Data Cleaning Documentation

Overview

This document outlines the data cleaning process for the cohort analysis project. The primary goal was to ensure data integrity and prepare it for meaningful analysis.

Data Limitations & Transparency

- The original dataset consisted of a single retail sales table. To better demonstrate SQL skills, it was split into three separate tables (`customers`, `products`, `invoices`). This restructuring was done manually and may introduce slight differences comparing to the original format.
- `age` and `gender` data were manually added to enhance demographic analysis. These attributes were inferred where possible, but they were not part of the original dataset.
- Price variations for the same `stock_code` and `price_change_date` were observed. These differences may be due to discounts, bulk pricing, or other factors that were not explicitly recorded in the dataset. The approach taken was to retain the highest recorded price for consistency.

Key Cleaning Steps

Handling Missing Customer IDs

- Orders without `customer_id` cannot be grouped into cohorts.
- A view was created to store only invoices with valid `customer_id` for cohort analysis.
- The original table retains all records to allow for:
    - Total revenue analysis
    - Product sales tracking
    - Average order value calculation
    - Seasonal trend analysis and overall business performance tracking

Missing and Inconsistent Product Prices

- `unit_price` values were imputed where possible.
- Prices were determined by taking the **highest available price** per `stock_code` and `price_change_date`, assuming discounts may have caused discrepancies.
- 6 products remained without an imputed `unit_price`. These were left as `NULL` instead of being removed, to avoid losing valuable product data.

Negative Quantity Handling

- Negative `quantity` values were identified as potential returns.

- Matching purchase-return pairs (based on `stock_code`, `customer_id`, and date constraints) were removed to ensure only valid sales remain.
- Remaining negative quantities were replaced with the **average quantity** for that `stock_code` where possible.
- After replacements, only **215 negative quantity entries** remained and were deleted.

Duplicate Removal

- **5,429 duplicate rows** (identical in all columns) were removed from the invoices table.
- Inconsistent pricing within the same `stock_code` and `price_change_date` was addressed by keeping the highest price in the products table (lower prices could've been the result of discounts but since the data is limited, it wasn't possible to confirm that).

Final Adjustments

- A structured approach was used to clean data without compromising key business metrics.
- Views were created to enable accurate cohort tracking while retaining full revenue data.