实验四–天猫复购预测

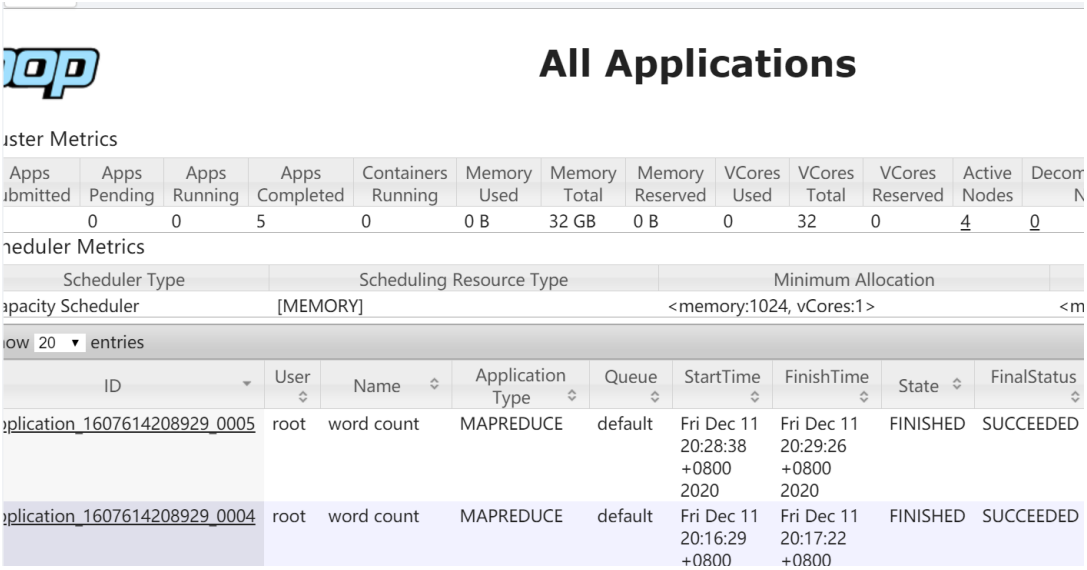
运行环境：BDKIT+本地anaconda&pyspark
结果输出：在每一个小题项目中的output文件夹中，以txt/csv形式呈现。

一. 热销商品&热门商家

分别编写MapReduce程序和Spark程序统计双十一最热门的商品和最受年轻人(age<30)关注的商家 （“添加购物车+购买+添加收藏夹”前100名）；

1）MapReduce统计双十一最热门的商品、双十一最受年轻人关注的商家

与wordcount词频统计类似，在map环节判断购买时间，筛选出**双十一当天（time_stamp=“1111”）**选购的产品，再进行“添加购物车+购买+添加收藏夹”的判断操作，若是有最受年轻人关注的限制条件，则将user_info中的信息处理为数组，用类似于停词的方法在user_log表中筛选出年龄小于30岁的用户，再进行统计，通过context.write写入，reduce过程中排序输出前100名。



2）spark统计双十一最热门的商品（语言：scala）

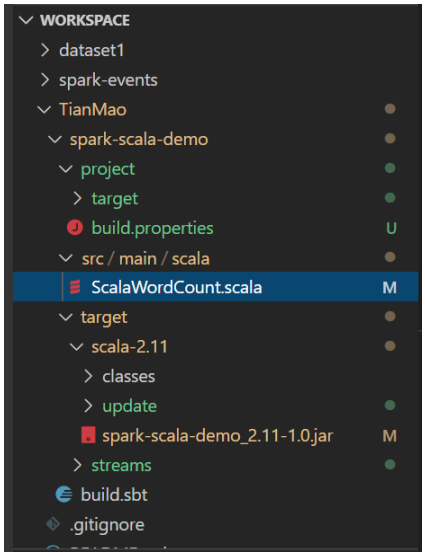
根据题目要求，在筛选阶段共进行两项任务，①判断_（5）是否为1111，即time_stamp是否位于双十一当天；②判断_（6）是否为1、2、3，即是否为添加购物车、购买或者添加收藏夹操作。利用filter函数进行筛选。

```
val rdd1 = input.filter(x => x.split(",")(5).equals("1111"))
val rdd2 = rdd1.filter(x => x.split(",")(6).equals("0")==false)
```

筛选完成后，进行map、reduceByKey以及sortByKey操作，只取排序过后的前100将结果保存为文件，在指定目录下输出。

```scala
val tmp = rdd2.map(x=>(x.split(",")(1),1)).reduceByKey(_+_)
val result = tmp.map(x=>(x._2,x._1)).sortByKey(false).map(x=>(x._2,x._1)).take(100)
val out = sc.saveAsTextFile(args(1))
sc.stop()
```

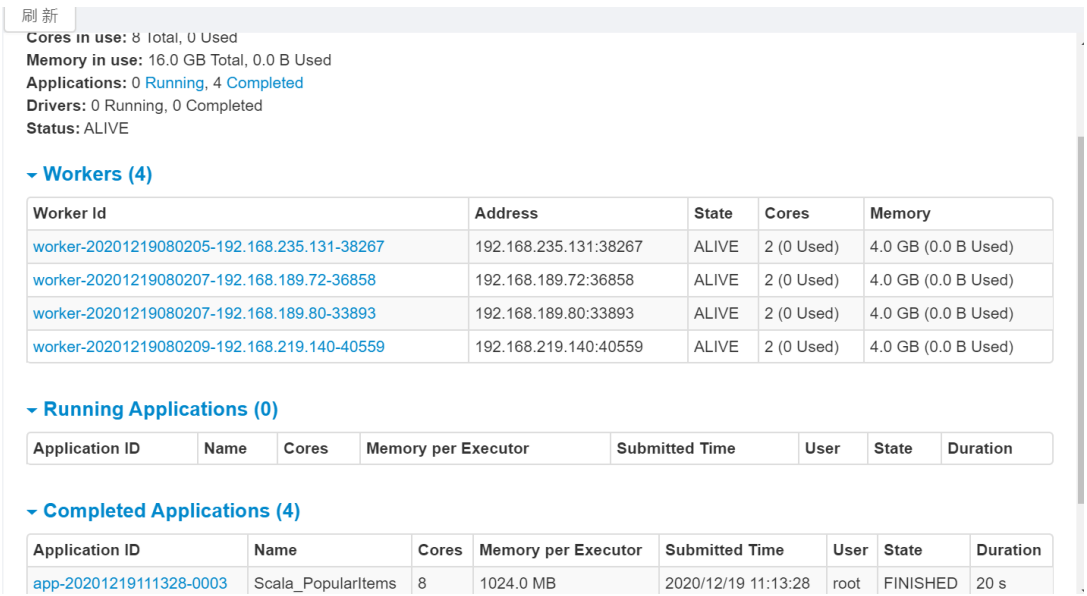
建立项目：



sbt打包：

```
root@lyyq181850099-master:/workspace/TianMao/spark-scala-demo# sbt clean
[info] Loading project definition from /workspace/TianMao/spark-scala-demo/project
[info] Loading settings for project spark-scala-demo from build.sbt ...
[info] Set current project to Spark Scala Demo (in build file:/workspace/TianMao/spark-scala-demo/)
[success] Total time: 0 s, completed Dec 19, 2020 10:17:13 AM
root@lyyq181850099-master:/workspace/TianMao/spark-scala-demo# sbt package
[info] Loading project definition from /workspace/TianMao/spark-scala-demo/project
[info] Loading settings for project spark-scala-demo from build.sbt ...
[info] Set current project to Spark Scala Demo (in build file:/workspace/TianMao/spark-scala-demo/)
[warn] There may be incompatibilities among your library dependencies; run 'evicted' to see detailed eviction warnings.
[info] Compiling 1 Scala source to /workspace/TianMao/spark-scala-demo/target/scala-2.11/classes ...
[success] Total time: 7 s, completed Dec 19, 2020 10:17:32 AM
```

运行得出结果，并查看监控页面的工作区状态：



2）spark统计双十一最受年轻人欢迎的商家（语言：scala）

根据题目要求，首先对于两张表分别进行处理。user-log信息表在筛选阶段共进行两项任务，①判断_（5）是否为1111，即time_stamp是否位于双十一当天；②判断_（6）是否为1、2、3，即是否为添加购物车、购买或者添加收藏夹操作。对于user-info信息表，筛选出具有完整年龄、性别信息的，然后根据题中要求filter处理得到子表进行后续操作。

最初的想法是将两张已经进行过数据处理的表通过join函数合并到一起，然后进行map和reduceBykey操作，如下所示：

```
val infoProcessed = userInfo.filter(x=>(x.split(",").length==3))
val young = infoProcessed.filter(x=>(x.split(",")(1).equals("1") || x.split(",")(1).equals("2") || x.split(",")(1
val fitusers = young.map(x=>(x.split(",")(0),1))//(id,1)


val rdd1 = userLog.filter(x=>x.split(",")(5).equals("1111"))
val rdd2 = rdd1.filter(x=>x.split(",")(6).equals("0")==false)
val users = rdd2.map(x=>(x.split(",")(0),x.split(",")(3)))//(id,merchant)

val rdd = users.join(fitusers)
```

然而经过实践后发现，由于join对于map的keyvalue对的要求，合并之后的表如果直接按照思路进行map和reduceBykey操作，会出现下面的情况(仅展示前六行)：

((4044,1),7248)
((3491,1),3634)
((1102,1),3565)
((3828,1),3416)
((4173,1),3333)
((3734,1),3277)

经过查询之后找到了和join函数功能互补的函数，在对于user-info处理的过程中把不符合要求的数据筛选出来，然后通过函数在user-log的表中删去这部分数据，再进行符合条件的merchant计数，不会造成key-value对混乱，完成题目要求。

刷新

Workers (4)

Worker Id	Address	State	Cores	Memory
worker-20201221004458-192.168.219.162-37695	192.168.219.162:37695	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221004458-192.168.219.165-35921	192.168.219.165:35921	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221004458-192.168.219.166-41131	192.168.219.166:41131	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221004458-192.168.219.169-34418	192.168.219.169:34418	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)

Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration

Completed Applications (4)

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration
app-20201221022947-0003	Scala_PopularStore	8	1024.0 MB	2020/12/21 02:29:47	root	FINISHED	2.5 min

二. 性别&年龄比例统计

编写Spark程序统计双十一购买了商品的男女比例，以及购买了商品的买家年龄段的比例
统计过程中，通过filter函数将info表中**不完整的数据的删去**（没有年龄或性别信息）。

1）统计双十一购买了商品的男女比例

此处和第一问思路并不相同，在统计最受欢迎的商家和商品的过程中，需要统计的是购买/收藏/添加购物车的操作次数，但是在统计男女比例的时候，需要考虑**一人购买多次**的情况，否则就会重复计算多次购买的人数，所以在处理数据的过程中，不能按照简单的思路直接在log表中处理，可以考虑在info表中进行统计人数，不会出现重复。此处注意：不计算gender一栏为2或者null的购买者。

```
root@lyyq181850099-master:/workspace/TianMao/spark-scala-demo# hdfs dfs -cat /spark/outputs/*
Java HotSpot(TM) 64-Bit Server VM warning: You have loaded library /usr/local/hadoop/lib/native/libhadoop.so which might
have disabled stack guard. The VM will try to fix the stack guard now.
It's highly recommended that you fix the library with 'execstack -c <libfile>', or link it with '-z noexecstack'.
20/12/22 05:43:29 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-jav
a classes where applicable
(285638,0)
(121670,1)
```

Applications: 0 Running, 7 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (4)

Worker Id	Address	State	Cores	Memory
worker-20201221115038-192.168.189.109-40390	192.168.189.109:40390	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221115038-192.168.189.86-42428	192.168.189.86:42428	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221115039-192.168.235.130-46175	192.168.235.130:46175	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221115040-192.168.235.133-35539	192.168.235.133:35539	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)

Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration

Completed Applications (7)

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration
app-20201221173542-0006	Scala_MTFRatio	8	1024.0 MB	2020/12/21 17:35:42	root	FINISHED	29 s

结果：

	numbers	proportion
female	285638	0.701282567
male	121670	0.298717433

2）统计购买了商品的买家年龄段的比例

与第一问思路几乎相同，此处要注意的是>=50岁年龄段的有两个参数7和8表示，统计结果计算比例的时候注意数据合并。

刷新

Applications: 0 Running, 13 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (4)

Worker Id	Address	State	Cores	Memory
worker-20201221115038-192.168.189.109-40390	192.168.189.109:40390	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221115038-192.168.189.86-42428	192.168.189.86:42428	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221115039-192.168.235.130-46175	192.168.235.130:46175	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)
worker-20201221115040-192.168.235.133-35539	192.168.235.133:35539	ALIVE	2 (0 Used)	4.0 GB (0.0 B Used)

Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration

Completed Applications (13)

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration
app-20201222055920-0012	Scala_AgeRatio	8	1024.0 MB	2020/12/22 05:59:20	root	FINISHED	34 s

结果：

	numbers	proportion
<18	24	0.000073

[18,24]	52420	0.160272
[25,29]	110952	0.339230
[30,34]	79649	0.243523
[35,39]	40601	0.124136
[40,49]	35257	0.107796
> = 50	8167	0.024970

三. 性别&年龄比例查询

基于Hive查询双十一购买了商品的男女比例，以及购买了商品的买家年龄段的比例。

进入Hive。

```
Hive Session ID = 49ff0d07-ca6a-4bba-911e-16a7a55f2c08
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark,
ez) or using Hive 1.X releases.
hive>
```

建立user_log_format1信息的table。

```
hive> create table userlog(
    > user_id int,
    > item_id int,
    > cat_id int,
    > merchant_id int,
    > brand_id int,
    > time_stamp int,
    > action_type int)
    > row format delimited
    > fields terminated by ',';
OK
Time taken: 1.393 seconds
```

导入本地user_log_format1.csv。

```
hive> load data local inpath
    > '/workspace/TianMaoPrediction/data_format1/user_log_format1.csv'
    > into table userlog;
Loading data to table default.userlog
OK
Time taken: 6.467 seconds
```

同理建立user_info_format1信息的table。

```
hive> create table userinfo(
    > user_id int,
    > age_range int,
    > gender int)
    > row format delimited
    > fields terminated by ',';
OK
Time taken: 0.059 seconds
```

导入本地user_info_format1.csv。

```
hive> load data local inpath
    > '/workspace/TianMaoPrediction/data_format1/user_info_format1.csv'
    > into table userinfo;
Loading data to table default.userinfo
OK
Time taken: 0.168 seconds
```

筛选time_stamp为1111即双十一当天进行操作的users。

```
hive> insert overwrite table userlog
    > select * from userlog where time_stamp=1111;
2020-12-23 14:22:45,287 ERROR [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] hdfs.KeyProviderCache: Could not find uri with key [dfs.encryption.k
ey.provider.uri] to create a keyProvider !!
Query ID = root_20201223142243_0a09121c-aeae-461e-91db-620e3b95c4e2
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
2020-12-23 14:22:45,841 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] Configuration.deprecation: mapred.submit.replication is deprecated.
Instead, use mapreduce.client.submit.file.replication
2020-12-23 14:22:46,005 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
2020-12-23 14:22:46,605 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
```

筛选action_type为2即进行了购买操作的users。

```
hive> create table fitusers as
    > select userlog.user_id from userlog
    > where action_type=2;
Query ID = root_20201223142501_b376780d-7f34-4a1a-a652-bf8846b8862d
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
2020-12-23 14:25:02,144 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
2020-12-23 14:25:02,178 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
```

对于userinfo表进行操作，首先去除没有年龄或者性别信息的数据。

```
hive> insert overwrite table userinfo
    > select * from userinfo where
    > (userinfo.gender is not null and userinfo.age_range is not null);
Query ID = root_20201223143408_c4ec1fd1-f189-4372-ad86-da357ef87365
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
2020-12-23 14:34:08,556 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
2020-12-23 14:34:08,594 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
```

建立table并使用intersect函数将info筛选过后的信息与fitsuers合并，利用count函数直接统计人数信息，统计年龄比例同理，创建表格以及筛选fitusers过程中重复的部分截图略去。

```
hive> create table femaleusers as
    > select userinfo.user_id from userinfo
    > where gender=0;
Query ID = root_20201223150545_688d7f6f-c57c-4758-ae09-f0464b2fab7e
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
2020-12-23 15:05:45,645 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
2020-12-23 15:05:45,686 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
```


```
hive> create table femaleusers as
    > select userinfo.user_id from userinfo
    > where gender=0;
Query ID = root_20201223150545_688d7f6f-c57c-4758-ae09-f0464b2fab7e
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
2020-12-23 15:05:45,645 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
2020-12-23 15:05:45,686 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
```


```
hive> insert overwrite table femaleusers
    > select user_id from femaleusers
    > intersect select user_id from fitusers;
Query ID = root_20201223150738_4b4511ad-4d7b-4fb5-8f3a-ba0d7f8c708b
Total jobs = 4
Launching Job 1 out of 4
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
2020-12-23 15:07:38,549 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
2020-12-23 15:07:38,574 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] client.RMProxy: Connecting to ResourceManager at lyyq181850099-maste
r/192.168.219.70:8032
```

```
2020-12-23 15:24:46,182 Stage-4 map = 0%,  reduce = 0%
2020-12-23 15:24:50,349 Stage-4 map = 100%,  reduce = 0%, Cumulative CPU 1.63 sec
2020-12-23 15:24:55,530 Stage-4 map = 100%,  reduce = 100%, Cumulative CPU 4.23 sec
MapReduce Total cumulative CPU time: 4 seconds 230 msec
Ended Job = job_1608732006728_0019
2020-12-23 15:24:57,695 INFO  [4aef1b57-5b23-45d3-afbf-ee8efeaa8bb7 main] mapred.FileInputFormat: Total input paths to process : 1
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 9.17 sec   HDFS Read: 831476 HDFS Write: 2684143 SUCCESS
Stage-Stage-5: Map: 1  Reduce: 1   Cumulative CPU: 66.92 sec   HDFS Read: 8254713 HDFS Write: 9358379 SUCCESS
Stage-Stage-2: Map: 2  Reduce: 1   Cumulative CPU: 36.61 sec   HDFS Read: 12055466 HDFS Write: 820630 SUCCESS
Stage-Stage-4: Map: 1  Reduce: 1   Cumulative CPU: 4.23 sec   HDFS Read: 9387 HDFS Write: 989 SUCCESS
Total MapReduce CPU Time Spent: 1 minutes 56 seconds 930 msec
OK
Time taken: 86.029 seconds
hive> select count(user_id) from maleusers;
OK
121655
Time taken: 0.087 seconds, Fetched: 1 row(s)
```

```
2020-12-23 15:45:59,300 Stage-4 map = 0%,  reduce = 0%
2020-12-23 15:46:04,502 Stage-4 map = 100%,  reduce = 0%, Cumulative CPU 1.57 sec
2020-12-23 15:46:09,694 Stage-4 map = 100%,  reduce = 100%, Cumulative CPU 4.0 sec
MapReduce Total cumulative CPU time: 4 seconds 0 msec
Ended Job = job_1608732006728_0024
2020-12-23 15:46:11,177 INFO  [eb950b7e-f36b-4960-89e0-f412ba5454f1 main] Configuration.deprecation: mapred.input.dir is deprecated. Instead,
use mapreduce.input.fileinputformat.inputdir
2020-12-23 15:46:11,182 INFO  [eb950b7e-f36b-4960-89e0-f412ba5454f1 main] mapred.FileInputFormat: Total input paths to process : 1
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 4.13 sec   HDFS Read: 11837 HDFS Write: 615 SUCCESS
Stage-Stage-5: Map: 1  Reduce: 1   Cumulative CPU: 71.06 sec   HDFS Read: 8254740 HDFS Write: 9358379 SUCCESS
Stage-Stage-2: Map: 2  Reduce: 1   Cumulative CPU: 26.19 sec   HDFS Read: 9371891 HDFS Write: 463 SUCCESS
Stage-Stage-4: Map: 1  Reduce: 1   Cumulative CPU: 4.0 sec   HDFS Read: 8843 HDFS Write: 264 SUCCESS
Total MapReduce CPU Time Spent: 1 minutes 45 seconds 380 msec
OK
Time taken: 87.451 seconds
hive> select count(user_id) from young;
OK
24
```

男女比例：

	numbers	proportion
female	285638	0.701282567
male	121670	0.298717433

年龄比例：

	numbers	proportion
<18	24	0.000073
[18,24]	52420	0.160272
[25,29]	110952	0.339230
[30,34]	79649	0.243523
[35,39]	40601	0.124136
[40,49]	35257	0.107796
> = 50	8167	0.024970

四. 消费预测

预测给定的商家中，哪些新消费者在未来会成为忠实客户，即需要预测这些新消费者在6个月内再次购买的概率。基于Spark MLlib编写程序预测回头客，评估实验结果的准确率。

运行环境：本地anaconda+pyspark
（bdkit.info回收之后bdkit.cn就没有办法进行使用，分别试验过凌晨、上午、下午、晚上，都处于无法package、无法submit的状态，所以第四题是在本地进行运行的。）

此处消费预测分为两步：

- 根据用户和商家信息，从user_log和user_info中提取特征值。

- 训练评估模型，预测会重复购买的用户。

其中，利用spark的join、groupby等函数，对于用户信息表和用户行为表进行操作，基于user_id和merchant_id纳入考虑的特征值包括：用户的年龄(age_range)、用户的性别(gender)、某用户在该商家日志的总条数(total_logs)、用户浏览的商品的数目，就是浏览了多少个商品(unique_item_ids)、浏览的商品的种类的数目即浏览了多少种商品(categories)、用户浏览的天数(browse_days)、用户单击的次数(one_clicks)、用户添加购物车的次数(shopping_carts)、用户购买的次数(purchase_times)、用户收藏的次数(favourite_times)。

本地安装spark：

```
C:\Users\baish>java --version
openjdk 14.0.1 2020-04-14
OpenJDK Runtime Environment (build 14.0.1+7)
OpenJDK 64-Bit Server VM (build 14.0.1+7, mixed mode, sharing)

C:\Users\baish>scala
Welcome to Scala 2.13.4 (OpenJDK 64-Bit Server VM, Java 14.0.1).
Type in expressions for evaluation. Or try :help.

scala>
```

```
Spark context Web UI available at http://172.27.133.53:4040
Spark context available as 'sc' (master = local[*], app id = local-1608950627243).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.4.7
      /_/

Using Scala version 2.11.12 (OpenJDK 64-Bit Server VM, Java 14.0.1)
Type in expressions to have them evaluated.
Type :help for more information.
```

在anaconda中install pyspark，然后运行代码。

读入文件（测试）：

```
(base) D:\pythonwork\SparkMLlib>python main.py
   user_id  item_id  cat_id  seller_id  brand_id  time_stamp  action_type
0   328862   323294     833       2882    2661.0         829            0
1   328862   844400    1271       2882    2661.0         829            0
2   328862   575153    1271       2882    2661.0         829            0
3   328862   996875    1271       2882    2661.0         829            0
4   328862  1086186    1271       1253    1049.0         829            0
```

提取特征值，以某用户在该商家日志的总条数为例，首先通过user_id、seller_id进行groupby操作，处理完毕后与读入的训练集train合并。其他特征值操作类似。

```
    total_logs_temp = userLog.groupby("user_id", "seller_id").count()
    ##print(total_logs_temp.head())
    total_logs_temp = total_logs_temp.withColumnRenamed("seller_id", "merchant_id")
    total_logs_temp = total_logs_temp.withColumnRenamed("count", "total_logs")
    train = train.join(total_logs_temp, on=["user_id", "merchant_id"], how="left")
```

特征值提取完毕后，得到train-feature.csv，输出检查，然后对于测试集进行同样的操作，得到test-feature.csv，输出后以便进行下一步训练模型、预测并评估结果。

得到特征值后，在预测用户是否会有回购行为中，需要解决的是一个分类问题 。读入train-feature.csv与test-feature.csv分别作为训练模型和预测模型的数据，选用随机森林进行模型训练和预测。种类特征指标，在对于模型进行改进的过程中，为正数分配更高的权重(cancelled == 1)，生成类权重以平衡数据。

```
#assign higher weights to the positives(cancelled == 1).
balancingRatio = float(data.select("label").where("label==0").count() / data. count())
calculateWeights = udf(lambda x: balancingRatio if x == 1 else (1.0-balancingRatio), FloatType())
dataSet = data.withColumn("classWeightCol", calculateWeights('label'))
```

首先，通过randomSplit函数对于train-feature.csv进行划分，其中设置不同的split比例和不同的numTrees来尝试，调用MulticlassClassificationEvaluator评估模型并计算accuracy，评价模型准确率，结果分别如下：

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
20/12/27 19:00:47 WARN ProcfsMetricsGetter: Exception when trying to compute pagesize, as a result reporting of ProcessT
ree metrics is stopped
acc = 0.665707
rfModelSummary: RandomForestClassificationModel: uid=RandomForestClassifier_aa8fde40997a, numTrees=50, numClasses=2, num
Features=10
```

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
20/12/27 19:02:06 WARN ProcfsMetricsGetter: Exception when trying to compute pagesize, as a result reporting of ProcessT
ree metrics is stopped
acc = 0.660539
rfModelSummary: RandomForestClassificationModel: uid=RandomForestClassifier_366c90246eda, numTrees=75, numClasses=2, num
Features=10
```

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
20/12/27 18:57:43 WARN ProcfsMetricsGetter: Exception when trying to compute pagesize, as a result reporting of ProcessT
ree metrics is stopped
acc = 0.661015
rfModelSummary: RandomForestClassificationModel: uid=RandomForestClassifier_e2899c116921, numTrees=100, numClasses=2, nu
mFeatures=10
```

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
20/12/27 19:03:41 WARN ProcfsMetricsGetter: Exception when trying to compute pagesize, as a result reporting of ProcessT
ree metrics is stopped
acc = 0.663659
rfModelSummary: RandomForestClassificationModel: uid=RandomForestClassifier_f7a4ae984436, numTrees=75, numClasses=2, num
Features=10
```

最终使用numTrees=100的情况来对于测试集进行预测。

```
train = data
test = tests
rf = RandomForestClassifier(labelCol="indexedLabel", featuresCol="indexedFeatures", weightCol="classWeightCol", numTrees=100)
```

输出result.csv，得到最终结果。

类权重参考：
https://blog.csdn.net/weixin_26726011/article/details/108494780?ops_request_misc=%25257B%252522request%25255Fid%252522%25253A%25252216089987581678028826390 8&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_v2~rank_v29-22-108494780.pc_search_result_cache&utm_term=pyspark%20ml%20%E9%9A%8F%E6%9C%BA%E6%A3%AE%E6%9E%97