山东大学＿＿计算机科学与技术＿＿学院

＿＿大数据分析实践＿＿课程实验报告

学号：202300130092	姓名：马浩鑫 任俊毅 王启源 王俊磊	班级： 23 数据
实验题目：spark 实践		
实验学时：2	实验日期：20251202	

实验目的：

1.熟悉 Apache Spark 的基本运行环境和编程模式；

2.掌握 Spark DataFrame API 进行数据读取、统计分析的方法；

3.学会使用 Spark SQL 对大规模数据进行结构化查询；

4.了解 Spark MLlib 中机器学习模型的基本使用流程；

5.通过小组分工协作，完成一个完整的数据分析与建模实验。

软件环境：

操作系统：Windows

Python 版本：Python 3.7

Spark 版本：Spark 2.4.8

开发工具：命令行 + PySpark

数据集：sales_data.csv（销售记录数据集）

实验步骤与内容：

1. 小组分工：

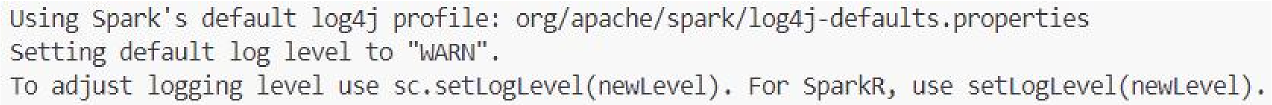
王俊磊：Spark 环境初始化、数据加载、整体流程整合；

王启源：Spark DataFrame API 数据分析；

任俊毅：Spark SQL 查询分析；

马浩鑫：Spark MLlib 机器学习建模

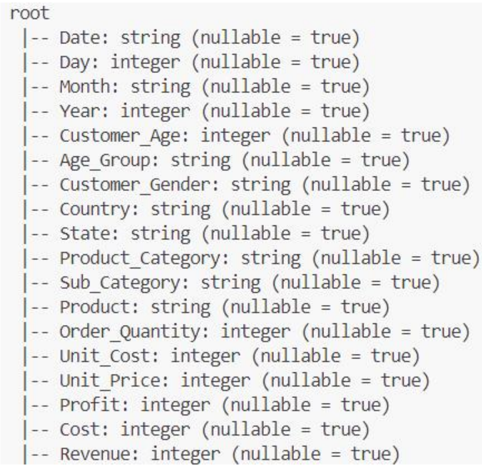
2. spark 环境配置与初始化：



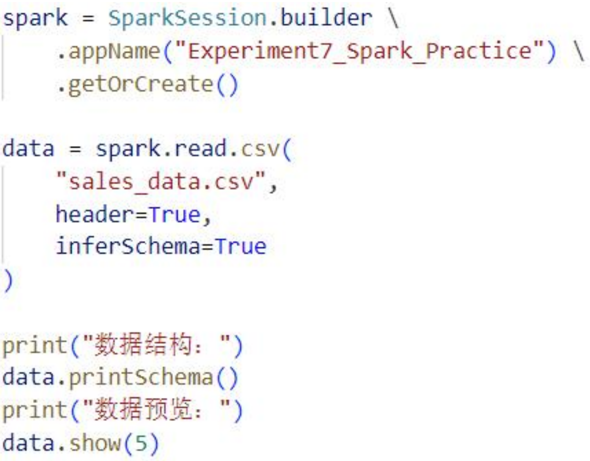
```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
```

Spark 运行环境配置正确，实验可以顺利进行。

数据结构预览



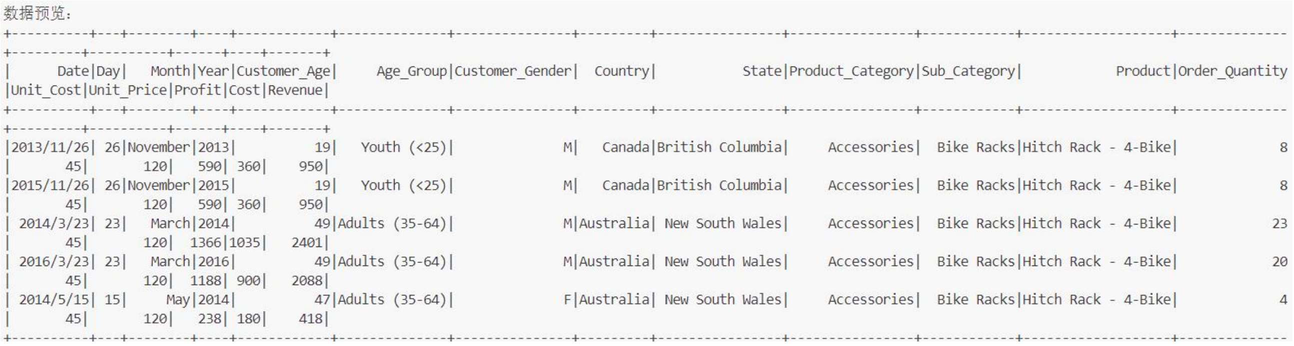
```
root
 |-- Date: string (nullable = true)
 |-- Day: integer (nullable = true)
 |-- Month: string (nullable = true)
 |-- Year: integer (nullable = true)
 |-- Customer_Age: integer (nullable = true)
 |-- Age_Group: string (nullable = true)
 |-- Customer_Gender: string (nullable = true)
 |-- Country: string (nullable = true)
 |-- State: string (nullable = true)
 |-- Product_Category: string (nullable = true)
 |-- Sub_Category: string (nullable = true)
 |-- Product: string (nullable = true)
 |-- Order_Quantity: integer (nullable = true)
 |-- Unit_Cost: integer (nullable = true)
 |-- Unit_Price: integer (nullable = true)
 |-- Profit: integer (nullable = true)
 |-- Cost: integer (nullable = true)
 |-- Revenue: integer (nullable = true)
```



```python
spark = SparkSession.builder \
    .appName("Experiment7_Spark_Practice") \
    .getOrCreate()

data = spark.read.csv(
    "sales_data.csv",
    header=True,
    inferSchema=True
)

print("数据结构：")
data.printSchema()
print("数据预览：")
data.show(5)
```

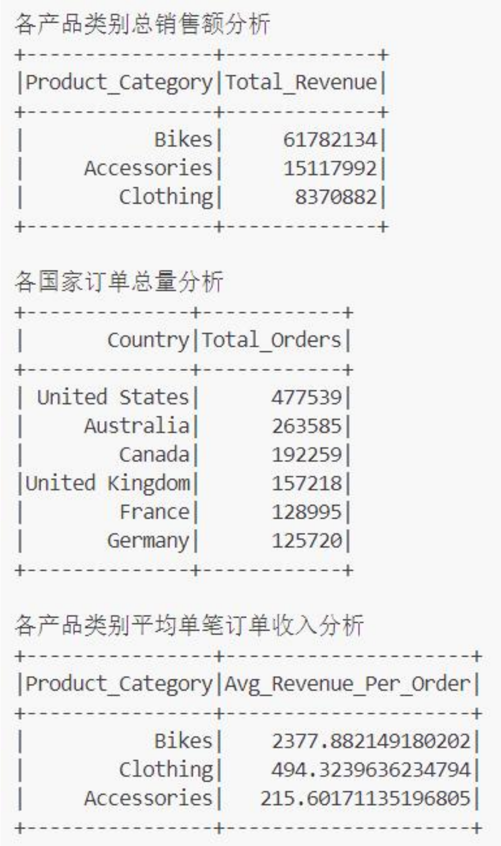
数据预览

能够正常显示数据前 5 行，数据加载无误。

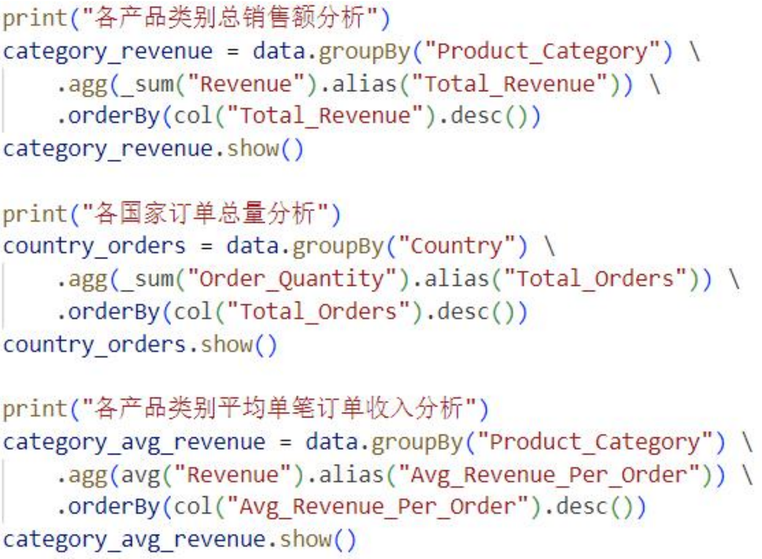


```
数据预览：
+----------+---+--------+----+------------+--------------+---------------+---------+----------------+----------------+------------+-------------------+-------------
+---------+----------+------+----+-------+
|      Date|Day|   Month|Year|Customer_Age|     Age_Group|Customer_Gender|  Country|           State|Product_Category|Sub_Category|            Product|Order_Quantity
|Unit_Cost|Unit_Price|Profit|Cost|Revenue|
+----------+---+--------+----+------------+--------------+---------------+---------+----------------+----------------+------------+-------------------+-------------
+---------+----------+------+----+-------+
|2013/11/26| 26|November|2013|          19|   Youth (<25)|              M|   Canada|British Columbia|     Accessories|  Bike Racks|Hitch Rack - 4-Bike|            8
|       45|       120|   590| 360|    950|
|2015/11/26| 26|November|2015|          19|   Youth (<25)|              M|   Canada|British Columbia|     Accessories|  Bike Racks|Hitch Rack - 4-Bike|            8
|       45|       120|   590| 360|    950|
| 2014/3/23| 23|   March|2014|          49|Adults (35-64)|              M|Australia| New South Wales|     Accessories|  Bike Racks|Hitch Rack - 4-Bike|           23
|       45|       120|  1366|1035|   2401|
| 2016/3/23| 23|   March|2016|          49|Adults (35-64)|              M|Australia| New South Wales|     Accessories|  Bike Racks|Hitch Rack - 4-Bike|           20
|       45|       120|  1188| 900|   2088|
| 2014/5/15| 15|     May|2014|          47|Adults (35-64)|              F|Australia| New South Wales|     Accessories|  Bike Racks|Hitch Rack - 4-Bike|            4
|       45|       120|   238| 180|    418|
+----------+---+--------+----+------------+--------------+---------------+---------+----------------+----------------+------------+-------------------+-------------
```

3.DataFrame API 业务分析



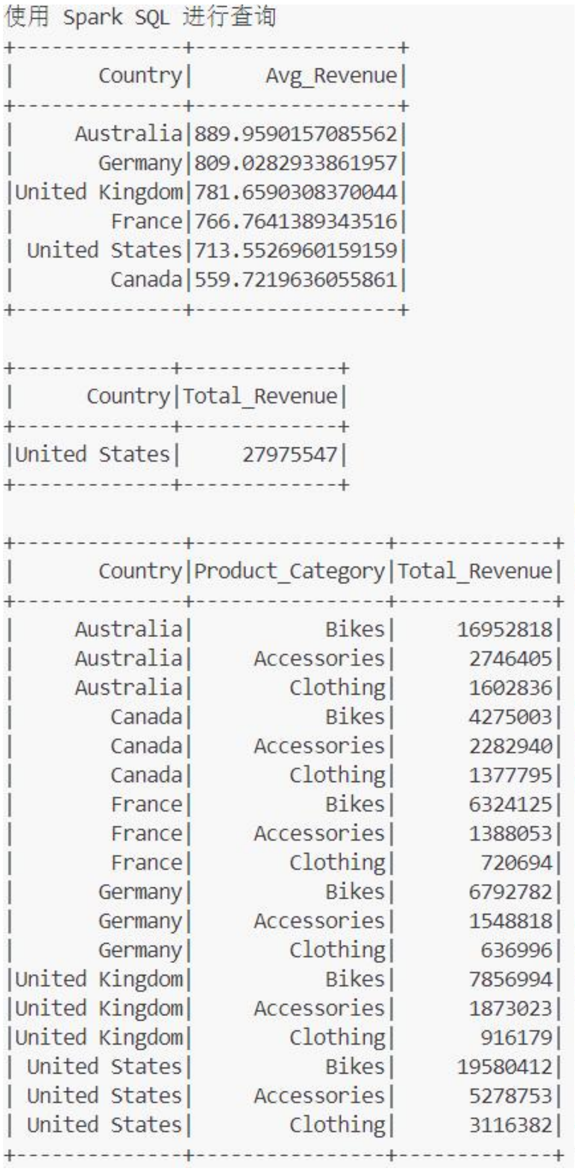
```
各产品类别总销售额分析
+----------------+-------------+
|Product_Category|Total_Revenue|
+----------------+-------------+
|           Bikes|     61782134|
|     Accessories|     15117992|
|        Clothing|      8370882|
+----------------+-------------+

各国家订单总量分析
+--------------+------------+
|       Country|Total_Orders|
+--------------+------------+
| United States|      477539|
|     Australia|      263585|
|        Canada|      192259|
|United Kingdom|      157218|
|        France|      128995|
|       Germany|      125720|
+--------------+------------+

各产品类别平均单笔订单收入分析
+----------------+---------------------+
|Product_Category|Avg_Revenue_Per_Order|
+----------------+---------------------+
|           Bikes|    2377.882149180202|
|        Clothing|    494.3239636234794|
|     Accessories|   215.60171135196805|
+----------------+---------------------+
```



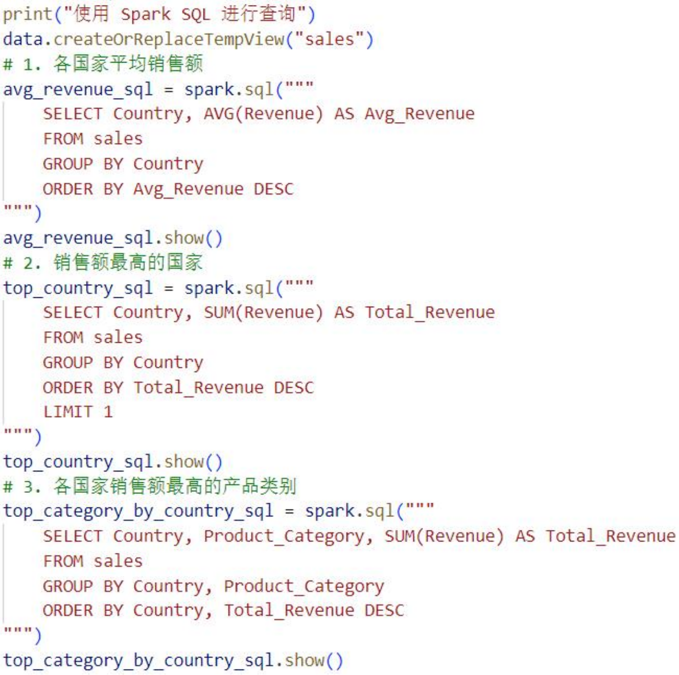
```
print("各产品类别总销售额分析")
category_revenue = data.groupBy("Product_Category") \
    .agg(_sum("Revenue").alias("Total_Revenue")) \
    .orderBy(col("Total_Revenue").desc())
category_revenue.show()

print("各国家订单总量分析")
country_orders = data.groupBy("Country") \
    .agg(_sum("Order_Quantity").alias("Total_Orders")) \
    .orderBy(col("Total_Orders").desc())
country_orders.show()

print("各产品类别平均单笔订单收入分析")
category_avg_revenue = data.groupBy("Product_Category") \
    .agg(avg("Revenue").alias("Avg_Revenue_Per_Order")) \
    .orderBy(col("Avg_Revenue_Per_Order").desc())
category_avg_revenue.show()
```

Bikes 类总销售额最高，远高于另外两者，美国的订单总量最多，是最大市场，bikes 类平均每笔订单收入最高价值也最高，clothing 和 accessories 作为服装和配件略少

4.Spark SQL 查询分析



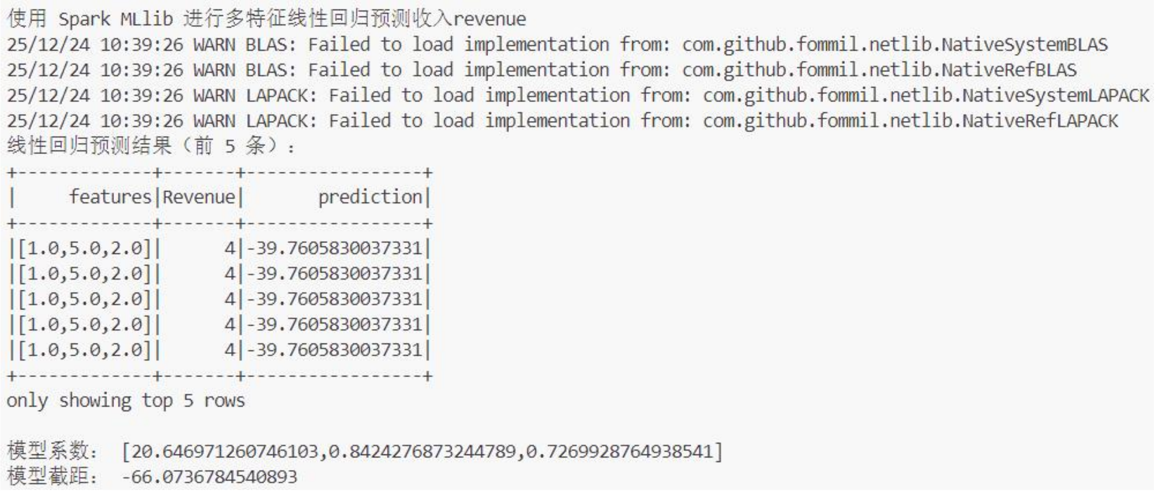
```
使用 Spark SQL 进行查询
+--------------+----------------+
|       Country|     Avg_Revenue|
+--------------+----------------+
|     Australia|889.9590157085562|
|       Germany|809.0282933861957|
|United Kingdom|781.6590308370044|
|        France|766.7641389343516|
| United States|713.5526960159159|
|        Canada|559.7219636055861|
+--------------+----------------+

+-------------+-------------+
|      Country|Total_Revenue|
+-------------+-------------+
|United States|     27975547|
+-------------+-------------+

+--------------+----------------+-------------+
|       Country|Product_Category|Total_Revenue|
+--------------+----------------+-------------+
|     Australia|           Bikes|     16952818|
|     Australia|     Accessories|      2746405|
|     Australia|        Clothing|      1602836|
|        Canada|           Bikes|      4275003|
|        Canada|     Accessories|      2282940|
|        Canada|        Clothing|      1377795|
|        France|           Bikes|      6324125|
|        France|     Accessories|      1388053|
|        France|        Clothing|       720694|
|       Germany|           Bikes|      6792782|
|       Germany|     Accessories|      1548818|
|       Germany|        Clothing|       636996|
|United Kingdom|           Bikes|      7856994|
|United Kingdom|     Accessories|      1873023|
|United Kingdom|        Clothing|       916179|
| United States|           Bikes|     19580412|
| United States|     Accessories|      5278753|
| United States|        Clothing|      3116382|
+--------------+----------------+-------------+
```



```python
print("使用 Spark SQL 进行查询")
data.createOrReplaceTempView("sales")
# 1. 各国家平均销售额
avg_revenue_sql = spark.sql("""
    SELECT Country, AVG(Revenue) AS Avg_Revenue
    FROM sales
    GROUP BY Country
    ORDER BY Avg_Revenue DESC
""")
avg_revenue_sql.show()
# 2. 销售额最高的国家
top_country_sql = spark.sql("""
    SELECT Country, SUM(Revenue) AS Total_Revenue
    FROM sales
    GROUP BY Country
    ORDER BY Total_Revenue DESC
    LIMIT 1
""")
top_country_sql.show()
# 3. 各国家销售额最高的产品类别
top_category_by_country_sql = spark.sql("""
    SELECT Country, Product_Category, SUM(Revenue) AS Total_Revenue
    FROM sales
    GROUP BY Country, Product_Category
    ORDER BY Country, Total_Revenue DESC
""")
top_category_by_country_sql.show()
```

澳大利亚用户的平均销售额最高，单笔消费能力最强，加拿大最低；美国是总消费额最大的国家；各国销售产品种类排序都是bikes>accessories>clothing

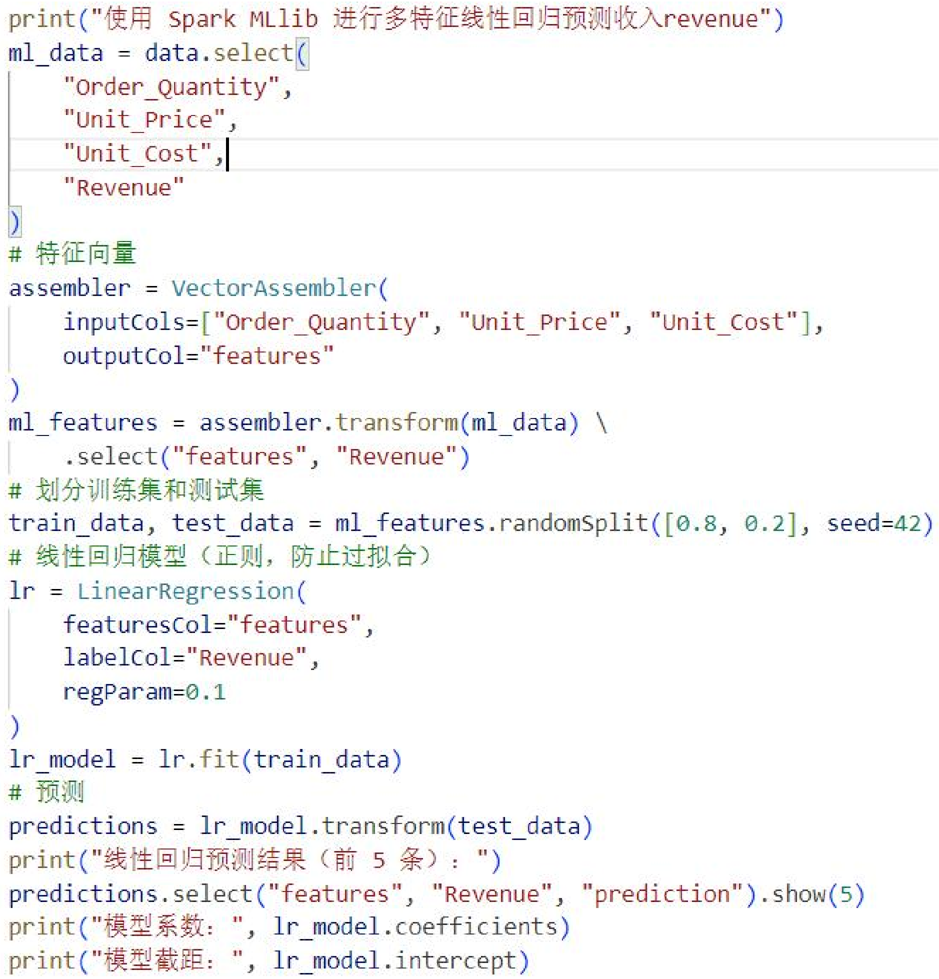
5.MLlib 机器学习建模



```
使用 Spark MLlib 进行多特征线性回归预测收入revenue
25/12/24 10:39:26 WARN BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeSystemBLAS
25/12/24 10:39:26 WARN BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeRefBLAS
25/12/24 10:39:26 WARN LAPACK: Failed to load implementation from: com.github.fommil.netlib.NativeSystemLAPACK
25/12/24 10:39:26 WARN LAPACK: Failed to load implementation from: com.github.fommil.netlib.NativeRefLAPACK
线性回归预测结果（前 5 条）：
+-------------+-------+-----------------+
|     features|Revenue|       prediction|
+-------------+-------+-----------------+
|[1.0,5.0,2.0]|      4|-39.7605830037331|
|[1.0,5.0,2.0]|      4|-39.7605830037331|
|[1.0,5.0,2.0]|      4|-39.7605830037331|
|[1.0,5.0,2.0]|      4|-39.7605830037331|
|[1.0,5.0,2.0]|      4|-39.7605830037331|
+-------------+-------+-----------------+
only showing top 5 rows

模型系数： [20.646971260746103,0.8424276873244789,0.7269928764938541]
模型截距： -66.0736784540893
```

使用订单数量、单价、单位成本等特征，对销售收入（Revenue）进行预测。

多特征线性回归（Linear Regression）



```python
print("使用 Spark MLlib 进行多特征线性回归预测收入revenue")
ml_data = data.select(
    "Order_Quantity",
    "Unit_Price",
    "Unit_Cost",
    "Revenue"
)
# 特征向量
assembler = VectorAssembler(
    inputCols=["Order_Quantity", "Unit_Price", "Unit_Cost"],
    outputCol="features"
)
ml_features = assembler.transform(ml_data) \
    .select("features", "Revenue")
# 划分训练集和测试集
train_data, test_data = ml_features.randomSplit([0.8, 0.2], seed=42)
# 线性回归模型（正则，防止过拟合）
lr = LinearRegression(
    featuresCol="features",
    labelCol="Revenue",
    regParam=0.1
)
lr_model = lr.fit(train_data)
# 预测
predictions = lr_model.transform(test_data)
print("线性回归预测结果（前 5 条）：")
predictions.select("features", "Revenue", "prediction").show(5)
print("模型系数：", lr_model.coefficients)
print("模型截距：", lr_model.intercept)
```

结论分析与体会：

1.本实验成功完成了 Spark 数据加载、分析、SQL 查询和机器学习任务；

2.DataFrame API 与 Spark SQL 结合使用，提高了数据分析效率；

3.Spark MLlib 能够快速构建分布式机器学习模型；

4.通过小组分工协作，提升了实验组织性和可维护性；