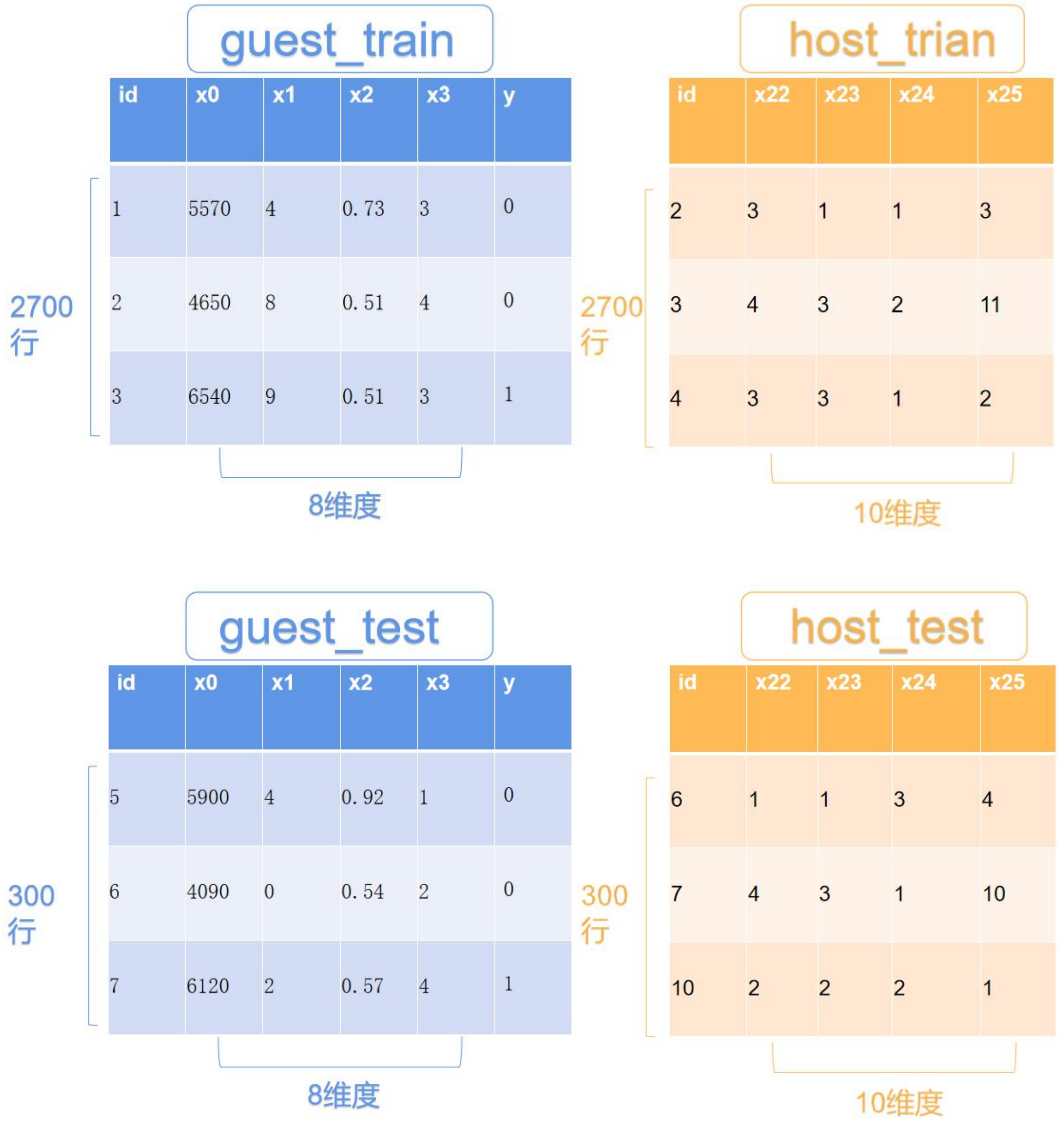
一、纵向联邦学习背景

纵向：GUEST 方和 HOST 方的数据特征维度有差异，但用户（ID）重合度高，适用于跨行业应用的场景。

二、数据准备

双方数据规模如下图所示，ID 存在重叠，特征不同，guest 方为有标签的一方：按 7:3 比例分割好训练集和测试集共四份数据 heterotrain_g.csv/heterotest_g.csv/heterotrain_h.csv/heterotest_h.csv。



二、上传数据

Guest 端：

上传训练集：python fate_flow_client.py -f upload -c examples/upload_try_g.json

- upload_try_g.json 文件如下：

{

"file": "/home/app/projects/heterotrain_g.csv",

"head": 1,

"partition": 50,

"work_mode": 1,

"namespace": "fate_testg",

"table_name": "datag"

}

file：文件路径
head：数据文件是否包含表头
partition：用于存储数据的分区数
work_mode：指定的工作模式，0 代表单机版，1 代表集群版
table_name&namespace：存储数据表的标识符号

上传测试集：python fate_flow_client.py -f upload -c

examples/upload_try_gg.json

- upload_try_gg.json 文件如下：

{

```
    "file": "/home/app/projects/heterotest_g.csv",
    "head": 1,
    "partition": 50,
    "work_mode": 1,
    "namespace": "fate_testgg",
    "table_name": "datagg"
}
```

Host 端：

上传训练集：python fate_flow_client.py -f upload -c examples/upload_try_h.json

- upload_try_h.json 文件如下：

```
{
    "file": "/home/app/projects/heterotrain_h.csv",
    "head": 1,
    "partition": 50,
    "work_mode": 1,
    "namespace": "fate_testh",
    "table_name": "datah"
}
```

上传测试集：python fate_flow_client.py -f upload -c examples/upload_try_hh.json

- upload_try_hh.json 文件如下：

{

"file": "/home/app/projects/heterotest_h.csv",

"head": 1,

"partition": 50,

"work_mode": 1,

"namespace": "fate_testhh",

"table_name": "datahh"

}

三、提交任务

Guest 端：

提交训练任务:

python fate_flow_client.py -f submit_job -d

examples/test_hetero_lr_job_dsl.json -c

examples/test_hetero_lr_job_conf.json

- test_hetero_lr_job_conf.json 文件如下：

```
{
    "initiator": {
        "role": "guest",
        "party_id": 9999
    },
    "job_parameters": {
        "work_mode": 1
    },
    "role": {
        "guest": [9999],
        "host": [10000],
        "arbiter": [10000]
    },
    "role_parameters": {
        "guest": {
            "args": {
                "data": {
                    "train_data": [{"name": "datag", "namespace":
"fate_testg"}],
                    "eval_data": [
                        {
                            "name": "datagg",
                            "namespace": "fate_testgg"
                        }
                    ]
                }
            },
            "dataio_0":{
                "with_label": [true],
```

```
                    "label_name": ["y"],
                    "label_type": ["int"],
                    "output_format": ["dense"]
                }
            },
            "host": {
                "args": {
                    "data": {
                        "train_data": [{"name": "datah", "namespace":
"fate_testh"}],
                        "eval_data": [
                            {
                               "name": "datahh",
                               "namespace": "fate_testhh"
                            }
                          ]
                    }
                },
                  "dataio_0":{
                    "with_label": [false],
                    "output_format": ["dense"]
                }
            }
        },
        "algorithm_parameters": {
            "feature_scale_0": {
                "method": "standard_scale",
                "need_run": true
              },
              "hetero_feature_binning_0": {
                "method": "quantile",
                "compress_thres": 10000,
                "head_size": 10000,
                "error": 0.001,
                "bin_num": 10,
                "cols": -1,
                "adjustment_factor": 0.5,
                "local_only": false,
                "need_run": true,
                "transform_param": {
                   "transform_cols": -1,
                   "transform_type": "bin_num"
                }
              },
```

```
    "hetero_feature_selection_0": {
            "select_cols": -1,
            "filter_methods": [
                "unique_value",
                "iv_value_thres",
                "coefficient_of_variation_value_thres",
                "iv_percentile",
                "outlier_cols"
            ],
            "local_only": false,
            "unique_param": {
                "eps": 1e-6
            },
            "iv_value_param": {
                "value_threshold": 1.0
            },
            "iv_percentile_param": {
                "percentile_threshold": 0.9
            },
            "variance_coe_param": {
                "value_threshold": 0.3
            },
            "outlier_param": {
                "percentile": 0.95,
                "upper_threshold": 10
            },
            "need_run": true
        },
    "hetero_lr_0": {
        "penalty": "L2",
        "optimizer": "rmsprop",
        "eps": 1e-5,
        "alpha": 0.01,
        "max_iter": 100,
        "converge_func": "diff",
        "batch_size": -1,
        "learning_rate": 0.15,
        "init_param": {
            "init_method": "random_uniform"
        }
    },
    "intersect_0": {
        "intersect_method": "rsa",
        "sync_intersect_ids": true,
```

```
                "only_output_key": false
            }
        }
}
```

- test_hetero_lr_job_dsl.json 文件如下：

```
{
    "components" : {
        "dataio_0": {
            "module": "DataIO",
            "input": {
                "data": {
                    "data": [
                        "args.train_data"
                    ]
                }
            },
            "output": {
                "data": ["train"],
                "model": ["dataio"]
            },
        "need_deploy": true
        },
        "dataio_1": {
            "module": "DataIO",
            "input": {
                "data": {
                    "data": [
                        "args.eval_data"
                    ]
                },
                "model": [
                    "dataio_0.dataio"
                ]
            },
            "output": {
                "data": ["eval_data"],
                "model": ["dataio"]
            }
        },
        "intersection_0": {
            "module": "Intersection",
            "input": {
```

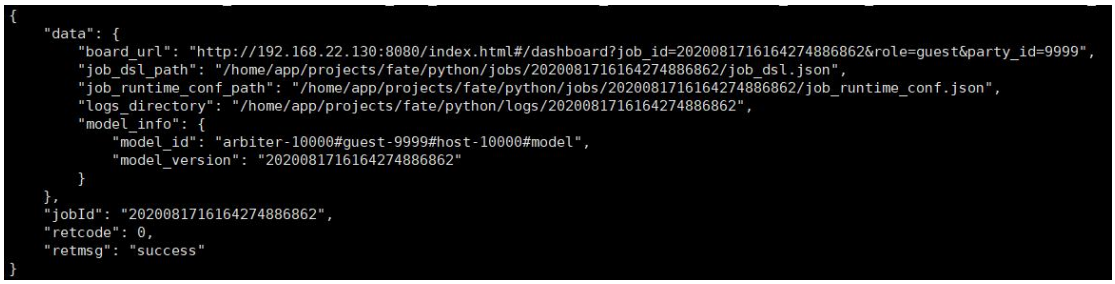
```
            "data": {
                "data": [
                    "dataio_0.train"
                ]
            }
        },
        "output": {
            "data": ["train"]
        }
    },
    "intersection_1": {
        "module": "Intersection",
        "input": {
            "data": {
                "data": ["dataio_1.eval_data"]
            }
        },
        "output": {
            "data": ["eval_data"]
        }
    },
    "feature_scale_0": {
        "module": "FeatureScale",
        "input": {
            "data": {
                "data": [
                    "intersection_0.train"
                ]
            }
        },
        "output": {
            "data": ["train"],
            "model": ["feature_scale"]
        }
    },
    "feature_scale_1": {
        "module": "FeatureScale",
        "input": {
            "data": {
                "data": [
                    "intersection_1.eval_data"
                ]
            },
            "model":["feature_scale_0.feature_scale"]
```

```
        },
        "output": {
            "data": ["eval_data"],
            "model": ["feature_scale"]
        }
    },
    "hetero_feature_binning_0": {
        "module": "HeteroFeatureBinning",
        "input": {
            "data": {
                "data": [
                    "feature_scale_0.train"
                ]
            }
        },
        "output": {
            "data": ["transform_data"],
            "model": ["binning_model"]
        },
        "need_deploy": false
    },
    "hetero_feature_binning_1": {
        "module": "HeteroFeatureBinning",
        "input": {
            "data": {
                "data": [
                    "feature_scale_1.eval_data"
                ]
            },
            "model":["hetero_feature_binning_0.binning_model"]
        },
        "output": {
            "data": ["transform_data_test"],
            "model": ["binning_model"]
        },
        "need_deploy": false
    },
    "hetero_feature_selection_0": {
        "module": "HeteroFeatureSelection",
        "input": {
            "data": {
                "data": [
                    "hetero_feature_binning_0.transform_data"
                ]
```

```
            },
            "isometric_model": [
                "hetero_feature_binning_0.binning_model"
            ]
        },
        "output": {
            "data": ["train"],
            "model": ["selected"]
        }
    },
    "hetero_feature_selection_1": {
        "module": "HeteroFeatureSelection",
        "input": {
            "data": {
                "data": [
                    "hetero_feature_binning_1.transform_data_test"
                ]
            },
            "model":["hetero_feature_selection_0.selected"],
            "isometric_model": [
                "hetero_feature_binning_1.binning_model"
            ]
        },
        "output": {
            "data": ["eval_data"],
            "model": ["selected"]
        }
    },
    "hetero_lr_0": {
        "module": "HeteroLR",
        "input": {
            "data": {
                "train_data": ["hetero_feature_selection_0.train"],
                "eval_data": ["hetero_feature_selection_1.eval_data"]
            }
        },
        "output": {
            "data": ["hetero_lr_data"],
            "model": ["hetero_lr"]
        }
    },
    "evaluation_0": {
        "module": "Evaluation",
        "input": {
```

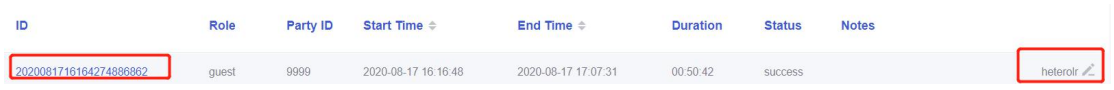
```
                "data": {
                    "data": ["hetero_lr_0.hetero_lr_data"]
                }
            },
            "output": {
                "data": ["evaluate"]
            }
        }
    }
}
```

- 若页面如图所示，则说明提交成功啦~



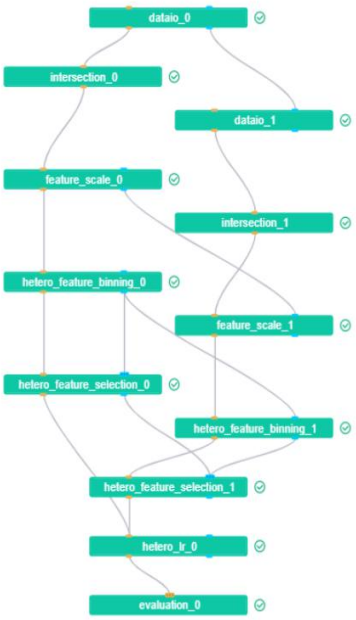
四、查看任务

- 根据 Jobid 识别任务，可在右侧添加备注（目前 1.4.2 仅支持英文备注）

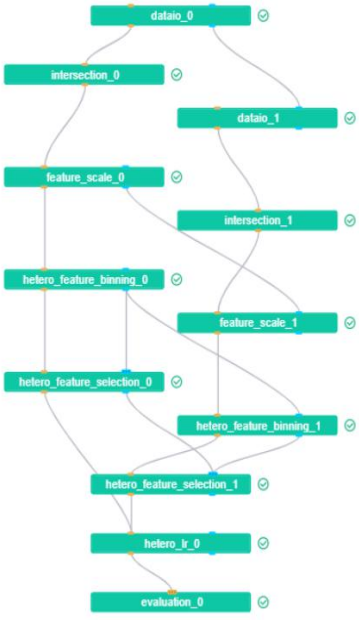


到达 board 界面可在上方查看到三方的流程图如下：

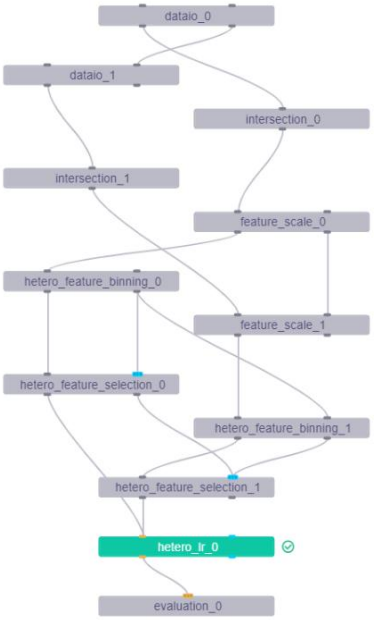
Guest 方



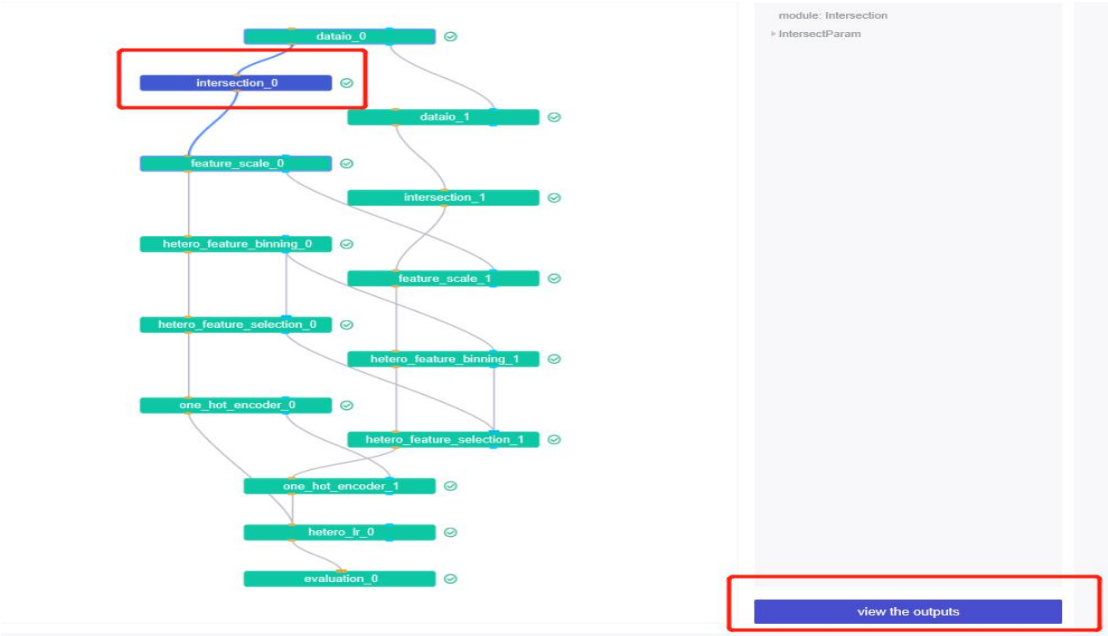
Host 方



Arbiter 方

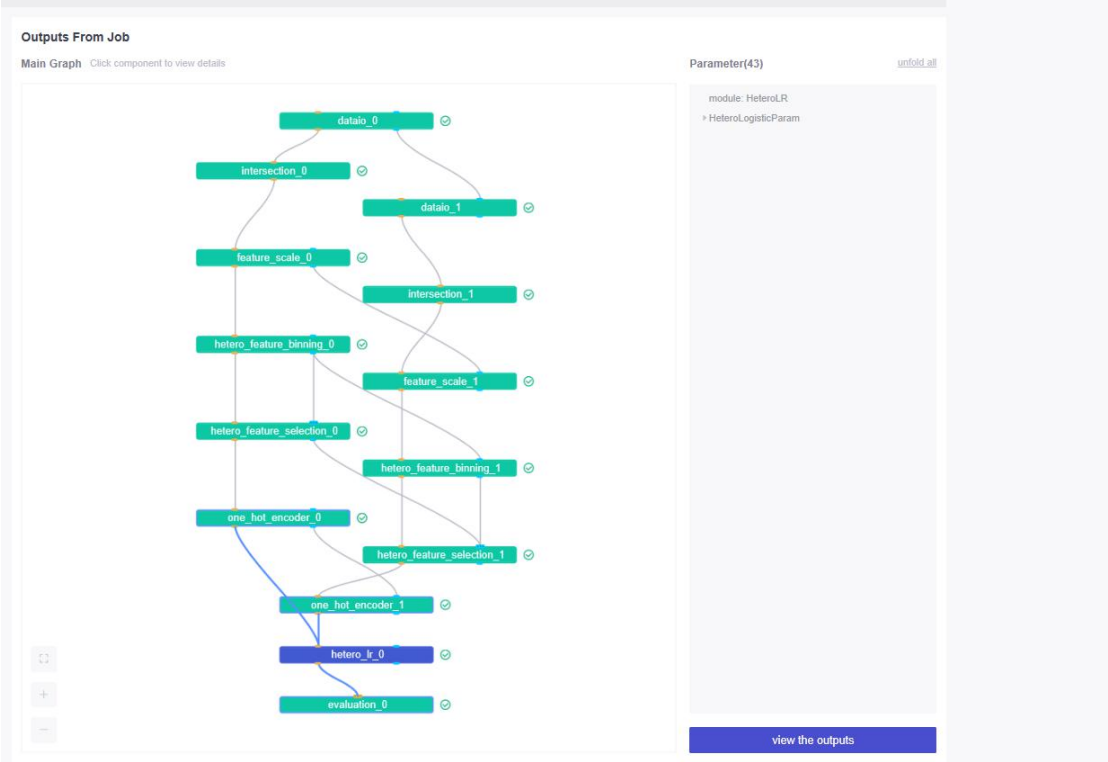


- 点击不同的组件后按 view the outputs 可查看具体的信息

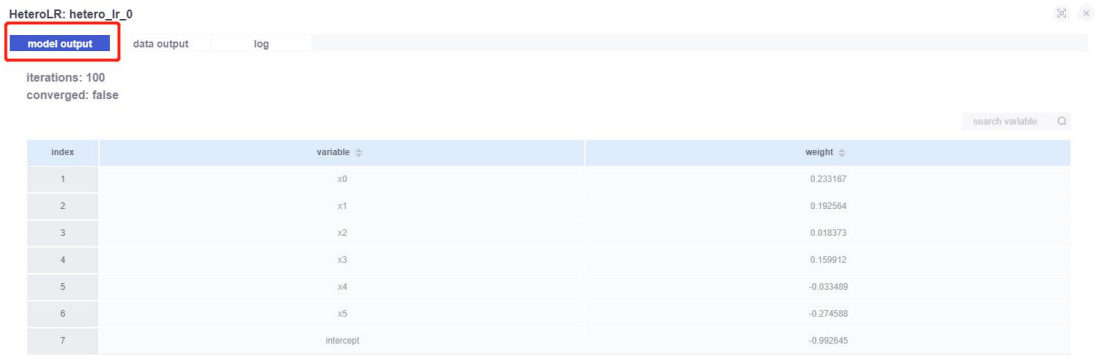


查看模型详情：

Guest 端：点击 hetero_lr_0



Model output：**仅能查看自身**的模型信息，包括最大迭代次数、模型是否收敛以及特征权重列表。

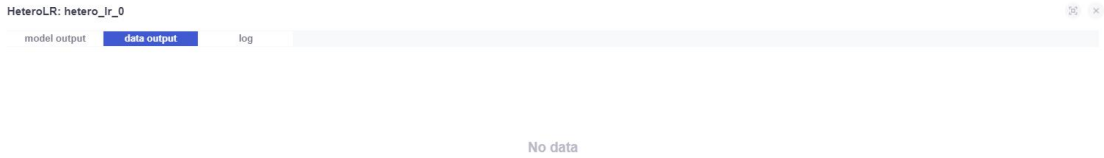


Data output：**查看模型训练后输出的预测标签结果**。

Host 端： 点击 hetero_lr_0

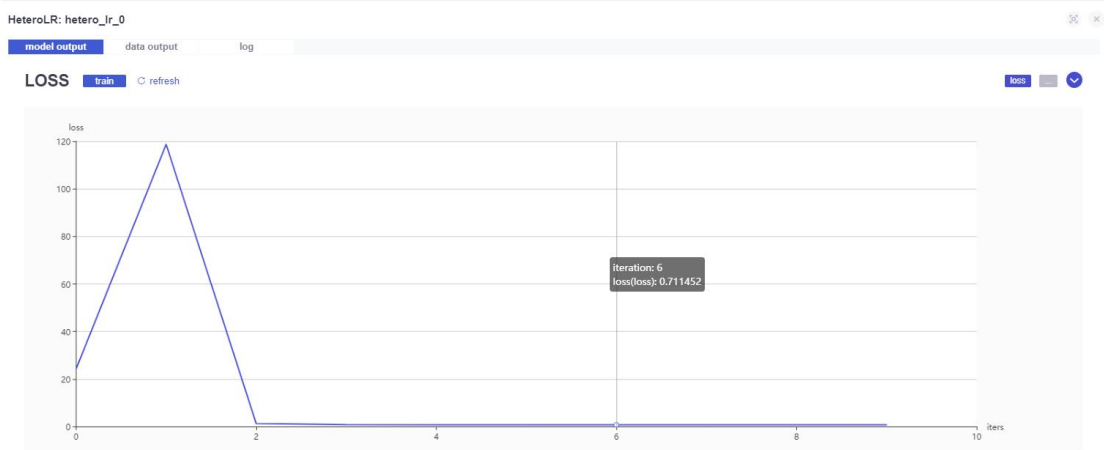
Model output：仅能查看自身的模型信息，包括最大迭代次数、模型是否收敛以及特征权重列表。

Data output：不输出数据

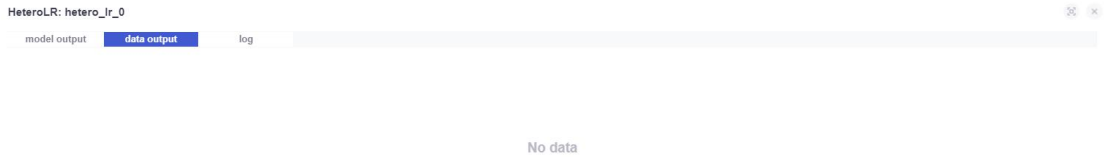


Aribter 端： 点击 hetero_lr_0

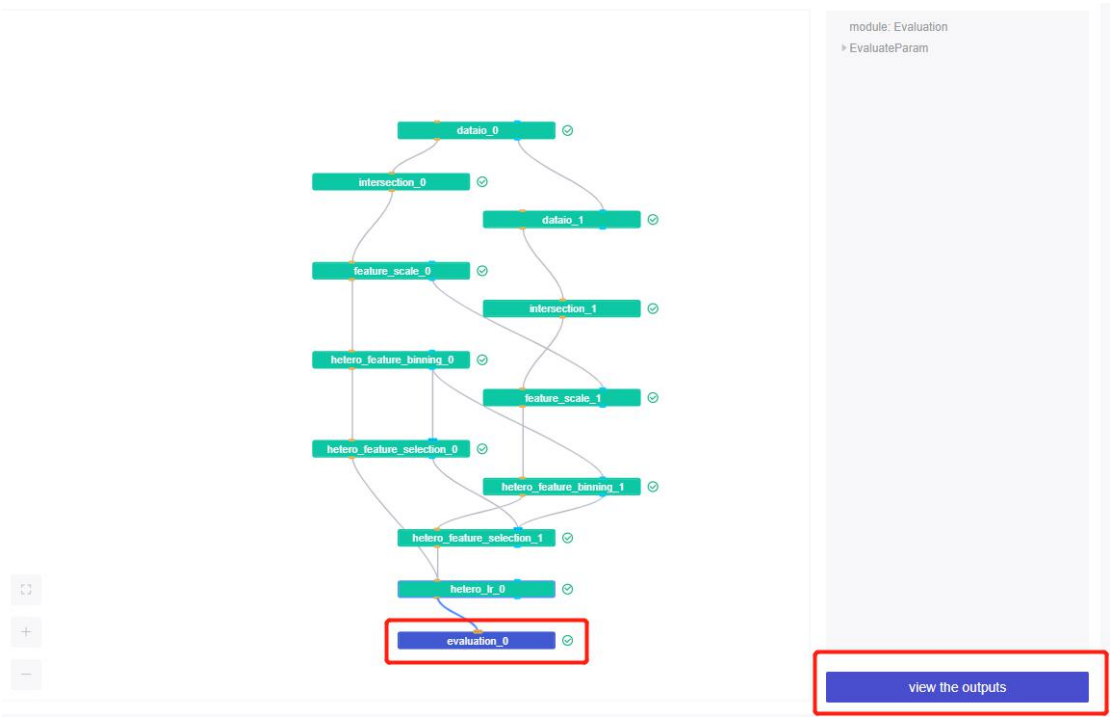
Model output：只可查看 loss 曲线图，除此以外无法查看 guest 及 host 方相关的其他信息。



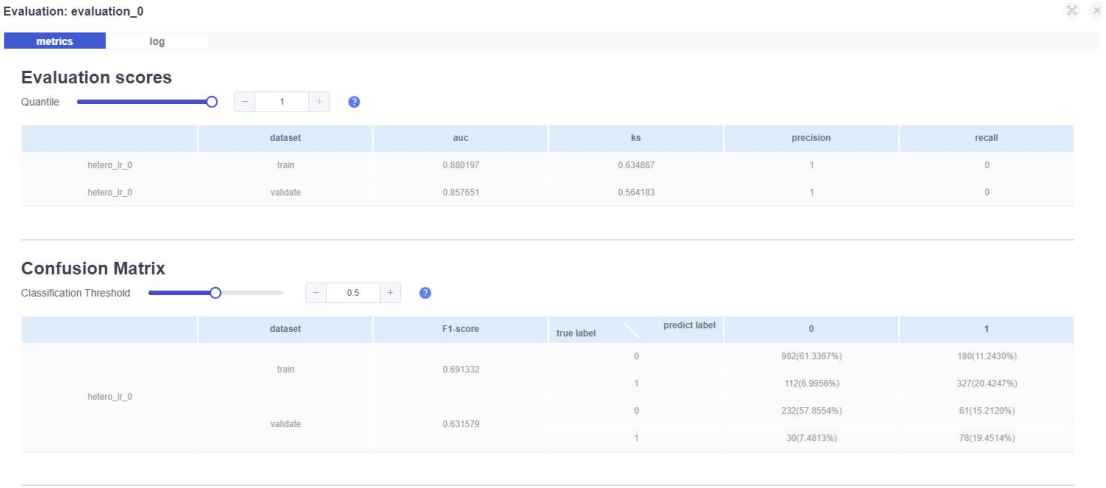
Data output：不输出数据



模型评估：点击 evalution 组件



Guest 端：评估指标表可查看模型在各数据集下的 AUC,KS，precision 和 recall 值、混淆矩阵、PSI 稳定度指标(population stability index ,PSI)可衡量测试样本及模型开发样本评分的的分布差异，为最常见的模型稳定度评估指标。



PSI Summary

index	predict_score	Expected %	Actual %	PSI
1	[0.027543, 0.101071)	4.9969%	4.2500%	0.001209
2	[0.101071, 0.130105)	4.9344%	3.2500%	0.007034
3	[0.130105, 0.161776)	5.0593%	5.5000%	0.000368
4	[0.161776, 0.191201)	4.9969%	4.2500%	0.001209
5	[0.191201, 0.220312)	4.9969%	6.0000%	0.001835
6	[0.220312, 0.249588)	4.9969%	3.7500%	0.003579
7	[0.249588, 0.274496)	4.9969%	4.2500%	0.001209
8	[0.274496, 0.300849)	4.9969%	4.0000%	0.002218
9	[0.300849, 0.333434)	4.9969%	5.7500%	0.001057
10	[0.333434, 0.368767)	4.9969%	6.2500%	0.002804
11	[0.368767, 0.401316)	4.9969%	5.7500%	0.001057

Quantile Distribution

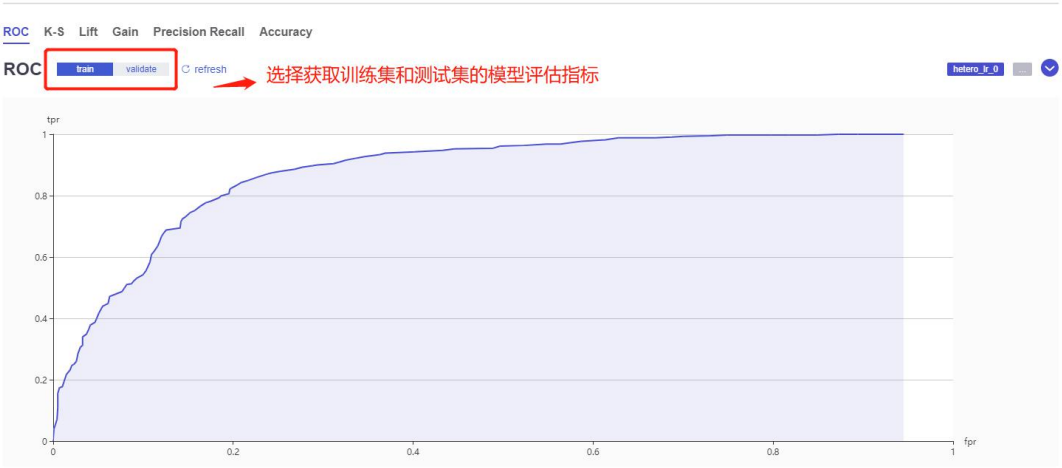
预测得分区间

位于预测得分区间的总样本数

正样本比例

index	predict_score	train		validation	
		instance_count(%total)	event_ratio	instance_count(%total)	event_ratio
1	[0.054496, 0.165254)	80 (4.9969%)	17.5000%	19 (4.7382%)	10.5263%
2	[0.165254, 0.211279)	80 (4.9969%)	7.5000%	16 (3.9900%)	25.0000%
3	[0.211279, 0.246403)	80 (4.9969%)	13.7500%	22 (5.4863%)	22.7273%
4	[0.246403, 0.287403)	80 (4.9969%)	18.7500%	23 (5.7357%)	21.7391%
5	[0.287403, 0.315589)	79 (4.9344%)	26.5823%	24 (5.9850%)	16.6667%
6	[0.315589, 0.348073)	81 (5.0593%)	20.9877%	19 (4.7382%)	21.0526%
7	[0.348073, 0.372187)	80 (4.9969%)	18.7500%	20 (4.9875%)	25.0000%
8	[0.372187, 0.397597)	79 (4.9344%)	26.5823%	17 (4.2394%)	23.5294%

图表区域可选择查看模型在各数据集下的 ROC,K-S，Lift，Gain，PR 和 Accuracy 曲线：



Host 端：查看不到模型指标