一、简答题，每小题10分，共40分

二、作图题，每小题10分，共10分

三、计算题，每小题10分，共30分

四、分析题，每小题10分，共20分

（安大重点）特征选择，朴素贝叶斯，主成分分析（作图），线性回归（作图），决策树（以西瓜为例），集成学习

（安师重点）机器学习概念与评价指标，降维与特征选择，朴素贝叶斯，神经网络，支持向量机，决策树与随机森林，集成学习。

1. 机器学习概念：机器学习是一门多领域交叉学科，它专门研究计算机怎样模拟或实现人类的学习行为，以获取新的知识或技能，重新组织已有的知识结构使之不断改善自身的性能。主要类型有监督学习，无监督学习，强化学习。
2. 评价指标：

（1）分类任务评价指标：准确率，精确率，召回率，F1-Score

1. 回归任务评价指标：均方误差，均方根误差，平均绝对误差。

（去年考题）

**解答题**

1. 为什么可以对高维数据进行降维？请列举三个数据降维算法。

**为什么可以对高维数据进行降维？**

高维数据通常存在冗余信息，这是因为：

1. **数据分布特性：**
   * 高维数据可能存在相关性（例如多个特征之间高度相关），导致实际信息量少于特征维度。
   * 数据通常集中分布在一个低维的流形上，而不是均匀分布在高维空间。
2. **噪声和冗余：**
   * 高维数据中可能包含噪声，这些噪声对分析结果贡献有限，可以通过降维去除。
3. **计算效率和可视化需求：**
   * 降维可以减少数据维度，从而降低计算复杂度，便于存储和处理。
   * 降维后的数据易于可视化，帮助人们理解数据的内在结构。

**数据降维算法**

1. **主成分分析 (PCA, Principal Component Analysis):**
   * 通过线性变换，将高维数据投影到方差最大的方向上，以保留尽可能多的信息。
   * 是一种线性降维方法，假设数据的主方向可以通过特征值分解得到。
2. **t-SNE (t-Distributed Stochastic Neighbor Embedding):**
   * 一种非线性降维方法，用于可视化高维数据。
   * 通过最小化高维空间和低维空间之间的概率分布差异来保持局部邻域关系。
3. **线性判别分析 (LDA, Linear Discriminant Analysis):**
   * 专注于有监督学习，将数据投影到能最大化类间方差与类内方差比率的方向上。
   * 通常用于分类任务中降维。
4. 对于线性回归向量机，通过什么方式解决特异点问题，线性SVM能处理这种情形吗？以什么方式？

**线性回归向量机如何解决特异点问题**

**特异点问题**是指某些数据点（如离群点）对模型造成不良影响的问题，特别是在线性回归中，少量异常点可能极大地改变回归结果。

在线性回归向量机中，通常通过以下方式解决特异点问题：

1. **引入正则化。**
2. **使用鲁棒损失函数：**
   * 将损失函数从传统的均方误差（MSE）换为对异常值更鲁棒的函数。

**线性 SVM 是否能处理特异点问题及解决方式**

线性支持向量机 (Linear SVM) 在处理分类问题时，具有一定的抗特异点能力，其解决方式包括：

1. **软间隔方法：**
   * 通过控制 CC 值，可以减少离群点对模型的影响。
2. **间隔最大化的本质：**
   * SVM 寻求的是一个最大化分类间隔的超平面，异常点通常会对间隔宽度产生较小影响，除非离群点位于分类边界附近。

**总结**

* **线性回归向量机**通过正则化和鲁棒损失函数解决特异点问题。
* **线性 SVM** 能一定程度上处理特异点问题，主要依赖软间隔方法。特异点会通过松弛变量的引入得到合理的权衡。

1. 深度神经网络出现梯度消失现象和局部最优现象的原因是什么？请分别举例两种方法。

#### 1. ****梯度消失现象的原因****

* **本质原因：**
  + 在深度神经网络中，梯度是通过反向传播从输出层向输入层逐层传播的。当使用激活函数如 Sigmoid 或 Tanh 时，其导数值会很小（尤其是在函数值接近于饱和区间时），导致梯度在层与层之间不断缩小，最终接近于零。这种现象在深层网络中尤为明显。
* **具体表现：**
  + 导致权重更新停滞，网络无法有效学习。
* **例子：**
  + 一个深层网络使用 Sigmoid 激活函数时，如果输入较大，Sigmoid 的梯度接近 0，造成梯度消失。

#### 2. ****局部最优现象的原因****

* **本质原因：**
  + 损失函数的优化过程可能陷入局部最优解或鞍点。深度神经网络的高维非凸优化问题通常伴随多个局部最优点和鞍点，导致梯度下降方法停滞在非全局最优点。
* **具体表现：**
  + 训练过程中，损失函数停滞不前，难以进一步优化。
* **例子：**
  + 一个简单的两层神经网络在优化非凸损失函数时，由于权重的随机初始化，可能陷入局部最优点。

### 方法举例

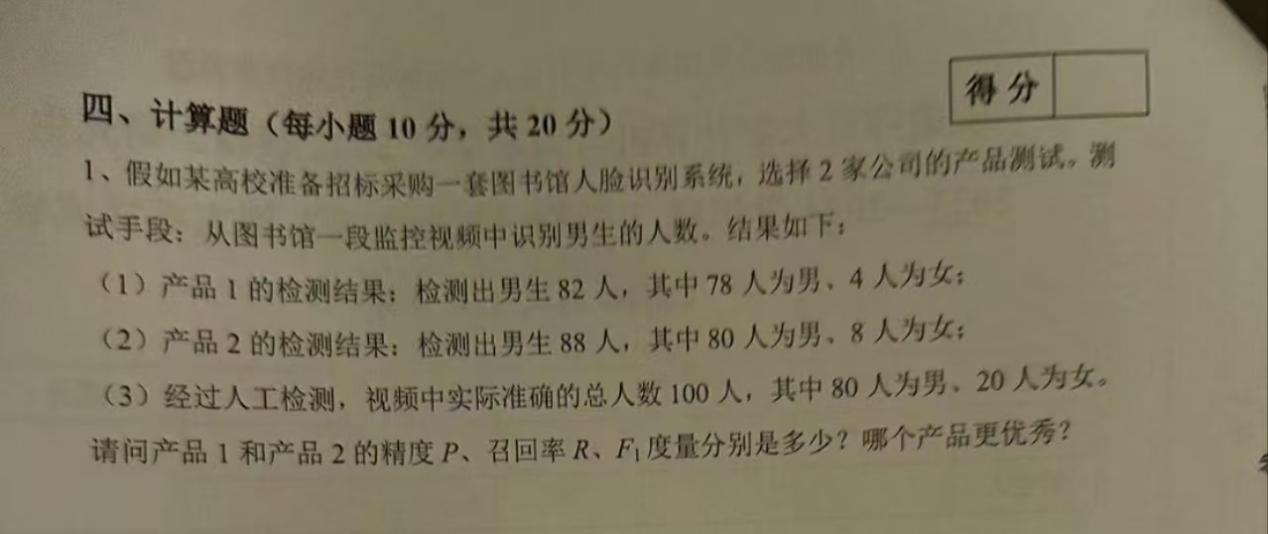
#### 1. ****应对梯度消失现象的方法****

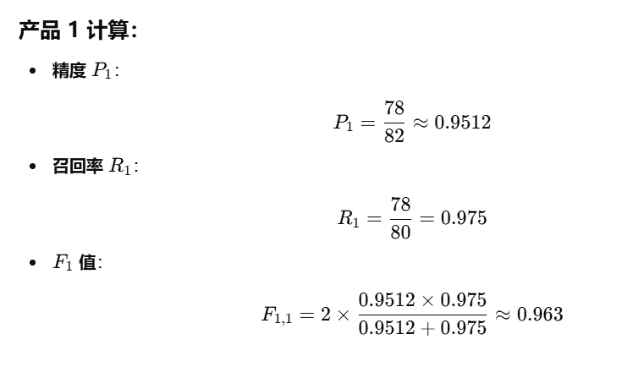
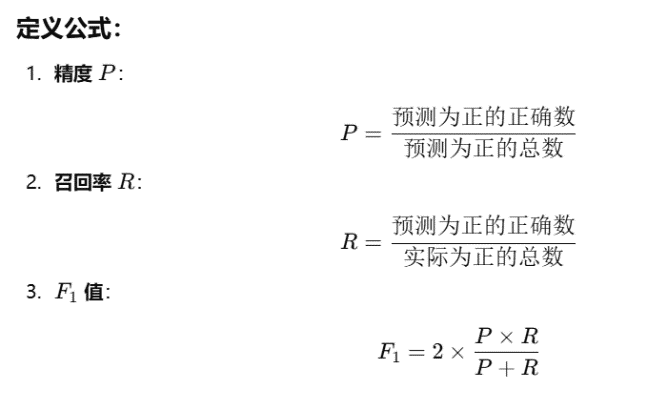
* **使用 ReLU 激活函数：**
  + ReLU（Rectified Linear Unit）函数的梯度为常数 1（正区间），避免了梯度消失的问题。
  + 例子：将激活函数从 Sigmoid 改为 ReLU，可显著改善深层网络的梯度流动。
* **权重初始化技巧：**
  + 通过 He 初始化或 Xavier 初始化，将权重初始化为适当的范围，减少梯度消失的风险。

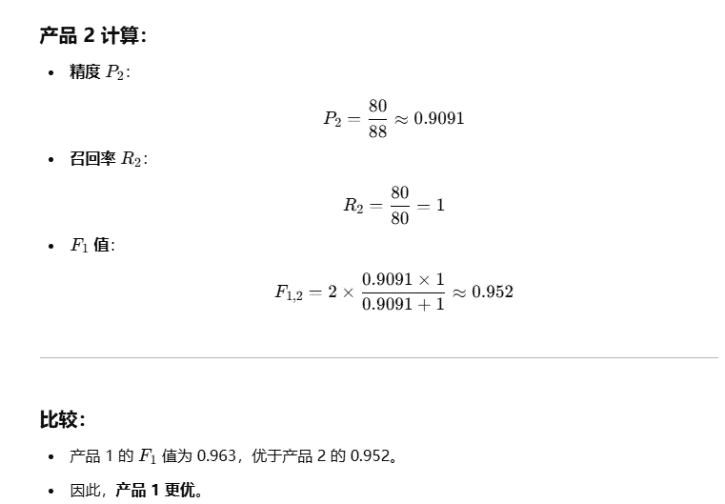
#### 2. ****应对局部最优现象的方法****

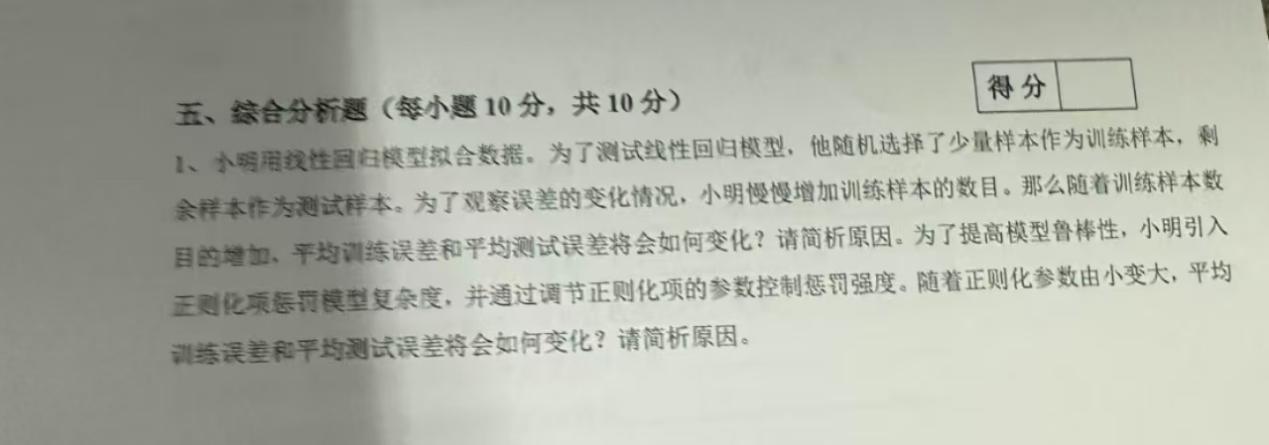
* **使用随机梯度下降变体：**
  + 添加动量项或采用 Adam 优化器，使优化过程更有可能摆脱局部最优点。
  + 例子：使用 Adam 优化器，可以动态调整学习率，跳出局部最优点。
* **增加网络深度或宽度：**
  + 更大的模型具有更平滑的损失表面，更容易找到接近全局最优的点。
  + 例子：将简单的两层网络改为多层卷积神经网络（如 ResNet），并通过跳跃连接减少陷入局部最优的概率。

**计算题**









### 问题 1：随机增加训练样本数量，平均训练误差与平均测试误差的变化及原因

#### 变化：

1. **平均训练误差**：
   * 随着训练样本数量增加，模型可以更好地拟合训练数据，平均训练误差通常会**减小**。
   * 但在极端情况下（样本数过多、模型容量不足），可能会导致训练误差趋于一个固定值。
2. **平均测试误差**：
   * 增加训练样本通常可以提高模型的泛化能力，减少过拟合，因此平均测试误差会**减小**。

#### 原因：

* 随着训练样本增多，模型能够接触到更多的数据分布，学习更稳定的特征，减少对少数异常数据的依赖，提升泛化性能。

### 问题 2：引入正则化项并调整参数强度后，平均训练误差与平均测试误差的变化及原因

#### 变化：

1. **平均训练误差**：
   * 引入正则化项后，模型的自由度降低，可能导致拟合训练数据的能力下降，因此**平均训练误差会增加**。
2. **平均测试误差**：
   * 正则化能够抑制过拟合，减少模型对训练数据的过度拟合，提升泛化能力。因此，**平均测试误差会减小**。

#### 原因：

* 正则化通过对模型复杂度的限制（如 L1 或 L2 正则化），避免模型过度拟合训练数据，从而提高泛化性能。
* 参数强度调整合理时（例如合适的正则化系数），测试误差下降幅度可能更明显。

（安建）

**为什么要进行特征选择？特征选择主要有哪3种方式？3种特征选择方式的主要区别是什么？**

（1）去除不相关特征降低学习难度；减轻维度灾难（仅在少量属性上构建模型、增加有限样本在属性空间的分布密度）；

（2）过滤式、包裹式、嵌入式；

（3）过滤式：先对数据集进行特征选择，其过程与后续学习无关；包裹式：就是一个分类器，它是将后续的学习器的性能作为特征子集的评价标准；嵌入式：是学习器自主选择特征。

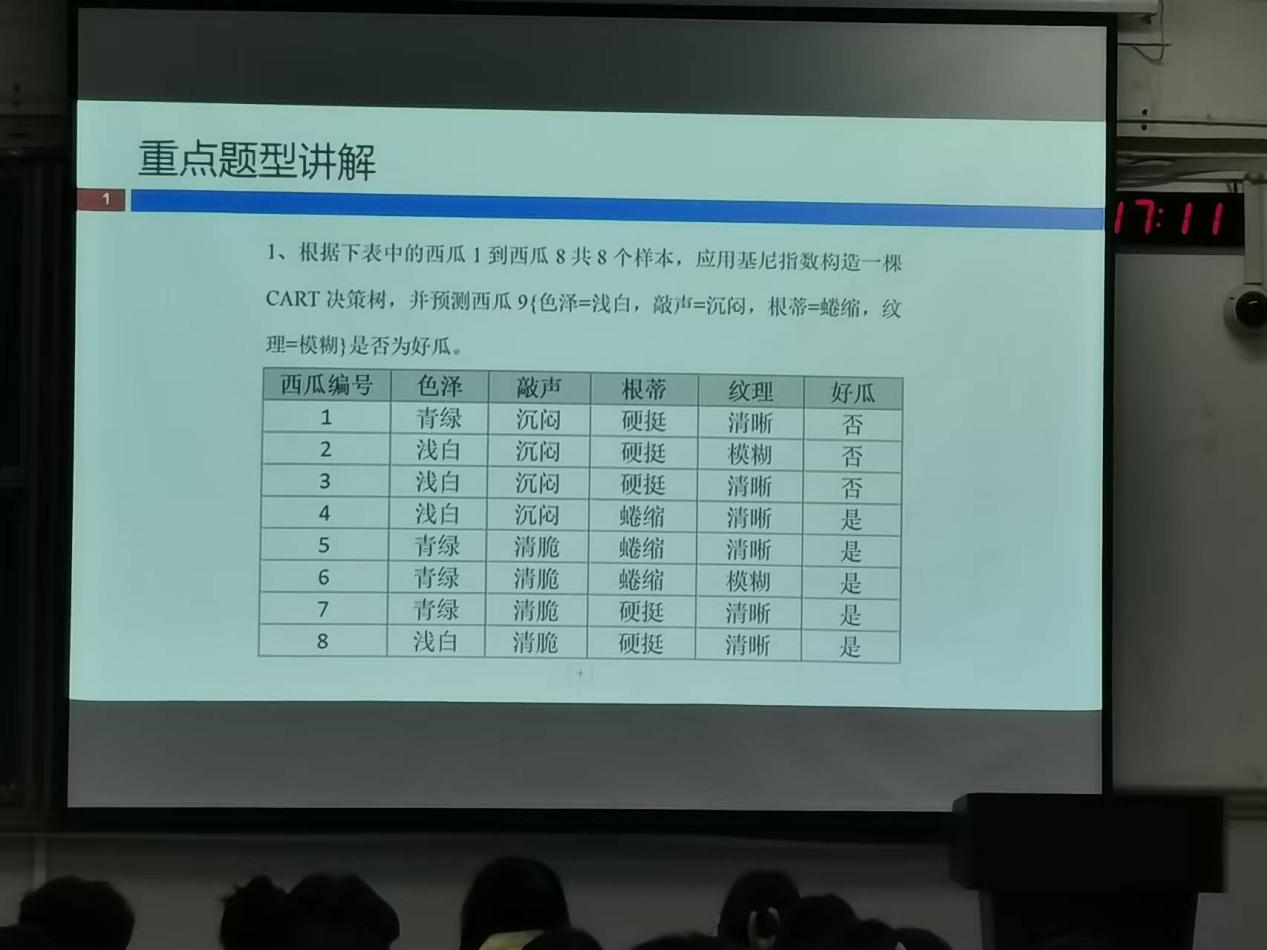
**朴素贝叶斯分类器采用了什么假设简化建模过程？引入该假设主要解决的困难是什么？若某个属性值在训练集中没有与某个类同时出现过，则直接计算该类概率会出现问题，解决的策略是什么？**

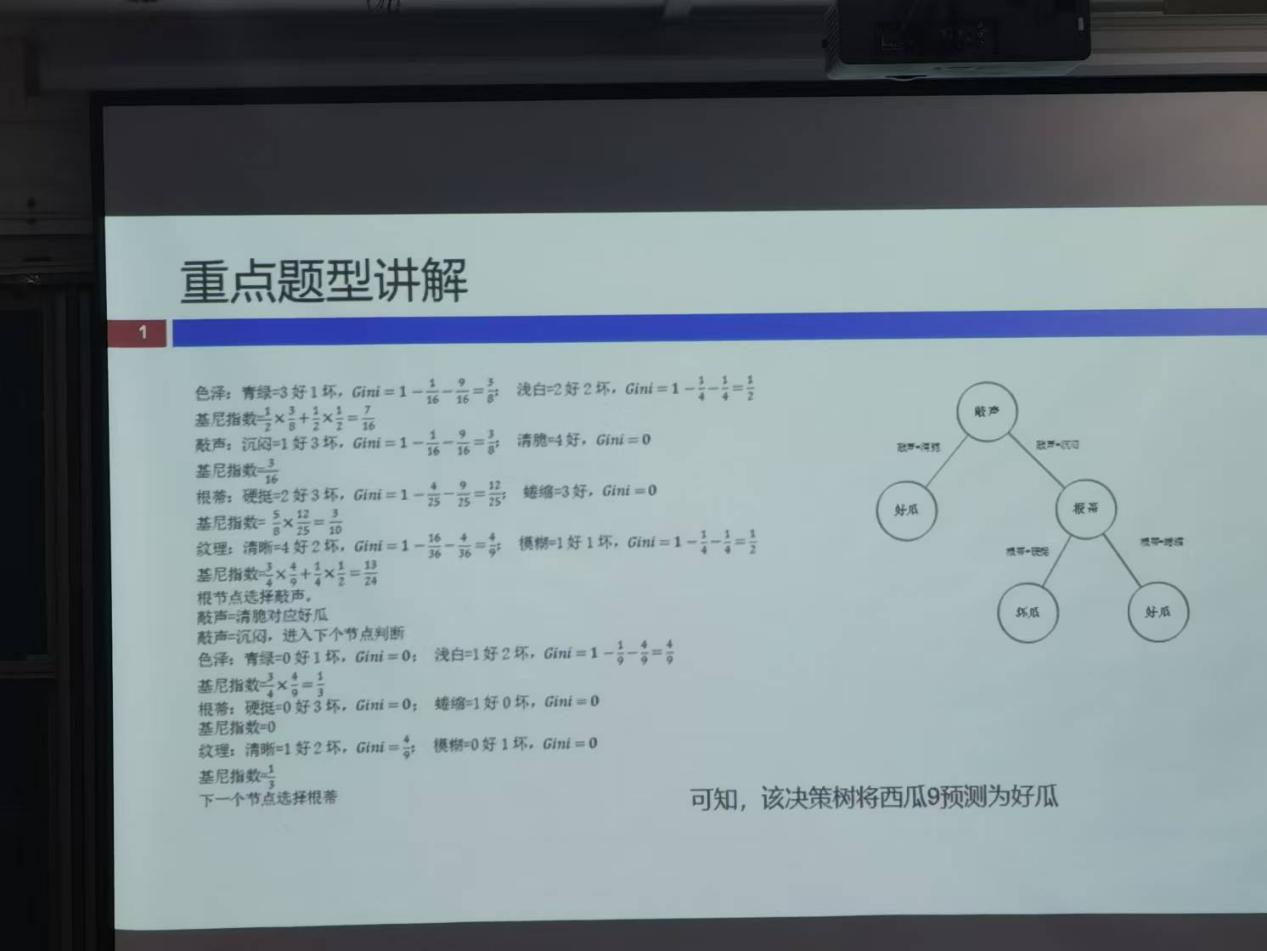
**1.** 朴素贝叶斯分类器采用了**属性条件独立性假设**。该假设认为，给定类别的情况下，所有属性之间是相互独立的。也就是说，每个属性对分类结果的影响是独立的，不受其他属性的影响。

**2.** 引入属性条件独立性假设主要解决了**类条件概率估计的困难**。在现实问题中，类条件概率是所有属性上的联合概率，直接估计这些联合概率需要大量的训练样本，尤其是在属性维度较高时，样本需求呈指数增长，导致难以从有限的训练样本中准确估计这些概率。

**3.** 当某个属性值在训练集中没有与某个类同时出现过时，直接计算类条件概率会出现**零概率问题**，即概率值为零，导致后续的计算无法进行。为了解决这个问题，常用的策略是**拉普拉斯修正（Laplace Smoothing）**。

拉普拉斯修正通过在计算类条件概率时，对每个属性的计数加上一个小的常数（通常为1），从而避免零概率的出现。通过拉普拉斯修正，即使某个属性值在训练集中没有与某个类同时出现过，其类条件概率也不会为零，从而保证了模型的鲁棒性。





敲声

/ \

沉闷 清脆

/ |

根蒂 好瓜

/ \

硬挺 蜷缩

| |

坏瓜 好瓜

**学习通题目**

### 6. (简答题, 15分)机器学习定义是什么？常见的机器学习的类型有哪些？谈谈你对机器学习的理解。

**我的答案：**

1. 机器学习是一门涉及计算机算法的科学，它使得计算机系统能够利用数据来不断改进性能，而无需进行明确的编程。机器学习就是让机器通过学习数据来提高自身性能的过程。

2. 常见的机器学习类型包括：

监督学习：通过训练数据集来学习一个模型，该数据集包含输入和对应的输出标签。

无监督学习：数据集不包含标签，算法需要自己发现数据的结构。

半监督学习：结合了少量的标记数据和大量的未标记数据。

强化学习：通过与环境的交互来学习，目标是最大化累积奖励。

迁移学习：将在一个任务上学到的知识应用到另一个不同但相关的任务上。

### 7. (简答题, 15分)输入空间、输出空间和假设空间之间的关系是怎样的？

**我的答案：**

输入空间：也称为特征空间，是所有可能的输入数据点的集合。

输出空间：是模型预测结果的集合，对于回归任务是连续值，对于分类任务是类别标签。

假设空间：是所有可能的模型的集合，每个模型都是输入空间到输出空间的一个映射。

关系：输入空间和输出空间定义了机器学习问题的范围，而假设空间包含了所有可能的解决方案。机器学习的目标是从假设空间中找到一个最佳的模型，使其能够准确地从输入空间映射到输出空间。

### 8. (简答题, 15分)简述机器学习三要素。

**我的答案：**

数据：用于训练模型的样本数据，质量和数量直接影响模型性能。

模型：通过算法定义的数据结构，用于从数据中学习和预测。

学习算法：优化模型参数的过程，通过调整以最小化损失函数来提高模型性能。

### 9. (简答题, 15分)常见的损失函数有哪些？各有什么特点？

**我的答案：**

均方误差：适用于回归问题，惩罚较大的误差，对异常值敏感。

交叉熵损失：主要用于分类问题，特别是二分类和多分类，衡量预测概率分布与真实分布的差异。

绝对误差：适用于回归问题，对异常值不敏感，相对稳健。

hinge损失：用于支持向量机，主要用于二分类问题，注重间隔的最大化。

0-1损失：只考虑分类是否正确，不关心预测值与真实值之间的差距。

### 10. (简答题, 15分)模型的方差和偏差是如何定义的？模型的期望泛化误差与哪些因素有关，曲线图是怎样的？

**我的答案：**

1. 方差：模型对训练数据变化的敏感程度。高方差模型可能过拟合训练数据。

偏差：模型对真实数据分布的假设程度。高偏差模型可能欠拟合。

2. 模型的期望泛化误差与以下因素有关：数据的数量和质量，模型复杂度，噪声程度。

3. 曲线图通常呈现出U形，随着模型复杂度增加，偏差降低但方差增加，最终导致总的期望泛化误差达到最低值后上升。

### 11. (计算题, 10分)

### 一个模型对15个样本进行预测，结果如下。

### 真实值：0 1 1 0 1 1 0 0 1 0 1 0 1 0 0

### 预测值：1 1 1 1 1 0 0 0 0 0 1 1 1 0 1

### 请分别计算：

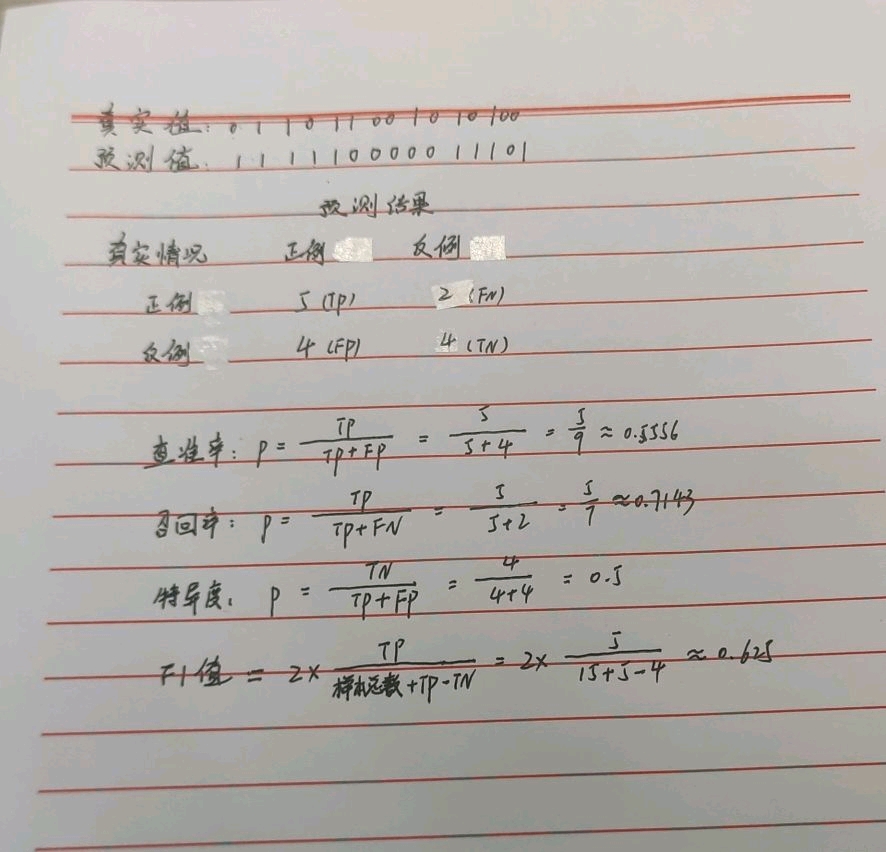
### （1）查准率(precision)

### （2）召回(recall)

### （3）特异度(specificity)

### （4）F1值(F1-score)

**我的答案：**



### 1. (简答题)简述梯度下降算法的原理，并简要叙述梯度下降算法的影响因素。

### 我的答案：

梯度下降算法是一种优化算法，用于求解损失函数最小化的问题。其核心思想是通过迭代的方式，沿着损失函数的负梯度方向逐步更新参数，使得损失函数逐渐减小。

具体的原理如下：

1. 初始化参数：选择初始参数的值。

2. 计算梯度：计算当前参数下损失函数的梯度。

3. 更新参数：根据梯度的反方向进行参数的更新。

4. 重复步骤2和步骤3，直到达到停止条件。

梯度下降算法的影响因素有以下几个：

1. 学习率：学习率决定了每次参数更新的步长。学习率过大会导致参数在损失函数表面跳动，无法收敛；学习率过小会导致收敛速度慢。

2. 初始参数：初始参数的选择对梯度下降的收敛速度和最终结果有一定影响。

3. 批量大小：在批量梯度下降中，每次计算梯度和参数更新时所使用的样本个数。批量大小过大会导致计算量增大，批量大小过小会导致参数更新不稳定。

4. 停止条件：确定何时停止迭代更新参数的条件，可以是达到最大迭代次数或损失函数变化较小等。

### 2. (简答题)分析批梯度下降法、随机梯度下降法和mini-batch梯度下降法的优缺点。

**我的答案：**

1‌.批量梯度下降法

优点‌：保证收敛到全局最优解，特别是在凸优化问题上表现良好。

对梯度的估计是无偏的，样例越多，标准差越低。

在训练过程中使用固定的学习率，不必担心学习率衰退现象的出现。

缺点‌：需要处理整个训练集，计算速度慢，内存消耗大。

当数据集很大时，遍历全部样例需要大量时间。

容易受到冗余样例的影响，导致一些样例对参数更新没有太大作用。

‌2.随机梯度下降法：

优点：计算速度快，每次迭代只使用一个样例。

可以帮助跳出局部最优解，由于随机性的存在，有助于探索更多的解空间。对学习过程添加更多的噪音，提高泛化能力。

‌缺点‌：由于每次只使用一个样例，方差较大，导致损失函数在最小值附近波动较大。

容易陷入局部最优解，而不是全局最优解。

不能使用向量化计算，学习过程较慢。

‌3.Mini-batch梯度下降法

‌优点‌：结合了批量梯度下降和随机梯度下降的优点，计算速度较快。每次迭代使用一部分样例进行参数更新，减少了计算量。增加了并行计算的可能性，提高了计算效率。

缺点‌：在迭代过程中存在噪音，导致学习过程波动较大。

需要调整学习率衰减项以避免过度振荡。

在最小值区域徘徊，可能无法精确收敛到最小值。

### 3. (简答题)感知机是一个线性分类器，是否可以使用感知机来解决非线性问题？

**我的答案：**

不能直接解决。

‌感知机是一个线性分类器，不能直接解决非线性问题。‌ 感知机模型是基于线性可分的数据集设计的，它通过一个超平面来分割两类数据，适用于线性可分的情况。然而，对于非线性可分的数据集，感知机无法直接处理。‌

为了解决非线性问题，可以使用核感知机算法。核感知机算法通过将数据映射到高维空间，使得原本线性不可分的数据在高维空间中变得可分。这种方法利用核函数来计算高维空间中的内积，从而实现了非线性分类。

### 4. (简答题)

### 异或问题可以在二维空间描述为函数f：

### 

### 显然在二维空间中，异或函数f线性不可分，无法直接使用感知机算法进行学习。

### 是否意味着异或函数无法使用感知机算法进行学习？简述理由。

**我的答案：**

异或函数在二维空间中是线性不可分的，这意味着直接使用一个单层感知机是无法正确学习并分类异或问题的。单层感知机的决策边界是一条直线，而异或函数的决策边界是一个非线性形状。

但是这并不意味着异或函数无法使用感知机算法进行学习。虽然单层感知机无法解决异或问题，但多层感知机可以。多层感知机通过引入一个或多个隐藏层，并在这些隐藏层中使用非线性激活函数，可以逼近更复杂的函数关系，包括非线性决策边界。

对于异或问题，可以使用一个包含隐藏层的多层感知机来解决。具体来说，可以设计一个具有两个输入节点、一个隐藏层（包含至少两个隐藏节点）和一个输出节点的神经网络。通过训练这个神经网络，可以使其学习到异或函数的非线性决策边界，并正确地对输入进行分类。

因此，虽然单层感知机无法解决异或问题，但多层感知机可以通过引入非线性激活函数和隐藏层来解决这个问题。

### 5. (简答题)什么是向量的范数？常见的向量的范数有哪些，其物理意义是什么？

**我的答案：**

向量的范数‌是数学中的一种基本概念，用于度量向量空间中向量的长度或大小。范数是一种从向量空间到非负实数的映射，满足非负性、齐次性和三角不等式等条件。常见的向量范数包括1范数、2范数和无穷范数，它们在数学和工程领域有着广泛的应用。

常见的向量范数及其物理意义

‌1范数：

定义：1范数是向量中各个元素绝对值之和。

物理意义：常用于表示向量中元素的和的大小，例如在信号处理中，1范数可以用来衡量稀疏性。

‌2范数：

定义：2范数是向量各个元素的平方和的平方根。

物理意义：表示向量的欧几里得长度，常用于计算两点之间的距离。

‌无穷范数：

定义：无穷范数是向量中各个元素绝对值的最大值。

物理意义：常用于衡量向量中元素的最大变化量，例如在控制系统设计中，无穷范数可以用来评估系统的稳定性。

### 1. (简答题, 10分)针对几何间隔和函数间隔，硬间隔和软间隔，谈谈你的理解。

**我的答案：**

‌函数间隔和‌几何间隔的定义和区别‌：

‌函数间隔‌是指数据点到‌超平面的线性距离的绝对值。在支持向量机（‌SVM）中，函数间隔被定义为由超平面到数据点的最短距离，且与超平面方程有关的量。这个距离是通过数据点代入超平面方程得到的，其数值大小与超平面参数的缩放有关。如果将超平面方程的‌法向量和截距项等比例放大或缩小，函数间隔也会相应地变大或变小，因此函数间隔的大小是可以任意调整的，并不是一个固定不变的值。

‌几何间隔‌是指数据点到超平面的最小距离，也就是垂直于超平面的最短距离。它是通过数据点与超平面上任意一点连线，并与超平面的法向量做点积后，除以法向量的模长（二范数）来计算得到的。由于涉及法向量模长的计算，几何间隔是一个固定的值，不会随着超平面方程参数的等比例缩放而改变。在SVM中，最大化几何间隔可以确保找到最优的分割超平面。

‌硬间隔和软间隔的定义和区别：

‌硬间隔‌是指在线性可分的情况下，训练数据可以被一个超平面完美地分开，没有任何错误。在这种情况下，找到的超平面使得不同类别的数据点之间的距离最大化，这个距离就是硬间隔。硬间隔支持向量机通过最大化硬间隔来学习一个分类器，确保所有训练数据点都被正确分类，且分类边界最远。

‌软间隔‌是指在线性不可分的情况下，允许一定的错误分类。软间隔支持向量机通过引入松弛变量来允许一些数据点被错误分类，同时最大化间隔。这种方法比硬间隔更灵活，能够处理更复杂的数据分布。软间隔支持向量机通过最小化目标函数和约束条件来找到最优的超平面。

### 2. (简答题, 10分)SVM是如何解决线性不可分问题的？请以异或函数为例来阐明你的结论。

**我的答案：**

SVM通过使用‌核函数将低维空间中的线性不可分数据映射到高维空间中，使其变得线性可分。核函数的作用是将原始数据映射到一个更高维度的空间，在这个新空间中，数据变得线性可分，从而可以找到一个最优的超平面进行分类。常用的核函数包括线性核、‌多项式核、‌高斯核和‌Sigmoid核等。

异或函数是一个典型的线性不可分问题。异或函数的输出取决于两个输入的不同之处，即如果两个输入不同，则输出为1；如果两个输入相同，则输出为0。在二维空间中，异或函数无法用一条直线进行划分，因此是线性不可分的。通过使用SVM和高斯核函数，可以将这个问题映射到高维空间中，使其变得线性可分，从而找到一个合适的超平面进行分类。

### 3. (简答题, 10分)线性回归中的岭回归模型应用在什么场合？正则化参数对模型有何影响？

**我的答案：**

‌岭回归模型的应用场合：岭回归模型主要应用于存在多重共线性数据的场合‌。岭回归通过添加一个‌惩罚项来防止过拟合，特别适用于那些自变量之间存在高度相关性的情况。此外，岭回归还适用于那些模型预测的准确性比无偏估计更为重要的场景。

‌正则化参数对模型的影响：正则化参数λ在岭回归模型中起着至关重要的作用‌。它决定了惩罚项的强度，进而影响模型的‌复杂度和泛化能力。当λ较小时，惩罚项的作用较弱，模型复杂度较高，可能导致过拟合；而当λ较大时，惩罚项的作用较强，模型复杂度降低，可能导致欠拟合。因此，选择合适的λ值对于岭回归模型的性能至关重要。

### 4. (简答题, 10分)Softmax回归模型用在什么场合？Softmax回归模型一定比多个逻辑回归模型更合适么？为什么？

**我的答案：**

‌Softmax回归模型适用于多分类问题‌，当任务涉及多个类别时，使用Softmax回归模型更为合适。例如，在图像识别任务中，模型需要预测图像属于某个特定类别的概率，Softmax函数可以帮助模型输出每个类别的概率，从而选择概率最高的类别作为预测结果‌。

在图像识别任务中，Softmax函数可以帮助模型预测图像属于某个特定类别的概率。例如，一个训练好的模型可以识别照片中的猫、狗或其他物体，并给出每种类别的概率‌。

在NLP任务中，Softmax函数可以用于语言模型预测下一个单词的概率分布。例如，在文本生成或机器翻译任务中，模型会根据当前的上下文预测下一个单词，并输出一个概率分布‌。

‌在推荐系统中，Softmax函数可以用来预测用户对不同项目（如电影、音乐或商品）的兴趣度，进而推荐用户可能喜欢的内容‌。

在强化学习中，Softmax函数可以用于计算代理在给定状态下采取不同行动的概率，帮助在探索和利用之间找到平衡点‌。

Softmax回归模型与多个逻辑回归模型的比较不一定‌更合适。虽然Softmax回归模型适用于多分类问题，但在某些情况下，使用多个逻辑回归模型可能更为合适。这取决于具体问题的复杂性和数据的特点。

对于涉及多个类别的分类问题，Softmax回归模型能够输出每个类别的概率，适合于需要同时考虑多个类别的场景‌。

对于只有两个类别的分类问题，逻辑回归模型更为简单和高效。逻辑回归适用于二分类问题，通过sigmoid函数将输出限制在0和1之间，适合于二分类任务‌。

Softmax回归模型在处理多分类问题时需要计算所有类别的概率，计算量较大；而多个逻辑回归模型分别处理每个类别，计算量相对较小。因此，在选择模型时需要考虑计算资源和数据特点‌。

### 5. (简答题, 10分)

### 对于多分类问题，Softmax回归模型是否一定比Logistic回归模型（分解成多个二分类任务）更合适？

**我的答案：**

对于多分类问题，Softmax回归模型和Logistic回归模型（分解成多个二分类任务）各有优缺点，选择哪种模型取决于具体的应用场景和数据特性。‌

Softmax回归模型适用于多分类问题，其输出层包含多个神经元，每个神经元对应一个类别。Softmax函数将输出层的值转换为概率分布，使得每个类别的概率之和为1。这种模型在类别互斥的情况下表现良好，即每个样本只属于一个类别。例如，在MNIST手写数字识别中，每个数字是一个独立的类别，因此使用Softmax回归模型是合适的‌。

Logistic回归模型分解为多个二分类任务。当类别之间不是互斥的，即一个样本可能同时属于多个类别时，将Logistic回归模型分解为多个二分类任务更为合适。例如，在音乐分类中，一首歌曲可能同时属于人声音乐、舞曲、影视原声等多个类别。在这种情况下，使用多个独立的Logistic回归分类器可以更好地处理这种情况‌1。

‌Softmax回归模型‌：适用于类别互斥的多分类问题，如MNIST手写数字识别、图像分类等。

‌Logistic回归模型分解为多个二分类任务‌：适用于类别非互斥的情况，如音乐分类、文本分类等，其中样本可能同时属于多个类别。

### 6. (简答题, 10分)简述SVM与Logistic回归模型的区别，并其中最本质的区别。

### 我的答案：

SVM和Logistic回归都是用于二分类的模型，但它们在寻找‌最优超平面、‌损失函数、处理‌非线性情况等方面存在区别。SVM通过寻找一个超平面，使得不同类别的样本之间的间隔最大化，而Logistic回归则通过Sigmoid函数将线性回归的输出映射到(0,1)区间，用于分类。SVM可以处理非线性情况，而Logistic回归通常用于线性分类问题。

SVM与Logistic回归模型最本质的区别：SVM与Logistic回归模型最本质的区别在于它们的损失函数不同。SVM使用‌hinge loss，而Logistic回归使用‌cross entropy loss。这两种损失函数的目的都是增加对分类影响较大的数据点的权重，减少与分类关系较小的数据点的权重。SVM通过只考虑支持向量来简化模型和计算，而Logistic回归则考虑所有样本点。

### 1. (简答题, 30分)深度神经网络中常见的激活函数有哪些，各有什么优缺点？

**我的答案：**

深度神经网络中常见的激活函数包括Sigmoid、Tanh、ReLU、LeakyReLU、ELU和MaxOut，每种激活函数都有其优缺点‌：‌

‌Sigmoid函数‌：

‌优点‌：输出范围在0到1之间，适合二分类问题。

‌缺点‌：存在梯度消失问题，输出不以0为中心，计算速度较慢。

‌Tanh函数‌：

‌优点‌：输出范围在-1到1之间，以0为中心，适合二分类问题。

‌缺点‌：虽然解决了Sigmoid的输出不以0为中心的问题，但仍然存在梯度消失问题。

‌ReLU函数‌：

‌优点‌：计算速度快，导数在正半轴为常数1，有助于加速反向传播算法的学习速度。

‌缺点‌：在负半轴输出为0，可能导致神经元死亡（Dead ReLU问题），且输出不以0为中心。

‌LeakyReLU函数‌：

‌优点‌：在负半轴有一个小的线性分量，避免了Dead ReLU问题，保留了ReLU的所有优点。

‌缺点‌：虽然理论上解决了ReLU的问题，但在实践中尚未完全证明其总是优于ReLU。

‌ELU函数‌：

‌优点‌：输出以0为中心，解决了ReLU的神经元死亡问题。

‌缺点‌：计算复杂，需要计算指数。

‌MaxOut函数‌：

‌优点‌：不会造成梯度消失，拟合能力强。

‌缺点‌：引入了更多的参数，容易引起过拟合现象。

### 2. (简答题, 30分)

### 梯度消失问题产生的原因是什么？如何应对深度神经网络中的梯度消失问题？

**我的答案：**

梯度消失问题产生的主要原因包括以下几个方面‌：‌

‌激活函数的选择‌：某些激活函数（如sigmoid、tanh）在输入较大或较小的情况下，梯度会非常接近于零，从而导致梯度消失。

‌权重初始化‌：如果网络的权重初始化过大或过小，也会导致梯度消失问题。

‌深层网络结构‌：深层网络中，梯度需要通过多个层传播，每一层都会引入一定的误差，这些误差会累积导致梯度消失。

‌应对深度神经网络中的梯度消失问题的策略包括‌：

‌使用残差网络（ResNet）‌：残差块的输出为H(x)+F(x)，即使F(x)的梯度为0，H(x)的梯度仍然可以通过跨层连接传递到前面的层，从而实现了更好的梯度流动。

‌使用批归一化‌：批归一化用来解决梯度消失和梯度爆炸问题，通过规范化层的输入来加速训练过程并减少初始化敏感度。

‌使用LSTM或GRU‌：对于循环神经网络，LSTM和GRU通过门控机制来避免梯度消失，如遗忘门、输入门和输出门来控制信息的遗忘、存储和输出。

‌适当的学习率和自适应学习率调整‌：选择合适的学习率以及使用自适应学习率调整方法，如Adam优化器，根据参数的梯度自适应调整每个参数的学习率。

‌深度监督‌：通过在网络的多个层次上添加监督信号，可以确保梯度在每个层次上都能有效地传播。

### 3. (简答题, 40分)

### 观看教学视频，阅读相关资料，进行反向传播算法的推导。并以此为基础，简述反向传播算法。

**我的答案：**

1. 前向传播：计算每层的输出，直到得到最终预测。
2. 计算损失：使用损失函数评估预测与真实值 之间的差异。
3. 反向传播：从输出层开始，递归地计算损失 函数关于每个参数的梯度。
4. 参数更新：使用梯度下降法更新网络的权重 和偏置。 这个过程在每个训练样本上重复进行，真到网 络在训练集上的性能达到满意的水平。

### 1. (简答题, 15分)简述进行概率模型参数估计的统计学界不同学派的观点。

**我的答案：**

主要有三种不同的学派观点：频率学派、贝叶斯学派和经验贝叶斯学派。

（1）频率学派：

认为模型参数是固定但未知的常数通过最大似然估计和矩估计等方法进行参数估计强调重复抽样，参数估计的好坏通过抽样分布和渐近性质来评判强调点估计和置信区间，不直接赋予参数概率分布代表性方法包括最小二乘法和最大似然估计。

（2）贝叶斯学派：

将参数视为随机变量，具有先验概率分布通过观测数据和先验信息，使用贝叶斯公式计算参数的后验分布估计结果不仅包括点估计，还包括整个后验概率分布允许融入先验知识，特别适合小样本和不确定性较高的情况代表性方法包括马尔可夫链蒙特卡洛和变分推断。

（3）经验贝叶斯学派：

介于频率学派和贝叶斯学派之间的折中方法从数据本身估计先验分布的参数，而非完全依赖主观先验知识通过数据经验推断先验分布，然后进行贝叶斯推断计算复杂度低于传统贝叶斯方法适用于大规模数据和复杂模型的参数估计

### 2. (简答题, 15分)决策树如何选择最优划分属性？请结合ID3算法、C4.5算法或CART算法加以说明。

**我的答案：**

决策树是一种监督学习方法，它通过构建一个树状模型来进行分类或回归预测。选择最优划分属性是构建决策树的关键步骤之一，不同的算法有不同的标准来衡量“最优”。

ID3 算法使用信息增益作为选择最优划分属性的度量标准。信息增益基于香农熵，ID3 算法选择具有最高信息增益的属性作为当前节点的最佳划分属性。

C4.5 是 ID3 的改进版本，除了可以处理数值型数据和缺失值外，它还引入了信息增益比来选择最优划分属性，以减少对具有大量值的属性的偏好。C4.5 算法选择信息增益比最高的属性作为最佳划分属性。

CART 算法既可以用于分类也可以用于回归问题。对于分类树，CART 使用基尼指数来选择最优划分属性；对于回归树，则使用方差或均方误差等指标。当考虑属性 aa 来进行划分时，CART 计算每个可能划分后的加权基尼指数，并选择使加权基尼指数最小化的属性作为最佳划分属性。对于二元分类问题，CART 总是创建二叉树，即每次划分将数据集分成两个子集。

故ID3、C4.5 和 CART 都有各自的选择最优划分属性的标准，这些标准帮助它们构建出有效的决策树模型。

### 3. (简答题, 15分)

### 简要叙述下集成学习中的Bagging与Boosting的算法思想。与Bagging集成策略相比，Boosting集成策略有哪些优点和缺陷？

**我的答案：**

集成学习通过组合多个模型的预测来改善机器学习的效果。Bagging 和 Boosting 是两种主要的集成策略，它们有不同的算法思想。

Bagging的核心思想是创建多个独立且并行的模型（通常是决策树），然后通过投票（分类任务）或平均（回归任务）的方式组合这些模型的预测结果。每个模型都是基于原始数据集的随机抽样（有放回抽样）构建的，这种技术称为自助法（Bootstrap）。因为每个模型的数据集是独立抽取的，所以这些模型之间具有较低的相关性，这有助于减少方差，提高模型的稳定性。

一个典型的Bagging算法是随机森林，它不仅对样本进行随机抽样，还对特征进行随机选择，以进一步增加模型之间的差异性。

Boosting 的核心思想是逐步构建一系列模型，其中每个新模型都试图纠正前一个模型的错误。Boosting 算法通过赋予被先前模型错误分类的样本更高的权重，使得后续模型更加关注这些难分类的样本。因此，Boosting 模型是顺序依赖的，即后一个模型依赖于前一个模型的结果。

Boosting与Bagging相比优点：

准确性：Boosting通常能够生成更准确的模型，因为它专注于修正之前模型的错误。

偏差降低：Boosting可以有效地减少模型的偏差，而不仅仅是方差。

适应性强：由于其迭代性质，Boosting对于处理不平衡数据集可能更为有效。

Boosting 与 Bagging 相比的缺点：

计算成本：Boosting算法的训练过程是串行的，这意味着不能并行化，导致训练时间较长，特别是在大型数据集上。

过拟合风险：由于Boosting不断尝试纠正错误，如果训练得当可能会导致模型过度拟合训练数据，尤其是在噪声较多的数据集上。

敏感性：Boosting算法对异常值和噪音更加敏感，因为它们会获得更高的权重，并可能影响后续模型的训练。

总的来说，Bagging 和 Boosting 都有各自的优势和适用场景。选择哪种方法取决于具体的问题以及数据的特点。

## 二. 名词解释（共2题，20分）

### 4. (名词解释, 10分)信息熵

**我的答案：**

信息熵是信息论中用于量化信息源各可能事件发生的不确定性的概念，表示信息源发出信息前的平均不确定性。信息熵越高，代表信息的不确定性越大。

信息熵是衡量信息源发出信息前平均不确定性的一个量化指标。信息熵越高，代表信息的不确定性越大，所需的信息量也就越大，反之则表明信息的不确定性较小，所需信息量相对较少。这一概念为信息的量化度量提供了科学依据，是信息论发展的基石之一。

### 5. (名词解释, 10分)拉普拉斯修正（Laplacian Correction）

**我的答案：**

拉普拉斯修正是一种在统计和概率估计中用于处理零概率问题的方法。

拉普拉斯修正，也称为加一平滑或拉普拉斯平滑，是一种对概率估计进行修正的技术。它主要用于解决在概率计算中出现零概率的情况。当我们在统计样本中遇到某些事件从未发生过时，直接计算其概率会得到零，这可能会导致后续概率计算的不准确。拉普拉斯修正通过在每个可能的类别或事件中人为地增加一个伪计数来解决这个问题。

## 三. 计算题（共1题，35分）

### 6. (计算题, 35分)

### 假设所有弱分类器均由x<v或x>v产生，其中阈值v使分类器在训练数据集上分类误差率最低。针对如下二分类问题Y∈{1,-1}的训练数据集D：

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| x | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| y | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | -1 |

### 

### 

### 初始的样本数据权值分布D1完全相等。使用AdaBoost算法过程中，函数log定义为以e为底的自然对数ln，现已得到如下若干弱分类器：

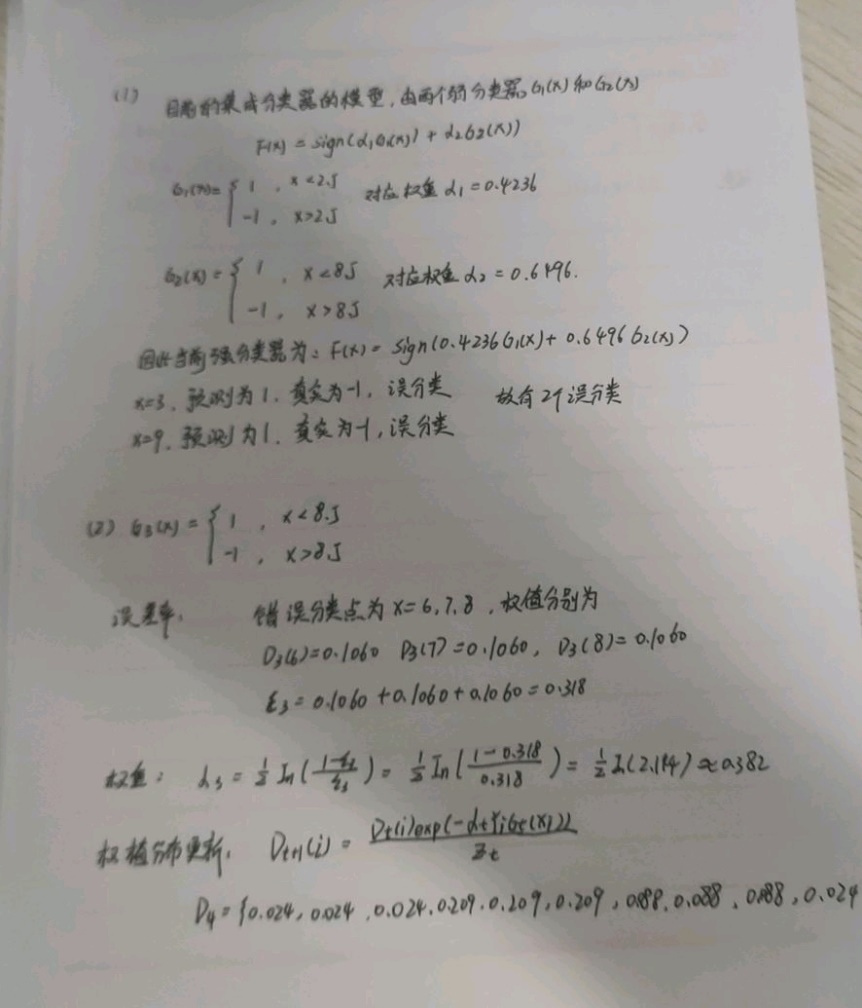
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 步骤 | 基本分类器 Gi | 分类器误差率 | 分类器权重 | 更新后的数据权值分布 |
| 1 |  | 0.3 | 0.4236 | D2={0.07143,0.07143,0.07143,0.07143,0.07143,0.07143,0.16667,0.16667,0.16667,0.07143} |
| 2 |  | 0.2143 | 0.6496 | D3={0.0455,   0.0455, 0.0455,0.1667,0.1667,0.1667,0.1060,0.1060,   0.1060,0.0455} |

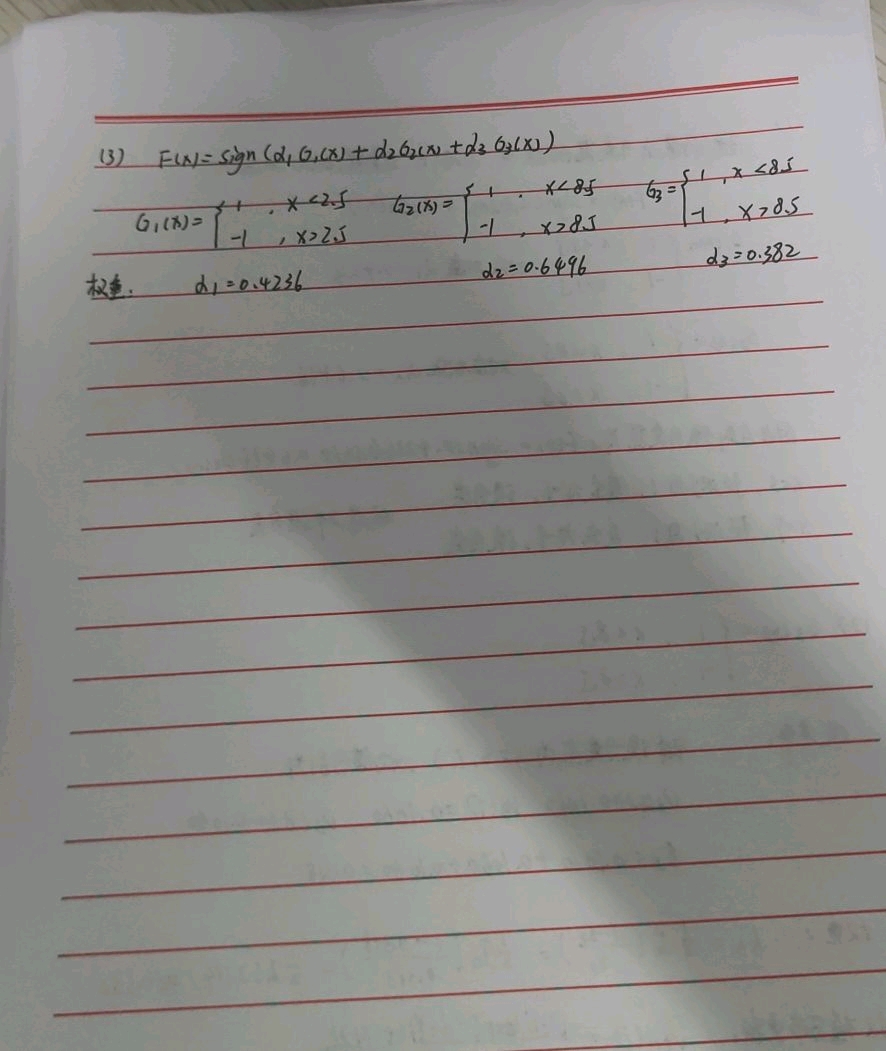
### （1）    目前的集成分类器的模型是什么？误分类点有几个？

### （2）    在权值分布D3上，基本分类器G3的模型是什么？计算G3的误差率、权重以及更新后的数据权值分布D4。

### （3）    给出最终的强分类器模型。

**我的答案：**





### 1. (简答题)简述流形学习中测地线距离的概念及计算方法。

**我的答案：**

在流形学习中，假设数据分布在一个低维的流形上，而测地线距离是指在这个流形上两个点之间的最短路径长度。与欧氏距离不同，测地线距离能够更准确地反映数据点在流形上的相对距离，因此在处理具有非线性结构的数据时尤为重要。

测地线是在流形上连接两点的最短路径。

测地线距离是测地线的长度，反映了数据点沿流形表面的真实距离。

测地线距离对于非线性分布的数据非常重要，因为直接使用欧氏距离可能忽略数据的流形结构，从而导致学习效果较差。

由于流形通常是隐式定义的，测地线距离的直接计算可能困难。因此，常用的近似计算方法包括以下步骤：

1. 构建邻接图

使用 K-近邻（k-NN）或 ε-邻域方法，基于欧氏距离将数据点连接成图。图中的边权重为点与点之间的欧氏距离。

2. 估计局部测地线距离

邻接点之间的欧氏距离可以近似为局部测地线距离。

3. 全局测地线距离的计算

使用最短路径算法计算图中任意两点之间的最短路径长度。最短路径长度即为两点之间的测地线距离的近似值。

### 2. (简答题)分析L1和L2范数正则化的优缺点。

**我的答案：**

L1 范数正则化优点：

（1）稀疏性特性：

L1 正则化倾向于使部分参数wi收缩为 0，从而实现特征选择，得到一个稀疏模型。

对于高维数据（如文本分类或基因数据），L1 正则化可以有效降低特征数量。

（2）易于解释：

稀疏模型中的非零特征较少，更容易解释模型的结果。

（3）适用于特征选择：

能自动选择对模型影响较大的特征，忽略不重要的特征，从而减少维度。

缺点：

（1）优化困难：

由于 ∣wi∣在 0 处不可导，L1 正则化会导致优化问题的非光滑性，需要特殊的优化算法（如坐标下降法）。

（2）处理相关特征时表现较差：

L1 正则化在处理高度相关的特征时，可能随机选择一个特征而忽略其他特征，导致性能下降。

L2 范数正则化优点

（1）收敛稳定性：

L2 正则化引入了二次惩罚项，导致损失函数光滑且凸，优化更加稳定，梯度下降容易收敛。

（2）适合处理多重共线性问题：

对于高度相关的特征，L2 正则化会平滑地减小权重，而不是完全忽略某些特征，从而保留更多信息。

（3）避免过拟合：

L2 正则化通过减小权重的大小，可以有效防止模型过拟合。

缺点：

（1）不具备稀疏性：

L2 正则化会让所有权重趋近于 0，但不会完全为 0，因此不能用于特征选择。

（2）难以解释：

因为所有特征都有非零权重，模型的解释性较弱。