**【P32】-【电商销售数据分析】实验结果与问题报告示例**

**成员：黄少伟，许世烜**

1. **项目背景与目标**

随着互联网的发展普及,电商在人们购物中占重要部分,基于此我们的项目目标为电商销售分析,具体功能为商品销售排名和销售趋势分析

**二、实验环境与工具**

1. **硬件环境**：
   * 处理器：Intel Core i5
   * 内存：12GB
   * 存储：512GB
2. **软件环境**：
   * 操作系统：Windows 10
   * 编程语言：Python
   * 大数据处理平台：Spark
   * 数据分析工具：Pandas、NumPy、Matplotlib、Scikit-learn,seaborn
   * 集成开发环境：Jupyter Notebook

**三、项目内容与步骤**

1. **数据收集**：

数据读取：从 CSV 文件中读取电商销售数据，假设文件包含 Product（商品）、Sales（销售额）、Quantity（销售量）和 Shipping Cost（运输成本）等字段。

数据聚合：按商品分组，计算每个商品的三个核心指标数据预处理：

* + 使用 Pandas 等工具对数据进行清洗，去除重复、错误或无效的数据。
  + 进行数据转换和格式统一，以便后续分析。
  + 进行数据集成，将多个数据集合并为一个数据集。

1. **数据预处理**：
   * 使用描述性统计方法对数据进行初步分析，了解数据的基本特征。
   * 使用窗口函数Window实现按特定指标的排名分别计算了销售额排名、销售量排名和利润排名
   * 对模型进行训练和评估，调整参数以优化模型性能。

**四、实验结果**

1. **数据预处理结果**：

**①提升数据质量**

**消除噪声**：去除无关数据或干扰信息，减少对分析的负面影响。

**纠正错误**：修复数据中的格式错误、逻辑矛盾或异常值。

**处理缺失值**：通过插补、删除或标记等方式合理处理缺失数据。

**统一标准**：确保数据格式（如日期、单位、编码）一致，避免后续解析错误。

**②增强模型性能**

**提高准确性**：干净的数据使模型学习真实规律，而非数据本身的噪声或偏差。

**加速收敛**：归一化/标准化后的数据使优化算法更易找到最优解。

**减少过拟合**：异常值处理和数据平滑可降低模型对噪声的敏感度。

**提升泛化能力**：增强的数据分布使模型在新数据上表现更稳定。

**③提高计算与分析效率**

**降低计算成本**：通过降维、采样或聚合减少数据量，加快训练速度。

**简化特征工程**：预处理后的数据更易于特征提取和转换。

**优化存储**：合理编码和压缩可减少存储空间占用。

1. **数据分析与挖掘结果**：

**①支持决策制定**

**数据驱动决策**：基于客观分析而非直觉，提高决策的科学性和准确性。

**趋势预测**：通过历史数据预测未来趋势（如销售、市场需求、用户行为）。

**风险评估**：识别潜在风险（如金融欺诈、供应链中断）并制定应对策略。

**②优化业务流程**

**发现效率瓶颈**：分析运营数据，找出流程中的低效环节。

**资源优化配置**：合理分配人力、物料、资金等资源，降低成本。

**自动化改进**：通过数据挖掘优化算法

1. **结果可视化展示**：

①**信息高效传递**

**降低理解门槛：**复杂数据通过图形化呈现，使非技术人员（如管理层、客户）能快速抓住核心信息

**突出关键发现：**通过视觉设计（颜色/大小/动效）强调重要结论，避免信息淹没在数字中

**讲好数据故事：**通过逻辑化的图表序列（如：现状→问题→解决方案）构建叙事链条

**② 团队协作增强**

**统一认知基准：**可视化报告确保各部门对数据解读一致

**远程协作：**云端看板支持多地团队同步分析

**知识沉淀：**可交互报告成为组织数据资产

**五、遇到的问题与解决方案**

1. **数据格式不统一问题**：
   * **问题描述**：在数据收集过程中，发现不同数据源的数据格式存在较大差异，导致数据整合困难。
   * **解决方案**：1. 建立统一的数据标准

**制定数据规范文档**：明确字段命名、数据类型、编码格式等标准

**创建数据字典**：定义核心字段的业务含义和取值范

**设计标准Schema**：建立目标数据模型，作为整合的基准框架

2. 数据预处理流水线

**ETL(抽取-转换-加载)流程**：

**抽取层**：保留原始数据，建立原始数据仓库

**转换层**：数据类型标准化

字段映射

单位统一转换

编码统一

**加载层**：将转换后数据加载到统一存储

1. **数据清洗复杂性问题**：
   * **问题描述**：数据清洗过程中，发现部分数据存在复杂的错误和异常值，难以通过简单规则去除。
   * **解决方案**：**1. 高级异常检测技术**

**机器学习方法**：

隔离森林（Isolation Forest）算法

局部离群因子（LOF）检测局部密度异常

一类SVM（One-Class SVM）用于无监督异常检测

自动编码器重构误差检测非常规模式

**时间序列专用方法**：

STL分解检测季节性异常

动态时间规整（DTW）检测形状异常

Prophet模型检测时间序列离群点

**2. 错误修正策略**

**基于模型的修正**：训练预测模型估计合理值（仅适用于可预测字段）

使用贝叶斯网络推断最可能的值

基于相似记录的k-NN插补

**交互式修正**：

开发可视化工具辅助人工审查

构建异常案例库

实现半自动化修正工作流

1. **模型性能波动问题**：
   * **问题描述**：在模型训练和评估过程中，发现模型性能存在波动，难以保持稳定。
   * **解决方案**：1. 数据层面优化

**数据质量检查**：确保数据清洗彻底，处理缺失值、异常值和重复数据

**数据分布分析**：检查训练集、验证集和测试集的数据分布是否一致

**数据增强**：对于小数据集，使用适当的数据增强技术增加多样性

**交叉验证**：采用k折交叉验证代替单一验证集，获得更稳定的性能评估

2. 模型训练优化

**学习率调整**：实现学习率衰减或使用自适应优化器(如Adam, RMSprop)

**早停机制**：监控验证集性能，在过拟合前停止训练

**正则化技术**：应用L1/L2正则化、Dropout或Batch Normalization

**模型集成**：使用Bagging或Boosting等集成方法减少方差

3. 超参数调优

**系统化搜索**：使用网格搜索、随机搜索或贝叶斯优化进行超参数调优

**参数冻结**：对于预训练模型，合理选择需要微调的层数

**六、总结与计划**

①项目总结

项目目标

基于 PySpark 实现电商商品的多维度排名分析（销售额、销售量、利润），为业务决策提供数据支持。

已实现功能

数据处理：读取 CSV 数据，计算商品的总销售额、总销售量和总利润。

排名分析：使用窗口函数对商品进行多维度排名。

结果展示：输出各维度排名前 N 的商品列表。

技术亮点

分布式计算：利用 Spark 处理大规模数据，支持横向扩展。

模块化设计：将排名逻辑封装为可复用的函数，提高代码可读性。

指标创新：通过 “销售额 - 运输成本” 计算实际利润，更精准反映商品盈利能力。

存在问题

数据局限性：仅分析单一 CSV 文件，未整合多源数据（如用户行为、库存等）。

可视化缺失：结果以文本形式展示，缺乏直观的数据可视化。

异常处理不足：未对数据缺失、格式错误等情况进行容错处理。

性能优化缺失：未针对大规模数据进行分区、缓存等优化。

② 短期计划（1-2 周）

功能增强

添加数据校验和异常处理，确保数据质量。支持参数化配置（如文件路径、排名字段、展示数量）。