**模块A：大数据(30分)**

**竞赛题目：游戏运营数据分析**

**背景简介**

游戏数据分析是对玩家行为的数字化理解。通过各式各样的数据，可以大致了解玩家在游戏中发生的情况，并通过思考推理，将数据写成分析报告，将发现的问题和多种解决方案一一罗列，以此为修改产品的重要依据。

游戏运营数据分析主要包括两方面：基础数据分析和整体优化分析。基础数据分析主要是对采集到的数据进行直观分析，如用户数量分析、用户活跃度分析、付费情况分析等，从而分别在具体环节，比如推广渠道上提出合理化的建议；整体优化分析是以基础数据为依托，建立优化模型，对游戏的投入到产出整体流程提出调整建议，从而优化游戏开发到市场推广的整个过程。

在下面的分析部分，提供了某款手游相关的一些数据，参赛选手需通过大数据的分析方式按要求完成一些常用的游戏数据分析指标计算，实现游戏数据的简单分析。

**环境介绍**

**远程环境：**

每位选手共三台aws云服务器，内部包含示与竞赛任务相关的Mysql数据库,以及大数据集群组件相关安装包。

**三台服务器环境信息如下：**

***master节点***

* 内网IP地址：用于内部大数据访问配置
* 操作系统： Red Hat 7.3.1-6
  + 用户名： ec2-user
* Hadoop 安装包目录~/hadoop搭建相关安装包
  + hadoop-2.9.2
  + sqoop-1.4.7
  + apache-hive-3.1.2
  + apache-flume-1.9.0

***node1节点***

* 内网IP地址：用于内部大数据访问配置
* 操作系统： Red Hat 7.3.1-6
  + 用户名： ec2-user
* Mysql 数据库
  + 用户名: root
  + 密码：123456
  + 数据库名：bric\_competition
  + 连接 Mysql 数据库命令：mysql -uroot -p123456

***node2节点***

* 内网IP地址：用于内部大数据访问配置
* 操作系统： Red Hat 7.3.1-6
  + 用户名： ec2-user
* Mysql 数据库（hive数据仓库的元数据库）
  + 用户名: root
  + 密码：123456

**本地环境：**

WINDOWS 环境（带 chrome 或 firefox 浏览器）

**注意事项**

* 部分问题需要手动保存答案，注意按题目要求的格式保存
* 部分问题已给出解题步骤，可按给出步骤执行
* Hadoop 集群相关配置文件已配置完成，不要格式化，可根据任务要求启动
* Sqoop 和 Hive 大数据组件已配置完成，可直接应用，不需要选手再次安装配置
* 遇到linux权限相关问题时，可在命令前加sudo
* 比赛完成后**禁止**点击**退出模块**按钮，否则后果自负

**数据介绍**

在 Hive 数据仓库中已经存在一个名为 deflaut 的库，该库下面包含多个表，其中game\_data（游戏数据）是后面竞赛所需要的数据表。

game\_data（游戏数据）：

| **字段** | **数据类型** | **字段说明** |
| --- | --- | --- |
| user\_id | bigint | 用户编码，用户唯一标识 |
| register\_time | string | 注册日期 |
| bd\_stronghold\_level | string | 要塞等级，相当于游戏账号等级 |
| wood\_reduce\_value | bigint | 木头消耗数量 |
| stone\_reduce\_value | double | 石头消耗数量 |
| ivory\_reduce\_value | double | 象牙消耗数量 |
| meat\_reduce\_value | double | 肉消耗数量 |
| magic\_reduce\_value | double | 魔法消耗数量 |
| general\_acceleration\_reduce\_value | bigint | 通用加速消耗数量 |
| building\_acceleration\_reduce\_value | bigint | 建筑加速消耗数量 |
| reaserch\_acceleration\_reduce\_value | bigint | 科研加速消耗数量 |
| training\_acceleration\_reduce\_value | bigint | 训练加速消耗数量 |
| treatment\_acceleration\_reduce\_value | bigint | 治疗加速消耗数量 |
| pvp\_battle\_count | bigint | 玩家对玩家次数 |
| pve\_battle\_count | bigint | 玩家对机器次数 |
| avg\_online\_minutes | double | 日均在线时间 |
| pay\_price | double | 消费金额 |
| pay\_count | bigint | 消费次数 |

**任务列表**

1. Hadoop 大数据集群启动
2. HDFS 文件操作
3. Sqoop 相关操作
4. Hive 数据分析
5. Spark 搭建与应用

**任务1：Hadoop 大数据集群启动（6分）**

1.1 配置hosts文件

1.2 修改 Hadoop 配置文件

1.3 启动 Hadoop 集群

**1.1 配置hosts文件**

分别获取三个节点中的内网IP地址与对应节点名称（可用ifconfig命令查询IP地址），并把三个IP地址以及对应节点名称一起分别写入到三个节点的/etc/hosts文件当中。（2分）

所示格式如下（如有多余master、node1、node2的配置，删除或者注释掉，否者会导致正确配置不起效）：

xxx.xx.xx.xx master  
xxx.xx.xx.xx node1  
xxx.xx.xx.xx node2

配置hosts文件提供以下两种方式，二选其一：

1. 直接修改/etc/hosts文件。
2. 执行bash ~/hosts\_config.sh命令，按照命令提示操作。

**1.2 修改 Hadoop 配置文件**

在每个节点上操作，修改Hadoop相关配置文件，增加限制HDFS块大小的配置（dfs.blocksize），原有的默认值为128MB，现限制块大小为32MB（32MB=33554432B），保存修改完成后的配置文件。（2分）

**1.3 启动Hadoop集群**

启动Hadoop集群，包含master节点上的NameNode、SecondaryNameNode、ResourceManager，以及node1、node2节点上的NodeManager、DataNode。（2分）

**任务2：HDFS文件操作（6分）**

2.1 HDFS 上新建目录

2.2 修改HDFS目录权限

2.3 上传本地文件到HDFS上

**2.1 HDFS上新建目录**

在master节点上操作，在HDFS上创建一个名为/task2的目录。（2分）

**2.2 修改HDFS目录权限**

在master节点上操作,HDFS上设置/task2目录权限为drwxrwxrwx。（2分）

**2.3 上传本地文件到HDFS上**

在master节点上操作,上传本地~/data/task2\_3.csv文件到HDFS的/task2/目录下。（2分）

**任务3：Sqoop 相关操作（6分）**

3.1 Sqoop连接Mysql并查询

3.2 Sqoop导入

3.3 Sqoop导出

**3.1 Sqoop 连接 Mysql并查询**

在master节点上操作,运用sqoop，查询node1节点上bric\_competition数据库中的task3\_1表，查询表task3\_1总记录数量，并保存结果到~/results/task3-1.txt文件中。（2分）

**3.2 Sqoop 导入**

在master节点上操作，使用Sqoop将node1节点上Mysql里bric\_competition数据库中task3\_2数据表导入到hive数据仓库中的default数据库里。（2分）

**3.3 Sqoop导出**

在master节点上操作，使用Sqoop将HDFS上的/sqoop/task3\_3文件，导入到node1节点Mysql里bric\_competition数据库中的task3\_3表里。（2分）

**任务4：Hive 数据分析（7分）**

4.1 计算用户注册数量

4.2 计算**ARPU**

4.3 计算用户付费转化率

**4.1 计算用户注册数量**

**注册用户量**指的是某款截止到某一时刻所拥有的已注册的用户总量。通常所说的用户量，如果没有特别说明，一般指的就是注册用户量。

探查 Hive中的game\_data表中数据，查看用户注册数量随时间的分布变化，查询出哪天注册的用户数量最多？并记录register\_time(注册日期)的查询结果到 master 节点上 ec2-user 用户下的~/results/task4-1.txt文件中。（2分）

**4.2 计算ARPU**

**ARPU**就是每用户平均收入。ARPU注重的是一个时间段内运营商从每个用户所得到的利润。很明显，高端的用户越多，ARPU越高。在这个时间段，从运营商的运营情况来看，ARPU值高说明利润高，这段时间效益好。**（ARPU=总付费金额/活跃人数）（活跃用户：日均在线时间达到30分钟的用户）**

探查 Hive 中的game\_data表中数据，计算出该游戏产品的ARPU值，保留两位小数，并记录结果到 master 节点上 ec2-user 用户下的~/results/task4-2.txt文件中。（2分）

**4.3 计算用户付费转化率**

**付费转化率**就是在一个游戏中，付费玩家占整个玩家的比例。在某种程度上，说明了游戏本身付费玩家转化能力、付费点、经济系统是否为玩家所接受，是游戏收益能力的一个有效指标**（付费转化率=付费用户/总用户数×100%）**

探查 Hive 中的game\_data表中数据，查询出达到哪个等级（bd\_stronghold\_level）时，用户付费转化率实现100%？并记录bd\_stronghold\_level(游戏账号等级)结果到 master 节点上 ec2-user用户下的~/results/task4-3.txt文件中。（3分）

**任务5：Spark 搭建与应用（5分）**

5.1 Spark 组件安装

5.2 Spark 连接 Hive

**5.1 Spark 组件安装**

在master节点上操作，安装 Spark 组件,并启动 Spark，包含master节点上的Master，以及node1、node2节点上的Worker。Spark 安装包所在路径：~/hadoop搭建相关安装包/spark-2.4.6-bin-hadoop2.7.tgz（3分）

**5.2 Spark 连接 Hive**

在master节点上操作，通过配置 spark 相关配置文件，使 spark-sql 连接到 Hive 读取数据。（2分）

**任务结束**

恭喜完成本次比赛，祝您取得好成绩。