Release Notes for Sophon 2.0

星环信息科技（上海）有限公司

版本号 Sophon V2.0.0, 2018-10-12

目录

免责声明

本说明书依据现有信息制作,其内容如有更改,恕不另行通知。星环信息科技（上海）有限公司在编写该说明书的时候已尽最大努力保证期内容准确可靠,但星环信息科技（上海）有限公司不对本说明书中的遗漏、不准确或印刷错误导致的损失和损害承担责任。具体产品使用请以实际使用为准。

注释: Hadoop® 和 SPARK® 是ApacheTM 软件基金会在美国和其他国家的商标或注册的商标。Java®是Oracle公司在美国和其他国家的商标或注册的商标。Intel® 和Xeon® 是英特尔公司在美国、中国和其他国家的商标或注册的商标。

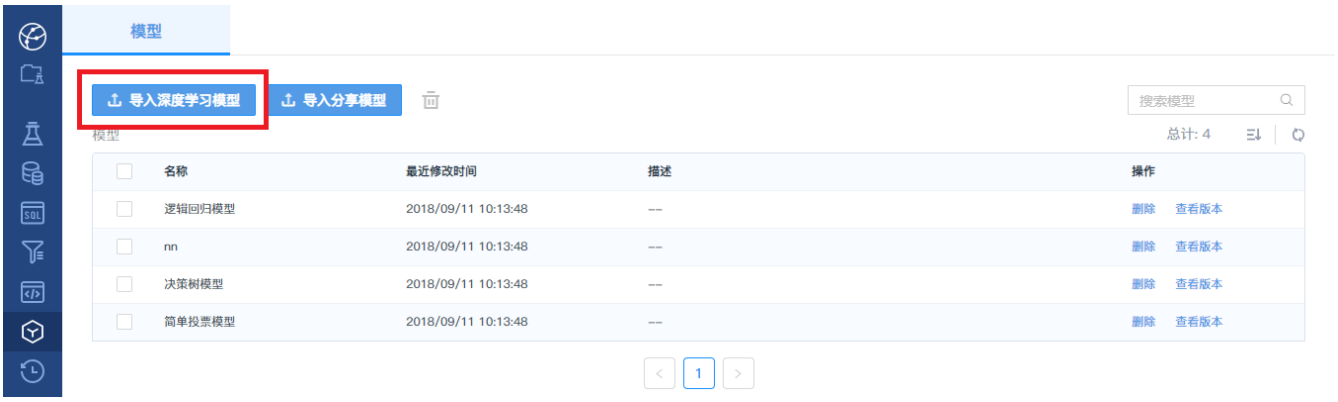
1. 新增功能和产品优化

Sophon 2.0是智子人工智能平台最新推出的发行版本。该版本在算法质量、整体可用性及产品化程度方面都有着显著的提升，极大地优化了通过人工智能解决实际业务问题的体验。依托项目导入导出与协作、文件管理、SQL编辑、深度学习模型查看和导出、数据统计分析和可视化等功能，Sophon 2.0进一步完善了一站式人工智能建模流程。以下是具体的新增功能和产品优化信息。

1.1. 新增项目导入导出

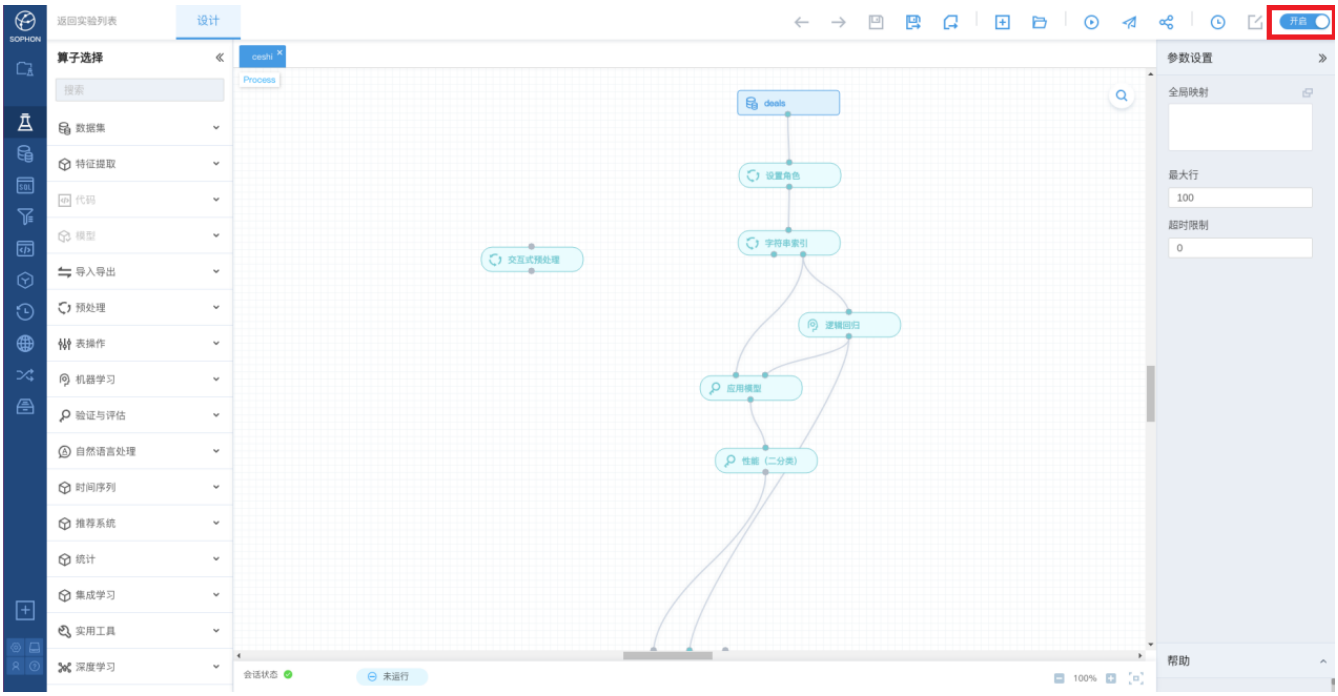
支持单个或批量导出项目、单个导入项目，方便用户将项目保存于本地，在不同场景和环境下使用。

若项目中包含API服务或深度学习模型，则在导入时将不会包含该API服务或深度学习模型。如需导入深度学习模型，可点击“导入深度学习模型”按钮，直接从文件管理中导入。

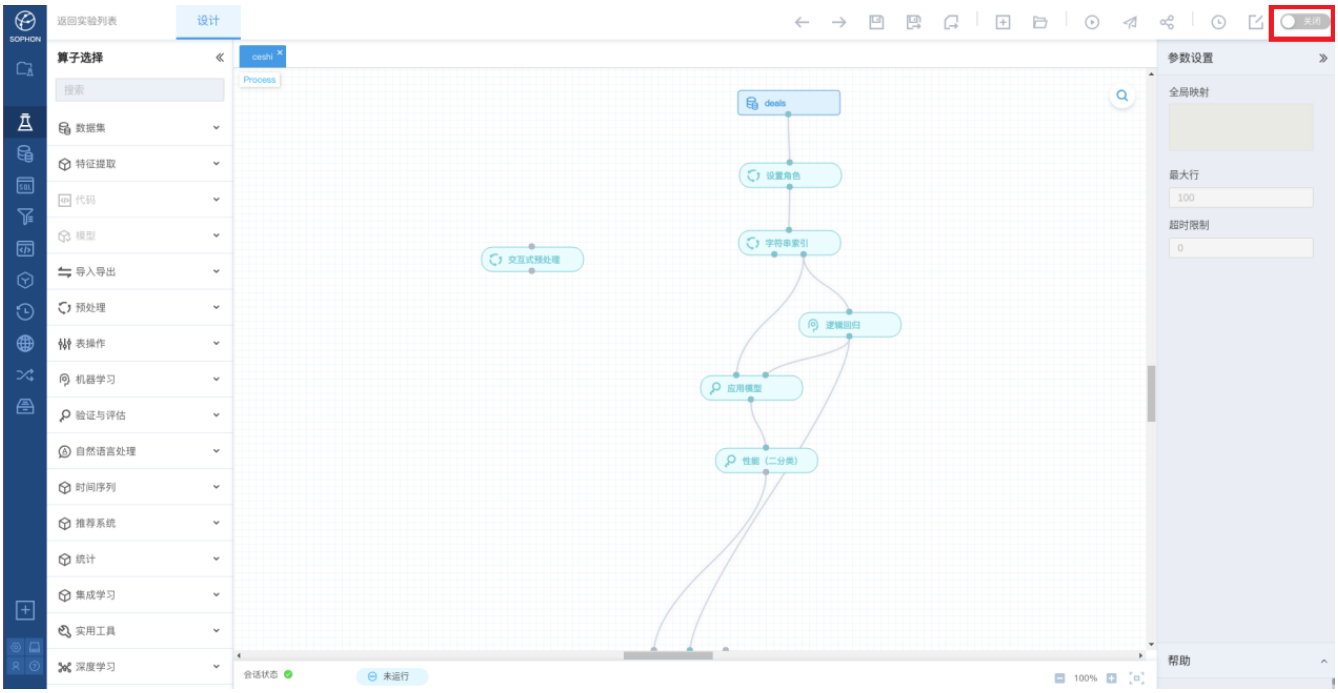


1.2. 新增项目协作

用户可添加组成员到项目协作中，2.0版本支持实验协作编辑。用户进入协作实验，点击编辑，若有其他用户在编辑，则进行提示；无其他用户编辑，则用户可编辑和保存，每次保存最新实验。
右上角显示蓝色“开启”按钮，表示当前用户可编辑：



右上角显示灰色“关闭”按钮，表示当前用户不可编辑：

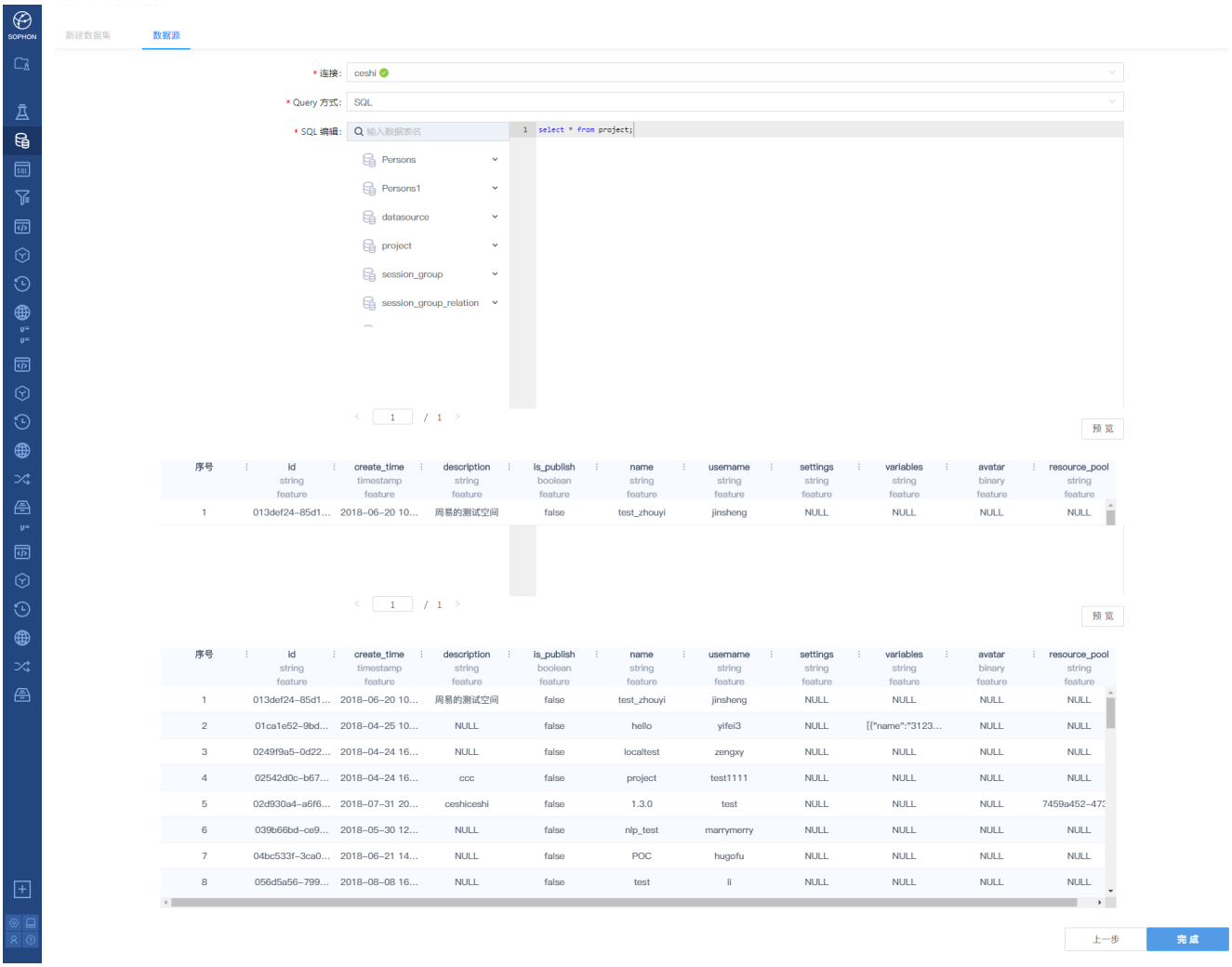


若有其他用户正在进行编辑，则会出现以下提示：

1.3. 优化数据集功能

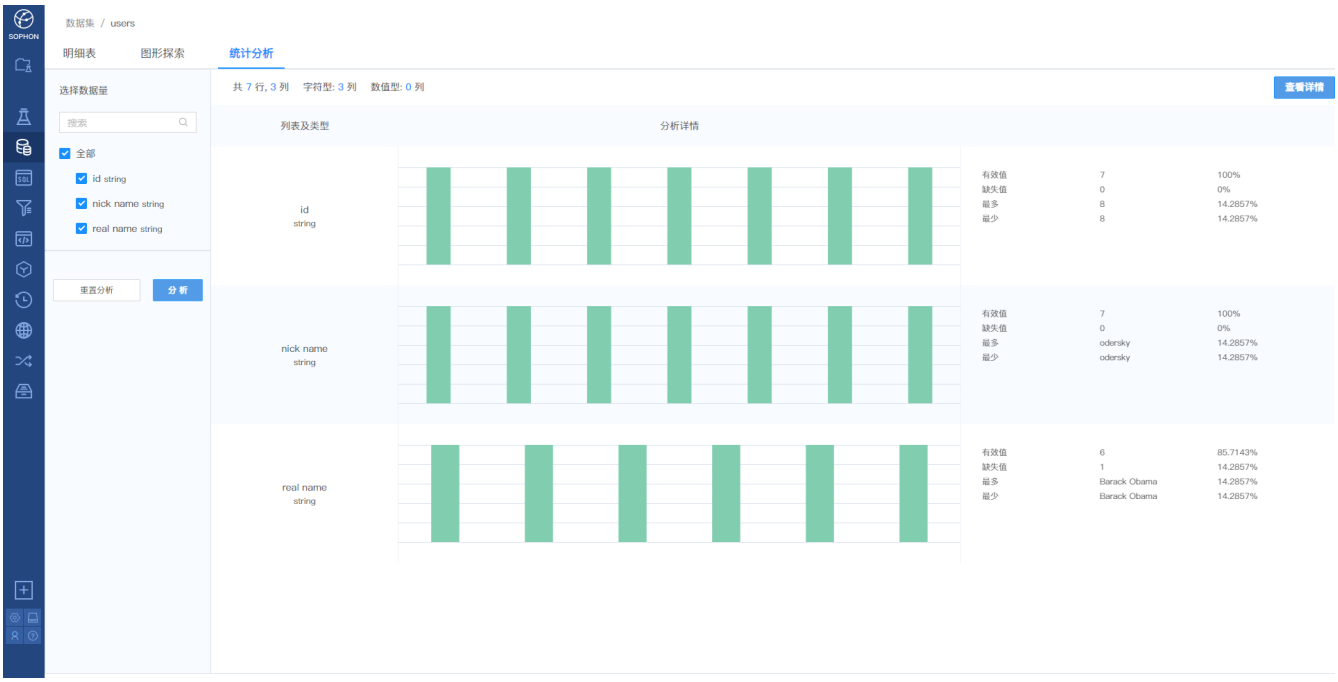
1.3.1. 新增多种支持的数据类型

- 支持从文件管理的集群分布式文件处导入数据集，用户可在文件管理的集群分布式文件处上传JSON、Parquet、ORC、Text以及csv格式的数据集
- HDFS连接新增支持JSON、Parquet、ORC以及Text数据集类型
- 本地导入图片，选择上传图片即可实现
- 数据库新增数据集，支持表导入和SQL编辑后的表导入两种方式



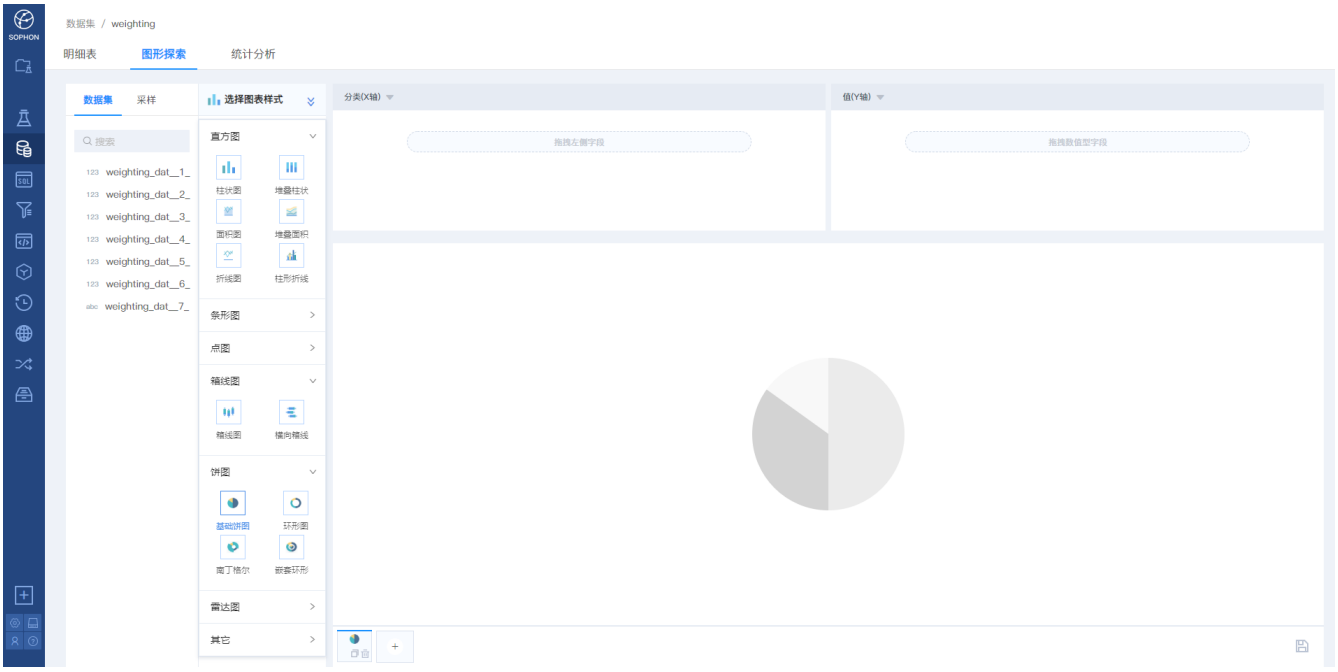
1.3.2. 新增数据集统计分析查看

- 新增统计分析模块，便于用户查看数据集管理的数据统计分析情况，含数据集中的行列显示、数据有效值和缺失值分布情况、统计最大/最小值/四分位数/平均值/方差等统计指标。点击查看详情，可查看所有字段的统计分析情况。



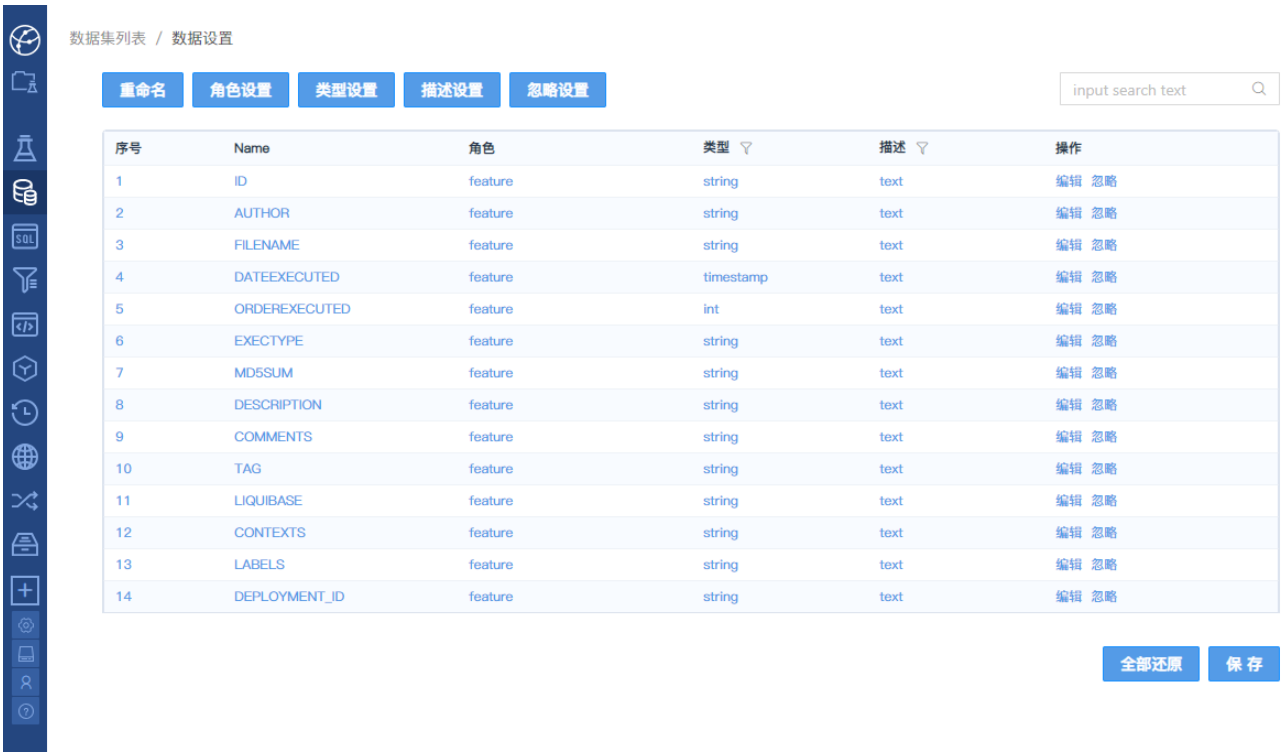
1.3.3. 优化数据集图形可视化

数据集查看明细和实验结果中，图形探索新增箱线图、饼图等14个图形。



1.3.4. 优化数据集设置模块

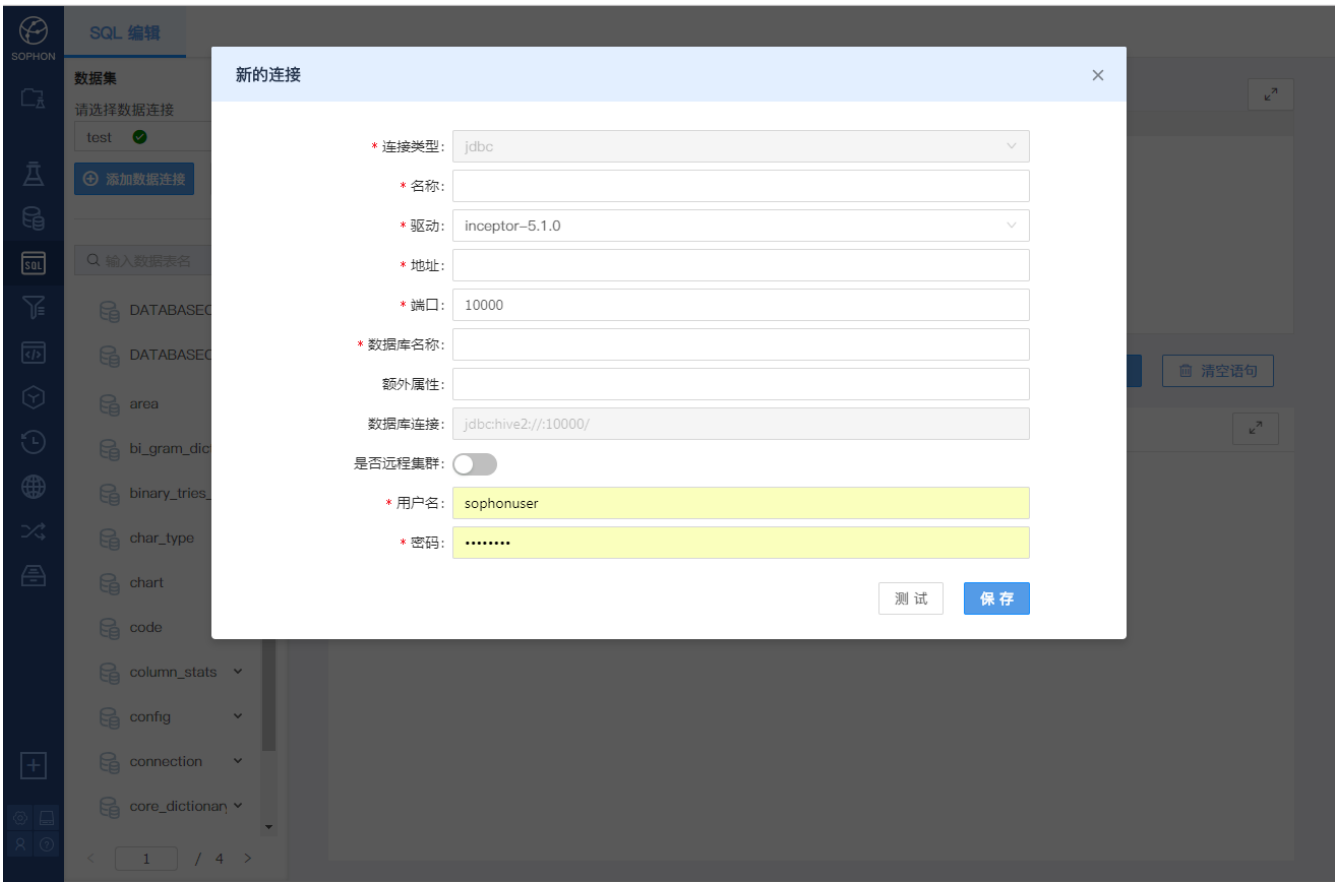
优化数据集重命名、角色、类型、描述、忽略等批量设置。



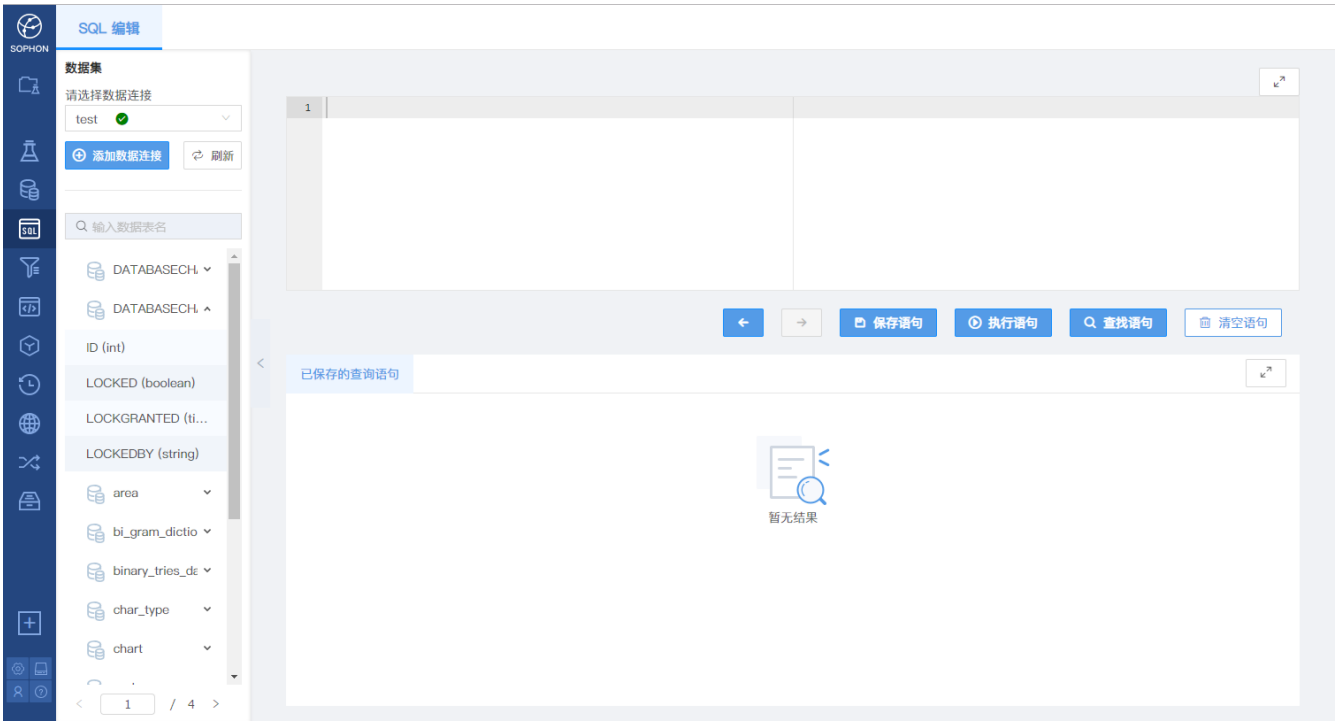
1.4. 新增SQL编辑模块

通过SQL编辑模块，可实现从数据库抽取单个/多个数据表、清洗以及转化等数据处理流程。

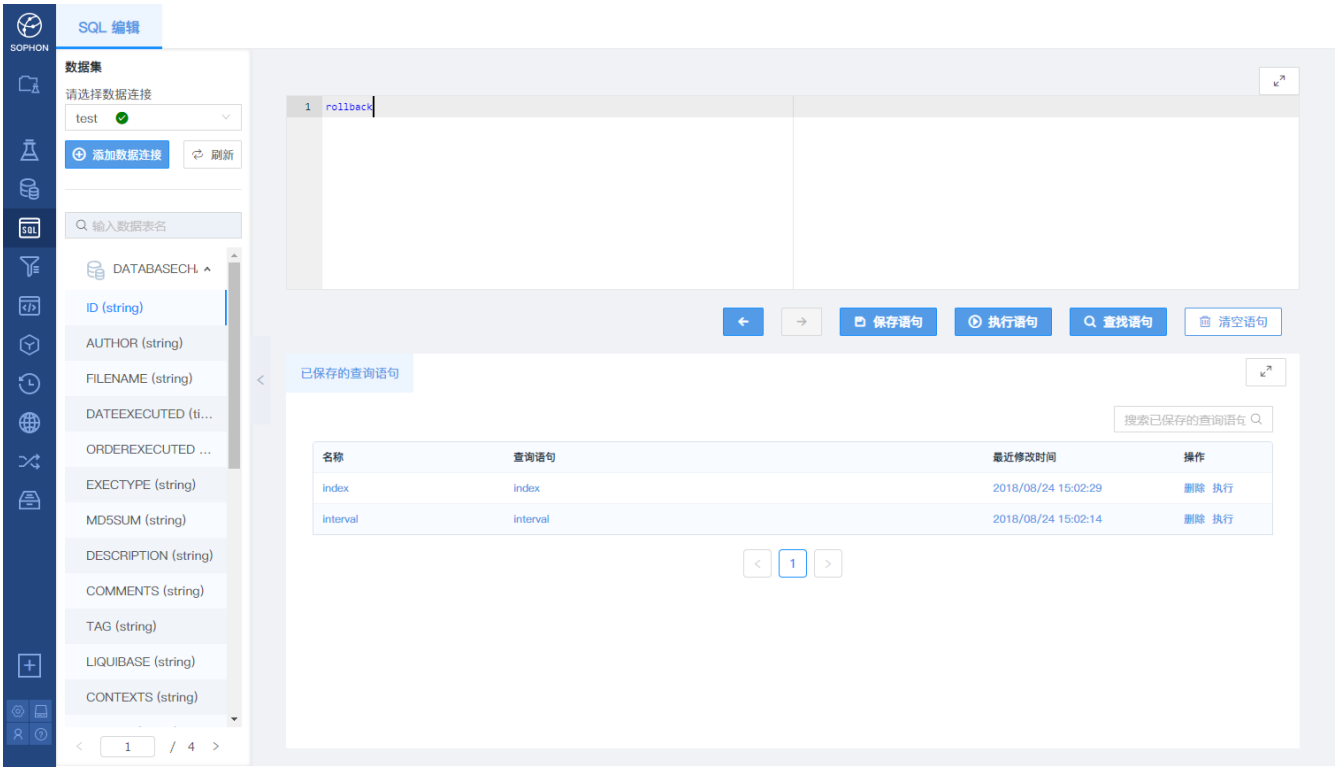
- 连接数据库。JDBC连接的数据库显示同步数据集模块的JDBC数据库连接，用户也可在此处新建JDBC数据连接。



- 查看数据表和字段。选中某一数据库后，显示该数据库下的数据表和字段明细，便于用户编辑SQL时参考。



- SQL编辑。SQL编辑支持撤回、恢复、查找（搜索）、全部保存语句、选中/全部执行、停止执行、清空语句。支持复制粘贴（Ctrl+C，Ctrl+V）等常用操作。用户可调用已保存的SQL语句。点击运行，将显示处理后的结果。

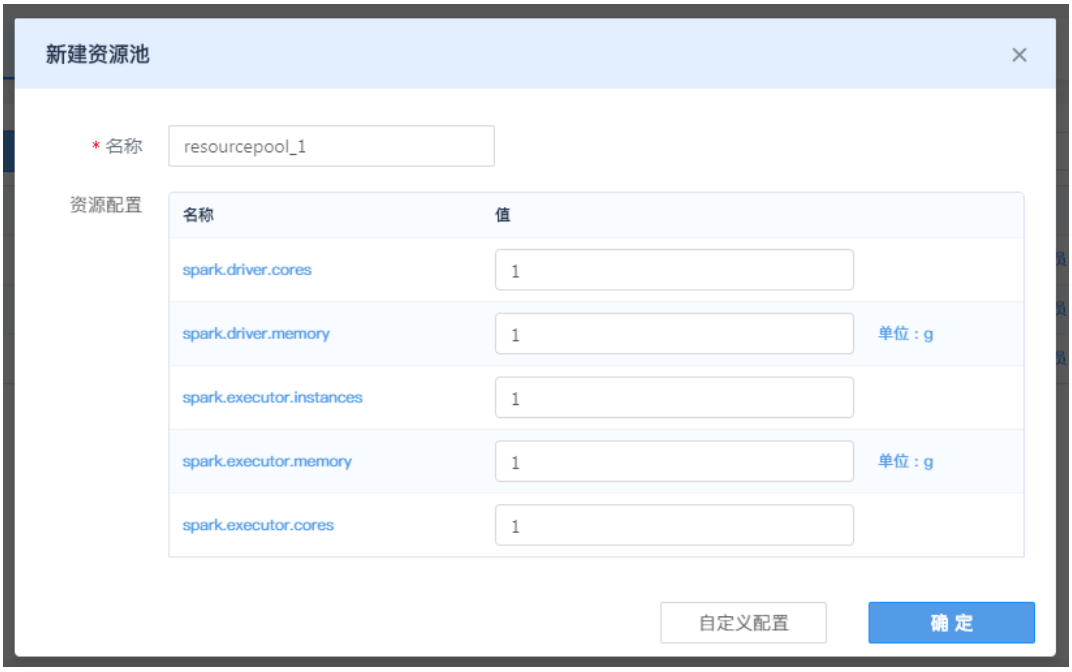


- 数据写入。经过SQL处理后的数据表，可采用SQL语句自带方式插入数据库中。

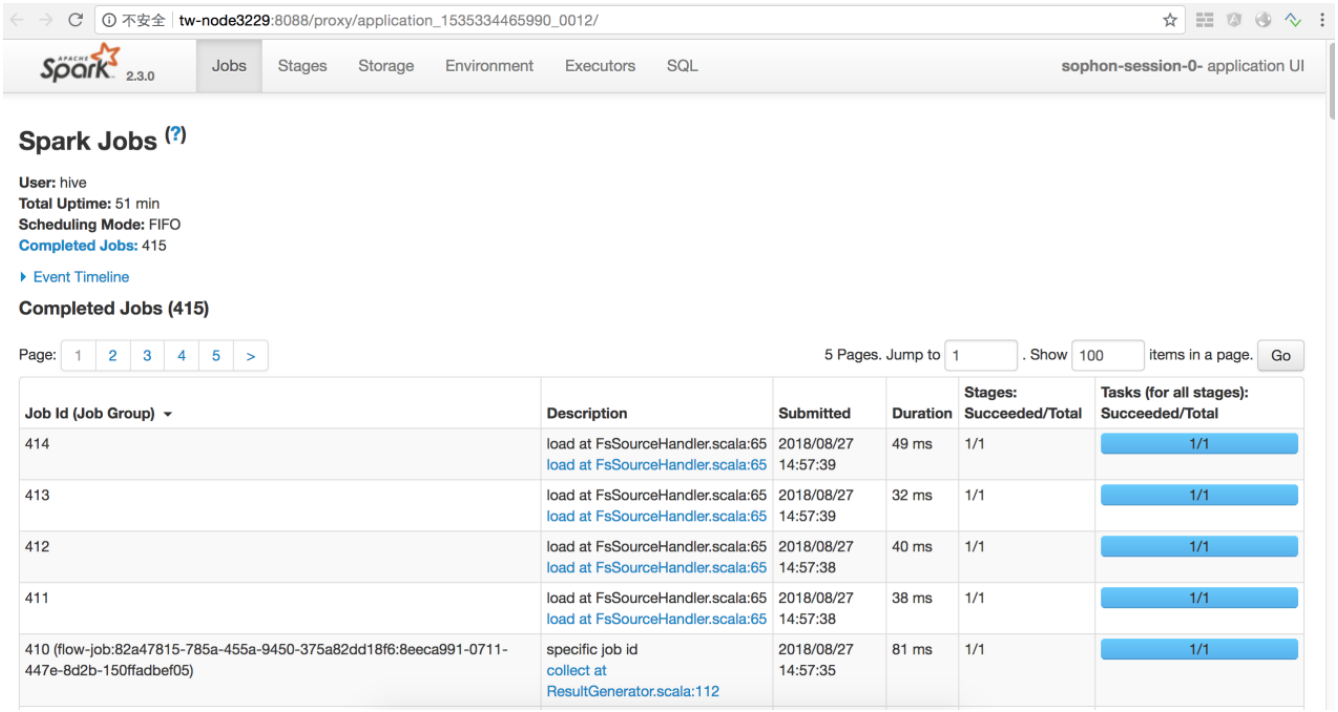
1.5. 优化管理中心

1.5.1. 资源池优化

管理员管理中心-资源池管理，新增资源池中新增中停止/开启，查看SPARK UI，自定义资源配置，删除资源池操作。



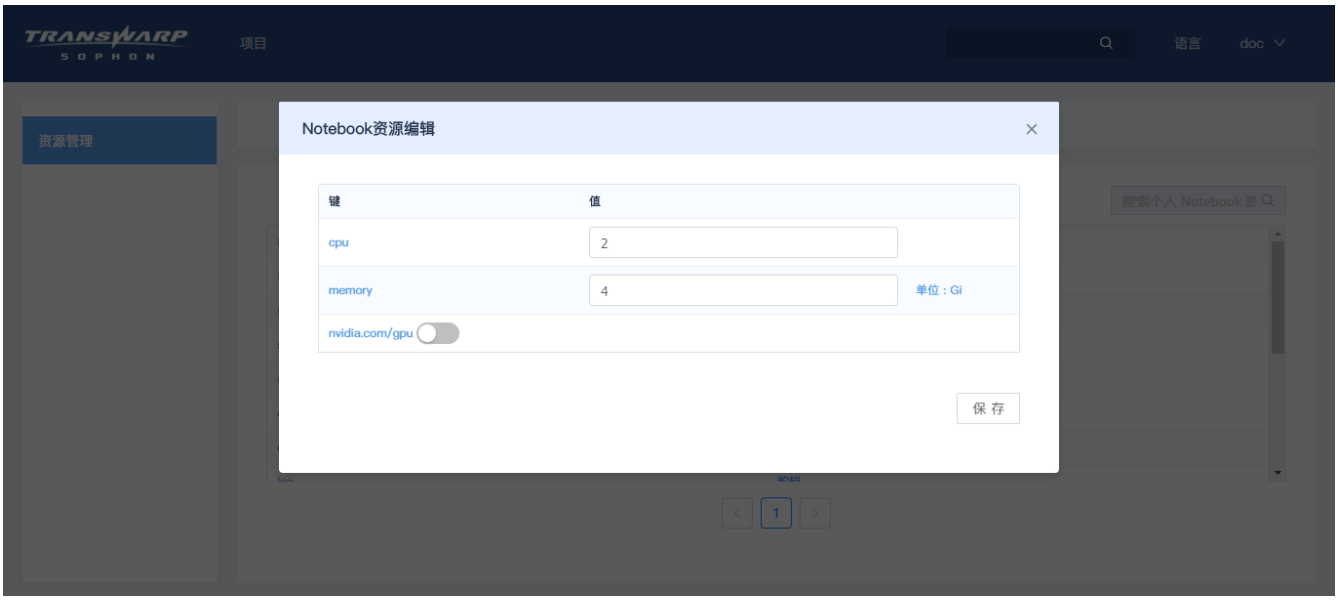
点击查看SPARK UI，新标签页打开SPARK UI页面，查看资源运行情况。



SPARK UI查看需要设置，详见本文4.3 访问Spark UI。

1.5.2. 新增个人Notebook资源管理

管理员对个人Notebook资源进行设置。
点击右上角“管理”可进入资源管理，选择个人Notebook资源，可对CPU的core、内存大小进行设置。若用户使用GPU版本的Sophon，则可根据需要配置GPU。

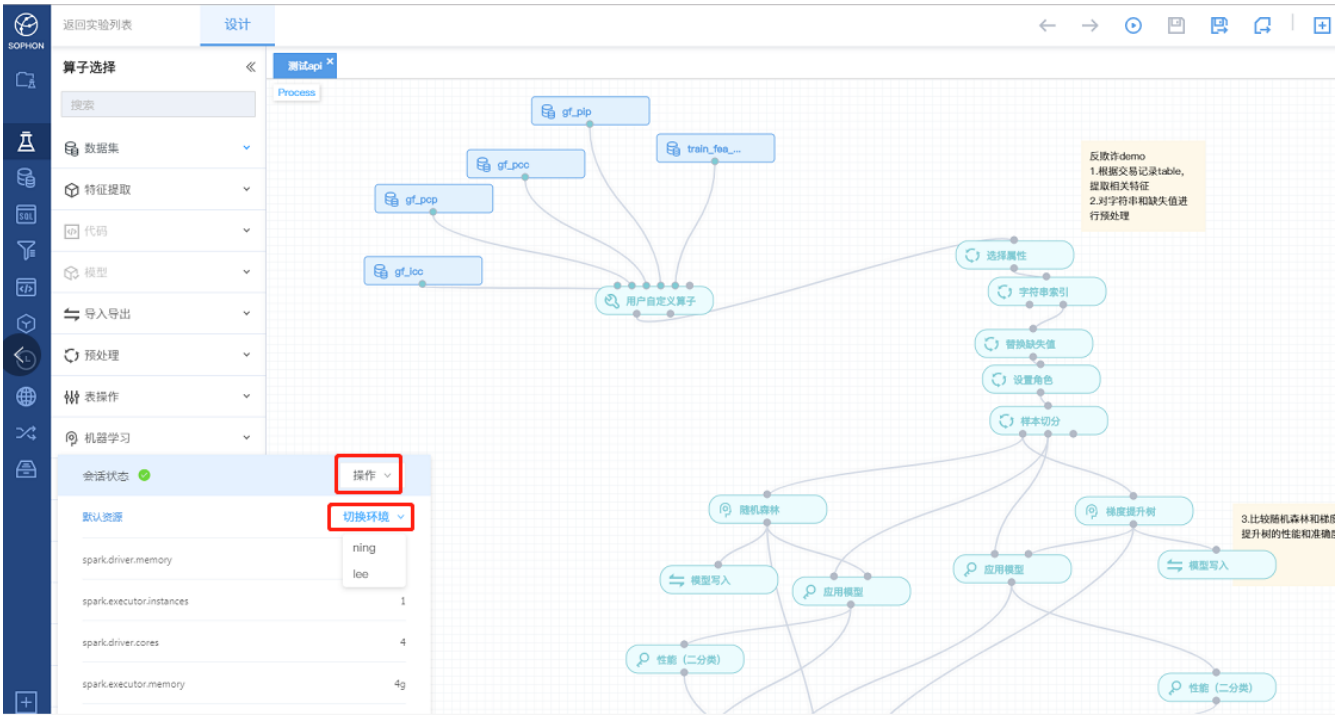


1.6. 新增个人中心查看被分配的资源

个人中心-我的资源，查看个人被管理员分配的资源池和个人Notebook资源。

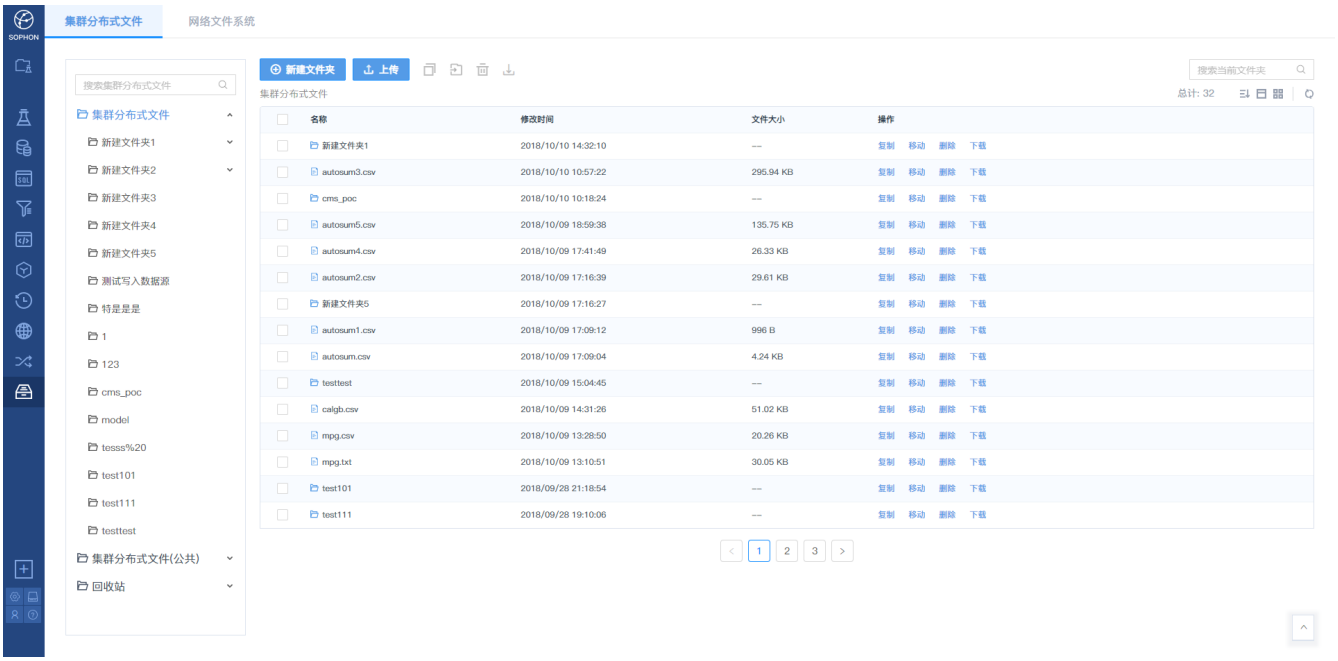
1.7. 会话管理支持资源池切换

会话启动失败可以手动重启该资源池或者切换其他资源池。



1.8. 新增文件管理

- 文件管理针对于用户本地上传和所有项目中形成的文件进行增删改等操作。文件管理中的内容可供所有项目使用。
- 支持集群分布式文件和网络文件系统，有个人文件和公共文件两种类型。集群分布式文件中，用户可上传数据集，在新增数据集时选择从文件系统导入方式。Notebook中产生的文件会在文件管理的网络文件系统中显示。
- 个人文件夹：支持单个/批量复制、移动、下载、删除；支持单个/批量上传文件；新建文件；支持搜索、排序功能。
- 公共文件夹：仅有管理员拥有上传和增加项目中的文件到公共区域的权限，其他功能管理员和个人操作相同。
- 回收站：单个/批量还原和删除，清空回收站。

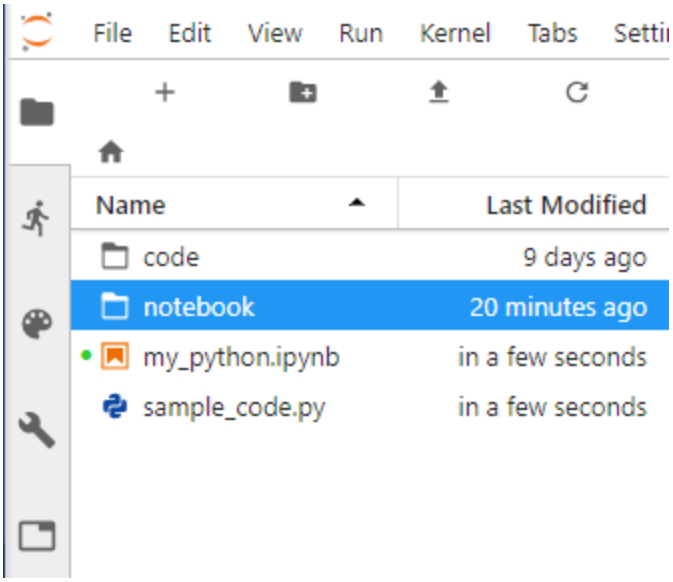


关于如何安装网络文件系统，请参考本文4.2 网络文件系统安装说明。

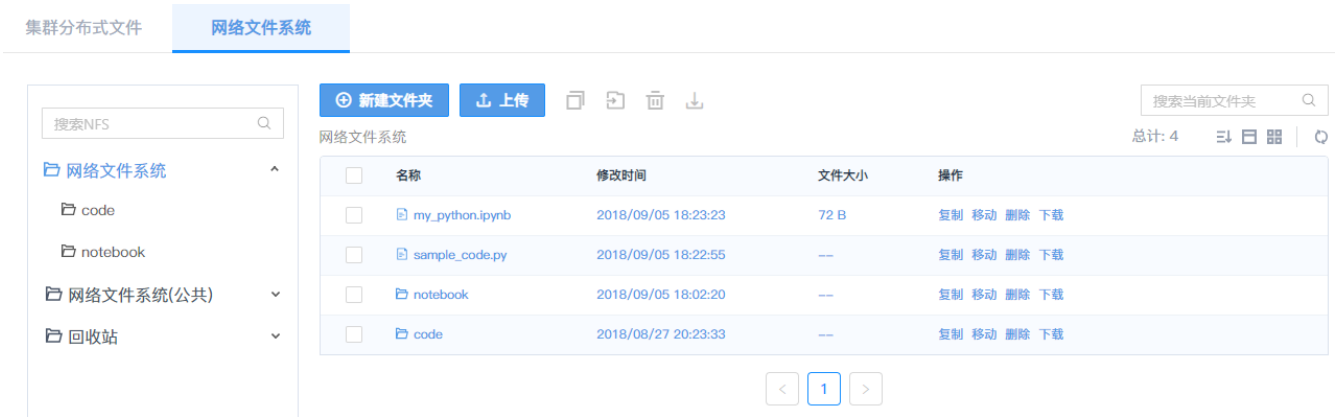
1.9. 代码优化

1.9.1. Notebook功能升级

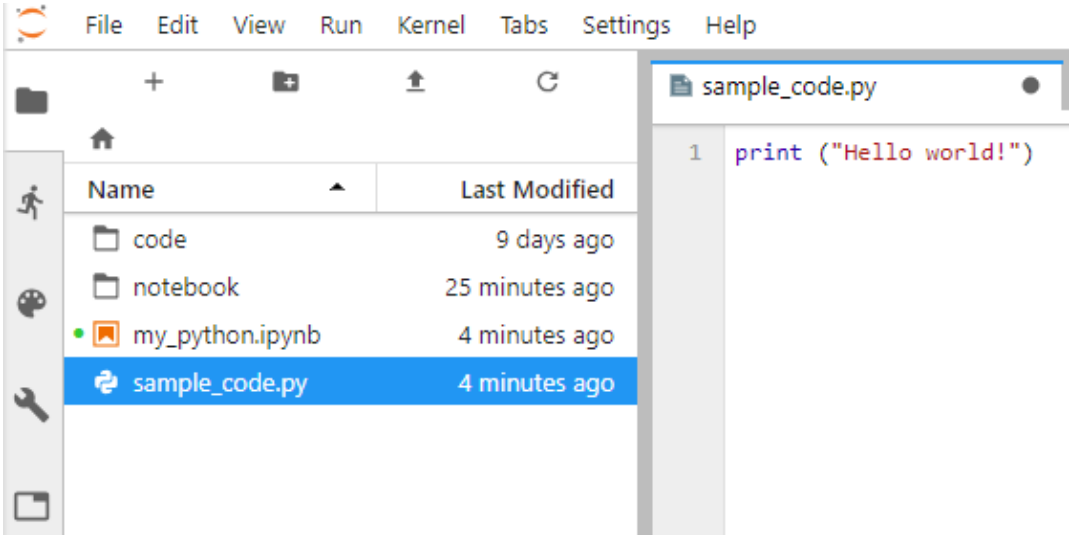
- Notebook编辑页面新增文件管理。Notebook中产生的文件会在文件管理的网络文件系统中显示，便于用户在Script中引用和下载到本地。
  Notebook中的文件管理界面：



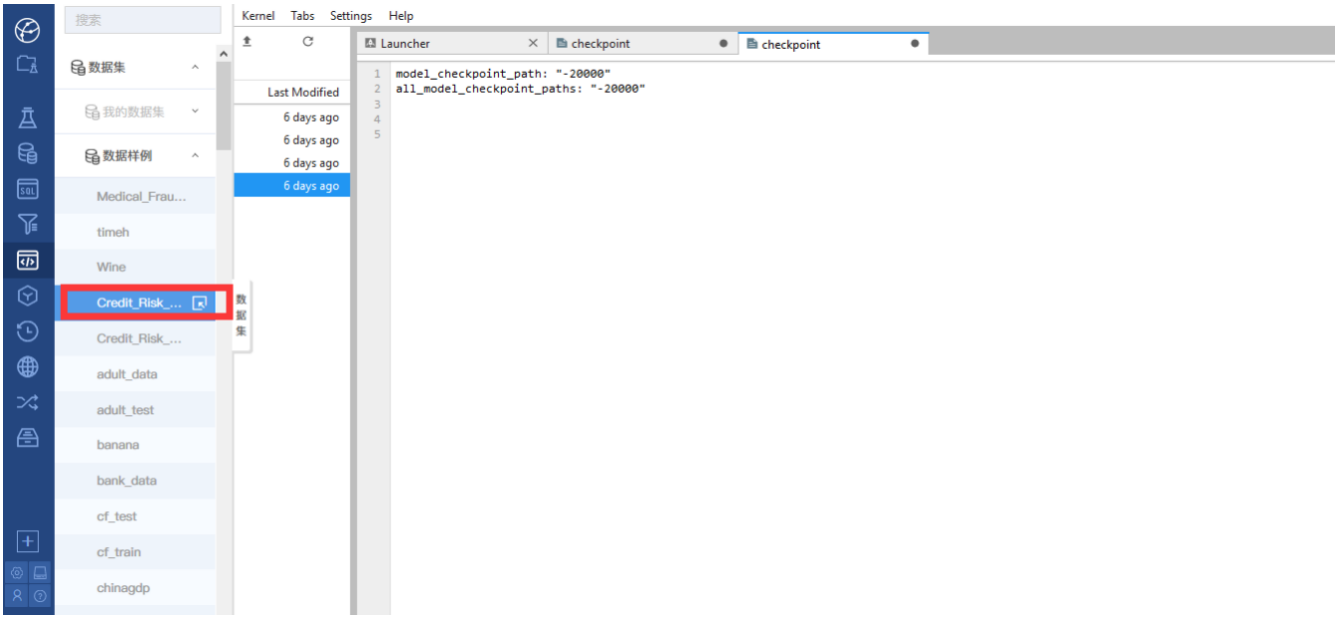
Sophon文件管理的网络文件系统部分：



打开文件管理中的代码，即可在新的标签页显示该代码内容：



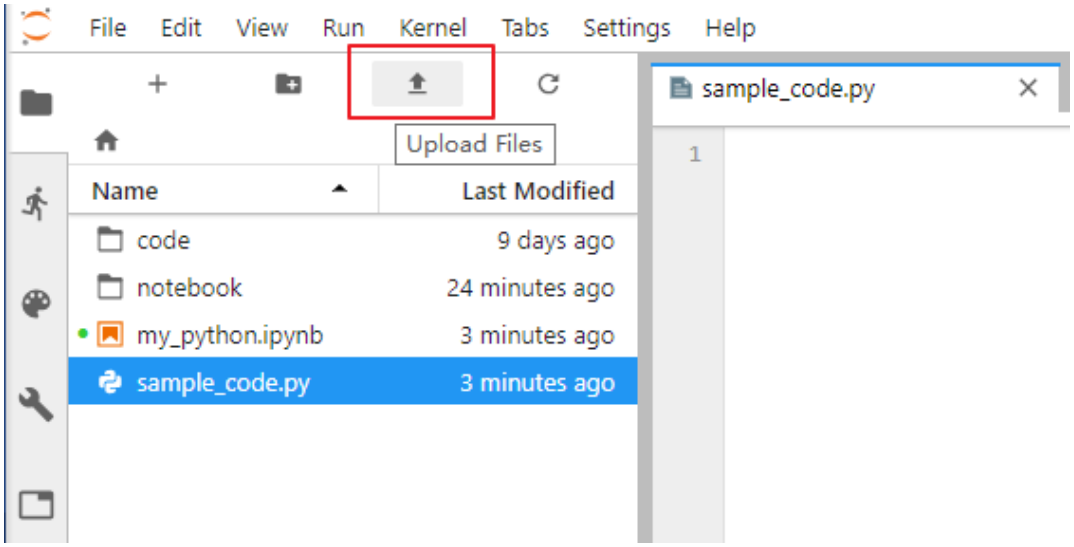
- 深度学习Notebook完善
  Notebook代码模块支持多种深度学习框架以及相关的图像处理包，支持GPU版本和CPU版本。框架支持TensorFlow 1.8.0、OpenCV 3.4.0、MXNet、PyTorch和Caffe。
  代码和数据通过分布式存储方式管理，支持多种框架对图片和文本数据的训练、预测、 模型的保存和存储等；支持多种模式数据的读写,如hdf5、lmdb等；支持TensorFlow模型的展示。
- 新增支持MXNet 1.2.1、Caffe 1.0.0、PyTorch 0.4.0以及Keras 2.2.2
- 新增Kernel管理
- 新增单机版Python和R的库
- 支持Hadoop分布式读取和写入
  Spark，PySpark3和SparkR中执行Spark相关的操作去读取数据，会得到spark dataframe类型的结果，可直接进行后续的分布式计算，无需用户对数据再次进行处理或者转换等操作。
- 支持传统数据库读取和写入
  支持读取和写入MySQL、SQL Server和Oracle数据库。
- 可通过左边栏的数据集插入Sophon中的数据集



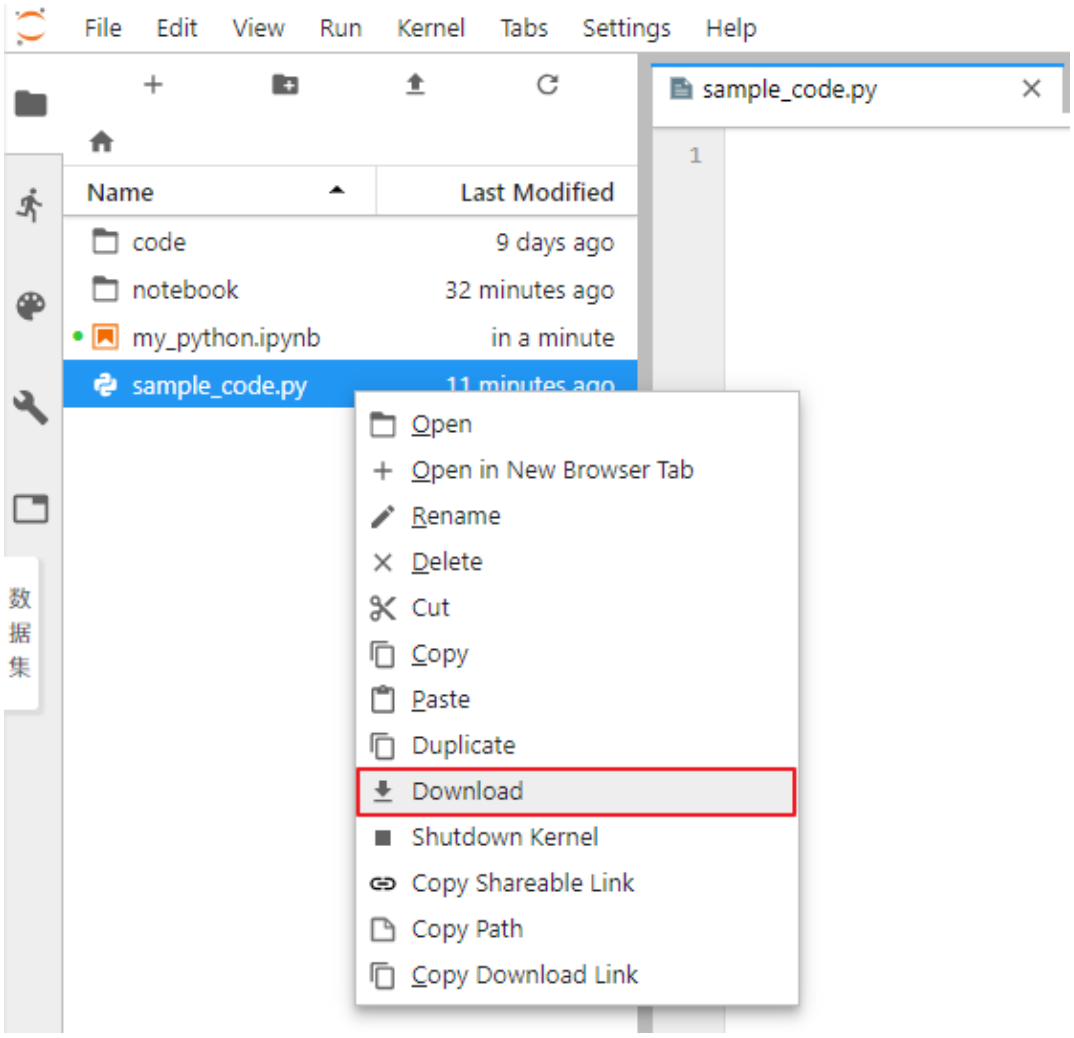
- 支持终端调试
- 支持Python、Spark及R语言单步调试

- 支持上传、下载文件

上传：



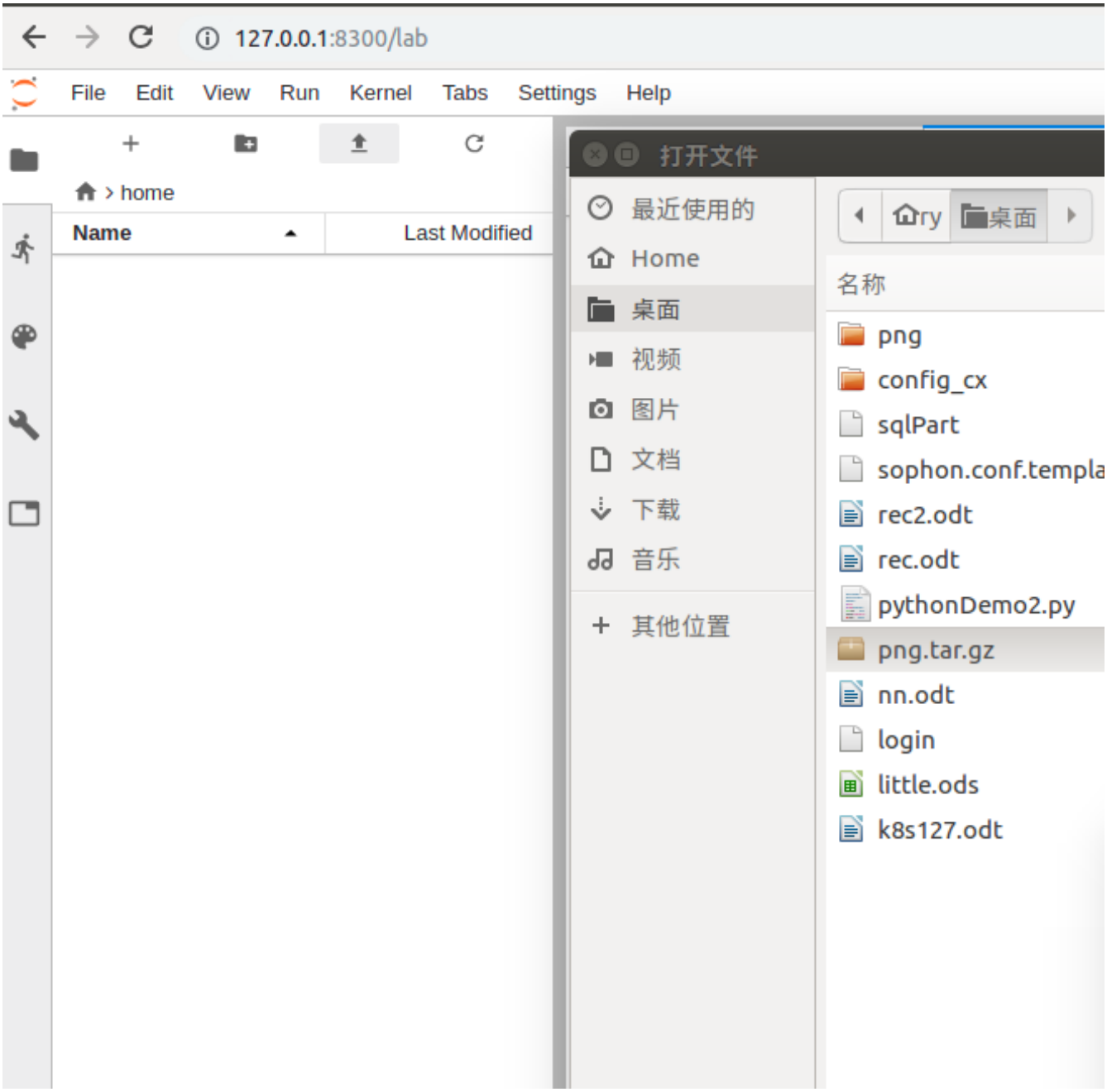
下载：



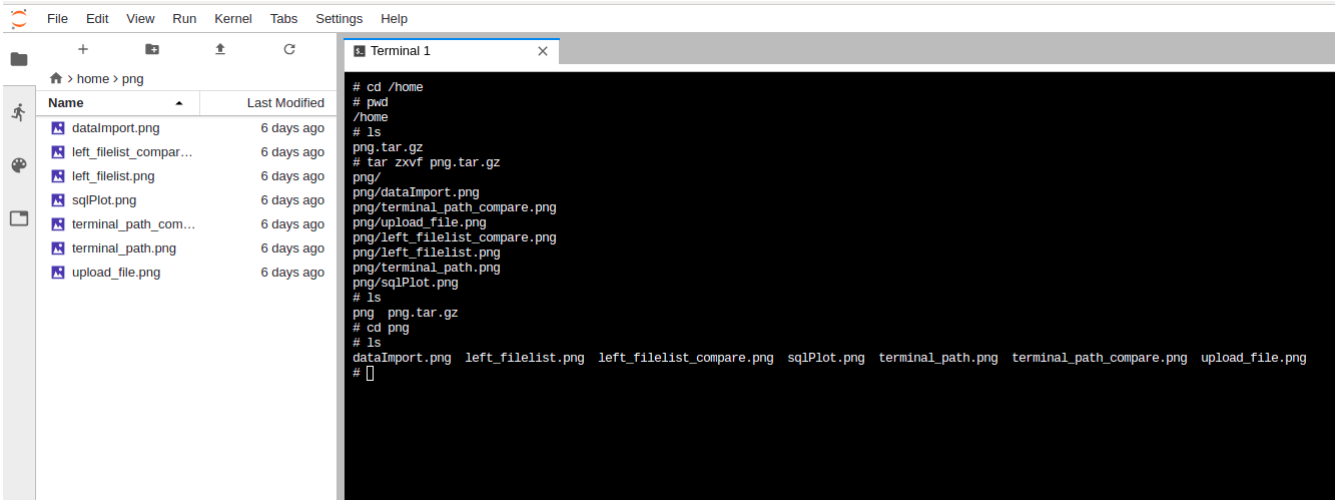
- 支持上传文件夹

JupyterLab部分不支持直接上传文件夹的功能，因此可以选择先将本地文件夹进行压缩操作，再上传压缩文

件：



进入Terminal，进行解压操作，完成后，可在左侧标签栏同一目录结构下看到该文件夹：



- 支持读取本地文件

以Python 3为例，读取本地文件获取数据：

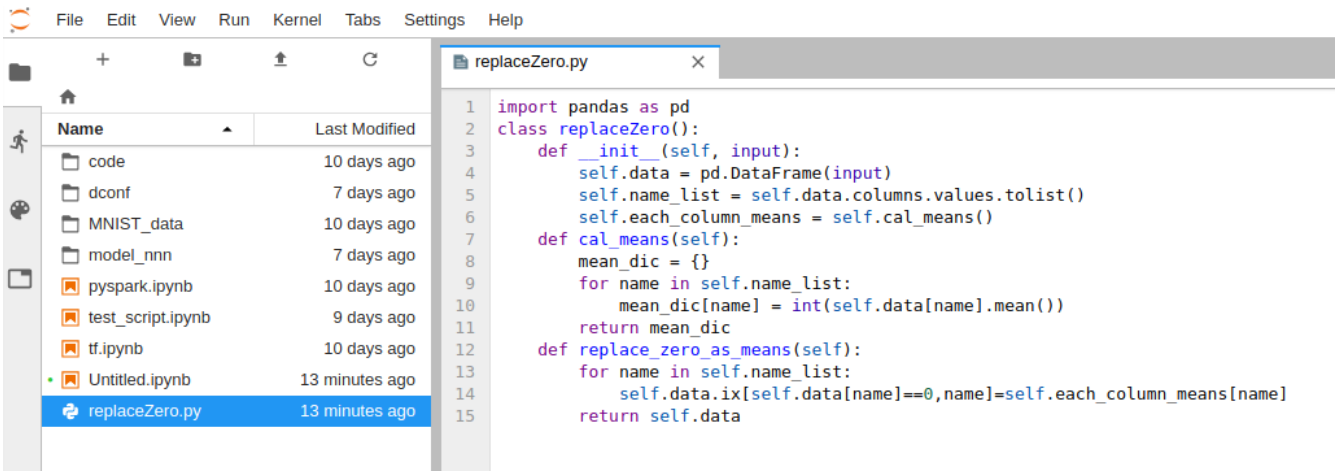
```
import pandas as pd
df1 = pd.read_csv("./data.csv")
#spark读取pandas dataframe,形成spark dataframe
sparkDF = spark.createDataFrame(df1)
sparkDF.show()
```

以R语言为例，读取本地文件获取数据：

```
data3<-read.csv('data.csv',sep=',',header=TRUE)
#转成SparkDataFrame结构数据
df <- as.DataFrame(data3)
```

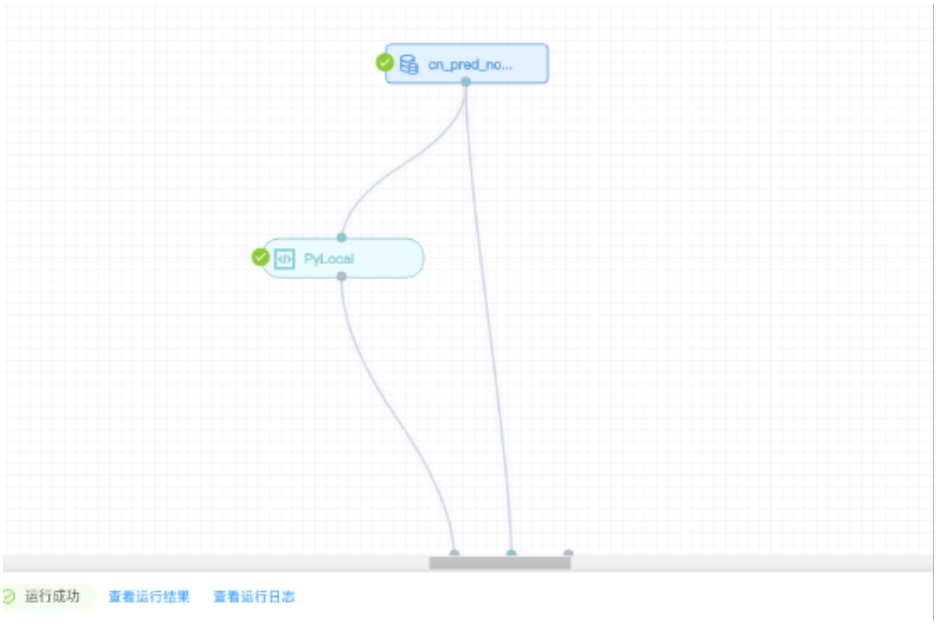
1.9.2. Script代码编辑可以引用Notebook文件

文件管理系统中可以存储用户自己定义的py代码，如图在replaceZero.py中定义了一个类,用每一列的均值替换pandas DataFrame中所有为0值，这样我们可以对实验中数据进行缺失值替换。



新建script代码，在代码中导入定义的replaceZero，使用replaceZero类对数据进行预处理，如图：

在实验中创建如下的实验，设置参数，如下，点击运行即可验证：



最后对结果进行对比，注意红框中值的不同：
替换前：

指标5 int feature	指标7 int feature	指标24 int feature	指标12 double feature
1100	9	0	82.00
1200	10	0	64.00
0	8	0	64.00
700	9	0	59.00
800	34	1200	47.00
0	54	3000	0.00
800	37	7600	0.00
800	54	700	0.00
600	10	1000	0.00
700	17	600	46.00
800	12	400	59.00
600	12	0	74.00
600	21	0	60.00
800	9	0	58.00
800	26	0	58.00

替换后：

指标5 bigint feature	指标7 bigint feature	指标24 bigint feature	指标12 double feature
1100	9	51289	82.00
1200	10	51289	64.00
1263	8	51289	64.00
700	9	51289	59.00
800	34	1200	47.00
1263	54	3000	71.00
800	37	7600	71.00
800	54	700	71.00
600	10	1000	71.00
700	17	600	46.00
800	12	400	59.00
600	12	51289	74.00
600	21	51289	60.00
800	9	51289	58.00
800	26	51289	58.00

1.9.3. 支持Zeppelin代码

如需打开Zeppelin，有以下两种配置方法：

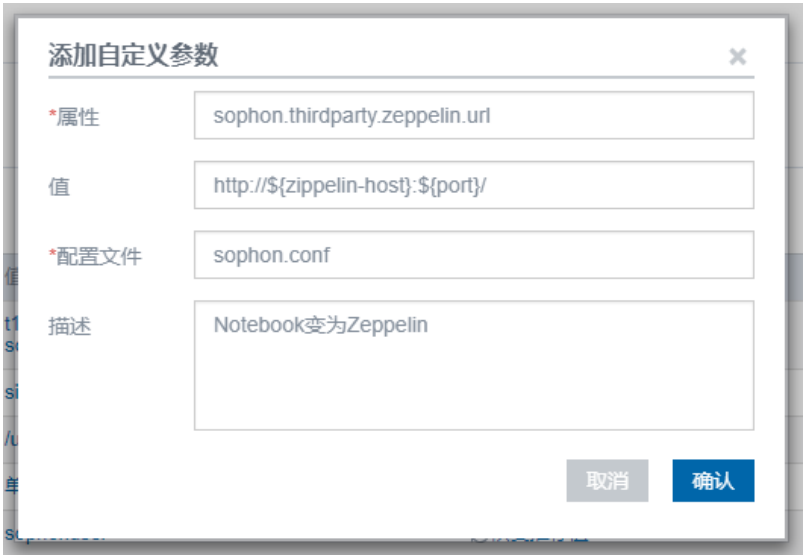
1 服务器端配置

用户在sophon.conf文件中添加下列配置，会触发Zeppelin的跳转，其中url的内容为用户根据实际情况添加。
sophon.thirdparty.zeppelin.url = http://${zippelin-host}:${port}/

2 Manager页面配置

进入Manager中的Sophon配置页面，单击右上角的“添加自定义参数”：

向配置文件添加下图所示的属性：



url需根据实际情况填写。
添加完成后，需要配置服务，重启Sophon。

1.10. 模型管理支持深度学习模型的导入导出和可视化查看

- 用户可导入深度学习模型，选择文件路径为Notebook编辑时存入文件管理的位置

- 用户可导出深度学习模型到本地
- Sophon 2.0暂不支持深度学习模型的分享、应用到实验
- 支持深度学习模型可视化使用TensorBoard查看

1.11. 教程优化

Sophon 2.0提供了丰富而完整的教程。用户可在软件中获取详细的算子信息，还可获取丰富的示例信息等。

- 增加资源池配置指引。详见【帮助-教程-2.11：资源参数调优】

2.11. 资源参数调优

spark的资源使用主要有以下5个参数来控制

- spark.driver.memory： driver进程使用的内存
- spark.driver.cores： driver进程的cpu核数
- spark.executor.instances: executor进程的数量
- spark.executor.memory：每个executor进程使用的内存数量
- spark.executor.cores：每个executor的cpu核数

资源的配置需要根据集群环境来调整。主要需要调整executor的配置。executor的配置有两种极端方案。

- 每个core一个executor。这种分配方式的问题是无法享受到JVM多任务处理的便利，以及spark的共享、缓存变量 需要在每个jvm保存一份，浪费内存资源
- 每个节点一个executor。即每台机器只有一个jvm进程。此种情况必然会调大memory，而过大的memory会影响GC效率， 建议spark.executor.memory不要高于64G

综合这两种情况，executor一般可以如下配置：

- spark.executor.cores = 5，可以满足HDFS的吞吐量
- spark.executor.instances = (集群所有core的数量 - spark.driver.cores）/ spark.executor.cores
- spark.executor.memory = 单台节点的内存数量 / (spark.executor.instances / 节点数量） * 0.9， 最后0.9这个参数用于调节heap的overhead等情况

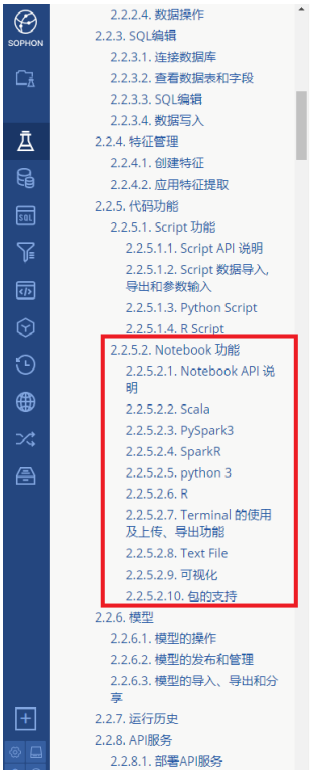
使用这种方式则集群基本用满，只能使用一个spark资源池。如果需要多个资源池同时使用，则主要可以调节instances和memory的值。

而driver则没有非常明确的配置。如果集群不需要做collect等收集大量数据到driver的操作，则1 core和1G的内存 也是足够的。如果不放心的话则可以适当增加spark.driver.memory，以防内存溢出。

2.12. tensorboard模型展示

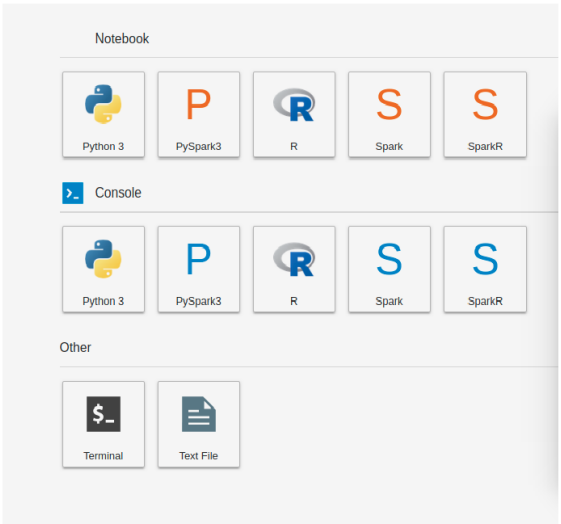
notebook生成的神经网络模型，可以通过导入到模型管理中，通过tensorboard来查看模型的参数及效果

- 增加Notebook使用指导教程。详见【帮助-教程-2.2.5.2：Notebook功能】



2.2.5.2. Notebook 功能

通过: 代码模块 →打开Notebook, 可以在页面上laucher中Notebook支持五种代码编辑方式, Other中有Terminal和Text File两种选项.



Sophon-web支持在线的代码编辑运行, 面向不同的用户群体, 目前支持scala, R 和python3 三种语言,均可直接使用spark相关的操作和算法库.

同时Notebook中还可以在Terminal中执行语句, 并且Text File中支持上传python3脚本文件进行运行. 本节先给出Spark, PySpark3和SparkR中的API, 再给出五种编辑方式各自常用的一些操作.并对Terminal和Text File两部分进行简单介绍, Terminal的介绍中会给出左侧标签栏的一些常用操作,如，上传文件、文件夹、导出文件等. 其中：Spark, PySpark3和SparkR中执行Spark相关的操作去

1.12. 其他优化

- 支持查看算子元信息

- 实验中算子复杂参数输入形式优化

复杂参数支持文本输入方式，便于用户复制粘贴参数列名或取值等参数快速输入。

- 可设置会话超时时间

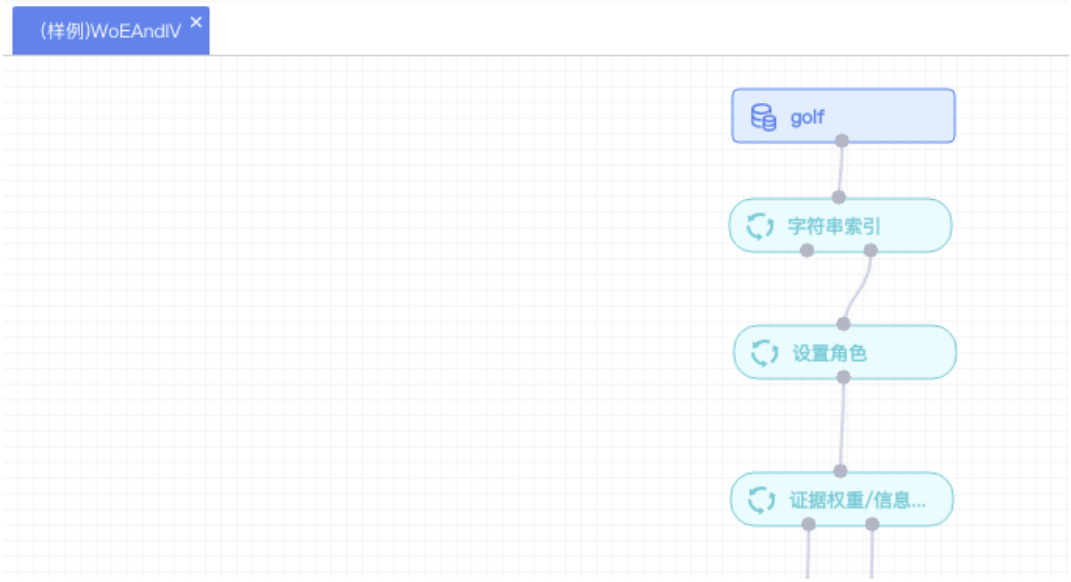
详细设置方法，请参考本文4.4 配置会话超时。

- 相关矩阵结果展现形式优化

2. 算子显著更新

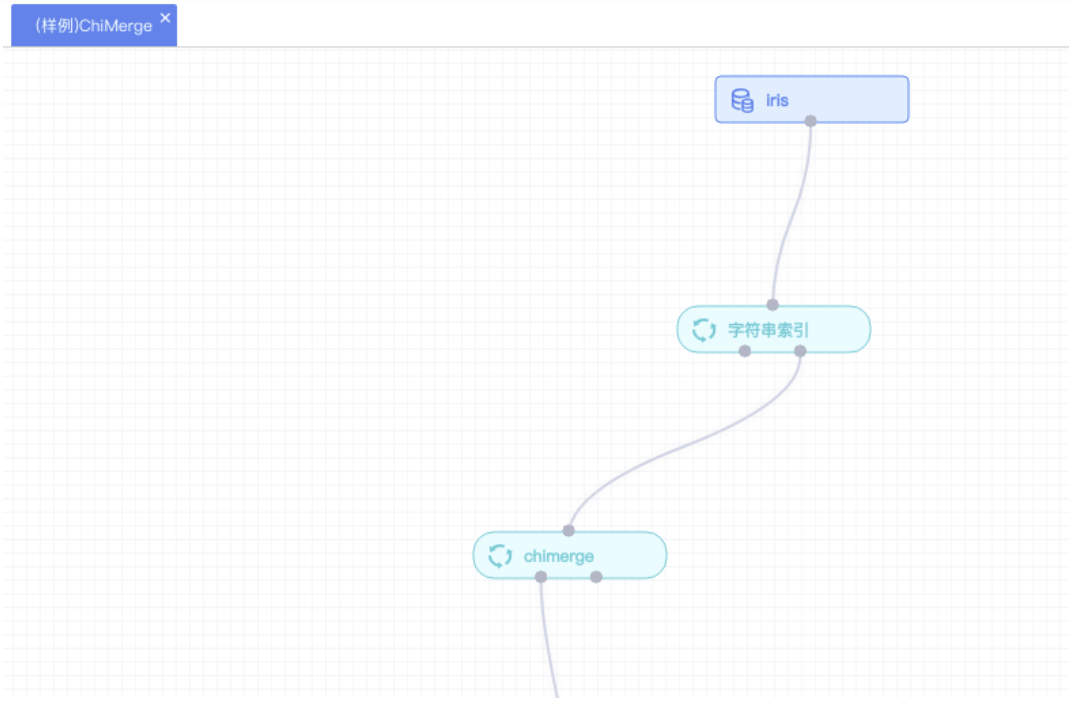
- 添加WoE/IV算子【算子名称：证据权重/信息价值】

WoE的全称是“Weight of Evidence”，即证据权重，是对原始自变量的一种编码形式；信息价值（IV）是一种可以用来衡量自变量的预测能力的指标。此算子计算属性的证据权重（WoE）值并进行变换。输入需要有label列并将连续变量离散化。



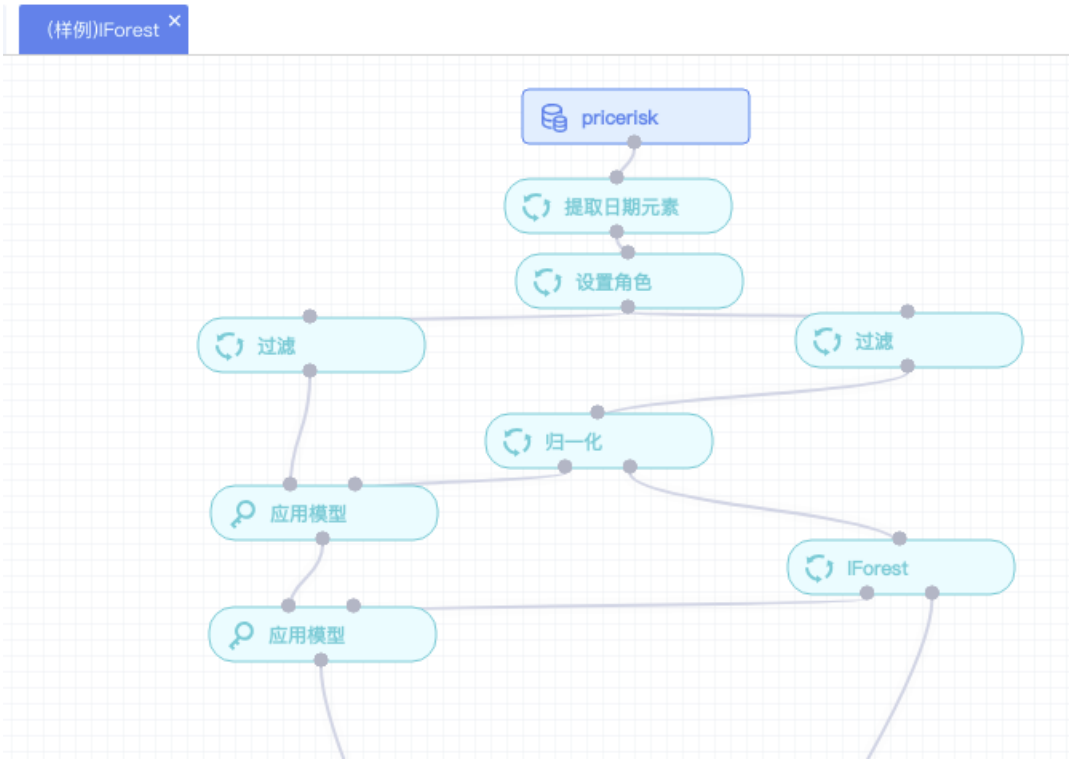
- 添加最优分桶（ChiMerge）算子【算子名称：chimerge】

ChiMerge分桶是一种基于 x 2的离散化方法,根据相邻区间的卡方值，对最小卡方的区 间进行合并，因为根据标签分布划分桶，因此要比常规分桶策略更好。



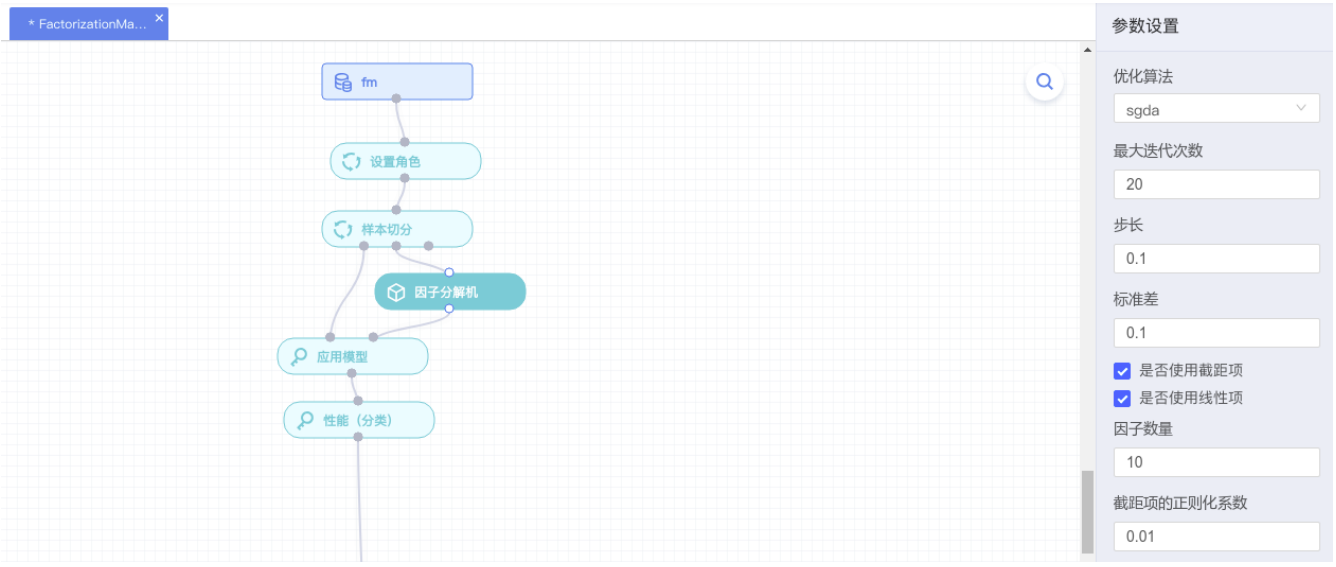
- 添加IForest【算子名称：IForest】

IForest是一种用于异常点检测的高效模型。



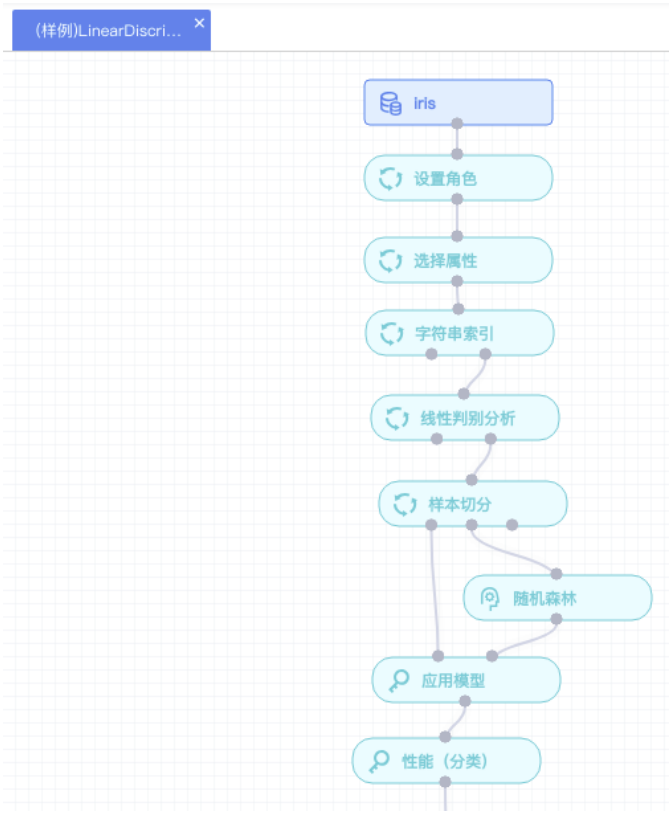
- 添加Adaptive Regularization for FM【算子名称：因子分解机】

新增采用自适应正则系数的批量随机梯度下降法(sgda)。



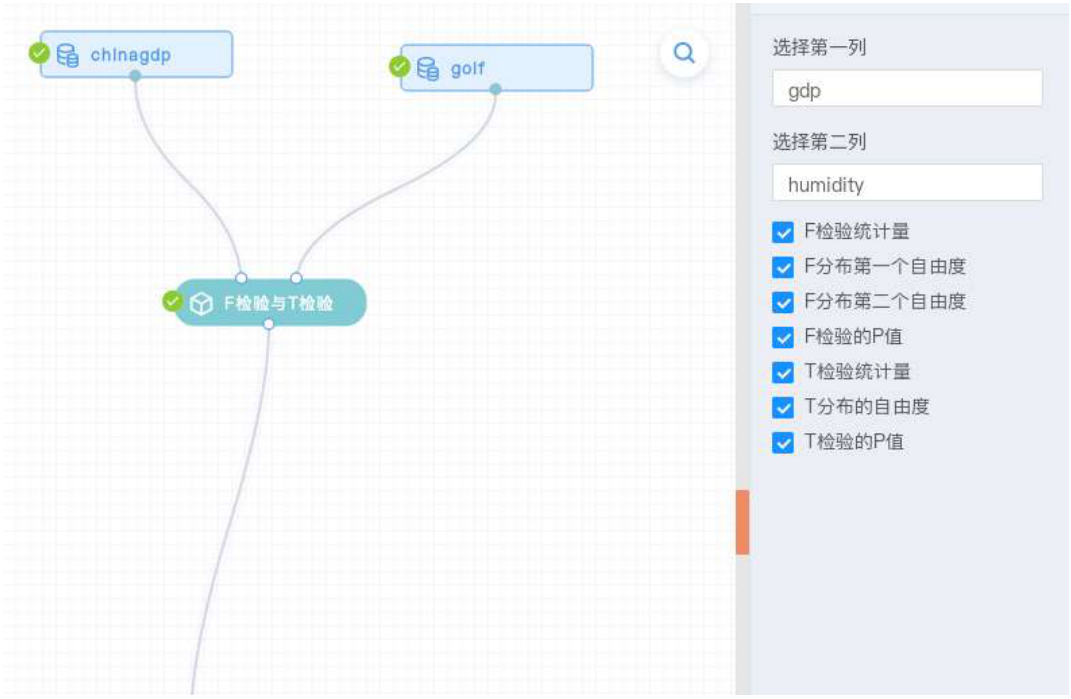
- 添加线性判别分析（Linear discriminant analysis）【算子名称：线性判别分析】

线性判别分析是一种广泛应用于降维的算法。注意，列的数目应小于样本数，且小于65536。



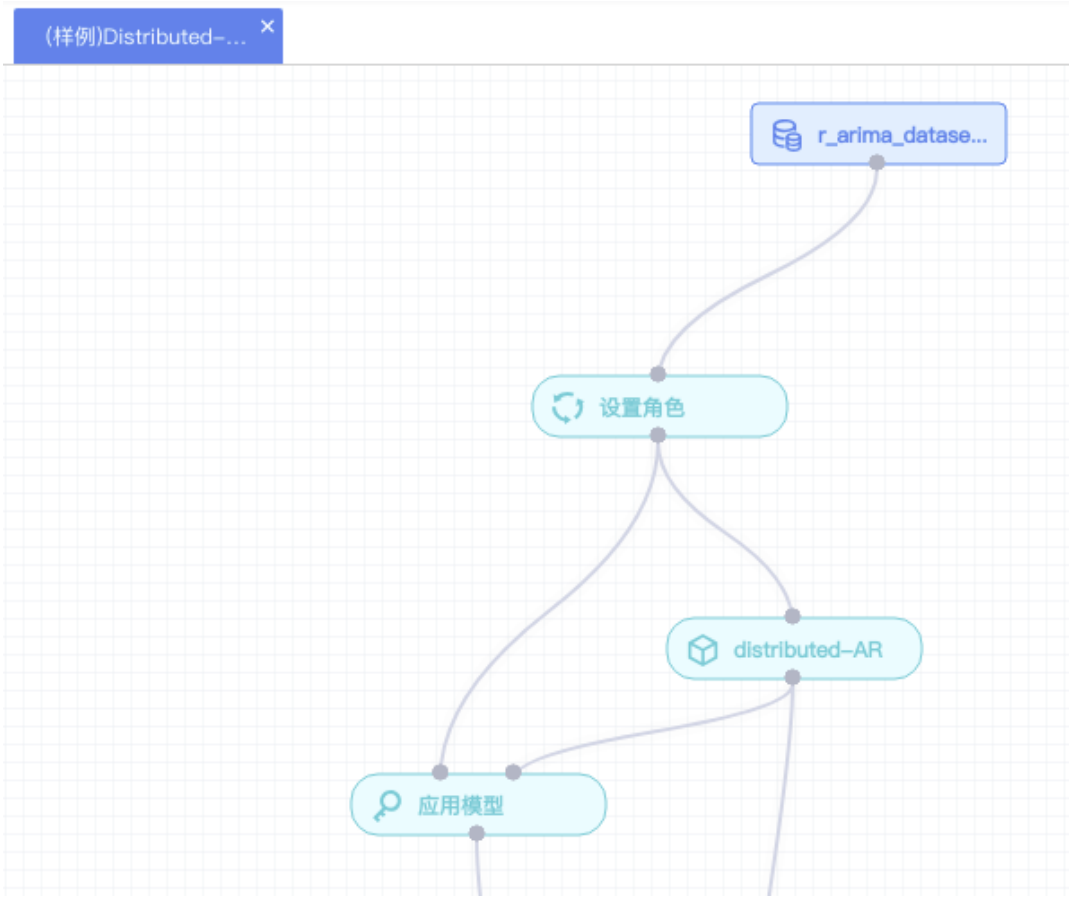
- 添加F-test和T-test算子【算子名称：F检验与T检验】

方差齐性检验与平均数差异检验,首先需要进行方差齐性检验,查看两组数据是否有显 著的方差差异,若有则不进行平均数差异检验.平均数差异检验目的为检验两组数据均 值是否有显著性差异。



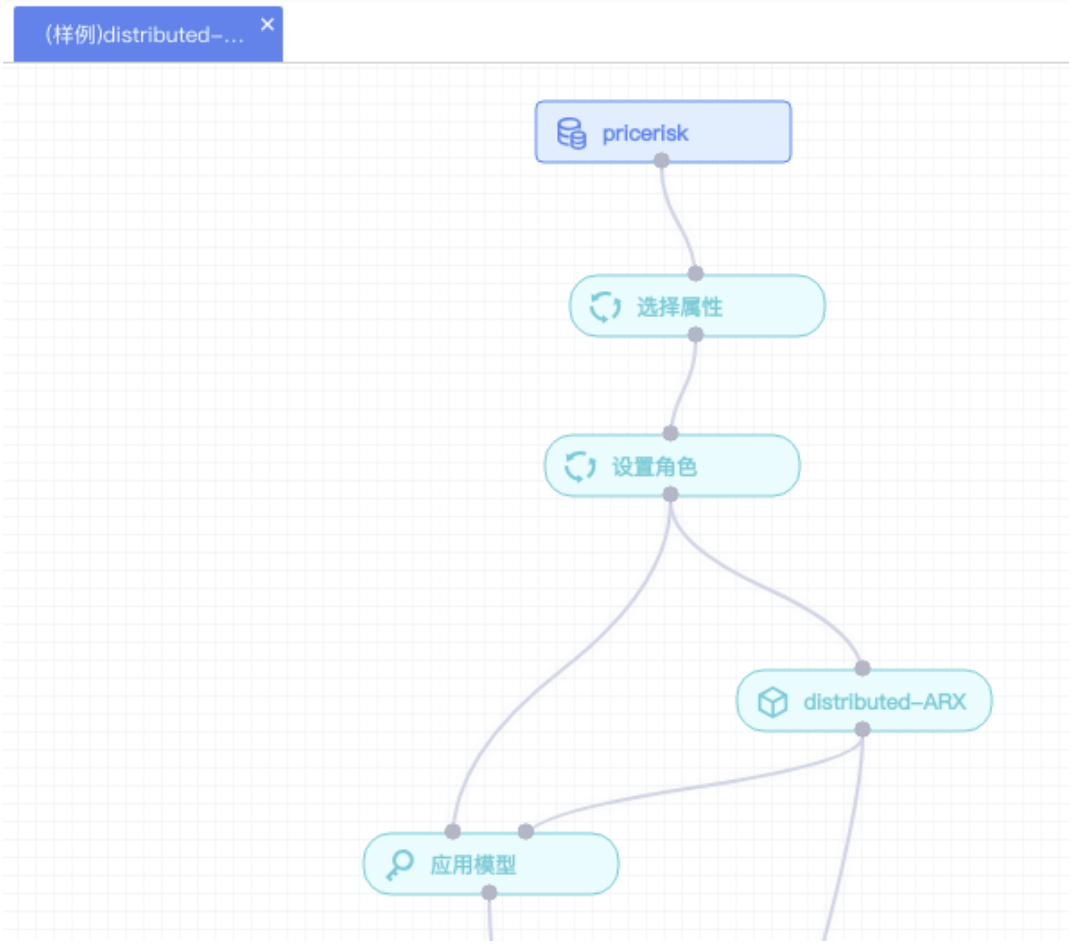
- 添加分布式时间序列分析AR算法【算子名称：distributed-AR】

AR(Auto Regressive Model)自回归模型分布式实现，AR算法是线性时间序列分析模型 中最简单的模型。通过自身前面部分的数据与后面部分的数据之间的相关关系（自相关）来建立回归方程，从而可以进行预测或者分析。应用场景：股票市场涨跌预测及分析、超市的市场营业额预测及分析、房价涨跌预测及分析、员工离职预测及分析、产品的销售量预测及分析、天气情况预测及分析等。



- 添加分布式时间序列分析ARX算法【算子名称：distributed-ARX】

考虑外生变量的自回归模型分布式实现。



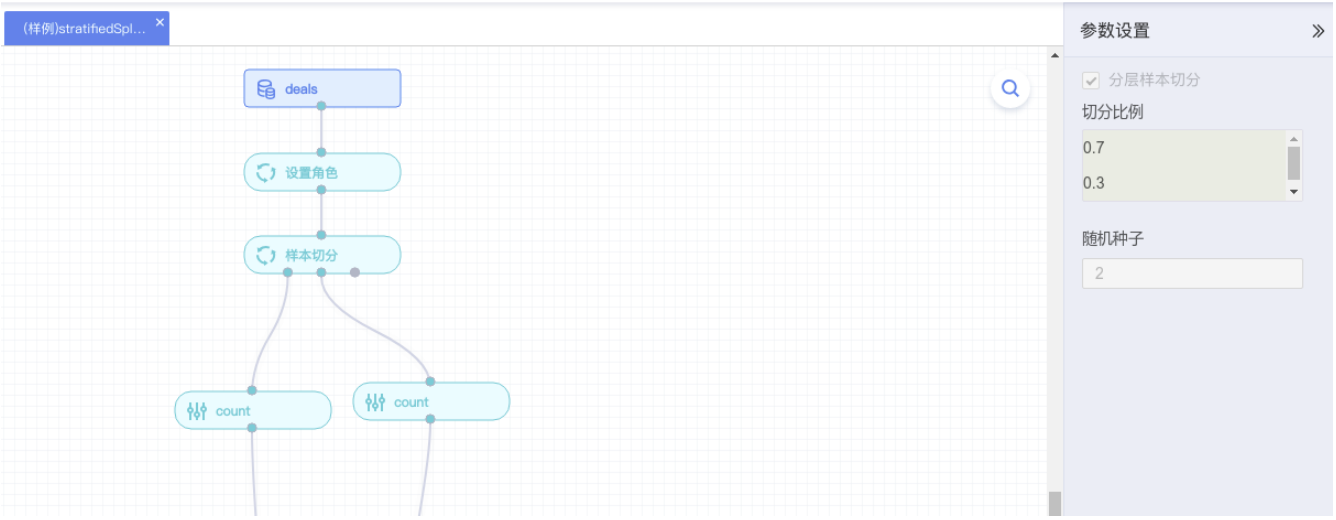
- 优化异常点检测LOF【算子名称：局部异常因子(LOF)】

改进异常点检测LOF的输出结果展现形式，新增结果输出LOF值后，可采用过滤算子过滤异常值。

序号	a1 double feature	a2 double feature	a3 double feature	lof double feature	vector struct<type:tinyint,size:int,indices:ar feature
1	-0.61	3.19	-1.00	5.11	[-0.61,3.19]
2	1.23	2.19	-1.00	3.22	[1.23,2.19]
3	-3.09	-0.83	-1.00	2.90	[-3.09,-0.83]
4	2.22	-2.04	-1.00	2.51	[2.22,-2.04]
5	2.27	0.28	-1.00	2.44	[2.27,0.28]
6	0.67	2.52	-1.00	2.17	[0.67,2.52]
7	2.59	-1.15	-1.00	2.10	[2.59,-1.15]
8	-1.91	-2.24	1.00	2.01	[-1.91,-2.24]
9	-1.58	0.46	-1.00	2.00	[-1.58,0.46]
10	1.05	-2.08	1.00	1.98	[1.05,-2.08]

- 优化样本切分算子【算子名称：样本切分】

增加样本切分中平衡数据和切分比例设置，便于测试集训练集的分层抽样划分。


(样例)stratifiedSpl...
deals
设置角色
样本切分
count
count
参数设置
分层样本切分
切分比例
0.7
0.3
随机种子
2

3. Bug修复

- 隐式狄克雷分布算子：改进隐式狄克雷分布（LDA，即Latent Dirichlet Allocation）算子，可以输出各主题的主题词分布情况
- 实验算子启用/禁用切换修复
- 提取日期元素算子修复
- 分词算子修复
- 二分类lift图修复
- 并集算子修复
- 时间转换算子修复

4. 注意事项

4.1. Sophon 2.0已去除网页端注册功能

如需添加用户，请在Guardian中进行账户分配。

4.2. 安装环境的资源要求

Sophon 2.0 每个安装节点需要的空闲core数量最少为3。

4.3. 网络文件系统安装说明

为了保证网络文件系运行的稳定性，需要专门给网络文件系分配一个磁盘分区，推荐分区大小为50 GB~1 TB。注意，为了保证流畅运行，请尽量把网络文件系卷组划分在数据盘下。如果数据盘较多且存储空间过剩，建议使用单独一块数据盘作为网络文件系分区。否则，建议将一块作为网络文件系分区的数据盘进行格式化处理。

网络文件系的安装有两种实现方式：ceph方式和单机方式，两者的执行时机不同。**Ceph方式需要用户在安装前完成规划，预留磁盘，由旧版本升级的用户同样需要额外提供新的磁盘分区；而单机方式则需在安装完Sophon后才可进行。**若系统无法提供额外分区，请采用单机方式。

由于网络文件系依赖于其他组提供的组件，现阶段组件处于慢慢完善阶段，2.0版本采用ansible工具运维管理ceph，在安装过程中可能会遭遇包依赖和安装冲突问题，安装网络文件系过程我们会参与支持。后期版本待其他组容器化部署方案准备好，采用容器化部署方式，安装过程将大大简化。
为了避免安装网络文件系过程遇到的许多问题，对于现场实施同事注意事项如下：
1. 最好要在一个新装机的环境上进行后续的安装；
2. 保证每个节点网络正常，repo都能正常使用（至少os.repo和transwarp.repo可用）；
3. 安装ceph前，必须先对osd待安装磁盘进行分区或者格式化，查看是否有问题，例如：sgdisk --zap-all --/dev/sde
4. 现场实施同事可以在自己的电脑上放个与centos版本有关的repo库（如centos7-base的repo库），把所有依赖包都放在里面，如果客户缺少某个包，可以在该base repo找到。

4.3.1. Ceph方式

集群中每个要配置安装的节点都准备一块新的空硬盘，有的话可以直接执行下一步，如果没有，可以准备一块使用率较低的硬盘，进行格式化。步骤如下:

- 执行lsblk命令，查看系统硬盘占用情况

```
[root@t7dcb02 ~]# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
vda    253:0    0   35G  0 disk
└─vda1 253:1    0   35G  0 part /
vdb    253:16   0  100G  0 disk /var/lib/docker
vdc    253:32   0  100G  0 disk
└─vdc1 253:33   0   30G  0 part /var/lib/ceph/osd/ceph-2
rbd0   252:0    0   90G  0 disk /var/lib/kubelet/pods/65d5c4c5-b4d8-11e8-a241-fa163eb90032/volumes/kubernetes.io~rbd/pvc-59314392-b4c6-11e8-a241-fa163eb90032
```

- 根据系统硬盘使用情况，准备一块用来安装的无重要数据的硬盘，如vdc，将手动进行格式化，所以请移除重要数据
- 格式化硬盘：
  a. 如果指定的硬盘如/dev/vdc 有分区占用，请卸载分区，否则直接格式化硬盘

b. 卸载vdc占用的分区 ： umount -v /dev/vdc1，如果有进程占用，可能会提示：目标忙。
解决办法：使用 umount -l /dev/vdc1 直接卸载

c. 格式化硬盘：mkfs.xfs -f /dev/vdc

d. 如果执行此命令时仍然提示设备忙，执行 fuser -m -v -i -k /dev/vdc
询问是否杀死/dev/vdc挂载的进程，输入y直接删除占用的进程

e. lsblk查看一下硬盘 vdc的确被格式化

```
[root@t7dcb02 /]# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
vda    253:0    0   35G  0 disk
└─vda1 253:1    0   35G  0 part /
vdb    253:16   0  100G  0 disk /var/lib/docker
vdc    253:32   0  100G  0 disk
```

按照以上方式配置集群中的每个节点。

4.3.2. 单机方式

下载文件： run_pvc.tar.gz
链接: https://pan.baidu.com/s/11jy5ajfTeZKr53qlMqEuww 提取码: b5ti

- 下载上述文件，依次执行下述命令：

```
bash run_pvc.sh #输入当前节点的ip
kubectl create -f nfs-pv.yaml
kubectl create -f nfs-pvc.yaml
```

- 可能需要修改nfs版本，修改默认为nfs 4.0:

```
vim /etc/sysconfig/nfs              // 修改默认为nfs 4.0
```

- 修改RPCNFSDARGS(大约14行) RPCNFSDARGS="-V 4.0":

```
systemctl enable rpcbind
systemctl start rpcbind
systemctl enable nfs
systemctl start nfs
```

4.4. 访问SparkUI

资源池运行过程中,可以访问SparkUI,查看任务的执行情况、存储信息、环境信息、executor的执行情况等。
为访问SparkUI，管理员用户需要配置hosts文件，增加集群的ip: hostname。

```
sudo vim /etc/hosts
```

Windows用户可修改位于C:\Windows\System32\drivers\etc路径下的hosts文件。

4.5. 配置会话超时

可以在服务器端配置会话超时时间，在sophon.conf这个文件中进行配置。
默认是true，打开超时检查，然后会根据livy.server.session.timeout的时间来判断是否超时
false则不检查是否超时

```
livy.server.session.timeout-check = true
```

#Livy在超时空闲会话之前等待的时间（以毫秒为单位）。

```
livy.server.session.timeout = 1h
```

4.6. 打开Notebook报错

若报错内容为Insufficient cpu ，请联系管理员调小CPU个数，最小可为0。

若报错内容为SchedulerPredicates failed due to Failed to get PersistentVolumeClaim "nfs-test", which is unexpected，请在环境中使用kubectl get pvc命令，查看pvc的名字和是否Bound。单机版网络文件系统请使用systemctl status nfs；Ceph则使用ceph -s查看ceph的状态，kubectl get po | grep nfs-provisioner查看nfs-providor，出错重启即可。

4.7. 协作项目中的资源池使用问题

如果项目拥有者在项目中使用的资源池，协作者不在资源池内，点击协作项目会跳出到项目列表页。若协作用户需要协作该项目，可以让管理员将其加入该项目所使用的资源池，或者拥有者改变项目所使用的资源池，即可解决问题。

5. 已知问题

5.1. 兼容问题

Sophon2.0模型保存与1.X版本不兼容，无法读取旧版模型的ZIP文件。

若升级到2.0后，执行在1.3中创建和保存的实验（含有模型），运行失败，解决方法是找到形成模型的实验，执行实验并重新保存模型，然后应用模型，便可成功执行。

5.2. api服务所蕴含的模型被修改或删除会引发api服务不可用

若创建的API服务包含模型m1和m2，初次上线该API服务可以成功响应请求；但如果删除m1或m2，再次上线服务则会出现IndexOutOfBoundsException，因为该API服务所包含的某个模型已被删除了，从而导致API服务不可用。

5.3. Inceptor数据集暂不支持orc格式的数据库表

如需打开orc格式的数据表，请选择创建数据库数据集。

5.4. 实验导出格式问题

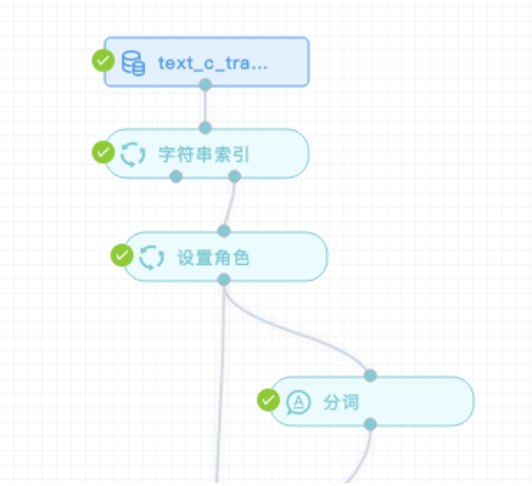
Sophon 1.3实验导出的Zip格式项目，需要解压修改为Json格式，才可以导入2.0的实验中。

5.5. 写入数据源算子问题

写入数据源算子选择jdbc方式时，连接报错。如果要写入inceptor和hive，可以使用写入inceptor算子，写入其他数据库，可使用写入数据库算子实现。
写入数据源算子保存到数据集之后，点击设置页面出错，可在实验中设置数据集。

5.6. 分词算子问题

分词算子会重置已设置好的label
例如，在如下的实验流程中：



在设置label列为label后，数据结果为：

result 1 result 2

表格 图表

序号	content string feature	label double label
1	欧盟委员会负责信息社会和媒体事务的委员维维亚娜·雷丁...	0.00
2	6月3日消息，中国三大电信运营商香港上市股票周二复牌...	0.00
3	微软正在欢送52岁的比尔·盖茨，他将在今天正式退休，...	0.00
4	城市无线局域网的概念是指把整个城市纳入一个无线地区，...	0.00
5	据知情人士透露，诺基亚将以1．094亿美元收购西门子...	0.00
6	微软通常会受到开发者群体的欢迎。但这一趋势已经不复存...	0.00
7	自从iPhone推出以来，苹果公司已经收到了一系列的...	0.00

而执行了分词算子之后，label列被重置回feature：

result 1 result 2

表格 图表

序号	content string feature	label double label
1	欧盟委员会负责信息社会和媒体事务的委员维维亚娜·雷丁...	0.00
2	6月3日消息，中国三大电信运营商香港上市股票周二复牌...	0.00
3	微软正在欢送52岁的比尔·盖茨，他将在今天正式退休，...	0.00
4	城市无线局域网的概念是指把整个城市纳入一个无线地区，...	0.00
5	据知情人士透露，诺基亚将以1．094亿美元收购西门子...	0.00
6	微软通常会受到开发者群体的欢迎。但这一趋势已经不复存...	0.00
7	自从iPhone推出以来，苹果公司已经收到了一系列的...	0.00

为了避免该问题，必须在分词算子后添加一个设置角色算子。

6. 附件

Notebook 2.0版本支持的包

6.1. Python支持的包

bleach 1.5.0	certifi	chardet 2.3.0	decorator 4.2.1
entrypoints 0.2.3	html5lib 0.9999999	idna 2.6	ipykernel 4.8.2
ipython 6.4.0	ipython-genutils 0.2.0	ipywidgets 7.2.1	jedi 0.12.0
Jinja2 2.8	jsonschema 2.6.0	jupyter	jupyter-client
jupyter-console	jupyter-core	jupyterlab	MarkupSafe 0.23
mistune 0.8.3	nbconvert 5.3.1	nbformat 4.4.0	numpy 1.14.1
pandas 0.22.0	pandocfilters 1.4.2	parso 0.2.0	pexpect 4.0.1
nose 1.3.7	mock 2.0.0	pickleshare 0.7.4	matplotlib 2.2.2
prompt-toolkit 1.0.15	ptyprocess 0.5	Pygments 2.2.0	python-dateutil 2.6.1
pytz	pyzmq 17.0.0	qtconsole 4.3.1	requests 2.9.1
Send2Trash 1.5.0	simplegeneric 0.8.1	six 1.10.0	terminado 0.8.1
testpath 0.3.1	tornado 5.0	traitlets 4.3.2	urllib3 1.13.1
wcwidth 0.1.7	webencodings 0.5.1	widgetsnbextension 3.2.1	requests_kerberos 0.8.0
tensorpack 0.8.5	onnx 1.1.2	lmdb 0.94	Pillow
scikit-learn 0.19.1	Cython	contextlib2	lxml
setuptools	scipy	keras	h5py

6.2. R支持的包

SparkR 2.3.0	arules 1.6-1	arulesViz 1.3-1	assertthat 0.2.0
atlas 1.0.0	backports 1.1.2	base 3.4.4	base64enc 0.1-3
BH 1.66.0-1	bibtex 0.4.2	bindr 0.1.1	bindrcpp 0.2.2
bit 1.1-14	bit64 0.9-7	bitops 1.0-6	blob 1.1.1
boot 1.3-20	brew 1.0-6	callr 2.0.4	caTools 1.17.1.1
chron 2.3-52	class 7.3-14	cli 1.0.0	cluster 2.0.6
codetools 0.2-15	colorspace 1.3-2	commonmark 1.5	compiler 3.4.4
corrplot 0.84	crayon 1.3.4	crosstalk 1.0.0	curl 3.2
data.table 1.11.4	datasets 3.4.4	DBI 1.0.0	dendextend 1.8.0
DEoptimR 1.0-8	desc 1.2.0	devtools 1.13.6	dichromat 2.0-0

digest 0.6.15	diptest 0.75-7	doParallel 1.0.11	dplyr 0.7.6
DT 0.4	e1071 1.7-0	evaluate 0.11	fansi 0.3.0
flexclust 1.3-5	flexmix 2.3-14	FNN 1.1.2.1	foreach 1.4.4
forecast 8.4	foreign 0.8-69	formatR 1.5	fpc 2.1-11.1
fracdiff 1.4-2	gclus 1.3.1	gdata 2.18.0	ggplot2 3.0.0
git2r 0.23.0	glmnet 2.0-16	glue 1.3.0	gplots 3.0.1
graphics 3.4.4	grDevices 3.4.4	grid 3.4.4	gridBase 0.4-7
gridExtra 2.3	gsubfn 0.7	gtable 0.2.0	gtools 3.8.1
hexbin 1.27.2	highr 0.7	hms 0.4.2	htmltools 0.3.6
htmlwidgets 1.2	httpuv 1.4.5	httr 1.3.1	igraph 1.2.2
IRdisplay 0.5.0	IRkernel 0.8.12.9000	irlba 2.3.2	iterators 1.0.10
jsonlite 1.5	kernlab 0.9-27	KernSmooth 2.23-15	kknn 1.3.1
knitr 1.20	ks 1.11.3	labeling 0.3	lambda.r 1.2.3
later 0.7.3	lattice 0.20-35	lazyeval 0.2.1	LiblineaR 2.10-8
lmtest 0.9-36	lpSolveAPI 5.5.2.0-17	lubridate 1.7.4	magrittr 1.5
manipulateWidget 0.10.0	markdown 0.8	MASS 7.3-50	Matrix 1.2-14
mclust 5.4.1	memoise 1.1.0	methods 3.4.4	mgcv 1.8-23
mime 0.5	miniUI 0.1.1.1	misc3d 0.8-4	modeltools 0.2-22
moments 0.14	multicool 0.1-10	munsell 0.5.	mvtnorm 1.0-8
nlme 3.1-131.1	nloptr 1.0.4	NMF 0.21.0	nnet 7.3-12
normtest 1.1	openssl 1.0.2	parallel 3.4.4	pbdZMQ 0.3-3
pillar 1.3.0	pkgbuild 1.0.0	pkgconfig 2.0.2	pkgload 1.0.0
pkgmaker 0.27	plogr 0.2.0	plotly 4.8.0	pls 2.6-0
plyr 1.8.4	pmmlTransformations 1.3.2	prabclus 2.2-6	praise 1.0.0
prettyunits 1.0.2	pROC 1.12.1	processx 3.2.0	promises 1.0.1
proto 1.0.0	ps 1.1.0	purrr 0.2.5	qap 0.1-1
quadprog 1.5-5	quantmod 0.4-13	R6 2.2.2	RColorBrewer 1.1-2
Rcpp 0.12.18	RcppArmadillo 0.9.100.5.0	RCurl 1.95-4.11	readr 1.1.1
registry 0.5	repr 0.15.0	reshape2 1.4.3	rJava 0.9-10
rlang 0.2.2	rmarkdown 1.10	rngtools 1.3.1	robustbase 0.93-2
rpart 4.1-13	rprojroot 1.3-2	Rserve 1.7-3	RSQLite 2.1.1
rstudioapi 0.7	scales 1.0.0	scatterplot3d 0.3-41	seriation 1.2-3
shiny 1.1.0	sourcetools 0.1.7	sp 1.3-1	spatial 7.3-11

splines 3.4.4	sqldf 0.4-11	statmod 1.4.30	stats 3.4.4
stats4 3.4.4	stringi 1.2.4	stringr 1.3.1	survival 2.42-6
tcltk 3.4.4	testthat 2.0.0	tibble 1.4.2	tidyr 0.8.1
tidyselect 0.2.4	timeDate 3043.102	tinytex 0.6	tools 3.4.4
tree 1.0-39	trimcluster 0.1-2.1	tseries 0.10-45	TSP 1.1-6
TTR 0.23-3	urca 1.3-0	uroot 2.0-9	utf8 1.1.4
utils 3.4.4	uuid 0.1-2	vcd 1.4-4	viridis 0.5.1
viridisLite 0.3.0	visNetwork 2.0.4	webshot 0.5.0	whisker 0.3-2
withr 2.1.2	xfun 0.3	xgboost 0.71.2	xtable 1.8-2
xts 0.11-0	xxIRT 2.1.0	yaml 2.2.0	zoo 1.8-3