Predicting Shipping Times and Profit Trends: A Data-Driven Approach

Members:
Ozafa Mahmood (oym)
Daniel Surina (dsa108)

Table of Contents

Problem Scope & Background

In the interconnected landscape of a global economy the seamless transfer of goods forms the backbone of modern commerce and trade. For businesses being able to use data to predict variables which are critical is not only required for success but- is a necessity for the survival of the businesses. This project aims to harness the power of data to answer pivotal business questions for a hypothetical company navigating the complexities of shipping logistics and sales dynamics. By analyzing a dataset of over 5 million records, we leverage our Data Science expertise to uncover actionable insights that could drive informed decision-making. Our Analysis is driven by two core questions:

Q1: Can we Predict how many days it will take to ship items based on features such as order priority, product type and region.
Q2: Does unit price have an impact on Profit?

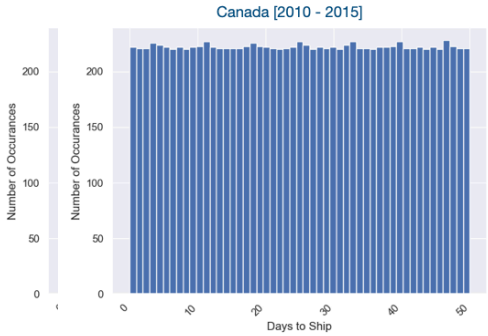
Through this analytical journey we aim to provide a detailed description of our exploratory data analysis journey and provide a peak into the world of data analysis for real world application. Our work involves various graphical representations to identify prevailing trends and patterns in the data. By applying appropriate statistical tests and predictive models, we seek to demonstrate the practical application of data science techniques to real-world challenges.

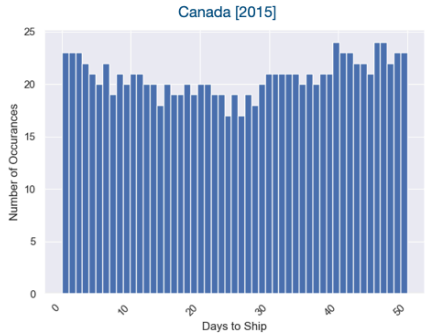
Data Cleaning & Exploration

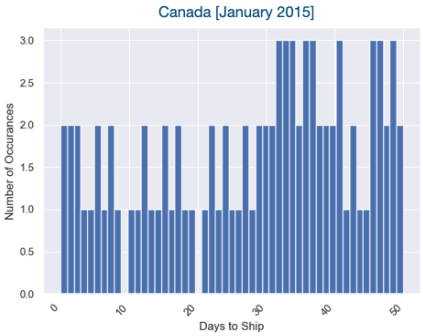
Data was found on excelbianalytics.com along with other similar datasets of various sizes ranging from 100 records to 5,000,000 records. We had a hard time finding the right data for this problem because there weren't many sources that would provide us with the data needed for this task. We chose the 5,000,000 dataset because more data gives us a better overall sample and hopefully a better model as a result. There was a mention that the data was generated using Visual Basic, so this immediately makes us realize that the data is pre-generated and there is a good chance that we would see some repeating pattern of the functions creating the values at some point, so this is one thing we have to keep in mind throughout the whole analysis.

We started off by simply analyzing the data by looking at the structure, from that we were able to find that the dataset contains following variables:
Region, **Country**, **Item Type**, **Sales Channel**, **Order Priority**, **Order Date**, **Order ID**. **Ship Date**, **Units Sold**, **Unit Price**, **Unit Cost**, **Total Revenue**, **Total cost**, **Total Profit**

Following this we have generated plots plotting various variables, one of the first plots was days to ship and how many occurrences there were in the data, to see roughly what the distributions look like. Through this plot we could see that there is a uniform distribution of the data no matter how small or big the timeframe potentially hinting a few things; it could mean that the data we analyze might not be that great to use for model training since everything is uniformly distributed, it might hint that the VBS script that generated it might not be as sophisticated and just generated random noise that has no underlying patterns inside of it, either way the graphs look quite uniform. While doing this we have realized that there are some features we would benefit from, especially in this step. We needed to get the number of days to ship the product but we only have Order Date and Ship Date, so by doing simple subtraction we have added one more column to the data labelled Days to Ship to simplify our life, but more on that below. (Look at graphs down below)







ETL:

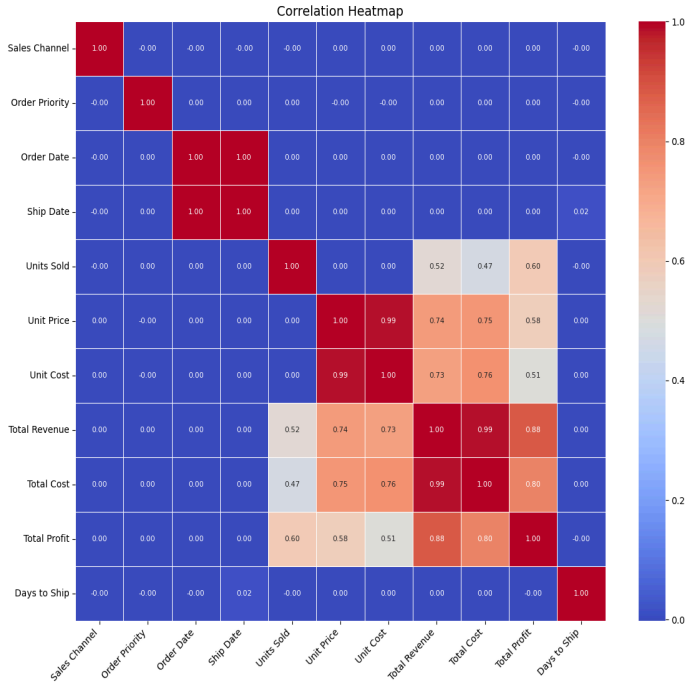
In most cases we would do the ETL step after the data exploration but as mentioned above, we needed some additional features in order to do the data exploration efficiently, so we have created a pipeline function cleanAndSort data that takes care of:

- **Dropping N/A values**
- **Dropping Duplicates** there were almost 2.3 million duplicate rows out of 5 million total
- **Dropping Order ID** this column was irrelevant to our calculations, would just confuse the model, we did this after dropping duplicate rows because there might be 2 different orders ordering the same item in the same quantity and date
- **Label Encoding**:
  1. Encoded the categories Low, Medium, High, Critical to 1, 2, 3, 4 respectively the reason for choosing Label encoding was so we could maintain the ordinality of this category as priority levels follow a natural order
  2. Encoded the Country Column with label encoding despite it being nominal data. Initially, one hot encoding was considered but this approach was impractical due to the high cardinality of the column due to 185 unique countries. Upon further research we found out that because Random Forest is not sensitive to ordinal nature of Label-encoded features and treats these values as distinct categories during splits this was the more computationally efficient choice.
- **One-Hot Encoding**:
  1. We Binary one-hot encoded the Sales Channel column ensuring that the categories are treated completely distinctly without any implicit order
  2. Item type had 12 distinct categories which was also one hot encoded into 12 new binary columns
  3. Containing 7 unique categories we one-hot encoded the region column as well given we were made aware of origin of shipping and therefore wanted to have all regions being treated as individual distinct categories with no implicit order

Correlation Analysis: First, we will lay down a heatmap to display the correlation coefficients between variables in the dataset. In this report, we will explain how the heatmap is formatted, and then we will dig into the numbers to analyze what the values in the heatmap signify:

Each square in the heatmap shows the correlation between the variables on the x and y axis. The color scale on the right indicates the strength and direction of the correlation:
○ Red tones indicate a positive correlation. Blue tones indicate a negative correlation.
○ The intensity of the color indicates the strength of the correlation (lighter colors are weaker and deeper colors are stronger).

As discussed above, since the data is artificially created there, we observe patterns that deviate from real-world datasets. For example, we can see that order priority in the heatmap has zero correlation to Days to ship. However, the heatmap still reveals some meaningful relationships between total revenue and unit price and unit sold which we will further explore in **Q2.**



Methodology:

For this project, we used VS Code as our primary programming environment. Utilizing Pythons wide range of libraries tailored for data science and machine learning. Pandas and NumPy were our go-to tools for organizing and preprocessing the data, making it easier to clean, analyze, and transform. For visualizations, we relied on Matplotlib and Seaborn to create clear, insightful plots which helped represent our data concisely. Most of the code is well documented in the README.md function, and the code is well commented so feel free to have a look at it.

When it came to building statistical models and running detailed analyses, we used the Stats models, Scikit-learn as our main drivers. Our approach included a mix of statistical and machine learning methods. We used Ordinary Least Squares (OLS) regression to uncover relationships within the dataset and used Random Forest models to handle predictive tasks. Correlation analysis also played an important role in identifying how variables interacted, helping us answer questions about shipping times, priorities, and profitability.

Question 1: Can we Predict how many days it will take to ship items.

Given our prior knowledge on the Data set it was highly unlikely for us to find a relationship between our features and the target variable Days to Ship regardless we wanted to explore whether we could build a model capable of getting a test score of greater than 1/50 (2%) since the days to ship ranges from 0 – 50.

Model Selection:
We decided to choose random forest classifier for this task for several reasons.

- Handles both label and one-hot encoded features well allowing us to utilize all of our features
- Effective in handling uniformly distributed data

Features and Target:
For our Feature set X we included the following:

X = Sales Channel, Order Priority, Country, Units Sold, Item Type, Region, Unit Price, Unit Cost, Total Revenue, Total Cost, Total Profit

All these features were either already numeric or were either **label encoded or one-hot encoded**, depending on whether they were nominal or ordinal.

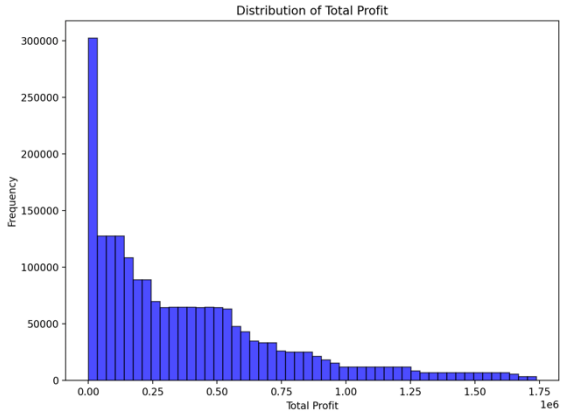
Our Target variable Y = Days to ship.
Initial Challenges:
Initially our results were discouraging, as we got a negative test score of -32% this was due to two main reasons we were over fitting our data due to incorrect parameters and secondly we were running a Random Forest Regressor which was predicting numeric values of Days to ship while we required Nominal predictions of Days.
Results:
After fine tuning our parameters, we got the following results

- **Train score = 15%**

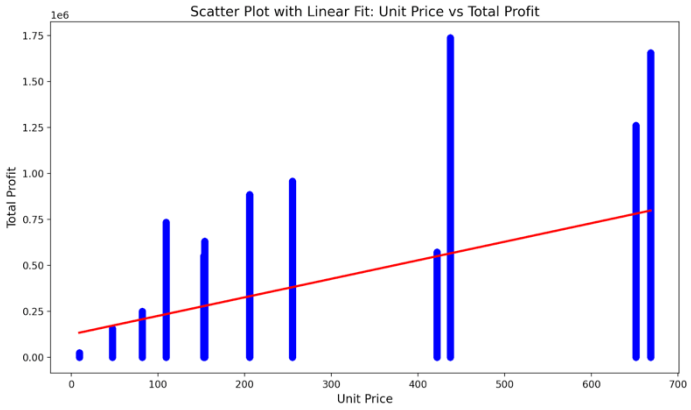
- **Test score = 1.9%**

Initially this looked like a failure. However since the data was normally distribution with days to ship having zero correlation with any features this result does make sense. The model's performance reflects the limitations of the data set rather than the model itself. Hence, with this data set it is impossible to predict days to ship.

Question 2 Does unit price have an impact on Profit?



To answer our question, we began by analyzing the distribution of unit prices. The data reveals a relatively uniform distribution across most price ranges, with a notable exception: unit prices between 150 and 160 have nearly double the frequency compared to other ranges. This suggests a possible bias or concentration of products within this range, potentially influencing overall trends. Next, we plotted total profit, visualized through its distribution [figure 2]. The right-skewed nature of the plot indicates that most products generate lower profits, while only a few contribute significantly higher profits. To investigate whether higher unit price products correspond to these higher-profit outliers, we plotted a scatter plot of unit price versus total profit and overlaid a linear fit to identify trends [figure 3]. The positive slope of the fitted line clearly indicates a linear relationship: as unit price increases, total profit tends to rise. This finding suggests that products with higher unit prices are generally more profitable, likely due to larger margins or other contributing factors.



To further validate this relationship, we ran an Ordinary Least Squares (OLS) regression after confirming the relation and seeing that the residuals are normally distributed. The results confirmed the statistical significance of the relationship between unit price and total profit.

- With a p-value less than 0.05, allowing us to reject the null hypothesis that the coefficient is zero. This reinforces our earlier observation: unit price has a measurable impact on total profit.
- The R-squared value for the model is 0.333, indicating that unit price explains about 33% of the variability in total profit.

In conclusion, our analysis highlights a positive and statistically significant relationship between unit price and total profit, and it would be safe to say that unit price does have a positive impact on profit. Based on this data it would be worthwhile to consider investing in higher-priced products, as they tend to offer greater profitability. However, further investigation into other contributing factors is recommended to gain a more comprehensive understanding of what the profit is impacted by.

Conclusion:

In this project we explored an Artificially generated data set of Shipping using various data science methodologies to address key business questions in shipping logistics and profitability. Although we encountered various limitations of the data set we still managed to use various statistical and ML tools such as Linear regression, OLS, Correlation matrix and Random Forest Regressor to try to come to meaningful conclusions to our first 2 starting questions. With the conclusions as follows

Q1: The dataset's uniformly distributed nature and lack of inherent relationships made it impossible to reliably predict how many days it will take to ship items

Q2: We found a statistically significant relationship between unit price and profitability suggesting as unit price increases so does profitability.

Project Experience Summary

Ozafa Mahmood:

- Built and deployed predictive models using Random Forest Classifier and statistical tools like Ordinary Least Squares (OLS) regression to analyze shipping logistics and profitability trends, effectively handling a dataset of over 5 million records.
- Engineered and preprocessed the dataset by implementing techniques such as label encoding, one-hot encoding, and feature generation (e.g., calculating "Days to Ship" from order and ship dates), ensuring the data was suitable for machine learning models.
- Performed exploratory data analysis (EDA) using Pandas, Matplotlib, and Seaborn to uncover insights such as the uniform distribution of shipping times and correlations between unit price and profit, enabling actionable recommendations.
- Optimized Random Forest hyperparameters (e.g., n_estimators, max_depth, min_samples_leaf) through iterative experimentation, improving test accuracy from an initial -32% to 1.9%, consistent with theoretical expectations for uniform data.
- Validated business insights by identifying a statistically significant relationship between unit price and profitability, confirmed through regression analysis and residual checks, demonstrating unit price's role in contributing to a 33% variation in profit.

Daniel Surina:

- Conducted comprehensive data analysis on a large-scale sales dataset containing approximately five million records, utilizing Pandas, scikit-learn, and a variety of statistical techniques to thoroughly examine and interpret the data.
- Implemented advanced predictive models, including RandomForestClassifier, KNeighborsClassifier, and SVC, to accurately estimate product shipping times based on key predictive features.
- Employed diverse visualization methods and rigorous statistical testing to uncover underlying trends, patterns, and insights, thereby enhancing the depth and quality of the overall analysis.