

目录

目录	1
注册与开通服务	2
步骤1: 注册金山云账号	2
步骤2: 开通 KS3 使用权限	2
步骤3: 开通 KingAI 使用权限	2
步骤4: 创建AK、SK	2
步骤5: 添加AK/SK	2
步骤6: 购买计算节点	2
步骤7: 分配资源	2
步骤8: 工作空间	2
创建AI容器开发环境	2
步骤1: 填写基本信息	2
步骤2: 节点定义	3
步骤3: 高级配置	3
步骤4: 创建完成, 查看登录信息	4
使用可视化建模	4
步骤1: 新建可视化建模 workflow	5
步骤2: workflow中读取数据示例	5
步骤3: 数据预处理及特征工程	5
步骤4: 模型训练及模型评估	6
步骤5: 模型注册及推理服务发布	7

注册与开通服务

为了方便您更有效地使用金山云KingAI人工智能平台，请先完成以下步骤：

步骤1：注册金山云账号

使用金山云服务前，需要您先拥有金山云注册账号，并完成实名认证。
若您已在金山云注册并完成实名认证，请忽略此步骤。

[点此注册金山云账号](#)

步骤2：开通 KS3 使用权限

您在使用KingAI人工智能平台之前，需要先开通金山云对象存储KS3使用权限。详情参见 [KS3开通指南](#)，如已完成 KS3 权限开通，可忽略此步骤。

步骤3：开通 KingAI 使用权限

进入 [KingAI人工智能平台](#) 点击联系客服，或 [创建工单](#) 开通 KingAI 人工智能平台权限。

[创建工单](#)

步骤4：创建AK、SK

KingAI 需要通过主账号AK/SK，在KS3中创建Bucket作为KingAI数据持久化位置。
获取 AK、SK 有两个办法：

1. 账号注册完成后，系统将自动产生的Access Key 文件，下载该文件后，即可查看AK、SK；
2. 若注册完成后，AK、SK无法找回，您可重新创建AK、SK，在控制台右上角点击 [Access Key](#) 即可进行Access Key管理。

步骤5：添加AK/SK

主账号初次进入KingAI平台，需要添加密钥。

您需要进入[资源管理](#)>[资源总览](#)页面，在密钥管理中单击[添加访问密钥](#)，输入步骤4申请过的AK/SK。

步骤6：购买计算节点

主账号初次登陆KingAI平台，资源为空，需购买计算节点，才可创建训练任务。

您需要进入[资源管理](#)>[集群管理](#)页面，单击[增加节点](#)，根据不同的需求购买不同的计算节点资源。

步骤7：分配资源

主账号购买完资源后，需要对要创建任务的用户设置资源，才可以进行KingAI平台的使用。具体操作包含：1. 设置算力型号；2. 算力分配。

操作步骤

1. 登录[人工智能KingAI控制台](#)。
2. **设置算力型号**：单击左侧导航栏[用户管理](#)，进入用户管理页面，选择要设置的用户（包含自己的主账号或子账号），单击该用户操作列中的[设置算力型号](#)，设置GPU型号使用上限、CPU使用上限和内存使用上限，设置的上限需要满足用户创建任务的资源需求。
3. **算力分配**：在用户管理页面，选择要设置的用户（包含自己的主账号或子账号），单击该用户操作列中的[算力分配](#)，给用户分配算力时，需要满足用户创建任务的资源需求。

建议：如果想避免主账号重复充值操作，可将使用者（子账号）的算力型号和算力时分配为一个较大的值。

步骤8：工作空间

用户初次登陆KingAI平台，会默认生成一个工作空间，允许工作空间的所有者在KingAI顶部栏中对自己的工作空间进行设置，如增删工作成员，同一工作空间的成员所有的数据、算法、模型共享。

创建AI容器开发环境

针对AI开发者，KingAI提供了即开即用的AI开发环境，支持用户弹性申请算力资源，编写、调试AI算法代码。

- KingAI 集成了基于开源的 JupyterLab，可为您提供在线的交互式开发调试工具，关于JupyterLab的详细操作指导，请参见[JupyterLab官方文档](#)。
- KingAI 支持提供SSH远程登录能力，配合VScode或者其他IDE进行远程代码开发，请参见 [Visual Studio Code Remote - SSH](#)。
- KingAI 容器开发环境预置了不同的AI框架，如TensorFlow、PyTorch、Mxnet等，方便您快速构建算法运行环境，详见 [KingAI预置镜像](#)。
- KingAI 容器开发支持用户自定义AI开发环境，您可以使用[自定义镜像](#)来实现。

操作步骤

步骤1：填写基本信息

填写AI容器的基本信息，各字段的解释及要求如下。

- **容器名称**：必填项，必须以小写字母开始，支持大小写字母、数字及下划线，长度为2-30字符。
- **备注**：选填项，最多可输入200个字符。
- **节点连接密钥**：必选项，KingAI容器开发，支持用户SSH远程登录，用户需要选定获取节点连接密钥，才能进行SSH远程登录。支持选择已有密钥及新建密钥两种方式进行远程SSH登录。
- **分配sudo权限**：必选项，包含是和否两个选项，当选择是时，可以对容器拥有sudo权限。

注意：如果通过sudo对系统目录进行了修改，容器重启后将无法持久化。

代码和数据请放在Home (/home/kai/) 目录下，否则容器重启后数据将会丢失!

基本信息

* 容器名称:

(选填) 备注:

* 节点连接密钥:

分配sudo权限: 否 是

如果通过sudo对系统目录进行了修改，容器重启后将无法持久化

步骤2：节点定义

填写AI容器名称后，即可进行镜像的选择及节点的定义。

- 镜像选择：**必选项，KingAI平台内含预置镜像，同时支持用户在全局管理>镜像管理中自定义上传镜像。预置镜像包含CPU、GPU两种类型，同时支持pytorch和tensorflow等多框架，您可根据自己需求选择相应的镜像。
注意：不同的镜像中包含了不同的深度学习框架，以及相应的Python软件依赖，注意如果没有申请GPU资源而使用 GPU镜像，可能会导致import 框架包时失败。
- 是否支持分布式：**必选项，容器启动支持单节点和分布式多节点两种方式。当选择支持分布式时，可增加节点数量，按节点维度配置资源类型。
- 节点资源配置：**表单中填写的计算资源均为使用上限，CPU和GPU资源如果在使用过程中达到上限会导致算法运行变慢 卡顿，内存超过使用上限可能会发生错误导致失败。另外资源的占用为排他性占用，特别是 GPU资源，因此最好合理根据算法的需要申请资源，避免申请过少影响性能，申请太多浪费资源。
- 文件挂载：**选填项，包含数据集和模型，两块内容均挂载在/home目录下，且均为只读权限。
- 挂载本地磁盘：**必填项，包含是与否两个选项，当选择是时，KingAI额外提供用户一个数据盘，目录为/home/cache，在该路径下读取数据可以优化IO，但需要注意的是该数据盘不做持久化存储，容器任务停止、删除后都会清空该目录下的全部数据。

* 镜像选择：

是否支持分布式： 否 是

节点实例数量：

资源选择：

当前资源配置信息：CPU：1核 内存：2GB

资源类型	型号	数量
CPU	-	<input type="text" value="1"/>
内存	-	<input type="text" value="2"/>
GPU:	<input type="text" value="不使用GPU"/>	<input type="text" value="0"/>

文件挂载：

挂载支持数据集及模型，挂载后将显示在/home目录下，仅有只读权限

挂载类型	挂载详情	挂载版本	操作
/(T o T)/~~ 没有找到亲爱的数据集~			

+ 添加

挂载本地磁盘： 否 是 (挂载路径/home/cache)
在该路径下读取数据可以优化IO。注意：该目录下的数据在容器停止后不会持久化

+ 增加一组实例

步骤3：高级配置

KingAI 容器支持用户自定义初始化脚本、Python依赖、开启5050端口等高级配置项，详情如下：

- 初始化脚本：**选填项，提供下载模板，可以按模板格式填写一个可执行的shell脚本。同时支持用户将常用的初始化脚本保存为模板，在下次创建容器任务的时候复用保存过的脚本。您可以在容器开发>开发管理>初始化脚本模板中查看保存过的所有初始化脚本。
- Python依赖：**选填项，提供下载模板，可以按模板格式填写python第三方依赖库。支持不同的第三方包以逗号或者回车间隔。
注意：由于安装包有时候需要花费一定时间，为避免创建完容器任务等待时间过长，安装任务在后台运行。所以您进入容器后，当搜索不到输入的第三方包时，可能是正在安装过程中。
- 开启5050端口：**必选项，包含是与否两种选项，端口开启后，可从外网访问容器中监听5050端口的服务。
- 共享内存比例：**容器可用共享内存（shared memory）=申请内存*共享比例，默认为0.5，一般情况不推荐修改默认值，可能会影响节点异常，异常情况下将继续计费。
- limits参数设置：**limits参数主要用来限制用户对系统资源的使用，详情见limits参数说明。

备注：共享内存比例权限和limits参数设置权限为高级配置项，子账号默认没有这个权限，主账号需要在用户管理>用户列表中对子账号选择操作列中的设置权限，开启两个权限项。

高级设置

初始化脚本: 选择已有初始化脚本

一个可执行的shell脚本

↓ 下载模板
↓ 另存为模板

python依赖: python第三方包名称, 建议以逗号或者回车间隔, 如果需要指定版本, 必须写作 package_name==package_version

↓ 下载模板

开启5050端口: 否 是
端口开启后, 可从外网访问容器中监听5050端口的服务。

共享内存比例: 0.5

limits参数设置: 请选择

参数名称	参数值
/(T o T)/~~ 没有找到亲爱的数据嘛~	

limits参数主要用来限制用户对系统资源的使用, 详见limits参数说明, 非必要情况下不建议修改配置, 可能会引起节点异常, 异常情况下将继续计费。

步骤4: 创建完成, 查看登录信息

AI容器创建完成后, 将自动返回到AI容器列表, 点击新创建的AI容器名称, 即可查看登录地址。登录地址显示notebook登录地址及SSH登录地址。同时显示容器的基本信息、节点信息、高级配置信息和启动历史信息等。

KingAI

- 信息中心
- 实验管理
- 数据管理
- 模型开发
- 容器开发
- 可视化建模
- 自动学习
- 模型训练
- 训练任务
- 算法管理
- 模型推理
- 模型管理
- 推理服务
- 推理 Pipeline
- 全局管理
- 资源管理
- 用户管理
- 活动日志
- 个人中心
- 镜像管理
- AI 商店

容器开发 > 详情

详情
启动日志
事件
监控

基本信息 (z_notebook)

环境ID: 64bf35a0-b4b0-4d50-b6d3-c16cce33f239	创建人: kdl_testuser01	状态: 运行中
创建时间: 2021-03-25 21:24:07	更新时间: 2021-03-26 11:59:04	备注信息:
sudo权限: 关闭		

链接信息

jupyter: [查看](#)

SSH 登录: 密钥: zeng
 远程访问: `ssh -i zeng.pem -p 2222 wyvonwgwwh@120.92.9.194` [复制](#)
 远程拷贝: `scp -i zeng.pem -P 2222 <您的文件> wyvonwgwwh@120.92.9.194:/tmp` [复制](#)

节点信息

镜像: notebook-v2.1.torch1.6.0-torchvision0.7.0-py3.6-cpu

节点实例数量: 1

节点资源: CPU: 2核 内存: 4GB

挂载文件:	挂载类型	挂载详情	挂载版本
/(T o T)/~~ 没有找到亲爱的数据嘛~			

挂载本地磁盘: 否 (挂载路径/home/cache)

高级配置信息

使用可视化建模

90KingAI人工智能平台通过可视化的拖拽布局, 组合各种数据源、预处理、特征工程、算法等模块, 为数据科学家打造从数据预处理、模型训练、评估、注册的全流程开发支持。更多详见[可视化建模简介](#)。

- 为了方便您更有效的使用KingAI进行可视化建模, 本文档将通过一个“某地信用卡客户贷款违约”案例向您演示使用流程。
- 在使用可视化建模前, 请确保您已经完成[注册与开通服务](#)。
- 在使用可视化建模前, 请确保您已完成算力分配。

操作步骤

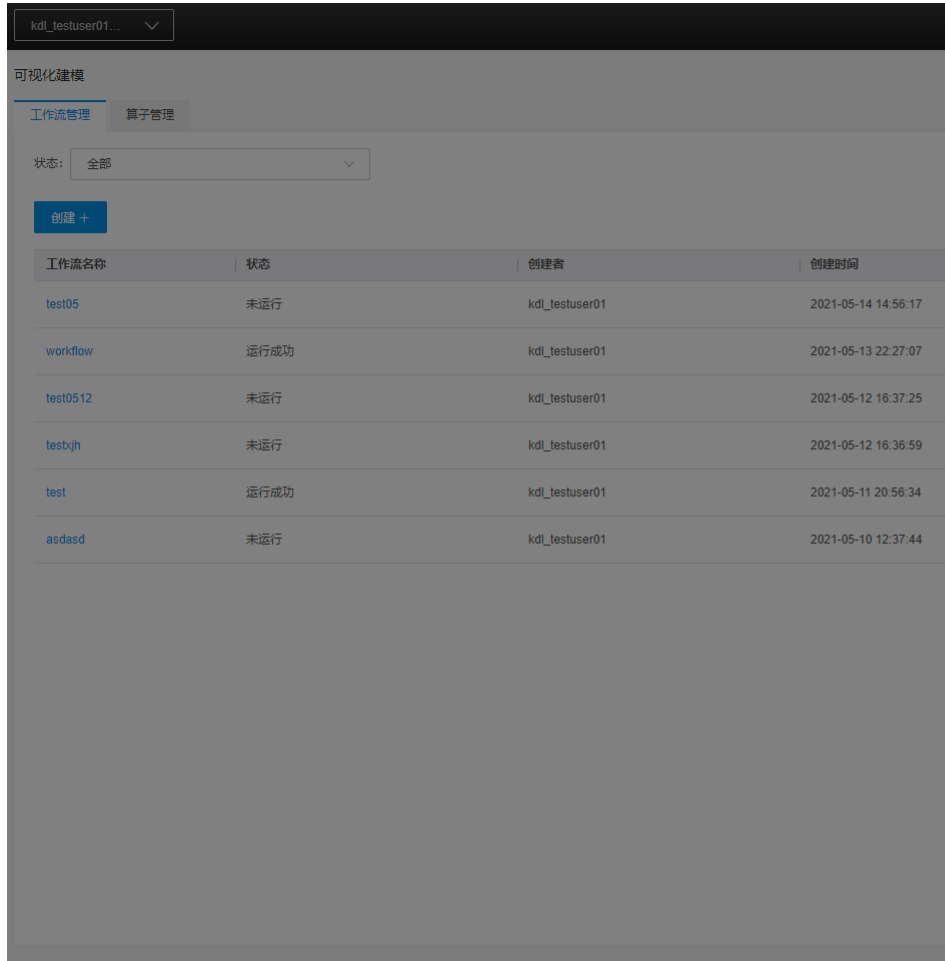
可视化建模主体流程如下：

- 登录KingAI进入控制台首页；
- KingAI人工智能学习工作室，点击【立即开始】按钮；
- 创建工作流（数据导入、数据预处理、特征工程、分类模型、模型预测、模型评估、模型选优、模型注册）；
- 运行工作流，得到最优模型及结果，进行模型注册

步骤1： 新建可视化建模工作流

使用KingAI可视化建模，首先需要完成[新建可视化建模](#)。

- 点击工作流，然后点击新建工作流；
- 填写相关信息，工作流名需要合法，且不与已存在的工作流重名，选择对应选项，最后点击确认；



新建工作流



步骤2： 工作流中读取数据示例

工作流中多种导入数据方式，此处只展示从工作流自带开源数据集、数据管理、KS3三种，本次演示我们将使用样本数据进行示例。

- 从左侧算子区展开-数据IO-拖动样本数据到空白处
- 右键点击样本数据，点编辑，选择某地信用卡借贷数据

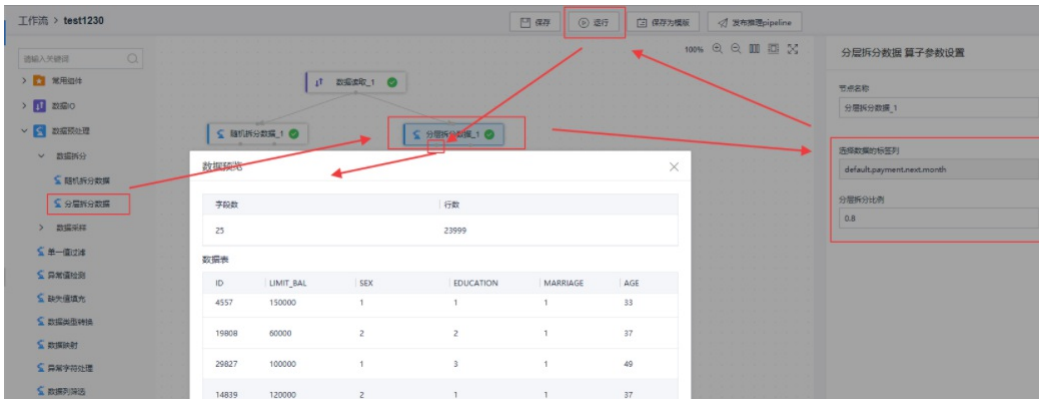
步骤3： 数据预处理及特征工程

数据预处理包括对数据进行数据列筛选、分层拆分、分层采样、主成分分析等，详细如下：



1. 数据列筛选 1) 除id外其他列全选 (随机拆分类似) 1) 选default.payment.next.month为标签列，拆分比例0.8

2. 分层拆分



3. 分层采样 1) 选default.payment.next.month为

标签列, 采样比例0.7 2) 除分层采样外, 工作流还支持随机采样等四种采样算子, 使用方式类似



4. 主成分分析 1) 降维的特征列全选 2) 附加的数



据列选default.payment.next.month

步骤4: 模型训练及模型评估

数据预处理及特征工程完成后, 即可进行算法训练, 训练完成后可对模型进行评估, 查看模型指标。

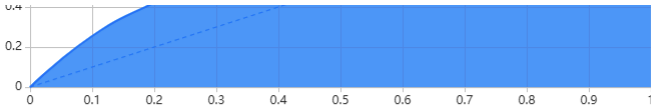
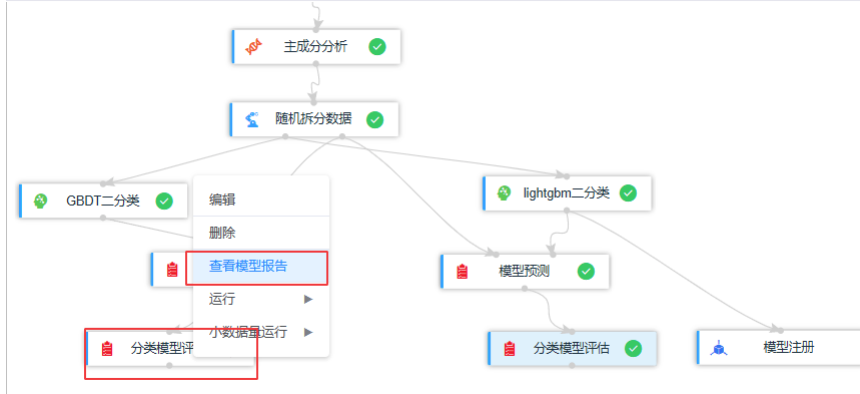
1. 模型训练 添加GBDT二分类模型, 特征列表除default.payment.next.month外全选, 模型标签选择default.payment.next.month。



2. 模型评估



选择分类模型评估算子，即可进行分类模型评估。



混淆矩阵:

真实/预测	预测为正(+)	预测为负(-)	汇总
实际为正(+)	705	1243	1948
实际为负(-)	1156	5896	7052
汇总	1861	7139	9000

准确率: $(705 + 5896) / 9000 = 0.7334$

召回率: $705 / (705 + 1243) = 0.3619$

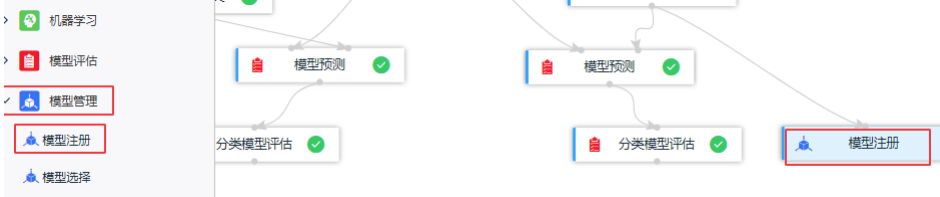
F1值: 0.3702

确定

步骤5: 模型注册及推理服务发布

模型训练评估完成后，可将意向模型进行注册，注册后模型将自动保存至模型管理中。模型保存后，支持将模型发布为推理服务生成restful接口进行调用。

1. 模型注册 拖动模型注册算子到画布中，与模型算法相连，进行模型注册



2. 模型推理服务发布 在模型管理中，找到刚注册完成的模型，点击推理服务发布，可将模型发布为推理服务，服务发布完成，即可进行调用。

模型管理服务

模型名称: test_gpt_001

模型名称	模型类型	数量	创建时间	更新时间	备注	操作
test_gpt_001	...	2	2020-12-31 11:07:08	2020-12-31 11:07:08		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	7	2020-12-31 10:49:15	2020-12-31 10:49:15		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	6	2020-12-31 21:03:01	2020-12-31 21:03:01		删除 模型下架 更多
test1230-评分卡生成_1	...	5	2020-12-30 20:10:46	2020-12-30 20:10:46		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	4	2020-12-30 20:07:02	2020-12-30 20:07:02		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	3	2020-12-30 20:04:09	2020-12-30 20:04:09		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	2	2020-12-30 20:03:09	2020-12-30 20:03:09		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	1	2020-12-30 20:02:15	2020-12-30 20:02:15		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	1	2020-12-30 20:00:26	2020-12-30 20:00:26		删除 模型下架 发布为服务
test1230-评分卡生成_1	...	1	2020-12-30 19:30:24	2020-12-30 19:30:24		删除 模型下架 发布为服务

模型名称: test_gpt_001

模型类型: CPU

数量: 2

确认

模型管理服务 > API测试

基本信息

名称: test_gpt_001

描述:

请求方式: POST

返回类型: JSON

API调用地址: <http://ai.console.koyun.com/5642937/ai/inference/online-service-8f1uc2bcrot/predict>

```
模型名称: test1230-评分卡生成_1
```

```
1 1
```

```
2 1
```

```
3 1
```

```
4 20000.0
```

```
5 2
```

```
6 2
```

```
7 1
```

```
8 24
```

```
9 2
```

```
10 2
```

```
11 -1
```

```
12 -1
```

```
13 -2
```

```
14 -2
```

```
15 2023.0
```

```
16 2023.0
```

```
1 result: {
```

```
2 {
```

```
3 "tag": 24,
```

```
4 "tag_score": 0,
```

```
5 "BILL_AMT1": 2913,
```

```
6 "BILL_AMT1_score": 0,
```

```
7 "BILL_AMT2": 3182,
```

```
8 "BILL_AMT2_score": 0,
```

```
9 "BILL_AMT3": 689,
```

```
10 "BILL_AMT3_score": 0,
```

```
11 "BILL_AMT4": 0,
```

```
12 "BILL_AMT4_score": 0,
```

```
13 "BILL_AMT5": 0,
```

```
14 "BILL_AMT5_score": 0,
```

```
15 "BILL_AMT6": 0,
```

```
16 "BILL_AMT6_score": 0,
```

```
17 "BILL_AMT7_score": 0,
```

```
18 "TOTALSCORE": 2,
```