**财报造假识别程序说明**

1. **程序使用方法**

程序主体：该项目旨在使用多个机器学习模型对财报造假进行识别，涉及样本选取、数据预处理、模型训练、结果展示和预测。程序主流程位于 **main.py**，可通过调节各类开关变量（如 update\_illegal, update\_logit 等）来控制各模块的执行。用户需在正确配置数据和路径后运行 **main.py** 调用整个流程。

为正确调用该函数，需在当前路径下准备如下数据：

**input文件夹中：**

**STK\_Violation\_Main.xlsx**：CSMAR的财务数据违规表，包含了财务违规案例、年份、违规类型等信息，用于选择违规样本

**手动指标名称对照.xlsx**：指标英文与对应中文含义的对照表。由于部分指标是从数据库中提取，部分为重新构建，需要手动对照各指标和其含义。

**output文件夹:**

用于存放输出数据



1. **函数（代码）说明文档**

代码需要按顺序运行，但是只要前面的步骤运行过（已经生成文件），则可以直接从中间步骤开始运行，方便调试。其中，3.1)、3.2)、3.3) 三个模型训练过程不存在先后关系，互相独立。但是需要三个模型都训练完毕，才能运行4）及之后的程序

* 1. **dataset\_obtain\_csmar.py**

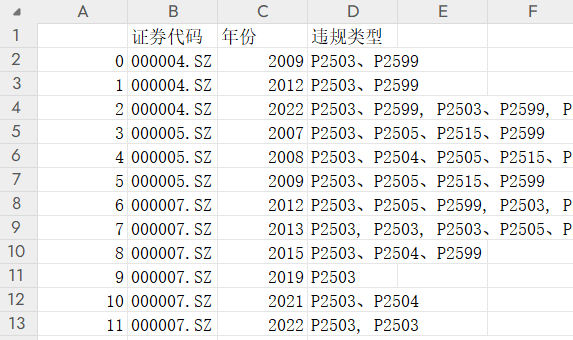
功能：从csmar数据库中，选取造假样本和控制样本，构造样本集合

输入：STK\_Violation\_Main.xlsx：财务数据违规表**（需要手动从网站下载，定期更换，位置为./input）**

Dataset\_path: 数据库的绝对路径

输出：违规案例整理\_csmar.csv

* 文件名: 违规案例整理\_csmar.csv
* 字段：
  + Pandas自带序号 (int)
  + 证券代码 (string) : 股票的代码，格式如000001.SZ
  + 年份 (int) : 对应数据的年份，如1999、2000
  + 违规类型 (string) : 违规行为的类型编码，没有违规则为空。具体参照CSMAR的说明文档，放在 ./input/STK\_Violation\_Main字段说明.xlsx



* 1. **dataset\_obtain\_wind\_ai.py**

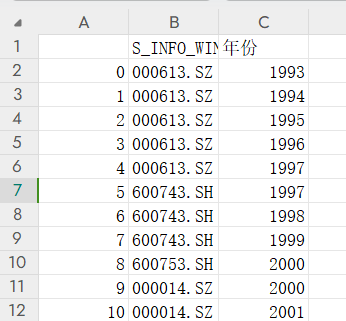
功能：从wind数据库中，选取造假样本和控制样本，并使用ai识别造假发生年份信息，构造样本集合。内置检测自动更新功能，检测是否与上次相比有更新。仅对更新的部分使用ai识别，节省成本。

输入：my\_api：调用大模型需要的api

Dataset\_path: 数据库的绝对路径

输出：违规案例整理\_wind\_ai.csv

* 文件名: 违规案例整理\_wind\_ai.csv
* 字段：
  + Pandas自带序号 (int)
  + S\_INFO\_WINDCODE (证券代码) (string) : 股票的代码，格式如000001.SZ
  + 年份 (int) : 对应数据的年份，如1999、2000



* 1. **dataset\_obtain\_combine.py**

功能：结合csmar和wind的结果，取并集，构造样本集合

输入：违规案例整理\_wind\_ai.csv

违规案例整理\_csmar.csv

Dataset\_path: 数据库的绝对路径

输出：违规案例整理.xlsx

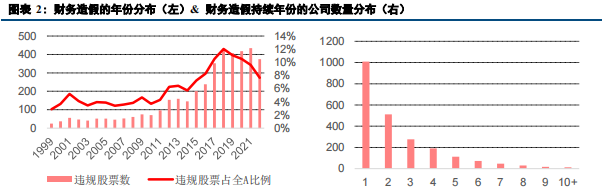
* 文件名: 违规案例整理.xlsx
* 字段：
  + Pandas自带序号 (int)
  + 证券代码 (string) : 股票的代码，格式如000001.SZ
  + 年份 (string) : 对应数据的年份，如1999、2000
  + 违规类型 (string) : 违规行为的类型编码，没有违规则为空。具体参照CSMAR的说明文档，放在 ./input/STK\_Violation\_Main字段说明.xlsx;对于仅出现于wind中的案例，标记为“wind:信息披露虚假或严重误导性陈述”
  + 所属行业 (string) : 股票所属的中信一级行业代码，格式如CI005021.WI, 为分类为空。
  + 证券简称 (string) : 股票的简称，例如“平安银行”
  + label (int) : 标签字段，表示是否存在违规行为，0表示没有违规，1表示有违规。



* 1. **Sample\_description.py(一般可以不使用)**

仅用于做样本数据的描述性分析，生成word报告第一章节：*一）财务造假样本筛选及特征分析* 中的图表

e.g.



* 1. **feature\_engineering.py**

功能：特征工程，构造特征表

输入：违规案例整理.xlsx

start\_year 训练集案例最早的年份，需要保证导入的财务数据库包含比参数更早1年的年份（需要计算同比）

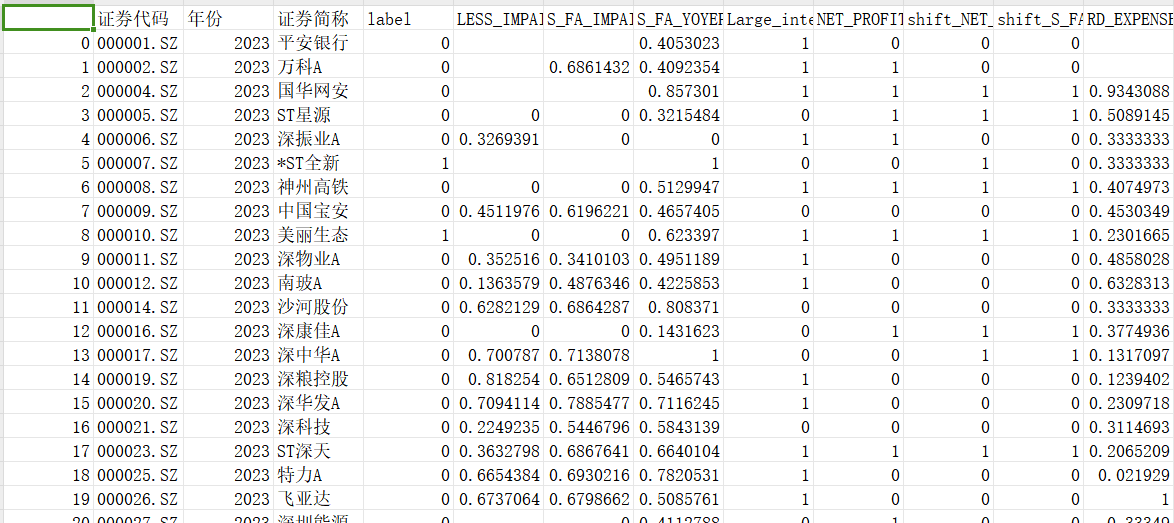
end\_year 训练集案例最晚的年份，需要包括predict\_year

predict\_year 构建模型和验证模型效果的时候，不使用该年份；模型预测中，预测该年份（一般为最后一年，等于end\_year）

dataset\_path 数据库的绝对路径

输出：features\_train.csv, features\_predict.csv

* 文件名: 违规案例整理.xlsx
* 字段：
  + Pandas自带序号 (int)
  + 证券代码 (string) : 股票的代码，格式如000001.SZ
  + 年份 (string) : 对应数据的年份，如1999、2000。features\_predict.csv中仅包含predict\_year年的样本，而features\_train.csv包含其他所有年份的样本
  + 证券简称 (string) : 股票的简称，例如“平安银行”
  + label (int) : 标签字段，表示是否存在违规行为，0表示没有违规，1表示有违规
  + 指标列 (float / int): 指标，允许空值



* 1. **logistic\_train.py**  
     功能： 使用Logistic模型进行分类训练。

输入：features\_train.csv 包含特征和标签的训练数据

this\_test\_size 测试集划分比例

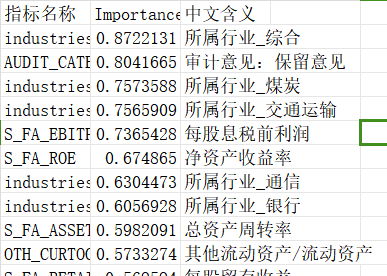
this\_random 随机种子，用于保持数据划分的可重复性

this\_beta 用于阈值选择时的beta系数

输出： Logistic.pkl 训练后的模型，并输出训练集和测试集上的性能指标。

features\_importance\_logistic.csv 特征重要性表格

* 文件名: features\_importance\_logistic.csv
* 字段：
  + 指标名称 (string)
  + Importance (float) : 降序排列
  + 中文含义 (string) : 对应中文含义



* 1. **MLP\_train.py**  
     功能： 使用MLP模型和GAN进行分类训练和生成少数类样本，应用Early Stopping机制以优化训练过程。

输入：features\_train.csv 包含特征和标签的训练数据

this\_test\_size 测试集划分比例

this\_random 随机种子，用于保持数据划分的可重复性

this\_beta 用于阈值选择时的beta系数

输出：mlp.pkl 保存MLP模型，并输出训练集和测试集上的性能指标

features\_importance\_mlp.csv 特征重要性表格

（格式同特征重要性表格 features\_importance\_logistic.csv）

* 1. **XGBoost\_train.py**  
     功能： 使用XGboost模型和GAN进行分类训练和生成少数类样本，应用Early Stopping机制以优化训练过程。

输入：features\_train.csv 包含特征和标签的训练数据

this\_test\_size 测试集划分比例

this\_random 随机种子，用于保持数据划分的可重复性

this\_beta 用于阈值选择时的beta系数

输出：best\_xgb\_model.pkl XGboost模型，并输出训练集和测试集上的性能指标

features\_importance\_xgb.csv 特别征重要性表格

（格式同特征重要性表格 features\_importance\_logistic.csv）

### **result\_presenting.py**

功能： 使用三种不同的机器学习模型（Logistic回归、XGBoost、MLP）进行二分类任务。它对这些模型测试评估，并结合集成学习法对模型结果进行融合。该脚本主要步骤包括模型加载、评估及平均特征重要性计算。

输入：this\_test\_size 测试集划分比例

this\_random 随机种子，用于保持数据划分的可重复性

this\_beta 用于阈值选择时的beta系数

best\_xgb\_model.pkl XGBoost模型

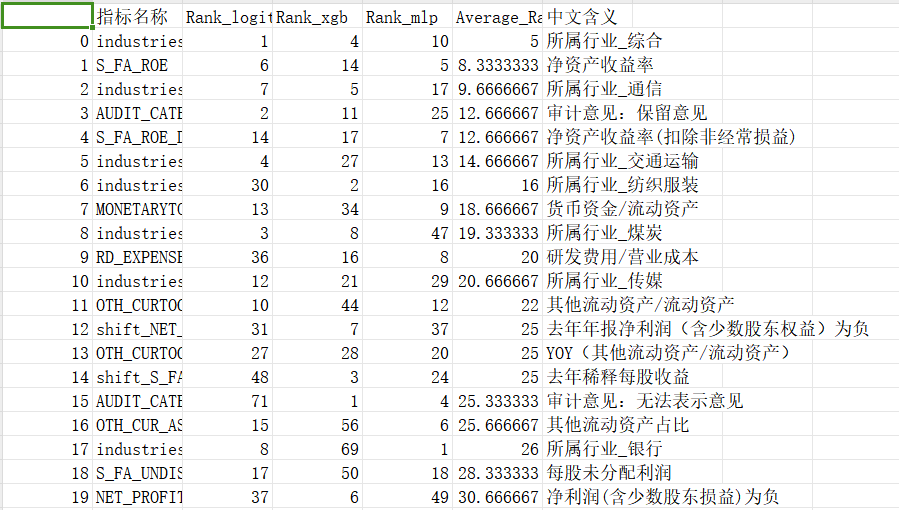
mlp.pkl MLP模型

Logit\_model.pkl Logistic模型

输出：features\_importance\_ave\_rank.csv 三个模型的特征重要性及其综合排名

评估结果：模型在训练集和测试集上的评估结果，包括加权F1分数、准确率等。

* 文件名: features\_importance\_ave\_rank.csv
* 字段：
  + Pandas自带序号（int）
  + 指标名称 (string)
  + Rank\_logit (int) : Logistic回归特征重要性排序
  + Rank\_xgb (int) : XGBoost特征重要性排序
  + Rank\_mlp (int) : MLP回归特征重要性排序
  + Average\_Rank (float) : 升序排序，特征重要性排序平均值
  + 中文含义 (string) : 对应中文含义



### **result\_predicting.py**

功能： 使用三个模型（Logistic回归、XGBoost和MLP）对输入数据进行预测，并计算预测结果的评估指标。最后通过软集成学习法，将各模型的预测结果融合，输出最终的综合预测结果。该脚本主要步骤包括数据处理、模型加载、评估以及输出预测结果。

输入：features\_train.csv 包含训练数据的特征文件。

features\_predict.csv 包含需要预测的数据文件。

手动指标名称对照.xlsx 包含特征的中文名称对照表。

logit\_model.pkl 训练好的Logistic回归模型。

best\_xgb\_model.pkl 训练好的XGBoost模型。

mlp.pth 训练好的MLP模型的参数文件。

this\_test\_size 测试集划分比例。

this\_random 随机种子。

this\_beta 用于加权F1分数的beta值。

输出：预测结果.csv 保存了每个模型的预测标签及最终集成预测标签的CSV文件

评估报告：每个模型及集成模型的预测结果及加权F1分数等评估指标。

* 文件名: 预测结果.csv
* 字段：
  + 证券代码（string）
  + 年份 (string)
  + 证券简称 (string)
  + 真实标签 (int) : 是否披露违规行为。
  + Logistic预测标签 (int) : 0为预测正常，1为预测造假。
  + MLP预测标签 (int) : 0为预测正常，1为预测造假。
  + XGBoost预测标签 (int) : 0为预测正常，1为预测造假。
  + 综合预测标签 (int) : 集成学习的判断结果，0为预测正常，1为预测造假**。**

