垂直联邦：用户覆盖相同，但是特征不同

# 垂直联邦框架介绍

## SF垂直联邦特性

1. 支持多种类型特征
2. 自定义数据预处理流程
3. 支持灵活自定义模型结构，内置了部分拆分模型（bst等）
4. 内置通信/计算优化、安全加强方法
5. 多种后端

## 训练流程

1. 数据对齐，保证用户相同
2. 前向传播获得Embedding
3. 通过安全方式把embedding发送给含有label的一方，或者先发给server再发给label方
4. Label方进行融合、前向传播得到predict y，产生loss和梯度，反向传播后回传
5. 融合结束，梯度拆分，发回给双方，继续反向传播

## 主要提供的API：SLModel

class SLModel

1. 如果有一方只有标签，没有特征，base model dict可以去掉这一方。这个参数里面的pyu是由特征的一方。
2. Device y是label方
3. DP是数据保护措施（数据会出域再入域）
4. Random seed是shuffle等操作时用来确保双方同步的

## Strategy

目前提供4种训练策略

1. Split\_nn, 传统同步模式，a到b计算完回到b到a
2. split\_async，是论文FedBCD的实现，自定义本地区训练多少论，然后再发给label方
3. split\_state\_async: 根据某些状态决定本地训练多少轮。减少传输次数，降低通信成本
4. pipeline: 并行度提高

## AggLayer

聚合层。用于让聚合方只能拿到聚合后的结果。可以通过某个device进行聚合（PYU、TEEU）

1. 如果没有agg\_method, 默认没有server
2. device\_agg可以是PYU、TEEU、SPU
3. 如果自定义agg\_method，需要用jax.numpy实现forward（自动推导backward）

目前提供的agg\_method是avg、sum两个

## DP

垂直联邦、拆分学习涉及到hidden layer出域，需要保护隐私，可以用dp

可以在发送前、聚合后发回去之前对梯度进行扰动，保证无法反向推导

可以用的dp有：

高斯embedding dp

label dp

使用方式：

1. 定义dp strategy（类型、参数）
2. Strategy绑定到strategy\_dict中
3. 传给SLModel的相关参数

# SLModel开发实践

## 水平FLModel实战

数据集：marketing，有16个特征，标签是二分类。从kaggle上下载

11162个数据，8929训练、2233测试

参考相应ipynb

# SFXgboost实践

sf的xgboost和明文xgboost基本对齐。

明文xgboost：

import xgboost as xgb

# read data

dtrain=xgb.DMatrix(“demo/data/agaricus.txt.train”)

dtest=xgb.DMatrix(“demo/data/agaricus.txt.test”)

# specify map

param={‘max\_depth’:2, ‘eta’:1, ‘objective’:’binary:logistic’}

num\_round=2

bst=xgb.train(param, dtrain, num\_round)

sf的xgboost：

bst=SFXgboost(server=self.davy, clients={self.alice, slef.bob})

params={

‘max\_depth’:4,

‘eta’:1.0,

‘objective’:’binary:logistic’,

‘label\_key’:’label’ # 标记hdataframe中的label列名

}

bst.train(self.hdf, self.hdf, params=params, num\_boost\_round=4)

注意事项：

目前水平XGBoost只支持HDataFrame类型的数据

模型超参数：和明文几乎一致

FineTune: bst.train()里面传入模型路径就可以了

Predict没有实现

剩下内容（详细代码）见ipynb