## 朴素贝叶斯（native Bayes）

将输入x分到后延概率最大的类y

#### 注意点：

1 朴素贝叶斯对条件概率分布做了条件独立性的假设。

#### 相关知识点：

1 先验概率与后验概率

事情还没有发生,要求这件事情发生的可能性的大小,是先验概率.

事情已经发生,要求这件事情发生的原因是由某个因素引起的可能性的大小,是后验概率.

先验概率是指根据以往经验和分析得到的概率，如全概率公式，它往往作为“由因求果”问题中的“因”出现。后验概率是指在得到“结果”的信息后重新修正的概率，如贝叶斯公式中的，是“执果寻因”问题中的“因”。先验概率与后验概率有不可分割的联系，后验概率的计算要以先验概率为基础。

2 利用极大似然估计计算的时候可能出现概率值为0的情况，会影响到后验概率的结果，使分类产生偏差。因此采取增加的方式，称为拉普拉斯平滑。

#### 优点：

1 因为独立性的假设，模型包含的条件概率的数量大为减少，因此高效，且易于实现。

2 对小规模的数据表现很好，适合多分类任务，适合增量式训练。

#### 缺点：

1 因为独立性的假设，牺牲了一定的分类准确率，分类的性能不一定高。

2 对输入数据的表达形式很敏感。

## KNN(k-nearest neighbor)

#### 注意点：

1 K值的减小意味着整体模型变得复杂，容易发生过拟合。K值变大意味着整体模型变得简单。 K值通常选取一个比较小的数据，采用交叉验证法来选取最优的K值。常用的分类决策规则是多数表决，对应于经验风险最小化。

2 KNN的基本做法是：对给定的训练实例点和输入实例点，首先确定输入实例点的k个最近邻训练实例点，然后利用这k个训练实例点的类的多数来预测输入实例点的类。

3 KNN模型对应于基于训练数据集对特征空间的一个划分。KNN中，当训练集，距离度量，k值和分类决策规则确定后，其结果唯一确定。

4 KNN的实现需要考虑如何快速搜索k个最近邻点。kd树是一种便于对k维空间中的数据进行快速检索的数据结构。kd树是二叉树，表示对k维空间的一个划分，其每个节点对应于k维空间划分的一个超矩形区域。利用kd树可以省去对大部分数据点的搜索，从而减少搜索的计算量。kd树的平均计算复杂度是O(logN).

#### 相关知识点：

#### 优点：

1 思想简单，理论成熟，既可以用来做分类也可以用来做回归。

2 可用于非线性分类。

3 训练时间复杂度为O(n).

4 准确度高，对数据没有假设，对outlier不敏感。

#### 缺点：

1 计算量大

2 样本不平衡（即有些类别的样本数量很多，而其他样本的数量很少）

3 需要大量的内存

## 决策树（decision tree）

熵（entropy）的定义：表示随机变量不确定性的度量。熵越大，随机变量的不确定性越大。

如果对数以2为底，那么熵的单位是比特（bit），如果以为底，那么单位是纳特（nat）。

信息增益（information gain）表示得知特征A的信息而使得训练数据集D的信息的不确定性减少的程度。信息增益打的特征具有更强的分类能力。

#### 注意点：

1 因为从可能的决策树中直接选取最有决策树是NP完全问题。现实中采用启发式方法学习次优的决策树。决策树的学习算法包含3部分，特征性选择，树的生成和树的剪枝、常用的算法有ID3，C4.5和CART。

2 特征选择的目的在于选取对训练数据能够分类的特征。特征选择的关键是其准则。常用的准则如下：

1. 样本集合D对特征A的信息增益（ID3）

其中，H(D)是数据集D的熵，H()是数据集的熵，H(D|A)是数据集D对特征A的条件熵。是D种特征A取第i个值的样本子集，是D中属于第k类的样本子集。|D|表示其样本容量，即样本个数。n是特征A取值的个数，K是类的个数。

1. 样本集合D对特征A的信息增益比（C4.5）

其中，g(D,A)是信息增益，H(D)是数据集D的熵。

1. 样本集合D的基尼指数（CART） CART的决策树是二叉树

特征A条件下集合D的基尼指数：

3 决策树的生成。通常使用信息增益最大，信息增益比最大或者基尼指数最小作为特征选择的准则。决策树的生成往往通过计算信息增益或其他指标，从根节点开始，递归地产生决策树。这相当于用信息增益或其他准则不断的选取局部最优的特征，或将训练集分割为能够基本正确分类的子集。

4 决策树的剪枝。由于生成的决策树存在过拟合的问题，需要对它进行剪枝，以简化学到的决策树。决策树的剪枝，往往从已生成的树上减掉一些叶节点或叶节点以上的子树，并将其父节点或根节点作为新的叶节点，从而简化生成的决策树。

决策树的减枝分为预剪枝和后减枝，预剪枝通过验证集的方式实现，后减枝可以通过验证集或者损失函数来实现。

#### 相关知识点：

#### 优点：

1 计算量简单，可解释性强，比较适合处理有缺失属性值的样本，能够处理不相关的特征。

#### 缺点：

1 容易过拟合（后续出现了随机森林，减小了过拟合的现象）

## 逻辑回归

## 线性回归

## 感知机（perceptron）

感知机是根据输入实例的特征向量x对其进行二类分类的线性分类模型：

感知机模型对应于输入空间（特徵空间）中的分离超平面

#### 注意点：

1 感知机学习算法是基于随机梯度下降法的对损失函数的最优化算法，有原始形式和对偶形式。原始形式中，首先任意选取一个超平面，然后利用梯度下降法不断极小化目标函数。在这个过程中一次随机选取一个误分类点使其梯度下降。

2 感知机的损失函数是根据误分类点到超平面的总距离

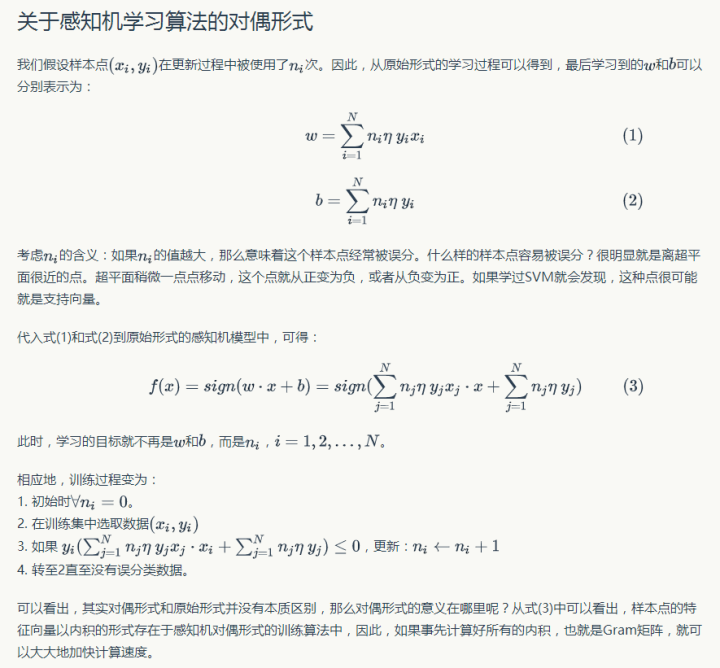
其中M是误分类点的集合。这个损失函数就是感知机学习的经验风险函数。

3 当训练数据集线性可分时，感知机学习算法是收敛的。感知机算法在训练数据集上的误分类次数k满足不等式：

当训练数据集线性可分时，感知机学习算法存在无穷多个解，其解由于不同的初值或不同的迭代顺序而可能有所不同。

4 采用随机梯度下降法进行优化，具体的实现要看书。

5 采用对偶形式的原因



#### 相关知识点：

#### 优点：

#### 缺点：

1 只能数据集线性可分时才能使用，只能用于二分类。

## 支持向量机（SVM）

#### 注意点：

1 支持向量机最简单的情况是线性可分支持向量机，或硬间隔支持向量机。构建它的条件是训练数据线性可分。其学习策略是最大间隔法。可以表示为凸二次规划问题，其原始最优化问题为

求的最优化问题的解为,得到线性可分支持向量机，分离超平面是

分类决策函数是

最大间隔法中，函数间隔与几何间隔是重要的概念。

线性可分支持向量机的最优解存在且唯一。位于间隔边界上的实例点为支持向量。最优分离超平面有支持向量完全决定。

二次规划问题的对偶问题是

#### 相关知识点：

#### 优点：

#### 缺点：

## 神经网络

## GBDT和XGBOOST

## 随机森林

## kmeans

## EM