DATA_ENG 300_HW1_Aircraft
Marcus Van Mieghem
Apr 17, 2025

Note to grader: I am deeply interested in the aviation industry and will be working at Republic Airways this summer. Because of this background, I may have used some industry-specific terminology or gone into greater detail in certain sections of the assignment. I spoke with Professor Chan, and he advised me to include this note for context.

Task 0: Importing the aircraft inventory dataset

After importing the aircraft dataset, I reviewed the columns and observed that several contained numerous null values. This was helpful context for approaching the cleaning steps later in the assignment.

Task 1: Investigating missing data in the dataset

CARRIER column: This field represents the IATA-assigned carrier code (e.g., UA = United Airlines). The only carrier with missing data was North American Airlines. All such rows had "North American Airlines" listed under CARRIER_NAME, but the CARRIER field was blank. Since IATA assigns "NA" to this airline, Python may have interpreted it as NaN. I assigned "NA" to the CARRIER and UNIQUE_CARRIER fields where applicable. After this fix, no missing values remained in this column.

CARRIER_NAME column: Upon inspection some of the rows with missing CARRIER_NAME had CARRIER OH, I researched this and discovered that those aircrafts were operated by PSA Airlines Inc.. So I assigned the missing rows all the values from complete PSA Airlines Inc. rows which included CARRIER_NAME, UNIQUE_CARRIER_NAME, AIRLINE_ID and UNIQUE_CARRIER. The other rows with missing CARRIER_NAME were ones with CARRIER code L4. The code L4 corresponds to Lynx Aviation d/b/a Frontier Airlines, which has a CARRIER_ID of 21217.0 and a UNIQUE_CARRIER of L3. I assigned the missing carrier rows the values used in the other rows were complete and aligned with L4. After completing these steps, there were not any rows with missing values.

MANUFACTURE_YEAR column: Before investigating, I set all rows with MANUFACTURE_YEAR of 0 to nan or missing. This is because a manufacturing year of 0 does not make sense. To fill in these missing values, I grouped the dataset by the MODEL column and imputed the missing MANUFACTURE_YEAR values using the median year for each respective aircraft model. This approach ensures that the imputed year is based on similar aircraft and is not skewed by outliers, as might occur if using the mean. In doing so, I am assuming that the year of manufacture can be reasonably estimated based on other aircraft of the same model. While this is a practical solution for general analysis, it's important to be cautious when using the MANUFACTURE_YEAR column for more sensitive applications, such as calculating aircraft age, warranty eligibility, depreciation, or compliance with maintenance or regulatory schedules, since the imputed values may not reflect the exact production year of the individual aircraft. After cleaning there were still 12 rows with a missing manufacture year. This

is due to them being unique planes in the dataset with no other complete rows to get a manufacture year from. **This imputation step is most effective after cleaning and standardizing the MODEL column, since inconsistent or overly specific model names (e.g., B737-8JP vs. B737) can prevent accurate grouping and reduce the effectiveness of the median calculation. Given this, this step was run later in the analysis after cleaning the MODEL column.**

NUMBER_OF_SEATS column: Upon inspection, there were 7 rows which had missing values. When looking at the rows, I saw that they are all cargo planes. This is seen in the model column. They are all 767 cargo jets. Cargo jets do not have seats (only 4 for pilots), as they transport cargo not humans. Given this, I set the number of seats to 0 for these rows, consistent with other rows which are cargo (FedEx). After completing these steps, there were not any rows with missing values.

CAPACITY_IN_POUNDS column: Some aircrafts in the dataset have a CAPACITY_IN_POUNDS value of 0, which is not realistic. To address this, I identified rows with zero capacity or missing values and, for each one, located other rows with the same MODEL. I then used the median capacity from those matching rows to impute a more reasonable value. However, some entries still remained with zero capacity. These correspond to aircraft with unique or uncommon model names that do not have other entries in the dataset with a valid (non-zero) capacity to reference. These cases may require manual review or external data sources to resolve. **This imputation step is most effective after cleaning and standardizing the MODEL column, since inconsistent or overly specific model names (e.g., B737-8JP vs. B737) can prevent accurate grouping and reduce the effectiveness of the median calculation. Given this, this step was run later in the analysis after cleaning the MODEL column.**

AIRLINE_ID column: Upon inspection, the only rows with missing AIRLINE_ID are the ones which are PSA Airlines, we can search for PSA Airlines AIRLINE_ID in other rows and copy to the missing row. After completing these steps, there were not any rows with missing values.

Task 2: Transformation/Standardization Investigation

In this part of the assignment, I investigated 4 columns: MANUFACTURER, MODEL. OPERATING_STATUS and AIRCRAFT_STATUS. Here is a in depth overview:

MANUFACTURER column: For this column, I first stripped whitespace and converted all values to lowercase for consistency. I then focused on cleaning the major manufacturers (such as McDonnell, Boeing, Airbus, Gulfstream, Learjet, Saab, and Beechcraft) using regex patterns. I searched for any strings that contained variations of these names and standardized them to a single, correct label. For example, entries like ‘theboeingco’, ‘boeingco’, and ‘boeingcompany’ were all converted to just boeing. Unfortunately, the dataset was quite messy, and several entries in this column were not manufacturers at all — some were airlines, like easyjet/investec. I wasn’t able to find a reliable way to handle these programmatically. If I had more time, I would have explored developing an algorithm to query FAA.gov using tail numbers and manufacturing years to identify the correct manufacturer, but that was beyond the scope of this assignment. Before cleaning, the column contained 184 unique manufacturers. After cleaning, this number was reduced to 97 — a significant improvement in standardization.

MODEL column: For this column, I followed a similar approach to the MANUFACTURER column, using regex to clean and standardize the values. One issue I encountered was that 706 rows had a value of 0 for the MODEL, which is not valid. I treated these as missing and set them to NaN. As with the MANUFACTURER column, if it had been within the scope of the assignment, I would have developed an algorithm to query FAA.gov using tail numbers to retrieve the correct model. Before cleaning, the column contained 1,341 unique model entries. After cleaning, this number was reduced to 493, a significant improvement in consistency and usability. As part of my regex pattern, I chose to remove the last three characters in aircraft models like B737-4GX, since these suffixes typically represent differences in engine types, configurations, and production batches, which were not relevant to the scope of this assignment. **Once the MODEL column was cleaned, I then ran the transformations described above for imputing CAPACITY_IN_POUNDS and MANUFACTURE_YEAR. These steps rely on grouping by aircraft model, and would only be effective after standardizing the MODEL values.**

OPERATING_STATUS column: This column represents if an aircraft in the dataset is currently operating day to day. Before cleaning, the unique values in this column were 'Y', 'N', 'y', and a blank entry. It was clear that the lowercase 'y' simply needed to be standardized to 'Y'. For the one blank value, I looked up the aircraft on FAA.gov using its tail number and confirmed that it is still in operation with the same airline. Based on this information, I set that entry to 'Y' as well. After cleaning, the column contained 126649 rows marked as 'Y' (operating) and 5664 rows marked as 'N' (not operating).

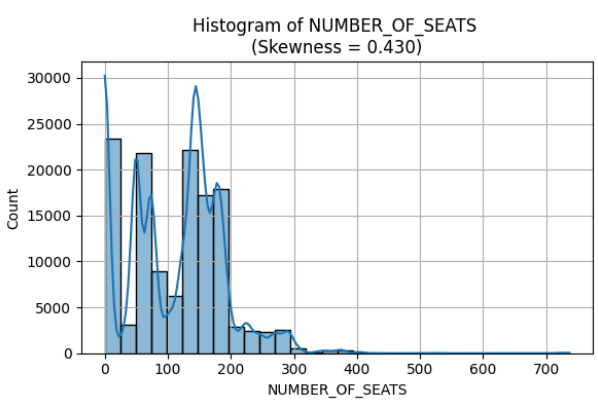
AIRCRAFT_STATUS column: This column records whether an aircraft is under a capital lease, operating lease, or is owned. According to the database's documentation, the valid values should be: 'O' for Owned, 'B' for Operating Lease, and 'A' for Capital Lease. Some entries were in lowercase, so I standardized all values by converting them to uppercase. Additionally, there were 122 rows with the value 'L', which was not documented. I found that all of these rows were from the year 2022 and operated by United Airlines. After investigating these aircraft on the FAA website, I determined that they were leased. To maintain consistency with the rest of the dataset, I converted 'L' to 'A', treating them as capital leases. After cleaning, the column contained 79,506 'O' values, 43,551 'B' values, and 9,256 'A' values.

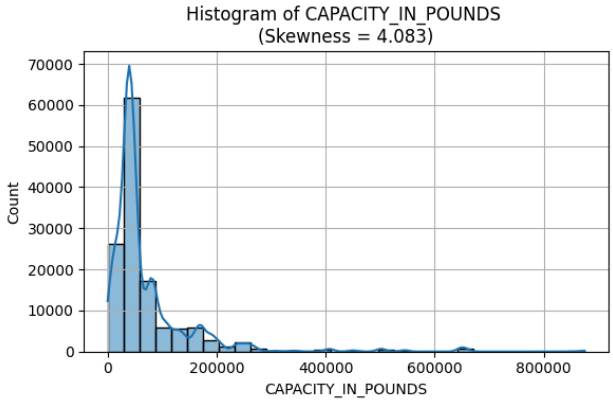
Task 3: Removing rows with missing values.

For this part of the assignment, I only removed rows with missing values in variables that are directly used later in the analysis. These include NUMBER_OF_SEATS, CAPACITY_IN_POUNDS, OPERATING_STATUS, and AIRCRAFT_STATUS. After filtering 132195 rows were retained, preserving approximately 99.91% of the original dataset. It's important to note that missing data still exists in other columns, but since those variables are not required for the current analysis, those rows were left untouched. Depending on the goals of future analyses, particularly those involving modeling, forecasting, or compliance, it may be necessary to drop or impute all missing values to ensure completeness and reliability.

Task 4: Transformation and derivative variables

We can investigate the skew in the variables NUMBER_OF_SEATS and CAPACITY_IN_POUNDS. Checking the skew and plotting a histogram for NUMBER_OF SEATS variable as well as CAPACITY_IN_POUNDS:



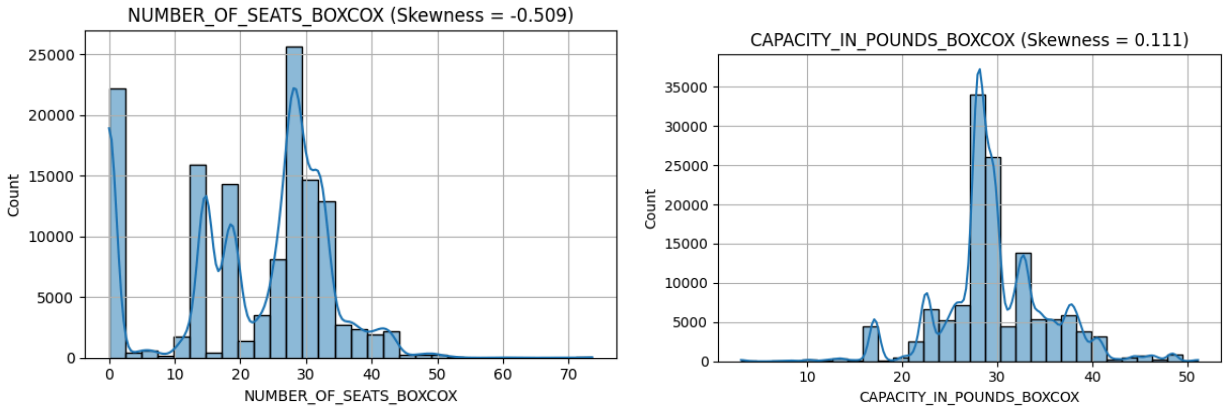


Looking at the plots and skewness values, both variables are right-skewed, as indicated by their positive skewness. This can also be seen in the histograms, both distributions have a long tail to the right, suggesting that most values are relatively low with a few much larger outliers. When comparing the two variables, CAPACITY_IN_POUNDS exhibits a noticeably higher skew than NUMBER_OF_SEATS. This could be due to the fact that seat configurations vary more widely across different aircraft models and airlines — for example, depending on how cabins are structured or segmented. Weight capacity on the other hand is more tightly constrained by engineering requirements and safety regulations, resulting in fewer extremely low or high values. Also, 0 values exist for the number of seats as cargo planes have 0 seats but no plane can have zero weight capacity.

Boxcox transformation

To make the variables more ‘normal-like’ we can apply a boxcox transformation. In order to apply the BoxCox transformation to NUMBER_OF_SEATS and CAPACITY_IN_POUNDS, all values must be positive, therefore I added 1 to all the NUMBER_OF_SEATS rows as cargo planes have 0, but 0 is not positive. Once this is done I can subtract 1 again. After applying the BoxCox transformation we can again look at the histograms of each variable. We expect the variables to be more normally distributed after applying the Box-Cox transformation.

Histograms after applying the boxcox transformation:

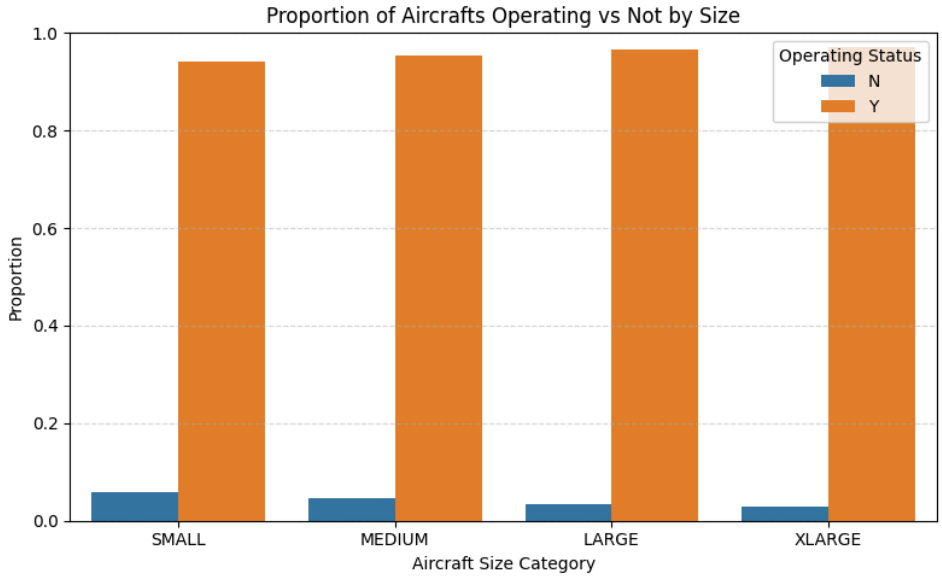


Observing before and after the boxcox transformation of NUMBER_OF_SEATS, the data looks more spread out, although the skew has changed from right skew to left skew(0.43 to -0.509). The data does not look normal although the distribution did become a little more symmetric overall. For CAPACITY_IN_POUNDS, the transformation was clearly more effective. Skewness dropped significantly from 4.08 to 0.11, indicating a much more symmetric distribution. This improvement is also evident in the histogram of the transformed variable, which now appears bell-shaped and much closer to a normal distribution compared to the heavily right-skewed shape prior to transformation.

Task 5: Transformation and derivative variables

We can investigate the proportion of aircrafts operating or not by size. Aircraft size is determined by quartiles based on the number of seats that the aircraft has.

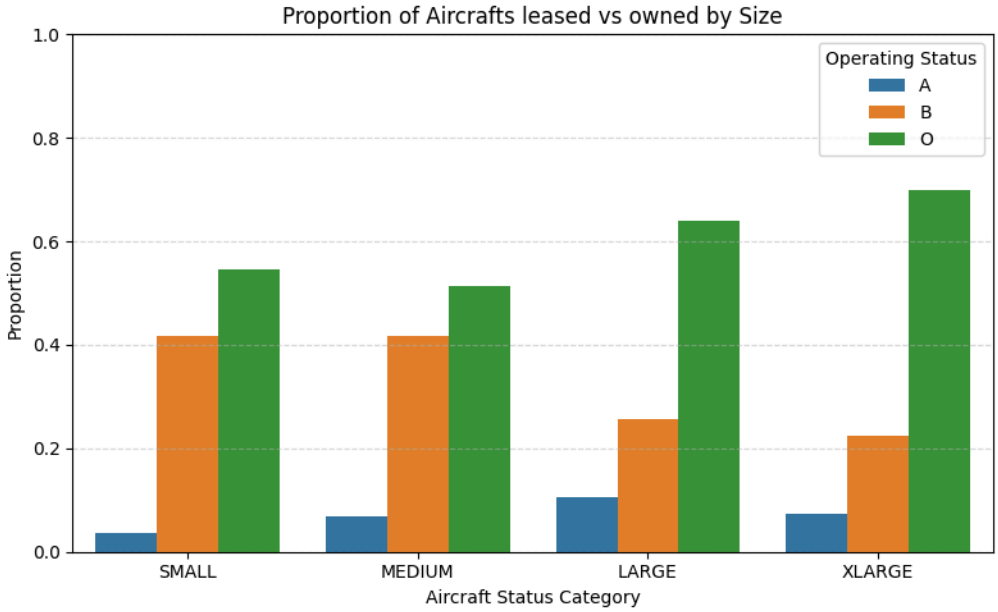
Chart outputted visualizing proportions:



	SIZE	OPERATING_STATUS	count	proportion
0	SMALL	N	2379	0.057845
1	SMALL	Y	38748	0.942155
2	MEDIUM	N	1147	0.045699
3	MEDIUM	Y	23952	0.954301
4	LARGE	N	1180	0.034206
5	LARGE	Y	33317	0.965794
6	XLARGE	N	898	0.028533
7	XLARGE	Y	30574	0.971467

Based on the proportions, we can see that the majority of aircrafts across all size categories are operating. While the overall differences in operating status by size are relatively small, there is a slight decreasing trend in the proportion of non-operating aircraft as size increases. This means that larger aircrafts within this dataset are slightly more likely to be operating than smaller ones.

We can also investigate the proportion of aircrafts leased (capital or operating) or owned by size. Aircraft size is determined by quartiles based on the number of seats that the aircraft has. Once again, 'O' for Owned, 'B' for Operating Lease, and 'A' for Capital Lease.
Chart outputted visualizing proportions:



SIZE	AIRCRAFT_STATUS	count	proportion
SMALL	A	1544	0.037542
SMALL	B	17130	0.416515
SMALL	O	22453	0.545943
MEDIUM	A	1730	0.068927
MEDIUM	B	10476	0.417387
MEDIUM	O	12893	0.513686
LARGE	A	3630	0.105227
LARGE	B	8807	0.255298
LARGE	O	22060	0.639476
XLARGE	A	2352	0.074733
XLARGE	B	7088	0.225216
XLARGE	O	22032	0.700051

Based on the histogram and proportion data, smaller aircraft are more commonly leased than owned. As aircraft size increases, the proportion of owned aircraft rises significantly. In the extra-large category, approximately 70% of aircraft are owned, compared to only 55% in the small aircraft group. Additionally, the proportion of aircraft under operating leases consistently decreases with increasing size. Capital leases are the least common ownership type across all size categories. This trend aligns with expectations: larger aircraft typically represent long-term investments made by major airlines or cargo operators with substantial capital. In contrast, regional or smaller carriers operating smaller planes may not have the financial resources to purchase aircraft outright and are therefore more likely to lease.

Generative AI disclosure

I used Generative Artificial Intelligence (ChatGPT) to support parts of the coding for this assignment. Specifically, I used ChatGPT to:

- Create advanced regex patterns when cleaning up the MODEL column in the dataset.
    - I attached a photo of the unique values in the MODEL column.
    - Prompt: 'Can you write a code to get rid of all endings on models, such as CARGO, PASSENGER,COMBI, etc.'
- Create group-based median imputation logic for the CAPACITY_IN_POUNDS and MANUFACTURE_YEAR columns using .groupby().transform().
    - I originally used a for loop but it did not work very well, I put that into chat and asked if there is a better way to do it.
- Create histogram in task 4: Checking skewness in variables NUMBER_OF_SEATS and CAPACITY_IN_POUNDS
    - After calculating skewness values for NUMBER_OF_SEATS and CAPACITY_IN_POUNDS, I asked ChatGPT how to plot histograms to visualize the distribution of the variables. Python plotting can be lots of lines, and Ai helps streamline this process.
- Creating proportion tables and plots in task 5
    - I did not know how to generate a proportion table to compare categories like OPERATING_STATUS and AIRCRAFT_STATUS by aircraft size. I asked ChatGPT how to compute proportions grouped by a category and visualize them using a bar chart. It helped me generate both the grouped summaries and corresponding plots.