Hive 数据仓库应用教程

第九章 综合案例：电商销售大数据分析

强义原稿（jiǎng yì yuán gǎo）

授课对象：专科院校学生

语言：中文（口语体）

【幻灯片 1】 项目九 综合案例：电商销售大数据分析（封面）

同学们好！
欢迎来到我们Hive数据仓库的第九章课程。
今天是我们这门课非常重要的一节课，因为我们要做一个综合（zōng hé）案例。
前面几章我们学了很多Hive的基础知识，比如建表、查询、函数等等。
今天，我们要把这些知识全部用起来，做一个真实的项目。

这个项目叫做’电商销售大数据分析’。
简单说，就是分析一个网上购物平台（píng tái）的销售数据。
就像淘宝（Taobao）、京东（JD）那样的电商平台，每天都会产生大量的订单数据。
我们要用Hive来分析这些数据，找出有用的信息。

好，我们开始吧！

【幻灯片 2】 任务 9-1 电商销售大数据分析

这一章只有一个大任务，就是任务9-1：电商销售大数据分析。
虽然（suī rán）只是一个任务，但是里面包含（bāo hán）了很多小的分析内容。
我们会从最简单的统计（tǒng jì）开始，一步一步做到复杂的分析。

大家不用紧张（jǐn zhāng），我会带着大家一步一步来。
每一步都会讲清楚为什么这样做，怎么做。

【幻灯片 3】 9.1.1 任务描述（rèn wù miáo shù）

好，我们先来看看这个任务到底要做什么。

▶ 任务背景（bèi jǐng）

全世界有数十亿人在使用电商平台进行购物。
想想看，光是中国的淘宝、京东，每天就有多少人在买东西？
这些平台在很短的时间内就产生了大量（dà liàng）的数据。

▶ 我们要做什么？

本项目（xiàng mù）要学会三件事：
第一，怎么用Hive数据仓库来装载（zhuāng zài）电商的历史销售数据。
第二，怎么用HiveQL对各项指标（zhǐ biāo）进行统计分析。
第三，通过这个案例，进一步熟悉（shú xī）Hive数据仓库的实际应用。

简单说：加载数据 → 写查询语句 → 得到分析结果。
这就是我们今天要做的事情。

【幻灯片 4】 9.1.2 任务准备知识 — 电商销售大数据

▶ 1. 什么是电子商务（diàn zǐ shāng wù）？

电子商务就是在网上做生意。

买家和卖家不用见面，通过网络就可以完成交易（jiāo yì）。

比如你在淘宝上买一件衣服，你不需要去实体（shí tǐ）店，直接在手机上下单就行了。

这就是电子商务。

▶ 2. 电商数据包括什么？

电商数据包括很多内容：

— 用户的浏览（liú lǎn）信息：你看了哪些商品

— 购买消费记录：你买了什么，花了多少钱

— 用户对商品的评价（píng jià）：好评还是差评

— 卖家的买卖记录

— 产品交易量、库存量（kù cún liàng）

— 商家的信用信息

▶ 3. 为什么电商需要大数据？

电商经历（jīng lì）了三个时代：

第一个时代：比用户数量，谁的用户多谁赢。

第二个时代：比销量（xiāo liàng），谁卖的多谁赢。

现在是第三个时代：比大数据，谁能更好地分析数据谁赢。

所以大数据分析对电商来说非常重要！

【幻灯片 5】 9.1.2 任务准备知识 — 数据概况（gài kuàng）

▶ 数据来源

我们用的数据来自一个奢侈品（shē chǐ pǐn）电商平台的后台历史销售数据。
就是像卖名牌包、名牌手表那样的网站。
平台想要通过分析数据，了解用户的特点，做精准营销（yíng xiāo）。

▶ 数据是什么样的？

数据是txt文本文件，每一行就是一条订单记录。
每条记录包含12个字段（zì duàn），我来一个一个说：

1. uid — 用户标识（biāo shí），就是用户的ID号码
2. pid — 商品标识，就是商品的ID号码
3. sex — 性别，M表示男，F表示女
4. age — 年龄段，比如18-25、26-35等等
5. vocation — 职业（zhí yè），用数字表示不同的职业
6. city — 居住城市，用字母表示
7. livetime — 居住时间
8. marriage — 婚姻（hūn yīn）状况，0表示未婚，1表示已婚
9. level1 — 商品一级分类
10. level2 — 商品二级分类
11. level3 — 商品三级分类
12. sale — 销售额（xiāo shòu é），就是花了多少钱

大家可以想象一下，这就像一个大表格，每一行是一个人买了一件东西的记录。

【幻灯片 6】 9.1.2 任务准备知识 — 数据准备（步骤1-2）

▶ 图片说明

幻灯片上有两张截图（jié tú）：

左边（图9-1）：是Xftp 5文件传输工具的界面（jiè miàn）。

我们可以看到，左边是本地电脑，有一个文件叫e-data.txt。

右边是node2服务器（fú wù qì）的/data目录，e-data.txt已经传上去了，大小是21.04MB。

右边（图9-2）：是Xshell 5远程连接工具的界面。

我们可以看到在node3上执行（zhí xíng）了Hive命令。

首先创建了数据库eshop，然后用show databases命令看到eshop已经创建成功了。

▶ 操作步骤讲解

步骤1：把历史销售数据文件e-data.txt上传到node2虚拟机（xū nǐ jī）的/data目录下。

这就像把文件从你的电脑拷贝到服务器上。

步骤2：创建Hive数据库eshop。

eshop就是 e-commerce shop 的缩写（suō xiě），电商商店的意思。

🔧 实习命令：创建数据库

```
create database eshop;
show databases;
```

【幻灯片 7】 9.1.2 任务准备知识 — 数据准备（步骤3）

▶ 图片说明

左边（图9-3）：显示了创建sales表的完整（wán zhěng）SQL语句。
我们可以看到，先用了 use eshop 切换到eshop数据库。
然后用 create table sales 创建了一张表。
表的字段就是我们前面说的那12个字段。
最后设置了行分隔符（fēn gé fú）用'|'（竖线），
还跳过了文件的第一行（因为第一行是标题行）。

右边（图9-4）：用 show tables 确认（què rèn）表已经创建成功了。
然后开始执行加载数据的命令。

▶ 步骤3详解：创建Hive销售订单表

注意几个重要的地方：

— uid是int类型，pid是string类型
— sex、age是string类型（因为age是一个范围，比如"18-25"）
— vocation是int类型（用数字代表职业）
— sale是double类型（销售额有小数）
— 分隔符是竖线'|'
— 跳过第一行标题行：skip.header.line.count="1"

🔧 实习命令：创建表并加载数据

```
use eshop;

create table sales (
  uid int,
  pid string,
  sex string,
  age string,
  vocation int,
  city string,
  livetime string,
  marriage int,
  level1 int,
  level2 int,
  level3 int,
  sale double
)
```

```
row format delimited fields terminated by '|'
tblproperties(
  "skip.header.line.count"="1"
);
```

【幻灯片 8】 9.1.2 任务准备知识 — 数据准备（步骤4）

▶ 图片说明（图9-5）

这张截图显示了数据加载成功后的查询结果。
我们可以看到执行了 select * from sales limit 1 查询。
返回了一条数据：
— uid=1000001，pid=P00069042，sex=F（女性）
— age=0-17（0到17岁），vocation=10，city=A
— livetime=2，marriage=0（未婚）
— level1=3，level2=NULL，level3=NULL
— sale=8370.0

可以看到，有些字段是NULL，这是正常的。
不是每条记录都有完整的分类信息。

▶ 步骤4：加载数据

用load data命令把txt文件的数据装载到sales表里。
local表示文件在本地，不是在HDFS上。
overwrite表示覆盖（fù gài）原来的数据。

🔧 实习命令：加载数据

```
load data local inpath '/data/e-data.txt'
  overwrite into table sales;

-- 验证数据是否加载成功
select * from sales limit 1;
```

【幻灯片 9】 9.1.3 任务实施 — 1. 统计订单总数

▶ 图片说明（图9-6）

截图显示了查询结果：

items = 475313

也就是说，这个数据集（shù jù jí）一共有475,313条订单记录。

这就是47万多条数据！这就是'大数据'的感觉。

▶ 查询分析讲解

我们要统计一共有多少条订单。

用什么函数呢？用count函数。

count(1) 的意思是：对每一行都计数，数一数一共有多少行。

as items 是给结果取一个别名（bié míng），叫items（订单数）。

这是最简单的统计，就像你数一数手上有多少张牌一样。

实习命令：统计订单总数

```
select count(1) as items from sales;
```

【幻灯片 10】 9.1.3 任务实施 — 2. 统计总销售额

▶ 图片说明（图9-7）

截图显示了三个统计结果：

— sales = 4.434399703E9（约44亿）— 这是总销售额

— avg_pay = 752741.419623154 — 这是每个用户的平均消费

— avg_price = 9329.430718284582 — 这是每笔订单的平均金额

▶ 查询分析讲解

这条查询同时计算了三个指标：

第一个：sum(sale) 总销售额

sum就是求和，把所有订单的销售额加起来。

结果大约44亿元，说明这个平台的总交易额非常大。

第二个：sum(sale)/count(distinct uid) 每位用户平均消费

总销售额除以用户数（去重后的用户数）。

distinct uid 表示去掉重复（chóng fù）的用户，同一个人只算一次。

每个用户平均花了大约75万元，因为是奢侈品电商，所以金额比较高。

第三个：sum(sale)/count(sale) 每笔订单的平均金额

总销售额除以订单总数。

每笔订单平均约9300元。

🔧 实习命令：统计总销售额、人均消费、订单均价

```
select
  sum(sale) sales,
  sum(sale)/count(distinct uid) avg_pay,
  sum(sale)/count(sale) avg_price
from sales;
```

【幻灯片 11】 9.1.3 任务实施 — 3. 统计各性别消费情况

▶ 图片说明（图9-8）

截图显示了男女消费情况的统计结果：

— M（男性）：4225个用户，人均消费804983.60，总消费约34亿，
用户占比71.72%，消费占比76.70%

— F（女性）：1666个用户，人均消费620254.50，总消费约10亿，
用户占比28.28%，消费占比23.30%

▶ 查询分析讲解

这个查询比前面的要复杂一些，用到了子查询（zǐ chá xún）和窗口函数。

先看内层查询（子查询）：按性别分组，统计每个性别的用户数、人均消费、总消费。
再看外层查询：计算每个性别的用户占比和消费占比。

几个重要的函数：

— cast(... as decimal(8,2))：把结果转换（zhuǎn huàn）成小数，保留2位。

— sum(users) over()：窗口函数，计算所有行的users总和。
over()里面是空的，表示对所有行计算。

— group by sex：按性别分组。

— order by pay_percent desc：按消费占比（zhàn bǐ）降序（jiàng xù）排列。

▶ 分析结论

从结果可以看出：男性用户虽然人数更多（71.72%），但是消费占比更高（76.70%）。
说明男性用户是这个奢侈品平台的主力（zhǔ lì）消费群体。

实习命令：统计各性别消费情况

```
select *,
  cast(users/(sum(users) over()) as decimal(8,2))
    as sex_percent,
  cast(sum_pay/(sum(sum_pay) over()) as decimal(8,2))
    as pay_percent
from (
  select sex,
    count(distinct uid) users,
```

```
    cast((sum(sale)/count(distinct uid))*100
      as decimal(8,2)) avg_user,
    sum(sale) sum_pay
  from sales group by sex
) as s order by pay_percent desc;
```

【幻灯片 12】 9.1.3 任务实施 — 4. 统计各年龄段消费情况

▶ 图片说明（图9-9）

截图显示了7个年龄段的统计结果：

— 26-35岁：2053个用户，消费占比39.79%，排第一！

— 36-45岁：1167个用户，消费占比20.13%

— 18-25岁：1069个用户，消费占比18.03%

— 46-50岁：531个用户，消费占比8.22%

— 51-55岁：481个用户，消费占比7.22%

— 55+岁：372个用户，消费占比3.94%

— 0-17岁：218个用户，消费占比2.67%

▶ 查询分析讲解

这个查询用了with ... as语法，也叫CTE（Common Table Expression，通用表表达式）。

CTE就是先定义一个临时（lín shí）的结果集s，然后在后面的查询中使用它。

好处是让SQL更清晰（qīng xī），更容易阅读。

内层：按年龄段分组，统计用户数、总消费、人均消费。

外层：用窗口函数计算占比。

▶ 分析结论

26-35岁是消费最多的群体，接近40%！

这就是用户画像（huà xiàng）的意义：知道谁在花钱，就可以针对他们做营销。

🔧 实习命令：统计各年龄段消费情况

```
with s as (
  select age,
    count(distinct uid) users,
    sum(sale) sum_pay,
    cast(sum(sale)/count(distinct uid)
      as decimal(10,2)) avg_pay
  from sales group by age
)
select s.*,
  cast((users/sum(users) over())*100
    as decimal(10,2)) age_percent,
```

```
  cast((sum_pay/sum(sum_pay) over())*100
    as decimal(10,2)) pay_percent
from s order by pay_percent desc;
```

【幻灯片 13】 9.1.3 任务实施 — 5. 统计各职业消费情况

▶ 图片说明（图9-10）

截图显示了21种职业的消费统计。

消费最高的前三个职业是：

— 职业4：740个用户，占比13.06%

— 职业0：688个用户，占比12.48%

— 职业7：669个用户，占比10.90%

消费最少的是职业8：只有17个用户，占比0.29%。

▶ 查询分析讲解

这个查询的结构（jié gòu）和年龄段分析几乎（jī hū）一样。

只是把group by age换成了group by vocation。

注意这里用了 vocation job 给vocation取了一个别名叫job，让结果更好理解。

▶ 分析结论

不同职业的消费差异（chā yì）很大。

如果平台想做精准营销，就应该重点关注消费占比高的职业群体。

实习命令：统计各职业消费情况

```
with s as (
  select vocation job,
    count(distinct uid) users,
    sum(sale) sum_pay,
    cast(sum(sale)/count(distinct uid)
      as decimal(10,2)) avg_pay
  from sales group by vocation
)
select s.*,
  cast((users/sum(users) over())*100
    as decimal(10,2)) vocation_percent,
  cast((sum_pay/sum(sum_pay) over())*100
    as decimal(10,2)) pay_percent
from s order by pay_percent desc;
```

【幻灯片 14】 9.1.3 任务实施 — 6. 统计各婚姻状况消费情况

▶ 图片说明（图9-11）

截图显示了婚姻状况的消费统计：

— 未婚（0）：3417个用户，占用户的58%，消费占比59.05%

— 已婚（1）：2474个用户，占用户的42%，消费占比40.95%

▶ 查询分析讲解

同样的分析套路（tào lù），按marriage分组。

marriage=0表示未婚，marriage=1表示已婚。

▶ 分析结论

未婚用户的数量和消费都略高于已婚用户。

但差距不是很大，说明婚姻状况对消费的影响（yǐng xiǎng）不太明显。

不过未婚用户可能有更多的可支配（kě zhī pèi）收入，所以消费略高。

实习命令：统计各婚姻状况消费情况

```
with s as (
  select marriage,
    count(distinct uid) users,
    sum(sale) sum_pay,
    cast(sum(sale)/count(distinct uid)
      as decimal(10,2)) avg_pay
  from sales group by marriage
)
select s.*,
  cast((users/sum(users) over())*100
    as decimal(10,2)) marriage_percent,
  cast((sum_pay/sum(sum_pay) over())*100
    as decimal(10,2)) pay_percent
from s order by pay_percent desc;
```

【幻灯片 15】 9.1.3 任务实施 — 7. 统计订单量排名前十的产品

▶ 图片说明（图9-12）

截图显示了订单量最多的10个产品：

— 第1名：P00265242，1624单，占比0.34%

— 第2名：P00025442，1420单，占比0.30%

— 第3名：P00110742，1407单，占比0.30%

虽然前十名的占比都不高（每个不到1%），但在47万条订单中，能有1000多单已经很不错了。

▶ 查询分析讲解

这里用到了一个新的函数：rank() 排名函数。

rank() over(order by sale_items desc) 的意思是：

按照订单数量从多到少排名。

然后外层用 where sales_rank<=10 只取前十名。

注意：这里不能直接在WHERE里写 rank()<=10。

因为窗口函数不能直接在WHERE中使用。

所以要用子查询，先算出排名，再在外层过滤（guò lǜ）。

实习命令：统计订单量排名前十的产品

```
with s as (
  select pid,
    count(pid) sale_items
  from sales group by pid
)
select * from (
  select rank() over(order by sale_items desc)
    sales_rank,
    s.*,
    cast((sale_items/sum(sale_items) over())*100
      as decimal(10,2)) items_percent
  from s
) t where sales_rank<=10;
```

【幻灯片 16】 9.1.3 任务实施 — 8. 统计销售额排名前十的产品

▶ 图片说明（图9-13）

截图显示了销售额最高的10个产品：

— 第1名：P00025442，销售额约2462万

— 第2名：P00110742，销售额约2333万

— 第3名：P00255842，销售额约2201万

▶ 查询分析讲解

和上一个查询类似（lèi sì），只是排名的依据（yī jù）从订单数量变成了销售额。

rank() over(order by sum_pay desc) 按销售额降序排名。

▶ 对比分析

有趣的是：订单量最多的产品和销售额最高的产品不完全一样。

比如P00025442在订单量排名第2，但在销售额排名第1。

说明这个产品单价比较高。

P00265242订单量排第1，但销售额没进前10，说明这个产品单价较低但卖得多。

实习命令：统计销售额排名前十的产品

```
with s as (
  select pid, sum(sale) sum_pay
  from sales group by pid
)
select * from (
  select rank() over(order by sum_pay desc)
    pay_rank,
    s.*,
    cast((sum_pay/sum(sum_pay) over())*100
      as decimal(10,2)) pay_percent
  from s
) t where pay_rank<=10 order by pay_rank;
```

【幻灯片 17】 9.1.3 任务实施 — 9. 统计一级产品类别

▶ 图片说明（图9-14）

截图显示了各一级分类的统计结果：

— 类别1：订单12万，占比25.68%，销售额占比37.44%，排第一！

— 类别5：订单13万，占比27.66%，销售额占比18.50%

— 类别8：订单近10万，占比20.89%，销售额占比16.79%

▶ 查询分析讲解

按level1（一级分类）分组，统计每个类别的订单数量和销售额。

然后计算订单占比和销售额占比。

▶ 分析结论

类别1虽然订单数不是最多的，但销售额占比最高（37.44%）。

说明类别1的商品单价比较高，是利润（lì rùn）最大的品类。

类别5的订单数最多，但销售额占比只有18.50%，说明这个品类的商品单价较低。

🔧 实习命令：统计一级产品类别

```
with s as (
  select level1,
    count(pid) sale_items,
    sum(sale) pay_sum
  from sales group by level1
)
select s.*,
  cast((sale_items/sum(sale_items) over())*100
    as decimal(10,2)) items_precent,
  cast((pay_sum/sum(pay_sum) over())*100
    as decimal(10,2)) pay_precent
from s order by pay_precent desc;
```

【幻灯片 18】 9.1.3 任务实施 — 10. 统计各性别销售额前五的产品

▶ 图片说明（图9-15）

截图显示了按性别分组后，每个性别销售额最高的5个产品：
女性（F）前5名：P00255842、P00110842、P00059442、P00025442、P00110742
男性（M）前5名：P00025442、P00110742、P00184942、P00112142、P00057642

▶ 查询分析讲解

这是本章最复杂的一个查询！
用到了 partition by，这是窗口函数中非常重要的一个概念（gài niàn）。

rank() over(partition by sex order by sum_pay desc)
  partition by sex 的意思是：在每个性别内部分别（fēn bié）排名。
  不是所有数据一起排名，而是男性有男性的排名，女性有女性的排名。

就好比班级考试排名：
  如果不加partition by，就是全班一起排。
  加了partition by sex，就是男生排男生的，女生排女生的。

最后用 where rank_pay<=5 取每个性别的前5名。
order by sex, rank_pay 让结果先按性别，再按排名排列。

▶ 分析结论

男性和女性购买的热门产品有所不同。
平台可以根据这个结果，分别向男性和女性推荐（tuī jiàn）不同的产品。

实习命令：统计各性别销售额前五的产品

```
with s as (
  select sex, pid, level1, level2, level3,
    count(pid) sale_items,
    sum(sale) sum_pay
  from sales
  group by sex, pid, level1, level2, level3
)
select * from (
  select s.*,
    rank() over(partition by sex
```

```
      order by sum_pay desc) rank_pay,
    cast((sum_pay/sum(sum_pay) over())*100
      as decimal(10,2)) pay_percent
  from s
) t where rank_pay<=5
order by sex, rank_pay;
```

【幻灯片 19】 项目总结

好，同学们，我们这一章的内容就全部讲完了！
让我们来回顾（huí gù）一下今天学了什么：

▶ 知识回顾

第一，我们了解了这个项目的背景和需求。
我们知道了电商数据长什么样，有哪些字段。

第二，我们学会了数据导入操作。
创建数据库 → 创建表 → 加载数据。
这三步是使用Hive分析数据的基础（jī chǔ）。

第三，我们学会了10种不同的数据分析方法：
1）统计订单总数 — count函数
2）统计总销售额 — sum函数
3）按性别统计 — group by + 窗口函数
4）按年龄段统计 — with...as + group by
5）按职业统计
6）按婚姻状况统计
7）订单量TOP10 — rank()排名函数
8）销售额TOP10
9）一级产品分类统计
10）分性别TOP5 — partition by分区排名

▶ 重点技能

— 聚合函数（jù hé hán shù）：count, sum
— 窗口函数：sum() over(), rank() over()
— 分区窗口函数：rank() over(partition by ...)
— CTE语法：with...as
— 数据类型转换：cast(... as decimal(...))

这些都是大数据分析中非常实用（shí yòng）的技能。
希望大家课后多练习（liàn xí），把这些查询都自己试一遍。

【幻灯片 20】 感谢您的观看

好的同学们，今天的课就到这里。
今天我们完成了一个完整的电商大数据分析项目。
从数据准备到各种统计分析，一共做了10个分析任务。

课后请大家把这些HiveQL语句自己敲一遍，理解每一步在做什么。
如果有不明白的地方，随时来问我。

下课！同学们再见！