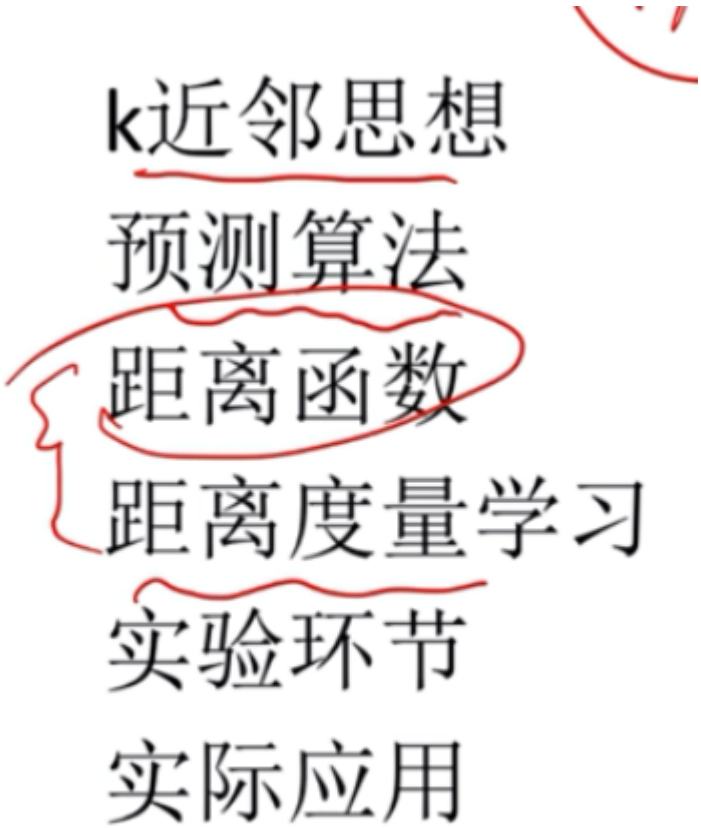
7 K 临近思想

重点：算法思想

7.1 本章学习简介

KNN

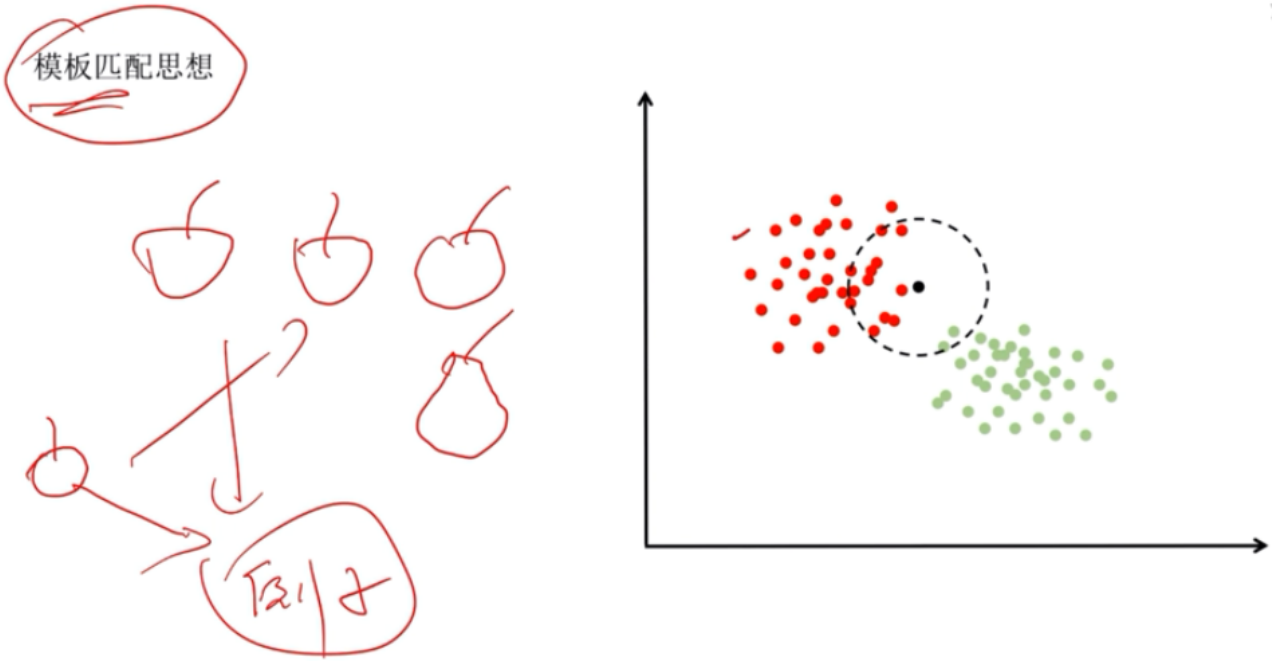


7.2 K近邻算法

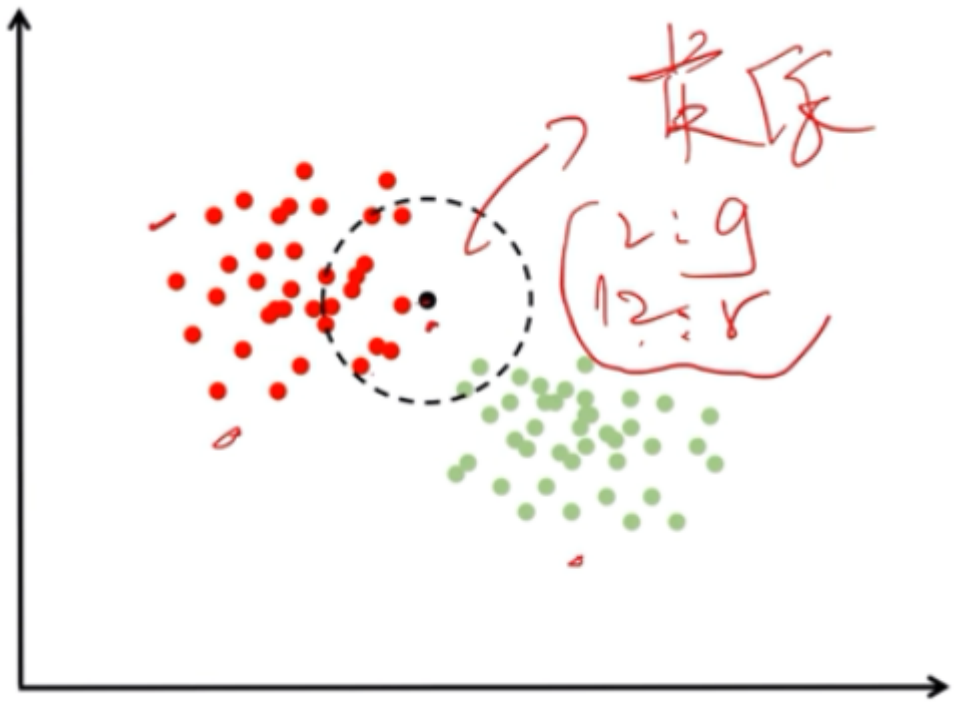
它基于模板匹配的思想

例如，确定一个水果类别，拿样本和标准模板去比。

例如拿一樱桃去和一堆水果比较，看他和哪个更像，就属于那个水果类别。



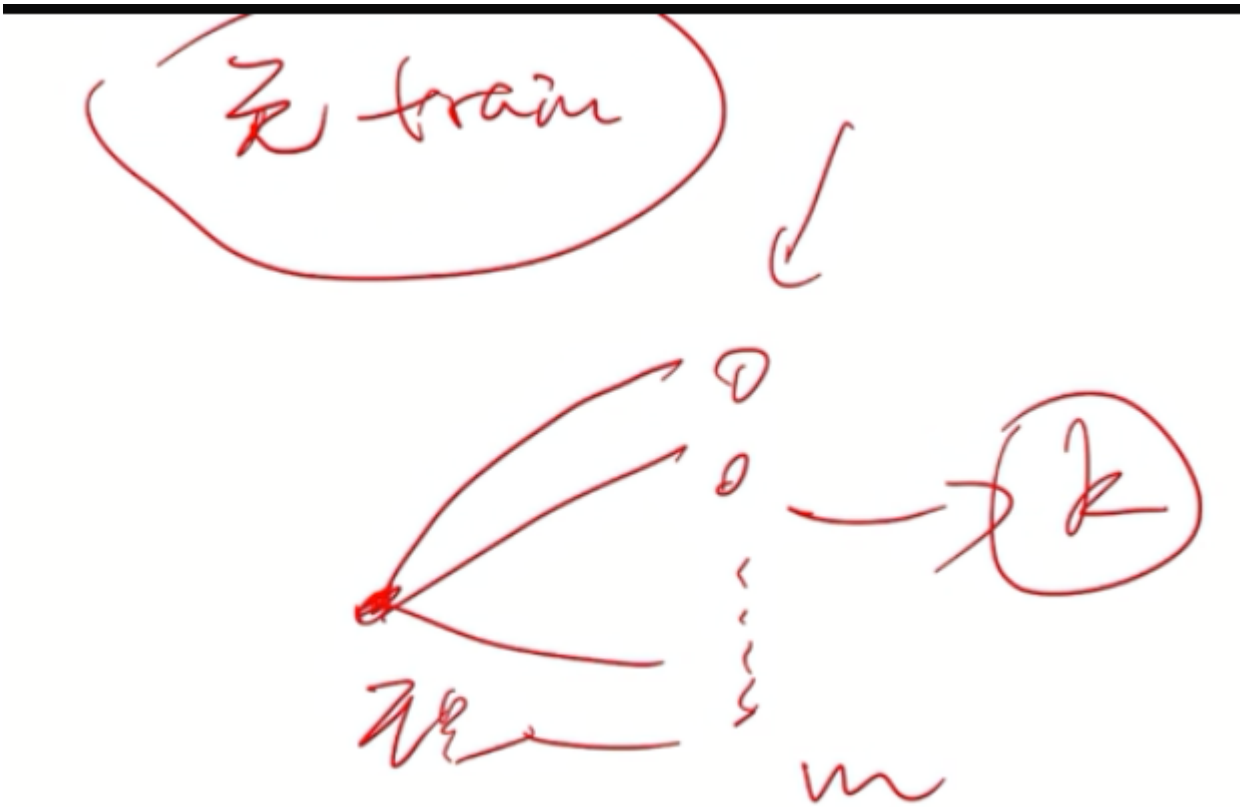
例如对图中黑色的点进行分类，在他周围画一个圈。统计和距离最近的样本，统计出红色的样本占据12个，绿色点占2个，所以黑色点属于红色的类。



小结：该算法，做分类。

7.3 预测算法

KNN没有训练过程,现去算，找出近邻的k个样本。

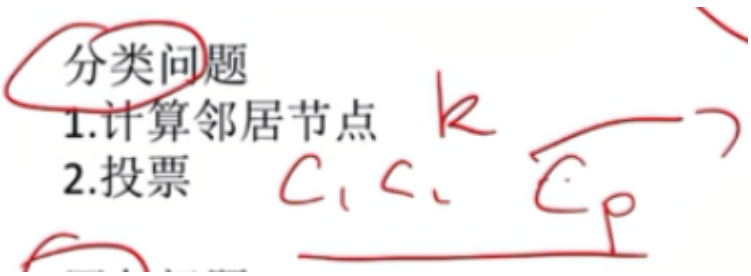


算法用途

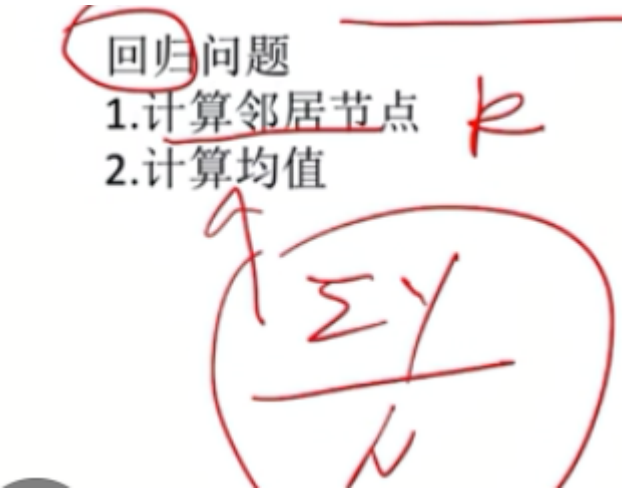
预测算法

分类问题
1.计算邻居节点
2.投票

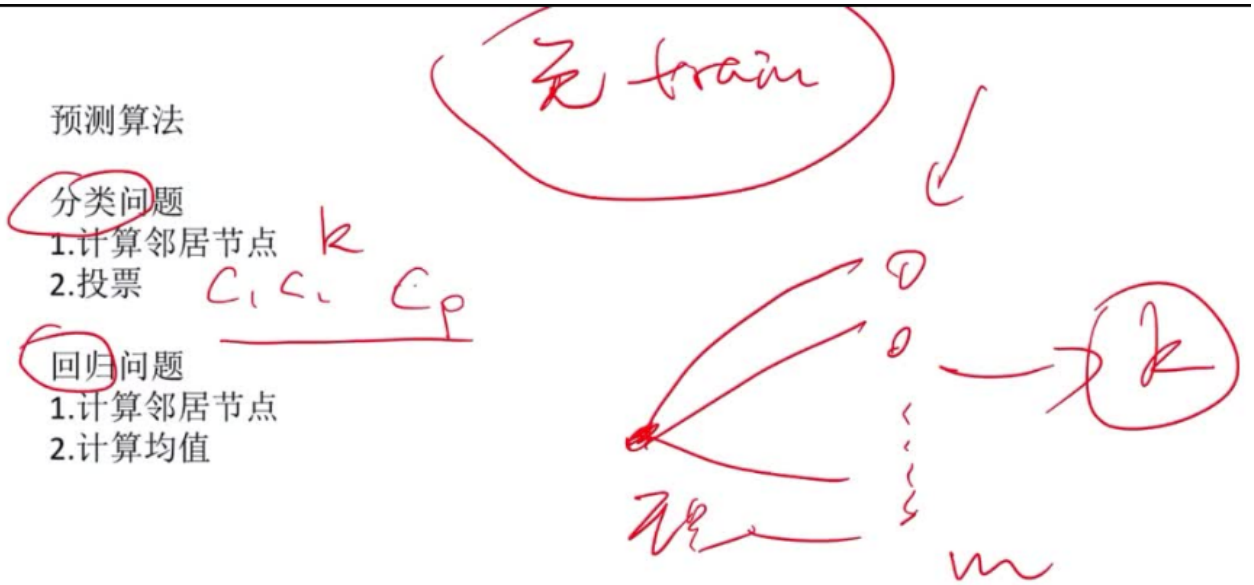
回归问题
1.计算邻居节点
2.计算均值



1. 先计算邻居节点



一般对于二分类问题，K设置为奇数。避免投票的时候，出现两派票数相等的情况。li七大常委是7个，奇数个。



7.4 距离函数

距离函数依赖于相似度，距离函数需要满足如下三个数学条件

距离函数

距离必须满足的数学条件

三角不等式： $d(x_i,x_k)+d(x_k,x_j)\geq d(x_i,x_j)$

非负性： $d(x_i,x_j)\geq 0$

对称性： $d(x_i,x_j)=d(x_j,x_i)$

距离必须满足的数学条件

三角不等式： $d(x_i,x_k)+d(x_k,x_j)\geq d(x_i,x_j)$

非负性： $d(x_i,x_j)\geq 0$

对称性： $d(x_i,x_j)=d(x_j,x_i)$

常用距离的定义

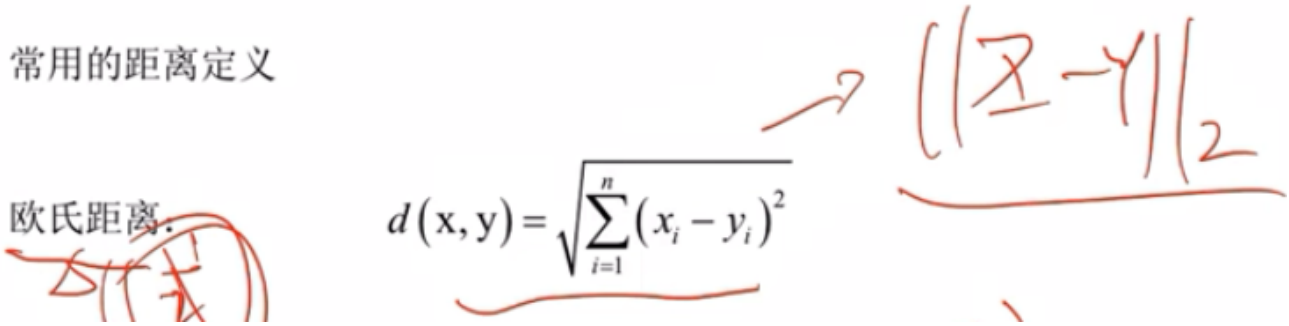
常用的距离定义

欧氏距离：$$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$$

Mahalanobis：$$d(x,y)=\sqrt{(x-y)^T S(x-y)}$$

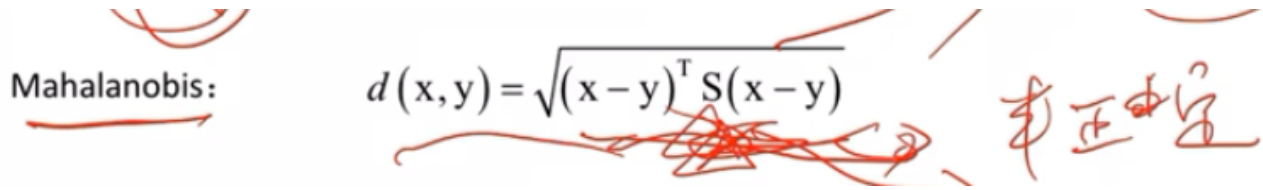
Bhattacharyya距离：$$d(x,y)=-\ln\left(\sum_{i=1}^{n}\sqrt{x_i\cdot y_i}\right)$$

常用的距离，欧式距离



马氏距离

需要矩阵是正定的。



7.5 距离度量学习

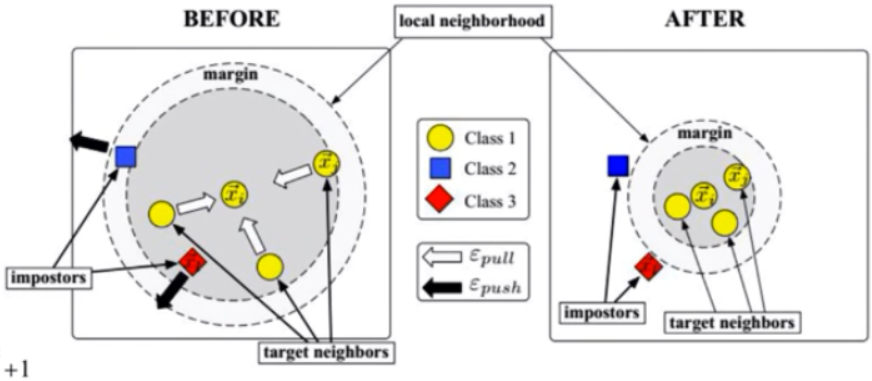
距离度量学习
通过训练样本学习一种距离函数

用于kNN算法
保证对样本进行变换之后，同类样本是k个最近的邻居，不同的样本尽可能远离本样本

$y = Lx$

目标邻居 $j \rightsquigarrow i$

冒充者 $\|L(x_i - x_l)\|^2 \le \|L(x_i - x_j)\|^2 + 1$



Kilian Q Weinberger, Lawrence K Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. 2009, Journal of Machine Learning Research.

马氏距离

距离度量学习

拉损失函数： $\varepsilon_{pull}(L) = \sum_{j \rightsquigarrow i} \|L(x_i - x_j)\|^2$

推损失函数： $\varepsilon_{push}(L) = \sum_{i, j \rightsquigarrow i} \sum_{l} (1 - y_{il}) \left[1 + \|L(x_i - x_j)\|^2 - \|L(x_i - x_l)\|^2 \right]_+$ $\quad [z]_+ = \max(z, 0)$

推损失函数只对不同类型的样本起作用

总损失函数： $\varepsilon(L) = (1 - \mu)\varepsilon_{pull}(L) + \mu\varepsilon_{push}(L)$

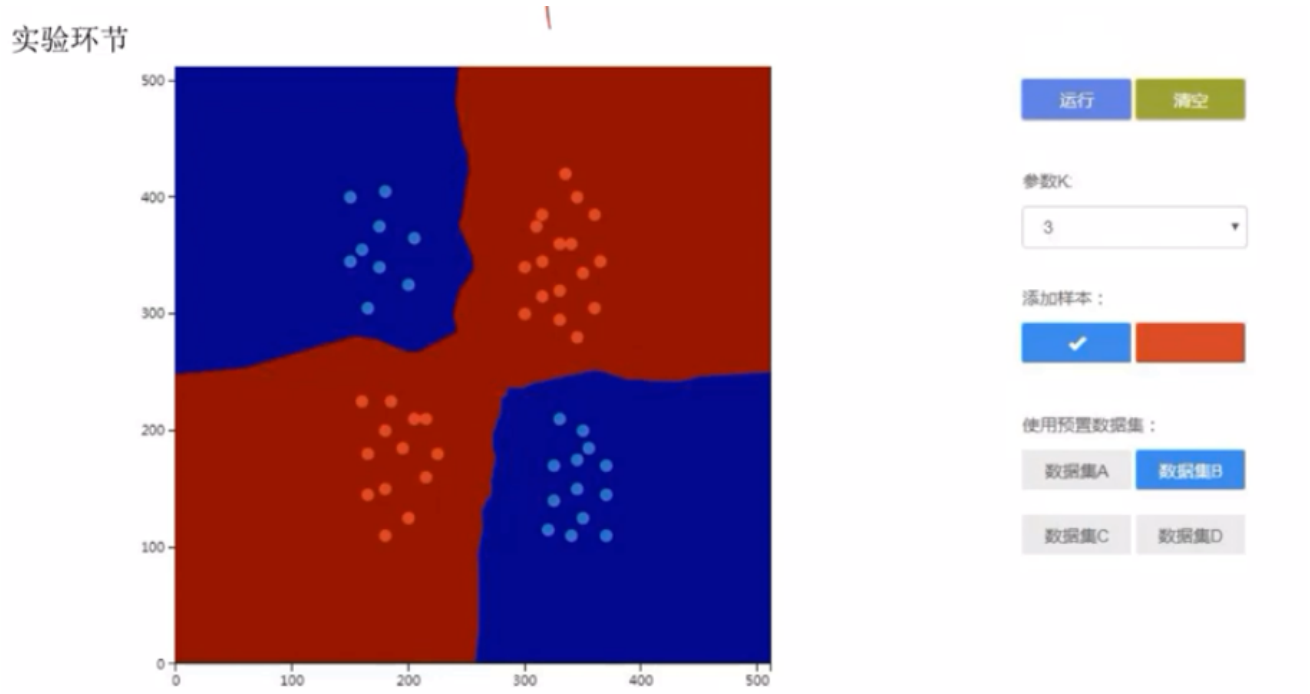
这部分讲得比较多。

分为拉函数。

同样的样本尽可能的相近，不同的样本尽可能的远。

还需要补充。

7.6 实验环节



	KNN	决策树
判别模型	是	
非线性模型	是	
多分类		

贝叶斯分类器，是一个生成模型，用来做分类的算法。

局册数也是一种判别模型，用来做分类。

7.7 实际应用

KNN