附录：个人贡献声明（Contribution Statement)

为了确保小组成员在项目中的贡献被准确记录，并促进任务的公平分工与高效合作，我们小组制定了详细的任务分解计划，并在实际完成过程中持续沟通、互相支持。以下为本项目中每位成员的个人贡献说明：

傅祉珏同学

主要任务

- 实现 MLP（多层感知机）模型与 K-means 聚类算法；
- 对六种算法的接口进行统一封装；
- 编写模型融合（Stacking 集成学习）部分的主程序逻辑。

具体贡献

- 实现了支持非线性分类的 MLP 模型，使用 ReLU 激活与 Softmax 输出；
- 编写了 K-means 聚类算法，支持多种初始中心选取策略，并加入聚类结果的后处理逻辑；
- 对各个模型训练与预测接口进行统一封装，确保输入输出格式一致，便于后续融合；
- 设计并实现了基于概率输出拼接的两层 Stacking 架构，构建了基模型与元模型之间的连接逻辑。

遇到的挑战与解决方案

在完成模型融合部分的过程中，我面临了一个较为棘手的问题：不同模型的输出格式并不统一，尤其是在使用 `predict_proba` 方法时，有些模型返回的是一维向量，有些是二维概率分布，这种不一致导致了 Stacking 阶段输入特征维度混乱，训练时频繁报错。为了彻底解决这个问题，我从根本上重新设计并统一了所有模型的 `predict_proba` 接口，并通过显式的 `assert` 语句对输出维度进行严格校验，确保各模型输出可直接拼接进入元模型。此外，在实现 MLP 模型时，由于训练数据量相对有限，模型在早期训练阶段出现了明显的过拟合现象。为此，我引入了 Dropout 层来增加模型的泛化能力，同时设置早停机制，在验证集性能下降时及时终止训练，从而有效避免了过度拟合的问题。

心得体会

通过本次项目，我更加深入地理解了多层感知机和聚类模型在教育数据分析中的应用价值，也从工程实践中体会到“接口一致性”在系统集成中的重要性。在 Stacking 架构设计中，我第一次真正体会到集成学习的魅力：通过组合多个异构模型的概率预测结果，模型在整体性能上得到了显著提升。整个开发过程不仅提升了我的编程规范意识和模块化设计能力，也增强了我独立分析问题与调试复杂系统的信心。

杨程骏同学

主要任务

- 实现逻辑斯谛回归与决策树分类器；
- 负责数据清洗、编码、归一化等预处理工作；
- 构建并维护可加载的数据集结构，供所有模型调用。

具体贡献

- 使用 Numpy 编写逻辑回归模型，支持 Softmax 多分类与交叉熵损失；
- 实现了基于 CART 思想的决策树模型，使用基尼指数作为划分准则；
- 编写数据预处理脚本，完成了缺失值处理、类别编码、特征标准化等流程；
- 封装了 PyTorch Dataset 类接口，使模型能够批量加载并复现训练数据。

遇到的挑战与解决方案

在进行数据预处理与模型构建过程中，我遇到的主要挑战来自数据本身的复杂性。原始数据中存在大量缺失值、异常格式（如字符串混杂数字）、类别特征分布不均等问题。起初我尝试手动处理，但效率极低、易出错。后来我将整个预处理流程模块化，设计了一条标准化的数据流水线，每一步都封装成可独立调用的 `transform` 函数，从而提高了可复用性和可维护性。在实现逻辑回归和决策树模型的过程中，我还遇到过一个细节性的 bug：由于浮点数精度问题，决策树划分节点时会出现不稳定性，导致分类边界偏移。通过加入一个微小的误差容忍度并调试划分条件，这个问题最终得以妥善解决。

心得体会

这次项目让我深刻体会到“数据决定上限，建模决定下限”的道理。预处理虽然繁琐，但它为整个模型训练提供了干净、可控的输入，是所有工作的基础。在实现逻辑回归与决策树的过程中，我进一步掌握了这些经典模型的原理和底层机制，理解了从梯度更新到划分准则背后的数理逻辑。此外，通过封装 `Dataset` 接口，我也积累了构建可复现训练流程的实战经验，为后续的机器学习项目开发打下了坚实基础。

谢敬豪同学

主要任务

- 实现 KNN（K近邻）算法与支持向量机（SVM）模型；
- 负责所有实验的执行与结果可视化；
- 撰写模型评估与对比分析部分的文字说明。

具体贡献

- 实现 KNN 分类器，支持多种距离度量，并加入 KD-Tree 加速查询；
- 实现多类 SVM 分类器，支持硬间隔与结构化损失函数；
- 使用 Matplotlib 与 Seaborn 绘制混淆矩阵、准确率对比图、训练曲线等；
- 综合各模型实验结果，撰写了模型性能比较与错误分析部分。

遇到的挑战与解决方案

我在实现支持向量机（SVM）模型时遇到了一个比较严重的问题：模型在训练过程中梯度更新不稳定，尤其是在样本分布不平衡时，权重更新常常出现震荡甚至发散现象。起初尝试通过调小学习率解决，但收敛速度变得极慢，训练效率大幅下降。最终我引入了指数衰减学习率策略，并结合小批量训练机制，使得模型在初期迅速收敛，同时后期也能保持稳定性。在可视化与评估部分，为了确保不同模型输出具有统一性，我花了大量时间设计标准化的图表格式，包括统一混淆矩阵的颜色条、坐标标签、图例风格等，使比较结果一目了然，便于撰写分析报告。

心得体会

整个项目过程中，我不仅系统掌握了 KNN 与 SVM 的底层逻辑和实现细节，也进一步加深了对模型在面对数据稀疏或类别失衡时表现差异的理解。尤其是在模型评估与可视化方面，我意识到一份清晰直观的图表不仅是对实验结果的呈现，更是对读者理解模型行为的有效支撑。此外，作为项目中负责实验执行和结果整合的成员，我也锻炼了整体流程统筹与细节把控的能力，增强了对多模型对比分析与综合评估的信心。