Short Report

Title: Warehouse Inventory →(Data Cleaning for Warehouse Inventory Dataset)

Author: Riddhi Sharma

Objective:

This project explores a warehouse inventory dataset (`warehouse.csv`) to

(1) conduct data cleaning and EDA,

(2) test statistical hypotheses about price variation across categories,

(3) build a regression model to predict product price using available features.

The objective of this phase of the project was to clean and prepare the *Warehouse Inventory* dataset (warehouse.csv) for further analysis and machine learning tasks. Since raw datasets often contain inconsistencies, missing values, and structural issues, the goal was to transform the dataset into a consistent, accurate, and analysis-ready format. This ensures that subsequent steps such as exploratory data analysis, statistical testing, and predictive modeling are performed on reliable and meaningful data.

Dataset Source & Description:

The dataset used in this project is a locally provided CSV file named **warehouse.csv**, containing product-level details for items stored in multiple warehouses.

Key attributes include:

- Product ID, Product Name
- Category, Warehouse, Location
- Quantity, Price
- Supplier, Status
- Last Restocked date

The raw dataset contained **mixed-format numerical values**, **missing entries**, **inconsistent text formats**, and a small number of **duplicate rows**, making cleaning essential for accurate analysis.

Dataset Overview:

The *Warehouse Inventory* dataset (warehouse.csv) contains product-level information from multiple warehouse locations. It includes 1,000 rows and 10 columns in the raw form, with slightly fewer records after removing duplicates and outliers during preprocessing. The dataset captures key attributes such as product details, category, warehouse location, quantity in stock, supplier information, and pricing.

Structure of the Dataset

- **Rows:** ~1000 (slightly fewer after cleaning)
- **Columns:** 10

Features Included

- **Product ID –** Unique numeric identifier for each product
- **Product Name –** Name/label of the product
- **Category –** Product category (Electronics, Clothing, Toys, etc.)
- **Warehouse –** Warehouse location name
- **Location –** Specific aisle/section inside the warehouse
- **Quantity –** Stock quantity (numeric; originally mixed format)
- **Price –** Product price in numerical form
- **Supplier –** Supplier/vendor name
- **Status –** Availability status (In Stock / Out of Stock)
- **Last Restocked –** Date when the product was last restocked

Types of Variables

- **Numerical:**
  - *Quantity*, *Price*, *Days_Since_Restock* (engineered feature)
- **Categorical:**
  - *Product Name*, *Category*, *Warehouse*, *Location*, *Supplier*, *Status*
- **Date/Time:**
  - *Last Restocked* (converted to datetime format)

Methods Used for Data Cleaning

A series of preprocessing steps were applied to enhance data quality and ensure consistency. Each method was selected based on issues identified in the dataset.

1. Text Standardization

All text-based columns (Category, Supplier, Warehouse, Status, etc.) were stripped of extra whitespace and converted to a consistent title-case format.
Purpose: Prevents duplicate categories arising from variations like "electronics", "Electronics", and "ELECTRONICS".

2. Quantity Conversion (Word-to-Number)

Some Quantity values were provided as number words (e.g., *"two hundred"*, *"fifty"*). These were converted into numeric form using a combination of manual mapping and automated parsing.
Purpose: Ensures Quantity is fully numeric and suitable for calculations, summaries, and modeling.

3. Handling Missing Values

- **Quantity**: Filled with the median since the column is skewed.
- **Price**: First filled using category-wise median, then global median for remaining gaps.
- **Last Restocked**: Converted to datetime, with missing values treated as NaT (Not a Time).
  Purpose: Retains as many records as possible while preserving statistical properties.

4. Duplicate Removal

Exact duplicates and repeated Product IDs were identified and removed.
Purpose: Prevents double-counting of inventory and avoids bias in statistics and modeling.

5. Outlier Detection and Removal

Outliers in Price and Quantity were removed using the Interquartile Range (IQR) method.
Purpose: Prevents extreme values from distorting statistical summaries and model behaviour.

6. Encoding Categorical Features

Categorical variables such as Category, Supplier, Warehouse, and Status were converted to numeric form using One-Hot Encoding.
Purpose: Makes them usable in machine learning models without implying false ordering.

7. Normalizing Numeric Features

Numeric variables (Quantity, Days Since Restock) were standardized using StandardScaler.
Purpose: Ensures that all numeric variables contribute proportionally during model training.

Exploratory Data Analysis (key findings):

- **Quantity and Price** show **right-skewed distributions**, common in inventory datasets.
- Boxplots revealed **high-value outliers**, later removed using the IQR method.
- Category-wise patterns show that **certain categories and suppliers consistently have higher median prices**.
- **Quantity and Price have low–moderate correlation**, indicating price is not strongly influenced by stock levels.
- **Visualizations** (histograms, boxplots, scatterplots) helped uncover variation, outliers, and category-driven trends.

Statistical Testing:

- Conducted a **one-way ANOVA** to test whether mean price varies across categories.
- **Null hypothesis**: All categories have the same average price.
- If p-value < 0.05 → **Reject the null**, meaning category significantly affects price.
- Confidence intervals for each category supported differences observed during EDA.

Modeling:

- **Model:** -
- Applied **Linear Regression** to predict Price using:
  - Quantity
  - Days Since Restock
  - Encoded categorical features (Category, Warehouse, Supplier, Status)
- Evaluated model using **R^2, RMSE, and MAE**.
- Residual analysis checked linear regression assumptions.
- Moderate performance indicated that **current features explain only part of price variation**.
- **Low R^2** suggests important factors like brand, demand, or product specifications are missing.

Limitations & Future Work:

- Missing potentially important features (brand, product specifications, demand signals).
- Price may be non-linearly related to features — consider tree-based models (Random Forest, XGBoost).
- Temporal dynamics: restock frequency and seasonality not fully modeled; time-series features could help.
- Better handling of textual product attributes using NLP could improve price modeling.

Key Takeaways:

- Price varies strongly by **product category**, confirmed by both EDA and ANOVA.
- Linear Regression captured some patterns but **only partially explains price variation** with current features.
- Improving model performance requires **richer product metadata**, historical trends, and more advanced algorithms.
- The cleaned dataset provides a **solid foundation for deeper analytics**, enabling reliable exploration and modeling.