Q1

First, the function .unique() was performed on all columns to see all unique values in each column. We will firstly analyze the North American Airlines, as prompted in the homework. It appears that the columns for carrier and unique carrier are missing; however because we do know that they are North American Airlines, we know that they are not actually missing. We can see that the carrier column is called “NA” for other data points for North American Airlines, so we can impute that for the rest of the carrier column. Because the unique carrier column is NaN for all of the North American Airlines entries, we can assume that we can also impute North American Airlines for this column as well. After further inspection, it appears that the “CARRIER” column’s NaN values only apply for North American Airlines. From the unique values of this column, the missing values only arise from a “NaN” value, so we can be confident only North American Airlines have NaN values in the “CARRIER” column. We can impute these with “NA.”

Moving onto “CARRIER_NAME,” when inspecting the column, the L4 and OH carriers are the only one with a missing “CARRIER_NAME.” Because there is no unique identification that would identify the carrier name, we cannot impute in this case.

For the “MANUFACTURE_YEAR” column, we know that the data is greater than 2025 and less than at least 1700. Most of the reasonable data was in the 1900s/200s. We also look for NaN data that we saw when we swept for unique data points. Because the invalid data for this constitutes about 0.04% of the dataset, we can call this insignificant and decide to drop this data.

For the “NUMBER_OF_SEATS” data, an initial glance at the data lets us know that the unusual data are the “0” data and “None”/”NaN” data. After inspecting planes with 0 seats, we see a lot of entries. This makes more sense after realizing that these were United Parcel Services flights, meaning that they are used to transport luggage. However, there are Amerijet International planes that have NaN seats. However, we do have knowledge that there might be other planes with the same aircraft type. After further evaluation, it appears that this aircraft type can be used in many different ways, especially in United Parcel Services. This cannot be reliably used to tell seat numbers. We can try to impute these data points with mean, median, KNN, and PMM imputation. After trying them all and finding the RMSE, the KNN imputation provides the lowest RMSE, so we will use KNN imputation for this.

We do something very similar for “CAPACITY_IN_POUNDS.” There is a substantial amount of NaN data, about 100 rows with missing data. We then also test for which form of imputation would be the best with mean, median, KNN, and PMM imputation. After trying them all and finding the RMSE, the KNN imputation provides the lowest RMSE, so we will use KNN imputation for this.

For “AIRLINE_ID,” the missing values are linked to L4 and OH. By inspection, we can see that the airline ID for L4 carrier is 21217, so we can impute for all L4 carriers. However, for the OH carriers, we can see two distinct airline IDs. Therefore, we see which airline ID is the majority for OH, and then we impute that.

Question 2

For “MANUFACTURER,” we are going to categorize each name we see into a standardization map. We will take the unique entries we see line up and put them to one name (e.g. All the Boeing variations will be linked to BOEING). For those with two names separated by a slash, we will check to see if either are known manufacturers, and if not, we will just use the full name as a manufacturer. From here, we were able to get 37,272 Boeing entries, 14,628 Airbus entries, and 10,635 Embraer entries, along with all the other manufacturers.

For “MODEL,” this one was quite tricky because there was a great variety of model names. Therefore, some numbers were identified to be correlated with different airlines. For example, 737 for Boeing and A318 for Aerobus. If these specific strings are identified within the model name, then they are categorized together in one model. The rest that weren’t as popular were left alone.

For “AIRCRAFT_STATUS,” we saw that there were only 3 distinct values, but the only variation in these was capitalization. Therefore, we were able to just make all values in the column upper-case, and that fixed that problem.

For “OPERATING_STATUS,” we saw a very similar thing happen. There was only Y and N, with variation in capitalization. Therefore, we were able to just make all values in the column upper-case, and that fixed that problem.

Question 3

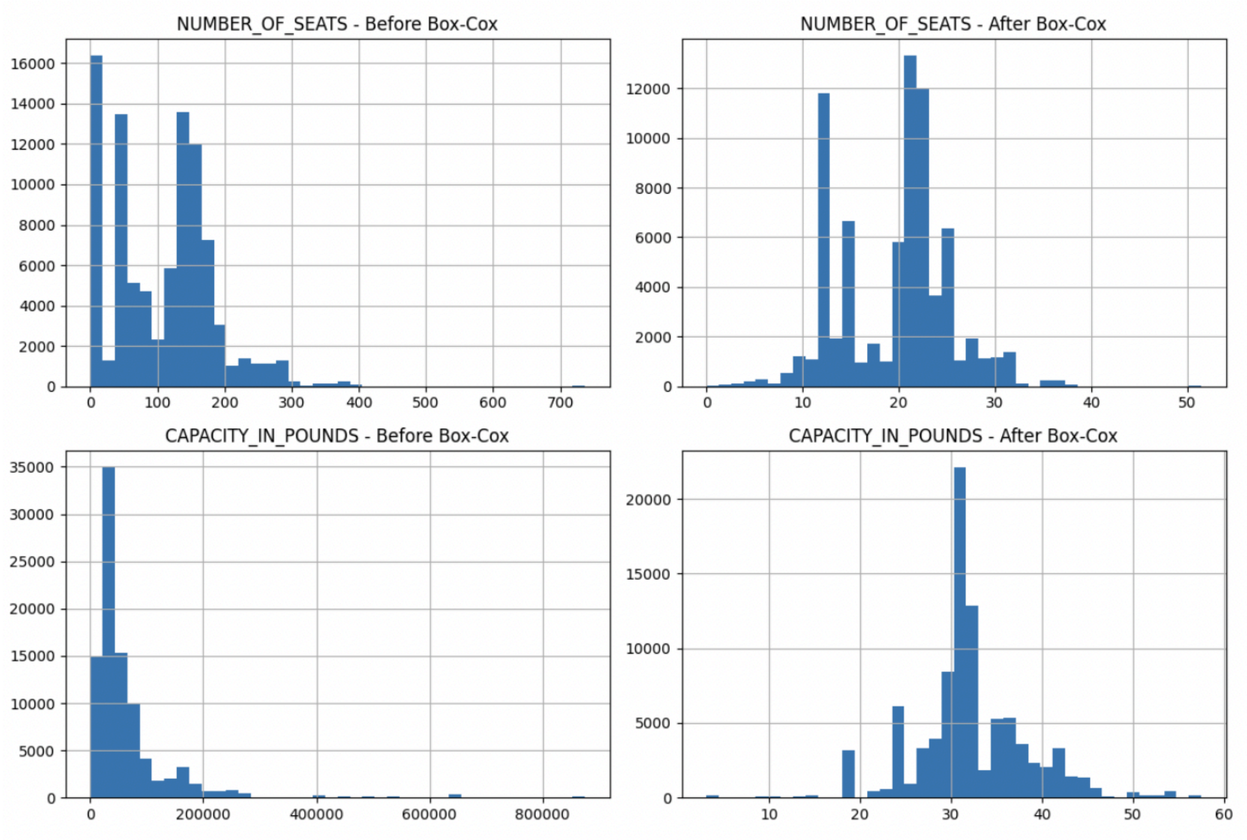
From here, I used the .dropna() function to just drop all the rows with missing data. After examining the data, the percentage of rows remaining is revealed to be 67%.

Question 4

For the “NUMBER_OF_SEATS” data, before applying box-cox, it is very right-skewed, with a lot of small jets with about 30 to 100 seats and a long tail up to 700. After applying box-cox, it is much more symmetric, with a clear central peak around the bulk (which is around 10-30 after the transformation).

For the “CAPACITY_IN_POUNDS” data, before applying box-cox, the data is heavily right skewed. Most aircrafts are under 200k pounds, but there are some outliers near 1 million pounds. After applying box-cox, it is much more symmetric, compressed into a tighter band of around 20-50 after the transformation.

This matters because a more normal distribution helps improve the performance and interpretability of more statistical methods, which include linear regression, clustering, and hypothesis testing.

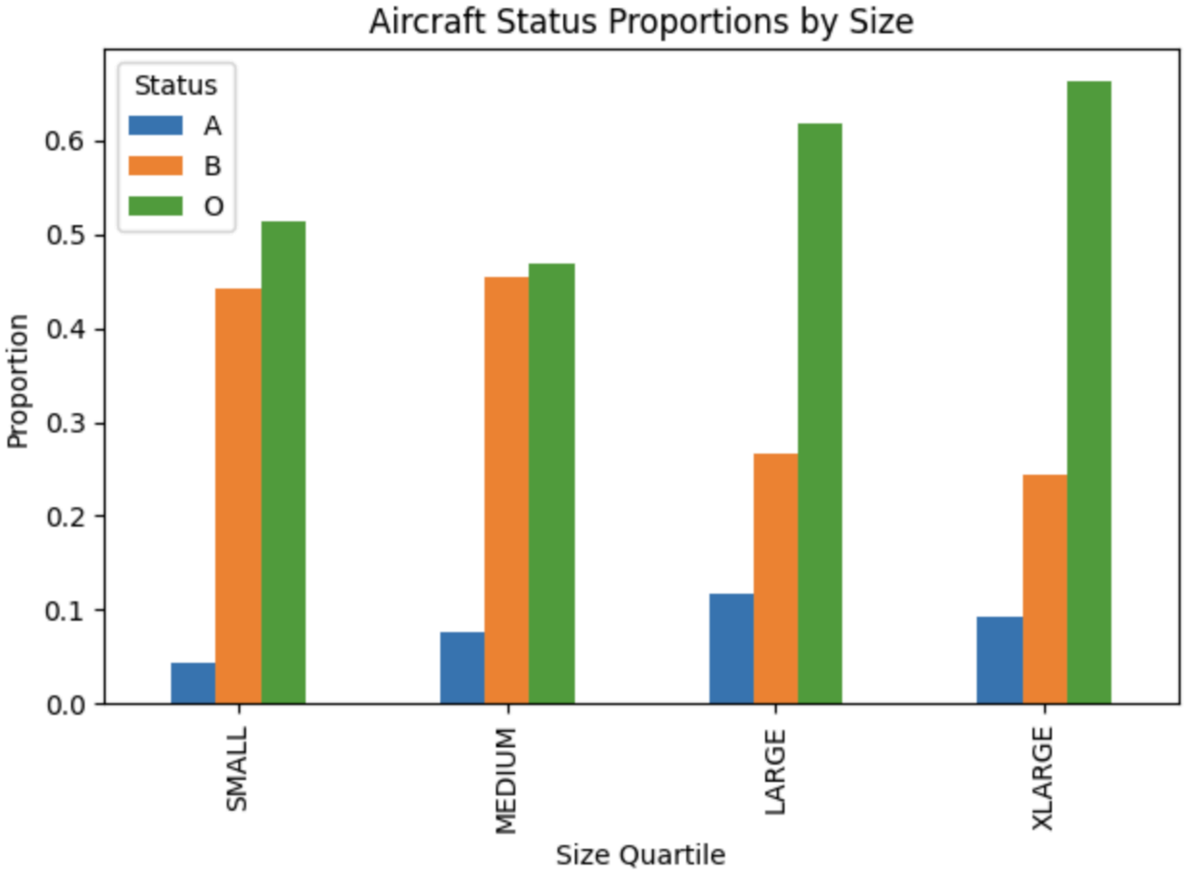


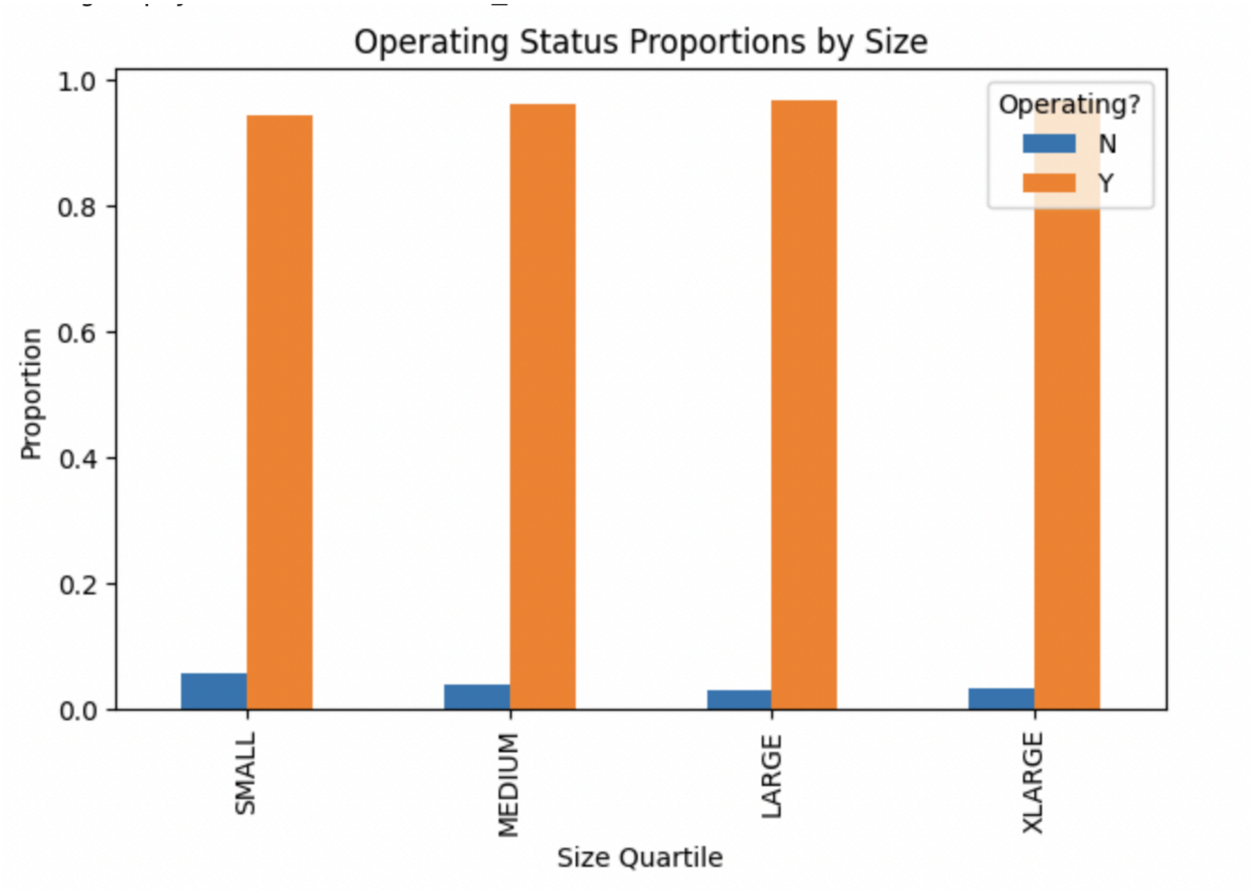
Question 5

In aircraft status, “A” rises from about 4% to 12% in the large-jet quartile, then it dips slightly in XLARGE. “B” peaks in the medium group and drops off sharply in the larger groups. “O” makes up the bulk for most everything, and it increases with size. We can see here that bigger jets are more likely to be actively flying. Medium-sized planes see the most storage “B.” “O” dominates across all sizes, but especially among the largest jets.

In operating status, “Y” increases with size, and “N” decreases with size, but Y is predominantly the majority to the extent of 95+%. Nearly every jet in a commercial fleet is expected to fly, so it makes sense that “Y” dominates. The small differences across size groups can reveal a lot more in this case. There is a higher “N” in smaller aircraft, which can signal a few things. This suggests that regional/smaller planes are intentionally cycled more often out of service, which represents the demand for these. The highest utilization of the larger planes highlights their role of being able to carry maximum load and therefore maximum profit.

A clear trend in the data is that the larger the aircraft, the more likely it is to be active and in-service.


Aircraft Status Proportions by Size
Status
A
B
O
Proportion
0.0
0.1
0.2
0.3
0.4
0.5
0.6
SMALL
MEDIUM
LARGE
XLARGE
Size Quartile



Generative AI:

For this assignment, most of the AI used was to look up functions that would assist in what I would want to be done. Because I haven't had much experience coding, it was very helpful to be able to look up functions that would be able to do the things I would want to do. For example, learning how to make all the data values in a column all capital, learning the .apply() function, and learning .unique() and .value_counts().