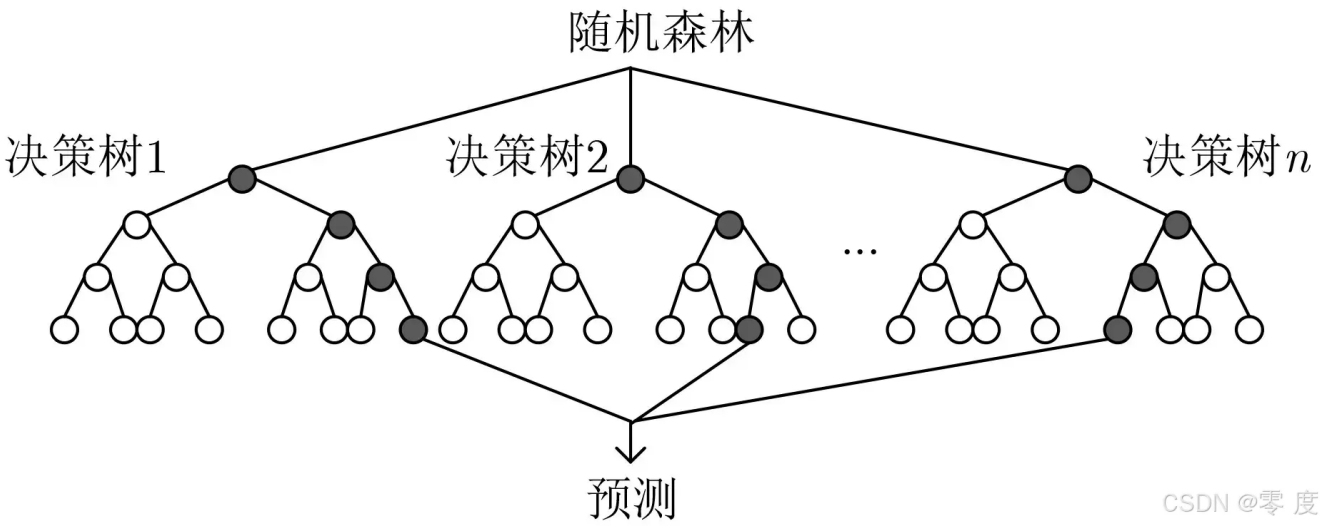
03-随机深林（1天）

https://blog.csdn.net/zdx2585503940/article/details/146241822　　随机森林（Random Forest）

1、概述



1.1、介绍：

- 随机深林在决策树的基础上引入了Bagging思想，通过构建多个决策树来进行实现分类或回归预测，不过它的每棵树的构建过程都引入了随机性：一方面通过有放回地随机抽取原始训练集中的样本，构造多个不同的子数据集，对每个子集训练一个独立的决策树，最终的预测结果是通过将所有模型的预测结果进行结合（通常是投票或平均）来得到的。另一方面它的

特征选择也是随机的，就是在每棵树的每个节点分裂时，随机选择一部分特征进行决策，而不是使用所有的特征。这样可以提高模型的泛化能力，减少过拟合的风险。

- 例外，因为每个模型的训练数据集是从原始数据集中通过**有放回抽样**得到的，这样每个模型的训练数据集可能包含重复样本。训练的多个模型是**并行**的，即它们的训练过程是相互独立的。

1.2、Bagging（套袋法）：

- **随机选择特征**：在每棵树的每个节点分裂时，随机选择一部分特征进行决策，而不是使用所有的特征。这种方法称为**特征随机性（feature randomness）**，能够提高模型的多样性，减少过拟合。
- **构建多棵决策树**：随机森林使用Bagging机制来构建多棵决策树，每棵决策树在训练时都使用不同的Bootstrap样本，并且每次分裂节点时仅使用部分特征。这使得随机森林的模型变得更加健壮。
- **模型集成**：每一棵决策树会独立地进行预测，最后通过集成所有决策树的预测结果来进行最终的分类或回归。对于分类任务，通常采用多数投票机制（majority voting），对于回归任务，通常采用平均值（mean）。

1.3、随机森林的Bagging机制流程：

- **数据集的Bootstrap抽样：**

从原始训练数据集中通过有放回抽样的方式，随机选择若干个样本组成一个新的数据集（Bootstrap样本）。每个样本数据集的大小通常与原始训练集相同，但是由于有放回抽样，某些样本可能会重复出现，而某些样本则不会出现在该子数据集中。

假设原始训练集包含5个样本：`[A, B, C, D, E]`。那么通过Bootstrap抽样得到的一个新的子数据集可能是：

  - `[A, C, C, D, E]`，即`C`重复出现，而`B`没有被选中。
  - 或者`[B, B, D, A, E]`，即`B`重复，其他样本被选中一次。

因此，**每个子数据集的大小与原始数据集的大小相同，但它的内容是随机的，且样本可以重复出现。**

- **训练多个决策树**：

在每个Bootstrap样本上训练一棵决策树。训练过程中，每棵树会选择一个随机的特征子集进行分裂，而不是使用全部的特征。这种做法有助于提

高多样性，减少单棵树的过拟合。

假设你有一个包含5个特征的数据集：`Feature1, Feature2, Feature3, Feature4, Feature5`。

- **随机选择特征子集**：

假设选择3个特征进行分裂，即m=3。那决策树在每个节点分裂时，会随机选择3个特征。例如，随机选择的特征可能是：`Feature2, Feature 4, Feature5`。

- **选择最佳分裂特征**：

然后，在这3个特征中选择最能提供信息增益或基尼指数最小的特征，作为该节点的分裂特征。例如，假设`Feature5`能提供最高的信息增益，那

么决策树就会在`Feature5`上进行分裂。

- **模型集成**：

每一棵决策树会独立地进行预测，最后通过集成所有决策树的预测结果来进行最终的分类或回归。对于分类任务，通常采用多数投票机制

（majority voting），对于回归任务，通常采用平均值（mean）。

1.4、随机深林的优缺点

- **优点：**
  - **减少过拟合**：通过集成多个模型，Bagging能有效减少模型的方差，尤其是对于高方差模型（如决策树）表现尤为明显。
  - **提高模型稳定性**：由于Bagging方法是基于多个独立训练的模型，其输出结果更加稳定，不容易受到单一模型误差的影响。
  - **适应高维数据**：Bagging适合处理维度较高的数据，尤其是在特征空间维度较大的情况下（如文本分类、图像分类等）。
  - **并行处理**：由于每棵树的构建是相互独立的，随机森林可以很容易地实现并行化，提高训练效率。
  - **高准确性**：通过构建多个决策树并将结果进行投票或平均，随机森林能够提高模型的

整体准确性。

- **缺点：**
  - **计算开销大**（训练时间长，内存消耗大）：由于训练多个模型需要消耗较多的计算资源，Bagging方法的训练时间和内存开销较大，尤其是模型较复杂时。
  - **模型复杂度较高**：Bagging构建了多个模型进行集成，虽然增加了模型的准确性，但也增加了模型的复杂度，可能导致模型的可解释性降低。
  - **模型解释性**：相比于单一决策树，随机森林作为集成模型，其内部工作机制不够透明，模型解释性较差。
  - **超参数选择**：随机森林的性能在一定程度上依赖于超参数的选择，如树的数量、分裂时考虑的特征数等，这些参数的调整可能需要大量的实验。

1.5、随机森林的调优策略

- 参数选择

参数调优是提高随机森林模型性能的关键步骤。在随机森林中，几个关键的参数包括树的数量、分裂时考虑的特征数、最大深度等。

  - **树的数量**：增加树的数量可以提高模型的稳定性和准确性，但同时会增加计算成本。一般而言，树的数量在几十到几百之间即可满足大多数需求。
  - **特征数**：在每个决策树的节点分裂时考虑的特征数量，通常设置为总特征数量的平方根，可以有效地增加模型的多样性，避免过拟合。
  - **最大深度**：限制树的最大深度可以防止模型过于复杂，但可能影响模型的表达能力。需要根据具体问题调整。

- 避免过拟合

尽管随机森林本身具有较好的抗过拟合能力，但在某些情况下，模型仍然可能出现过拟合现象。

  - 增加树的数量：更多的树可以提供更全面的预测视角，减少过拟合的风险。
  - 减少特征数：减少在分裂时考虑的特征数可以降低模型的复杂度，避免对训练数据的过度拟合。
  - 剪枝：虽然随机森林通常不剪枝，但在必要时可以通过设置最大深度或最小样本数来限制树的生长。

- 样本抽样：通过自助采样法（bootstrap sampling）抽取样本时，可以引入更多的随机性，减少过拟合。

使用交叉验证来评估不同参数设置下模型的性能，是一种有效的调优方法。通过比较不同参数组合的交叉验证分数，可以找到最优的参数配置。此

外，还可以利用网格搜索（Grid Search）或随机搜索（Random Search）等策略来自动化参数选择过程。

1） 为什么要随机抽样训练集?

如果不进行随机抽样，每棵树的训练集都一样，那么最终训练出的树分类结果也是完全一样的，这样的话完全没有bagging的必要；

2）为什么要有放回地抽样?

如果不是有放回的抽样，那么每棵树的训练样本都是不同的，都是没有交集的，这样每棵树都是"有偏的"，都是绝对"片面的"（当然这样说可能不对），也就是说每棵树训练出来都是有很大的差异的；而随机森林最后分类取决于多棵树（弱分类器）的投票表决，这种表决应该是"求同"，因此使用完全不同的训练集来训练每棵树这样对最终分类结果是没有帮助的，这样无异于是"盲人摸象"。