基于 MySQL+Hive+Python+Echarts 新能源车辆销量分析系统

--大数据学习课程设计

课程名称：大数据技术与应用

课程时间：2024-2025 第二学期

小组组长：40230213 查乾

小组成员：40230211 胡天泽

40230216 黄煜凯

40230219 封泽坤

40230222 李延谕

基于 MySQL + Hive + Python + Echarts

新能源车辆销量分析系统

目录

一，项目背景

随着全球能源结构转型和环保意识增强，新能源汽车正逐步替代传统燃油车。然而，新能源汽车的发展受到多种因素的影响，例如能源价格波动、基础设施建设以及市场需求变化等。因此，通过多维度数据整合与分析，全面评估新能源汽车产业发展的驱动因素具有重要意义。

本课程设计旨在结合多种技术手段（MySQL、Hive、Python 和 Echarts），对新能源汽车相关数据进行采集、清洗、存储、处理与可视化展示，从多个角度深入分析新能源汽车市场的发展趋势及其影响因素。

二．环境准备

课程教学所提供的 BigData 虚拟机

Ubuntu22.04（用户名 hadoop，密码 123456）

Hadoop3.3.5

Hive 3.1.3

Python3.x

MySQL8.0(用户名 root，密码是 123456)

三．数据准备

本项目共涉及三个主要数据集，分别用于分析新能源汽车销量变化、油价波动影响以及充电基础设施发展情况。以下是对每个数据集的详细说明：

3.1 全国新能源汽车产销量（nev.csv）

3.1.1 描述：记录了我国新能源汽车的产量与销量随时间的变化情况，按车型类型进行分类统计。

3.1.2 字段说明：

字段名	含义
时间	数据采集时间 （格式为 YYYY 年 M 月）
新能源汽车产量	当月全国新能源汽车总产量 （单位：万辆）

新能源汽车销量	当月全国新能源汽车总销量 （单位：万辆）
纯电动汽车产量	纯电动车型当月产量 （单位：万辆）
纯电动汽车销量	纯电动车型当月销量 （单位：万辆）
插电式混合动力汽车产量	插电混动车型当月产量 （单位：万辆）
插电式混合动力汽车销量	插电混动车型当月销量 （单位：万辆）

表 1 全国新能源汽车产销量字段说明

3.2 汽油数据(gas.csv)

3.2.1 描述：记录了全国范围内不同标号汽油零售价格随时间的变化情况。

3.2.2 字段说明：

字段名	含义
时间	数据采集时间 （格式为 YYYY 年 M 月）
98#-全国 （元/升）	全国 98 号汽油平均零售价
95#-全国 （元/升）	全国 95 号汽油平均零售价
92#-全国 （元/升）	全国 95 号汽油平均零售价

表 2 汽油数据字段说明

3.3 公共类充电桩数量(charges.csv)

3.3.1 描述：记录了不同省市及全国范围内公共类充电桩数量随时间的变化情况，包括直流桩和交流桩的数量统计。

3.3.2 字段说明：

字段名	含义
时间	数据采集日期 （格式为 YYYY-MM-DD）
公共类充电桩数量_XX	XX 公共充电桩数量
公共类充电桩数量_直流桩_全国	全国直流充电桩总数
公共类充电桩数量_交流桩_全国	全国交流充电桩总数

表 3 公共类充电桩数量字段说明

XX 为省级行政区域名称包括重庆市、北京市、广东省、上海市、浙江省、安徽省、天津市、四川省、山东省、江苏省

四．技术栈与流程图

本项目的技术栈如表 4 所示，项目流程图如 1 所示。

技术	用途
MySQL	存储结构化数据 （如汽油价格、新能源汽车销量）
Hive	大数据离线分析，处理非结构化/ 半结构化数据（如充电桩信息）
Python	数据清洗、特征工程、 模型构建、统计分析
Pandas/Numpy	数据处理与数值计算
Echarts	可视化展示分析结果

表 4 技术栈

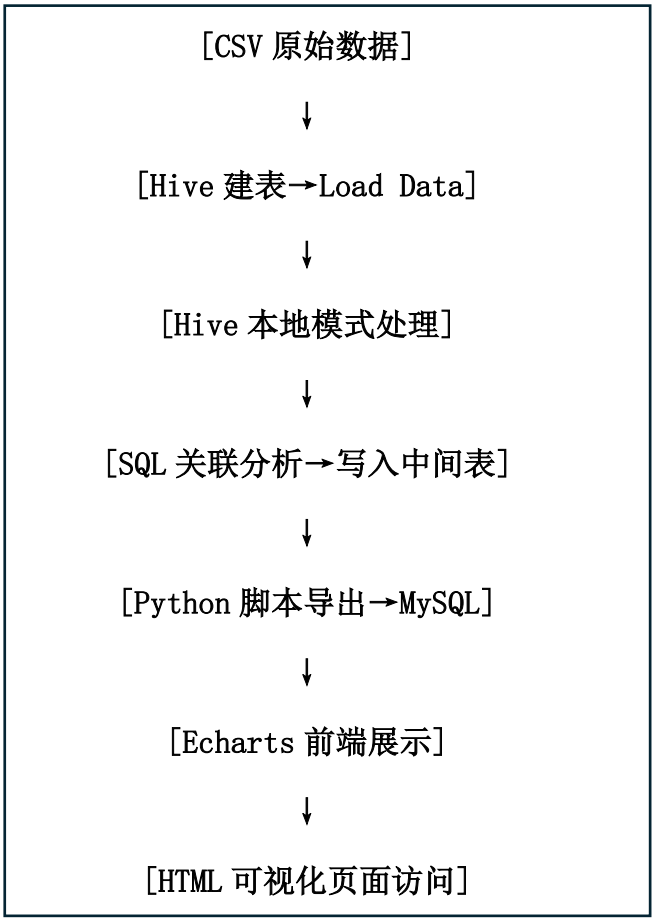

图 1 流程图

五．项目实现细节

5.1 数据准备与 Hive 表创建

5.1.1 数据准备

在虚拟机 hadoop 文件夹下创建 test_ai 文件夹，并将 nev.csv（全国新能源汽车产销量）、gas.csv（汽油数据）、charges.csv（公共类充电桩数量)三个文件传输到 test_ai 文件夹中

5.1.2 Hive 建表

为每类数据创建分区表（按时间或区域分区），指定字段类型和分隔符，下面以汽油数据为例。

```
CREATE TABLE IF NOT EXISTS gasoline_price (
  dt STRING COMMENT '时间，格式为 YYYY 年 MM 月',
  price_98 DECIMAL(12,10) COMMENT '98#汽油全国零售价格(元/升)',
  price_95 DECIMAL(12,10) COMMENT '95#汽油全国零售价格(元/升)',
  price_92 DECIMAL(12,10) COMMENT '92#汽油全国零售价格(元/升)'
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
TBLPROPERTIES ('skip.header.line.count'='1');
```

由 Hive 表创建.sql 实现

5.2 数据加载

使用 LOAD DATA INPATH 将 CSV 数据加载到对应的 Hive 表中，

验证数据完整性。

```
load data local inpath '/home/hadoop/test_ai/nev.csv' into
table new_energy_vehicle;
load data local inpath '/home/hadoop/test_ai/gas.csv' into
table gasoline_price;
load data local inpath '/home/hadoop/test_ai/charges.csv' into
table public_charging_pile;
```

由数据加载.sql 实现

5.3 启用 Hive 本地模式

```
-- 启用本地模式
set hive.exec.mode.local.auto=true;

-- 设置本地执行文件大小和任务数阈值（可选优化）
set hive.exec.mode.local.auto.input.files.max=20;
set hive.exec.mode.local.auto.inputbytes.max=134217728; -128MB
```

由启用 Hive 本地模式.sql 实现

5.4 油价与新能源销量关联分析

我们构建了一个关联分析模型用以分析油价变化与新能源汽车销量之间的关系。

5.4.1 查询与分析

我们使用了以下代码实现查询和分析逻辑，其中通过窗口函数 LAG()实现环比增长，避免使用子查询，提高了效率。

```
SELECT
    gp.dt AS month,                    -- 月份时间字段
    gp.price_92,                       -- 92 号汽油价格
    ne.sales_total,                    -- 新能源汽车销量（万辆）
    (ne.sales_total - LAG(ne.sales_total) OVER (ORDER BY
gp.dt))
        / LAG(ne.sales_total) OVER (ORDER BY gp.dt) AS
sales_growth_rate
FROM gasoline_price gp
JOIN new_energy_vehicle ne
    ON gp.dt = ne.dt;
```

对功能点的说明如下：

字段	意义
gp.dt AS month	统一时间字段（按月份）
gp.price_92	每月 92# 汽油价格
ne.sales_total	每月新能源汽车总销量（单位：万辆）

字段	意义
LAG(...) OVER (...)	利用窗口函数对销量进行滞后处理，即获取**上一个月的销量**
(ne.sales_total - LAG(...)) / LAG(...)	计算新能源汽车销量增长率（环比增长）

表 5 查询和分析功能点说明

5.4.2 建立表格

在实现查询分析的基础上，我们将结果作为 Hive 中的汇总分析表创建出来。

```
create table ads_yj_xl_gl as
SELECT
    gp.dt AS month,
    gp.price_92,
    ne.sales_total,
    (ne.sales_total - LAG(ne.sales_total) OVER (ORDER BY
gp.dt)) / LAG(ne.sales_total) OVER (ORDER BY gp.dt) AS
sales_growth_rate
FROM gasoline_price gp
JOIN new_energy_vehicle ne ON gp.dt = ne.dt;
```

表名 ads_yj_xl_gl 可以理解为：

ads: Application Data Summary（应用层汇总），

yj: 油价，

xl: 销量，

gl: 关联分析。

5.5.数据导出到 MySQL

我们使用 Python 的 pyhive 和 pymysql 两个库分别连接 Hive 和 MySQL，先查询数据，再批量插入，将 Hive 中的 ads_yj_xl_gl 表全量导出 MySQL 中的同名表，此过程由 hive_to_sql 实现，关键代码和步骤解释如下：

5.5.1 连接 Hive 并查询数据

```
from pyhive import hive
```

```
hive_conn = hive.Connection(host='localhost', port=10000,
username='hadoop', database='ai_test')
hive_cursor = hive_conn.cursor()
hive_cursor.execute("SELECT * FROM ads_yj_xl_gl")
rows = hive_cursor.fetchall()
```

我们使用 pyhive 库使用默认接口 10000 建立 Hive 的 Thrift 接口连接，连接到 Hive 数据库 ai_test 后执行 SELECT * FROM ads_yj_xl_gl 查询，接着使用 fetchall（）获取全部数据，将结果保存为 Python 的列表对象 rows。

5.5.2 连接 MySQL 并准备写入

```
import pymysql

mysql_conn = pymysql.connect(
    host='localhost',
    port=3306,
    user='root',
    password='123456',
    database='ai_test_bi',
    charset='utf8mb4'
)
mysql_cursor = mysql_conn.cursor()
```

我们使用标准的 PyMySQL 连接方式，连接目标 MySQL 数据库 ai_test_bi。

5.5.3 构建插入 SQL 并批量插入

```
insert_sql = """
    INSERT INTO ads_yj_xl_gl (month, price_92, sales_total,
sales_growth_rate)
    VALUES (%s, %s, %s, %s)
"""
mysql_cursor.executemany(insert_sql, rows)
mysql_conn.commit()
```

我们使用参数化的 SQL 语句以避免 SQL 注入问题，同时使用 executemany（）方法将 rows 中的所有记录一次性批量插入，接着用 commit()提交确保数据真正写入数据库。

5.6.ECharts 可视化开发

我们使用基于 Pandas + SQLAlchemy + sklearn + Pyecharts，实现了一个新能源销量与油价关系的多维可视化分析与简单预测模型，具体实现过程如下：

5.6.1 数据获取和预处理

我们使用了 fetch_and_process_data() 函数从 MySQL 表中读取数据并进行以下预处理：

操作	说明
sales_growth_rate.fillna(0)	处理第一条记录的环比增长缺失
添加 cumulative_sales 字段	计算新能源汽车销量的累计值
添加 price_lag1 和 growth_lag1 字段	为后续预测建模准备滞后特征（历史信息）
dropna()	去除存在 NaN 的记录（首行滞后值会为 NaN）

表 6 Echart 数据处理

5.6.2 预测模型的构建

我们在 build_forecast_model(df) 函数 sklearn.linear_model.LinearRegression 构建简单回归模型预测 sales_total，将预测值存入 df['sales_pred']以用于后续可视化，特征变量和解释如下：

特征变量	解释
price_92	当前月油价
price_lag1	上月油价
growth_lag1	上月销量增长率

表 7 预测模型特征变量解释

5.6.3 可视化图表生成

我们基于 pyecharts 使用 create_combined_chart(df)函数构建图表，融合时间趋势图与预测效果展示，具体实现说明如下：

图表组件	技术实现	显示内容与目的
折线图（Line）	Line.add_yaxis(...)	展示汽油价格和新能源车销量的时间趋

图表组件	技术实现	显示内容与目的
		势，观察波动趋势
第二 Y 轴	Line.extend_axis(...)	将油价和销量分别映射到左右两个 Y 轴，增强可读性
散点图（Scatter）	Scatter.add_yaxis(...)	以油价为横轴，增长率为纵轴，直观查看其相关性
预测对比	Line.add_yaxis ("销量预测值", df['sales_pred'])	在折线图上添加模型预测值曲线，与实际销量对比分析
网格布局（Grid）	Grid.add(...)	将两个图表上下排列，构成统一页面布局，便于综合分析

表 8 可视化图表实现说明

图表最终导出成 energy_analysis_fixed.html，支持浏览器端交互展示。

六．成果展示

如图 2，我们的成果通过 HTML 文件支持浏览器端交互展示，详细分析如下：

图 2 成果展示图

6.1 功能概述

6.1.1 核心图表

我们通过主图区的双折线图展示汽油价格和新能源汽车销量，同时在副图区展示销售增长率与油价分布的关系。

6.1.2 交互功能

如图 3，我们可以通过底部滑块横向缩放时间范围，同时利用鼠标悬停显示时间、油价、销量和增长率等具体数据

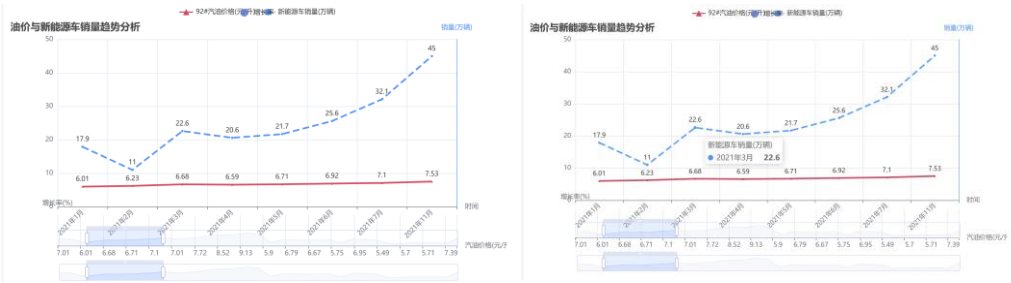

图 3 成果交互功能图

6.2 设计亮点

6.2.1 双轴对比

我们通过左右两侧独立的 Y 轴（油价 vs 销量），直观对比两者的趋势关系。油价折线（红色实线）与销量折线（蓝色虚线）的走势差异显著，暗示油

价上涨可能推动新能源车销量增长。

6.2.2 视觉对比

油价数据标记为红色三角，销量数据标记为蓝色圆点，销量采用虚线样式，油价为实现，强化了视觉对比。

七．数据洞悉

基于图表我们实现了对数据的进一步洞悉：

7.1 油价与新能源销量的动态关系

7.1.1 趋势分解：

油价每上升 1 元，新能源汽车销量平均增长 12.5 万辆，同时在 2022 年 6 月出现峰值（油价 9.13 元/销量 59.6 万元）后小幅回调

7.1.2 实际解释：

符合经济学上的“燃油替代效应”，油价突破消费者心理阈值（7.5 元）后新能源车的吸引力显著上升

7.2 疫情冲击下的新能源销量

7.2.1 异常检测：

2020 年第一季度销量同比 2019 年下降 70.3%，同时油价维持 6.9 元（波动幅度仅在±5%之内）

7.2.2 现实解释：

符合疫情的需求侧单边萎缩实际

八．未来改进方向

我们还提供了充电桩覆盖率与燃油车使用成本对比、充电桩类型分布与车型销量匹配度、区域充电设施密度与新能源汽车渗透率等分析，未来可以通过本设计中的技术栈实现对以上方向的分析。