BÀI THỰC HÀNH NYC YELLOW TAXI

Phân tích và Dự đoán Giá cước Taxi với PySpark

Mục lục

1 Tổng quan và Mục tiêu Học tập

1.1 Tổng quan

Notebook này hướng dẫn sinh viên thực hành xử lý và phân tích dữ liệu lớn bằng **PySpark** thông qua bộ dữ liệu *NYC Yellow Taxi*. Sinh viên sẽ học cách nạp, tiền xử lý, khám phá (EDA), xây dựng pipeline học máy, và đánh giá mô hình dự đoán giá cước.

1.2 Mục tiêu học tập

- Hiểu quy trình phân tích dữ liệu thực tế trên môi trường phân tán.
- Thực hành xử lý dữ liệu lớn với Spark DataFrame.
- Áp dụng pipeline Machine Learning trong Spark MLlib.
- Đánh giá và diễn giải kết quả mô hình hồi quy.

2 Cài đặt Môi trường

2.1 Thư viện cần thiết

Cài đặt các thư viện Python sau:

- `pyspark`: xử lý dữ liệu lớn và xây dựng pipeline ML.
- `pandas`, `numpy`: thao tác dữ liệu và tính toán ma trận.
- `matplotlib`, `seaborn`: trực quan hóa dữ liệu.
- `scikit-learn`: hỗ trợ đánh giá mô hình học máy.

2.2 Lệnh cài đặt

```
pip install pyspark seaborn matplotlib scikit-learn plotly
```

2.3 Yêu cầu hệ thống

- Python 3.8+.
- Kết nối Internet tới Google Colab
- Hoặc Spark chạy trên môi trường local hoặc cluster nhỏ.

3 Quy trình Phân tích Dữ liệu

3.1 Bước 1: Nạp dữ liệu

- File dữ liệu: `yellow_tripdata_2025-07.parquet`
- Lookup zone: `taxi_zone_lookup.csv`

Thực hiện:

```
# Đọc dữ liệu taxi
df = spark.read.parquet(DATA_PATH)
print(f" Loaded taxi data: {df.count():,} records")

# Đọc dữ liệu khu vực
zones = spark.read.option("header", True).option("inferSchema", True).csv(ZONES_PATH)
print(f" Loaded zones data: {zones.count()} zones")

# Kiểm tra cấu trúc
df.printSchema()
df.show(5, truncate=False)
```

Kết quả mong đợi: Xem được schema và 5 dòng dữ liệu đầu, đảm bảo các cột hợp lệ (thời gian, khoảng cách, phí...).

3.2 Bước 2: Tiền xử lý dữ liệu

1. Loại bỏ các bản ghi thiếu hoặc giá trị bất thường.
2. Giới hạn hợp lý: $0 < fare\_amount < 200$, $0 < trip\_distance < 100$.
3. Điền trung vị cho `passenger_count`.
4. Kiểm tra lại số lượng sau khi làm sạch.

3.3 Bước 3: Phân tích Khám phá Dữ liệu (EDA)

- Phân phối `fare_amount`, `trip_distance`.
- Quan hệ giữa `trip_distance` và `fare_amount`.
- Phân tích dữ liệu để tìm ra giờ cao điểm và khu vực nhiều hành khách nhất.

Ví dụ:

```
df.groupBy(hour("tpep_pickup_datetime").alias("hour")) \
  .count().orderBy("hour").show()
```

3.4 Bước 4: Xây dựng Pipeline Học máy

Mục tiêu: Dự đoán giá cước `fare_amount`. Xây dựng hai kịch bản :

1. **Kịch bản 1 :** Sử dụng một vài đặc trưng cơ bản.
2. **Kịch bản 2 :** Sử dụng tất cả các đặc trưng.

Ví dụ cho kịch bản 1 :

1. Tạo đặc trưng thời gian: `hour`, `day_of_week`, `is_weekend`.
2. Biến đổi định danh bằng `StringIndexer` và `OneHotEncoder`.
3. Chuẩn hóa dữ liệu với `StandardScaler`.
4. Huấn luyện mô hình `LinearRegression`.

3.5 Bước 5: Đánh giá Mô hình

- Chia dữ liệu train/test (80/20 hoặc tùy chỉnh).
- Tính các chỉ số: RMSE, MAE, R^2.

```
from pyspark.ml.evaluation import RegressionEvaluator
evaluator = RegressionEvaluator(labelCol="fare_amount", predictionCol="prediction")
rmse = evaluator.evaluate(predictions, {evaluator.metricName: "rmse"})
r2 = evaluator.evaluate(predictions, {evaluator.metricName: "r2"})
print("RMSE:", rmse, "R2:", r2)
```

4 Phân tích kết quả

- Mô hình đạt $R^2 \approx 0.9$ (kịch bản 1) và $R^2 \approx 0.99$ (kịch bản 2) cho thấy khả năng dự đoán của mô hình học máy tốt.
- Sai số RMSE phản ánh chênh lệch trung bình giữa dự đoán và thực tế.
- Các đặc trưng như `trip_distance`, `hour` có ảnh hưởng mạnh.

5 Kết luận và Gợi ý Mở rộng

5.1 Tóm tắt

Chúng ta đã hoàn thiện quy trình phân tích dữ liệu NYC Taxi:

1. Nạp và kiểm tra dữ liệu từ Parquet/CSV.
2. Làm sạch, phân tích EDA.
3. Xây dựng pipeline ML và huấn luyện mô hình hồi quy.
4. Đánh giá và diễn giải kết quả dự đoán.

5.2 Hướng phát triển

- Thử nghiệm các mô hình khác như Random Forest hoặc Gradient Boosting.
- Dùng mô hình cho dự đoán theo thời gian thực (Streaming).

5.3 Lưu ý đạo đức và pháp lý

- Dữ liệu NYC Taxi là công khai; cần tuân thủ chính sách chia sẻ dữ liệu của TLC.
- Không sử dụng dữ liệu hành khách cho mục đích thương mại trái phép.

Lộ trình thực hành gợi ý

1. Giới thiệu và thiết lập môi trường.
2. Nạp và kiểm tra dữ liệu.
3. Tiền xử lý và EDA.
4. Xây dựng pipeline ML.
5. Đánh giá mô hình và thảo luận kết quả.