

天猫推荐算法竞赛

快速上手篇（HELLO WORLD）

宋爽（刘永凯）

竞赛简介

1 竞赛题目：

根据用户在前四个月对品牌的行为数据，预测用户在第五个月中可能会购买的品牌。



2 竞赛评估：


计算预测的（用户数-品牌数）对的准确率和召回率，最终得到算法成绩 F1-SCORE

$$\text{precision} = \frac{\sum_i^N \text{hitBrands}_i}{\sum_i^N \text{pBrands}_i} \quad \text{其中 } N \text{ 为参赛队预测的用户数, } \text{pBrands}_i \text{ 为对用户 } i \text{ 预测他/}$$

她会购买的品牌列表, hitBrands_i 为用户 i 真实购买的品牌列表中命中的品牌数。

$$\text{Recall} = \frac{\sum_i^M \text{hitBrands}_i}{\sum_i^M \text{bBrands}_i} \quad \text{其中 } M \text{ 为产生实际交易的品牌用户数量, } \text{bBrands}_i \text{ 为用户 } i \text{ 真}$$

实购买的品牌数, hitBrands_i 为用户 i 真实购买的品牌列表中命中的品牌数。


$$F_1 = \frac{2 * P * R}{P + R}$$

可运行示例（ HELLO WORLD ）



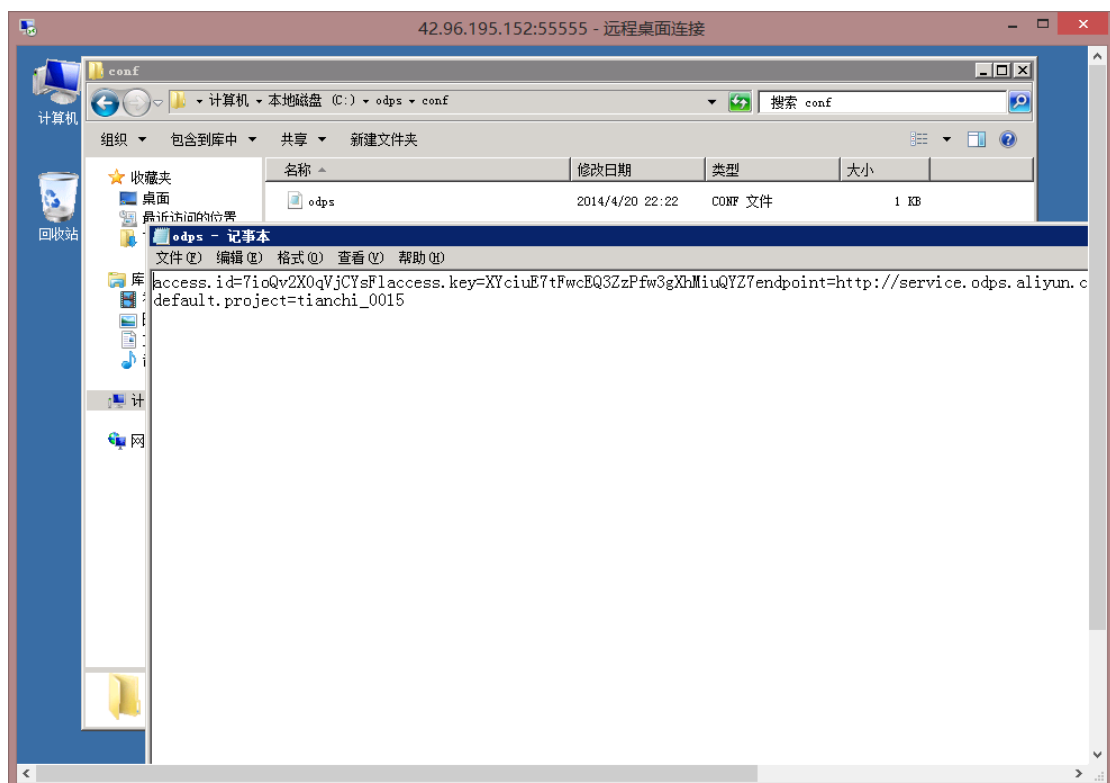
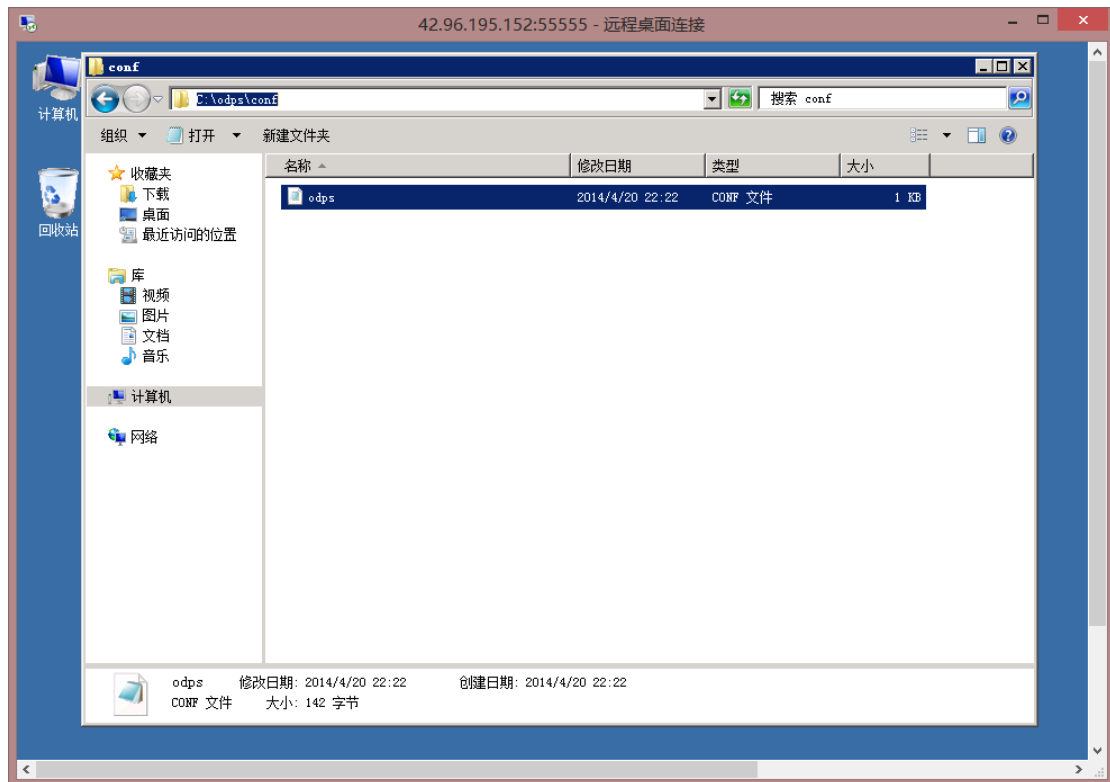
1 理解数据：

在天猫，每天都会有数千万的用户通过品牌发现自己喜欢的商品，品牌是联接消费者与商品最重要的纽带。本届赛题的任务就是根据用户在天猫的行为日志，建立用户的品牌偏好，并预测他们在将来对品牌下商品的购买行为。我们会开放如下数据类型：

| 字段 | 字段说明 | 提取说明 |
|-------------|------------|-------------------------------|
| user_id | 用户标记 | 抽样&字段加密 |
| Time | 行为时间 | 精度到小时级别 |
| action_type | 用户对品牌的行为类型 | 点击-0 购买-1 收藏-2 购物车-3 |
| brand id | 品牌ID | 抽样&字段加密 |

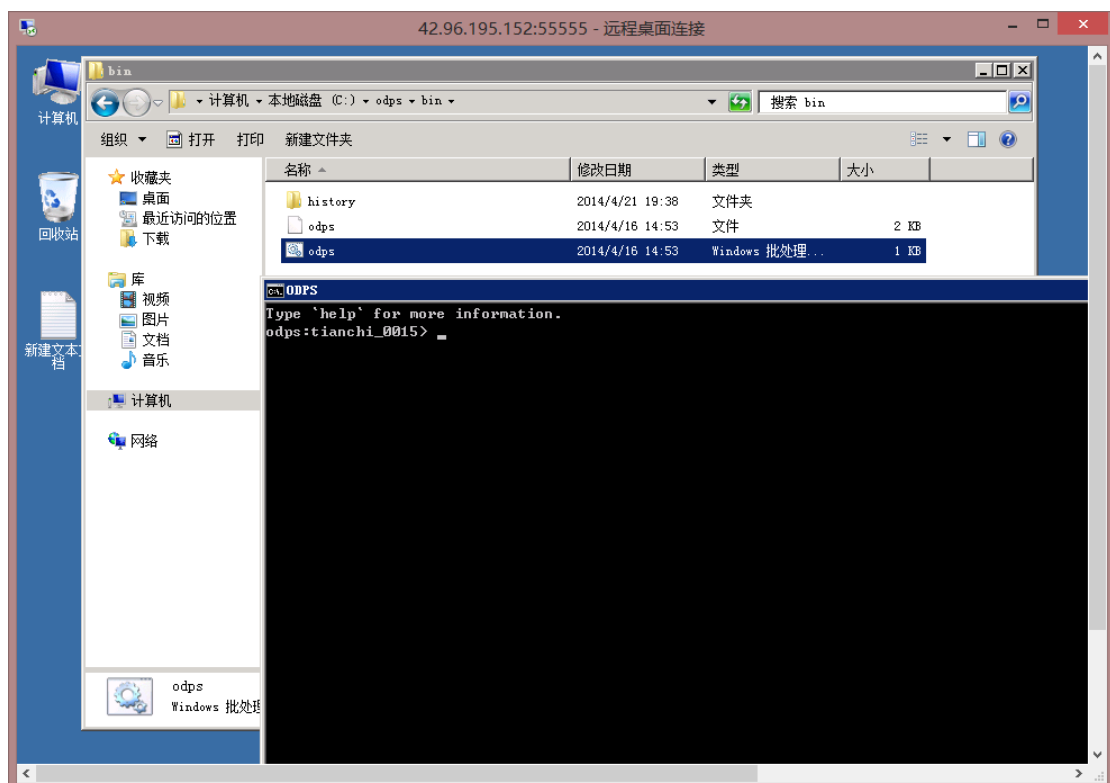
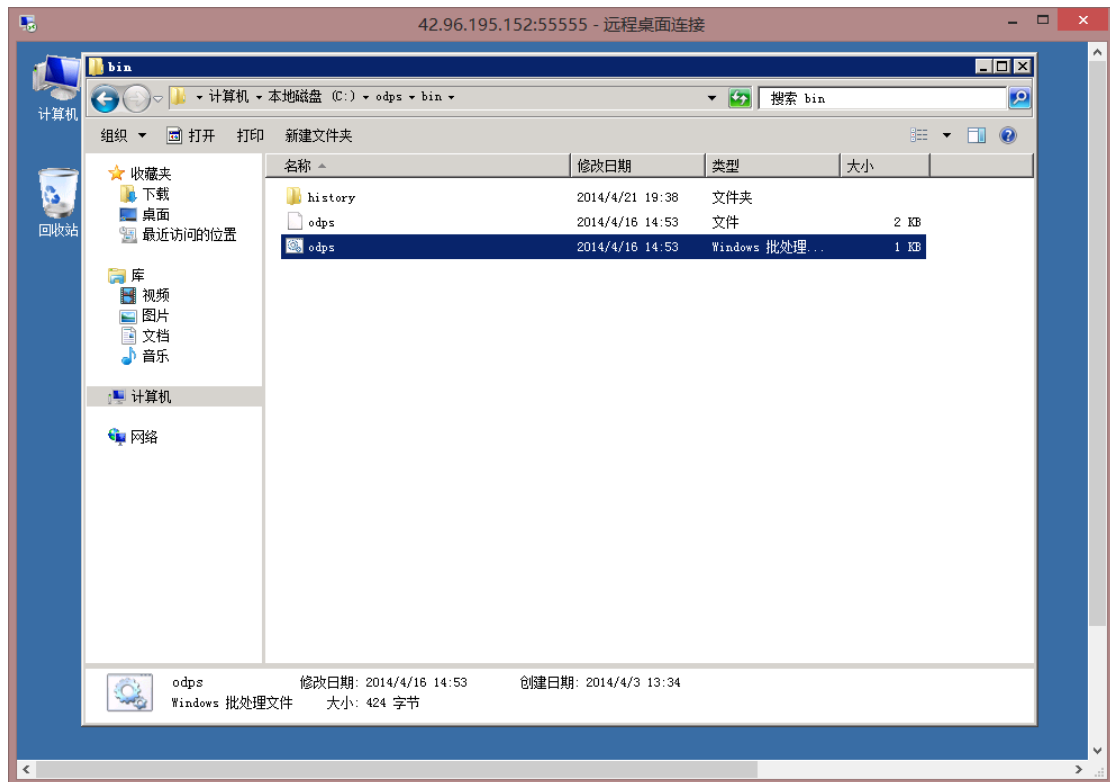
查看可用的数据，可按照以下的步骤：

首先，找到 odps 的目录，进入到 conf 目录，修改你的账号信息（注册之后会给每一个比赛团队分配一组 id 和 key），例如：

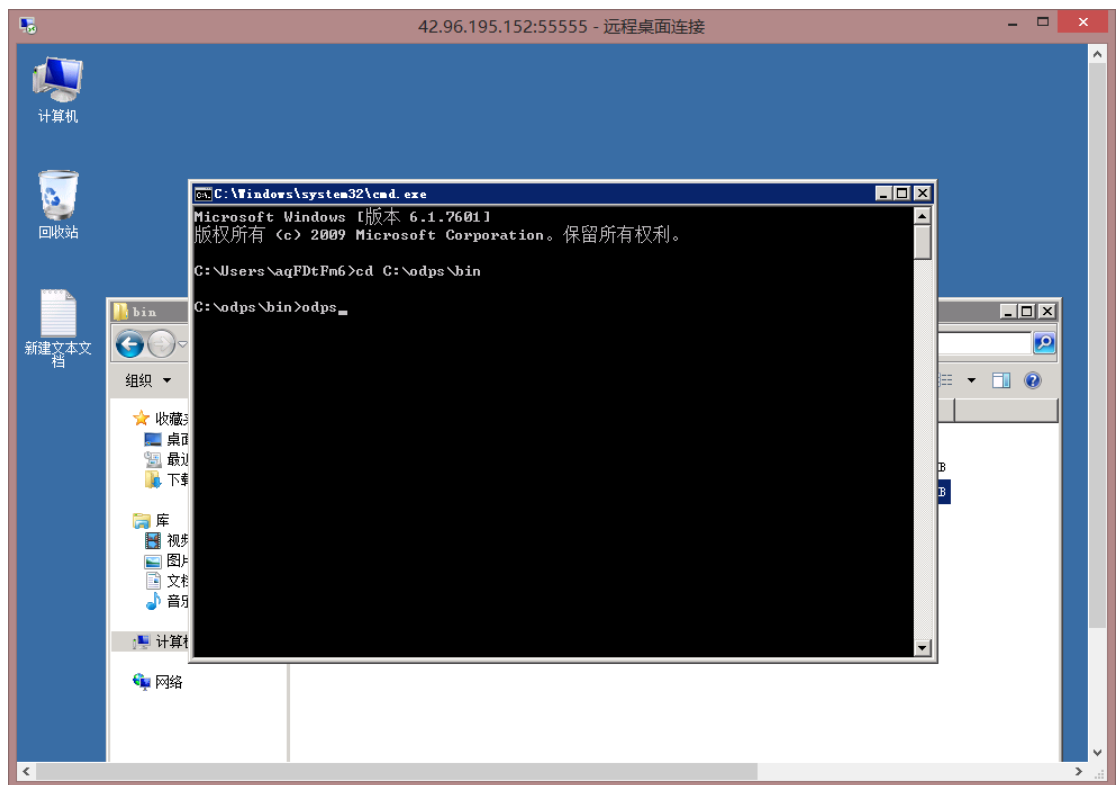
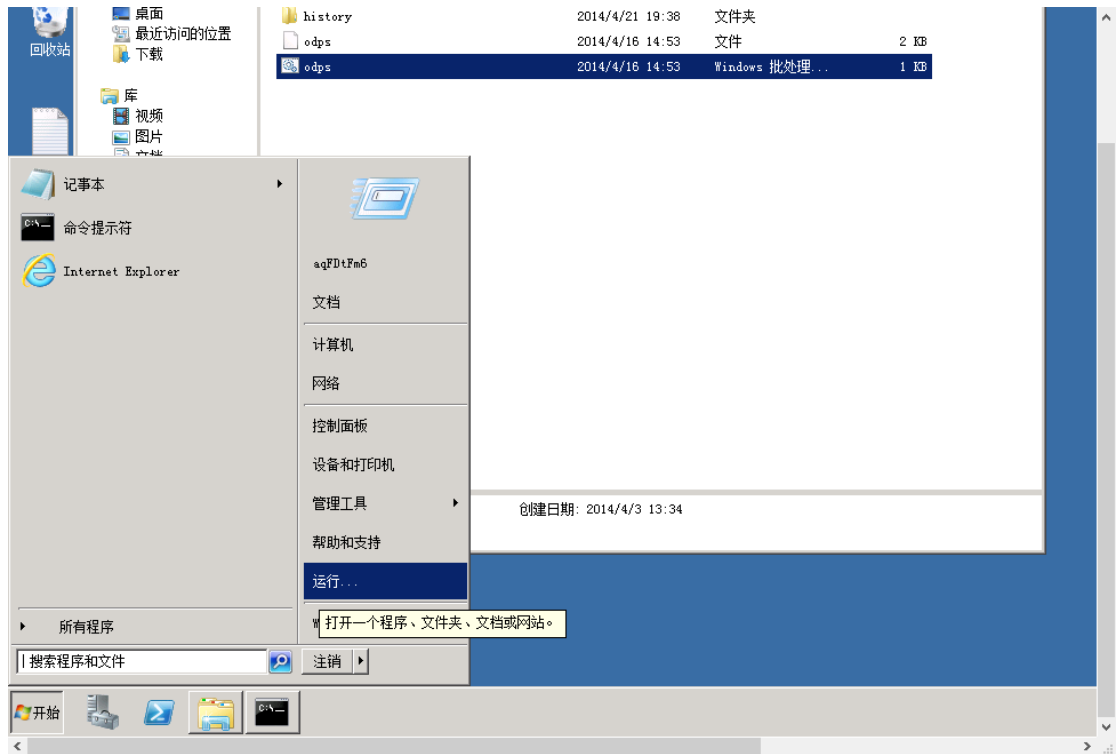


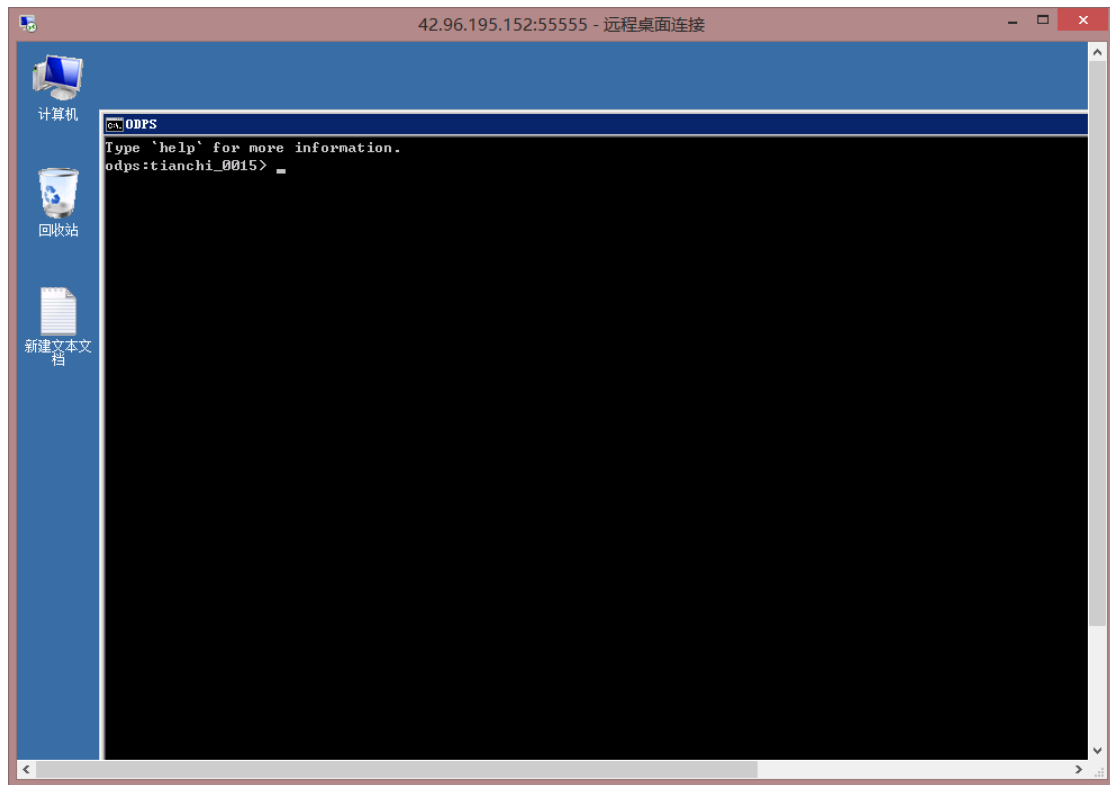
然后，启动 odps 命令行模式，有两种方式：

A 直接进入到了 odps 的 bin 目录，双击 odpscmd.bat：

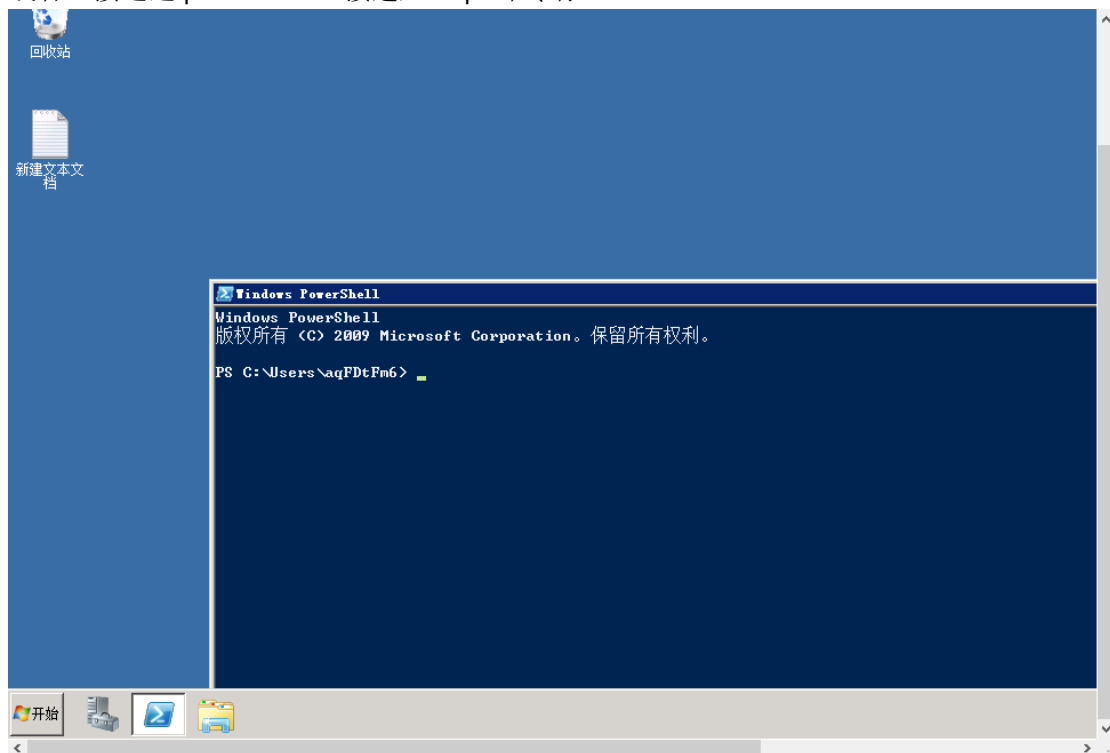


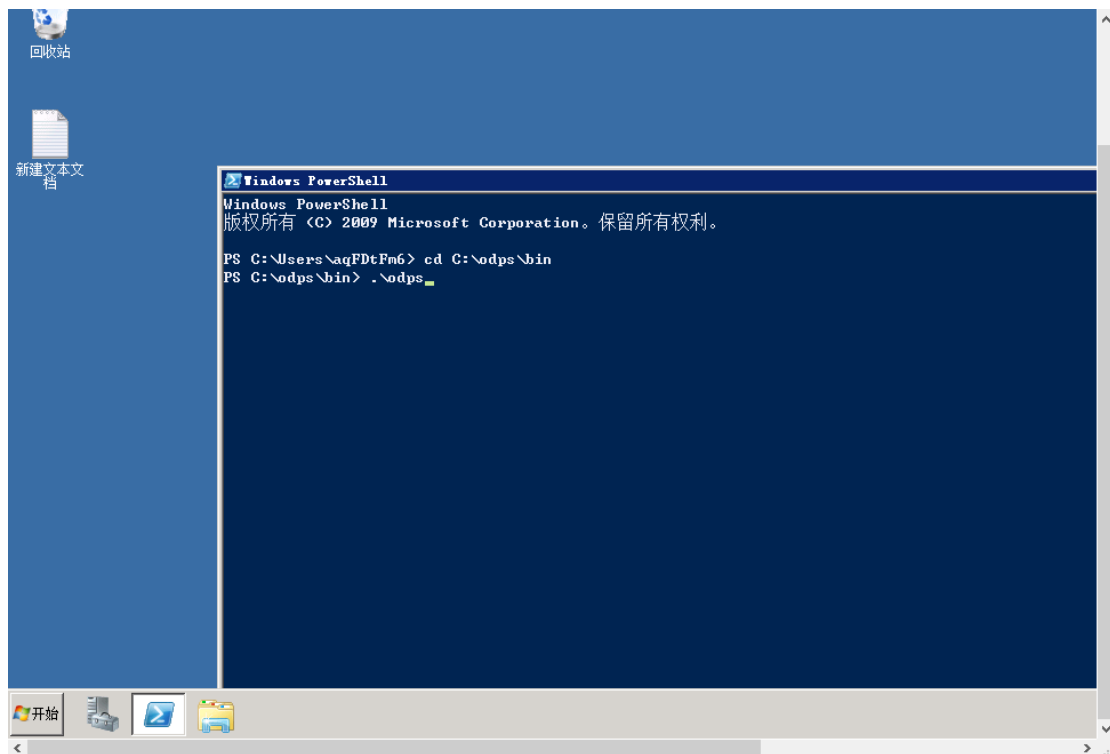
B 通过 windows 的命令工具（如运行->cmd 或者 PowerShell），进入到 odps 的安装目录来启动 odps



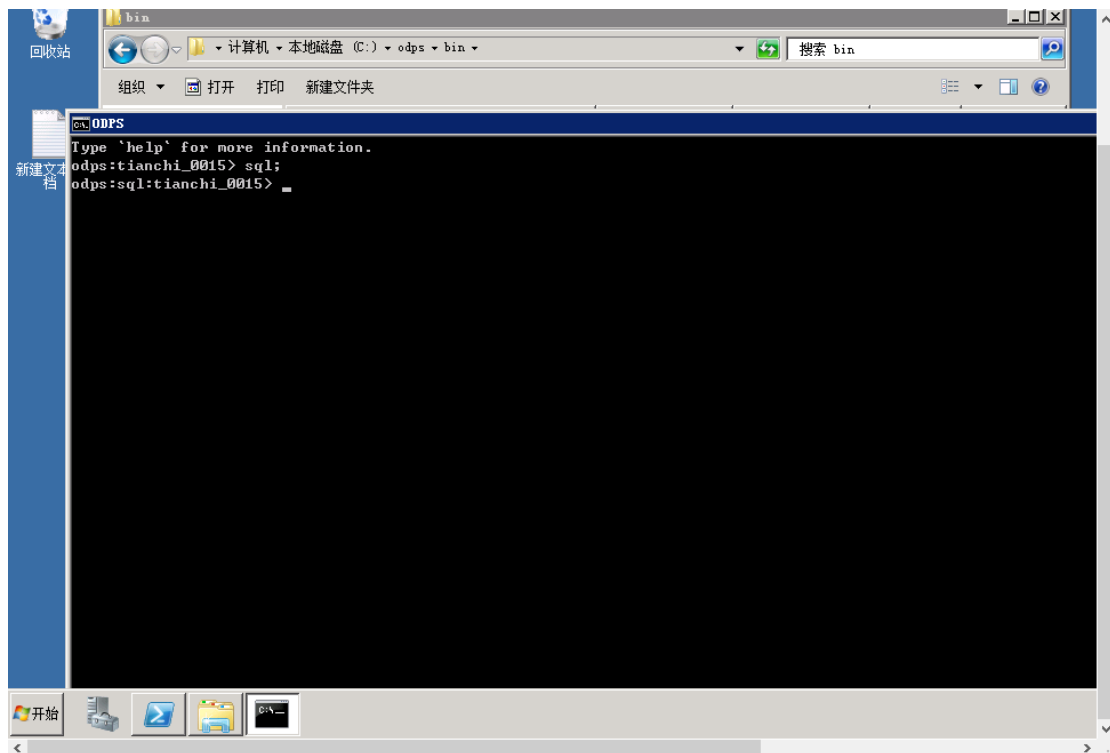


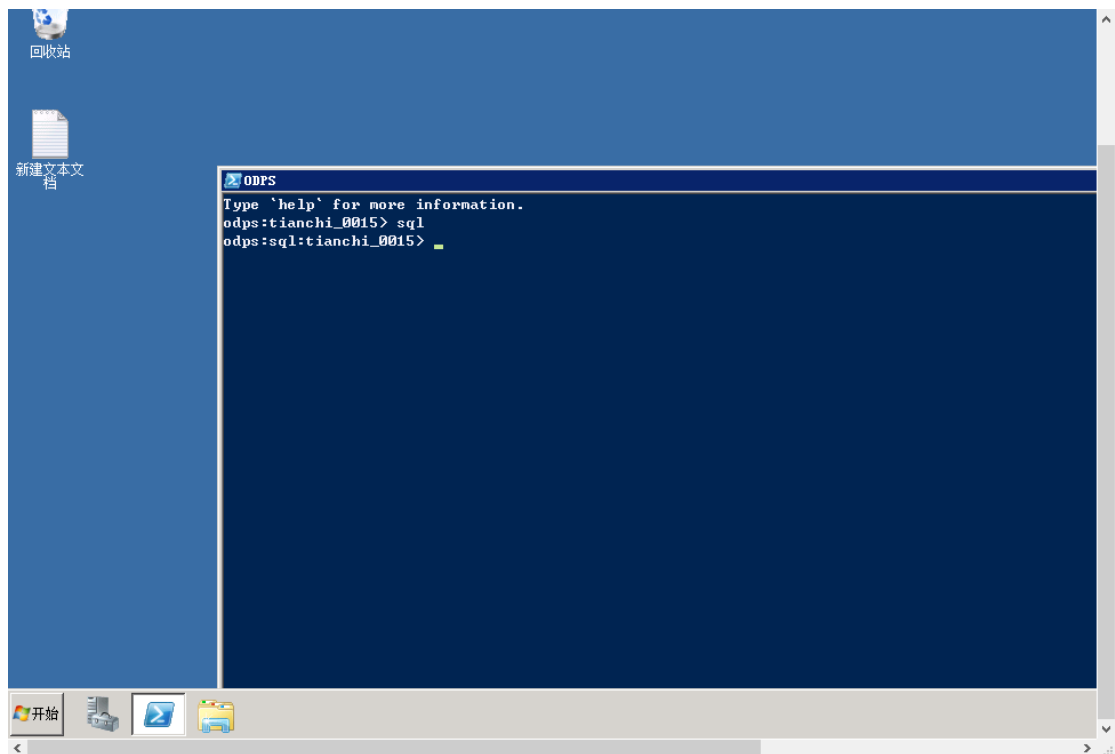
或者直接通过 powershell 直接进入 odps 命令行:



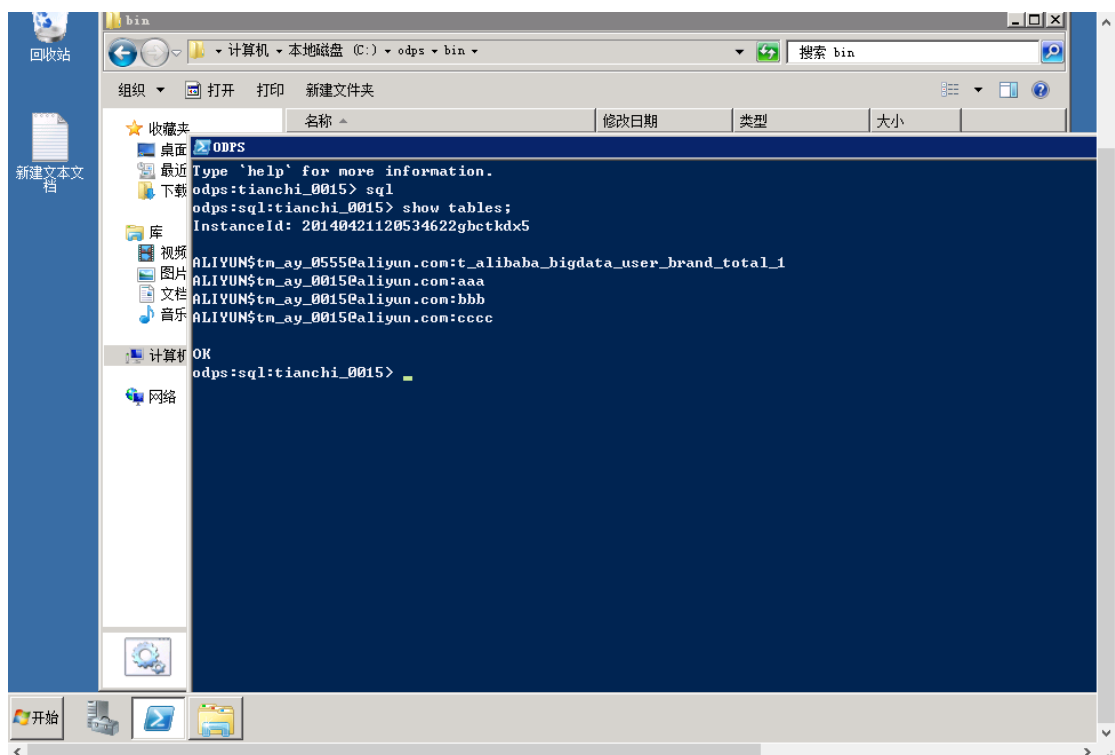


3 敲入 sql，进入到 SQL 模式：

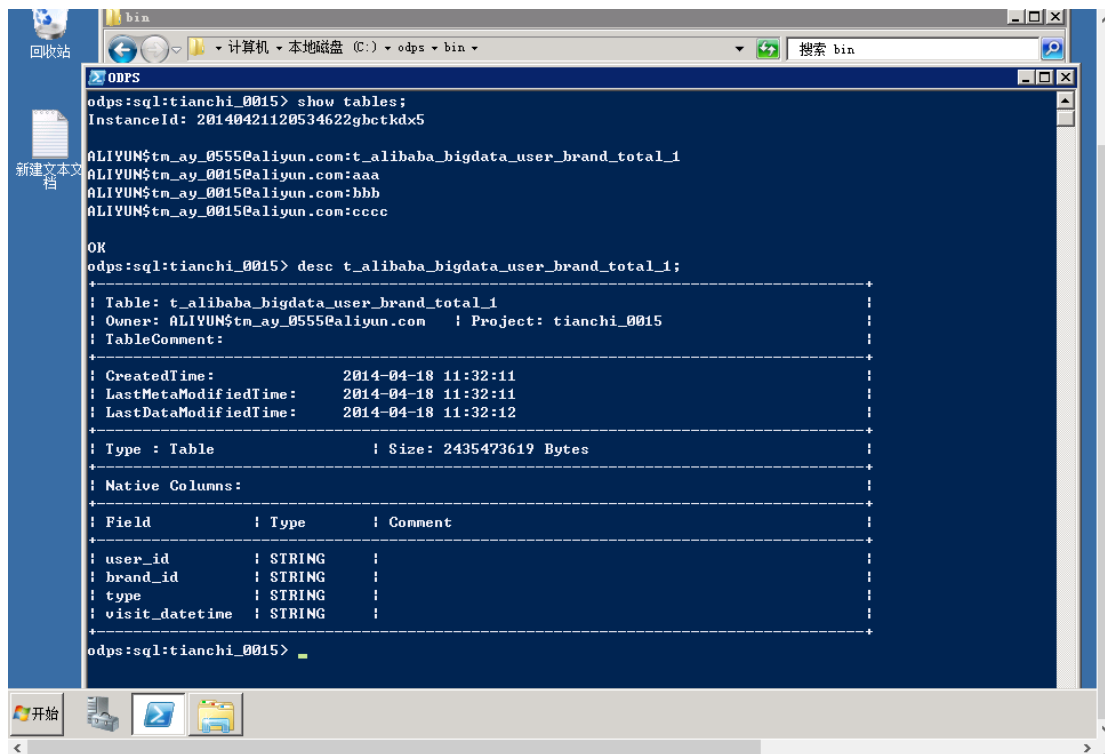




比赛用到的数据会放到一张公开的表里面，例如 `t_alibaba_bigdata_user_brand_total_1`，你可以用 SQL 语句查看数据表(`show tables`)来查看比赛的数据表是否存在。



接着，我们可以通过 `desc` 语句来查看表结构：



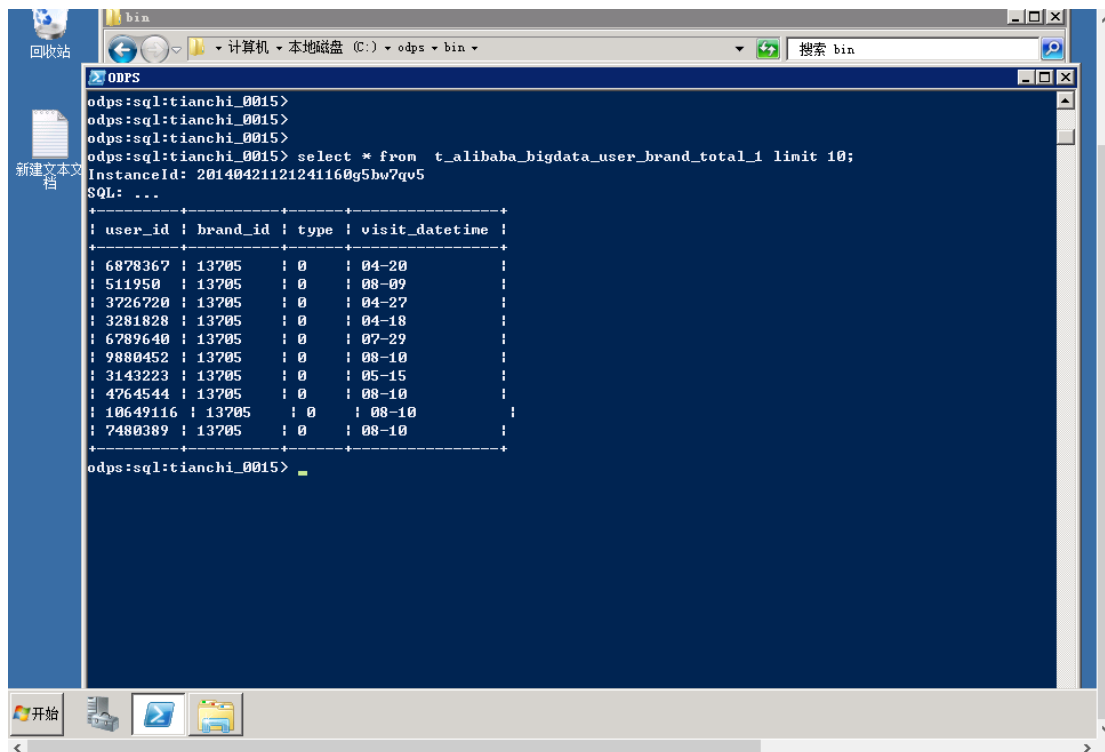
```
odps:sql:tianchi_0015> show tables;
InstanceId: 20140421120534622gbetkdx5

ALIYUN$tm_ay_0555@aliyun.com:t_alibaba_bigdata_user_brand_total_1
ALIYUN$tm_ay_0015@aliyun.com:aaa
ALIYUN$tm_ay_0015@aliyun.com:bbb
ALIYUN$tm_ay_0015@aliyun.com:cccc

OK
odps:sql:tianchi_0015> desc t_alibaba_bigdata_user_brand_total_1;
+-----+-----+
! Table: t_alibaba_bigdata_user_brand_total_1 !
! Owner: ALIYUN$tm_ay_0555@aliyun.com      ! Project: tianchi_0015 !
! TableComment:                            !
+-----+-----+
! CreatedTime:          2014-04-18 11:32:11 !
! LastMetaModifiedTime: 2014-04-18 11:32:11 !
! LastDataModifiedTime: 2014-04-18 11:32:12 !
+-----+-----+
! Type : Table           ! Size: 2435473619 Bytes !
+-----+-----+
! Native Columns:       !
+-----+-----+
! Field      ! Type   ! Comment !
+-----+-----+
! user_id    ! STRING !          !
! brand_id   ! STRING !          !
! type       ! STRING !          !
! visit_datetime ! STRING !          !
+-----+-----+

odps:sql:tianchi_0015>
```

最后，你还可以用 `select` 语句来查看一些基本的数据：



```
odps:sql:tianchi_0015>
odps:sql:tianchi_0015>
odps:sql:tianchi_0015>
odps:sql:tianchi_0015> select * from t_alibaba_bigdata_user_brand_total_1 limit 10;
InstanceId: 20140421121241160g5bw7qv5
SQL: ...

+-----+-----+-----+-----+
! user_id ! brand_id ! type ! visit_datetime !
+-----+-----+-----+-----+
! 6878367 ! 13705    ! 0    ! 04-20          !
! 511950  ! 13705    ! 0    ! 08-09          !
! 3726720 ! 13705    ! 0    ! 04-27          !
! 3281028 ! 13705    ! 0    ! 04-18          !
! 6789640 ! 13705    ! 0    ! 07-29          !
! 9880452 ! 13705    ! 0    ! 08-10          !
! 3143223 ! 13705    ! 0    ! 05-15          !
! 4764544 ! 13705    ! 0    ! 08-10          !
! 10649116 ! 13705    ! 0    ! 08-10          !
! 7480389 ! 13705    ! 0    ! 08-10          !
+-----+-----+-----+-----+

odps:sql:tianchi_0015>
```

更多的 SQL 语句可以参考官方的文档，其语法结构与主流的 `mysql` 和 `oracle` 都基本相似，很容易上手。

2 设计算法

为了能使同学们快速上手，成功运行第一个 **hello world**，在这个例子中，我们采用最简单最粗暴的最热门推荐算法作为示例：

- 1 计算购买次数最多的 **TOP-N** 品牌
- 2 给每个用户都推荐这 **TOP-N** 个品牌

实现这个算法有 2 种方式：

A 通过 SQL 方式（**注意每一段代码之间不要有空行**）

```
create table brand_info as

select *

        ,rank() over (partition by 1 order by buy_cnt desc) as brand_rank

from (

    select brand_id

        ,count(1) as buy_cnt

    from t_alibaba_bigdata_user_brand_total_1

    where type=0

    group by brand_id

) temp

;
```

```
create table hot_brands as

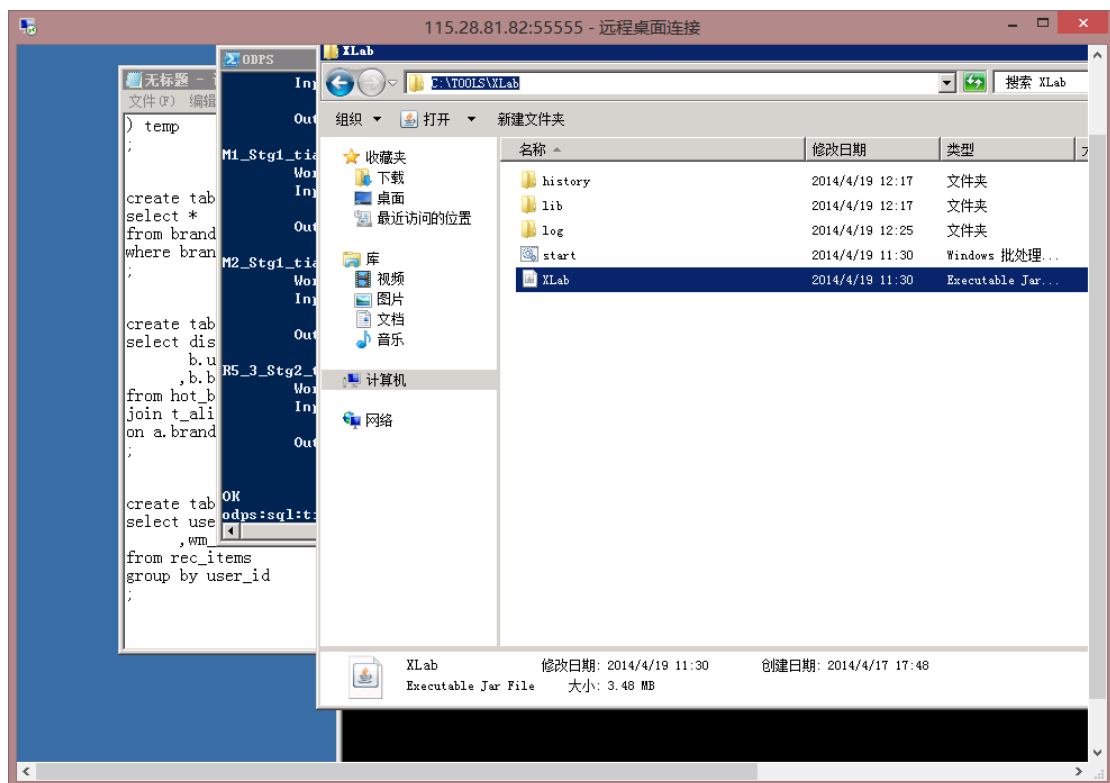
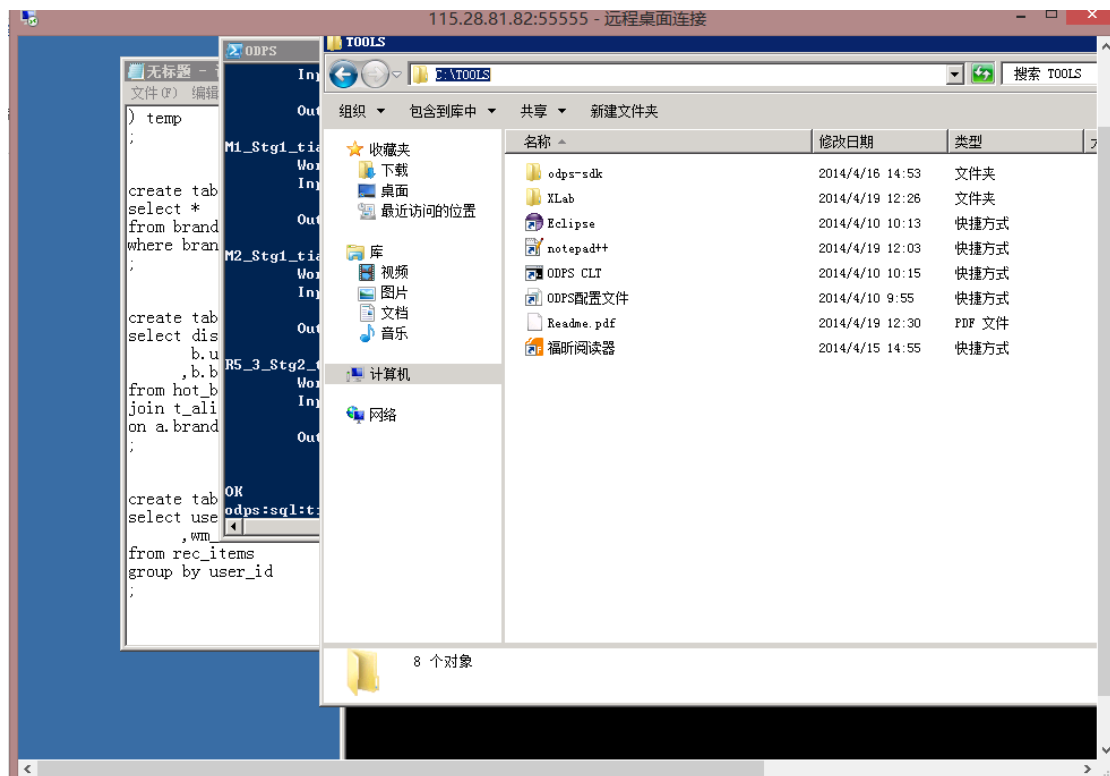
select *

from brand_info

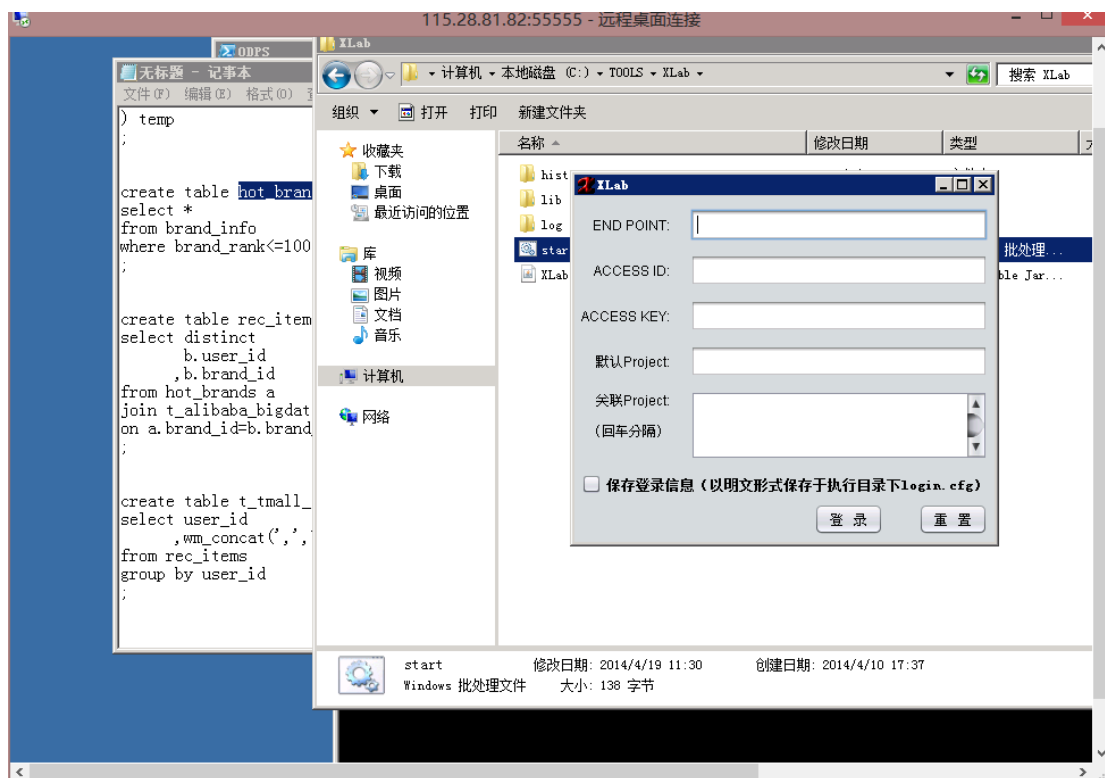
where brand_rank<=5

;
```

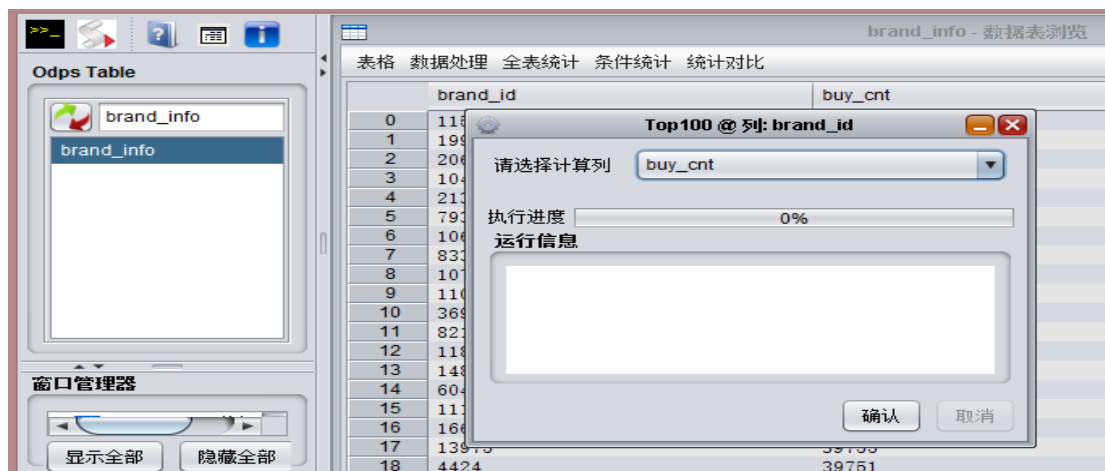
B 通过官方提供的数据挖掘算法工具XLab：



从 conf 里面的 odps 配置里面，对应填上 id 和 key 等信息：



选定要分析的表，从数据处理里面，选择 TOP100 来找出购买最多的前 100 名品牌。



更多的统计算法和挖掘算法，请关注官方提供的文档。

3 提交结果：

最终，我们需要输出每个用户的推荐品牌列表。如下示例代码，其中 `t_tmall_add_user_brand_predict_dh` 是竞赛要求输出的结果表名。

```
create table rec_items as
```

```
select distinct
```

```

        b.user_id

    ,b.brand_id

from hot_brands a

join tianchi10.t_alibaba_bigdata_user_brand_total b

on a.brand_id=b.brand_id

;

```

--结果输出的表明假设为t_tmall_add_user_brand_predict_dh

--需要用wm_concat来进行推荐品牌的拼接

```

create table t_tmall_add_user_brand_predict_dh as

select user_id

        ,wm_concat(' ',brand_id) as brand

from rec_items

group by user_id

;

```

输出表类似于：

```

+-----+-----+
| user_id | brand |
+-----+-----+
| 10000017 | 15539 |
| 1000002  | 14934,20643,5070 |
| 10000027 | 7930  |
| 1000003  | 10320 |
| 10000032 | 1124,18846,5500 |
| 10000036 | 699,10794,9888,21330 |
| 10000044 | 13856 |
| 10000069 | 7874,13975,1604,1110 |

```

官方会在每天的定点进行评估，然后按照 F1-SCORE 进行排名，公布结果