DANet为baseline代码，是针对表格类数据集的模型，可以完成分类和回归任务

DANet提出了一种用于表格数据的新颖且灵活的神经组件，称为抽象层（AbstLay），它学习对相关输入特征进行显式分组，并为语义抽象生成更高级别的特征。此外，设计了一种结构重新参数化方法来压缩AbstLay，从而在参考阶段将计算复杂度降低了明显的幅度。使用AbstLays构建了一个特殊的基本模块，我们通过堆叠这些模块构建了一个用于表格数据分类和回归的深度抽象网络（DANets）系列。在 DANet 中，引入了一个特殊的快捷方式路径，用于从原始表格要素中获取信息，从而帮助跨不同级别的要素交互。

**一、数据集**

数据集格式为.csv的表格类数据集。

数据集通过./utils中的get\_idx.py进行了训练集和测试集的划分。

**二、训练**

在配置文件中设置超参数（./config/default.py或./config/\*.yaml）。 特别是，.yaml文件中的超参数将覆盖default.py中的。

通过python main.py --c [config\_path] --g [gpu\_id]运行。

-c: 配置文件路径

-g: GPU设备ID 检查点模型和最佳模型将保存在./logs文件中。

**三、推理**

用包含你训练模型/权重的文件路径替换resume\_dir路径。

使用python predict.py -d [dataset\_name] -m [model\_file\_path] -g [gpu\_id]运行代码。

-d: 数据集名称

-m: 加载模型的路径

-g: GPU设备ID

**四、配置超参数**

**普通参数**

dataset: 字符串

给定的数据集名称必须与./data/dataset.py中的名称匹配。

task: 字符串

选择一个预设的任务，'classification'（分类）或'regression'（回归）。

resume\_dir: 字符串

包含检查点模型的日志路径。

logname: 字符串

保存在./logs的模型目录名称。

seed: 整数

随机种子。

**模型参数**

layer: 整数（默认=20）

堆叠的抽象层数

k: 整数（默认=5）

掩码的数量

base\_outdim: 整数（默认=64）

抽象层中的输出特征维度。

drop\_rate: 浮点数（默认=0.1）

快捷模块中的丢弃率

**拟合参数**

lr: 浮点数（默认=0.008）

学习率

max\_epochs: 整数（默认=5000）

训练的最大周期数。

patience: 整数（默认=1500）

在执行早停之前，连续没有改进的周期数。如果patience设置为0，则不执行早停。

batch\_size: 整数（默认=8192）

每批的样本数。

virtual\_batch\_size: 整数（默认=256）

用于“Ghost Batch Normalization”的迷你批次大小。virtual\_batch\_size必须能整除batch\_size。