浙江大学
ZHEJIANG UNIVERSITY

基于R语言的分布式机器学习平台设计与实现

顾全　Z15030801　软件工程

导师　汤斯亮　副教授

CONTENTS

目录

1. 绪论
2. 关键技术与框架
3. 机器学习算法分析
4. 需求分析与架构设计
5. 平台实现及成果展示
6. 总结与展望

浙江大学
ZHEJIANG UNIVERSITY

CONTENTS

1 绪论

2 关键技术与框架

3 机器学习算法分析

4 需求分析与架构设计

5 平台实现及成果展示

6 总结与展望

绪论

- 许多大公司都开发了自己的机器学习云平台，其目的是为了加速整个研发过程、降低人工成本。
- 国外：微软的AMLS、亚马逊的Amazon Machine Learning
  国内：阿里PAI、百度BML

微软Azure Machine Learning Studio

Amazon Machine Learning

阿里云机器学习 PAI

百度机器学习 BML

大型机器学习平台介绍

Microsoft Azure Machine Learning | Home Studio Gallery

Search experiment items

- Saved Datasets
- Data Format Conversions
- Data Input and Output
- Data Transformation
- Feature Selection
- Machine Learning
- OpenCV Library Modules
- Python Language Modules
- R Language Modules
- Statistical Functions
- Text Analytics
- Web Service
- Deprecated

Binary Classification: Direct marketing

In draft

Reader

Metadata Editor

Project Columns
remove columns that are part of the label

Split

Two-Class Boosted Decision T...

Two-Class Support Vector Ma...

Split

Split

Sweep Parameters

Sweep Parameters

Score Model

Score Model

Evaluate Model

Properties

Two-Class Boosted Decision Tree

Create trainer mode: Single Parameter

Maximum number of leav...: 20

Minimum number of sam...: 10

Learning rate: 0.2

Number of trees construct...: 100

Random number seed: 0

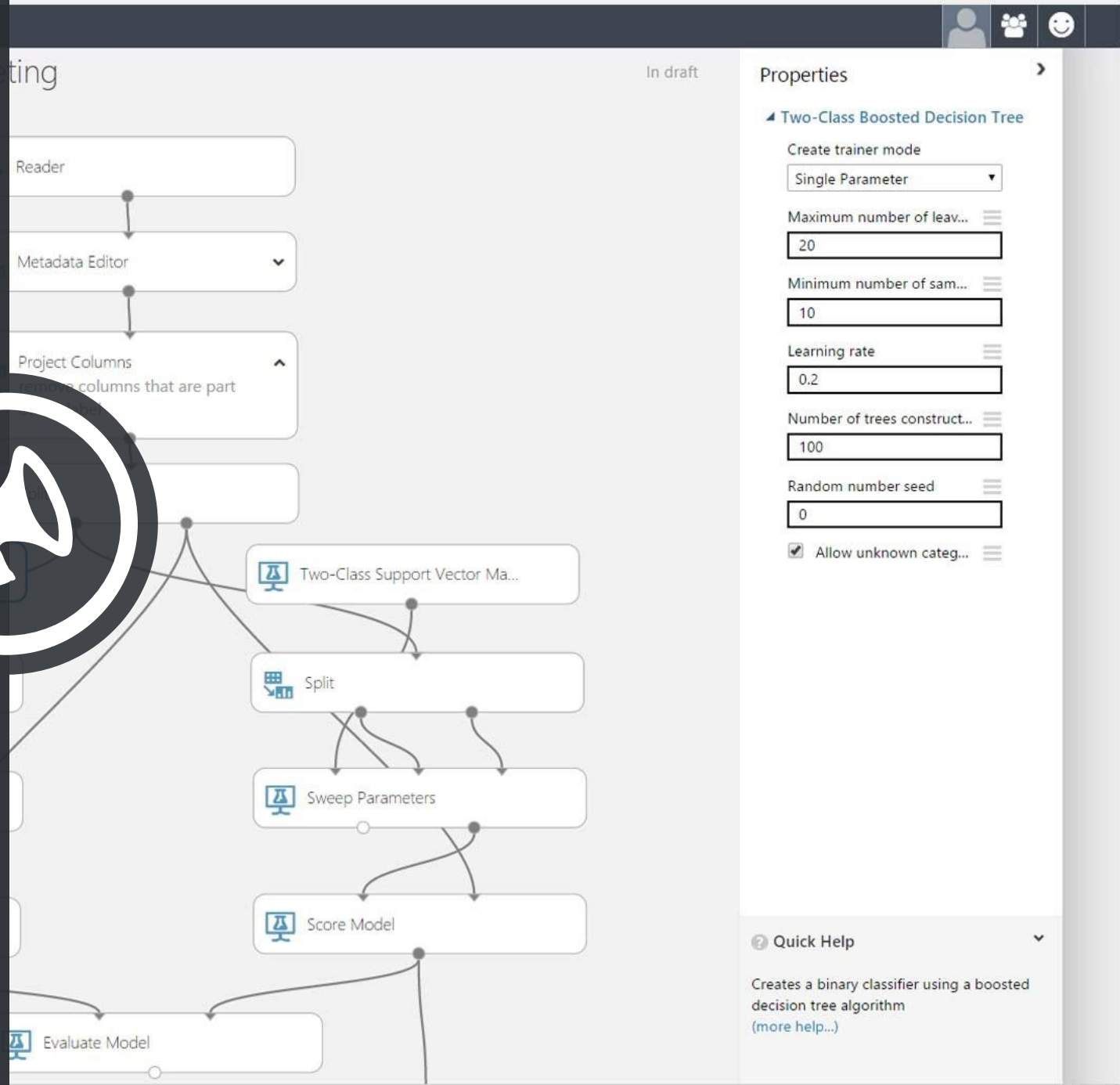
Allow unknown categ...

Quick Help

Creates a binary classifier using a boosted decision tree algorithm
(more help...)

Azure Machine Learning Studio

- 使用拖放手势和简单的数据流图来设置实验
- 支持R和Python包以及像Xbox和Bing等微软业务中的一流算法
- 支持R和Python的自定义代码
- 还支持写 SQL



ML model settings

You can use the automatically suggested ML model settings, or you can choose to customize.

ML model type BINARY

ML model target y

ML model name (Optional) ML model: Banking.csv

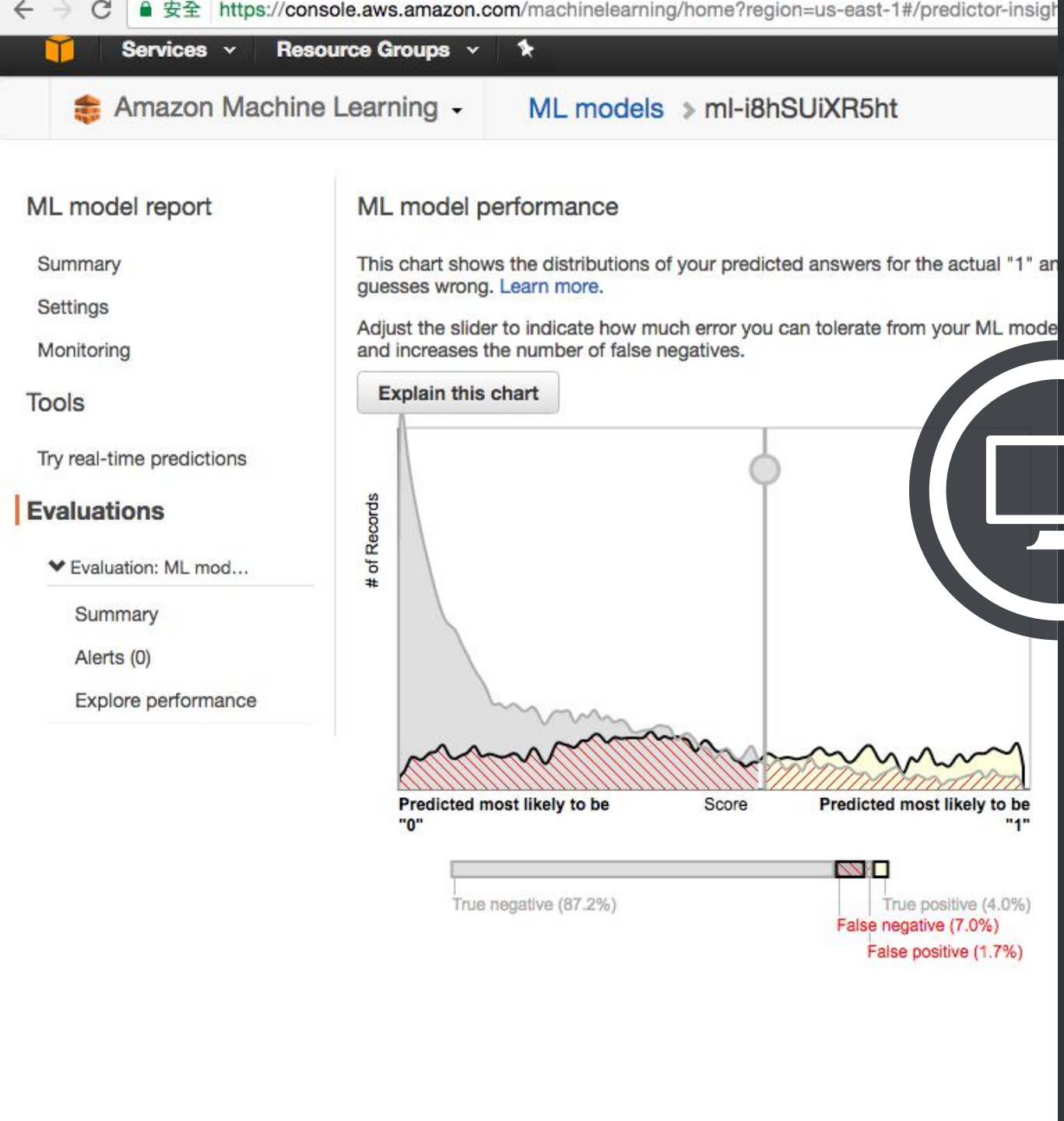
Select training and evaluation settings Recipes and training parameters control the ML model training process. You can select these settings for your ML model or use the defaults provided by Amazon ML. In either case, you can choose to have Amazon ML reserve a portion of the input data for evaluation. Learn more.

Default (Recommended)
- Generate a default recipe
- Use default training parameters
- Set aside 30% of your training data to evaluate the training
- Split the evaluation data sequentially

Custom
- Modify the recipe Amazon ML generates
- Modify training parameters
- Randomly or sequentially split your evaluation data

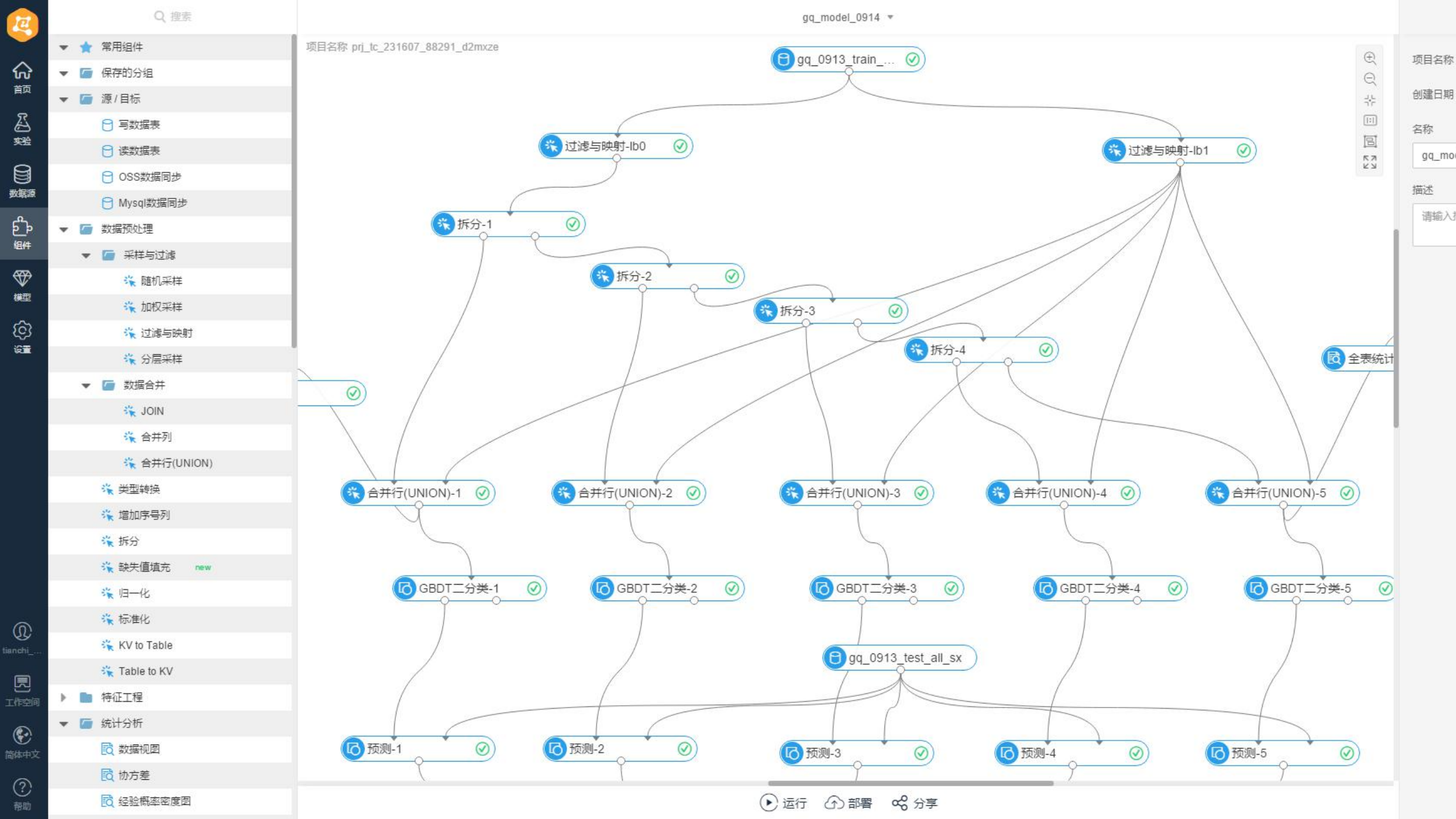
Evaluation Name Evaluation: ML model: Banking.csv

Cancel | Previous | Review



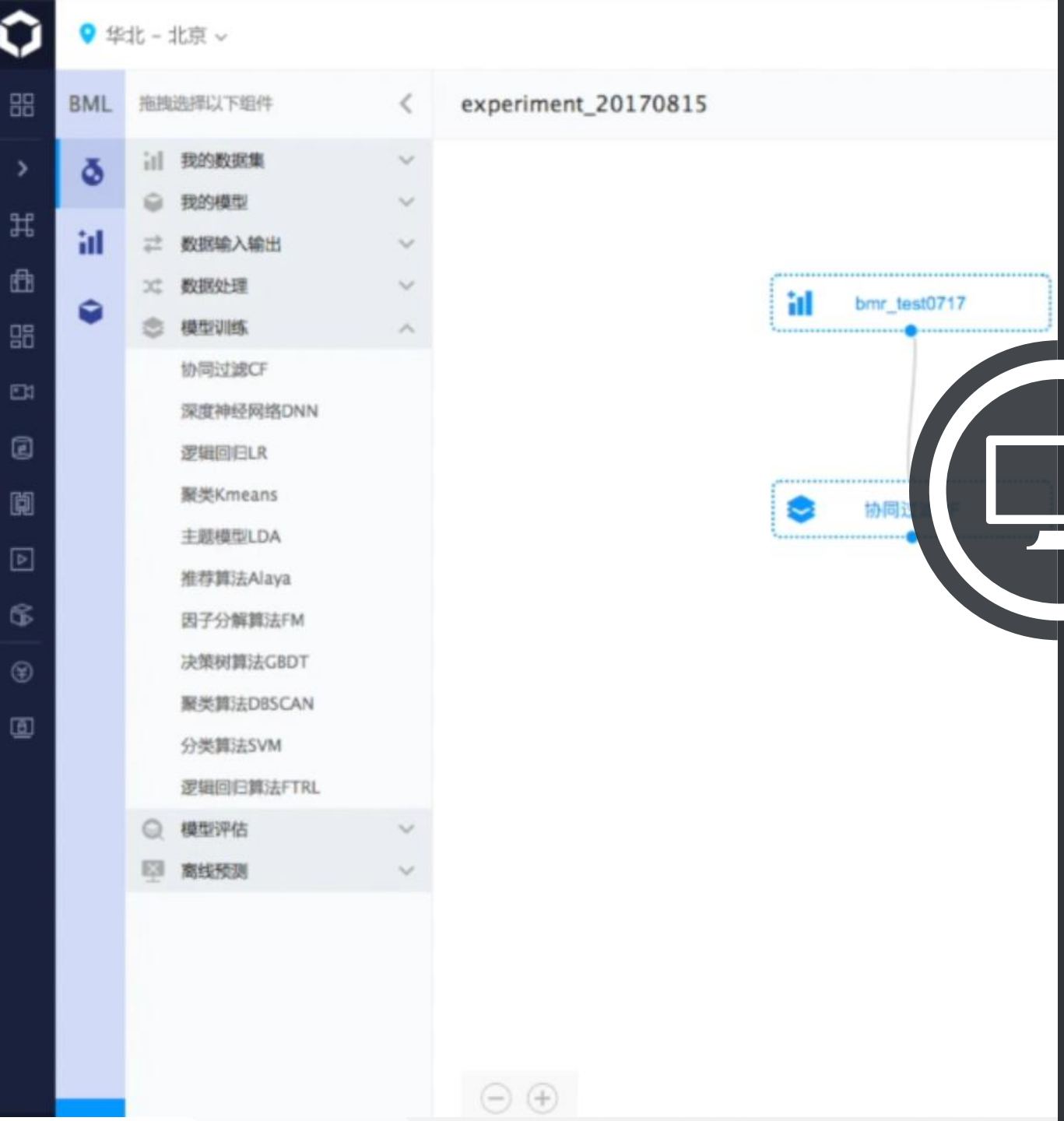
Amazon Machine Learning

- AWS 管理控制台和 API 提供了数据和模型可视化工具
- 创建机器学习模型、衡量模型质量
- 精细调整预测结果以符合应用程序要求
- 复选框形式的建模方式


gq_model_0914
项目名称 prj_tc_231607_88291_d2mxze
首页
实验
数据源
组件
模型
设置
搜索
常用组件
保存的分组
源 / 目标
写数据表
读数据表
OSS数据同步
Mysql数据同步
数据预处理
采样与过滤
随机采样
加权采样
过滤与映射
分层采样
数据合并
JOIN
合并列
合并行(UNION)
类型转换
增加序号列
拆分
缺失值填充
new
归一化
标准化
KV to Table
Table to KV
特征工程
统计分析
数据视图
协方差
经验概率密度图
gq_0913_train_...
过滤与映射-lb0
过滤与映射-lb1
拆分-1
拆分-2
拆分-3
拆分-4
全表统计
合并行(UNION)-1
合并行(UNION)-2
合并行(UNION)-3
合并行(UNION)-4
合并行(UNION)-5
GBDT二分类-1
GBDT二分类-2
GBDT二分类-3
GBDT二分类-4
GBDT二分类-5
gq_0913_test_all_sx
预测-1
预测-2
预测-3
预测-4
预测-5
项目名称
创建日期
名称
描述
请输入
运行
部署
分享
工作空间
简体中文
帮助

阿里云机器学习PAI

- 上手简单，提供拖拉拽的可视化操作环境
- 提供特征工程、数据预处理、统计分析、机器学习、深度学习多种算法
- 提供了在线预测以及离线调度
- 也支持写 SQL



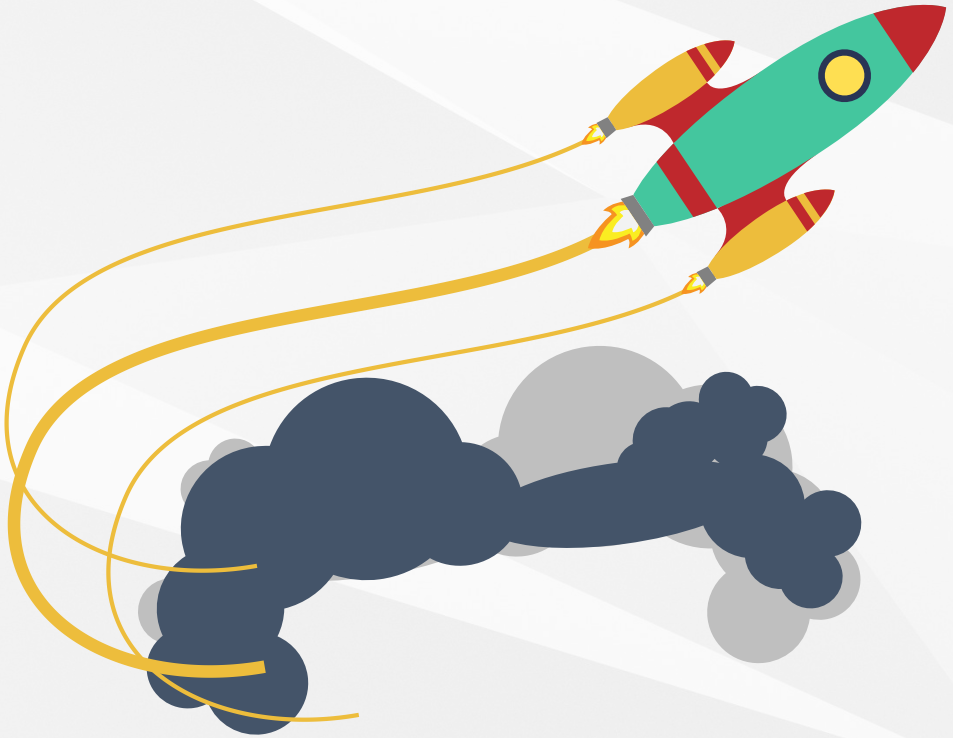
百度机器学习BML

- 支持连接百度用户画像数据
- 算法丰富
- 操作简单
- 高效的数据预处理

大型机器学习平台比较

表 1.1 各大机器学习平台对比

	实验建模方式	界面可交互程度	R/Python扩展	是否支持SQL	多算法支持度	模型评估方式
微软机器学习 AMLS	组件拖拽	高	支持	是	高	多样
Amazon Machine Learning	选择和点击	低	无	否	低	基本
阿里云机器学习 PAI	组件拖拽	高	无	是	高	较丰富
百度机器学习 BML	组件拖拽	高	无	否	中	基本



Meaning of Machine Learning Platform

本文认为机器学习平台的意义在于，通过交互的界面降低了技术门槛，使用者可以轻松实现数据挖掘的工作，而无需太多经验；其次，机器学习平台提供了从元数据到模型部署整套流程，通过提供基本的组件，使用者可以搭建各个垂直场景下的解决方案。

1. 绪论

1 第1章

探索机器学习的研究现状，分析国内外各大机器学习平台的优缺点并进行比较

2 第2章 3 第3章

研究和掌握机器学习平台开发所需的相关技术

4 第4章

结合实际项目需要，分析机器学习平台的开发需求并设计平台的逻辑架构

5 第5章

编程实现分布式机器学习平台的前端界面和后端数据计算逻辑

6 第6章

总结和展望

多核并行

- 刘闯. 基于多核计算的分类数据挖掘算法研究[D]. 南京:南京航空航天大学, 2011.]
- 该文献实现了多核上的一些机器学习算法，包括 KNN（K-nearest neighbor）、决策树、贝叶斯网络等

Hadoop Map Reduce

- 卫洁, 石洪波, 冀素琴. 基于 Hadoop 的分布式朴素贝叶斯文本分类[J]. 计算机系统应用, 2012, 21(2): 210-213.
- 该文献实现了基于Hadoop的海量文本挖掘的贝叶斯算法

Spark MLlib

- Machine Learning Library (MLlib) Guide [EB/OL]. (2014) [2014-02-01]. https://spark.apache.org/docs/latest/.
- 基于Spark的并行化机器学习算法

Tensorflow GPU/TPU

- https://www.tensorflow.org
- 谷歌第二代机器学习系统，深度学习表现出色

1. 问题定义和需求分析

- 查阅文献资料，撰写课题开题报告
- 需求分析和技术准备（R、Hive、Shiny、Spark）

2017年8月到10月

2. 平台设计和初步实现

- 依据机器学习的基本流程设计平台通用的模块和页面，以及每个模块应该具备的功能和主要实现方式。

2017年10月到11月

3. 程序开发阶段

- 利用R语言开发出一个简单易用的机器学习平台，并利用CSS、JavaScript等前端技术增强平台交互性、可用性，并利用Spark进行分布式扩展。

2017年11月到12月

4. 平台测试和发布

- 利用真实数据集对平台进行测试，并在服务器上进行发布
- 撰写论文正文、答辩

2017.12到2018.03

浙江大学
ZHEJIANG UNIVERSITY

CONTENTS

1 绪论

2 关键技术与框架

3 机器学习算法分析

4 需求分析与架构设计

5 平台实现及成果展示

6 总结与展望

关键技术与框架

2. 关键技术与框架

关键技术 | 主要框架

- 数据 → 训练 → 模型 → 预测
- 对机器学习最为常见的分类是有监督学习（Supervised Learning）和无监督学习（Unsupervised Learning）两大块，区分的依据就是训练数据是否包含因变量"y"

01

02

- 数据科学领域最为流行的编程语言之一(另一个为Python)
- 在机器学习领域有丰富的扩展包可使用
- 提供Spark和Hive的接口

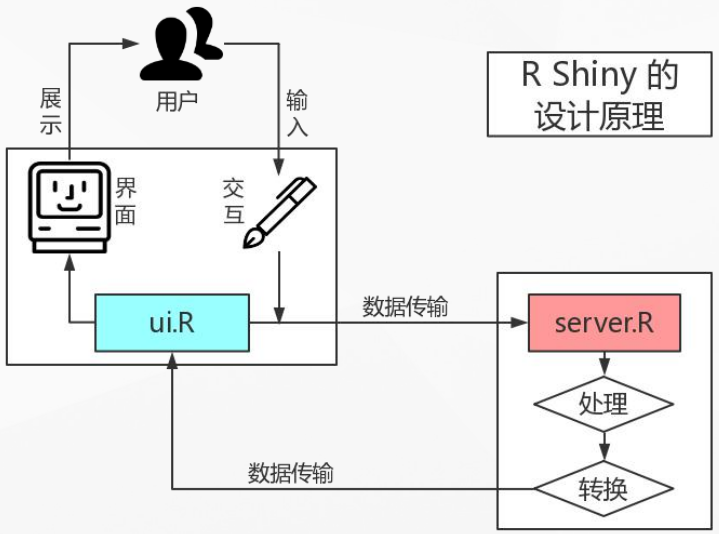
03

- 高效的分布式计算框架
- 基于RDD（Resilient Distributed Datasets）的内存式集群计算，可以提高应用程序的处理和响应速度，官方宣称Spark相对于MapReduce有10到100倍的速度提升

04

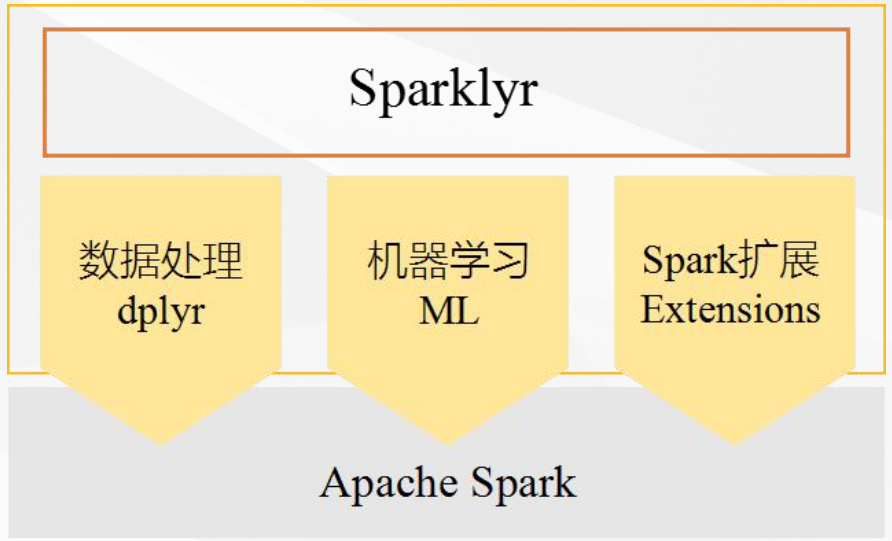
- 分布式数据仓库
- Hive的特点包括：存储schema、底层为HDFS、为OLAP多维分析设计、支持Hive SQL且查询较快、较为成熟、容易扩展等

2. 关键技术与框架



Web开发框架

Shiny



算法集成框架

Caret

分布式数据接口

Sparklyr

Classification and Regression Training

浙江大学
ZHEJIANG UNIVERSITY

CONTENTS

1 绪论

2 关键技术与框架

3 机器学习算法分析

4 需求分析与架构设计

5 平台实现及成果展示

6 总结与展望

机器学习算法分析

3. 机器学习算法分析

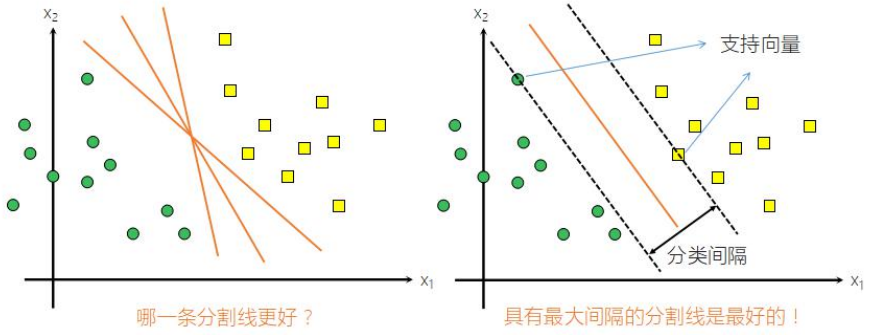
有监督学习 | 无监督学习

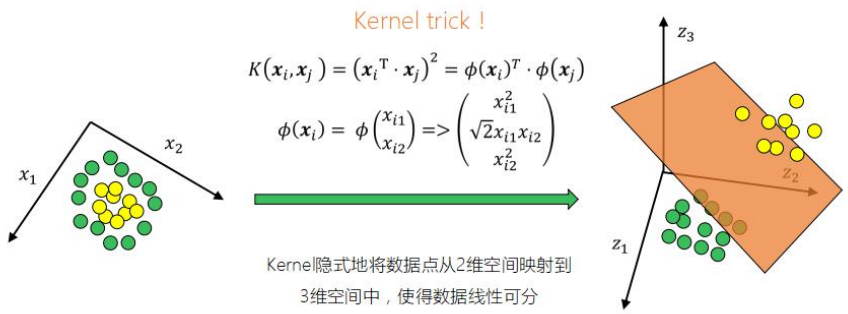
SVM | Random Forest | GBDT | XGBoost　　Apriori | Eclat

- 支持向量机SVM的基本目标可以转化为如下带约束的优化问题：

$$\underset{w}{\text{minimize}} \quad \frac{\|w\|^2}{2}$$

$$\text{subject to} \quad y_i(\vec{w}\cdot\vec{x}_i + b) \geq 1, \quad i = 1,2,\ldots,N$$





- 随机森林(RandomForest)的思想易于理解，简单点说就是决策树算法加上Bagging的集成。

- 所谓"森林"就是形象的指很多棵决策树；所谓"随机"是指每棵树的生成依赖的算法是一样的，不同的是仅仅所依赖的数据是随机在原始数据中进行有放回的重采样获得

- 随机森林的优点包括：
  a) 对于很多数据，它都可以保持较高的准确率。
  b) 在输入特征数量很多的时候，依然快速有效。
  c) 它可以很好的在不太影响模型准确率的同时处理缺失特征。
  d) 它对异常值的鲁棒性很强。
  e) 它可以一定程度上对类别不均匀的样本进行平衡。
  f) 它可以在模型训练完成后输出特征的重要性评估。
  g) 学习过程是比较快速的。

3. 机器学习算法分析

有监督学习 | 无监督学习

SVM | Random Forest | GBDT | XGBoost　　Apriori | Eclat

- GBDT（Gradient Boosting Decision Tree），翻译为"梯度提升决策树"，是梯度下降版的决策树提升模型。

记决策树的参数为 w，即 $f_t(x) = f_t(x;w)$

输入：样本 (x_i, y_i)，损失函数 $L(y_i, \hat{y}_i^{(t)})$

执行：

1. 初始化：$f_0(x) \leftarrow 0$
2. *For t = 1 to T do:*

   2.1 计算目标响应：$\hat{f}_t = -\dfrac{\partial L(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}}$

   2.2 学习第 t 棵树的参数：$w_t^* \doteq \underset{w}{argmin} \sum_{i=1}^{N} (\hat{f}_t - f_t(x;w))^2$

   2.3 寻找最优步长（可选）：$\alpha_t^* \doteq \underset{\alpha}{argmin} \sum_{i=1}^{N} L(y_i, \hat{y}_i^{(t-1)} + \alpha \cdot f_t(x_i; w^*))$

   2.4 生成第 t 棵树并更新模型：

$$f_t(x) \doteq \alpha^* \cdot f_t(x;w^*)$$

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + f_t(x)$$

输出：$\hat{y}^{(T)}$

- XGBoost（eXtreme Gradient Boosting），意为"极限梯度提升"，是牛顿迭代版的决策树提升模型。

输入：当前节点的样本集合 I，m 个特征

执行：

1. 初始化：$Gain \leftarrow 0$，$G \leftarrow \sum_{i \in I} g_i$，$H \leftarrow \sum_{i \in I} h_i$

2. 外层循环遍历 m 个特征：

   For k = 1 to m do:

   2.1 左节点初始化：$G \leftarrow 0$，$H \leftarrow 0$

   2.2 内层循环寻找该特征最佳分割点：

   For j in sorted(I, by x_k):

$$G_L = G_L + g_j,\ H_L = H_L + h_j$$

$$G_R = G - G_L,\ H_R = H - H_L$$

$$score = \max(score, \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda})$$

循环结束记录 *max(score)* 及 *j* 的值

循环结束记录 k 的值

输出：*max(score)* 下的最佳分割特征和分割点

3. 机器学习算法分析

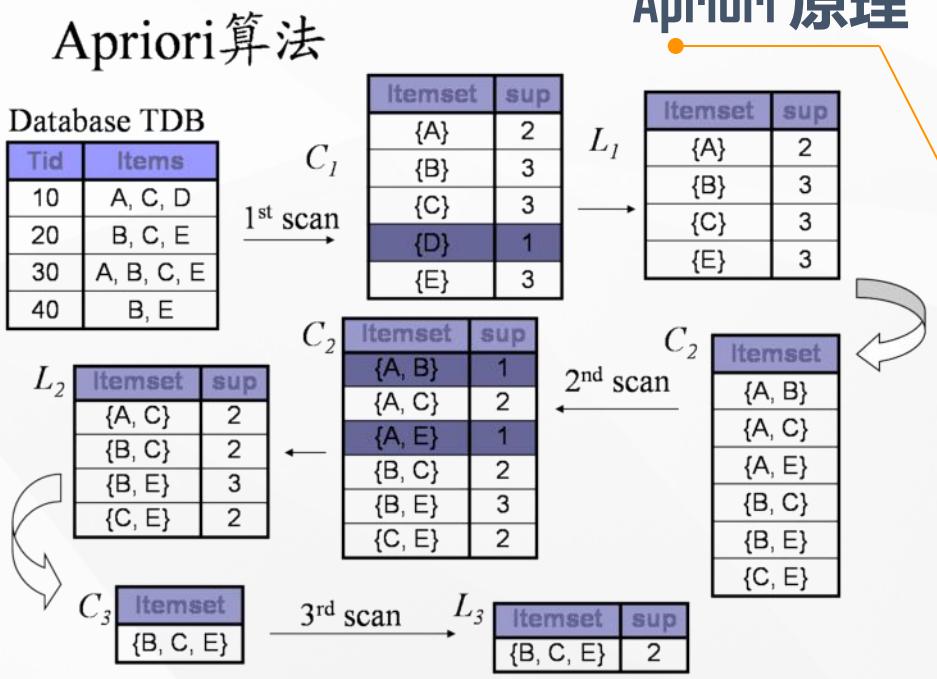
有监督学习 | 无监督学习

SVM | Random Forest | GBDT | XGBoost

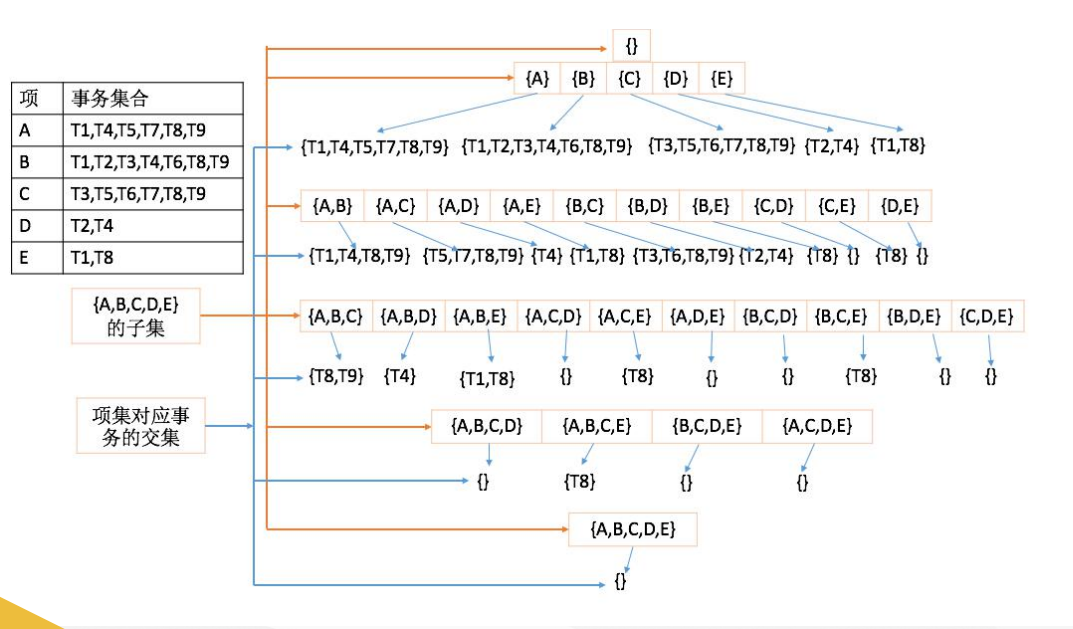
Apriori | Eclat

- Apriori 和 Eclat 都是频繁项集挖掘算法，最常用于"购物篮分析"问题，该机器学习平台借鉴其算法思路用于挖掘客户的共性特征。

Apriori 原理



- 之所以取名为Apriori，是由于其核心思想是先验准则——如果某个项集是频繁的，那么由于子集支持度必然不小于原集支持度，故该项集所有子集肯定也是频繁的。



Eclat 原理

- Eclat 全称是 "Equivalence Class Transformation"，其特点在于利用的是商品（item）对购物篮（basket）而非购物篮对商品的"倒排"的方式进行频繁项集挖掘。"倒排"处理是一种等价转换。

浙江大学
ZHEJIANG UNIVERSITY

CONTENTS

1 绪论

2 关键技术与框架

3 机器学习算法分析

4 需求分析与架构设计

5 平台实现及成果展示

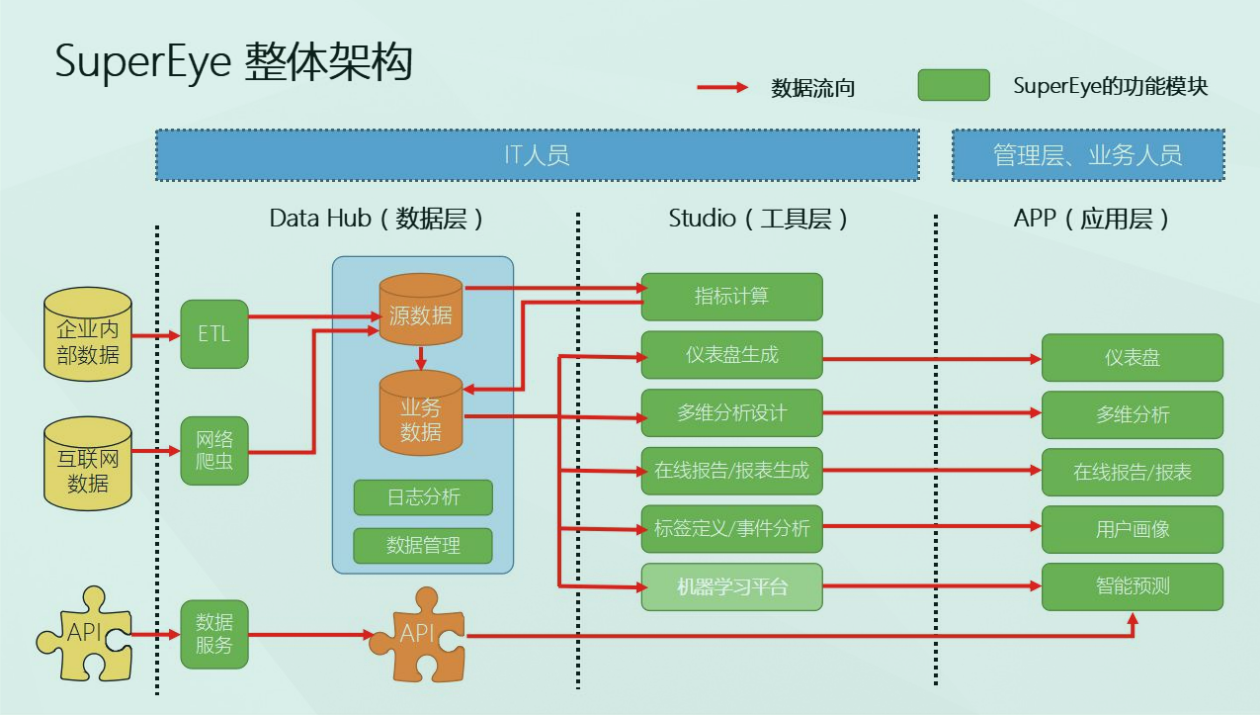
6 总结与展望

需求分析与架构设计

4. 需求分析与架构设计

✓ 某公司的SuperEye智能运营系统

- 机器学习平台属于SuperEye Studio层，它承担着支撑对企业数据的智能分析及预测的功能。由于机器学习平台是SuperEye系统的重要组成部分之一，因此该机器学习平台被命名为 SEML，为SuperEye Machine Learning的简称



✓ 某证券公司两融业务的客户分析需求

- 针对客户群对比和单客群分析的功能，该证券公司希望通过使用机器学习的手段能自动对客户数据进行分析，而减少人工操作、降低人工成本

SEML

SEML需要实现的目标是：

a) 依据人机交互的原则，开发出一个初级机器学习工程师也能够便捷使用，交互体验较好的机器学习平台，该平台可以完整实现机器学习的基本流程，并提供分布式数据库的数据源接口。

b) 提供常用的多种机器学习算法接口，拥有进一步开发的潜力。

c) 结合某证券公司的具体业务，有针对性的开发出定制化的功能。

4. 需求分析与架构设计

机器学习的一般流程：

- 在机器学习领域，数据的流向通常具有一定的方向性。
- 通常来说，一次完整的机器学习过程应当包括五个步骤：数据源获取、数据预处理、特征工程、模型训练和评估、预测推断

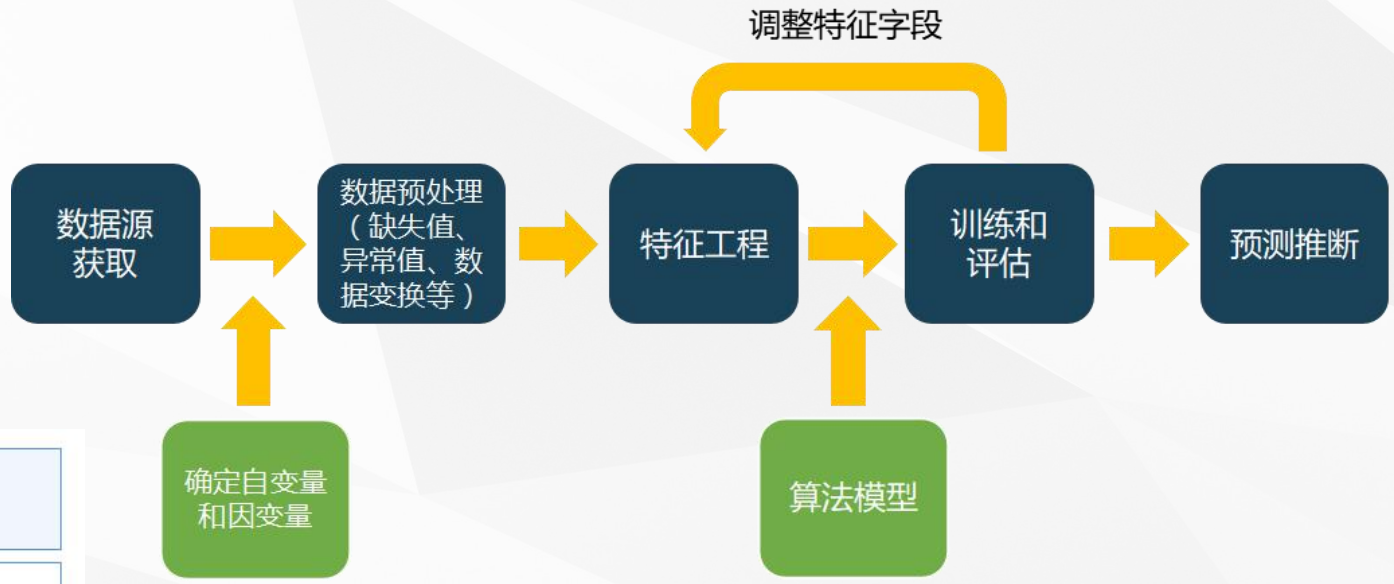


图 4.2 机器学习的一般流程

图 4.3 SEML的整体架构

- 基于有监督学习的标准化模块
  - 数据导入模块
  - 模型训练模块
  - 预测推断模块
- 基于业务场景的定制化模块
  - 客户差异性特征识别
  - 客户共性特征发现
- 模块独立性：可以让机器学习平台根据不同需求进行变化和更新，使其具备更好的灵活性和可扩展性

浙江大学
ZHEJIANG UNIVERSITY

CONTENTS

1 绪论

2 关键技术与框架

3 机器学习算法分析

4 需求分析与架构设计

5 平台实现及成果展示

6 总结与展望

平台实现及成果展示

5. 平台实现及研究成果

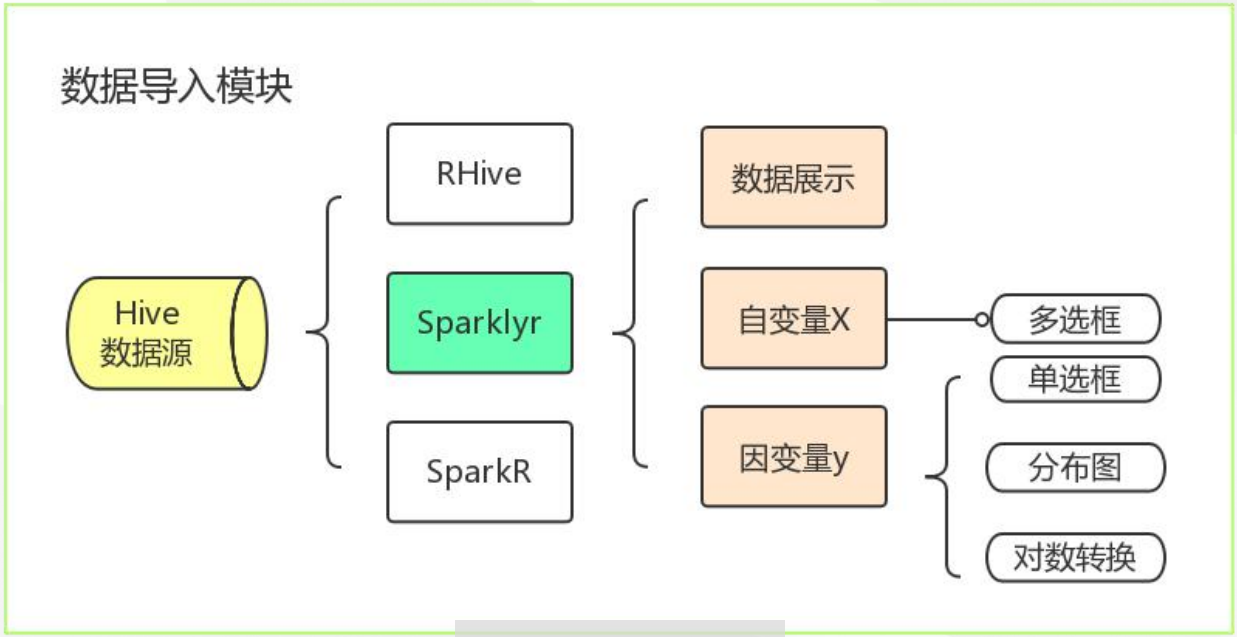
数据导入模块 | 模型训练模块 预测推断模块 | 定制化模块

```
...
## Global
# 加载依赖包
library(Sparklyr)
library(rJava)
library(DBI)
# 通过Sparklyr访问Hive数据库
sc <- spark_connect(master = "yarn-client", app_name = "machine_learning",
                    version = "1.6.0", hadoop_version = "2.6.0",
                    spark_home = "/opt/cloudera/parcels/CDH-5.10.0-1.cdh5.
# 查询Hive数据库
databases <- dbGetQuery(sc, "show databases")$result

## UI端
# 数据查看
actionButton('btn_viewData',label = 'View Data',icon=icon('table'))
bsModal('data',title = 'Dataset',trigger = 'btn_viewData',
        size = 'large', DT::dataTableOutput('rawdata')
)
# 特征观察与对数变换
conditionalPanel(condition = "output.YtypePrint.substr(6) == 'numeric' |
                 output.YtypePrint.substr(6) == 'integer'",
                 checkboxInput('chk_logY',label = 'log transform'))
plotOutput('Yplot', height=260)
tableOutput('Ystats')
...

## Server端
# 数据表选择框即时更新
observe({
  updatePickerInput(session = session, inputId = "HiveTable", choices = ta
})
# 因变量yvar的选择框即时更新
observe({
  tables <- dbGetQuery(sc, paste0("show tables in ", input$HiveDatabase))
  updateSelectizeInput(session, 'yvar', choices=names(rawdata()),
                       selected = tail(names(rawdata()),1))
})
...
```

关键代码实现



模块逻辑设计

研究成果界面展示

192.168.1.22:10831

SUPEROYE · 机器学习

- Step 1: 数据导入与查看
- Step 2: 模型训练和调参
- Step 3: 模型保存及预测
  - 使用模型预测
  - 可用模型列表
- 差异性特征识别
- 共性特征发现

n obs: 2507

Y var: total_asset

Input Dataset

请选择数据库： guquan

请选择数据表： traindf

View Data

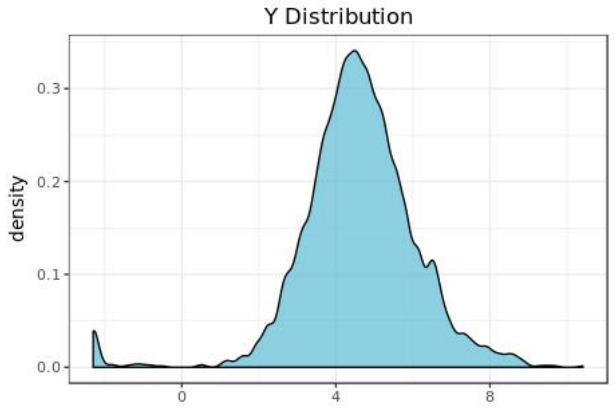
y variable

请选择因变量（目标变量）：

y var

total_asset

变量类型： numeric



log transform

summary	value
Min.	-2.30
1st Qu.	3.78
Median	4.59
Mean	4.57

X vars

自变量（多选）：

customer_gender degree_code profession_code customer_income industry_type en_contact_type contact_freq socialral_type customer_from corp_risk_level current_balance stock_value bail_balance fin_balance total_debit credit_assure_scale profit_day_sum turnover_rate max_total_asset min_total_asset max_total_debit min_total_debit max_total_fund min_total_fund max_total_stock min_total_stock max_total_fin min_total_fin max_total_slo max_income min_income stock_hold concentrate open_years_level exchange_frequency fund_in_3mago fund_out_3mago stock_in_3mago stock_out_3mago age open_days IShigh

研究成果界面展示

192.168.1.22:10831

SUPERGYE · 机器学习

Step 1: 数据导入与查看

Step 2: 模型训练和调参

Step 3: 模型保存及预测

使用模型预测

可用模型列表

差异性特征识别

共性特征发现

n obs: 2507

Y var: total_asset

Dataset

Show 10 entries

Search:

	customer_gender	degree_code	profession_code	customer_income	industry_type	en_contact_type
1	男	中专	企事业单位干部	150000	missing	
2	男	大专	商贸	5	missing	
3	男	学士	行政企事业单位工人	500000	房地产业	3
4	女	高中	商贸	100000	missing	4
5	女	高中	行政企事业单位工人	1000000	制造业	3
6	女	大专	行政企事业单位工人	1000000	其他	
7	男	大专	金融	50000	missing	
8	男	大专	商贸	100000	其他	3
9	女	大专	行政企事业单位工人	500000	其他	3
10	男	大专	个体	200000	其他	

Showing 1 to 10 of 2,507 entries

Previous 1 2 3 4 5 … 251 Next

Close

5. 平台实现及研究成果

数据导入模块 | **模型训练模块**

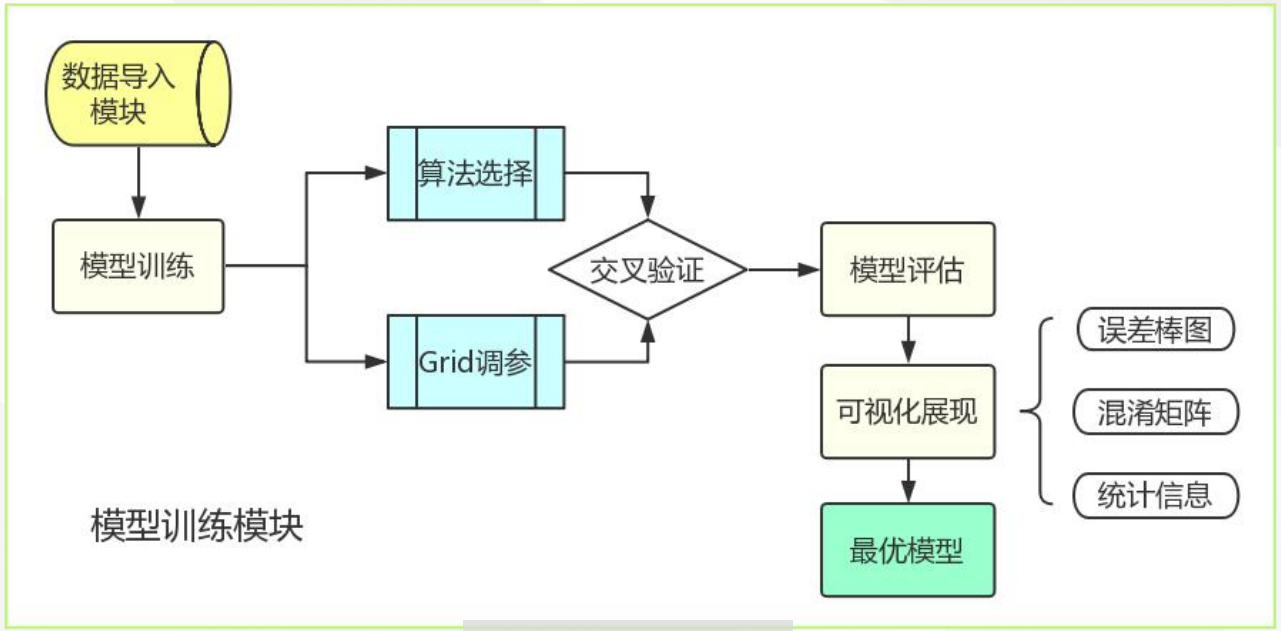
预测推断模块 | 定制化模块

```r
...
## Global
# 参数空间
tuneParams <- list(
  'svmLinear'=data.frame(C=c(0.01,0.1,1)),
  'rf' = expand.grid(mtry = c(3,4,5)),
  'gbm' = expand.grid(n.trees = c(50,100,200),
                      n.minobsinnode = 10,
                      interaction.depth = c(3,4,5),
                      shrinkage = c(0.01,0.1)),
  'xgbLine' = expand.grid(nrounds=c(5,10), eta=c(0.01,0.1),
                          lambda=c(0,1), alpha=c(0,1)),
  'xgbTree' = expand.grid(nrounds=c(10,50,100), max_depth=c(3,4,5),
                          eta=c(0.01,0.1), gamma=0,
                          colsample_bytree=c(0.5,1),
                          min_child_weight=1, subsample=1)
)

## UI端
# 算法多选框
selectInput('slt_algo',label = '机器学习算法：'%>%label.help('lbl_algo'),
            choices = reg.mdls, selected = reg.mdls, multiple=T)
# 交叉验证选择器
radioButtons('rdo_CVtype',label = '交叉验证：'%>%label.help('lbl_CV'),
             choices = c('3-fold'=3,'5-fold'=5,'10-fold'=10),inline = T)

## Server端
# 模型信息生成函数
getRes <- function(i){
  Model <- names(fits)[i]
  res <- fits[[i]]$results
  df <- res[(ncol(res)-3):ncol(res)]
  apply(res,1,function(r) paste(r[1:(ncol(res)-4)],collapse = '-')) %>%
    paste(Model,.,sep='-') -> modelName
  cbind.data.frame(modelName,df,Model=Model[[1]],stringsAsFactors =F)
}
# 循环调用生成展示信息
df <- plyr::ldply(1:length(fits), getRes)
...
```

关键代码实现



模块逻辑设计

研究成果界面展示

192.168.1.22:10831

SUPEROYE · 机器学习

- Step 1: 数据导入与查看
- Step 2: 模型训练和调参
- Step 3: 模型保存及预测
  - 使用模型预测
  - 可用模型列表
- 差异性特征识别
- 共性特征发现

n obs: 2507

Y var: IShigh

模型控制

机器学习算法：

svmLinear | Random Forest | Gradient Boosting | Xgboost Linear | Xgboost Tree

调参策略

Coarse auto-tune (fast)

交叉验证：

3-fold　5-fold　10-fold

Train Models

模型总结

最优模型： xgbLine-10-0.01-1-0

准确率(Accuracy)： 0.9051

Kappa统计量： 0.7106

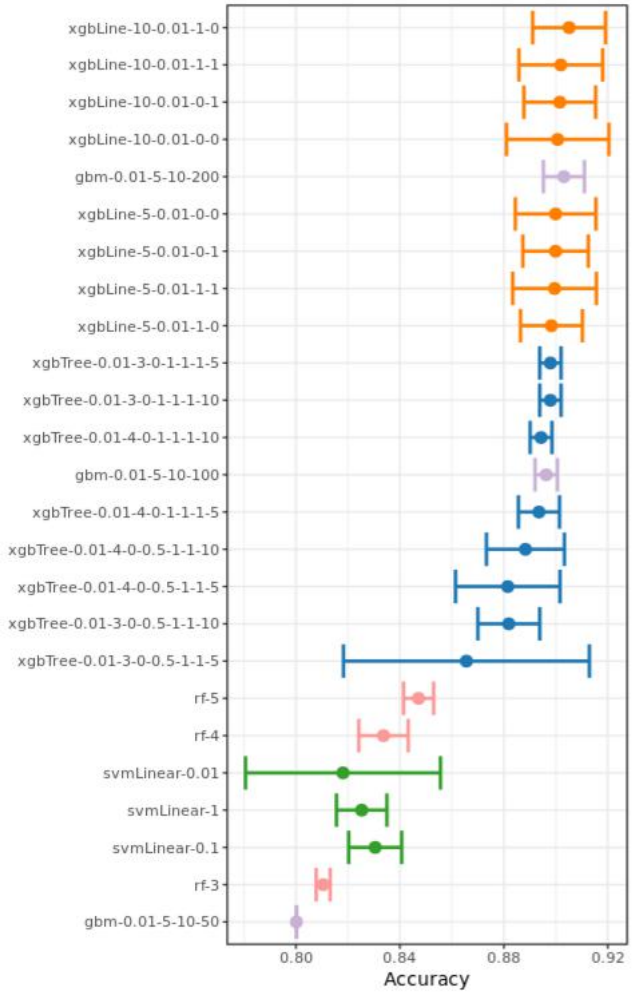
模型种类： Classification

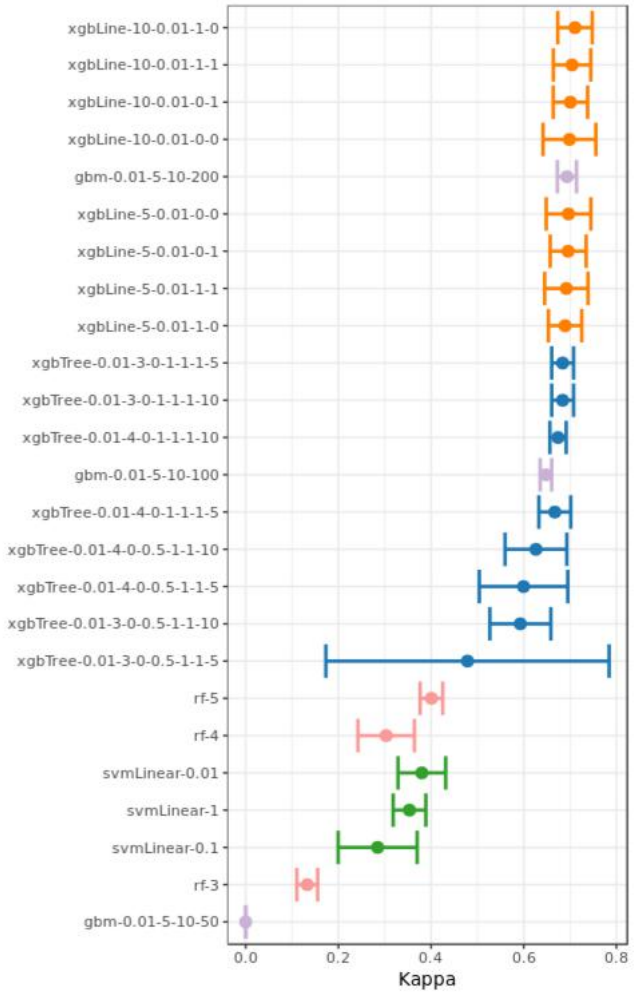
交叉验证折数： 3

模型训练总数： 25

模型综合排序　预测值 vs 真实值　模型统计信息

基于交叉验证的综合效果排序：





研究成果界面展示

192.168.1.22:10831

SUPEROYE · 机器学习

- Step 1: 数据导入与查看
- Step 2: 模型训练和调参
- Step 3: 模型保存及预测
  - 使用模型预测
  - 可用模型列表
- 差异性特征识别
- 共性特征发现

n obs: 2507

Y var: IShigh

模型控制

机器学习算法：

svmLinear | Random Forest | Gradient Boosting | Xgboost Linear | Xgboost Tree

调参策略

Coarse auto-tune (fast)

交叉验证：

3-fold　5-fold　10-fold

Train Models

模型总结

最优模型： xgbLine-10-0.01-1-0

准确率(Accuracy)： 0.9051

Kappa统计量： 0.7106

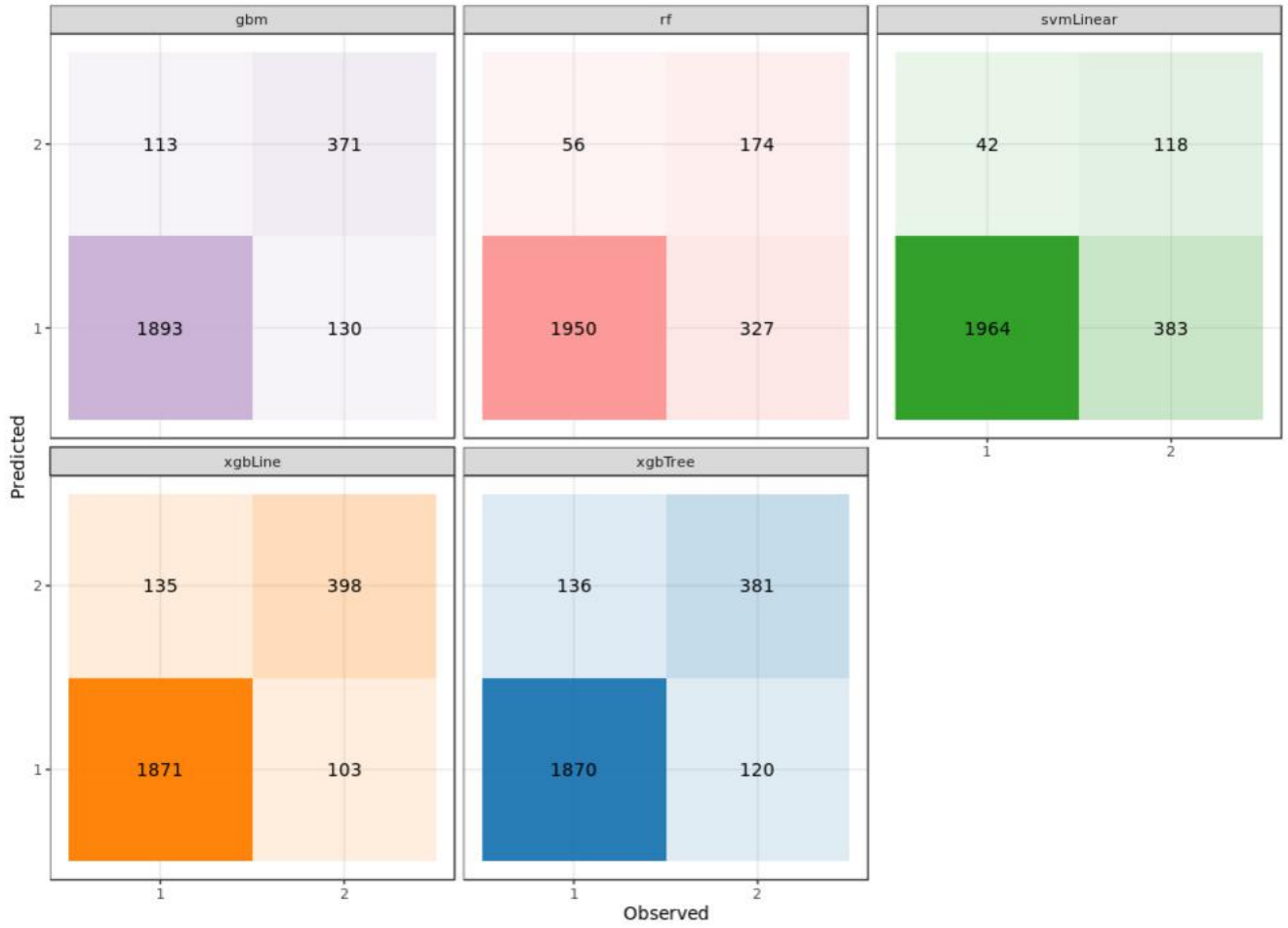
模型种类： Classification

交叉验证折数： 3

模型训练总数： 25

模型综合排序 | 预测值 vs 真实值 | 模型统计信息

选择每种算法中的最优模型（参数）进行预测值和真实值的比较：



研究成果界面展示

192.168.1.22:10831

SUPERGYE · 机器学习

- Step 1: 数据导入与查看
- Step 2: 模型训练和调参
- Step 3: 模型保存及预测
  - 使用模型预测
  - 可用模型列表
- 差异性特征识别
- 共性特征发现

n obs: 2507

Y var: IShigh

模型控制

机器学习算法：

svmLinear | Random Forest | Gradient Boosting | Xgboost Linear | Xgboost Tree

调参策略

Coarse auto-tune (fast)

交叉验证：

3-fold　5-fold　10-fold

Train Models

模型总结

最优模型： xgbLine-10-0.01-1-0

准确率(Accuracy)： 0.9051

Kappa统计量： 0.7106

模型种类： Classification

交叉验证折数： 3

模型训练总数： 25

模型综合排序　预测值 vs 真实值　模型统计信息

基于交叉验证的模型评价和统计量信息：

Show 10 entries

rank	modelName	Accuracy	Kappa	AccuracySD	KappaSD	Model
1	xgbLine-10-0.01-1-0	0.9051	0.7106	0.0140	0.0372	xgbLine
2	xgbLine-10-0.01-1-1	0.9019	0.7042	0.0161	0.0404	xgbLine
3	xgbLine-10-0.01-0-1	0.9015	0.7007	0.0138	0.0371	xgbLine
4	xgbLine-10-0.01-0-0	0.9007	0.6986	0.0197	0.0570	xgbLine
5	gbm-0.01-5-10-200	0.9031	0.6932	0.0079	0.0206	gbm
6	xgbLine-5-0.01-0-0	0.8999	0.6966	0.0155	0.0482	xgbLine
7	xgbLine-5-0.01-0-1	0.8999	0.6958	0.0126	0.0388	xgbLine
8	xgbLine-5-0.01-1-1	0.8995	0.6919	0.0161	0.0468	xgbLine
9	xgbLine-5-0.01-1-0	0.8983	0.6892	0.0119	0.0359	xgbLine
10	xgbTree-0.01-3-0-1-1-1-5	0.8979	0.6839	0.0041	0.0236	xgbTree

Showing 1 to 10 of 25 entries

Previous　1　2　3　Next

5. 平台实现及研究成果

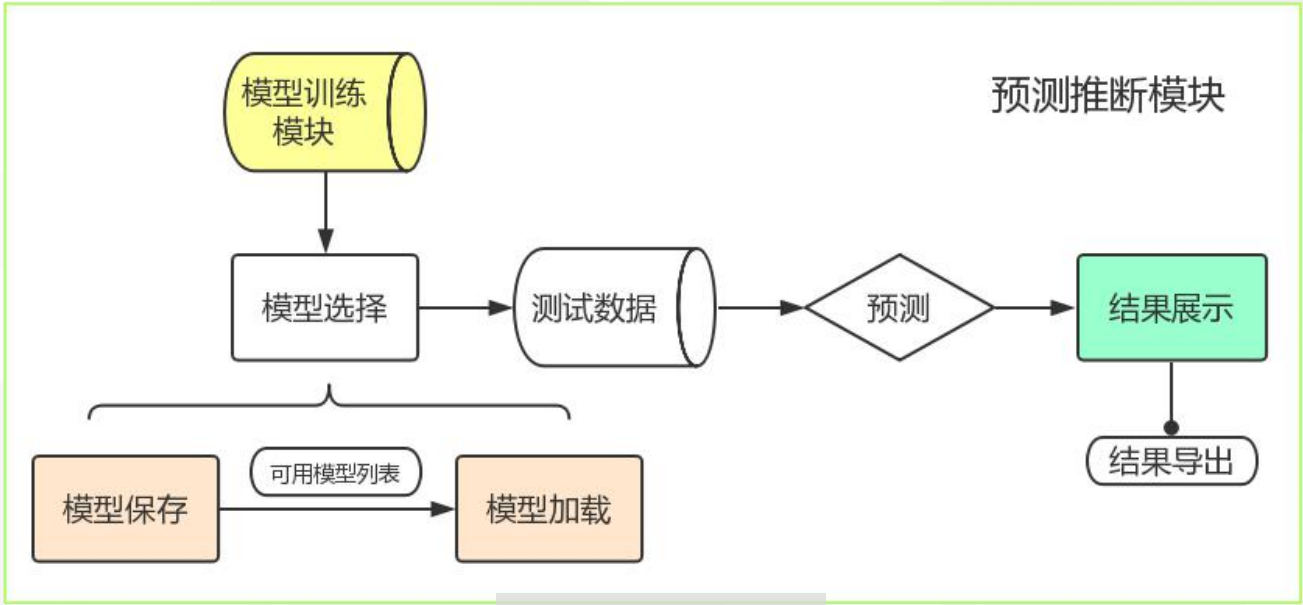
数据导入模块 | 模型训练模块

预测推断模块 | 定制化模块

```
...
## UI端
# 模型保存按钮
shinySaveButton('modelSave', '保存当前模型', '保存当前选择：',
                filetype=list(data='Rdata'), buttonType='success')
# 模型加载按钮
shinyFilesButton('modelSelect', '试试其他模型？',
                 '加载其他模型后当前模型将失效',
                 multiple=FALSE, buttonType='info')
...

## Server端
observe({
    # 模型保存
    currentModels <- rbind(currentModels, thisModel)
    write.csv(currentModels, './currentModels.csv', row.names = F)
    sendSweetAlert(
      messageId = "modelOut", title = "模型保存成功！", type = "success"
    )
    ...
    # 模型加载
    Finalalgor <- readRDS(path2)
    sendSweetAlert(messageId = "modelIn", title = "模型加载成功！",
                   text = "当前模型已被替换", type = "warning")
    Finalalgor <- CVtune[[input$slt_Finalalgo]]
    topList = topModels()
    algorithm = names(topList)[topList==input$slt_Finalalgo]
    ...
    # 预测推断
    preds = factor(preds)
    levels(preds) = levels(factor(rawdata()[[input$yvar]]))
    testRes[['Preds']][noMiss] = as.character(preds)
    probs = predict.train(Finalalgo, dataTest(), type = 'prob')
    probs = round(probs, 4)
    names(probs) = paste0('Prob_', levels(preds))
    for(pCol in names(probs)){
      testRes[[pCol]] = NA
      testRes[[pCol]][noMiss] = probs[[pCol]]
    }
})
```

关键代码实现



模块逻辑设计

研究成果界面展示

192.168.1.22:10831

SUPEROYE · 机器学习

- Step 1: 数据导入与查看
- Step 2: 模型训练和调参
- Step 3: 模型保存及预测
  - 使用模型预测
  - 可用模型列表
- 差异性特征识别
- 共性特征发现

n obs: 2507

Y var: IShigh

模型应用

Final Model:

xgbLine-10-0.01-1-0

注：默认选择是所有算法中效果最好的模型

保存当前模型

请选择测试数据集：

testdf

试试其他模型？

模型名称：当前模型

xgbLine-10-0.01-1-0

90.51 %

Accuracy

预测结果

Copy CSV Excel

Search:

ıago	stock_in_3mago	stock_out_3mago	age	open_days	IShigh	Preds	Prob_0	Prob_1
0	0	0	45	6009	0	0	0.9704	0.0296
0	0	0	41	2176	1	1	0.1536	0.8464
0	0	0	36	3738	0	0	0.5615	0.4385
0	0	0	51	6138	0	0	0.8874	0.1126
2.5	0	0	47	6856	0	0	0.8981	0.1019
12.8	0	0	55	2730	0	0	0.8521	0.1479
0	0	0	50	6177	0	0	0.9708	0.0292
2	0	0	48	5812	0	0	0.9403	0.0597
0	0	0	47	503	0	0	0.9161	0.0839
0	0	0	33	3244	0	0	0.9708	0.0292

Showing 1 to 10 of 502 entries

Previous 1 2 3 4 5 … 51 Next

研究成果界面展示

192.168.1.22:10831
SUPEROYE · 机器学习
Step 1: 数据导入与查看
Step 2: 模型训练和调参
Step 3: 模型保存及预测
使用模型预测
可用模型列表
差异性特征识别
共性特征发现
n obs: 2507
Y var: IShigh
模型应用
Final Model:
xgbLine-10-0.01-1-0
注：默认选择是所有算法中效
请选择测试数据集：
testdf
模型名称：当前模型
xgbLine-10-0.01-1-0
90.51 %
Accuracy
加载其他模型后当前模型将失效
模型管理
111112.Rd...
gbm0946....
gbmIShigh...
guquan22....
nb12aaa.R...
svm1852.R...
test
xgbline201...
xTree2009....
取消
选择
Search:
customer_income
industry_type
en_contact_
Showing 1 to 10 of 502 entries
Previous
Next

研究成果界面展示

192.168.1.22:10831

SUPEROYE · 机器学习

- Step 1: 数据导入与查看
- Step 2: 模型训练和调参
- Step 3: 模型保存及预测
  - 使用模型预测
  - 可用模型列表
- 差异性特征识别
- 共性特征发现

n obs: 2507

Y var: IShigh

当前可用模型

Column visibility　刷新　　Search: 2017-07

	fileName	algorithm	createTime	type	evaluation	yvar
1	gbm1706.Rdata	gbm-0.01-5-10-200	2017-07-05 17:06:16	Regression-with-Log	Rsquared: 0.9154	stock_value
2	6666.Rdata	xgbLine-10-0.01-0-0	2017-07-06 16:13:46	two-Classfication	Accuracy: 0.8159	Survived
3	111112.Rdata	gbm-0.01-5-10-200	2017-07-06 16:26:55	two-Classfication	Accuracy: 0.9059	IShigh
4	55555.Rdata	svmLinear-1	2017-07-06 16:52:46	two-Classfication	Accuracy: 0.7924	Survived
5	svm1852.Rdata	svmLinear-1	2017-07-06 18:55:26	two-Classfication	Accuracy: 0.7969	Survived
6	66666.Rdata	gbm-0.01-5-10-200	2017-07-06 19:44:21	two-Classfication	Accuracy: 0.8238	Survived
7	nb12aaa.Rdata	xgbTree-0.01-3-0-1-1-1-10	2017-07-06 19:44:40	two-Classfication	Accuracy: 0.8103	Survived
8	guquan22.Rdata	xgbLine-10-0.01-1-0	2017-07-06 19:52:47	two-Classfication	Accuracy: 0.9075	IShigh
9	rf5-IShigh.Rdata	rf-5	2017-07-12 18:23:44	two-Classfication	Accuracy: 0.8496	ishigh

Showing 1 to 9 of 9 entries　　Previous　1　Next

5. 平台实现及研究成果

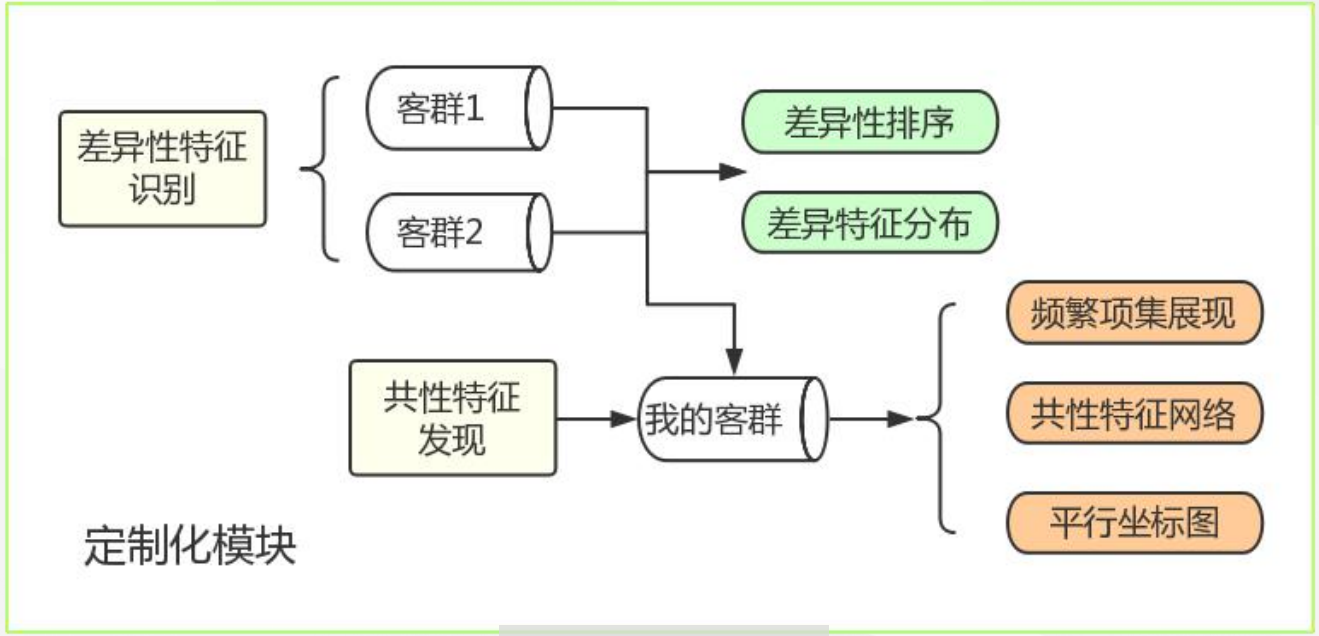
数据导入模块 | 模型训练模块
预测推断模块 | **定制化模块**

```
...

## UI端
# 利用Highchart可视化差异性分布
highchartOutput('diffChart', height = "600px")
# 共性特征网络图和平行坐标图
column(width = 4, plotOutput('graph')),
column(width = 8, parcoordsOutput('pcoords', width = "100%", height = "45

## Server端
# 特征重要性函数
feature_importance <- function(X, y, n_trees=50, random_state=2017, max_d
  set.seed(random_state)
  clf = randomForest(X, y, ntree=n_trees, maxnodes=2^max_depth)
  imp_df = as.data.frame(importance(clf))
  imp_df$feature = row.names(imp_df)
  imp_df = imp_df[order(-imp_df[,1]),]
  return(imp_df)
}
# 购物篮生成函数
generate_bskt <- function(df_dscrtz, colsEN, colsCN){
  allcols = colnames(df_dscrtz)
  df_cols = allcols[allcols != 'customer_no']
  for(col in df_cols){
    colCN = colsCN[which(colsEN==col)]
    df_dscrtz[[col]] = factor(df_dscrtz[[col]], exclude = NULL)
    setnames(df_dscrtz, old = col, new = colCN)
  }
  if('customer_no' %in% allcols){
    bskt = as(df_dscrtz[, -c('customer_no')], 'transactions')
    bskt@itemsetInfo$transactionID = df_dscrtz$customer_no
  } else {
    bskt = as(df_dscrtz, 'transactions')
  }
  return(bskt)
}
# 频繁项集挖掘
sets = Eclat(bskt, parameter = list(support = 0.7, minlen = 6),
             control = list(verbose = F))
...
```

关键代码实现



模块逻辑设计

研究成果界面展示

192.168.1.22:10831

SUPEROYE · 机器学习

- Step 1: 数据导入与查看
- Step 2: 模型训练和调参
- Step 3: 模型保存及预测
  - 使用模型预测
  - 可用模型列表
- 差异性特征识别
- 共性特征发现

n obs: 2507

Y var: IShigh

客群选择

客群1：

tb1

客群2：

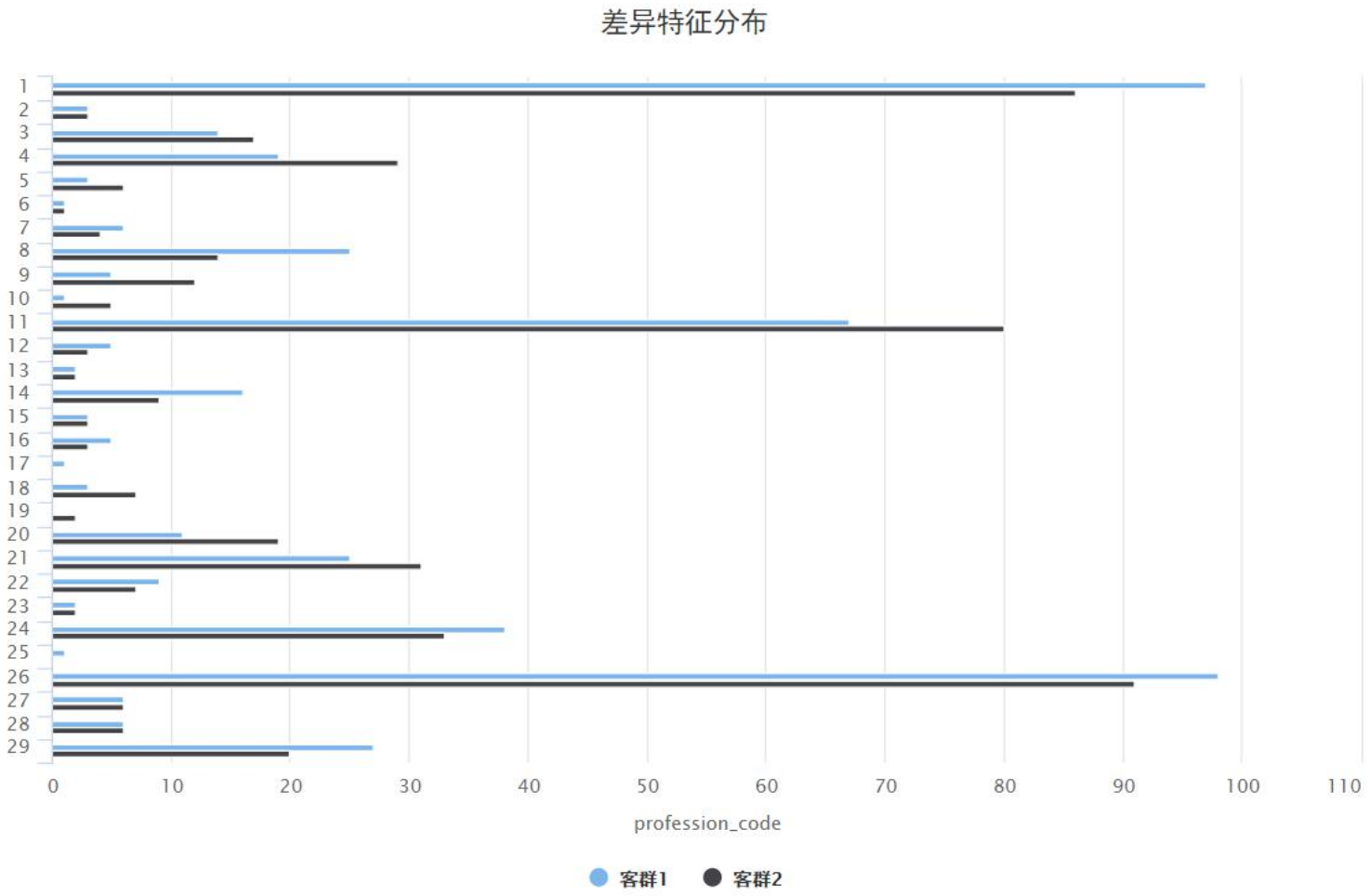
tb2

特征差异性排序

feature	importance
profit_day_sum	159.13
min_income	25.20
profession_code	13.62
max_income	10.15
industry_type	9.61
credit_assure_scale	5.64
customer_income	4.59
fin_balance	4.31
total_debit	4.00
turnover_rate	3.77

请选择特征：

profession_code



研究成果界面展示

192.168.1.22:10831

Step 2: 模型训练和调参

Step 3: 模型保存及预测

使用模型预测

可用模型列表

差异性特征识别

共性特征发现

n obs: 2507

Y var: IShigh

请选择客群：

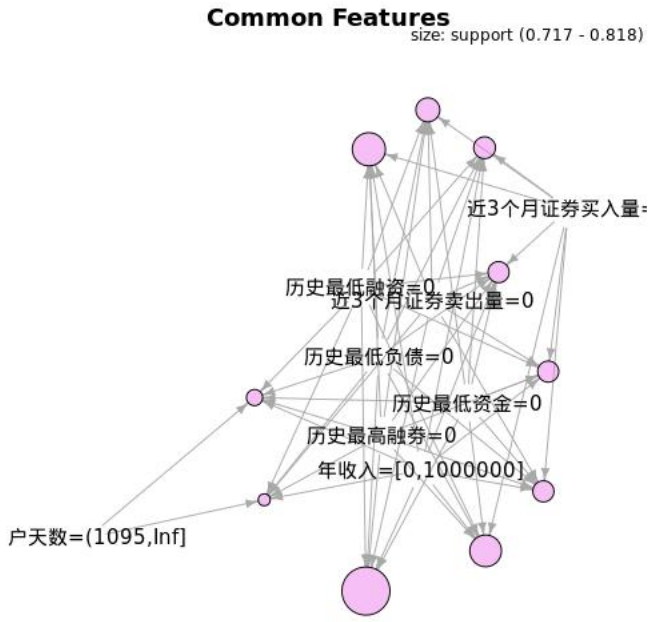
cust2

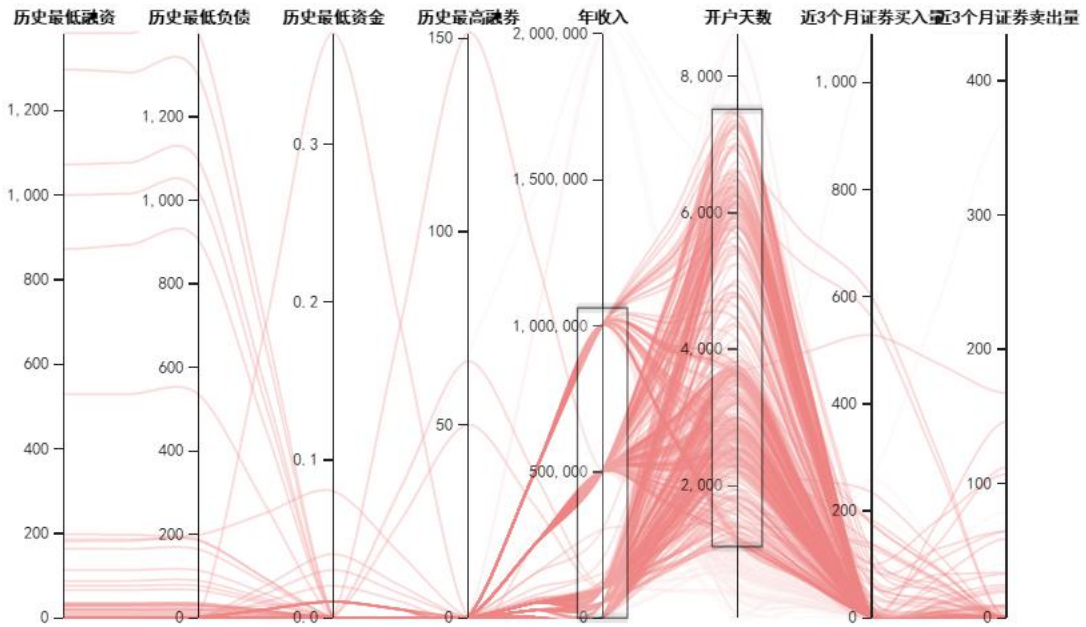
	items	support
1	{年收入=[0,1000000],历史最低负债=0,历史最低融资=0,历史最高融券=0,近3个月证券卖出量=0,开户天数=(1095,Inf]}	0.716566866267465
2	{年收入=[0,1000000],历史最低负债=0,历史最低资金=0,历史最低融资=0,历史最高融券=0,开户天数=(1095,Inf]}	0.728542914171657
3	{年收入=[0,1000000],历史最低负债=0,历史最低资金=0,历史最低融资=0,历史最高融券=0,近3个月证券买入量=0,近3个月证券卖出量=0}	0.74251497005988
4	{历史最低负债=0,历史最低资金=0,历史最低融资=0,历史最高融券=0,近3个月证券买入量=0,近3个月证券卖出量=0}	0.750499001996008
5	{年收入=[0,1000000],历史最低负债=0,历史最低资金=0,历史最低融资=0,近3个月证券买入量=0,近3个月证券卖出量=0}	0.744510978043912

Showing 1 to 5 of 10 entries

Previous 1 2 Next

共性可视化





浙江大学
ZHEJIANG UNIVERSITY

CONTENTS

1 绪论

2 关键技术与框架

3 机器学习算法分析

4 需求分析与架构设计

5 平台实现及研究成果

6 总结与展望

总结与展望

6. 总结与展望

工作总结 | 下步计划

本文在设计和实现平台的过程中，主要完成的工作列举如下：

a) 先将SEML粗分成标准化模块和定制化模块，然后前者又细分为数据导入、模型训练和预测推断三块，后者又细分为差异性特征和共性特征两块。对每一细分的模块，为SEML设计一个页面。

b) 对平台将要使用到的机器学习算法逐一进行分析，理解算法的原理、进行数学推导并完成伪代码。这些算法包括SVM、RF、GBDT、XGBoost、Eclat等。

c) 利用Spark的R接口Sparklyr连接到Hive数据库，用代码实现数据导入模块。

d) 利用CARET集成R中的上述算法库，代码实现模型训练以及预测推断模块，包括自动化调参、交叉验证、模型评估可视化、测试数据的导入及预测等功能。

e) 利用Eclat算法，用代码实现定制化模块的功能。

f) 最后利用R的Web开发框架Shiny，打通前后台，真正开发出具有可视化界面的分布式机器学习平台。

以上所有设计和开发工作均由笔者一人独自完成。

工作总结

下一步计划

笔者认为SEML可以在以下几个方向作出改进：

a) 从数据的分布式扩展为算法的分布式。SEML的分布式性能体现在数据源的获取上（Hive之中），但进行机器学习算法建模时，依然是在单机上进行。故后期可以考虑通过Spark的R接口来使用MLlib进行算法的分布式训练。

b) 从点击式的操作到拖拽式的操作。微软AMLS、阿里PAI以及百度BML平台的亮点在于可以通过拖拽模块来"画出"DAG图，进而完成机器学习。为了更好的交互性，这将是SEML的发展方向。

c) 支持内嵌R和SQL的代码。SEML希望可以和微软AMLS一样，除了模块的拖拽与调用外，还支持用户在平台上书写R或SQL的代码进行数据挖掘，这个功能将会让它具备更好的灵活性。

Thank You !!!