## 机器学习简答题

1、什么是机器学习？简述机器学习的一般过程。

答：机器学习是通过算法使得机器从大量历史数据中学习规律，从而对新样本做分类或预测。

一般分为训练阶段、测试阶段和工作阶段。训练阶段的主要工作是根据训练数据建立模型，测试阶段的主要工作是利用验证集对模型评估与选择，工作阶段的主要工作是利用建立好的模型对新的数据进行预测与分类

### 2、什么是监督学习和非监督学习，它们之间的区别有哪些？

监督学习，是指训练集的数据已经分好类别，通过对带有标签的数据进行学习，来调整分类器的参数，使其达到所要求性能的过程。当用测试集对模型进行测试时，给出D测={xi }=>{yi}。

非监督学习，需要将一系列没有标签和类别未知的数据，输入到算法中，需要根据样本之间的相似性对样本集进行分类(聚类)试图使类内差距最小化，类间差距最大化

重点：有无标签

监督学习：分类和回归（决策树，线性回归，神经网络，支持向量机，贝叶斯分类器）

非监督学习：聚类（原型聚类：k均值算法，学习向量化，高斯混合聚类。

密度聚类：DBSCAN算法 层次聚类：AGNES算法）

**3、什么是过拟合和欠拟合，产生的原因，以及解决办法**

过拟合：在训练集上使用了一个非常复杂的模型，以至于这个模型在拟合训练集时表现非常好，但是在测试集的表现非常差。

过拟合原因：训练数据集样本单一、训练样本噪音数据干扰过大、模型过于复杂。

过拟合解决办法：

①在训练和建立模型的时候，一定要从相对简单的模型开始，不要一上来就把模型调得非常复杂、特征非常多。

②数据采样一定要尽可能地覆盖全部数据种类。

③在模型的训练过程中，我们也可以利用数学手段预防过拟合现象的发生，例如：可以在算法中添加惩罚函数来预防过拟合。

欠拟合：如果模型过于简单，对于训练集的特点都不能完全考虑到的话，那么这样的模型在训练集和测试集的表现都会非常的差。

欠拟合原因：模型复杂度过低、特征量过少

欠拟合解决办法：

①通过增加新特征来增大假设空间。

②添加多项式特征，例如将线性模型通过添加二次项或者三次项使模型泛化能力更强。

③减少正则化参数。

④使用非线性模型，比如决策树、深度学习等模型。

⑤调整模型的容量，模型的容量是指其拟合各种函数的能力，容量低的模型可能很难拟合训练集。

### 4、解释分类、聚类、回归、损失函数

分类：根据一些给定的已知类别标号的样本，通过训练得到某种目标函数，使它能够对未知类别的样本进行分类。

聚类：指事先并不知道任何样本的类别标号，希望通过某种算法来把一组未知类别的样本划分成若干类别，这在机器学习中被称作无监督学习。

回归：用于预测输入变量和输出变量之间的关系，特别是当输入变量的值发生变化时，输出变量的值随之发生的变化。

损失函数：用来估量模型的预测值f(x)与真实值Y的不一致程度，它是一个非负实值函数,通常使用L(Y, f(x))来表示，损失函数越小，模型的鲁棒性就越好。

### 属性空间、标签空间、假设、版本和版本空间

属性空间（又称为样本空间，输入空间）：描述数据的各个数据维度张成的空间

标签空间（又称为标记空间）：所有标记的集合

假设：一个从有限属性空间到有限标签空间的映射。

版本：与训练集相符合，也就是能够将训练集上所有特征毫无错误的映射到对应标签的假设称为版本。

版本空间：所有版本的集合称为版本空间。

### **训练误差、测试误差、泛化误差，训练误差很小，测试误差一定很小吗？**

训练误差：训练集上的训练所得模型与目标的差异

泛化误差：在除去训练集的样本空间上训练所得模型与目标的差异

测试误差：测试集上的训练所得模型与目标的差异

哪个指标是衡量模型性能最公平的指标？

答：泛化误差

为什么要谈测试误差？

答：因为泛化误差要遍历整个除去训练集的样本空间，现实情况下不可能（因为你连样本空间的边界都不知道，几乎所有样本空间都是不可数的。比如自动驾驶，你能列出所有可能的驾驶场景吗？）

**7、如何划分数据集以及评估方法有哪几种**

留出法：就是将整个数据集 按照某种比例进行划分成训练集和测试集，要注意分层对数据采样，多次重复划分，测试集最好保持在20-30%的数据量上

交叉验证法：将全部数据集D分成 k个不相交的子集，进行k次训练和测试，每次从分好的子集中里面，拿出一个子集作为测试集，其它k-1个子集作为训练集，计算k次测试结果的平均值，作为该模型的真实结果。留一法：是交叉验证法的一种，例如D中有m个样本，令k=m，则每个子集仅包含一个样本。适合小样本数据。

自助法：假定D中包含m个样本，通过对它进行采样产生数据集D’，每次随机从D中挑选一个样本，将其拷贝放入D’中，然后再将该样本放回D中，这个过程重复m次，则得到了包含m个样本的数据集D’，可将D’用在训练集，D\D’用作测试集。

**8、简述线性回归与逻辑回归的区别**

线性回归：用一条高维一次曲线或曲面，去拟合变量之间的关系。利用线性回归方程的最小平方函数对一个或多个自变量和因变量之间关系进行建模的一种回归分析。

线性回归模型公式：h(x)=ω1x1+ω2x2+⋅⋅+ωnxn+b

{x}是模型的特征空间，ω是特征向量的权值，b是常量。

逻辑回归：一个应用非常广泛的机器学习分类算法，它将数据拟合到一个logit函数中，从而完成对事件发生概率的预测。

区别：

①线性回归要求变量服从正态分布，逻辑回归对变量分布没有要求。

②线性回归要求因变量是连续性数值变量，而逻辑回归要求因变量是分类型变量。

③线性回归要求自变量和因变量呈线性关系，而逻辑回归不要求自变量和因变量呈线性关系

④逻辑回归是分析因变量取某个值的概率与自变量的关系，而线性回归是直接分析因变量与自变量的线性关系

⑤线性回归优化的目标函数是均方误差（最小二乘法)，而逻辑回归优化的是似然函数

⑥逻辑回归处理的是分类问题，线性回归处理的是回归问题，这也导致了两个模型的取值范围不同：0-1和实数域

联系：

①两个都是线性模型，线性回归是普通线性模型，逻辑回归是广义线性模型

②表达形式上，逻辑回归是线性回归套上了一个Sigmoid函数

线性回归优缺点：

优点：

①思想简单，实现容易，建模迅速，模型容易理解，对于小数据量、简单的关系很有效。

②是许多强大的非线性模型的基础。

③能解决回归问题。

缺点：

①对于非线性数据或者数据特征间具有相关性多项式回归难以建模.

②难以很好地表达高度复杂的数据，容易欠拟合。

逻辑回归优缺点：

优点：

①训练速度快，分类的时候，计算量仅仅只和特征的数目相关

②简单易理解，模型的可解释性非常好

③适合二分类问题，不需要缩放输入特征

缺点：

①不能用LR去解决非线性问题

②对多重共线性数据较为敏感

③很难处理数据不平衡问题

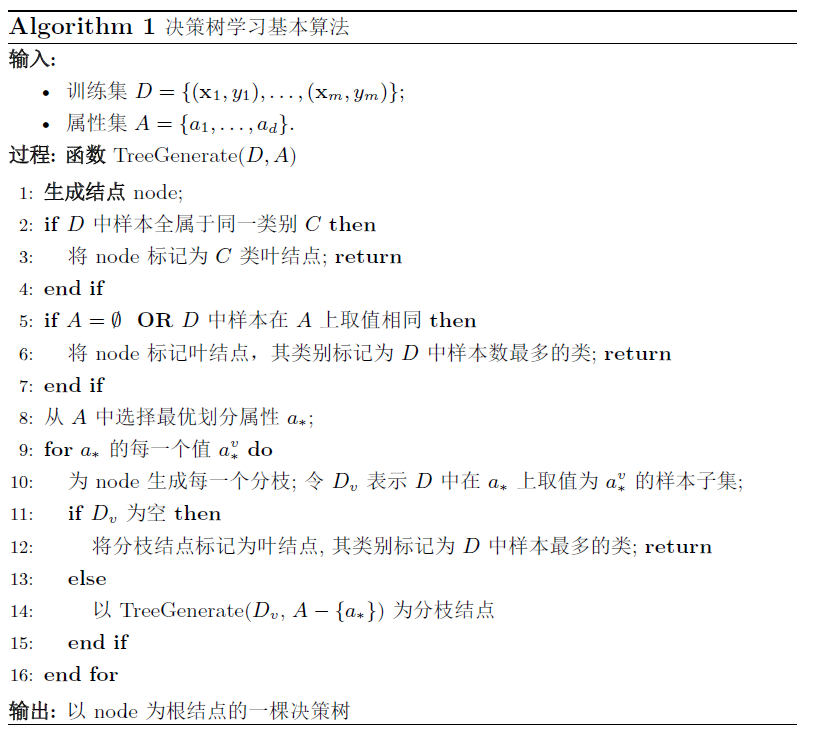
④准确率并不是很高，因为形式非常简单，很难去拟合数据的真实分布

1. **生成决策树算法**

特征选择：特征选择是指从训练数据中众多的特征中选择一个特征作为当前节点的分裂标准，如何选择特征有着很多不同量化评估标准，从而衍生出不同的决策树算法。

决策树生成：根据选择的特征评估标准，从上至下递归地生成子节点，直到数据集不可分则停止决策树，停止生长。树结构来说，递归结构是最容易理解的方式。

剪枝：决策树容易过拟合，一般来需要剪枝，缩小树结构规模、缓解过拟合。剪枝技术有预剪枝和后剪枝两种。

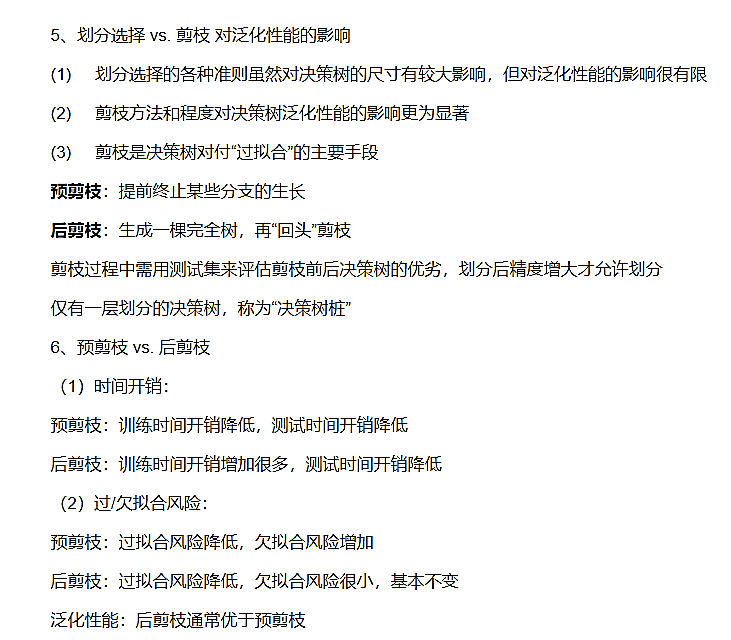


1. **简述剪枝的目的以及常用的两种剪枝方式的基本过程**

目的：剪枝是决策树学习算法对付“过拟合”的主要手段，通过主动去掉一些分支来降低过拟合的风险。基本策略有“预剪枝”和“后剪枝”。

“预剪枝”对每个结点划分前先进行估计，若当前结点的划分不能带来决策树泛化性能的提升，则停止划分，并标记为叶结点。

“后剪枝”先从训练集生成一棵完整的决策树，然后自底向上对非叶子结点进行考察，若该结点对应的子树用叶结点能带来决策树泛化性能的提升，则将该子树替换为叶结点。



1. **请简述随机森林的生成方法以及其随机性体现在哪里？**

生成方法：

1, 假设有N个样本，则有放回的随机选择N个样本（每次随机选择一个样本，然后将该样本放回并继续选择）。采用选择好的N个样本用来训练一个决策树，作为决策树根节点处的样本。

2. 假设每个样本有M个属性，在决策树做节点分裂时，随机从这M个属性中选取m个属性，满足条件 m << M 。然后采用某种策略（比如信息增益最大化）从m个属性中选择一个最优属性作为该节点的分类属性。

3. 决策树形成过程中重复步骤2来计算和分裂节点。一直到节点不能够再分裂，或者达到设置好的阈值（比如树的深度，叶子节点的数量等）为止。注意整个决策树形成过程中没有进行剪枝。

4. 重复步骤1~3建立大量的决策树，这样就构成了随机森林。

随机森林的随机性体现在每棵树的训练样本是随机的，树中每个节点的分裂属性集合也是随机选择确定的，如下：

（1）随机采样：随机森林在计算每棵树时，从全部训练样本（样本数为N）中选取一个可能有重复的、大小同样为N的数据集进行训练（即Booststrap采样）。

（2）特征选取的随机性：在节点分裂计算时，随机地选取所有特征的一个子集，用来计算最佳的分割方式。

优点

特征和数据的随机抽样

1. 它可以处理很多高维度（特征很多）的数据，并且不用降维，无需做特征选择；
2. 如果有很大一部分的特征遗失，仍可以潍柴准确度；
3. 不容易过拟合；

树模型的特性

（4）较好的解释性和鲁棒性；

（5）能够自动发现特征间的高阶关系；

（6）不需要对数据进行特殊的预处理如归一化；

算法结构

（7）训练速度比较快，容易做成并行方法；

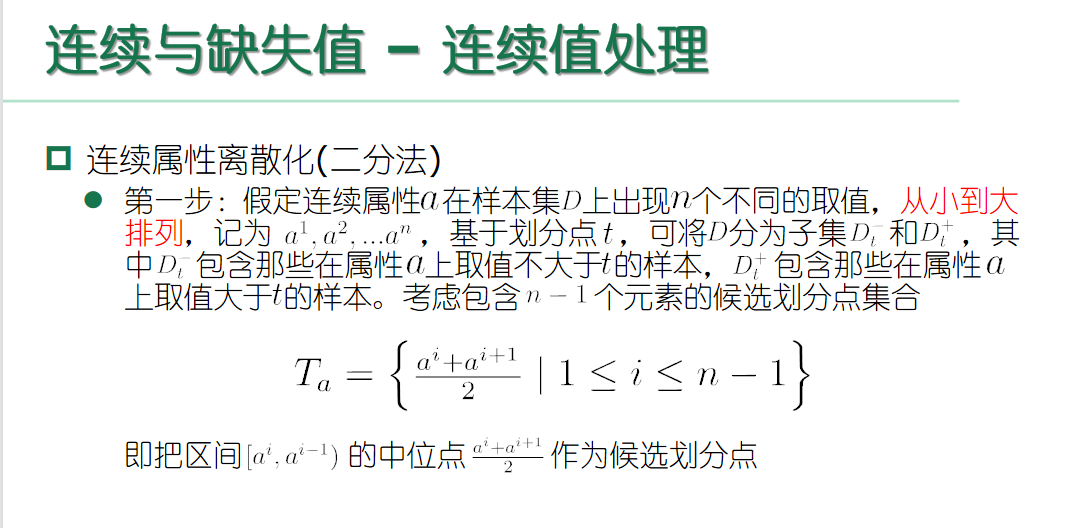
（8）实现起来比较简单。

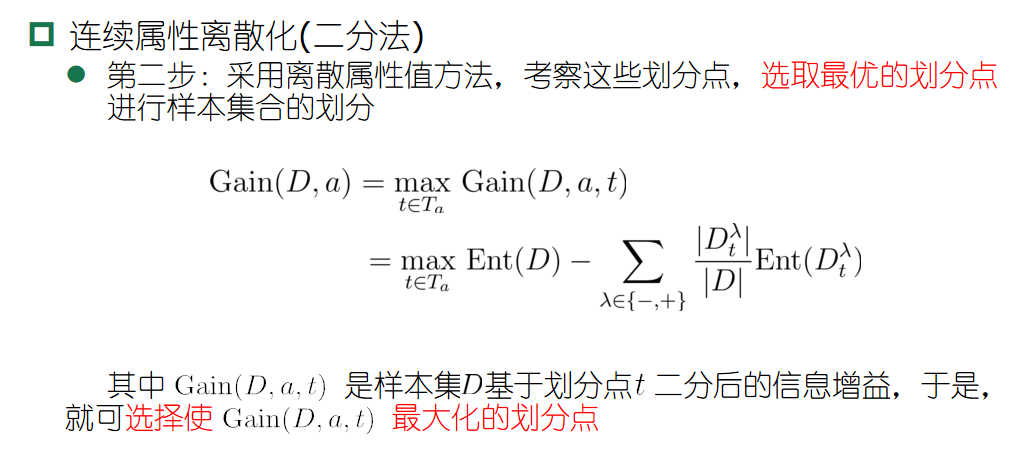
缺点

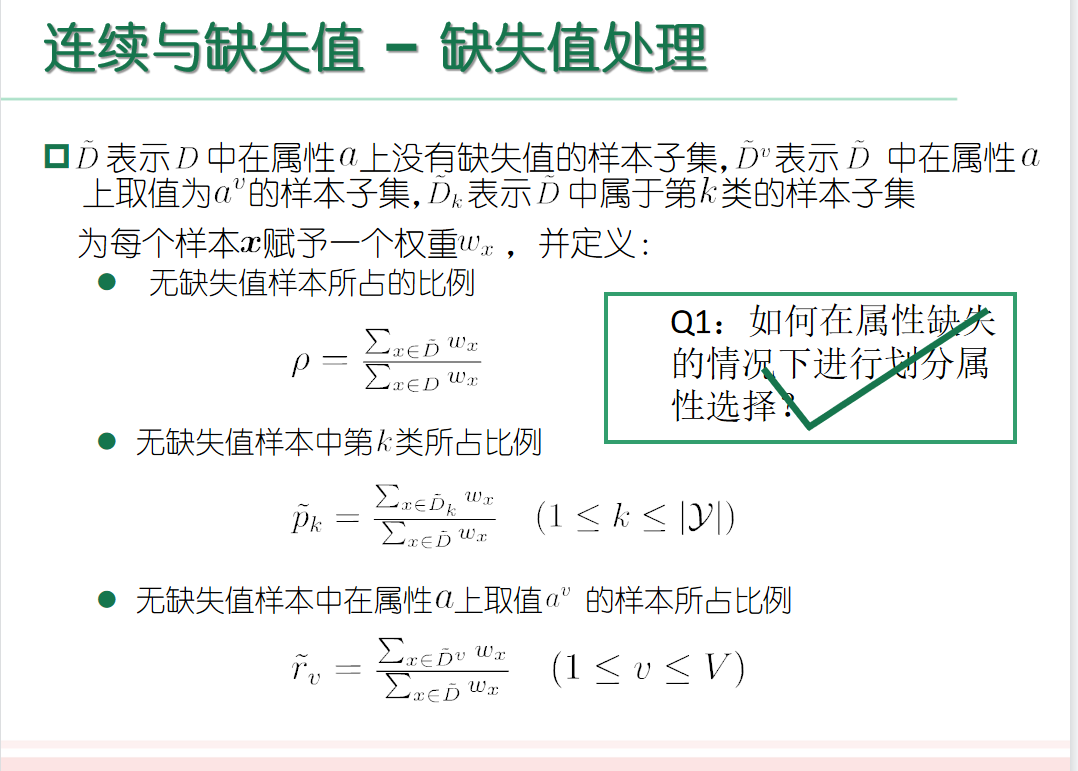
随机森林已经被证明在某些噪音较大的分类或回归问题上会过拟合。（决策树的学习本质上进行的是决策节点的分裂，依赖于训练数据的空间分布）

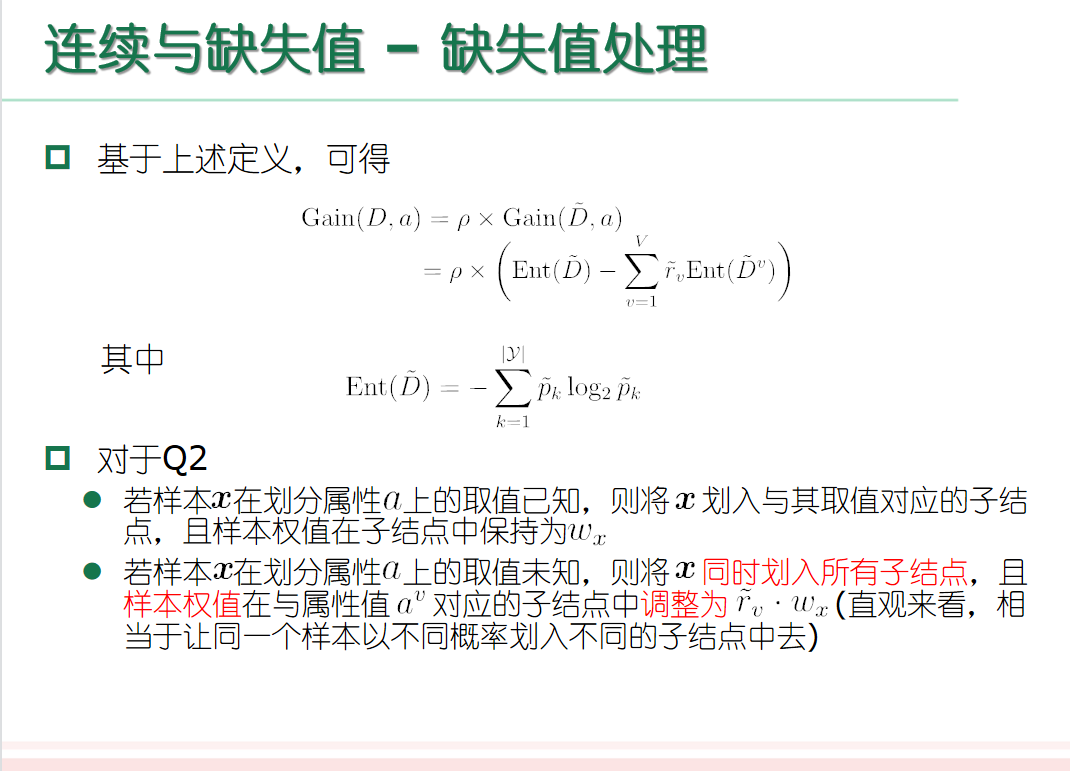
对于有不同取值的属性的数据，取值划分较多的属性会对随机森林产生更大的影响，所以随机森林在这种数据上产出的属性权值是不可信的。

1. **生成决策树时对缺失值与连续值怎么处理**









1. **BP算法**

BP算法：学习过程由信号的正向传播与误差的反向传播两个过程组成。

BP算法的基本思想是：学习过程由信号的正向传播与误差的反向传播两个过程组成。正向传播时，输入样本从输入层传入，经各隐层逐层处理后，传向输出层。若输出层的实际输出与期望的输出(教师信号)不符，则转入误差的反向传播阶段。误差反传是将输出误差以某种形式通过隐层向输入层逐层反传，并将误差分摊给各层的所有单元，从而获得各层单元的误差信号，此误差信号即作为修正各单元权值的依据。

这种信号正向传播与误差反向传播的各层权值调整过程，是周而复始地进行的。权值不断调整的过程，也就是网络的学习训练过程。此过程一直进行到网络输出的误差减少到可接受的程度，或进行到预先设定的学习次数为止。

BP学习过程：

(1)组成输入模式由输入层经过隐含层向输出层的“模式顺传播”过程。

(2)网络的期望输出与实际输出之差的误差信号由输出层经过隐含层逐层休整连接权的“误差逆传播”过程。

(3)由“模式顺传播”与“误差逆传播”的反复进行的网络“记忆训练”过程。

(4)网络的总体误差趋向极小值的“学习收敛”过程。

BP算法不足：

(1)训练时间较长。对于某些特殊的问题，运行时间可能需要几个小时甚至更长，可以采用自适应的学习率加以改进。

(2)易陷入局部极小值。这主要是因为BP算法所采用的是梯度下降法，不同的起始点可能导致不同的极小值产生，即得到不同的最优解。

(3)训练过程中，学习新样本时有遗忘旧样本的趋势。

(4)梯度越来越稀疏，从顶层越往下，误差校正信号越来越小，梯度扩散

(5)通常，算法只能用于有标签的数据训练，但大部分数据是没有标签的

1. **多层神经网络一般由哪些层组成，每层（节点）什么含义**

在神经网络中，处理单元通常按层次分布于神经网络的输入层、隐含层和输出层

输入层：接受与处理训练数据集中的各输入变量值

隐含层：实现非线性数据的线性变换

输出层：给出输出变量的分类或预测结果

1. **全局最小与局部最小**

局部极小解：参数空间中某个点，其邻域点的误差函数值均不小于该点的误差函数值。  
全局最小解：参数空间某个点，所有其他点的误差函数值均不小于该点的误差函数值。

局部极小可以有多个，而全局最小只有一个。全局最小一定是局部极小，而局部极小却不一定是全局最小。

在现实任务中，通常使用以下策略尽可能接近全局最小：  
1.以多组不同参数值初始化网络，迭代停止后，取误差最小解为最终参数  
2.使用模拟退火技术  
3.使用随机梯度下降，即计算梯度时加入随机因素，使得在局部极小时梯度仍可能不为0，从而继续迭代

1. **支持向量机与感知机，及两者的区别与联系**

感知机：二维空间中找到一条直线可以把所有二元类别分离开，三维或多维空间中，找到一个分离超平面把所有二元类别分离开。

f(x)=sign(wx+b) 将f(x)称为感知机，w,b分别为该感知机的权值和偏置，sign是符号函数，有两种输出结果 IMG_256

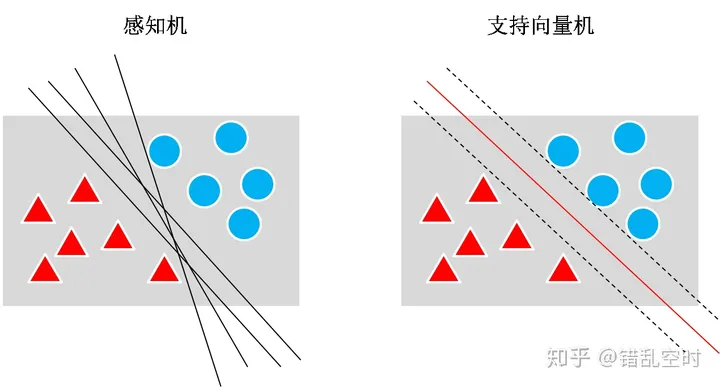
定义损失函数：IMG_256 （M为误分类点集合）

补：多层感知机：除了输入和输出层之外，它中间可以有多个隐含层。

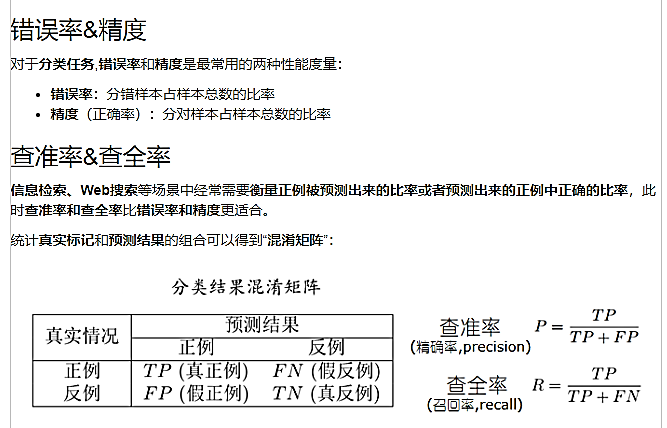
支持向量机：是一种二分类监督学习模型，定义在特征空间上间隔最大的线性模型。

分类决策函数：IMG_256

区别：支持向量机支持在特征空间中寻找出非线性的平面(非线性支持向量机).支持向量机寻找出的超平面唯一且最优，而感知机是根据误分类点定义出的代价函数求得的超平面不唯一，包含多个。感知机，只需要找到可以将数据正确划分的超平面即可，而SVM需要找到间隔最大的超平面将数据划分开，感知机超平面无数个，而SVM只有一个



1. **错误率&精度&查全率&查准率**



1. **简述基于核函数的非线性支持向量机的基本思想**

核函数不显示定义映射函数，在原始样本空间中通过计算核函数的值来代替特征空间中的内积。

基于核函数的非线性支持向量机的基本思想就是通过非线性变换将输入空间对应到一个特征空间上，使得输入空间中的超**曲面**模型对应于特征空间上的一个超**平面**模型也就是支持向量机，问题转化为在特征空间中求解支持向量机。

1. **简述卷积神经网络与传统的神经网络的区别**

传统的神经网络是一种层级结构，由输入层，隐藏层，输出层构成，每层神经元与下层神经元完全互连，神经元之间不存在同层连接，也不存在跨层连接。

卷积神经网络相较于传统神经网络的特殊性在于权值共享和局部连接两个方面。局部连接：每个神经元只与上一层的部分神经元相连，只感知局部，而不是整幅图像；权值共享：每一个神经元都可以看作一个滤波器，同一个神经元使用一个固定的卷积核去卷积整个图像,可以认为一个神经元只关注一个特征，而不同的神经元关注多个不同的特征。

1. **朴素贝叶斯分类器解决了什么障碍？ 它的关键假设是什么？**

基于贝叶斯公式来估计后验概率P(c|x)的主要困难在于：类条件概率P(x|c)是所有属性上的联合概率（即x代表的是多个属性），难以从有限的训练样本直接估计而得。为了避开这个障碍，朴素贝叶斯分类器（naive Bayes classifier）采用了“属性条件独立性假设”:对已知类别，假设所有属性相互独立。换言之，假设每个属性独立地对分类结果发生影响。

1. **集成学习，Bagging与Boosting方法及两者区别**

集成学习：就是将多个个体学习器用某种策略组合起来成为强学习器，通常个体学习器是由已有的学习算法从训练数据中学习产生的。有同质集成和异质集成。同质集成，即集成中只包含同种类型的个体学习器，这些个体学习器亦被称为基学习器，产生这些基学习器的学习算法叫作基学习算法。异质集成，即集成中的个体学习器由不同的学习算法生成。

Bagging：是一种并行集成算法。该算法的思想是分别构造多个基学习器（弱学习器），多个基学习器相互之间是并行的关系，通过自助采样法进行训练，最终将多个基学习器结合。对分类问题采用投票方式，对回归问题采用简单平均方法对新示例进行判别。

Baggingt特点：

①训练集是在原始集中有放回选取的，从原始集中选出的各轮训练集之间是独立的

②使用均匀取样，每个样例的权重相等

③所有预测函数的权重相等。

④各个预测函数可以并行生成

⑤趋于降低方差，使模型更稳定

Boosting：是一种迭代算法。每轮迭代中会在训练集上产生一个新的分类器，然后使用该分类器对所有样本进行分类，以评估每个样本的重要性。Boosting算法要涉及到两个部分，加法模型和前向分步算法。加法模型就是说强分类器由一系列弱分类器线性相加而成。前向分步就是说在训练过程中，下一轮迭代产生的分类器是在上一轮的基础上训练得来的。

Boosting特点：

①每一轮的训练集不变，只是训练集中每个样例在分类器中的权重发生变化。而权值是根据上一轮的分类结果进行调整。

②根据错误率不断调整样例的权值，错误率越大则权重越大。

③每个弱分类器都有相应的权重，对于分类误差小的分类器会有更大的权重。

④各个预测函数只能顺序生成，因为后一个模型参数需要前一轮模型的结果。

⑤趋于降低偏差，模型准确率更高。

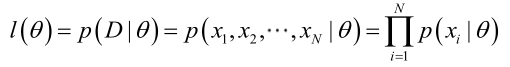
1. **最小二乘法&极大似然法**

最小二乘法：是一种求解无约束最优化问题的常用方法，并且也可以用于曲线拟合，来解决回归问题。最小二乘法实质就是最小化“均方误差”。

f(xi)=wxi+b， (w\*,b\*)=arg min IMG_256

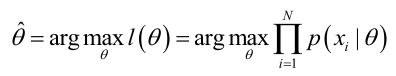
极大似然法：极大似然估计的目的就是：利用已知的样本结果，反推最有可能（最大概率）导致这样结果的参数值。

 似然函数（linkehood function）：联合概率密度函数IMG_256称为相对于IMG_257的θ的似然函数。



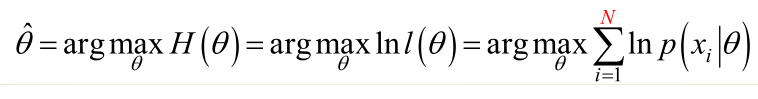
# 求解极大似然函数

        ML估计：求使得出现该组样本的概率最大的θ值。



         实际中为了便于分析，定义了对数似然函数：

IMG_257



对该函数求导其值为0求得解。

1. **简述K均值聚类算法的流程**

假设有m条数据，n个特性，则K均值聚类算法的流程如下：

1. 随机选取k个点作为起始中心(k行n列的矩阵，每个特征都有自己的中心)；（2）遍历数据集中的每一条数据，计算它与每个中心的距离；（3）将数据分配到距离最近的中心所在的簇；（4）使用每个簇中的数据的均值作为新的簇中心；（5）如果簇的组成点发生变化，则跳转执行第2步；否则，结束聚类。