**Project 2**

**Khaja Muneebuddin Adil**

1. **Finding and Analysing Data:**

The primary goal of this project is to optimize inventory management and enhance sales forecasting for an online retail platform. By leveraging the available data, the project aims to ensure that inventory levels align with customer demand, thereby reducing excess stock and stockouts, and ultimately improving customer satisfaction and profitability.

**Data Sources:**

To achieve the business objective, the following data sources from the dataset will be utilized:

* **Sales Transactions:**

**Customer Purchases:** The transaction\_id, timestamp, customer\_id, product\_id, product\_category, quantity, price, discount, and total\_amount columns will provide insights into purchasing behaviour, including peak purchasing times and popular product categories.

* **Customer Data:**

**Demographics:** The customer\_age, customer\_gender, and customer\_location columns will allow for segmenting customers based on demographics, helping to tailor marketing strategies and inventory decisions.

* **Product Data:**

**Product Details:** The product\_id, product\_category, price, and discount will help analyse product performance, evaluate the effectiveness of pricing strategies, and assess the impact of discounts on sales volume.

**Success Metrics:**

Success will be measured using the following metrics:

* **Reduced Stockouts:** Track the frequency and impact of stockouts on sales and customer satisfaction. Aim for a target reduction percentage based on historical data.
* **Improved Forecast Accuracy:** Analyse the accuracy of sales forecasts against actual sales data. Aiming for an accuracy percentage improvement based on historical forecasting errors.
* **Optimized Ordering:** Measure the reduction in excess inventory costs by analysing inventory turnover rates and aligning orders with forecasted demand.
* **Sales Growth:** Monitor overall sales growth by evaluating the total revenue generated over specific periods compared to previous periods.
* **Customer Satisfaction:** Use feedback and return rates to measure the satisfaction levels of customers regarding product availability and order fulfilment.

I got the dataset from Kaggle, and it comprises of 100,000 rows and includes the following columns:

* **transaction\_id (IntegerType)**: A unique identifier for each transaction.
* **timestamp (StringType)**: The date and time when the transaction occurred.
* **customer\_id (IntegerType)**: A unique identifier for each customer.
* **product\_id (IntegerType)**: A unique identifier for each product.
* **product\_category (StringType)**: The category to which the product belongs
* **quantity (IntegerType)**: The number of units purchased in the transaction.
* **price (FloatType)**: The price per unit of the product before any discounts.
* **discount (FloatType)**: The discount applied to the transaction as a percentage.
* **payment\_method (StringType)**: The method of payment used for the transaction
* **customer\_age (IntegerType)**: The age of the customer at the time of the transaction.
* **customer\_gender (StringType)**: The gender of the customer
* **customer\_location (StringType)**: The geographical location of the customer
* **total\_amount (FloatType)**: The total amount charged for the transaction, after applying discounts.

A screenshot of a computer

Description automatically generated

1. **Architectural Diagram:**

A diagram of data processing

Description automatically generated

1. **Data Pipeline Creation:**

**Data Ingestion:** Data was ingested from the on-premises SQL Server using Azure Data Factory (ADF) and a self-hosted integration runtime (IR) configured on my laptop. Once extracted, the data was copied to an Azure Data Lake Storage (ADLS) Gen2 container. I added a trigger in ADF to run the pipeline on a weekly basis.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. **Transformation and Analytics:**

I added a trigger in Azure Data Factory (ADF) to automate the pipeline execution on a weekly basis. For secure connectivity, I used an Azure service principal, stored in Azure Key Vault, to connect Azure Databricks to Azure Data Lake Storage (ADLS). By leveraging the Key Vault, Databricks was able to securely retrieve the service principal credentials, ensuring a seamless and secure connection to ADLS.

A computer screen with many icons

Description automatically generated

The data cleaning and transformation process was implemented using PySpark within an Azure Databricks environment. The logic was designed to ensure data consistency and prepare the dataset for analysis by addressing missing values and duplicates. I performed data cleansing by first identifying and handling missing values. Any rows containing null values in critical columns, such as customer\_id, product\_id, and total\_amount, were removed to ensure that all records used in the analysis were complete and reliable. Additionally, duplicates in the dataset were removed to eliminate redundancy and ensure data accuracy.

The dataset was checked for duplicate records based on customer transactions. All duplicate rows were dropped to maintain the integrity of the data, ensuring that each transaction was unique, and no duplicate insights were generated.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

As part of the data transformation process, I created two new columns to enhance the dataset and facilitate better analysis:

I introduced a new column called age\_group, which categorized customers based on their age into three groups, young (age <= 35), middle-aged (36 <= age <= 55) and senior (age > 55).

This segmentation allowed for more detailed demographic analysis, especially when comparing customer behaviour across different age groups.

A screenshot of a computer

Description automatically generated

A new total\_amount column was added, calculated as the product of quantity and price. This provided an aggregated value for each transaction, which was crucial for sales analysis, enabling insights into overall revenue generated per transaction.

A screenshot of a computer

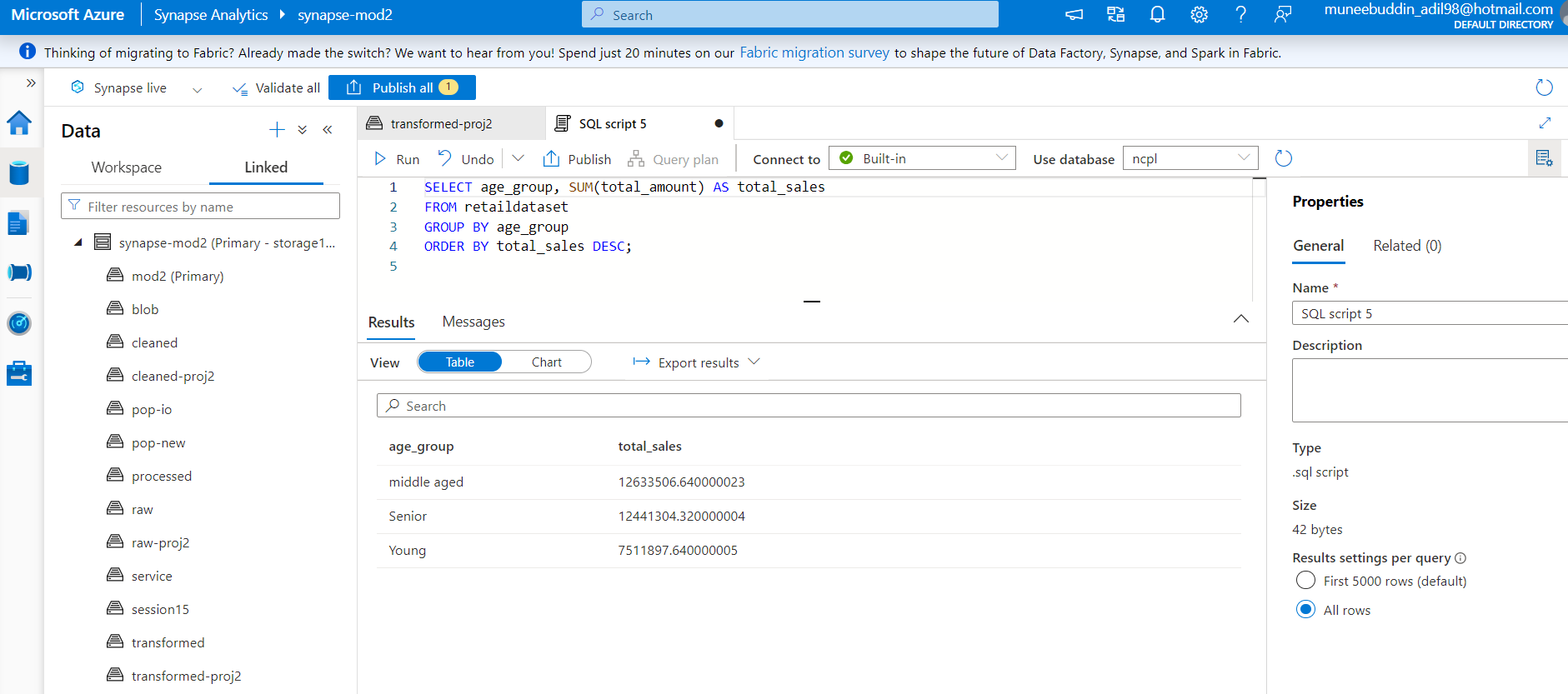
Description automatically generated

I utilized “when” function (imported from PySpark) to assign values to the new age\_group column. This approach was preferred over traditional methods because the when function offers a more concise and readable syntax for conditional operations. It allows for streamlined code when applying multiple conditions, making the transformation process more efficient and reducing the complexity that would arise from using longer, manual conditional logic.

After completing the data cleaning and transformation processes, the data was stored in Azure Data Lake Storage (ADLS) in Parquet format. Parquet was chosen for several reasons like efficient storage, optimized query performance and compatibility.

Azure Synapse Analytics was utilized to create external tables and perform SQL queries on the transformed dataset stored in Azure Data Lake Storage (ADLS). After establishing the external tables, various SQL queries were executed to derive meaningful insights from the data. The following queries were performed:

Total Sales by Age Group: This query calculated overall sales figures categorized by different age groups, providing insights into the purchasing behaviour of various demographics.



Average Quantity Sold per Product Category: This query determined the average quantity of products sold across different categories, helping to identify trends and popular product types.

A screenshot of a computer

Description automatically generated

Sales Distribution by Payment Method: This analysis highlighted the distribution of sales based on payment methods, offering insights into customer preferences.

A screenshot of a computer

Description automatically generated

Top 5 Products by Total Sales: This query identified the top five products with the highest sales, which is crucial for inventory and marketing strategies.

A screenshot of a computer

Description automatically generated

Connected Azure Synapse Analytics to Power BI to access the transformed dataset. This integration allowed for the creation of interactive dashboards and reports.

1. **Total Sales by Age Group and Gender**: A clustered bar chart displays the sum of total\_amount broken down by age group (Young, Middle-aged, Senior) and customer gender (Male, Female, Other). This visualization helps understand how different age groups and genders contribute to overall sales.
2. **Sales Distribution by Customer Location and Product Category**: A map visualization highlights sales across different customer locations (North America, Europe, Africa, Asia, etc.) combined with product category breakdown (Beauty, Books, Clothing, Electronics, etc). This offers a geographic view of product demand, allowing businesses to align inventory & marketing efforts with regional preferences.
3. **Top Customers by Total Sales:** A pie chart represents the sum of total\_amount by customer\_id, helping identify the top customers driving the most revenue.
4. **Total Sales and Discounts by Product Category and Gender**: A scatter plot visualizes the relationship between sum of total\_amount and sum of discount across different product categories and genders. This helps evaluate the effectiveness of discount strategies for various product lines.
5. **Overall Total Sales**: A gauge chart highlights the sum of total\_amount, showcasing overall sales performance against a set goal.
6. **Payment Method Distribution:** The payment method slicer, highlighted in orange, allows users to filter the dataset by payment methods (Credit Card, Debit Card, PayPal, and Gift Card), offering flexibility in exploring how different payment methods contribute to total sales.
7. **Filters for Interactive Insights**: The report includes slicers for both customer location and age group, enabling dynamic filtering to gain insights into specific demographics or regions. These filters, combined with the payment method slicer, allow users to drill down and analyse data according to their needs.

A screenshot of a computer

Description automatically generated

1. **Conclusion:**

This project encapsulates a comprehensive data engineering process designed to empower a retail company with valuable insights for optimizing business operations. By employing the Medallion Architecture, the data was systematically organized through various stages, ensuring a structured approach to data ingestion, quality assurance, and transformation.

Through meticulous data analysis and visualization, significant improvements in inventory management and sales forecasting were achieved. The insights derived from SQL queries executed in Azure Synapse Analytics provided a deeper understanding of customer behaviour, enabling informed decisions regarding stock levels, product offerings, and marketing strategies. By automating the data pipeline and utilizing Azure services for data processing and visualization, the retail company can now respond more agilely to market trends and customer preferences.