方法

Applications of clustering

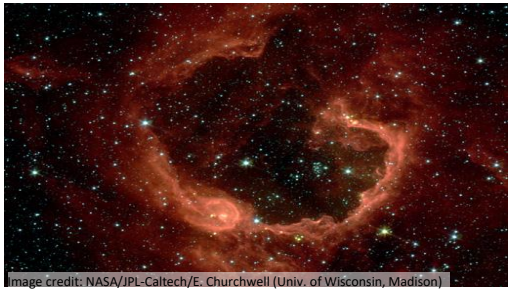


Market segmentation



Social network analysis

Organize computing clusters



Astronomical data analysis

聚类

- People

- Images

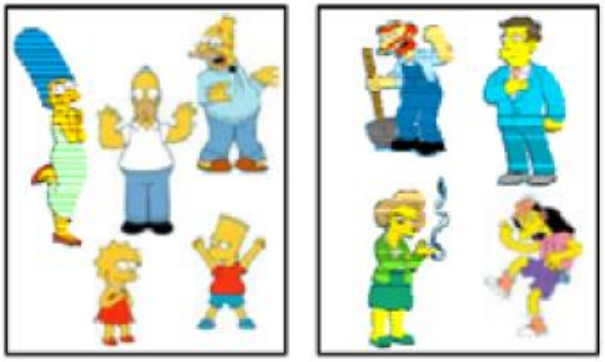
- Language

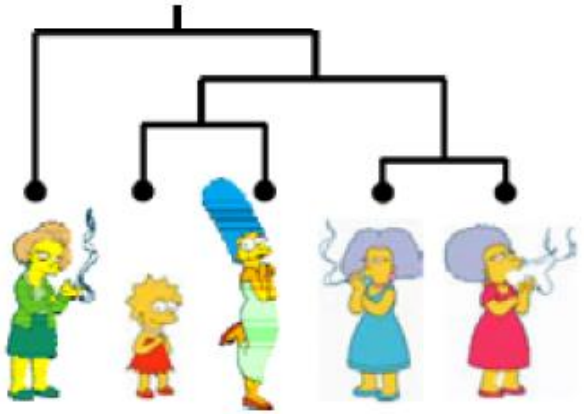
Piotr Pyotr Petros Pietro Pedro Pierre Piero Peter Peder Peka Peadar

- species

聚类的应用

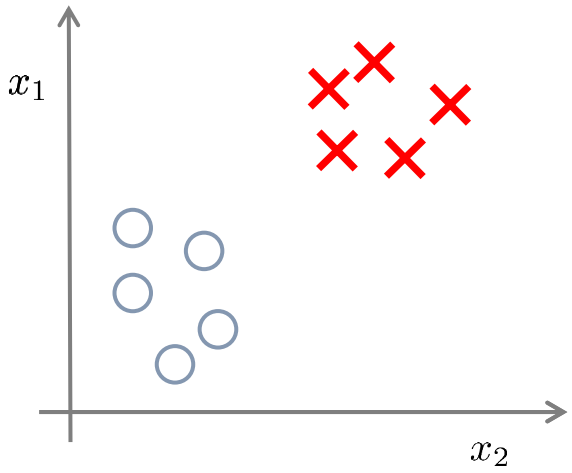
- 聚类方法大致可分为：
  - Partitional algorithms: 划分为互不相交的簇
    - 基于原型的聚类：K-Means, GMM等
    - 基于密度的聚类：DBSCAN, MeanShift等
  - Hierarchical algorithms
    - Bottom-up, agglomerative
    - Top-down, divisive



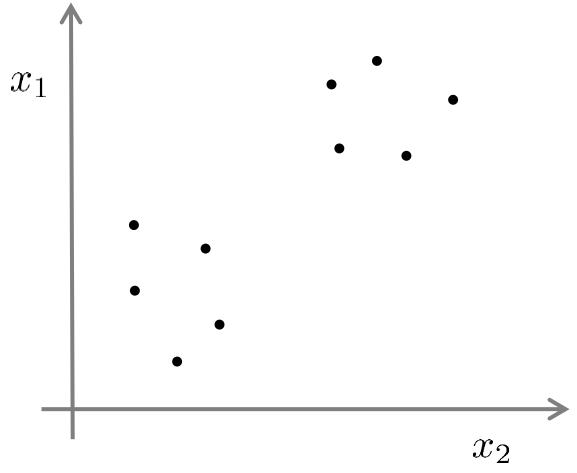


Recap:监督学习 vs.非监督学习

非监督学习： learning from raw (unlabeled, unannotated, etc) data, as opposed to supervised data where a classification of examples is given



Training set: $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), (x^{(3)}, y^{(3)}), \ldots, (x^{(m)}, y^{(m)})\}$

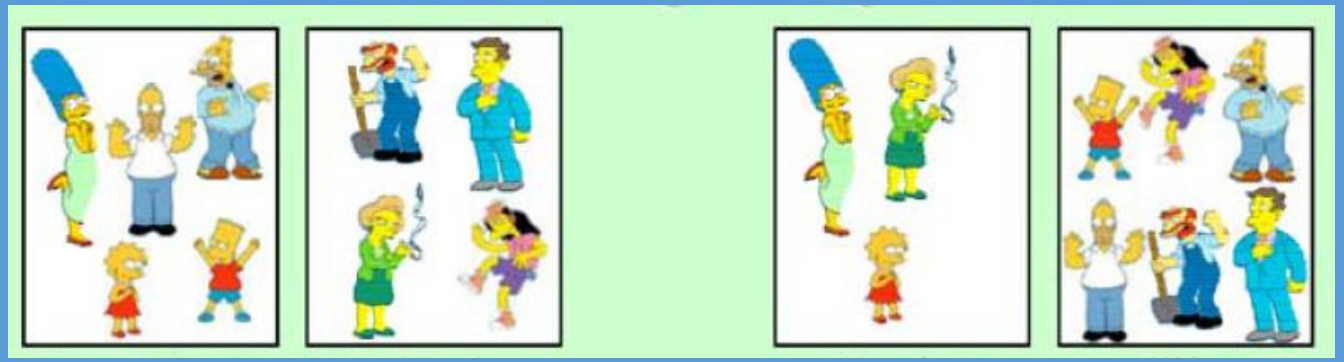


Training set: $\{x^{(1)}, x^{(2)}, x^{(3)}, \ldots, x^{(m)}\}$

聚类需要考虑的问题

- What is a natural grouping among these objects?
  - Definition of "groupness"
- What makes objects “related”?
  - Definition of "similarity/distance"
- How many clusters?
  - Fixed a priori?
  - Completely data driven?
- 如何评价聚类算法的有效性？

聚类的依据



聚类是主观的

如何定义相似性(或距离)

Hard to define!
But *we know it*
when we see it

Recap: 距离

若$x \in \mathbb{R}^n, y \in \mathbb{R}^n, z \in \mathbb{R}^n$, 函数$D(x, z)$为一个距离度量需要满足如下条件：

- Symmetry: $D(x, z) = D(z, x)$
- Constancy of Self-Similarity: $D(x, x) = 0$
- Positivity Separation: $D(x, z) \geq 0$ and $D(x, z) = 0$ iif $x = z$
- Triangular Inequality: $D(x, z) \leq D(x, y) + D(y, z)$

Recap: Minkowski距离

Minkowski距离: $D_{mk}(x,z) = \left(\sum_{i=1}^{n} |x_i - z_i|^p\right)^{\frac{1}{p}}$

若$p = 2$, 则Minkowski距离即为欧式距离:

$D_{ed}(x,z) = \|x - z\| = \sqrt{\sum_{i=1}^{n} |x_i - z_i|^2}$

若$p = 1$, 则Minkowski距离即为曼哈顿距离(城市距离):

$D_{man}(x,z) = \|x - z\|_1 = \sum_{i=1}^{n} |x_i - z_i|$

若$p = +\infty$, 则Minkowski距离为“sup”距离:

$D_{sup}(x,z) = \|x - z\|_\infty = \max_{i=1}^{n} |x_i - z_i|$

Recap: Hamming距离

- 当特征为二值特征时， Minkowski距离称为Hamming距离
- 如： Gene Expression Levels Under 17 Conditions (1-High,0-Low)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
GeneA	0	1	1	0	0	1	0	0	1	0	0	1	1	1	0	0	1
GeneB	0	1	1	1	0	0	0	0	1	1	1	1	1	1	0	1	1

Hamming Distance = ?

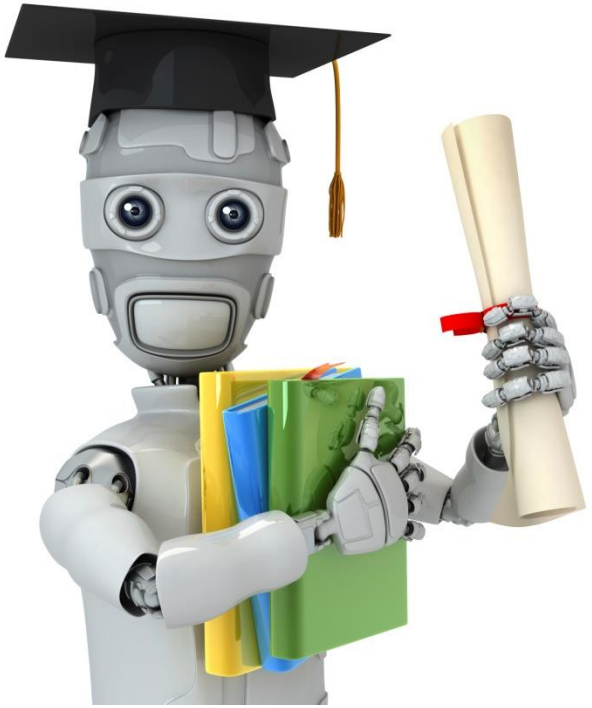
Recap: 相似性度量

- Pearson correlation coefficient

$$S_p(x,z) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(z_i - \bar{z})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \times \sum_{i=1}^{n}(z_i - \bar{z})^2}}$$

- 余弦距离

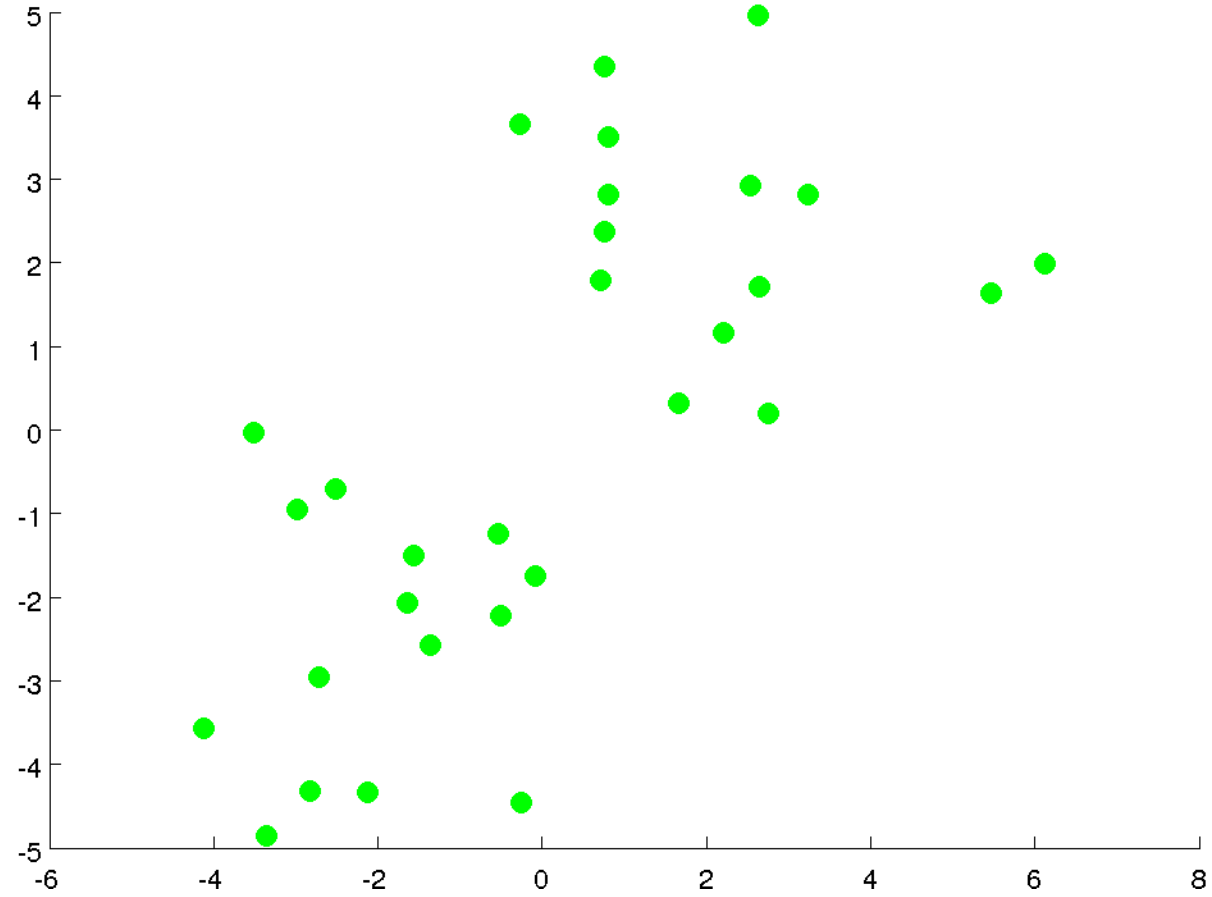
$$S_c(x,z) = \frac{x^T z}{\|x\|\|z\|}$$

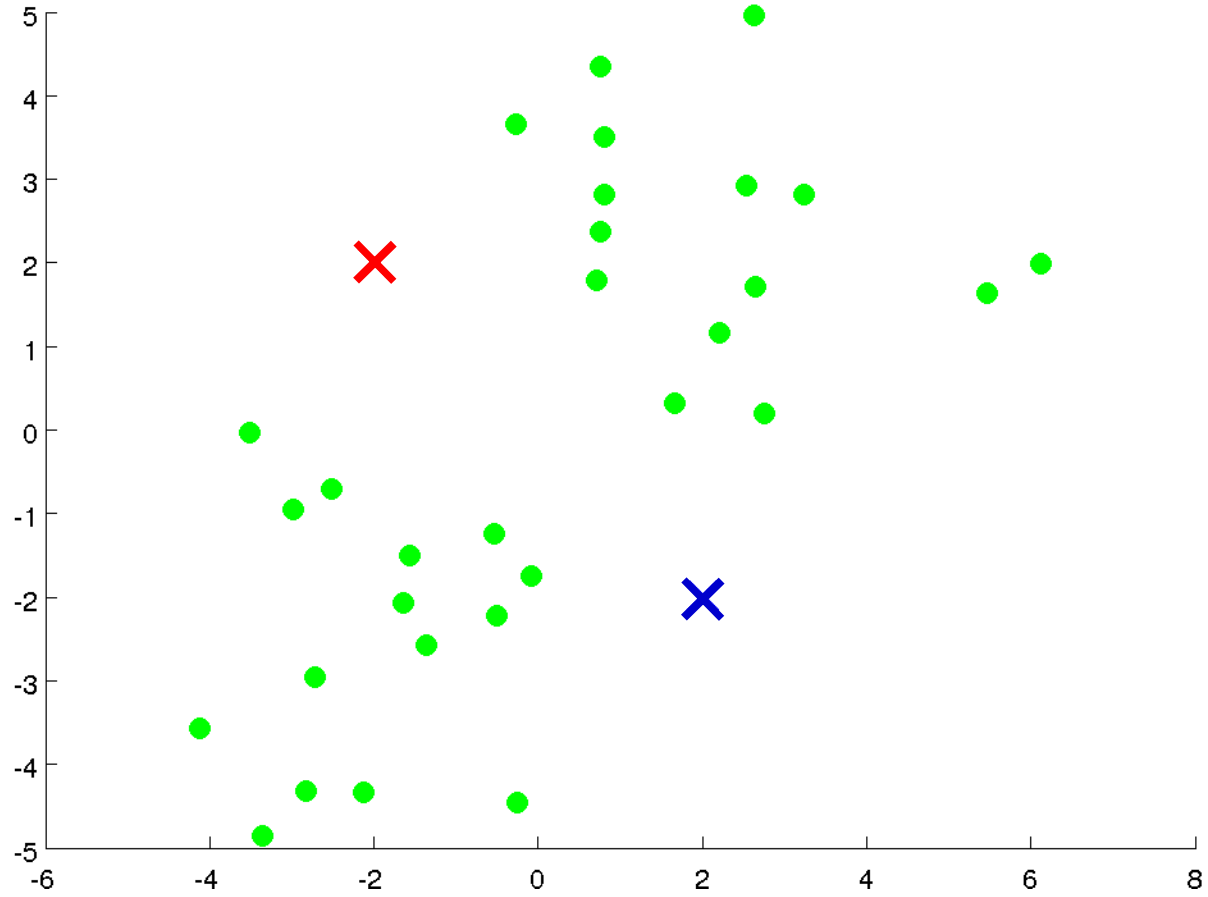


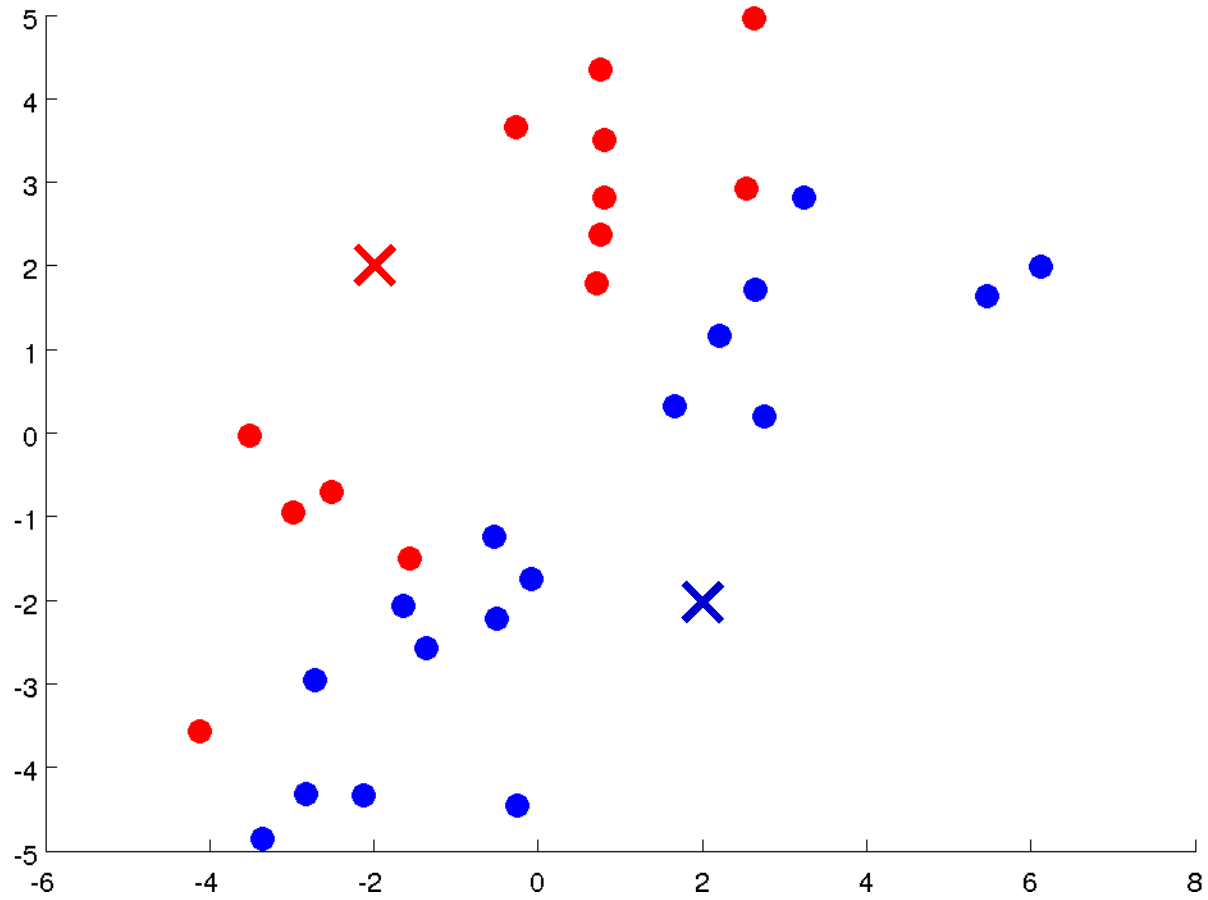
Machine Learning

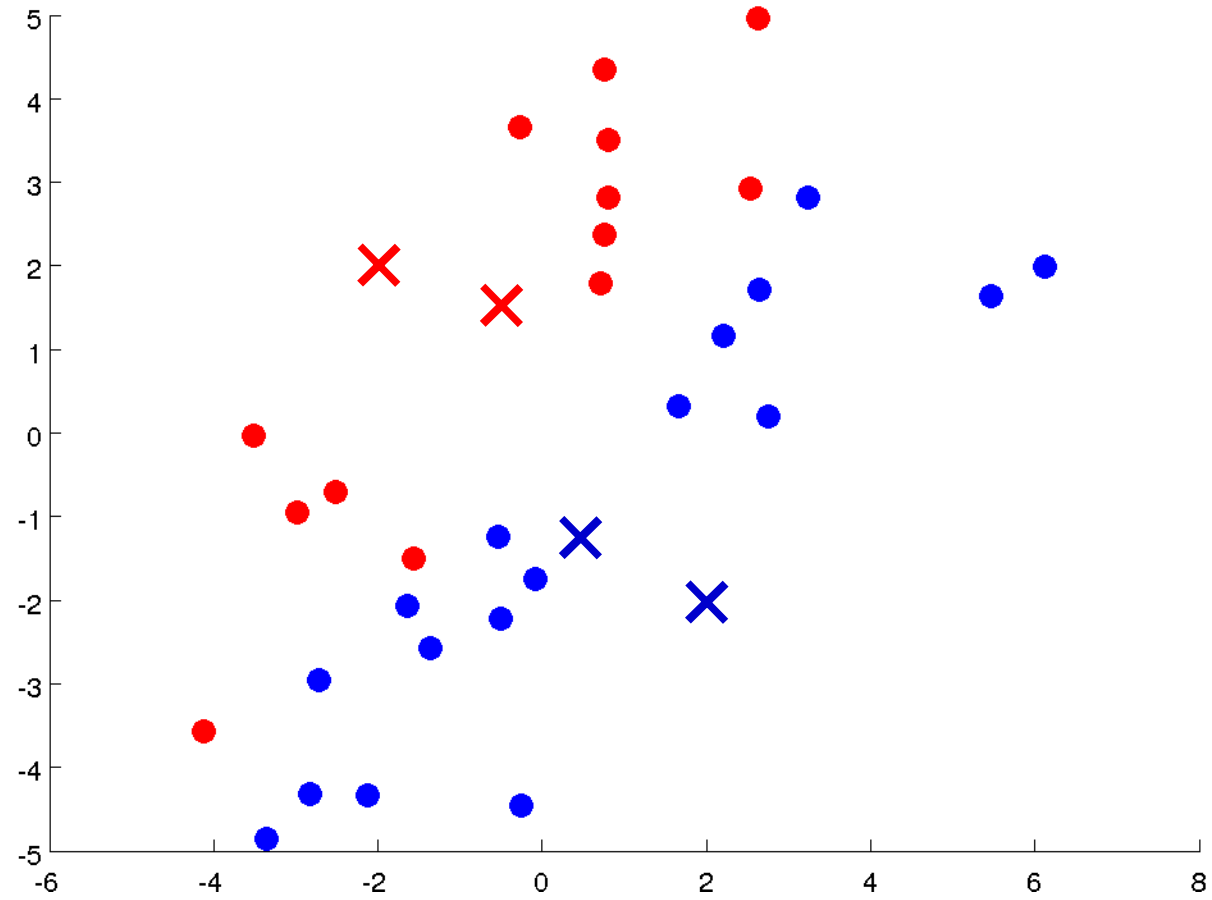
Clustering

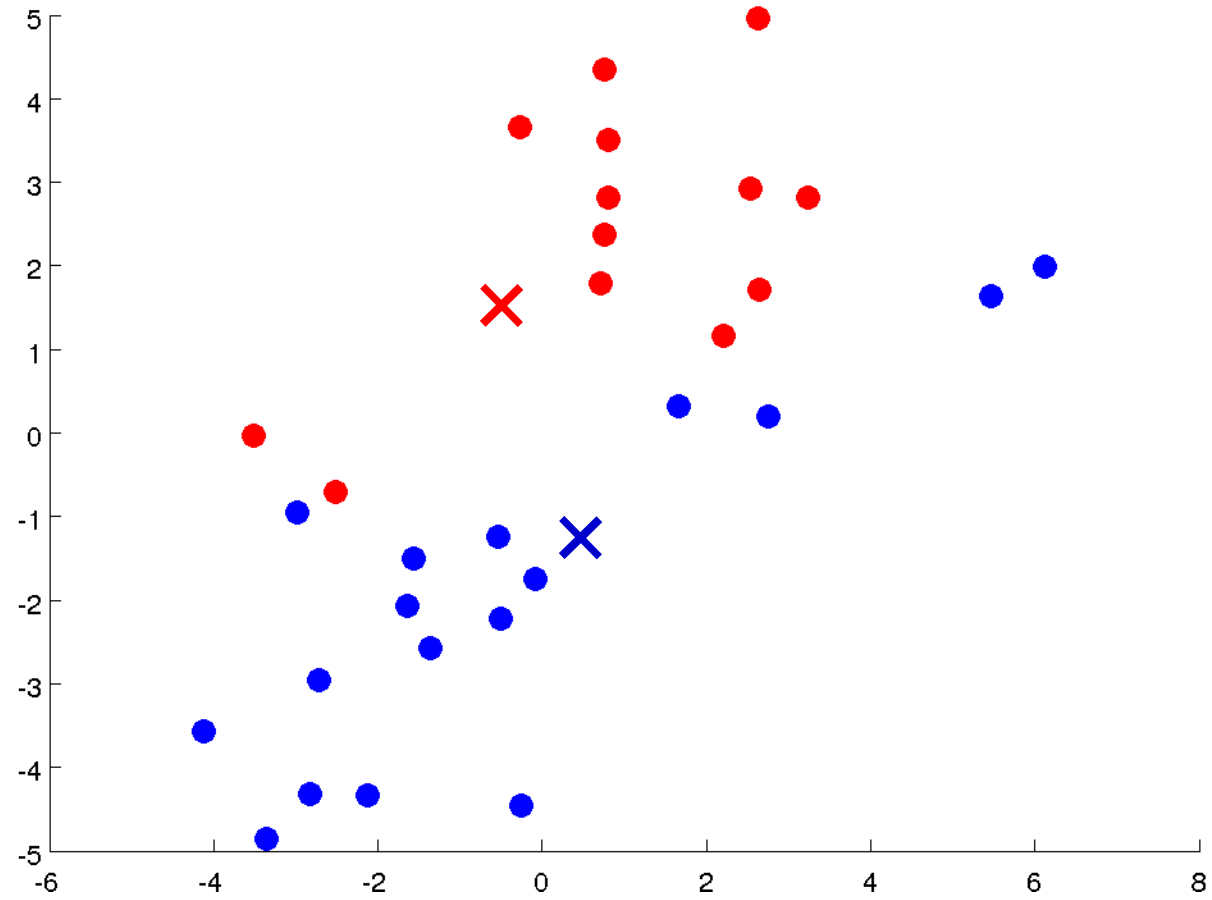
K-means algorithm

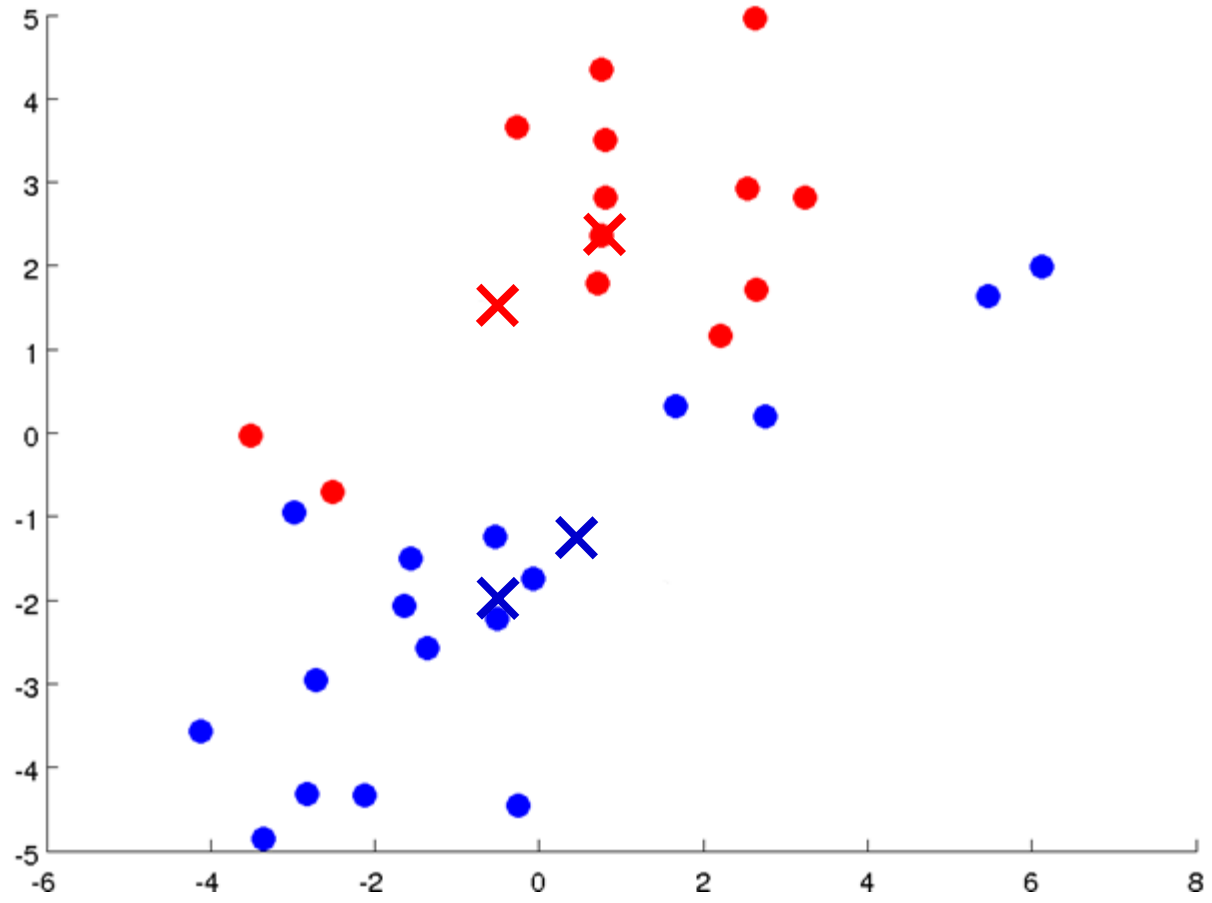

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

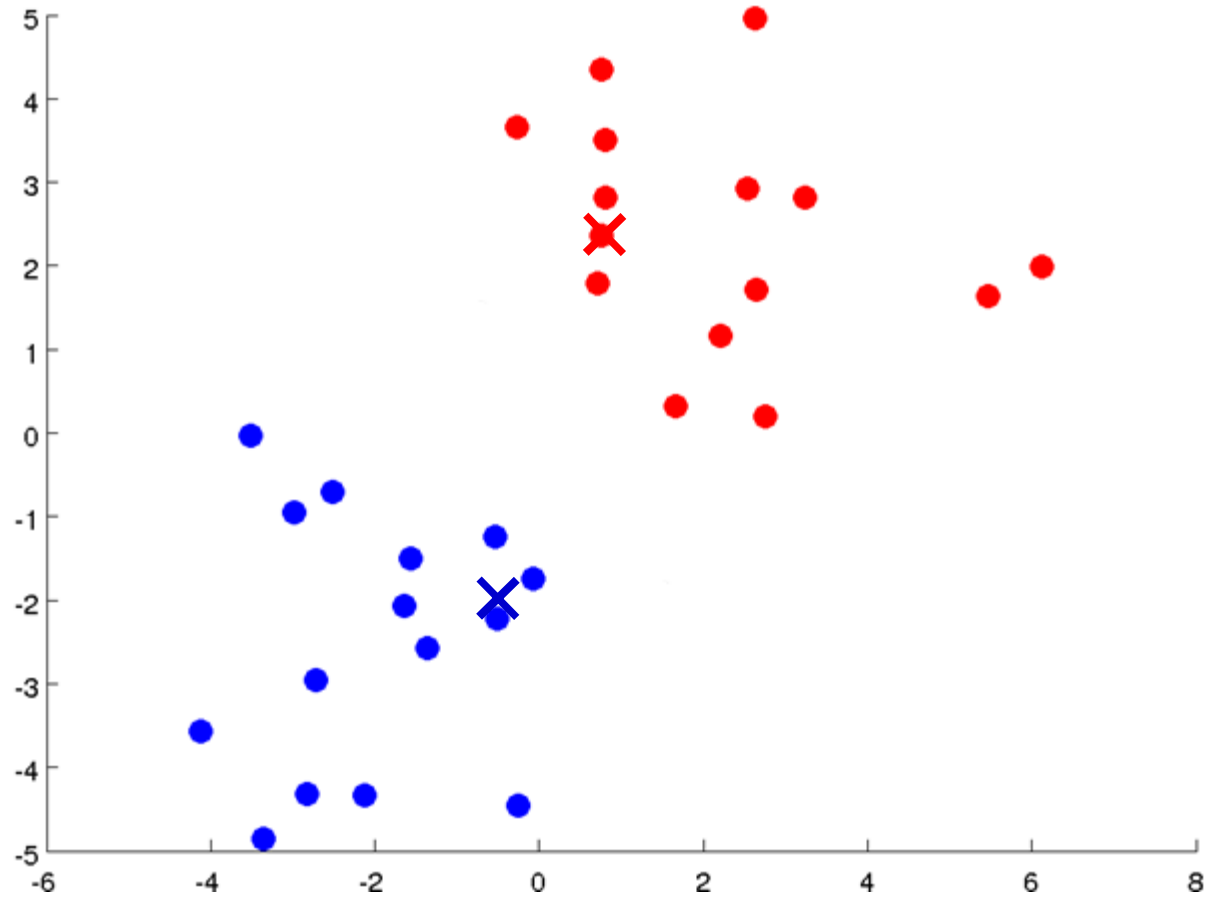

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

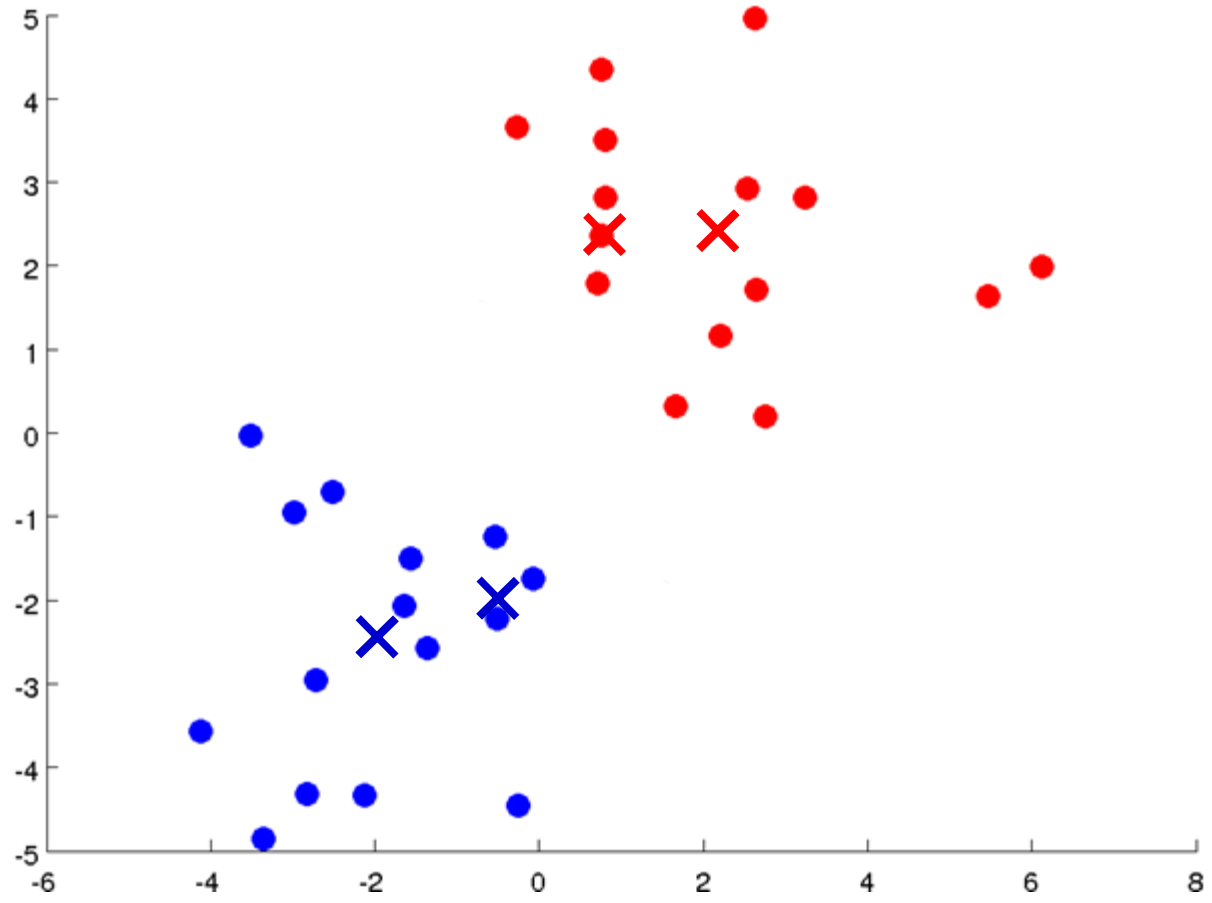

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

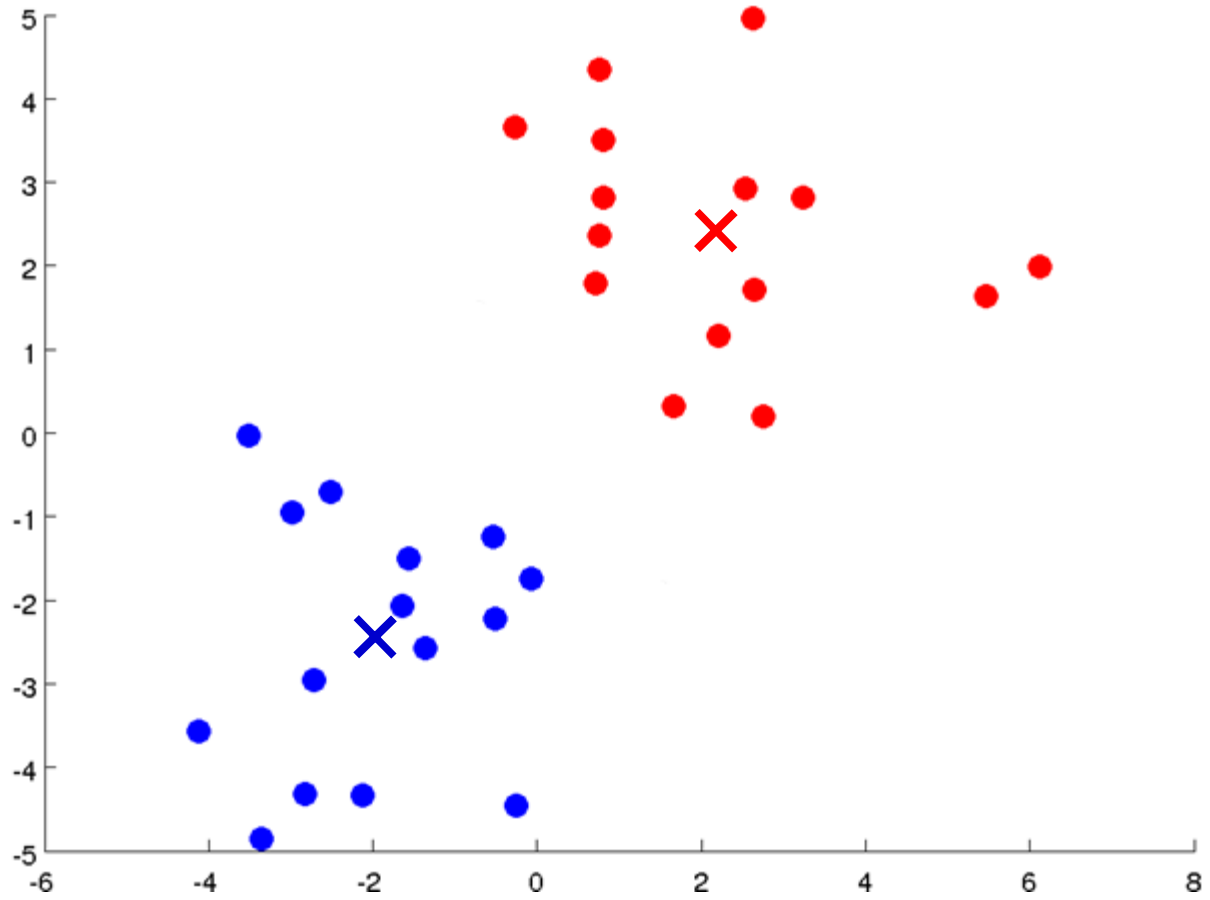

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

K-means algorithm

- 输入： (1) K 聚类簇的个数; (2) 训练集$\{x^{(1)}, x^{(2)}, \cdots, x^{(m)}\}$
- 首先随机指定K个类的中心(seed) $\mu_1, \mu_2, \cdots, \mu_K \in \mathbb{R}^n$, 然后迭代地更新该中心直到收敛:

  (1) for $i = 1$ to m

  用最近邻分类器进行分类：计算到$x^{(i)}$距离最近的聚类簇的中心,将其作为$x^{(i)}$的类别$y^{(i)} \in \{1, 2, \cdots, K\}$，即 $y^{(i)} = \arg\min_k \|x^{(i)} - \mu_k\|^2$

  (2) for $k = 1$ to K

  更新聚类簇的中心：用所有属于第k个簇的样本的均值去更新μ_k, 即$\mu_k = \mathtt{avg}(x^{(i)}|y^{(i)} = k)$

K-means algorithm

- 符号： $y^{(i)} \in \{1, 2, \cdots, K\}$表示样本$x^{(i)}$的“类别”; $\mu_k \in \mathbb{R}^n$表示第k个聚类簇的中心; $\mu_{y^{(i)}}$表示样本$x^{(i)}$对应的簇的中心
- K-means算法的目标函数为：

$$J(y^{(1)}, \cdots, y^{(m)}, \mu_1, \cdots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \| x^{(i)} - \mu_{y^{(i)}} \|^2$$

$$\min_{\substack{y^{(1)}, \cdots, y^{(m)} \\ \mu_1, \cdots, \mu_K}} J(y^{(1)}, \cdots, y^{(m)}, \mu_1, \cdots, \mu_K)$$

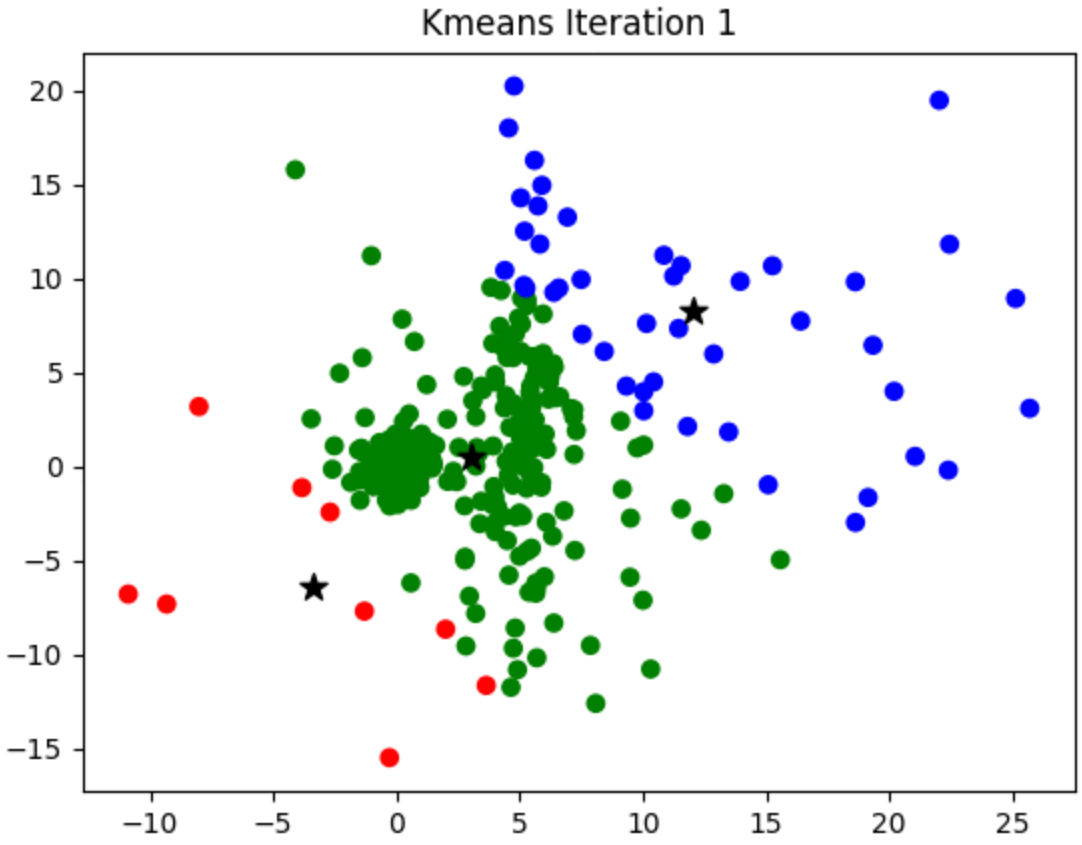
K-means algorithm

$$\min_{\substack{y^{(1)},\cdots,y^{(m)} \\ \mu_1,\cdots,\mu_K}} \frac{1}{m}\sum_{i=1}^{m}\|x^{(i)}-\mu_{y^{(i)}}\|^2$$

- 非凸函数优化问题，采用坐标下降法
- 若已知μ_k容易求$y^{(i)}$，反之亦然 (“鸡和蛋”问题)
- 属于Expectation Maximization (EM)算法的特例，能保证收敛到局部极小

K-means algorithm

- Time Complexity
  - 计算两个点之间的距离需要$O(n)$, 这里n是特征的维度
  - 最近邻分类时需要对所有m个样本计算到K个聚类簇中心的距离，即$O(Kmn)$
  - 更新聚类簇中心需要$O(mn)$
  - 假设需要l次迭代，则总的复杂度为$O(lKmn)$
- 初始化聚类簇中心
  - 最终的聚类结果依赖于初始化，不同的初始化可能得到不同的聚类结果
  - 随机初始化



https://sandipanweb.wordpress.com/2017/03/19/hard-soft-clustering-with-k-means-weighted-k-means-and-gmm-em/

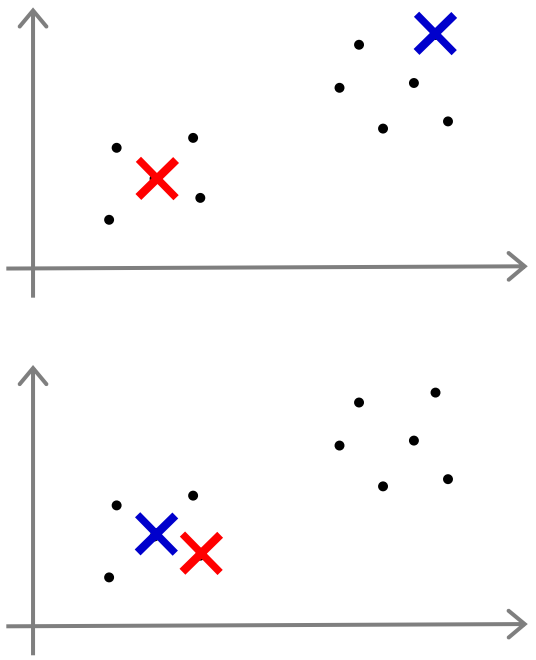
K-means algorithm

Random initialization

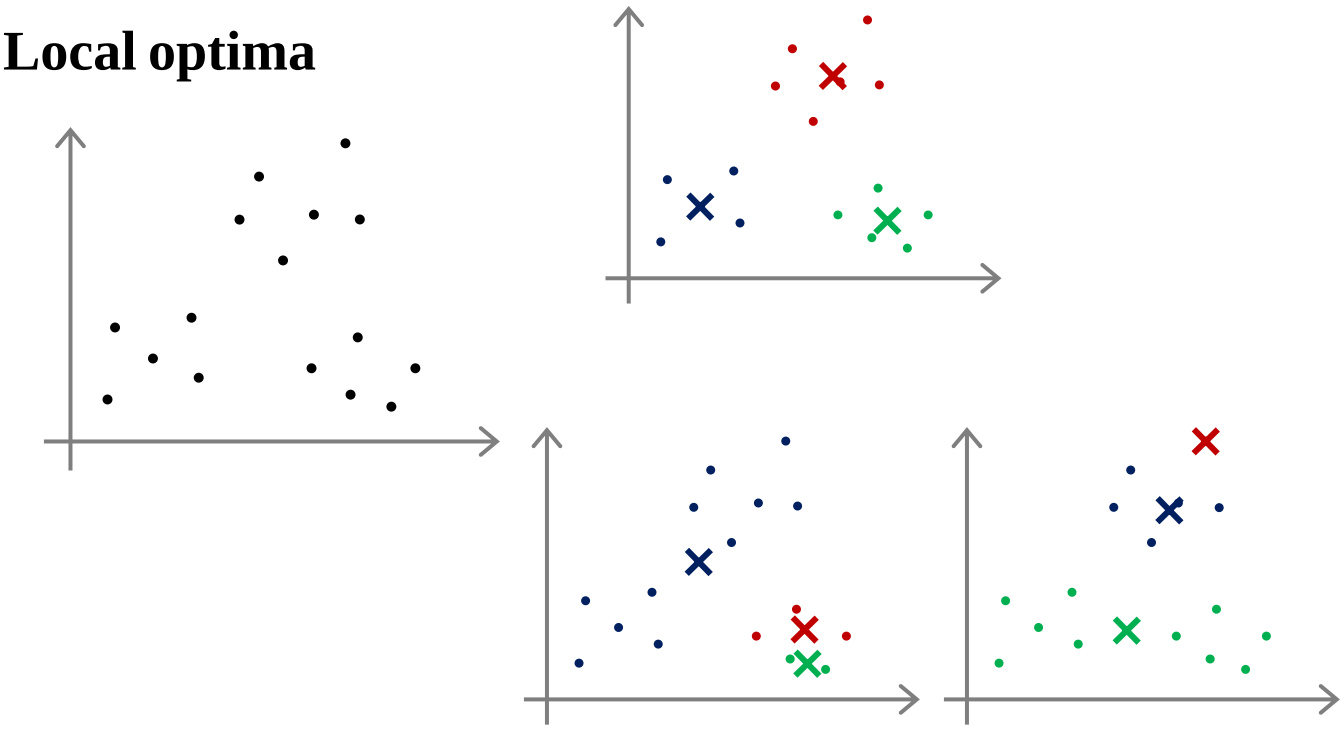
Should have $K < m$

Randomly pick K training examples.

Set $\mu_1, \ldots, \mu_K$ equal to these K examples.



K-means algorithm



K-means algorithm

Random initialization

For i = 1 to 100 {

Randomly initialize K-means.

Run K-means. Get $c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K$

Compute cost function (distortion)

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

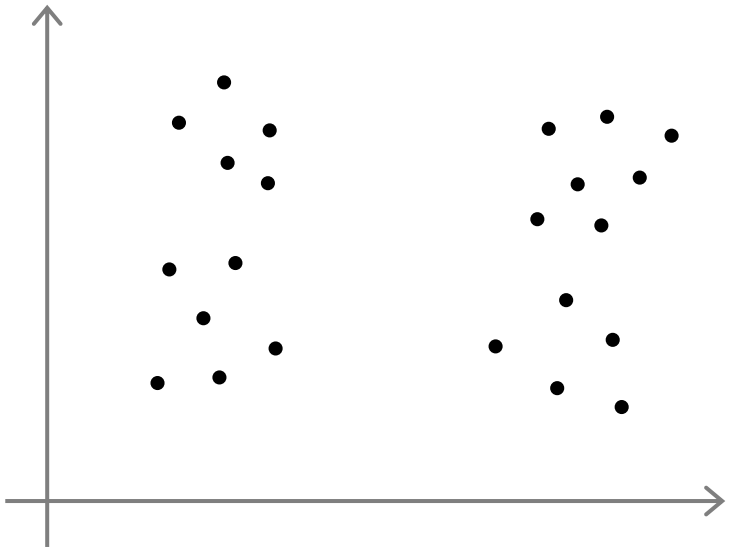
}

Pick clustering that gave lowest cost

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

K-means algorithm

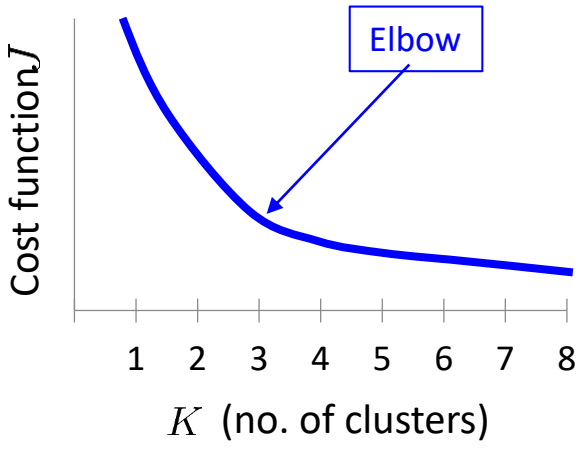
- 如何选择聚类个数K?

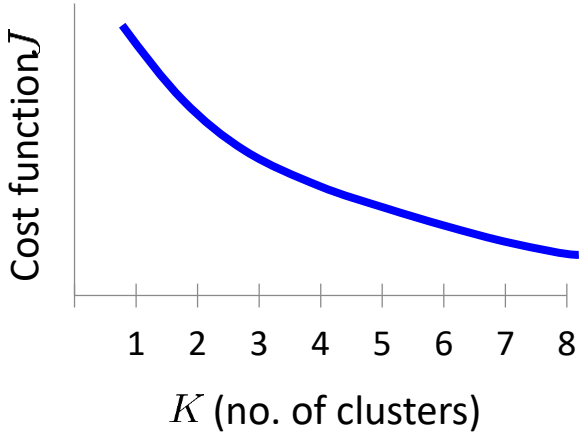


K-means algorithm

Choosing the value of K

Elbow method:



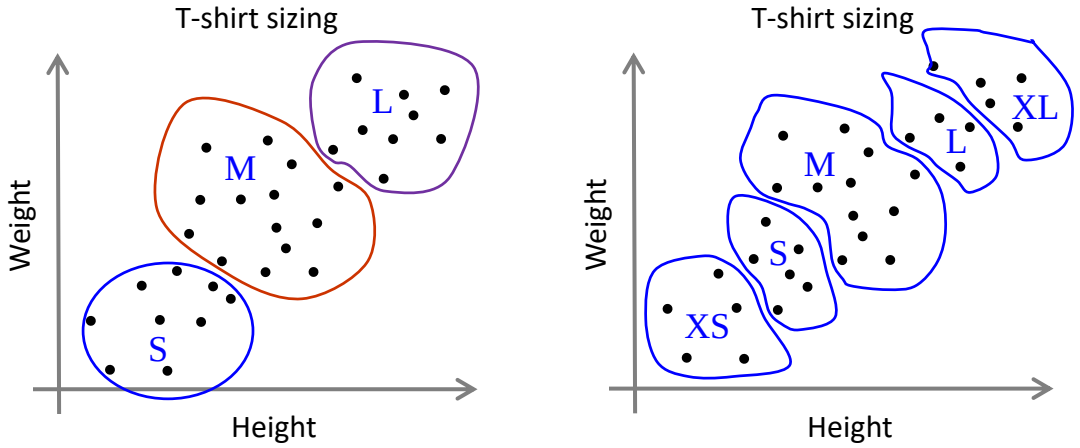


K-means algorithm

Choosing the value of K

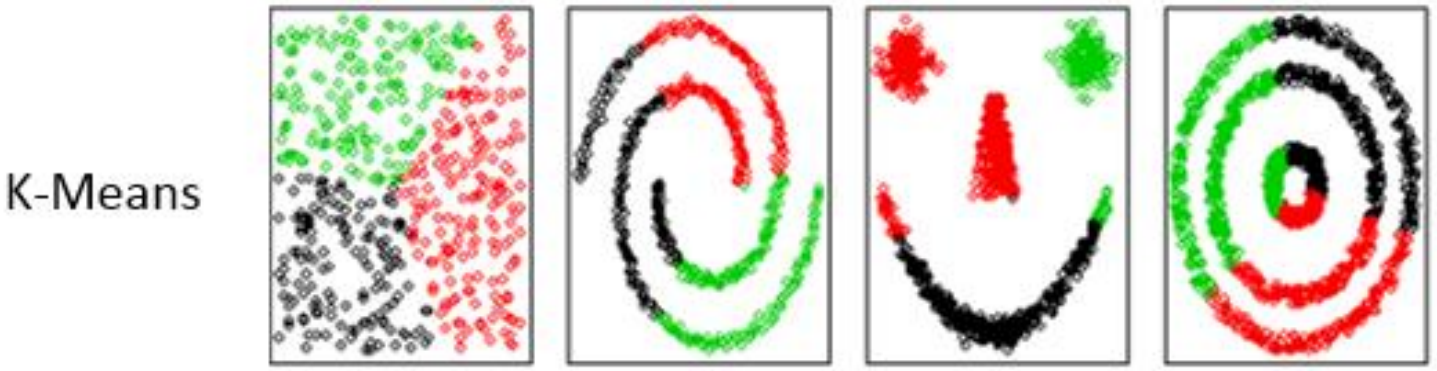
很多情况下*K*-means只是一个复杂任务的前处理步骤（如经典的Bag of Words分类方法），因此可能需要结合最终任务的性能来调整对应的*K*

E.g.



Kernel K-means

- K-means算法依赖于距离度量的选择
  - 常采用欧式距离
  - Distance Metric Learning
  - Kernel K-means: 通过隐式映射ϕ将数据变换到某个空间后，再用欧式距离度量



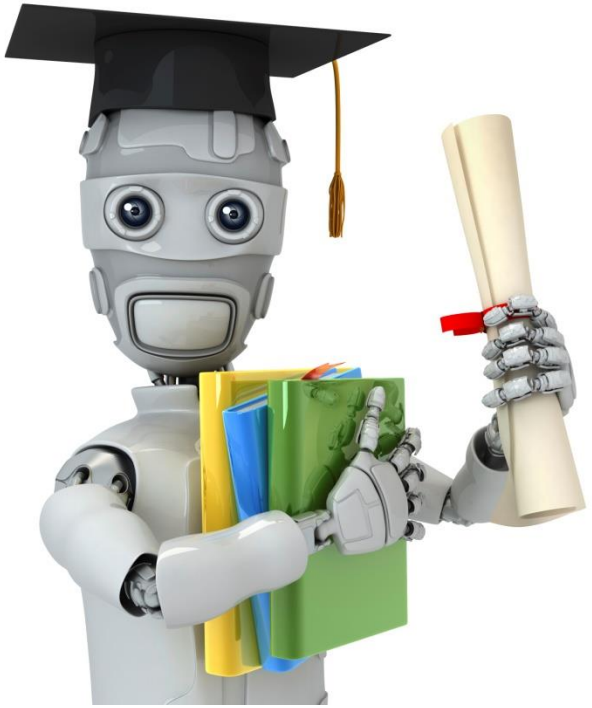
Kernel K-means

计算在变换空间中第k个聚类簇中心μ_k:

$$\mu_k = \frac{1}{\gamma_k} \sum_{y^{(i)}=k} \phi(x^{(i)}),$$

这里γ_k表示类别属于第k类的个数。考虑到在变换空间中样本到中心的欧式距离:

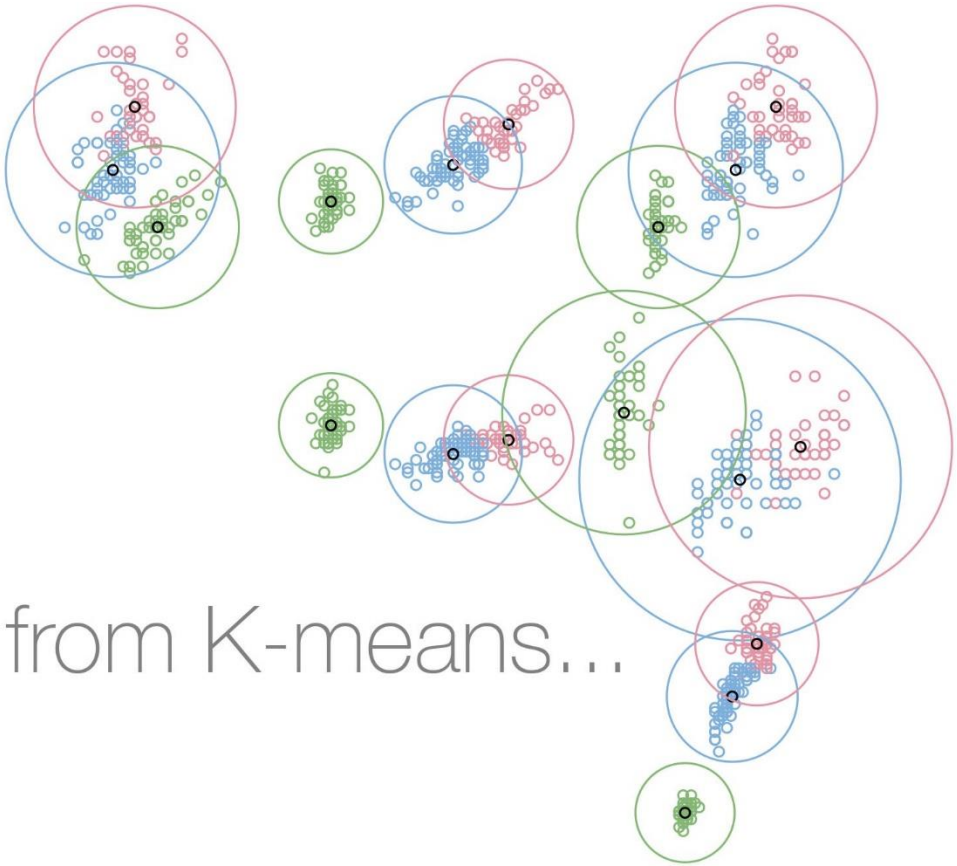
$$\begin{aligned}
\|\phi(x) - \mu_k\|^2 &= \|\phi(x) - \frac{1}{\gamma_k} \sum_{y^{(i)}=k} \phi(x^{(i)})\|^2 \\
&= \left(\phi(x) - \frac{1}{\gamma_k} \sum_{y^{(i)}=k} \phi(x^{(i)})\right)^T \left(\phi(x) - \frac{1}{\gamma_k} \sum_{y^{(i)}=k} \phi(x^{(i)})\right) \\
&= \phi(x)^T\phi(x) - 2\frac{1}{\gamma_k} \sum_{y^{(i)}=k} \phi(x)^T\phi(x^{(i)}) + \frac{1}{\gamma_k^2} \sum_{y^{(i)}=k} \sum_{y^{(j)}=k} \phi(x^{(i)})^T\phi(x^{(j)}) \\
&= k(x,x) - \frac{2}{\gamma_k} \sum_{y^{(i)}=k} k(x, x^{(i)}) + \frac{1}{\gamma_k^2} \sum_{y^{(i)}=k} \sum_{y^{(j)}=k} k(x^{(i)}, x^{(j)})
\end{aligned}$$



Machine Learning

Clustering

Gaussian Mixture Model, GMM



from K-means...

... to GMM

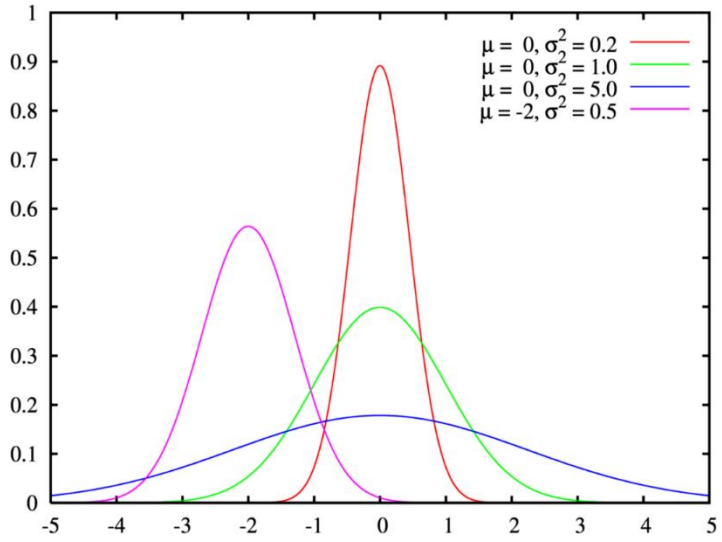
https://towardsdatascience.com/mixture-modelling-from-scratch-in-r-5ab7bfc83eef

Recap: Gaussian Distribution

- 一维高斯分布: 若$x \in \mathbb{R}$, $x \sim \mathcal{N}(\mu, \sigma^2)$, 则

$$p(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$$\begin{aligned} E(x) &= \mu \\ \mathtt{var}(x) &= \sigma_j^2 \end{aligned}$$



Recap: Gaussian Distribution

- 多维高斯分布: 若$x \in \mathbb{R}^n$, $x \sim \mathcal{N}(\mu, \Sigma)$, 则

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

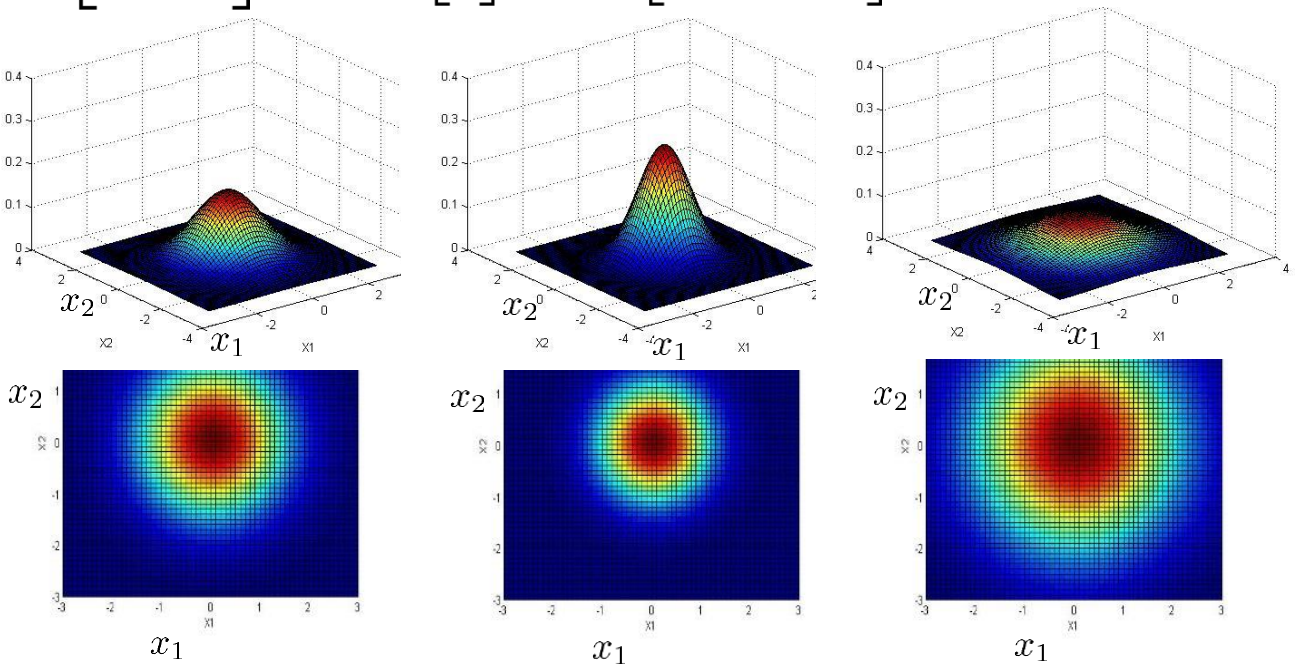
这里$|\cdot|$表示矩阵的行列式, $\Sigma \in \mathbb{R}^{n \times n}$称为covariance matrix.

$$\begin{aligned} E(x) &= \mu \\ \mathsf{cov}(x) &= \Sigma \end{aligned}$$

Recap: Gaussian Distribution

Multivariate Gaussian (Normal) examples

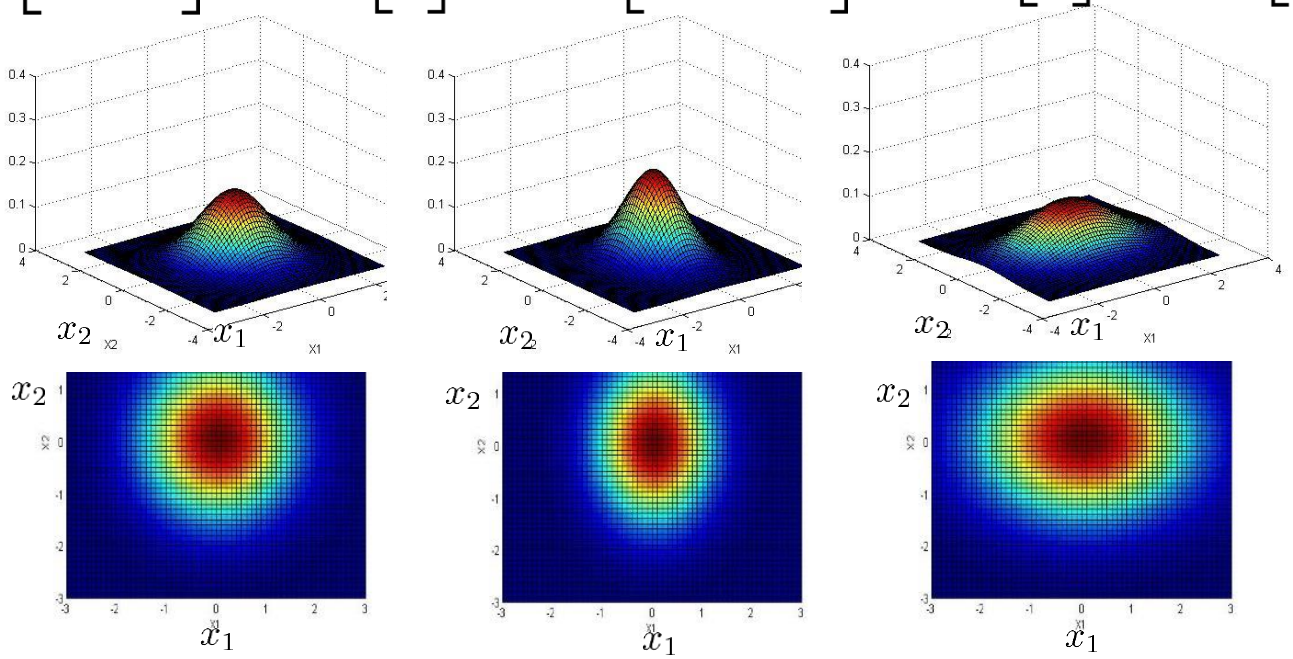
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 0.6 & 0 \\ 0 & 0.6 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$



Recap: Gaussian Distribution

Multivariate Gaussian (Normal) examples

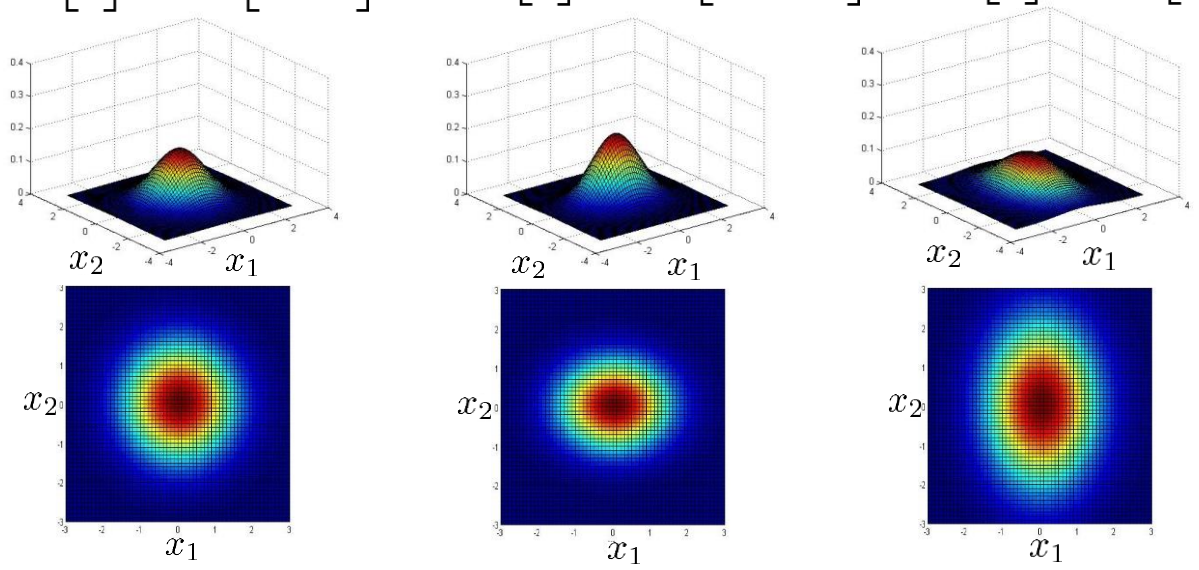
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 0.6 & 0 \\ 0 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$$



Recap: Gaussian Distribution

Multivariate Gaussian (Normal) examples

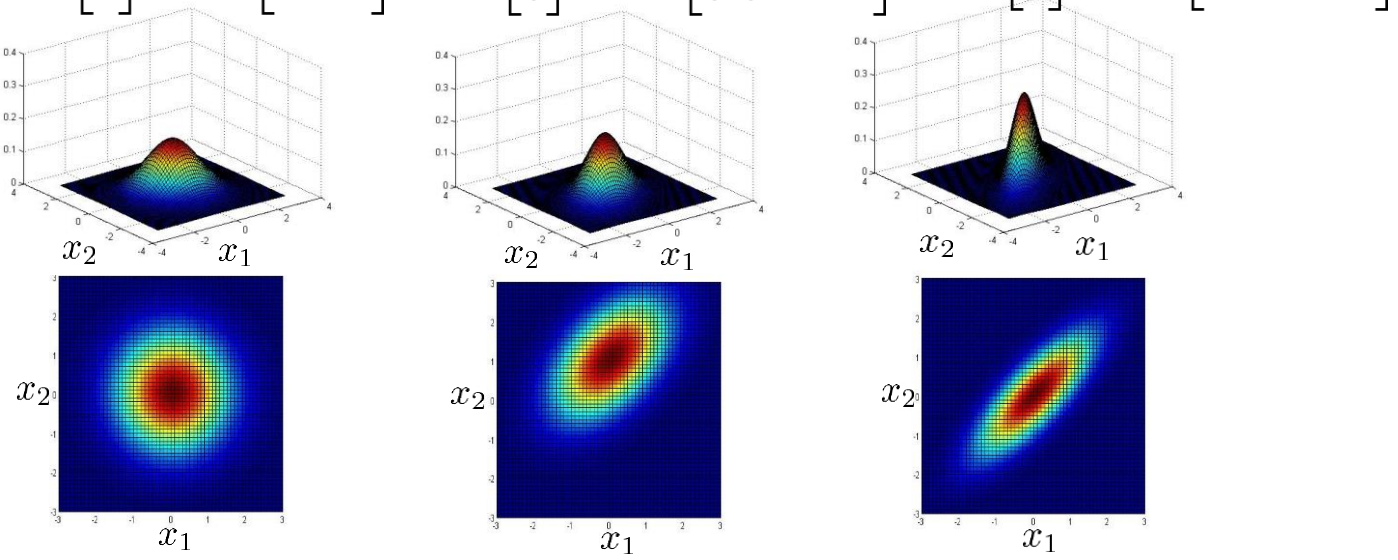
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 0.6 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$



Recap: Gaussian Distribution

Multivariate Gaussian (Normal) examples

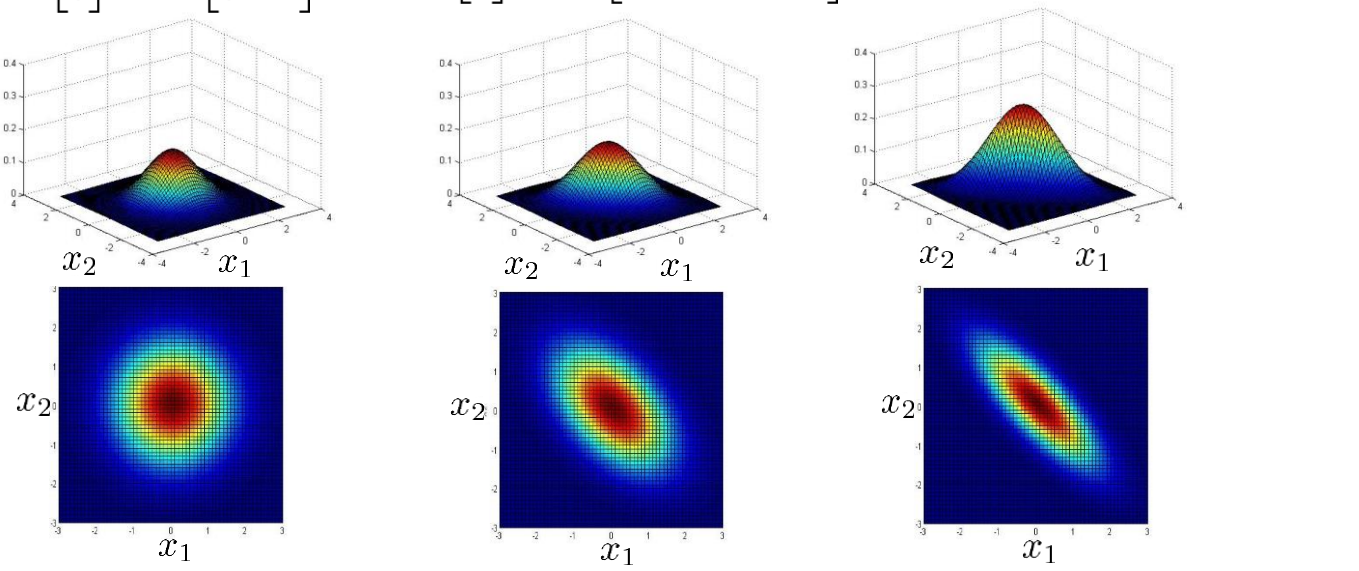
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$



Recap: Gaussian Distribution

Multivariate Gaussian (Normal) examples

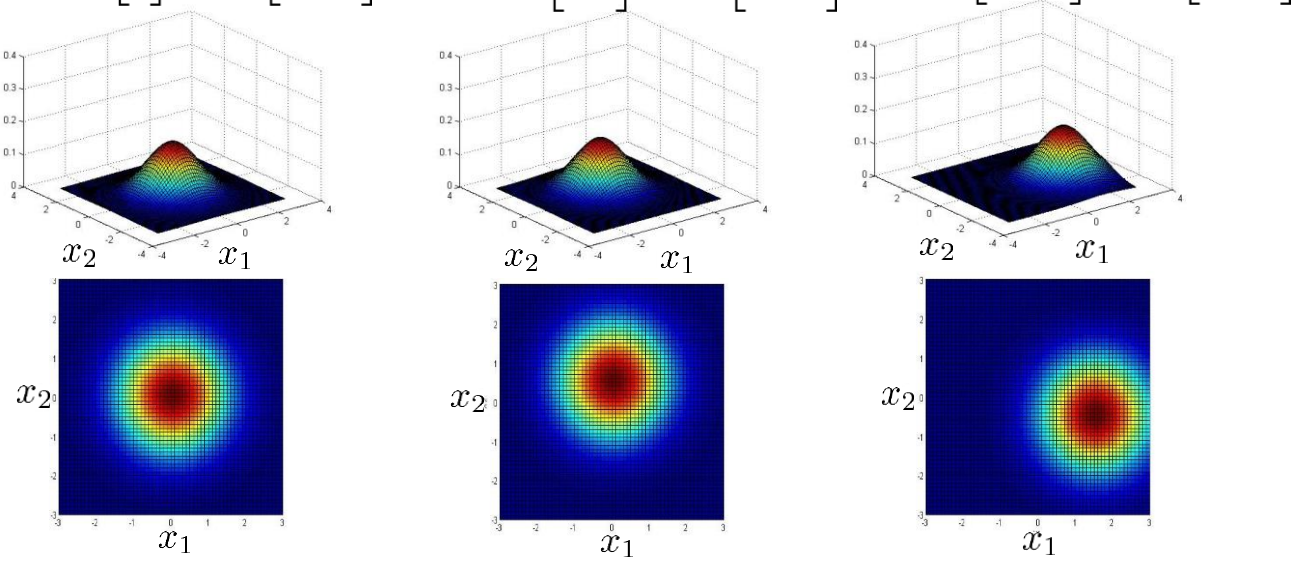
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 1 \end{bmatrix} \quad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & -0.8 \\ -0.8 & 1 \end{bmatrix}$$



Recap: Gaussian Distribution

Multivariate Gaussian (Normal) examples

$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $\mu = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $\mu = \begin{bmatrix} 1.5 \\ -0.5 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

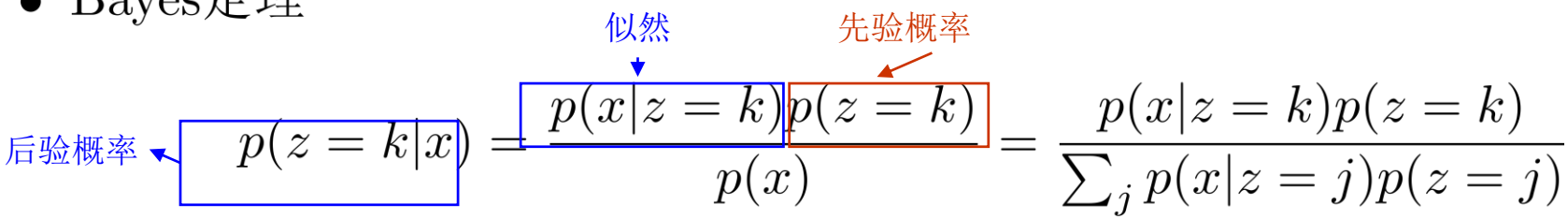


Bayes Decision

- 全概率公式

$$p(x) = \sum_{j=1}^{K} p(x|z=j)p(z=j)$$

- Bayes定理



Bayes Decision

$$p(x) = \sum_{j=1}^{K} p(x|z=j)p(z=j)$$
$$p(z=k|x) = \frac{p(x|z=k)p(z=k)}{p(x)} = \frac{p(x|z=k)p(z=k)}{\sum_j p(x|z=j)p(z=j)}$$

- 贝叶斯分类器

$$h(x) = \arg\max_k p(z=k|x) = \arg\max_k \frac{p(x|z=k)p(z=k)}{\sum_j p(x|z=j)p(z=j)}$$

- 高斯贝叶斯分类器

$$h(x) = \arg\max_k \frac{p(x;\mu_k,\Sigma_k)p(z=k)}{\sum_j p(x;\mu_j,\Sigma_j)p(z=j)}$$

$$p(x|z=j) = p(x;\mu_j,\Sigma_j) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_j|}} \exp\left(-\frac{1}{2}(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)\right)$$

Bayes Decision

- Bayes定理

$$p(z=k|x)=\frac{p(x|z=k)p(z=k)}{p(x)}=\frac{p(x|z=k)p(z=k)}{\sum_j p(x|z=j)p(z=j)}$$

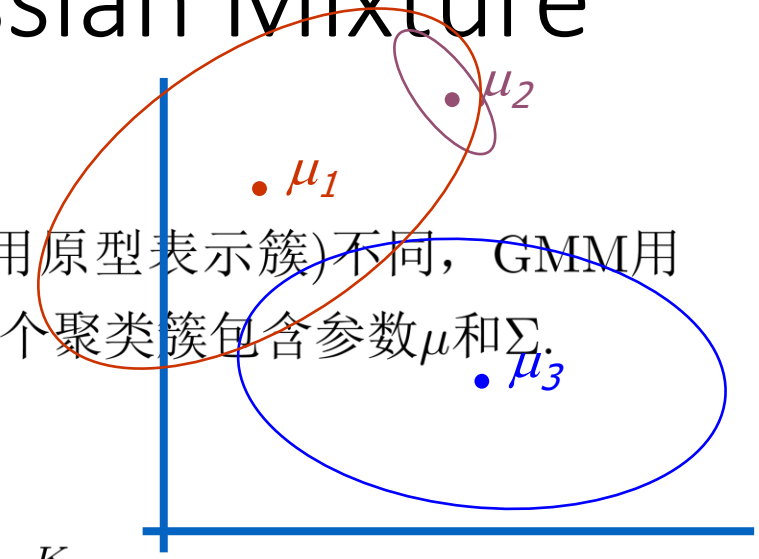
- 朴素贝叶斯分类器: 假设x的各个特征（属性）相互独立

$$p(z=k|x)=\frac{p(x|z=k)p(z=k)}{p(x)}=\frac{p(z=k)}{p(x)}\prod_{i=1}^{n}p(x_i|z=k)$$

对应当贝叶斯分类器可简化为

$$h(x)=\arg\max_k p(z=k)\prod_{i=1}^{n}p(x_i|z=k)$$

高斯混合模型 (Gaussian Mixture Model, GMM)



- 与K-means算法采用均值来表示聚类簇(即用原型表示簇)不同，GMM用一个高维高斯模型表示一个聚类簇，此时每个聚类簇包含参数μ和Σ.
- 可定义高斯混合模型如下：

$$p(x) = \sum_{k=1}^{K} p(z=k)p(x|z=k) = \sum_{k=1}^{K} \alpha_k p(x;\mu_k,\Sigma_k)$$

- μ_k和Σ_k分别为第k个聚类簇的参数，$\alpha_k = p(z=k)$为混合系数，表示属于第k个聚类簇的概率(由第k个高斯成分产生的概率)，显然有$\sum_k \alpha_k = 1$

$$p(x|z=k) = p(x;\mu_k,\Sigma_k) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}} \exp\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)$$

高斯混合模型 (Gaussian Mixture Model, GMM)

$$p(x|z=k)=p(x;\mu_k,\Sigma_k)=\frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}}\exp\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)$$

- 与K-means算法不同的是，这里并没有强制规定样本属于某个聚类簇，而是给出属于各个簇的概率α_k，因此可看成是“soft” K-means clustering
- 若已知所有的参数α_k,μ_k,Σ_k，根据贝叶斯定理，可计算出样本$x^{(j)}$由第k个高斯成分产生的后验概率

$$p(z^{(j)}=k|x^{(j)})=\frac{p(x^{(j)}|z^{(j)}=k)p(z^{(j)}=k)}{\sum_l p(x^{(j)}|z^{(j)}=l)p(z^{(j)}=l)}=\frac{\alpha_k p(x^{(j)};\mu_k,\Sigma_k)}{\sum_l \alpha_l p(x^{(j)};\mu_l,\Sigma_l)}$$

由贝叶斯分类器，可将样本的簇标记为具有最大后验概率对应的高斯成分。得到每个样本的簇标记，即完成聚类。问题的关键在于如何确定$\{\alpha_k,\mu_k,\Sigma_k\}_{k=1}^K$？

高斯混合模型 (Gaussian Mixture Model, GMM)

$$p(x) = \sum_{k=1}^{K} \alpha_k p(x; \mu_k, \Sigma_k)$$
$$p(x; \mu_j, \Sigma_j) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_j|}} \exp\left(-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)\right)$$

- 如何求$\{\alpha_k, \mu_k, \Sigma_k\}_{k=1}^{K}$？——最大化（对数）似然

$$LL = \log \prod_{j=1}^{m} p(x^{(j)}) = \sum_{j=1}^{m} \log \left(\sum_{k=1}^{K} \alpha_k p(x^{(j)}; \mu_k, \Sigma_k) \right)$$

令$\frac{\partial LL}{\partial \mu_k} = 0$, 有

$$\sum_{j=1}^{m} \frac{\alpha_k p(x^{(j)}; \mu_k, \Sigma_k)}{\sum_{l=1}^{K} \alpha_l p(x^{(j)}; \mu_l, \Sigma_l)} (x^{(j)} - \mu_k) = 0$$

$$\gamma_{jk} = p(z^{(j)} = k | x^{(j)}) = \frac{\alpha_k p(x^{(j)}; \mu_k, \Sigma_k)}{\sum_{l=1}^{K} \alpha_l p(x^{(j)}; \mu_l, \Sigma_l)},$$

$$\mu_k = \frac{\sum_{j=1}^{m} \gamma_{jk} x^{(j)}}{\sum_{j=1}^{m} \gamma_{jk}}$$

高斯混合模型 (Gaussian Mixture Model, GMM)

$$p(x) = \sum_{k=1}^{K} \alpha_k p(x; \mu_k, \Sigma_k)$$

$$p(x; \mu_j, \Sigma_j) = \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_j|}} \exp\left(-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)\right)$$

$$LL = \sum_{j=1}^{m} \log\left(\sum_{k=1}^{K} \alpha_k p(x^{(j)}; \mu_k, \Sigma_k)\right)$$

- 类似的，为了求Σ_k, 令$\frac{\partial LL}{\partial \Sigma_k} = 0$, 有

$$\Sigma_k = \frac{\sum_{j=1}^{m} \gamma_{jk}(x^{(j)} - \mu_k)(x^{(j)} - \mu_k)^T}{\sum_{j=1}^{m} \gamma_{jk}}$$

高斯混合模型 (Gaussian Mixture Model, GMM)

GMM算法中关于Σ的推导, 这里需要用到矩阵微分的两个性质：

1. 行列式的微分:

$$\frac{\partial|\Sigma|}{\partial\Sigma} = |\Sigma|(\Sigma^{-1})^T$$

2. 矩阵逆的微分:

$$\frac{\partial x^T\Sigma^{-1}y}{\partial\Sigma} = -\Sigma^{-T}xy^T\Sigma^{-T}$$

Petersen, Kaare Brandt, and Michael Syskind Pedersen. "The matrix cookbook." *Technical University of Denmark* 7, no. 15 (2008): 510.

高斯混合模型 (Gaussian Mixture Model, GMM)

$$
\begin{aligned}
\frac{\partial}{\partial \Sigma_k} p(x;\mu_k,\Sigma_k) &= \frac{\partial}{\partial \Sigma_k}\left[\frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}}\exp\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)\right] \\
&= \exp\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)\cdot \\
&\left[\frac{\partial}{\partial \Sigma_k}\left(\frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}}\right)+\frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}}\frac{\partial}{\partial \Sigma_k}\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)\right] \\
&= \frac{1}{(2\pi)^{n/2}}\exp\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)\cdot \\
&\left[-\frac{1}{2}|\Sigma_k|^{-\frac{3}{2}}|\Sigma_k|(\Sigma_k^{-T})+|\Sigma_k|^{-\frac{1}{2}}\left(\frac{1}{2}\Sigma_k^{-T}(x-\mu_k)(x-\mu_k)^T\Sigma_k^{-T}\right)\right] \\
&= \frac{1}{(2\pi)^{n/2}\sqrt{|\Sigma_k|}}\exp\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)\cdot \\
&\left[-\frac{1}{2}(\Sigma_k^{-T})+\left(\frac{1}{2}\Sigma_k^{-T}(x-\mu_k)(x-\mu_k)^T\Sigma_k^{-T}\right)\right] \\
&= p(x;\mu_k,\Sigma_k)\left[-\frac{1}{2}(\Sigma_k^{-T})+\left(\frac{1}{2}\Sigma_k^{-T}(x-\mu_k)(x-\mu_k)^T\Sigma_k^{-T}\right)\right]
\end{aligned}
$$

高斯混合模型 (Gaussian Mixture Model, GMM)

令$\frac{\partial LL}{\partial \Sigma_k} = 0$,

$$\begin{aligned}
\frac{\partial LL}{\partial \Sigma_k} &= \sum_{j=1}^{m} \frac{1}{\sum_{k=1}^{K} \alpha_k p(x^{(j)}; \mu_k, \Sigma_k)} \left(\frac{\partial}{\partial \Sigma_k} \alpha_k p(x^{(j)}; \mu_k, \Sigma_k) \right) \\
&= \sum_{j=1}^{m} \frac{\alpha_k p(x^{(j)}; \mu_k, \Sigma_k)}{\sum_{k=1}^{K} \alpha_k p(x^{(j)}; \mu_k, \Sigma_k)} \left(-\frac{1}{2}(\Sigma_k^{-T}) + \frac{1}{2}\Sigma_k^{-T}(x-\mu_k)(x-\mu_k)^T\Sigma_k^{-T} \right) \\
&= \sum_{j=1}^{m} \gamma_{jk} \left(-\frac{1}{2}(\Sigma_k^{-T}) + \frac{1}{2}\Sigma_k^{-T}(x-\mu_k)(x-\mu_k)^T\Sigma_k^{-T} \right) = 0
\end{aligned}$$

去掉常数，先左乘Σ_k再后乘Σ_k,并且注意到协方差矩阵的对称性$\Sigma_k^T = \Sigma_k$，有

$$\sum_{j=1}^{m} \gamma_{jk}(-\Sigma_k + (x-\mu_k)(x-\mu_k)^T) = 0$$

即

$$\Sigma_k = \frac{\sum_{j=1}^{m} \gamma_{jk}(x^{(j)}-\mu_k)(x^{(j)}-\mu_k)^T}{\sum_{j=1}^{m} \gamma_{jk}}$$

高斯混合模型 (Gaussian Mixture Model, GMM)

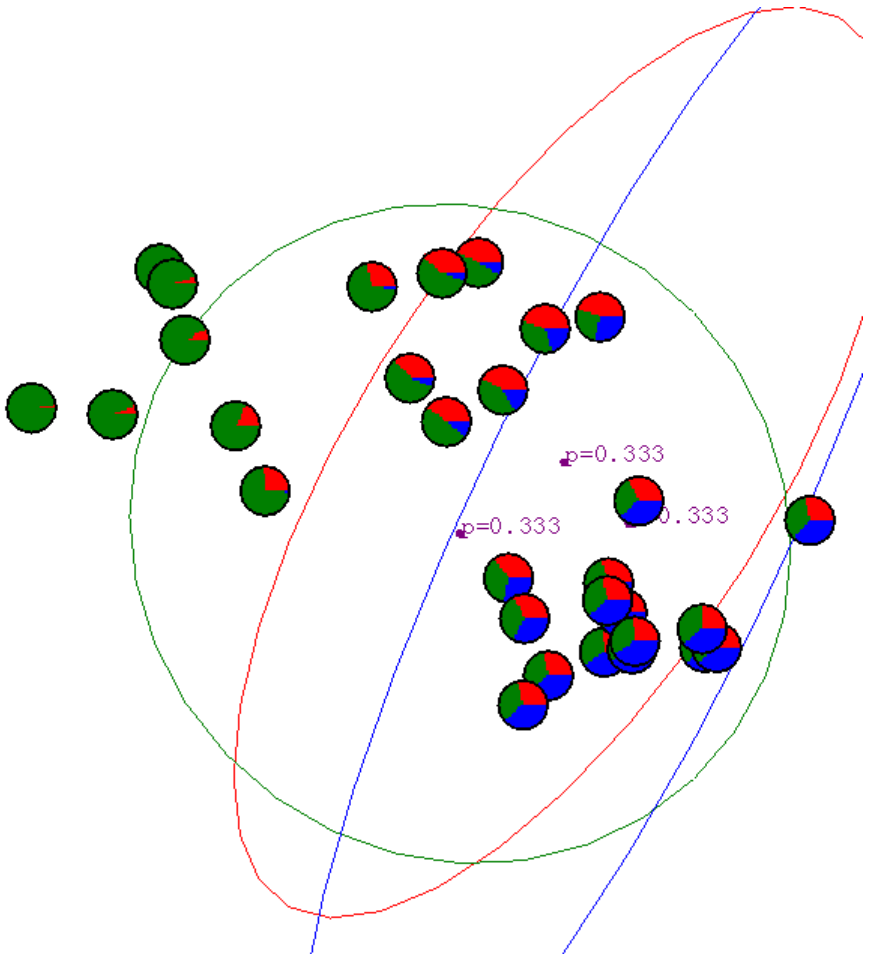
- 为了求先验α_k，注意到α_k除了要满足最大化对数似然外，还需要满足概率的性质：$\alpha_k \geq 0, \sum_k \alpha_k = 1$. 引入拉格朗日乘子λ，对应的拉格朗日函数为

$$LL + \lambda \left(\sum_{k=1}^{K} \alpha_k - 1 \right)$$

令其对α_k的导数为零，有(请自行推导)

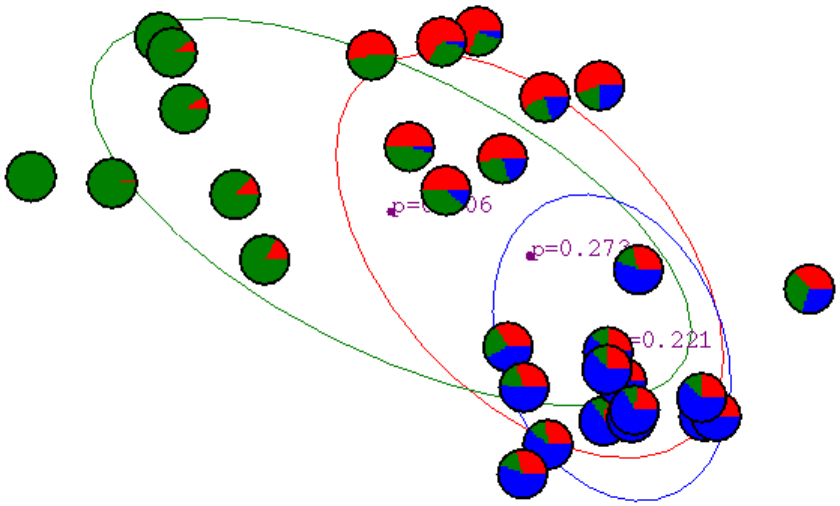
$$\alpha_k = \frac{1}{m} \sum_{j=1}^{m} \gamma_{jk}$$

GMM聚类



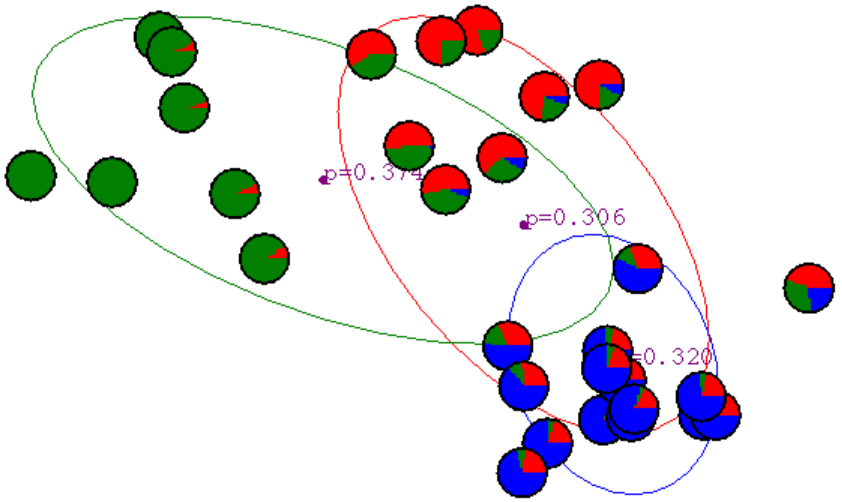
GMM聚类

After first iteration



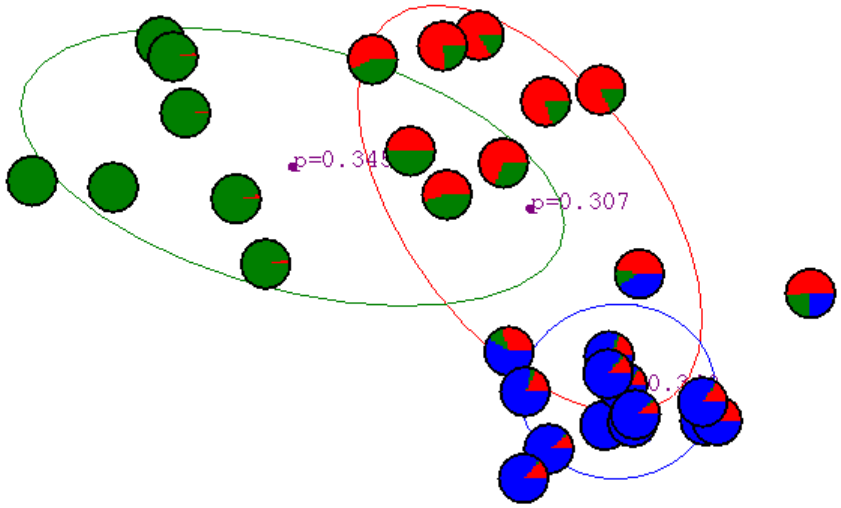
GMM聚类

After 2nd iteration



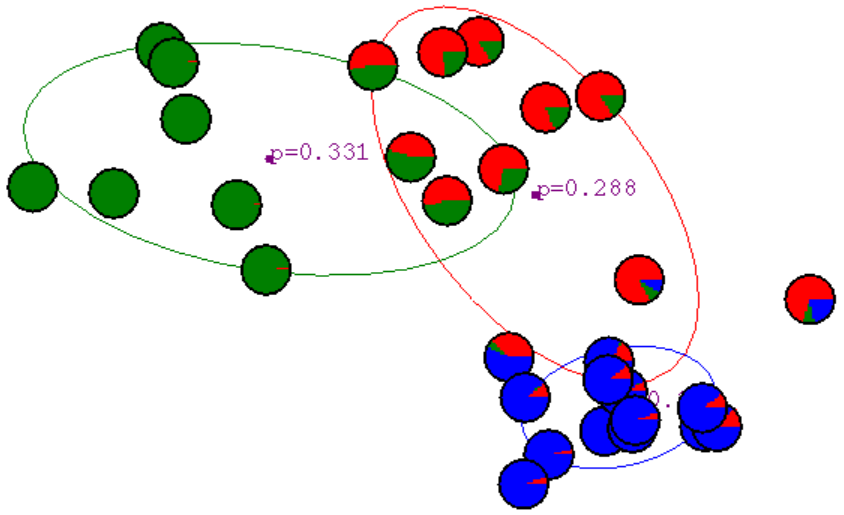
GMM聚类

After 3rd iteration



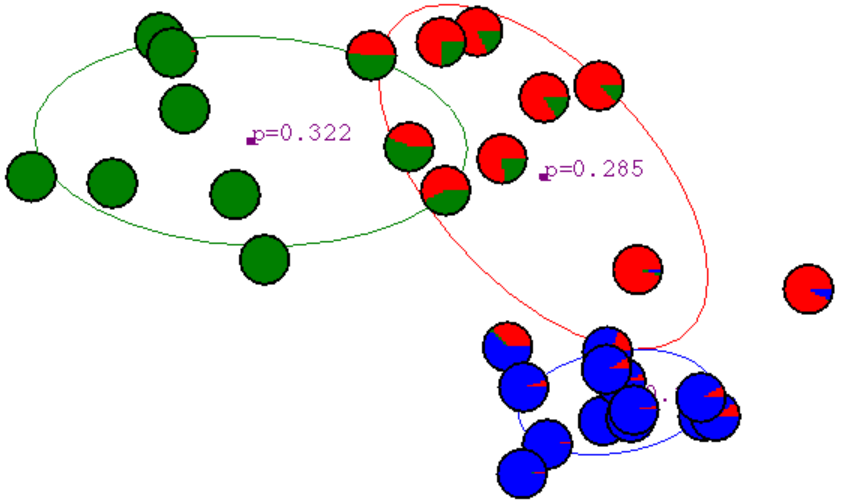
GMM聚类

After
4th
iteration



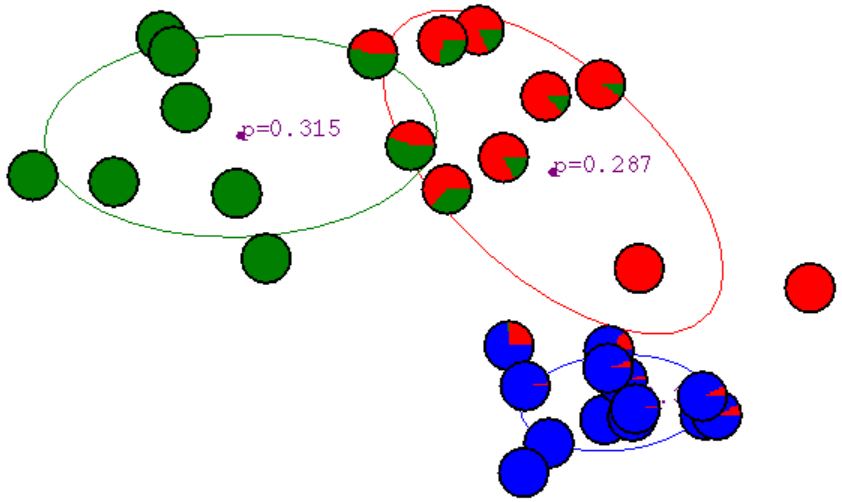
GMM聚类

After
5th
iteration



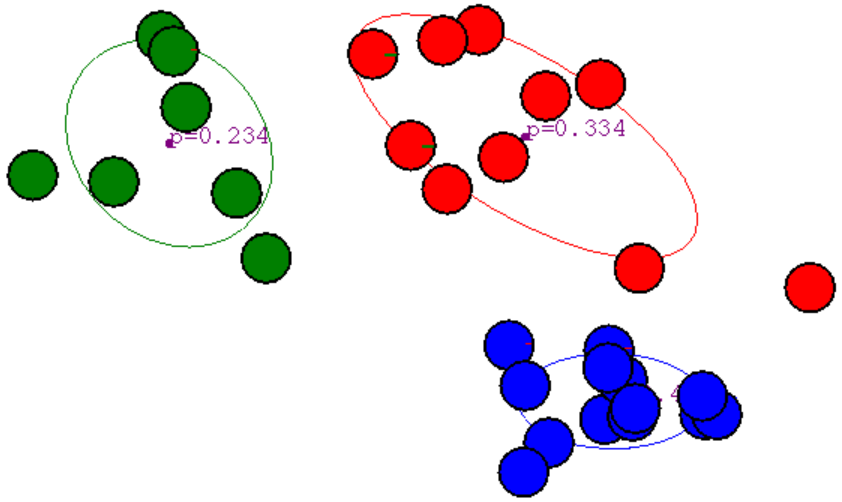
GMM聚类

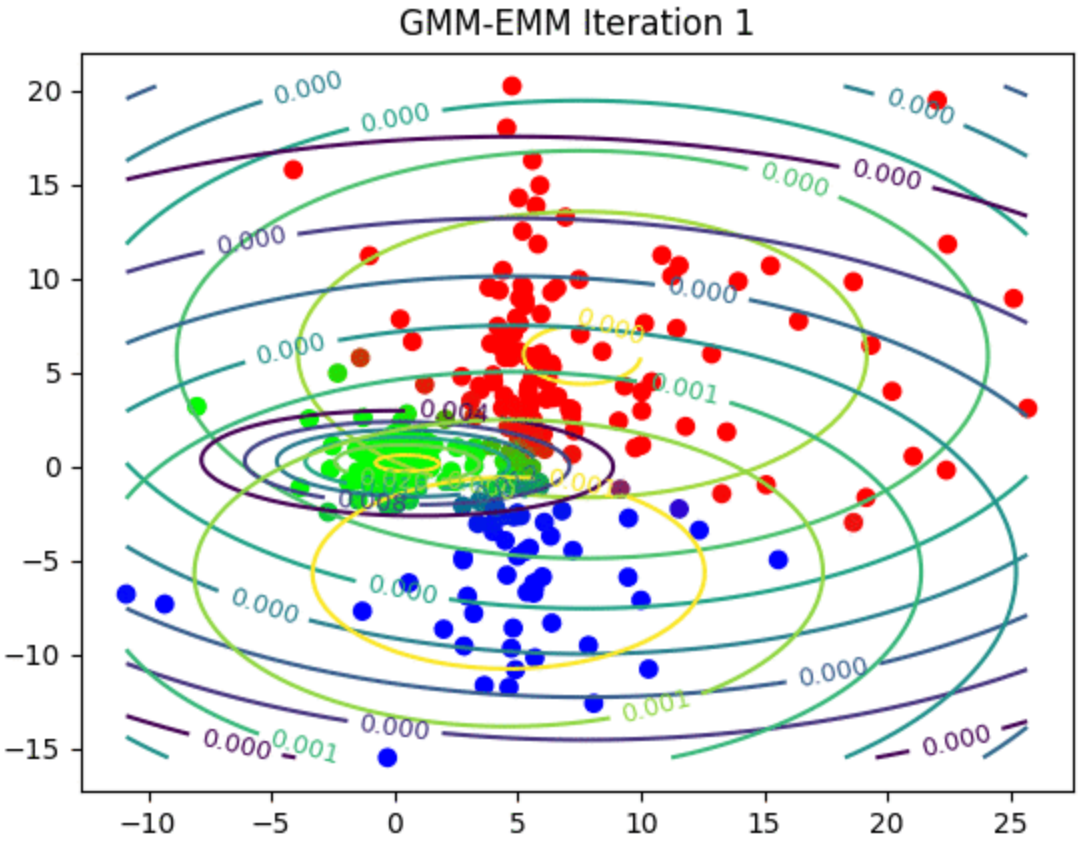
After
6th
iteration



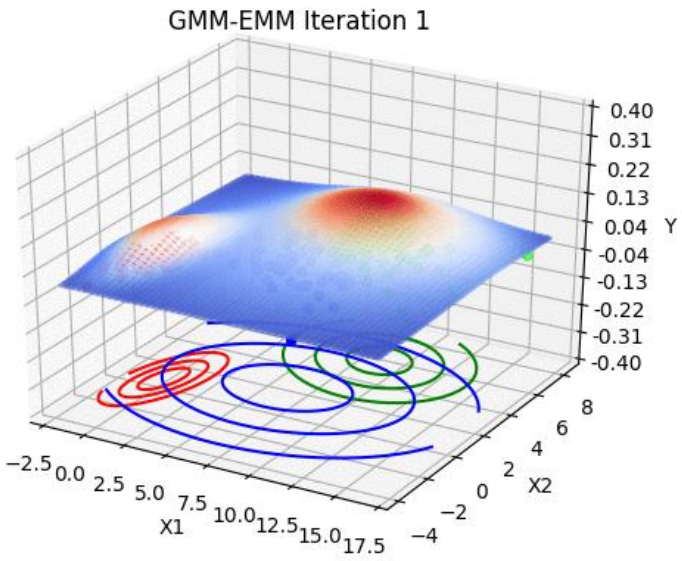
GMM聚类

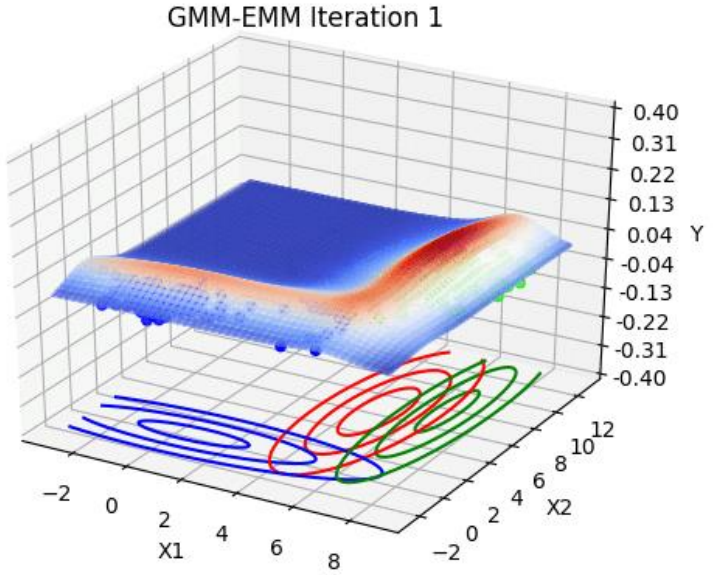
After
20th
iteration



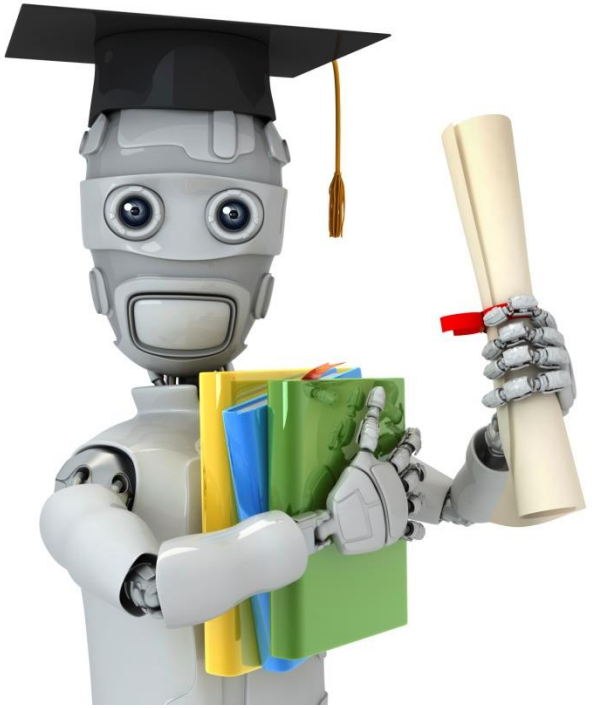


https://sandipanweb.wordpress.com/2017/03/19/hard-soft-clustering-with-k-means-weighted-k-means-and-gmm-em/





https://sandipanweb.wordpress.com/2017/03/19/hard-soft-clustering-with-k-means-weighted-k-means-and-gmm-em/



Machine Learning

Expectation Maximization (EM)

Recap: Convex Function

- 凸函数：若函数$f(x)$对任意的$t \in [0,1]$有

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2),$$

  - 等价于$f''(x) \geq 0,\ \forall x.$
  - 若x为矢量，则对应的条件变为Hessian矩阵H 为半正定矩阵$(H \geq 0)$

- Strictly convex: 若$\forall t \in (0,1),\ \forall x_1 \neq x_2$有

$$f(tx_1 + (1-t)x_2) < tf(x_1) + (1-t)f(x_2)$$

  对于矢量，则对应的条件变为Hessian矩阵H正定。

Aside: Jensen's Inequality

Theorem. Let f be a convex function, and let X be a random variable. Then:

$$E[f(X)] \geq f(E(X)).$$

Moreover, if f is strictly convex, then $E[f(X)] = f(E(X))$ holds true if and only if $X = E[X]$ with probability 1 (i.e., if X is a constant).

Remark. Recall that f is [strictly] concave if and only if $-f$ is [strictly] convex (i.e., $f''(x) \leq 0$ or $H \leq 0$). Jensen's inequality also holds for concave functions f, but with the direction of all the inequalities reversed ($E[f(X)] \leq f(E(X))$, etc.).

期望最大化(EM)

给定包含m互相独立的训练样本集$\{x^{(1)}, \ldots, x^{(m)}\}$，期望找到模型$p(x, z; \theta)$的最优参数，其中$z$为未知的隐变量。其对数似然函数为

$$\begin{aligned} \ell(\theta) &= \sum_{i=1}^{m} \log p(x^{(i)}; \theta) \\ &= \sum_{i=1}^{m} \log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)}; \theta). \end{aligned}$$

若已知$z^{(i)}$，则容易采用最大似然估计进行求解。但由于隐变量$z^{(i)}$未知，无法直接进行优化。 Our strategy will be to instead repeatedly construct a lower-bound on ℓ (E-step), and then optimize that lower-bound (M-step).

期望最大化(EM)

For each i, let Q_i be some distribution over the z's ($\sum_z Q_i(z) = 1$, $Q_i(z) \geq 0$). Consider the following:

$$\begin{aligned}
\sum_i \log p(x^{(i)};\theta) &= \sum_i \log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)};\theta) \\
&= \sum_i \log \sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} \\
&\geq \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}
\end{aligned}$$

类比Jensen不等式, $\log(\cdot)$看成是凸函数f(因为$f''(x) = -\frac{1}{x^2} \leq 0$), Q是一个分布，则$\sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}$可看成期望$E(X)$, 这里的$X = \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}$, 而$\sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}$则可看成是$E(f(X))$，即

$$f\left(E_{z^{(i)}\sim Q_i}\left[\frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}\right]\right) \geq E_{z^{(i)}\sim Q_i}\left[f\left(\frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}\right)\right].$$

期望最大化(EM)

$$\ell(\theta) = \sum_i \log p(x^{(i)};\theta) \geq \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}$$

即右边是$\ell(\theta)$的下界。由于直接求$\ell(\theta)$存在困难，转而通过求下界以逼近$\ell(\theta)$(当不等式变成等式时，完美逼近)。若θ已知，则其值仅依赖于$Q_i(z^{(i)})$和$p(x^{(i)}, z^{(i)})$, 因此可以通过调整两者使得等式成立。根据Jensen不等式中等号成立的条件是X为常量，

$$\frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} = c.$$

即

$$Q_i(z^{(i)}) \propto p(x^{(i)}, z^{(i)};\theta).$$

因为$\sum_z Q_i(z) = 1$, 有

$$Q_i(z^{(i)}) = \frac{p(x^{(i)}, z^{(i)};\theta)}{\sum_z p(x^{(i)}, z;\theta)} = \frac{p(x^{(i)}, z^{(i)};\theta)}{p(x^{(i)};\theta)} = p(z^{(i)}|x^{(i)};\theta)$$

期望最大化(EM)

$$Q_i(z^{(i)}) = p(z^{(i)}|x^{(i)};\theta)$$

至此在给定θ的情况下，我们解决了如何选择$Q_i(z^{(i)})$来求似然函数$\ell(\theta)$的下界。这就是E-step。M-step就是去找到最优的θ去最大化似然函数$\ell(\theta)$的下界。如此反复直到收敛为止，即

Repeat until convergence {

(E-step) For each i, set

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta).$$

(M-step) Set

$$\theta := \arg\max_\theta \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}.$$

}

GMM Revisited

Repeat until convergence {

(E-step) For each i, set

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta).$$

(M-step) Set

$$\theta := \arg\max_\theta \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}.$$

}

对于GMM而言，

- 参数为$\theta = (\alpha, \mu, \Sigma) = \{\alpha_k, \mu_k, \Sigma_k\}_{k=1}^K$
- 隐变量$z \in \{1, 2, \cdots, K\}$表示生成样本$x$的高斯混合成分，即 $p(z = k) = \alpha_k$

这种情况下E-step可简单由下式表示

$$\gamma_{ik} = Q_i(z^{(i)} = k) = p(z^{(i)} = k|x^{(i)};\theta) = p(z^{(i)} = k|x^{(i)};\alpha, \mu, \Sigma),$$

here, "$Q_i(z^{(i)} = k)$" denotes the probability of $z^{(i)}$ taking the value k under the distribution Q_i.

GMM Revisited

Repeat until convergence {

(E-step) For each i, set

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta).$$

(M-step) Set

$$\theta := \arg\max_\theta \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}.$$

}

在M-step, 我们需要找到最优的 $\theta = (\alpha, \mu, \Sigma)$使得下式取得最大值

$$\sum_{i=1}^{m} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \alpha, \mu, \Sigma)}{Q_i(z^{(i)})}$$

$$= \sum_{i=1}^{m} \sum_{k=1}^{K} Q_i(z^{(i)} = k) \log \frac{p(x^{(i)}|z^{(i)} = k; \mu, \Sigma) p(z^{(i)} = k; \alpha)}{Q_i(z^{(i)} = k)}$$

$$= \sum_{i=1}^{m} \sum_{k=1}^{K} \gamma_{ik} \log \frac{\frac{1}{(2\pi)^{n/2}|\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_k)^T \Sigma_k^{-1}(x^{(i)} - \mu_k)\right) \cdot \alpha_k}{\gamma_{ik}}$$

GMM Revisited

Repeat until convergence {

(E-step) For each i, set

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta).$$

(M-step) Set

$$\theta := \arg\max_\theta \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}.$$

}

首先求μ。令上式对$\mu_l (l = 1, 2, \cdots, K)$的偏微分为零，有

$$\begin{aligned}
&\nabla_{\mu_l} \sum_{i=1}^{m} \sum_{k=1}^{K} \gamma_{ik} \log \frac{\frac{1}{(2\pi)^{n/2}|\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_k)^T \Sigma_k^{-1}(x^{(i)} - \mu_k)\right) \cdot \alpha_k}{\gamma_{ik}} \\
= \quad & -\nabla_{\mu_l} \sum_{i=1}^{m} \sum_{j=1}^{k} \gamma_{ik} \frac{1}{2}(x^{(i)} - \mu_k)^T \Sigma_k^{-1}(x^{(i)} - \mu_k) \\
= \quad & \frac{1}{2} \sum_{i=1}^{m} \gamma_{il} \nabla_{\mu_l} 2\mu_l^T \Sigma_l^{-1} x^{(i)} - \mu_l^T \Sigma_l^{-1} \mu_l \\
= \quad & \sum_{i=1}^{m} \gamma_{il} \left(\Sigma_l^{-1} x^{(i)} - \Sigma_l^{-1} \mu_l\right) = 0
\end{aligned}$$

因此μ_l更新规则为

$$\mu_l := \frac{\sum_{i=1}^{m} \gamma_{il} x^{(i)}}{\sum_{i=1}^{m} \gamma_{il}},$$

GMM Revisited

Repeat until convergence {

(E-step) For each i, set

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta).$$

(M-step) Set

$$\theta := \arg\max_\theta \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}.$$

}

再考虑求α。目标函数中与α有关的部分为

$$\sum_{i=1}^{m}\sum_{k=1}^{K}\gamma_{ik}\log\alpha_k,$$

注意到概率的约束$\sum_k \alpha_k = 1$ (按道理这里还应该加上约束$\alpha_k \geq 0, \forall k$, 但最优解的结果直接满足该条件，因此可以省略)，因此对应的拉格朗日函数为

$$\mathcal{L}(\alpha) = \sum_{i=1}^{m}\sum_{k=1}^{K}\gamma_{ik}\log\alpha_k + \beta(\sum_{k=1}^{K}\alpha_k - 1),$$

令导数为零有

$$\frac{\partial}{\partial\alpha_k}\mathcal{L}(\alpha) = \sum_{i=1}^{m}\frac{\gamma_{ik}}{\alpha_k} + \beta = 0$$

有$\alpha_k = -\frac{1}{\beta}\sum_{i=1}^{m}\gamma_{ik}$. 再利用$\sum_{k=1}^{K}\alpha_k = 1$, 可以得到$-\beta = \sum_{i=1}^{m}\sum_{k=1}^{K}\gamma_{ik} = \sum_{i=1}^{m}1 = m$. 因此有

$$\alpha_k := \frac{1}{m}\sum_{i=1}^{m}\gamma_{ik}.$$

关于Σ的推导请自行求解。

Thanks!

Any questions?