横向DNN算法参数介绍文档

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数含义 | 默认值 | 可选值 |
| secure\_aggregate | 是否开启安全聚合算法 | True | 为bool值，可以设置为False |
| aggregate\_every\_n\_epoch | 进行安全聚合的频率，即每n轮进行一次聚合更新 | 1 | 为int型整数 |
| config\_type | 参数类型，判断具体模型参数采用何种后端包进行实现 | "nn" | "nn"、"keras"、"pytorch" |
| nn\_define | 描述模型结构的字典结构 |  |  |
| nn\_define.layer | config\_type为keras时使用，标注模型名称 | Dense | "Dense"、"Dropout" |
| nn\_define.units | config\_type为keras时使用，全连接层神经元数量即输出维度 |  | 正整数 |
| nn\_define.use\_bias | config\_type为keras时使用，是否使用偏置 | True | bool值，可设置为False |
| nn\_define.activation | config\_type为keras时使用，激活函数 |  | keras支持的激活函数均可，常见如RELu、Softmax |
| nn\_define.kernel\_initializer | config\_type为keras时使用，模型权重初始化方法 |  | keras支持的初始化方法均可，常见如zeros、ones、glorot\_uniform等 |
| nn\_define.bias\_initializer | config\_type为keras时使用，模型偏置初始化方法 | "Zeros" | keras支持的初始化方法均可，常使用zeros |
| nn\_define.kernel\_regularizer | config\_type为keras时使用，模型权重正则化器 |  | keras支持的正则化方法均可，一般不使用 |
| nn\_define.bias\_regularizer | config\_type为keras时使用，偏移量正则化器 |  | keras支持的正则化方法均可，一般不使用 |
| nn\_define.activity\_regularizer | config\_type为keras时使用，激活值正则化器 |  | keras支持的正则化方法均可，一般不使用 |
| nn\_define.kernel\_constraint | config\_type为keras时使用，权重的约束 | None | 不能设置，该参数对于多方计算不安全 |
| nn\_define.bias\_constraint | config\_type为keras时使用，偏移量的约束 | None | 不能设置，该参数对于多方计算不安全 |
| nn\_define.rate | config\_type为keras，nn\_define.layer为Dropout时使用，表示该层的dropout率 |  | 0-1之间的数值 |
| nn\_define.noise\_shape | config\_type为keras，nn\_define.layer为Dropout时使用，表示该层的噪声的维度形状 | None | 根据设计的噪声进行输入 |
| nn\_define.seed | config\_type为keras，nn\_define.layer为Dropout时使用，表示该层的随机数种子 | None | 整数 |
| nn\_define.layer | config\_type为pytorch时使用，表示该层的模型类型 |  | layers:  "Linear"、"BatchNorm2d"、"Dropout"  activate:  "Relu"、"Selu"、"LeakyReLU"、"Tanh"、"Sigmoid" |
| nn\_define.type | config\_type为pytorch时使用，对应nn\_define.layer |  | "normal"、"activate" |
| nn\_define.name | config\_type为pytorch时使用，标识模型结构的名称，具有唯一性 |  | 接收字符串 |
| nn\_define.config | config\_type为pytorch时使用，为长度为2的数组 |  | [x, y]：  前者数值为输入维度，后者数值为输出维度 |
| optimizer | 优化器方法，为模型更新当中的基础设定 | "SGD" | 接收两种内容：  1）字符串：  "Adadelta"、"Adagrad"、"Adam"、"Adamax"、"Nadam"、"RMSprop"、"SGD"  2）字典结构，带有支持的优化器名及其所适配的属性键值对（如学习率等），具体结构见keras文档<https://keras.io/api/optimizers/> |
| optimizer.optimizer | 字典结构的优化器方法中的必要字段，对应使用的优化器 |  | "Adadelta"、"Adagrad"、"Adam"、"Adamax"、"Nadam"、"RMSprop"、"SGD"等 |
| optimizer.learning\_rate | 字典结构的优化器方法中的必要字段，对应优化器的学习率 |  | 0-1之间的数字，一般不超过0.1 |
| loss | 描述模型损失函数的参数 | str=None | 字符串型变量，具体输入详见文档<https://keras.io/api/losses/> |
| metrics | 描述模型评估指标的参数 | None | 可以接受字符串和字符串列表形式的输入 |
| max\_iter | 训练中的最大迭代轮次 | 100 | int型，可随意设置 |
| batch\_size | 训练更新模型时所使用的数据规模 | -1 | int型，默认-1为使用全部数据，即不使用mini-batch策略 |
| early\_stop | 判断训练是否收敛的方法 |  |  |
| early\_stop.early\_stop | 具体判断训练是否收敛的方法 | "diff" | 字符串变量：  1）"diff"：利用两轮迭代间的loss差分值判断  2）"weight\_diff"：利用两轮迭代间模型权重的差分值判断  3）"abs"：利用loss值的绝对值与阈值进行比对判断 |
| early\_stop.eps | 判断训练收敛算法中的精确度 | 0.0001 | 可写成0.0001或1e-4等格式，精确度可调 |
| encode\_label | 是否对标签项进行热编码 | False | bool值，可设置为True |
| predict\_param | 预测相关参数 |  |  |
| predict\_param.threshold | 用于区分正负样本的阈值 | 0.5 | 0到1之间的任意值 |
| cv\_param | 参数名:参数含义 |  |  |
| cv\_param.n\_splits | 指定在KFold中使用多少个分割点 | 5 | 大于0任意正整数 |
| cv\_param.mode | 指明当前的任务模式 | Hetero | Homo |
| cv\_param.role | 在当前任务中扮演什么角色 | Guest | Host,Arbiter |
| cv\_param.shuffle | 在KFold之前是否打乱顺序 | True | False |
| cv\_param.random\_seed | 随机种子 | 1 | 任意整数 |
| cv\_param.need\_cv | 是否要运行该模块 | False | True |
| cv\_param.output\_fold\_history | 是否输出每个fold使用的每个id表 | True | False |