Virginia Polytechnic Institute and State University

Rayden Dodd

Reza Jafari, Ph.

7/7/2025

Information Visualization CS5764

https://dashapp-278644791327.us-east1.run.app/

Dash

Table of content

1. Abstract
2. Introduction
3. Description of the Dataset
4. Pre-processing Dataset
5. Outlier Detection & Removal
6. Principal Component Analysis (PCA)
7. Normality Test

8. Data Transformation

9. Heatmap & Pearson Correlation Coefficient Matrix

10. Statistics

11. Data Visualization

12. Subplots

13. Tables

14. Dashboard

15. Observations

16. Conclusion

    a. Insights from Graphs

    b. Dashboard Utility

    c. User-Friendliness & Feedback

    d. Functionality

17. Appendix: Python Code

18. References

Table of figures and tables.

Abstract.

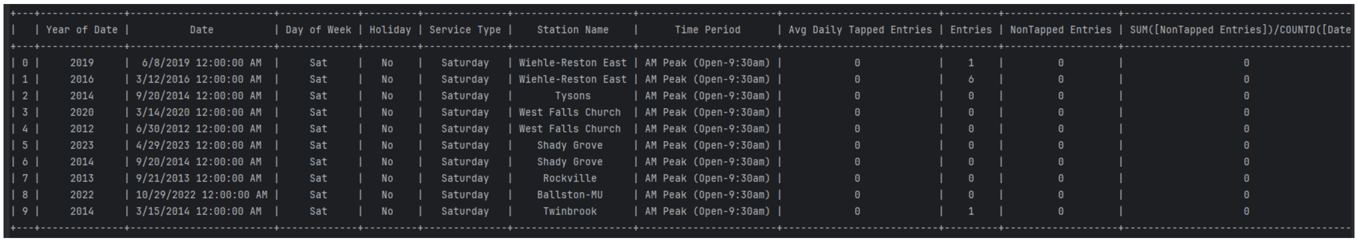
The Purpose of this report is to go through and analyze a data set draw conclusions and information out. We accomplish this by using various data analysis and data visualization tools, enabling clear interpretation and communication of the underlying patterns and trends within the data.

Introduction

This report presents a comprehensive analysis of metro ridership data as part of the Final Term Project (FTP). The primary objective is to explore, clean, transform, and visualize the data to uncover patterns, identify anomalies, and provide insights. To achieve this, we have developed an interactive Python-based dashboard using Dash and Plotly, which allows users to explore the dataset dynamically. While also provided static graphs to better understand more complex relationships. In this report we will cover our data cleaning, transformation, outlier removal, normalization, statistics and visualization.

Description of the dataset

This dataset was pulled from WMATA's (Washington Metropolitan Area Transit Authority) website directly. They two provide an interactive dashboard for the user to be able to manipulate different graphs. The data I downloaded from their Metrorail Ridership Summary Page was all the available data from Jan 2012 – Jun 2025. The data set contained rows that were by date, station, and hours. Overall, there were 13 features of which the important categorical ones were metro stations and time periods, and the important numerical features were entries, exits, unpaid entries and unpaid exits. Unpaid entrances and exits only started being added Jan 2023. The data set is very large, roughly 1,000,000 rows of data and nearly 10 million Metro rides taken over this time range. In this dataset I believe that stations and time will be independent variables while entries and exits will be the dependent variables. This data set could be important for industry in knowing the flow of the metro system when and how many people are go to and from work. Where should you buy real estate or set up a coffee shop at what times of day are these areas busy. For WMATA they can use the data to increase security or service and know where the bottlenecks in their system are.

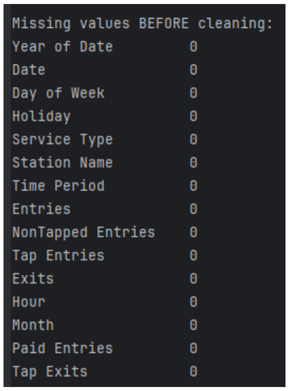


```
+---+--------------+------------------------+-------------+---------+--------------+--------------------+-----------------------+--------------------------+---------+-------------------+-------------------------------------
|   | Year of Date |          Date          | Day of Week | Holiday | Service Type |    Station Name    |      Time Period      | Avg Daily Tapped Entries | Entries | NonTapped Entries | SUM([NonTapped Entries])/COUNTD([Date
+---+--------------+------------------------+-------------+---------+--------------+--------------------+-----------------------+--------------------------+---------+-------------------+-------------------------------------
| 0 |     2019     |  6/8/2019 12:00:00 AM  |     Sat     |   No    |   Saturday   | Wiehle-Reston East | AM Peak (Open-9:30am) |            0             |    1    |         0         |                  0
| 1 |     2016     | 3/12/2016 12:00:00 AM  |     Sat     |   No    |   Saturday   | Wiehle-Reston East | AM Peak (Open-9:30am) |            0             |    6    |         0         |                  0
| 2 |     2014     | 9/20/2014 12:00:00 AM  |     Sat     |   No    |   Saturday   |       Tysons       | AM Peak (Open-9:30am) |            0             |    0    |         0         |                  0
| 3 |     2020     | 3/14/2020 12:00:00 AM  |     Sat     |   No    |   Saturday   | West Falls Church  | AM Peak (Open-9:30am) |            0             |    0    |         0         |                  0
| 4 |     2012     | 6/30/2012 12:00:00 AM  |     Sat     |   No    |   Saturday   | West Falls Church  | AM Peak (Open-9:30am) |            0             |    0    |         0         |                  0
| 5 |     2023     | 4/29/2023 12:00:00 AM  |     Sat     |   No    |   Saturday   |    Shady Grove     | AM Peak (Open-9:30am) |            0             |    0    |         0         |                  0
| 6 |     2014     | 9/20/2014 12:00:00 AM  |     Sat     |   No    |   Saturday   |    Shady Grove     | AM Peak (Open-9:30am) |            0             |    0    |         0         |                  0
| 7 |     2013     | 9/21/2013 12:00:00 AM  |     Sat     |   No    |   Saturday   |     Rockville      | AM Peak (Open-9:30am) |            0             |    0    |         0         |                  0
| 8 |     2022     | 10/29/2022 12:00:00 AM |     Sat     |   No    |   Saturday   |    Ballston-MU     | AM Peak (Open-9:30am) |            0             |    0    |         0         |                  0
| 9 |     2014     | 3/15/2014 12:00:00 AM  |     Sat     |   No    |   Saturday   |     Twinbrook      | AM Peak (Open-9:30am) |            0             |    1    |         0         |                  0
+---+--------------+------------------------+-------------+---------+--------------+--------------------+-----------------------+--------------------------+---------+-------------------+-------------------------------------
```

Table 1: Head of the dataset

Pre-processing dataset

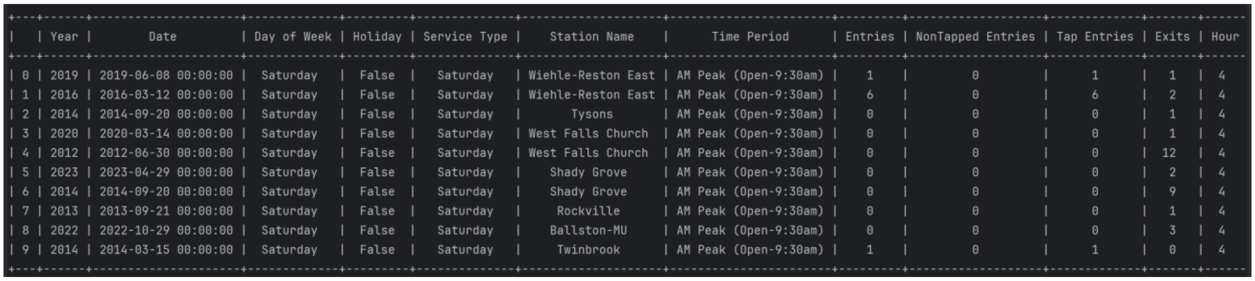
For data cleaning we first checked if there were any na or null values which there weren't



```
Missing values BEFORE cleaning:
Year of Date        0
Date                0
Day of Week         0
Holiday             0
Service Type        0
Station Name        0
Time Period         0
Entries             0
NonTapped Entries   0
Tap Entries         0
Exits               0
Hour                0
Month               0
Paid Entries        0
Tap Exits           0
```

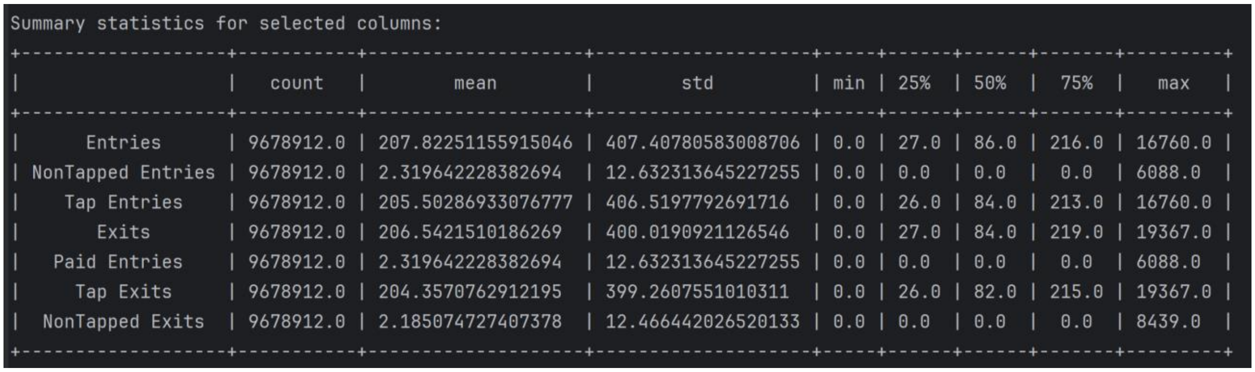
Table 2: No NA data

Next we removed all unnecessary rows from the data. As well turned all the date rows into datetimes and created new columns for months, Days of the Week. As well as turning Holidays from Yes and No to True and False. I also clipped out any negative data to 0.



```
+---+------+---------------------+-------------+---------+--------------+--------------------+-----------------------+---------+-------------------+-------------+-------+-----
|   | Year |        Date         | Day of Week | Holiday | Service Type |    Station Name    |      Time Period      | Entries | NonTapped Entries | Tap Entries | Exits | Hour
+---+------+---------------------+-------------+---------+--------------+--------------------+-----------------------+---------+-------------------+-------------+-------+-----
| 0 | 2019 | 2019-06-08 00:00:00 |  Saturday   |  False  |   Saturday   | Wiehle-Reston East | AM Peak (Open-9:30am) |    1    |         0         |      1      |   1   |  4
| 1 | 2016 | 2016-03-12 00:00:00 |  Saturday   |  False  |   Saturday   | Wiehle-Reston East | AM Peak (Open-9:30am) |    6    |         0         |      6      |   2   |  4
| 2 | 2014 | 2014-09-20 00:00:00 |  Saturday   |  False  |   Saturday   |       Tysons       | AM Peak (Open-9:30am) |    0    |         0         |      0      |   1   |  4
| 3 | 2020 | 2020-03-14 00:00:00 |  Saturday   |  False  |   Saturday   | West Falls Church  | AM Peak (Open-9:30am) |    0    |         0         |      0      |   1   |  4
| 4 | 2012 | 2012-06-30 00:00:00 |  Saturday   |  False  |   Saturday   | West Falls Church  | AM Peak (Open-9:30am) |    0    |         0         |      0      |  12   |  4
| 5 | 2023 | 2023-04-29 00:00:00 |  Saturday   |  False  |   Saturday   |    Shady Grove     | AM Peak (Open-9:30am) |    0    |         0         |      0      |   2   |  4
| 6 | 2014 | 2014-09-20 00:00:00 |  Saturday   |  False  |   Saturday   |    Shady Grove     | AM Peak (Open-9:30am) |    0    |         0         |      0      |   9   |  4
| 7 | 2013 | 2013-09-21 00:00:00 |  Saturday   |  False  |   Saturday   |     Rockville      | AM Peak (Open-9:30am) |    0    |         0         |      0      |   1   |  4
| 8 | 2022 | 2022-10-29 00:00:00 |  Saturday   |  False  |   Saturday   |    Ballston-MU     | AM Peak (Open-9:30am) |    0    |         0         |      0      |   3   |  4
| 9 | 2014 | 2014-03-15 00:00:00 |  Saturday   |  False  |   Saturday   |     Twinbrook      | AM Peak (Open-9:30am) |    1    |         0         |      1      |   0   |  4
+---+------+---------------------+-------------+---------+--------------+--------------------+-----------------------+---------+-------------------+-------------+-------+-----
```

Table 3: Cleaned dataset

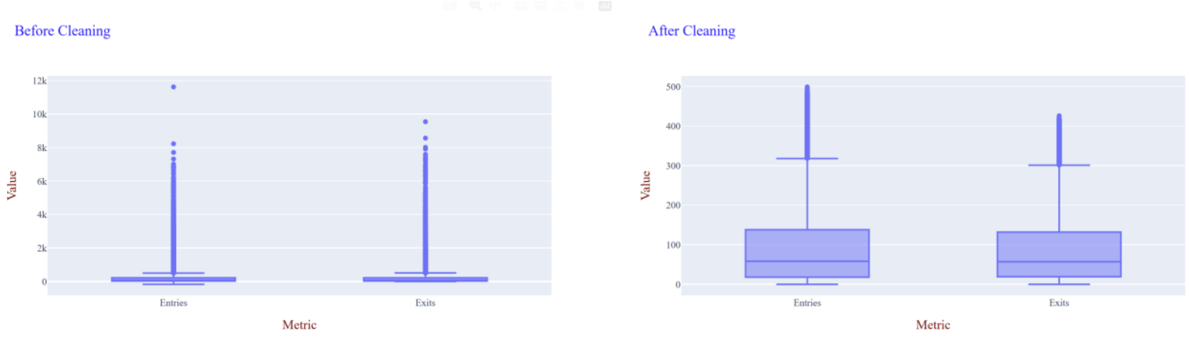


```
Summary statistics for selected columns:
+-------------------+-----------+--------------------+--------------------+-----+------+------+-------+---------+
|                   |   count   |        mean        |        std         | min | 25%  | 50%  |  75%  |   max   |
+-------------------+-----------+--------------------+--------------------+-----+------+------+-------+---------+
|      Entries      | 9678912.0 | 207.82251155915046 | 407.40780583008706 | 0.0 | 27.0 | 86.0 | 216.0 | 16760.0 |
| NonTapped Entries | 9678912.0 | 2.319642228382694  | 12.632313645227255 | 0.0 | 0.0  | 0.0  |  0.0  | 6088.0  |
|    Tap Entries    | 9678912.0 | 205.50286933076777 | 406.5197792691716  | 0.0 | 26.0 | 84.0 | 213.0 | 16760.0 |
|       Exits       | 9678912.0 | 206.5421510186269  | 400.0190921126546  | 0.0 | 27.0 | 84.0 | 219.0 | 19367.0 |
|   Paid Entries    | 9678912.0 | 2.319642228382694  | 12.632313645227255 | 0.0 | 0.0  | 0.0  |  0.0  | 6088.0  |
|     Tap Exits     | 9678912.0 | 204.3570762912195  | 399.2607551010311  | 0.0 | 26.0 | 82.0 | 215.0 | 19367.0 |
|  NonTapped Exits  | 9678912.0 | 2.185074727407378  | 12.466442026520133 | 0.0 | 0.0  | 0.0  |  0.0  | 8439.0  |
+-------------------+-----------+--------------------+--------------------+-----+------+------+-------+---------+
```

Table 4: Statistics of cleaned dataset

Outlier detection & removal

For outlier detection I used Z-Score and box plots to view the distribution of the data. I noticed a lot of outliers in the box plots. I believe this is because most days of the year the metro is Soley used for commuting and generally has the same about of low volume of people going to and from work each day. However, Saturdays roll around, holidays, protests, elections, or sporting events and the metro gets a huge influx of passengers on top of the already normal daily passages for that day. I figured for what I am trying to achieve with my graphs the outliers were not an issue but showed demand for the metro and was valuable info. However, on my interactive dashboard the user may choose to have a z-score removal of outliers.

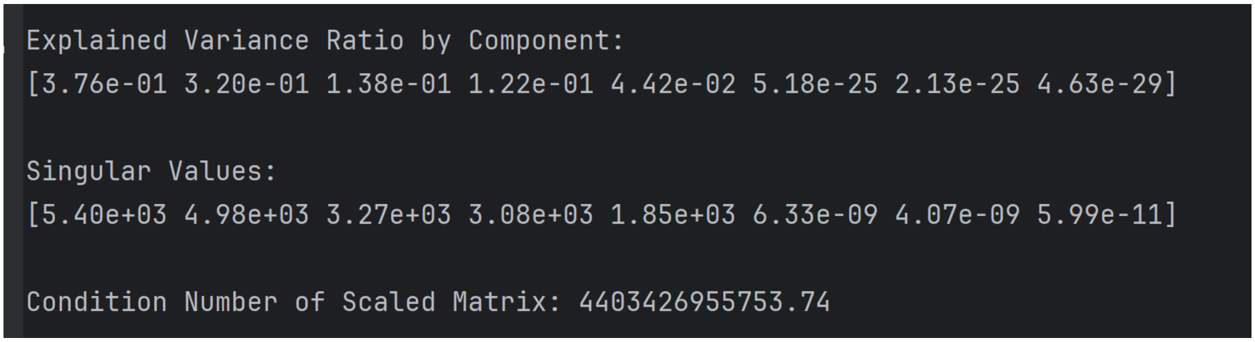


Plot 1: Box plot before and after z-score outlier removal

You can see that after the removal there are less outliers the box begins to widen.

Principal Component Analysis (PCA)

Using my cleaned dataset, I ran a PCA on my data and got the following results:



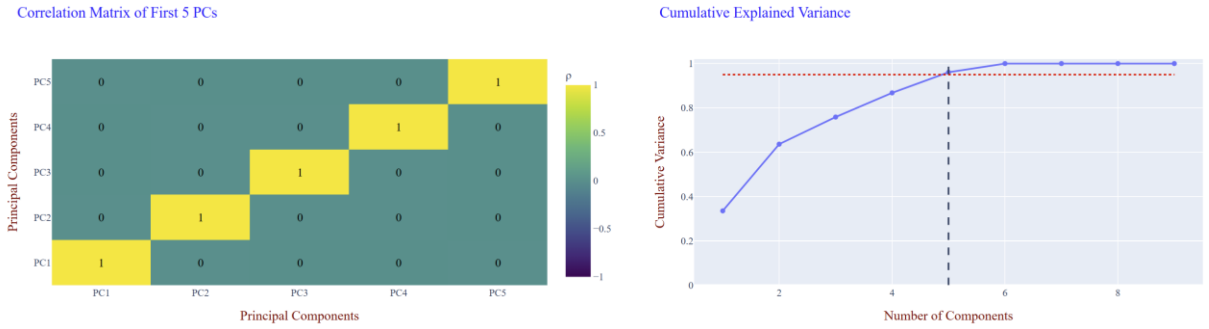
```
Explained Variance Ratio by Component:
[3.76e-01 3.20e-01 1.38e-01 1.22e-01 4.42e-02 5.18e-25 2.13e-25 4.63e-29]

Singular Values:
[5.40e+03 4.98e+03 3.27e+03 3.08e+03 1.85e+03 6.33e-09 4.07e-09 5.99e-11]

Condition Number of Scaled Matrix: 4403426955753.74
```

Figure 1: PCA Explained variance, singular values, and condition number

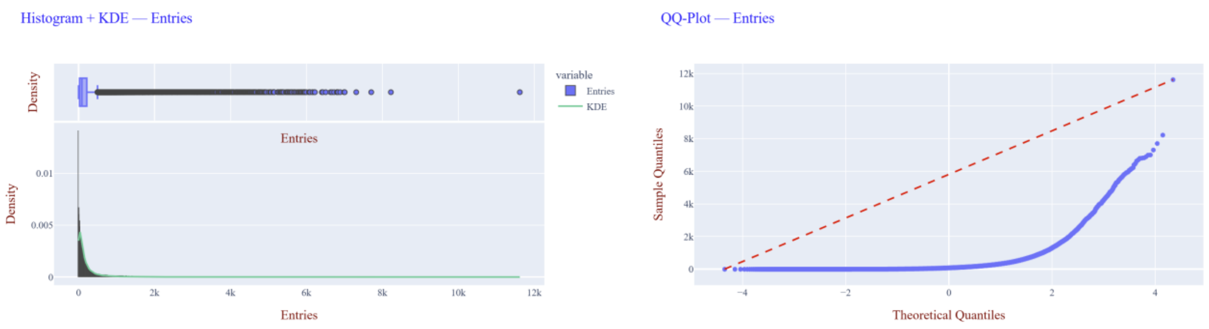
I also created a heatmap to visualize the correlation between the principal components, along with a cumulative explained variance plot (also known as an elbow plot). This helped me determine that using 5 principal components was ideal, as they collectively captured about 95% of the total variance in the data. You can also see this with the singular values in Figure 1 as 5 PCS have actually values while the last 3 are near zero showing that they are redundant to the data set.



Plot 2: PCA Heatmap and Cumulative Explained Variance plot showing 5 PC as the ideal

Normality Test

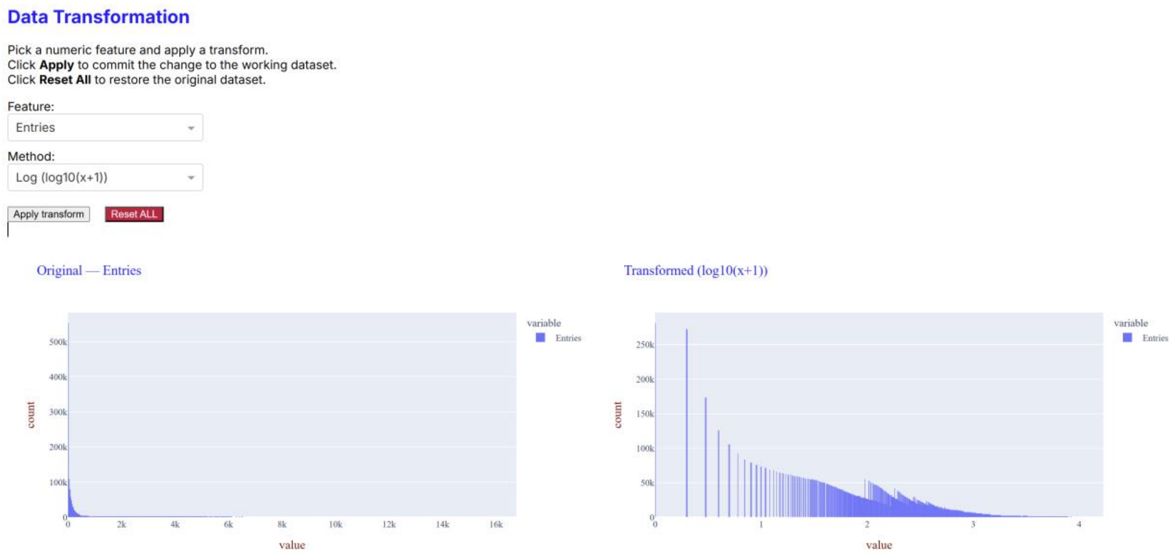
For normality Testing I went with Histogram plots with KDEs and a QQ plot. I could instantly tell from HIST that my data was very right skewed with all those large outliers. Most of the data clumped up on the left side of the graph with a large tail to the right. The QQ plot confirmed this as the blue line diverts from the ideal normal dashed red line quite a bit showing that my data is not normal.



Plot 3: Histogram+KDE with a horizontal box lot and a QQ plot

Data Transformation

The outliers represent real-world, high-traffic days at certain metro stations (e.g., special events, rush hours, or anomalies tied to usage patterns). Transforming the data would suppress meaningful variance, potentially distorting these operationally significant values. Since my goal is to understand actual metro usage. However, if the user wants to transform one of the numerical features then they can in the interactive dashboard.



Data Transformation

Pick a numeric feature and apply a transform.
Click **Apply** to commit the change to the working dataset.
Click **Reset All** to restore the original dataset.

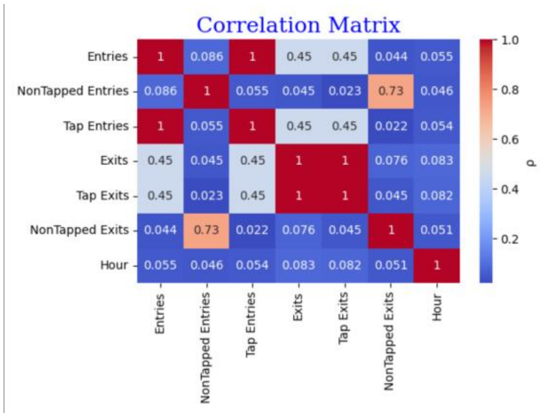
Feature: Entries

Method: Log (log10(x+1))

Figure 2: Data Transformation tab showing how you can change certain features and plotting histogram plots to show the before and after difference.

Here we see the difference between the dist plot normal vs the dist plot where entries have been log10 this helps visual it more as it makes those smaller in amount large numbers more visible on a logarithmic scale.

Heatmap & Pearson correlation coefficient matrix



Correlation Matrix

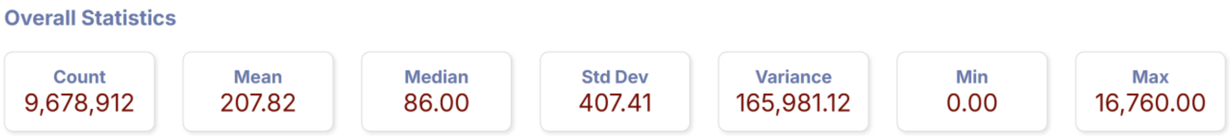
	Entries	NonTapped Entries	Tap Entries	Exits	Tap Exits	NonTapped Exits	Hour
Entries	1	0.086	1	0.45	0.45	0.044	0.055
NonTapped Entries	0.086	1	0.055	0.045	0.023	0.73	0.046
Tap Entries	1	0.055	1	0.45	0.45	0.022	0.054
Exits	0.45	0.045	0.45	1	1	0.076	0.083
Tap Exits	0.45	0.023	0.45	1	1	0.045	0.082
NonTapped Exits	0.044	0.73	0.022	0.076	0.045	1	0.051
Hour	0.055	0.046	0.054	0.083	0.082	0.051	1

Plot 4: Heat map of numerical features

	Entries	Nontapped Entries	Tap Entries	Exits	Tap Exits	NonTapped Exits	Hour
Entires	Same						
Nontapped Entries	Very few per Entry	Same					

Tap Entries	Pretty much similar as non-tapped only came around recently	Low Correlation	Same				
Exits	Slightly positively correlated	Low Correlation	Since tap entries is corealted 1.0 to Entries it has the same correlation to Exits as Entries does	Same			
Tap Exits	Same as Exits as tap Exits is strongly correlated with Exits	Low Correlation	^	Not many people fair evading so tap exits almost equals exits	Same		
NonTapped Exits	Low Correlation	Strong Correlation possibly due to people who don’t pay going in don’t pay coming out	Low Correlation	Low Correlation	Low Correlation	Same	
Hour	Low Correlation	Low Correlation	Low Correlation	Low Correlation	Low Correlation	Low Correlation	Same

Statistics



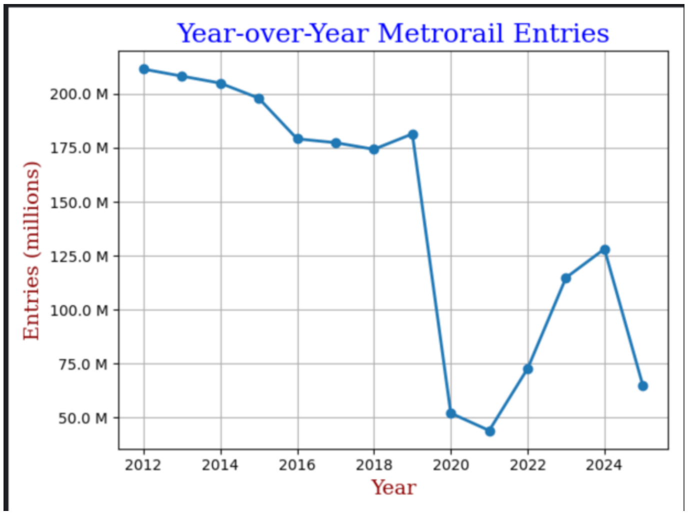
Overall Statistics

Count	Mean	Median	Std Dev	Variance	Min	Max
9,678,912	207.82	86.00	407.41	165,981.12	0.00	16,760.00

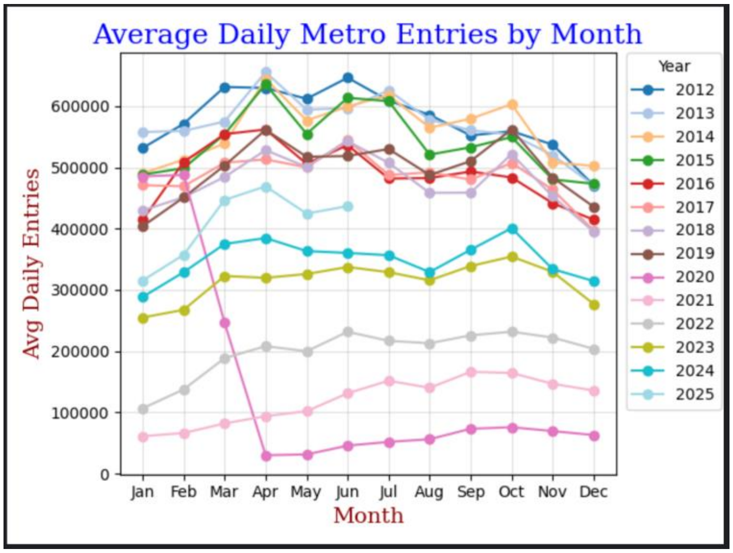
Figure 3: Statistics of Entries

The dataset is highly right-skewed, with a mean of 207.82 and a median of just 86, showing that most values are low but there are a few very high spikes. The large standard deviation and variance confirm that there's a lot of variability due to real-world outliers like special events or busy stations.

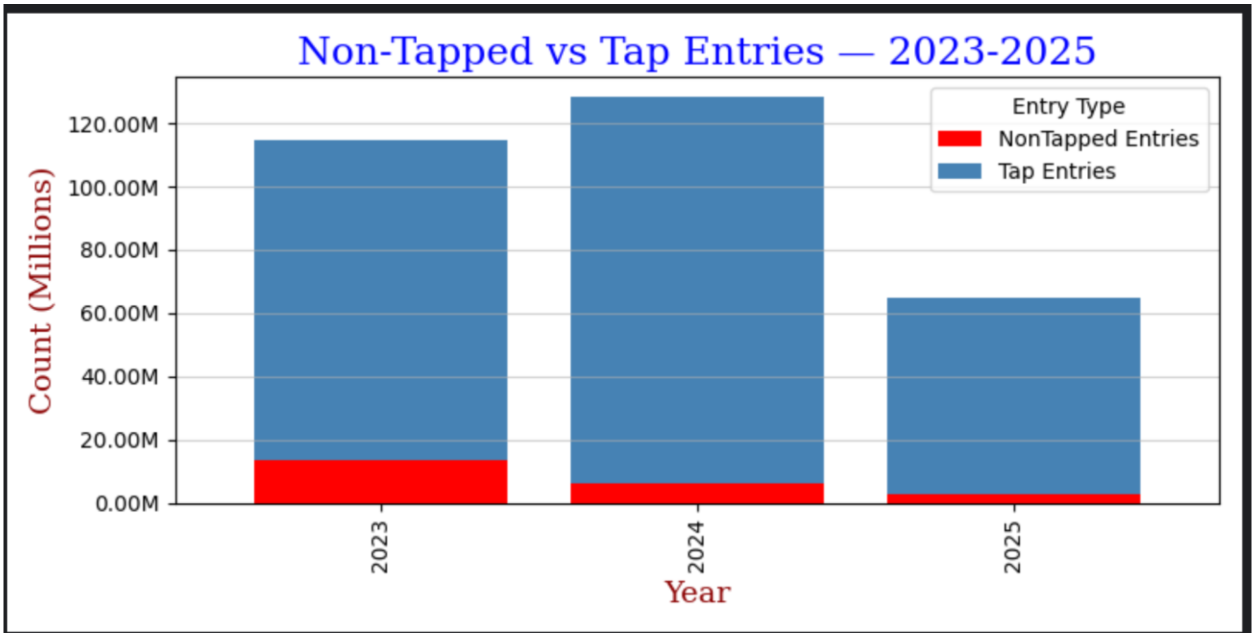
Data Visualization



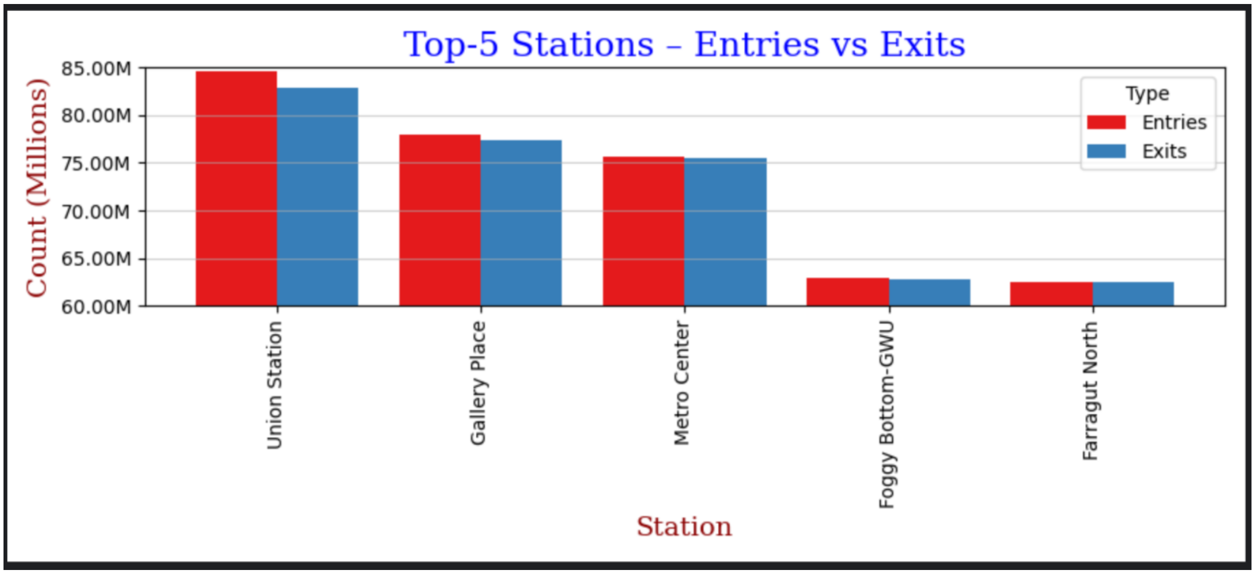
This line chart shows the riders per year and clearly shows a steep fall of in 2020 dude to the covid 19 pandemic and we can see we have not recovered yet.



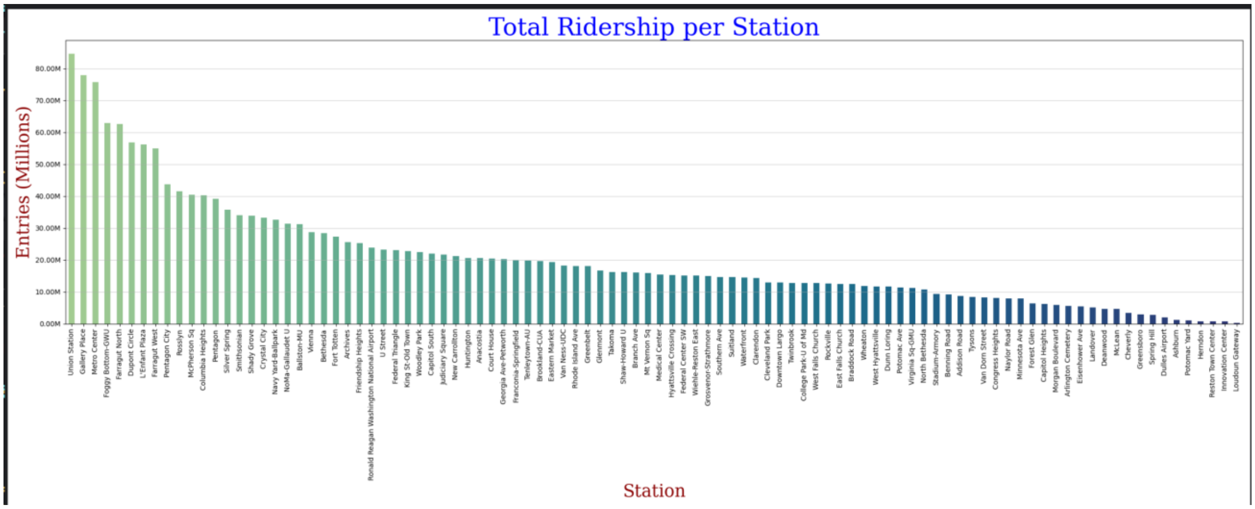
This plot shows the average daily metro riders by month with years 2012-2025 overlayed over. We can see that around Oct and the summer months there is a spike in ridership while Jan tends to be on the low end.



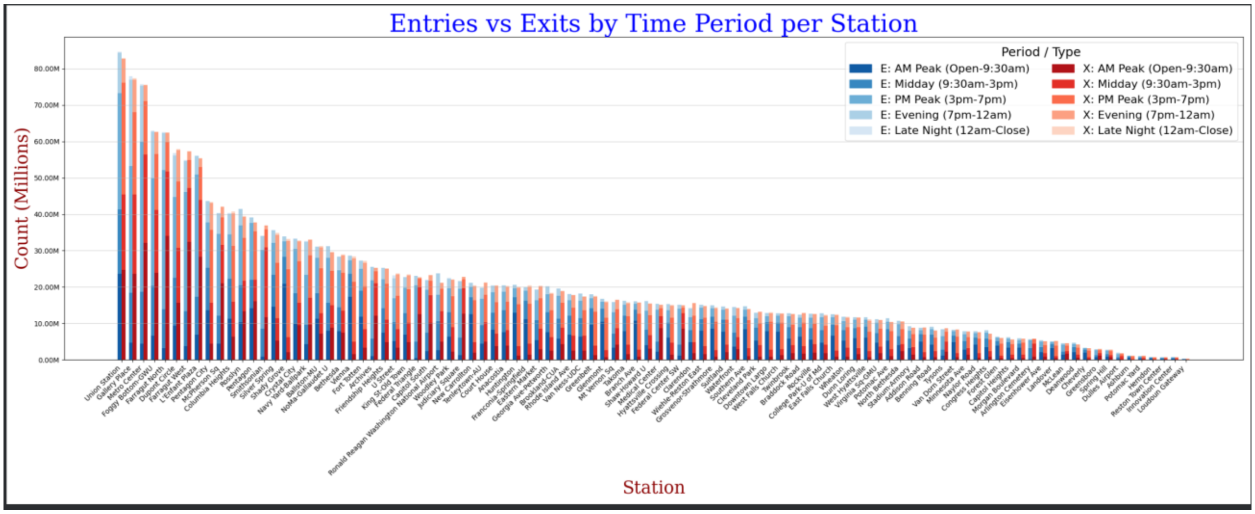
We can see from this stacked bar chart unpaid and paid entries. we can see that even though ridership went up in 2024 unpaid entries went down. Probably due to increased security and making it easier to pay.



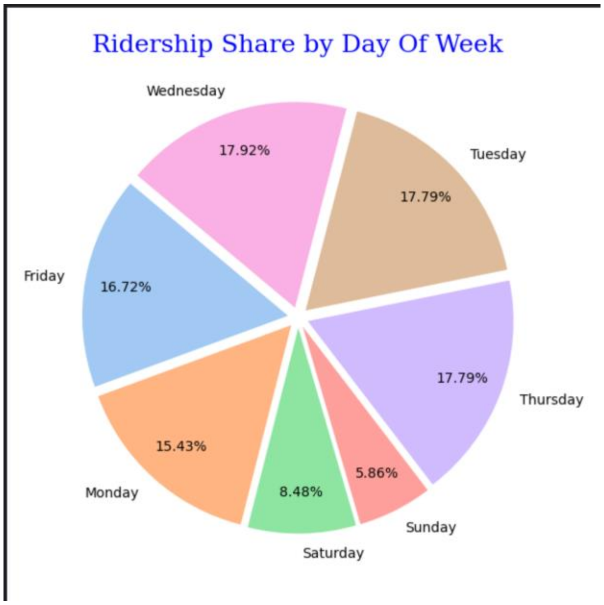
Here we have the top 5 most ridden stations in a side by side bar chart comparing entries to exits. We can see that they all have more entries than exits and this is expected as some exits glitch out peoples phones die and must be let through. Doesn’t show too much of a disparity.



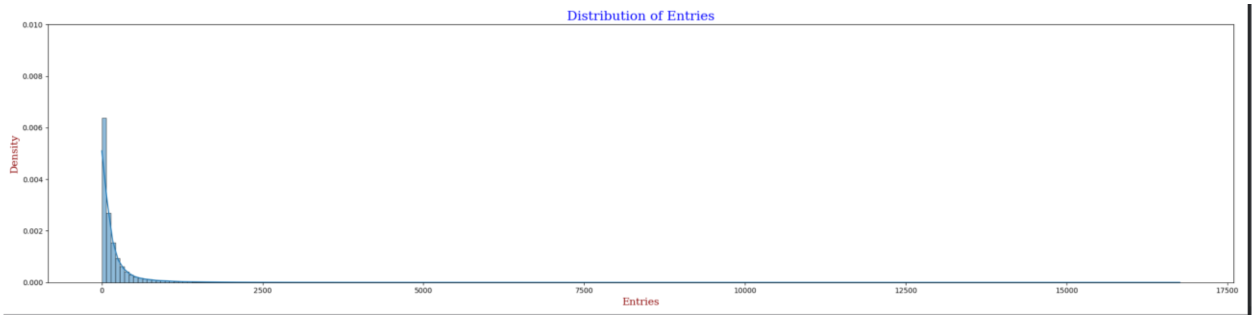
Here is a bar chart of total rider ship from 2012-2025 by station showing our most traveled two station union station on the left and the least traveled Loudoun gateway. This is mainly due Union Station being in the heart of dc and connected to the 2nd busiest Amtrak station in America. While Loudoun gateway is relatively new and 1 hour and 30 mins away from DC.



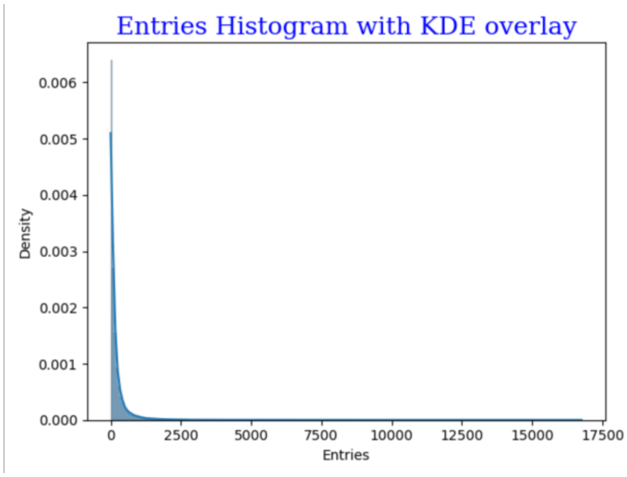
This is like the chart above but splits entries and exits and splits it further by period of day. You can metro center has very low early morning entrance and a ton of early morning exits showing that this is most likely a place where a lot of people are going to for work. This shows which stations are busy when and for people coming or going.



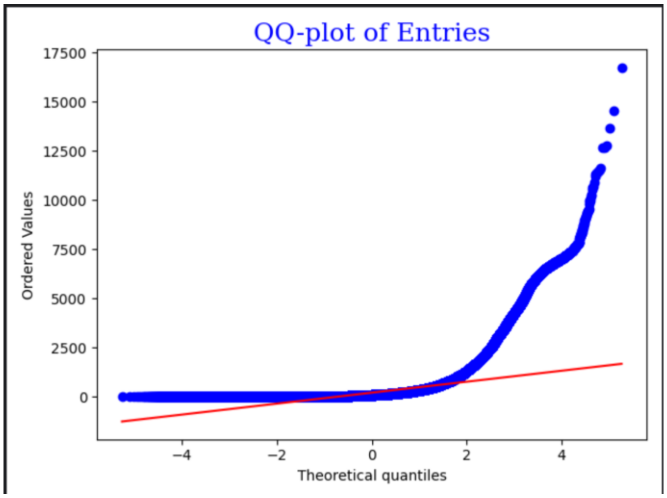
This pie chart shows rider ship share by day of the week showing that the metro is mainly commuter rail as Saturday and Sunday are ridden less. We can also see that Wed-Thurs is the most popular probably due to Fridays and Mondays being holidays and people not wanting to go into work on those days.



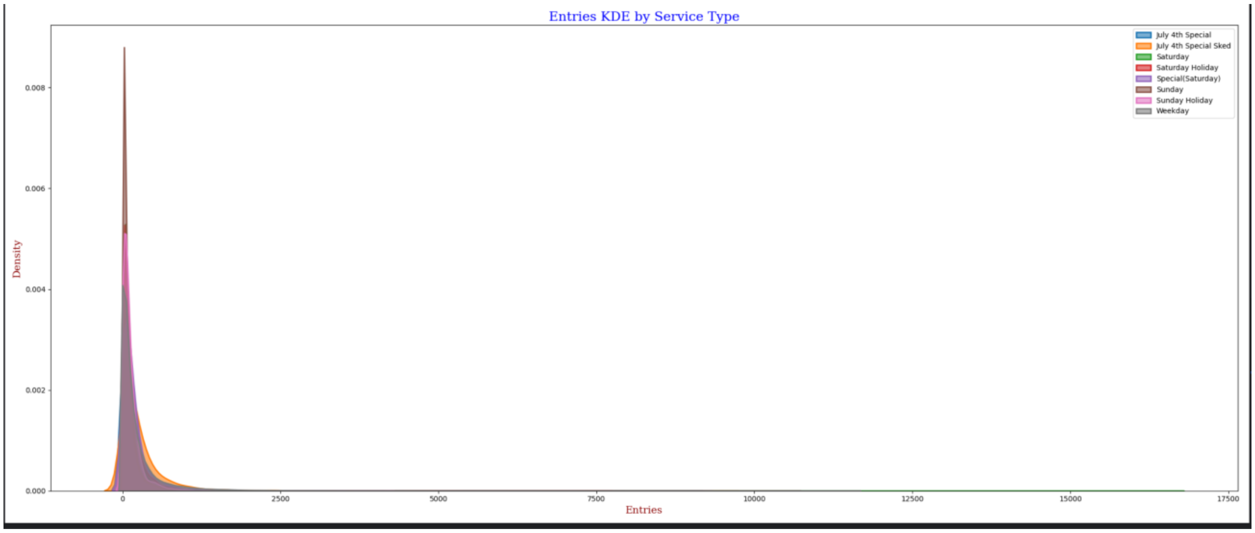
Distribution plot of 240 bins and a KDE overlay. Showing how rightward skewed our data is and how our data is not normal.



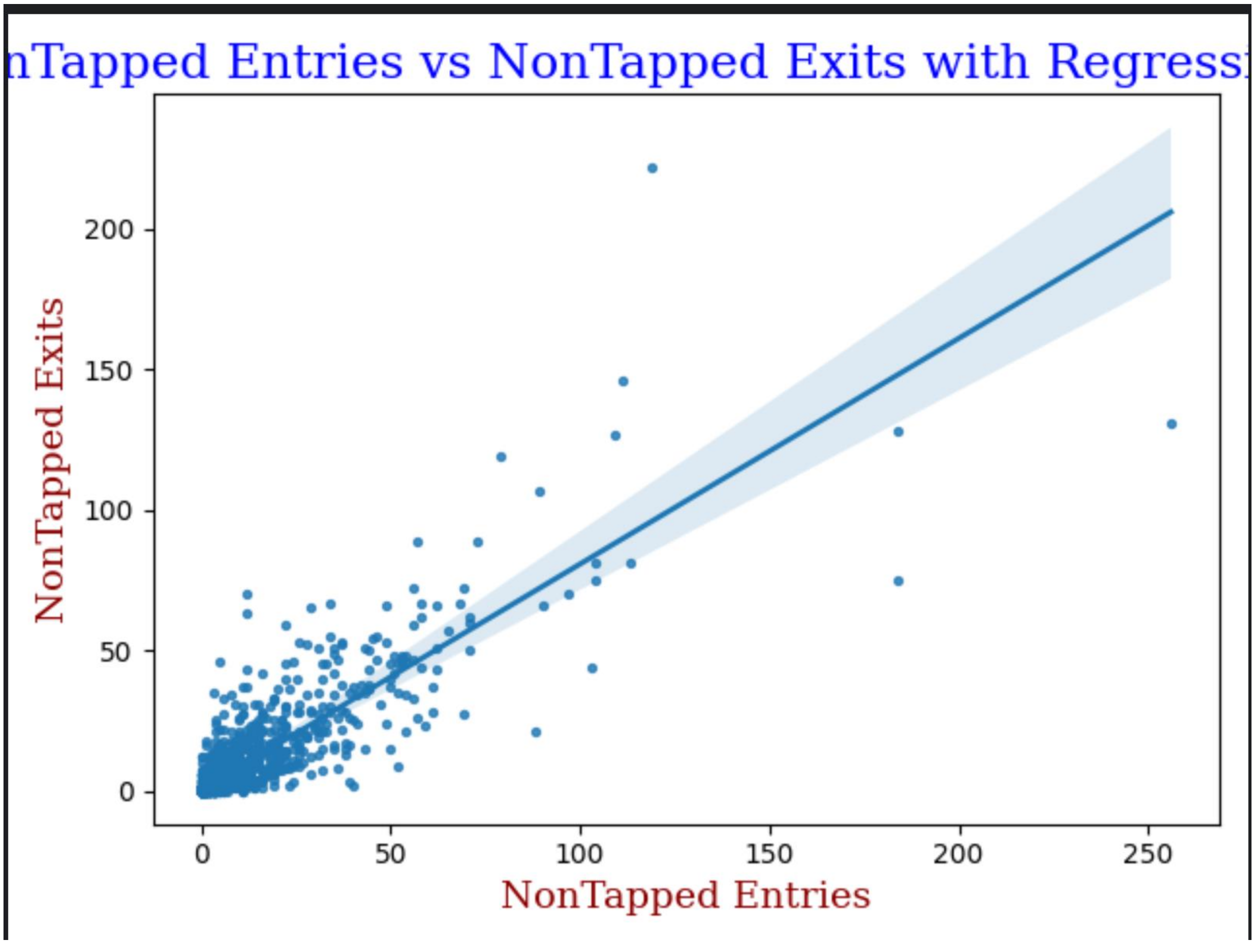
Histogram plot also with KDE overlay showing that large right skew .



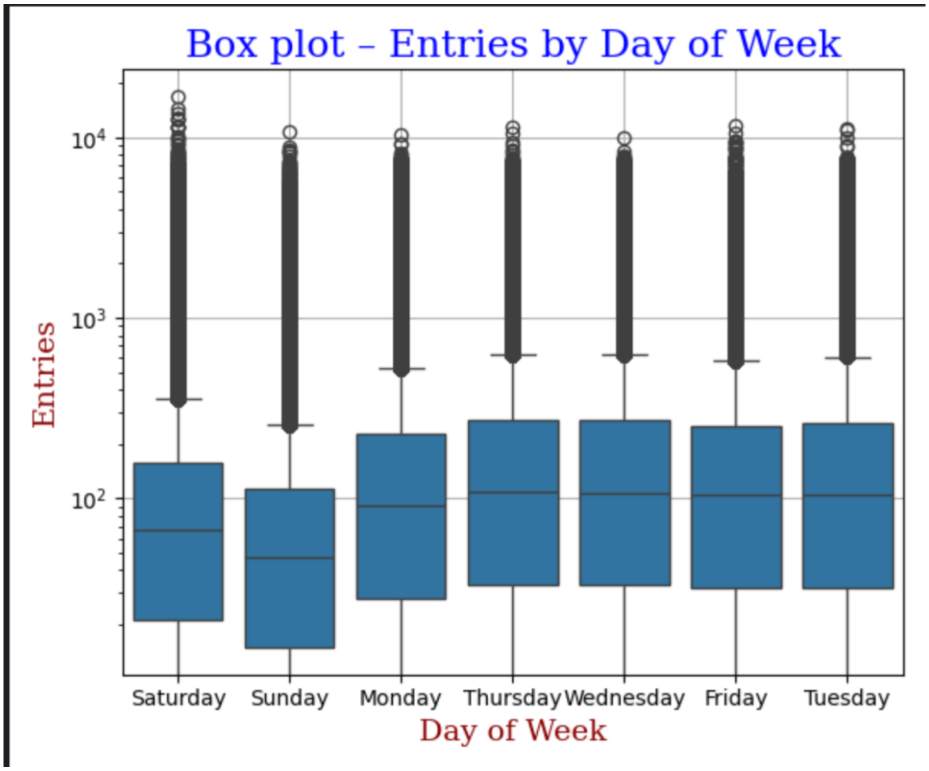
QQ plot also confirming how our data is not normal as it does not following the theoretical ical normal red line and skews up and even has some outliers.



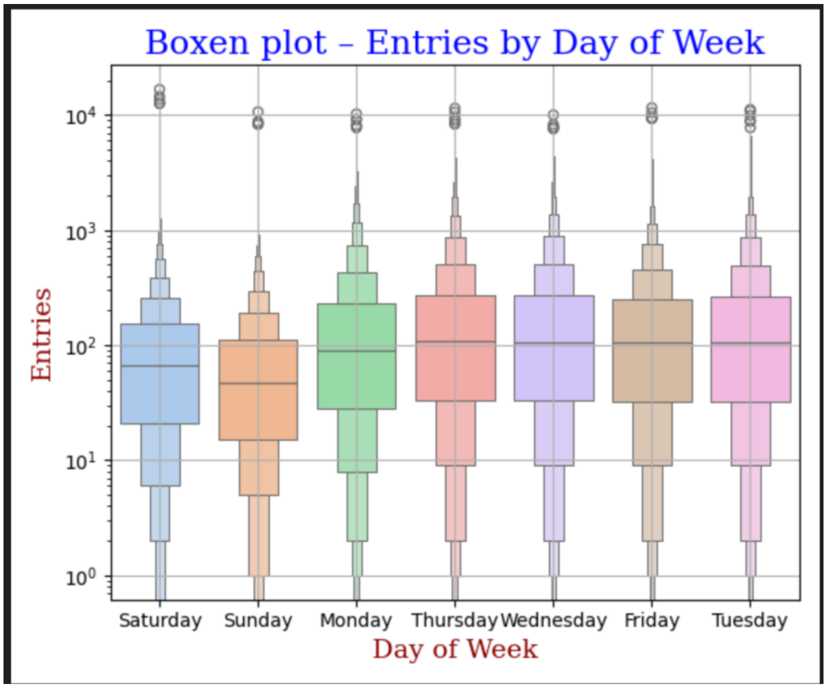
Histogram with KDE this time with a hue for service type. We ca n se that 4th of July is bigger than the other has it has more average travel on that day but then we can also see the large right tail is green meaning that it is Saturdays that will have those large outliers



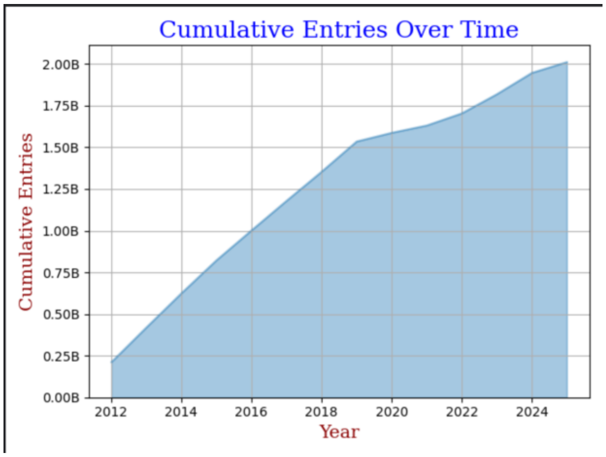
Shows how non tapped entries and non trapped exits are strongly correlated at a r of 75. This could be due to the same people that don’t pay to enter also don’t pay to leave.



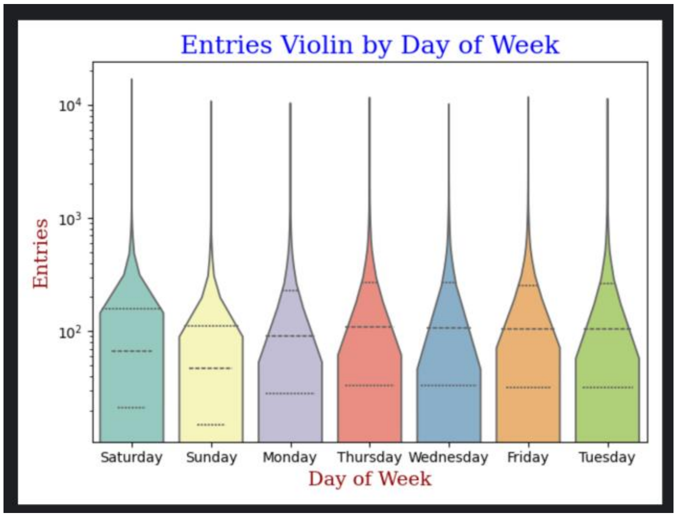
Box plots per day We can see that each week day is around the exact same will the weekend days have a lower median. But here again we can see Saturday has the highest and most outliers . This is portably due to sporting events and protests normally being held on Saturdays.



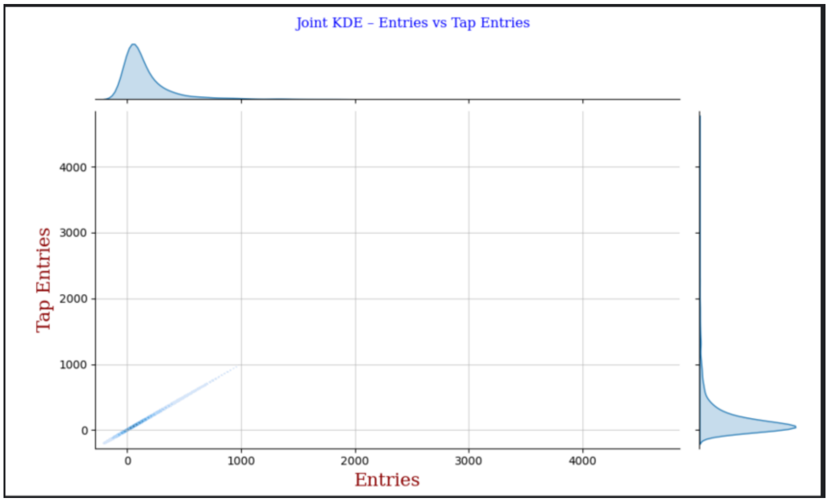
Boxen plot shows very similar to the box plots but we can see that the weekend day tend to be more favorable to have low numbers on normal weekends when there aren't outlier events.



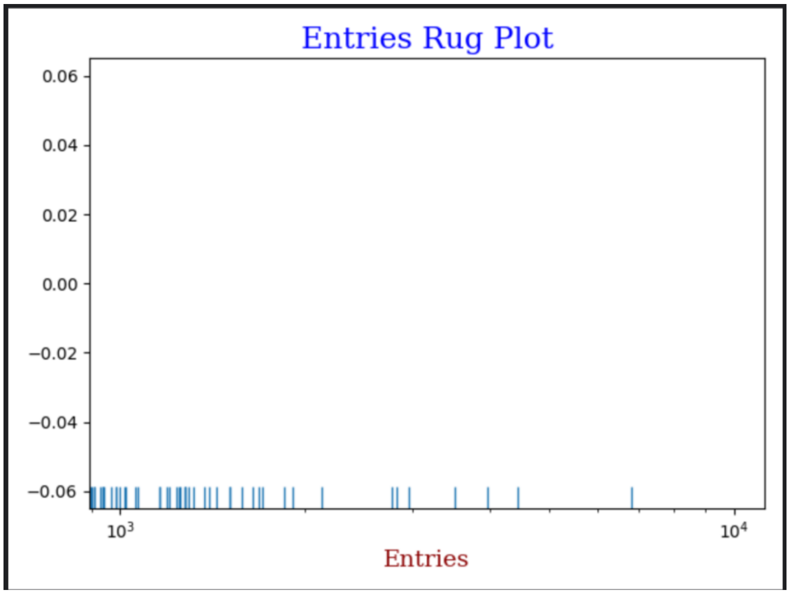
Area Plot of cumulative ridership over the years. You can see 2012-202 had a steady growth due to each year have the same number of yearly riders then covid hits and you can see the decline in growth and the low curve back up to where we used to be.



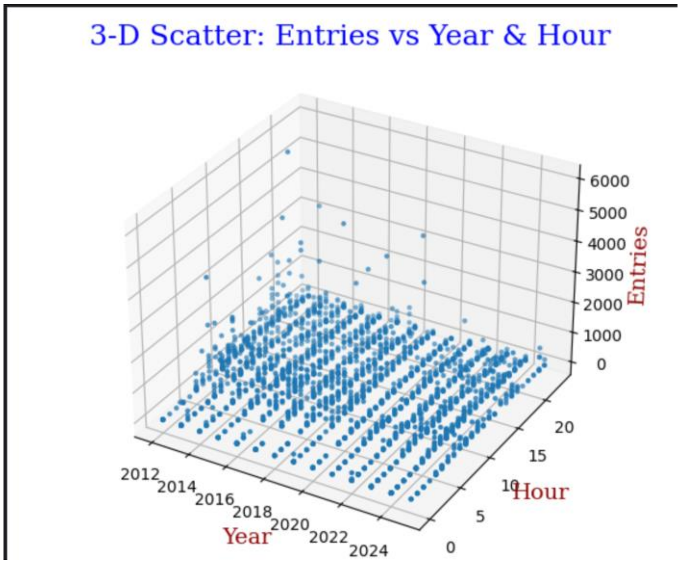
With the violin plot we see similar stuff to that of the box plots. this really shows how weekends do have lower rider ship on weekends however have a greater potential for outliers.



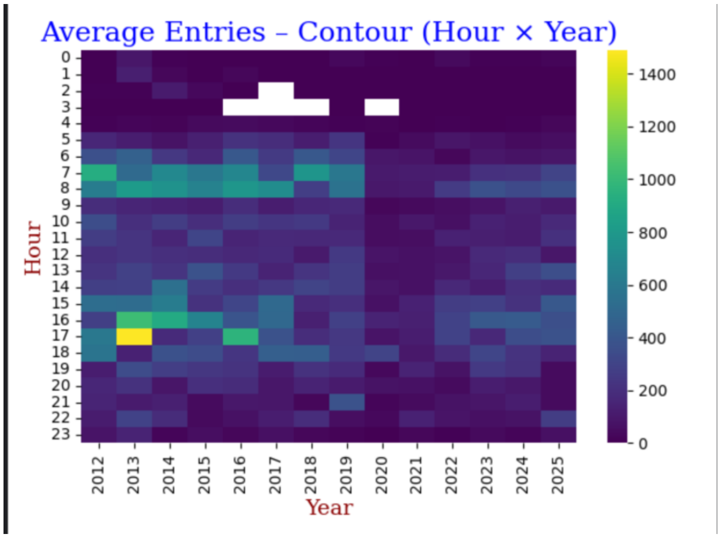
Joint plot of Entries and Tap Entries show how they are nearly 1 to 1 correlated but we can see in the weird shape of this graph that the right skew is still in effect here from all of those outliers.



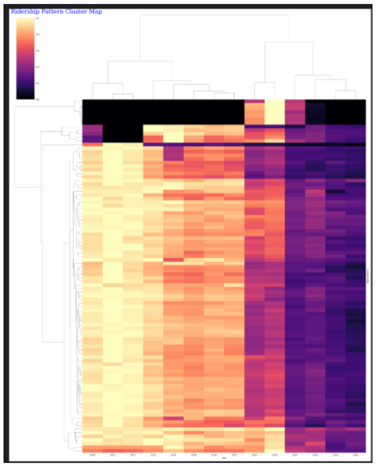
Rug plot here of Entries helping to show that the median of our data is closer to the left but we do still have those outliers out on the right skew dragging us out there from those big events like protests and election days.



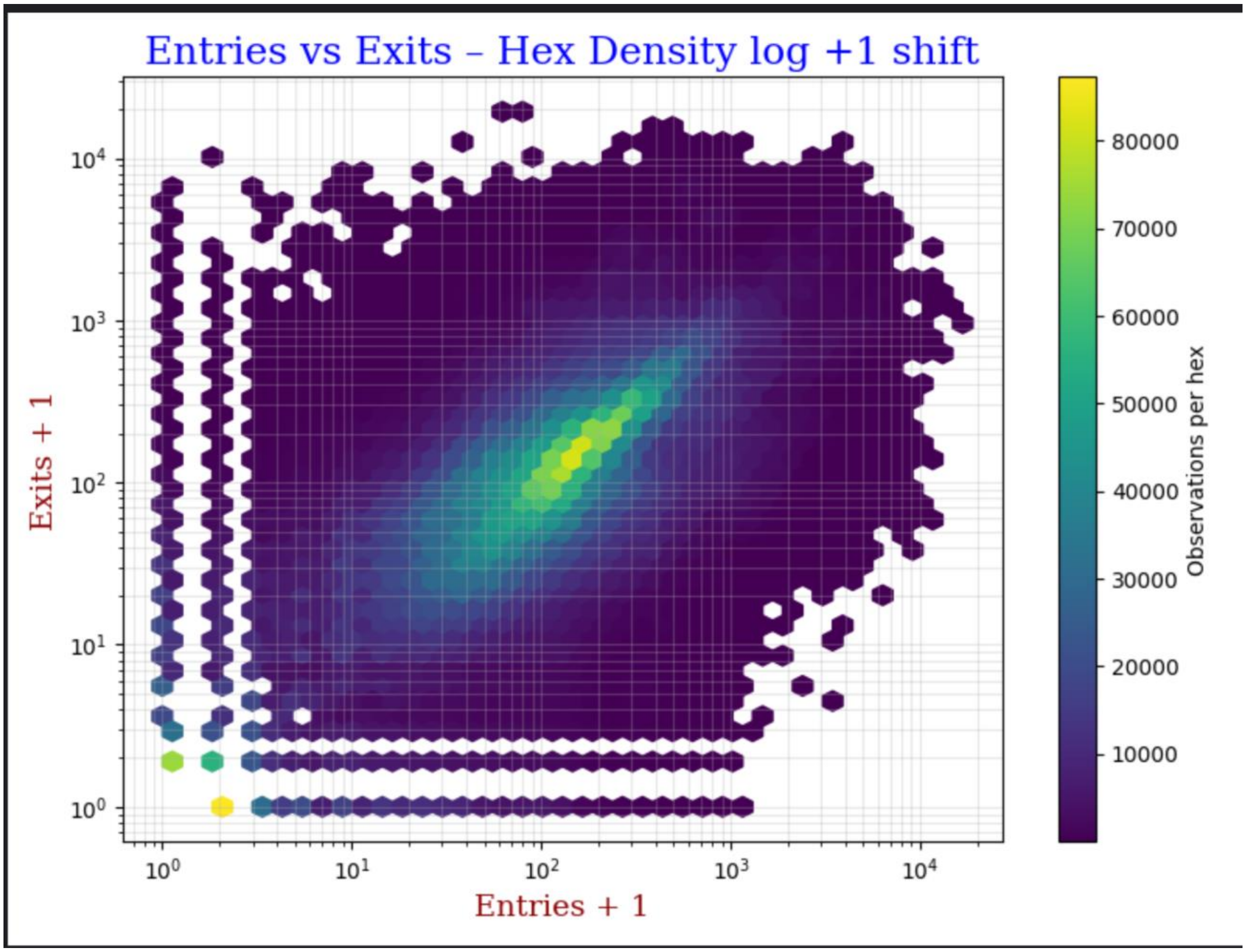
Our 3d plot shows us 3 things at once we can see that 2020-224 has less entries going on than 2012-2020. And that more outliers happened back then two. We can also see how entries are appear to make a bell curve through out the hours of the days being low in the morning and night and high around mid-day.



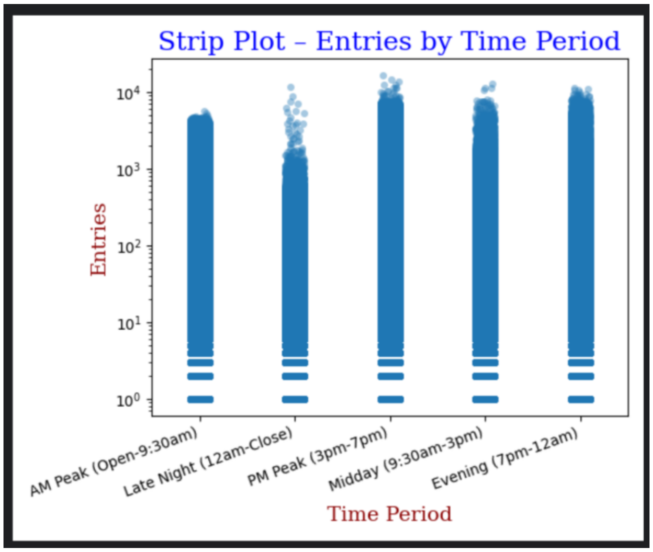
Contour plot of hour vs year shows just what I was talking about look at how early in the morning and late at night numbers are low but increase with pockets at morning rush hour and afternoon rush hour. And low traffic in between while people are at work. You can also see on the post covid years that morning and afternoon rush hour isn't as prominent probably due to people using the metro less for commuting and more for other things now that work from home is widespread.



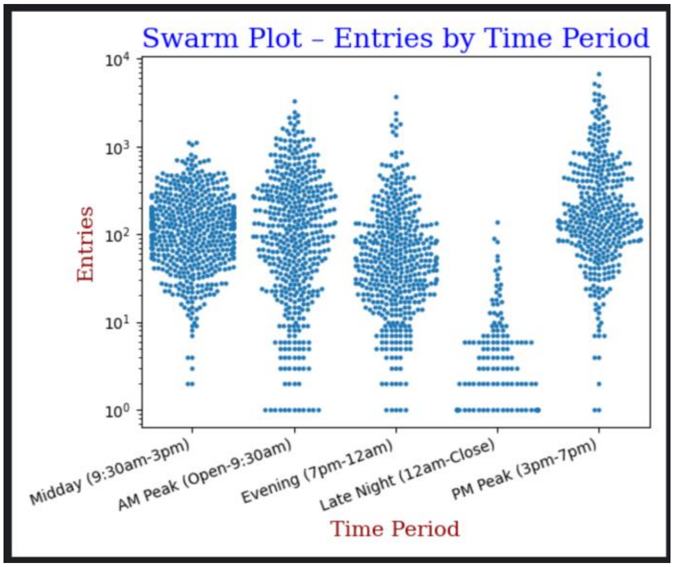
Really shows how less intense the post covid years are with ridership. And how much the metro was getting used pre covid.



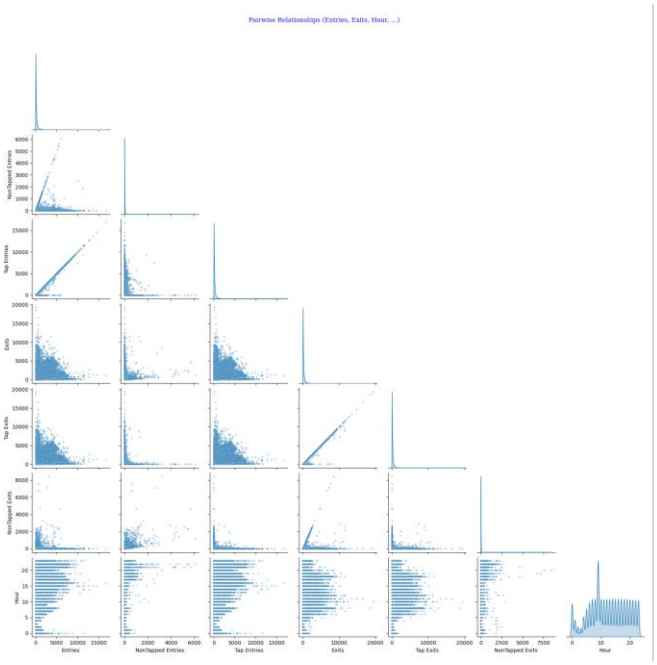
Turned into log scale to mitigate the outliers so that we can see that entries and exits positivity strongly correlated.



It appears late night has more outliers could be due to late night events being held like concerts and late night hokey games.

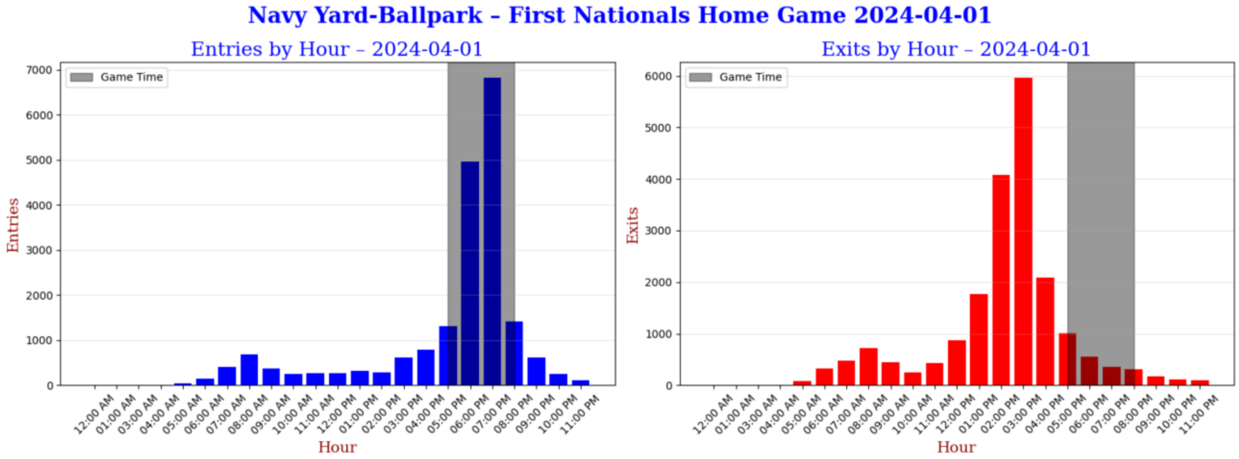


This plot shows how less late night is taken from the other times and how densely packed midday is and evenly spread of people leaving early or arriving late to work

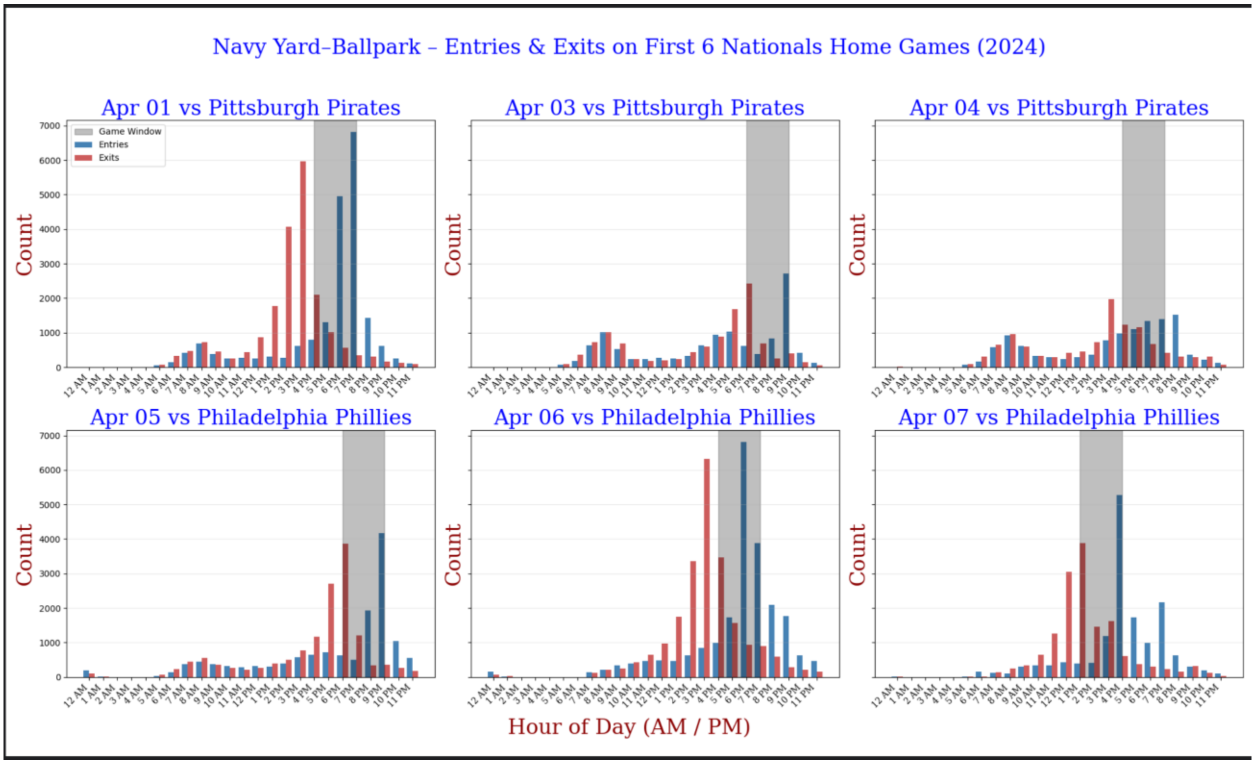


This pairwise plot shows the correlation between our numerical values matches that of what we saw in the heatmap plots with a KDE going down the diagonal.

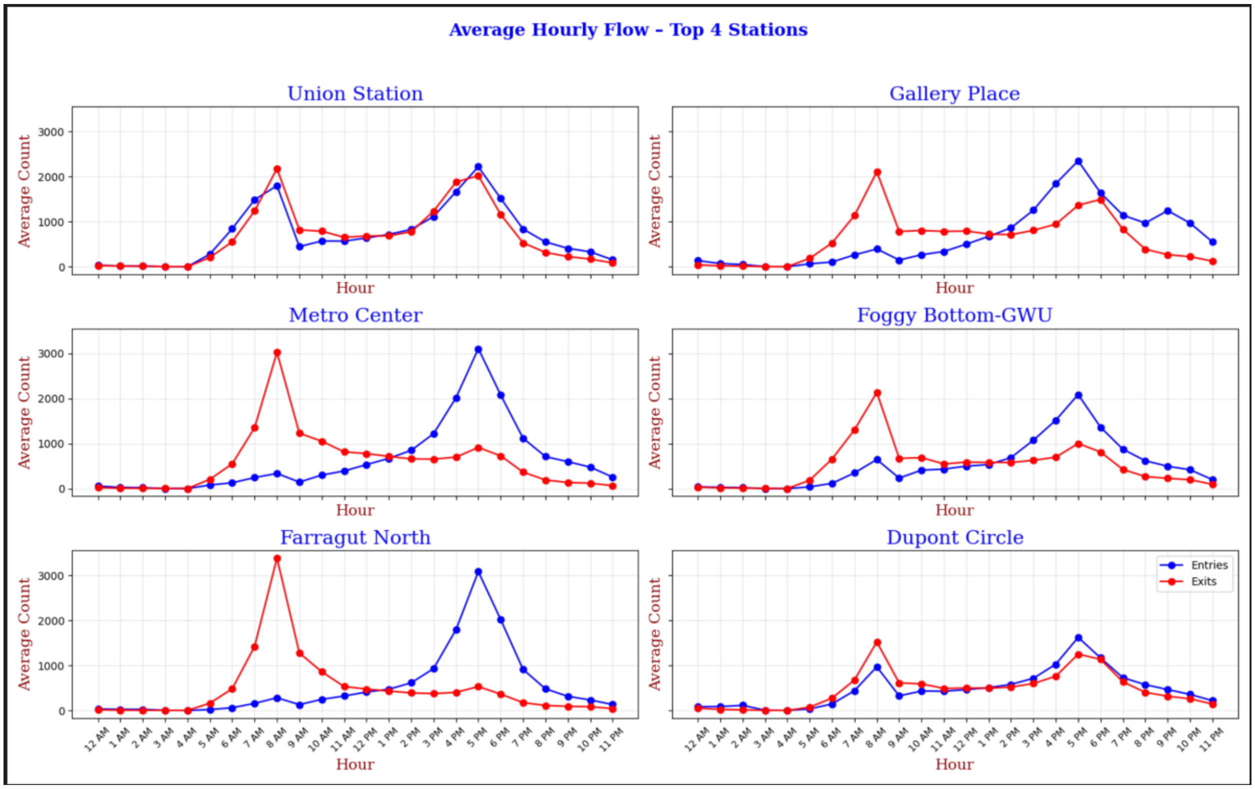
Subplots



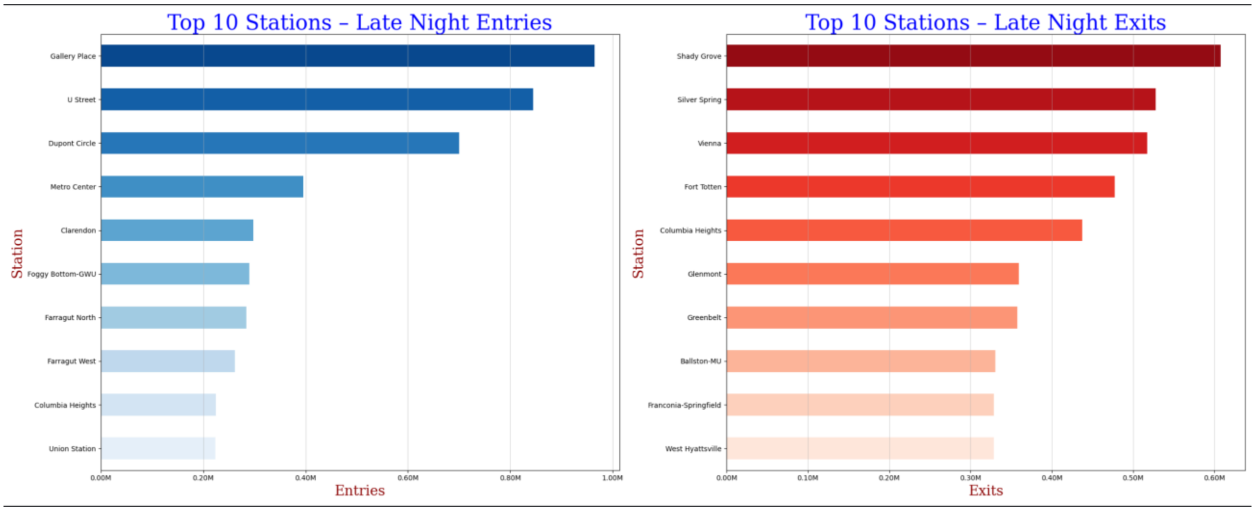
Shows how people get to the Games 30 mins early bu then leave during or before it ends



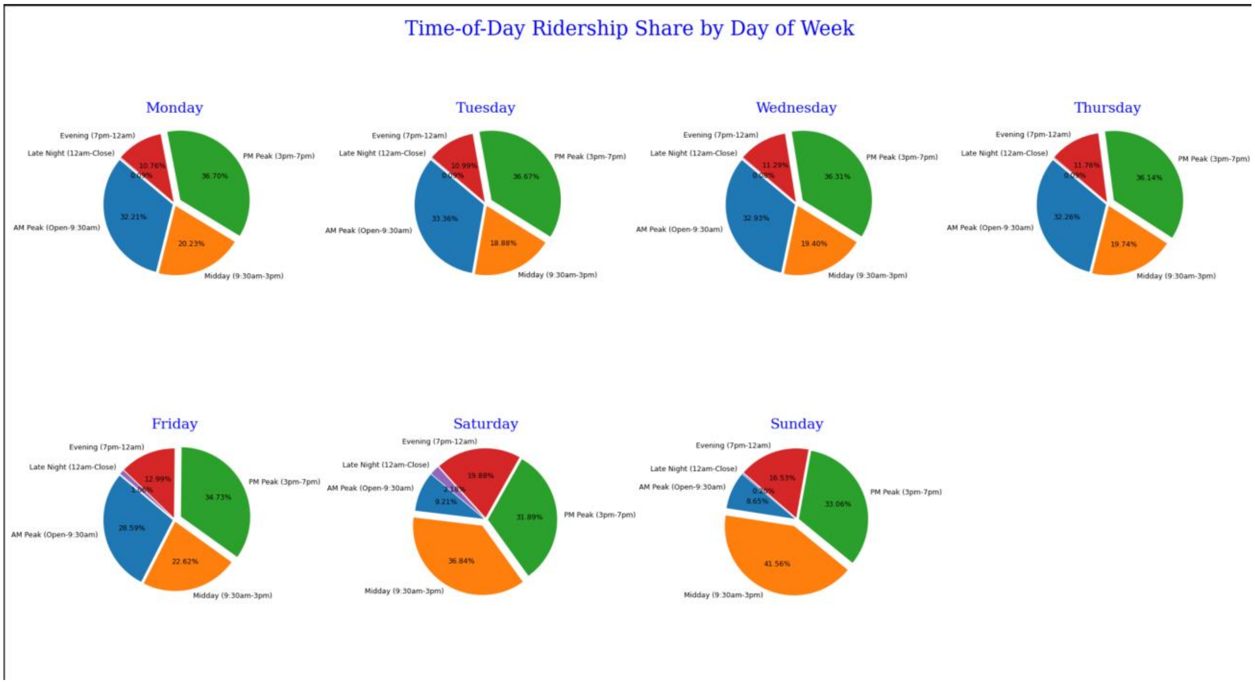
Shows multiple games to verify and be able to predict how early people arrive before games and when people start to leave the stadium can be used for traffic control.



Great for showing how some stations are just work horses like union station they are oncstanlty having people come and go while other stations like metro center and Farragut north are for people commuting to work not going to work you can see people get off there in the morning then get back on after work to go home



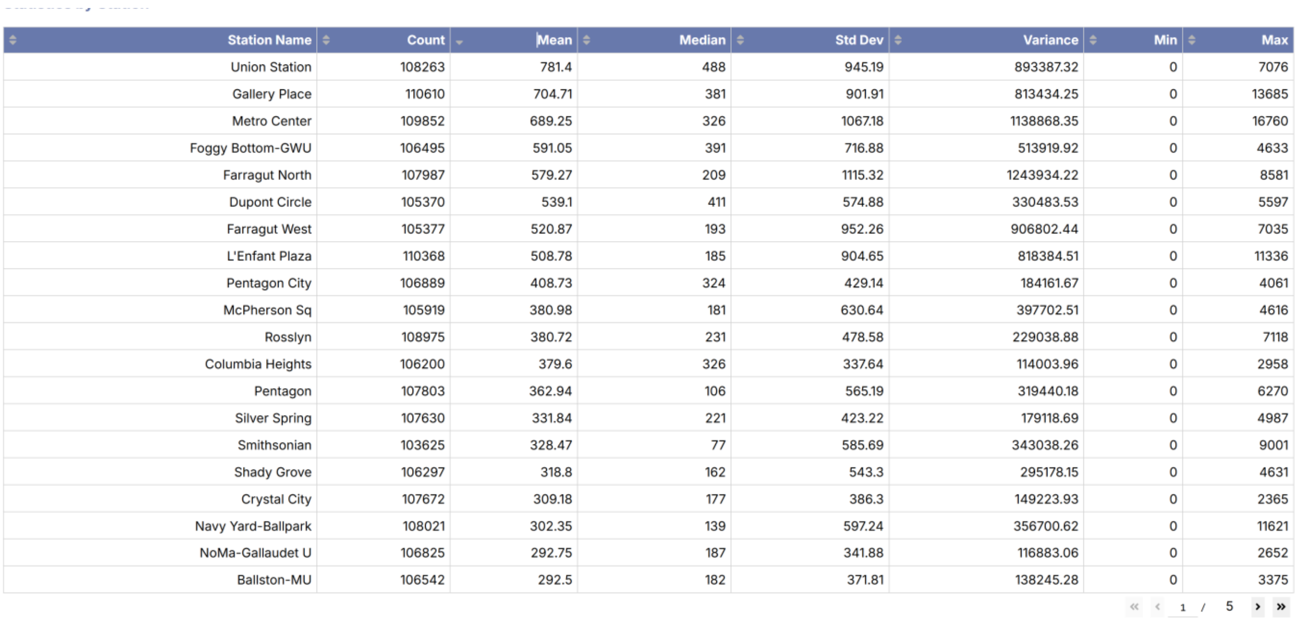
Shows how people are entry stations at late night are like popular bar spots but the stations on the right are late night exits so where people are going to bars -> edge of the metro . all of the top ones on the right are at the end of the metro line or have large parking lots



Show how weekdays mainly for pm peak during afternoon rush hour while the weekends most people are going during mid day to enjoy the day . we also see Friday and staruday are the few days that have late night percentage as that’s when people re going to bars

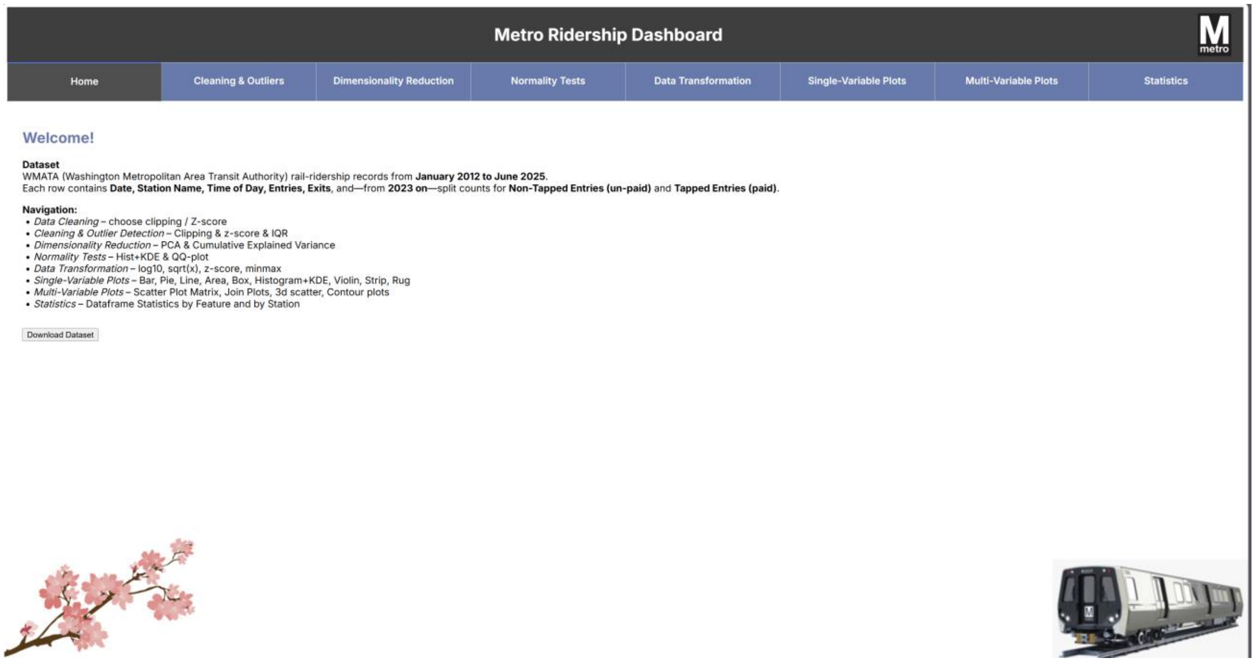
Tables

This shows the stations by station just the first page useful for knowing which station has the most or min riders or average .



Station Name	Count	Mean	Median	Std Dev	Variance	Min	Max
Union Station	108263	781.4	488	945.19	893387.32	0	7076
Gallery Place	110610	704.71	381	901.91	813434.25	0	13685
Metro Center	109852	689.25	326	1067.18	1138868.35	0	16760
Foggy Bottom-GWU	106495	591.05	391	716.88	513919.92	0	4633
Farragut North	107987	579.27	209	1115.32	1243934.22	0	8581
Dupont Circle	105370	539.1	411	574.88	330483.53	0	5597
Farragut West	105377	520.87	193	952.26	906802.44	0	7035
L'Enfant Plaza	110368	508.78	185	904.65	818384.51	0	11336
Pentagon City	106889	408.73	324	429.14	184161.67	0	4061
McPherson Sq	105919	380.98	181	630.64	397702.51	0	4616
Rosslyn	108975	380.72	231	478.58	229038.88	0	7118
Columbia Heights	106200	379.6	326	337.64	114003.96	0	2958
Pentagon	107803	362.94	106	565.19	319440.18	0	6270
Silver Spring	107630	331.84	221	423.22	179118.69	0	4987
Smithsonian	103625	328.47	77	585.69	343038.26	0	9001
Shady Grove	106297	318.8	162	543.3	295178.15	0	4631
Crystal City	107672	309.18	177	386.3	149223.93	0	2365
Navy Yard-Ballpark	108021	302.35	139	597.24	356700.62	0	11621
NoMa-Gallaudet U	106825	292.75	187	341.88	116883.06	0	2652
Ballston-MU	106542	292.5	182	371.81	138245.28	0	3375

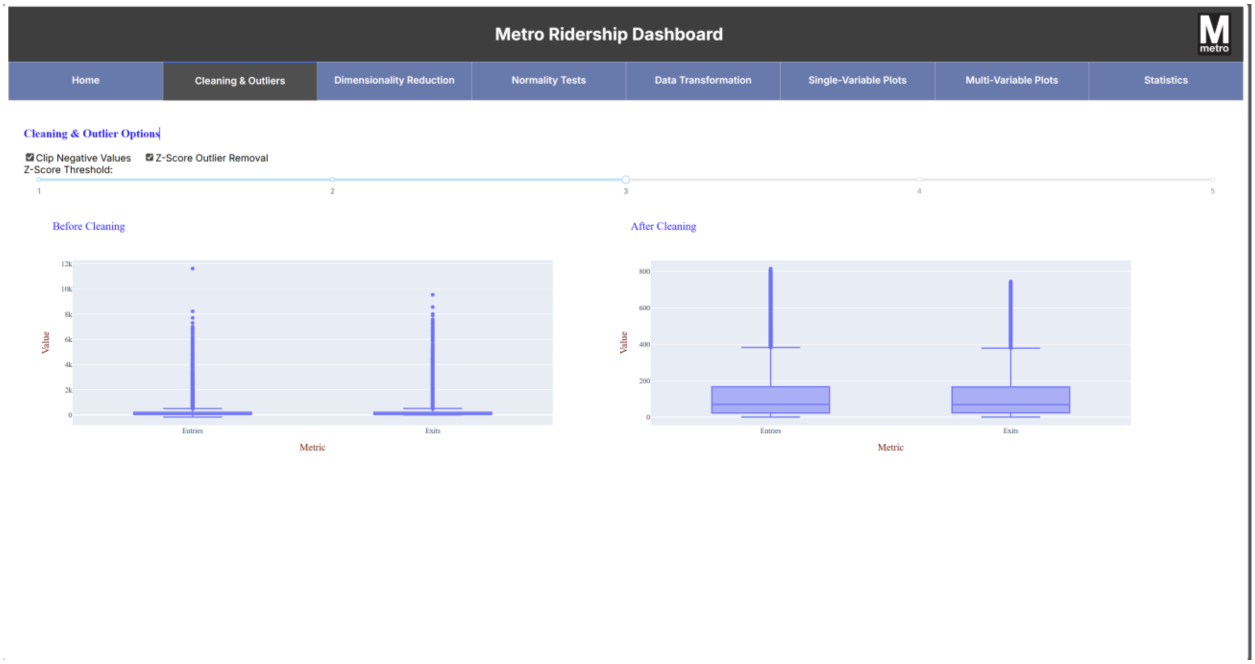
Dashboard



Welcome!

Dataset
WMATA (Washington Metropolitan Area Transit Authority) rail-ridership records from **January 2012 to June 2025**.
Each row contains **Date, Station Name, Time of Day, Entries, Exits**, and—from **2023 on**—split counts for **Non-Tapped Entries (un-paid)** and **Tapped Entries (paid)**.

Navigation:
- *Data Cleaning* – choose clipping / Z-score
- *Cleaning & Outlier Detection* – Clipping & z-score & IQR
- *Dimensionality Reduction* – PCA & Cumulative Explained Variance
- *Normality Tests* – Hist+KDE & QQ-plot
- *Data Transformation* – log10, sqrt(x), z-score, minmax
- *Single-Variable Plots* – Bar, Pie, Line, Area, Box, Histogram+KDE, Violin, Strip, Rug
- *Multi-Variable Plots* – Scatter Plot Matrix, Join Plots, 3d scatter, Contour plots
- *Statistics* – Dataframe Statistics by Feature and by Station

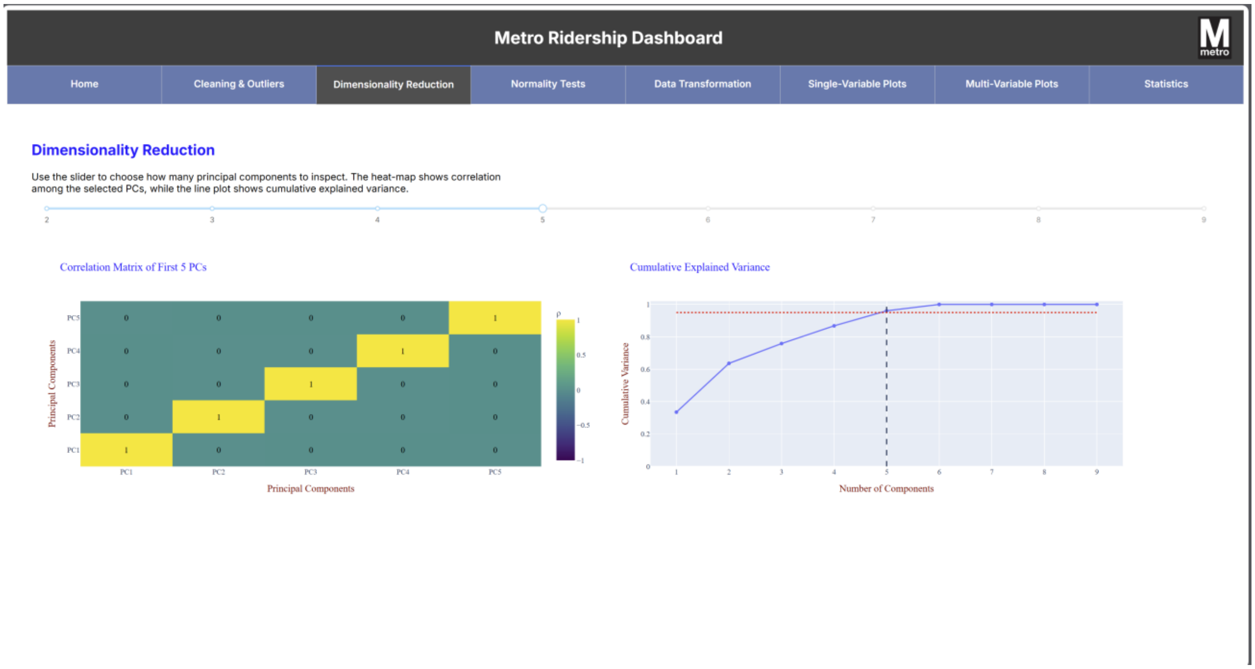


Metro Ridership Dashboard

Cleaning & Outlier Options

☑ Clip Negative Values ☑ Z-Score Outlier Removal

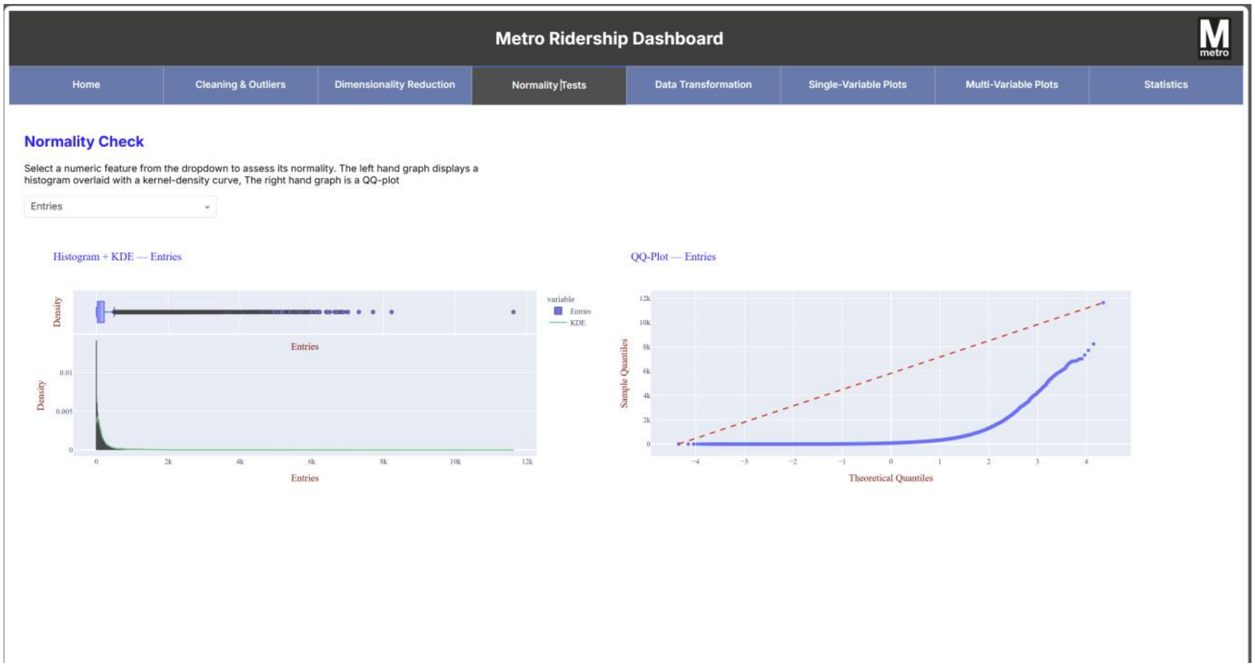
Z-Score Threshold: 3



Metro Ridership Dashboard

Dimensionality Reduction

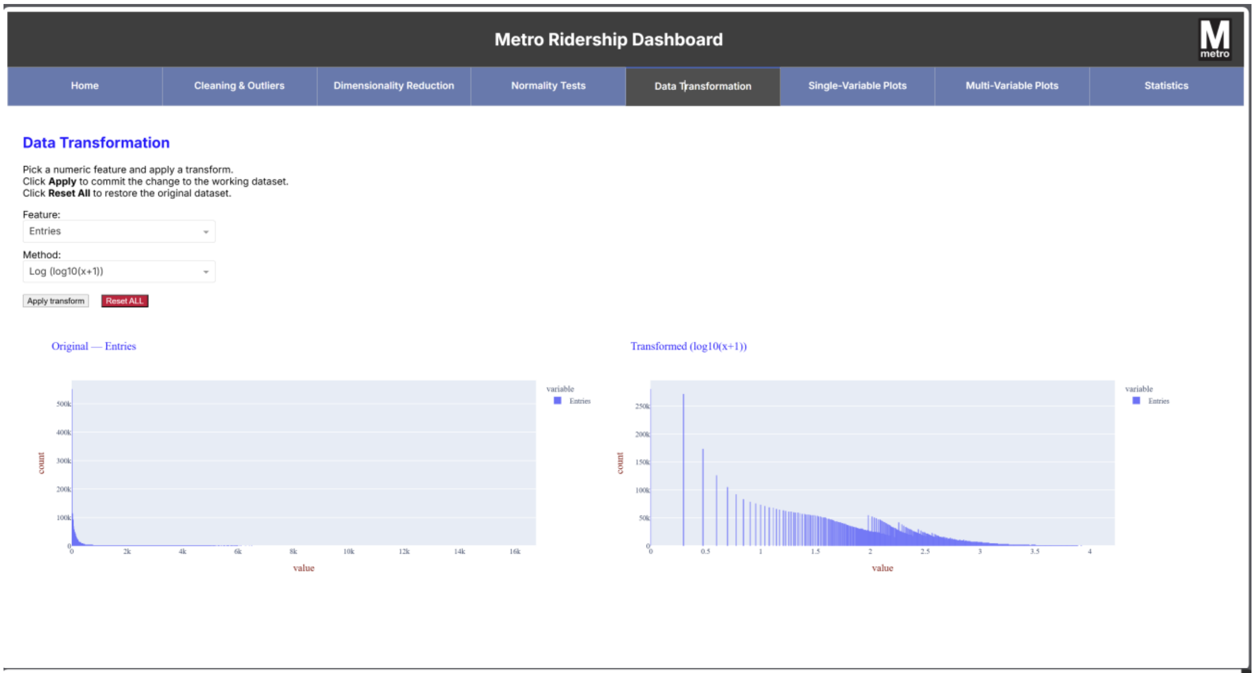
Use the slider to choose how many principal components to inspect. The heat-map shows correlation among the selected PCs, while the line plot shows cumulative explained variance.



Normality Check

Select a numeric feature from the dropdown to assess its normality. The left hand graph displays a histogram overlaid with a kernel-density curve, The right hand graph is a QQ-plot

Entries

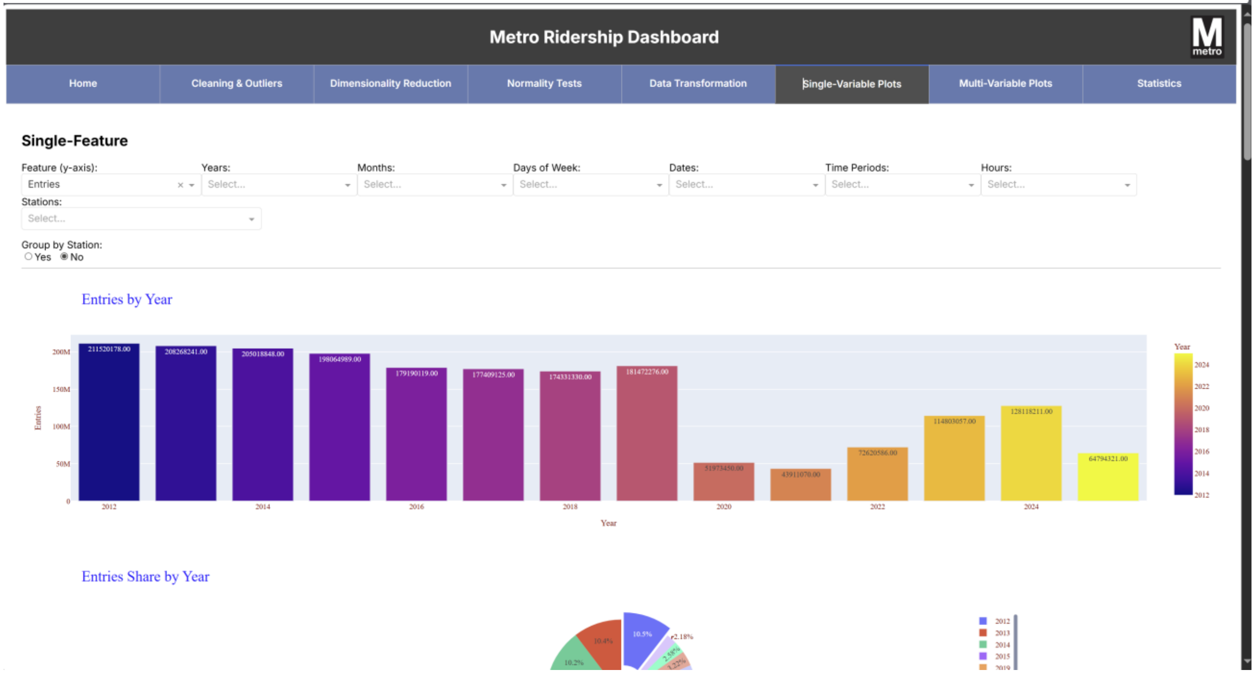


Data Transformation

Pick a numeric feature and apply a transform.
Click **Apply** to commit the change to the working dataset.
Click **Reset All** to restore the original dataset.

Feature: Entries

Method: Log (log10(x+1))



Single-Feature

Feature (y-axis): Entries

Years: Select...

Months: Select...

Days of Week: Select...

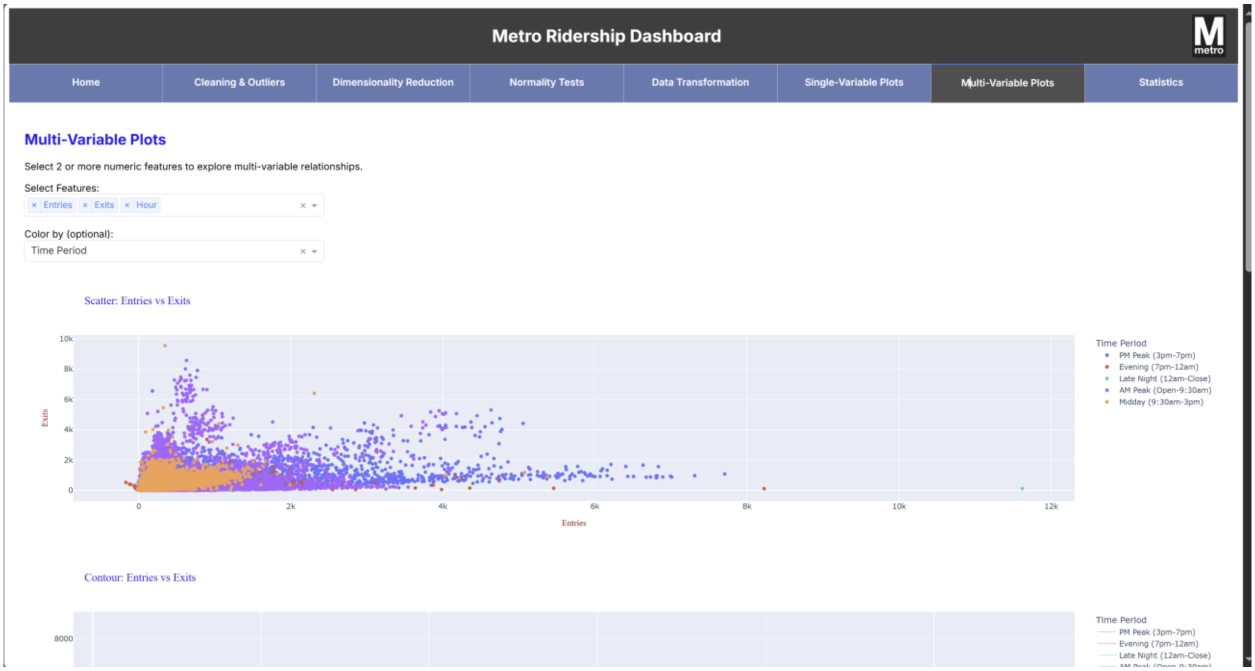
Dates: Select...

Time Periods: Select...

Hours: Select...

Stations: Select...

Group by Station:
○ Yes ● No

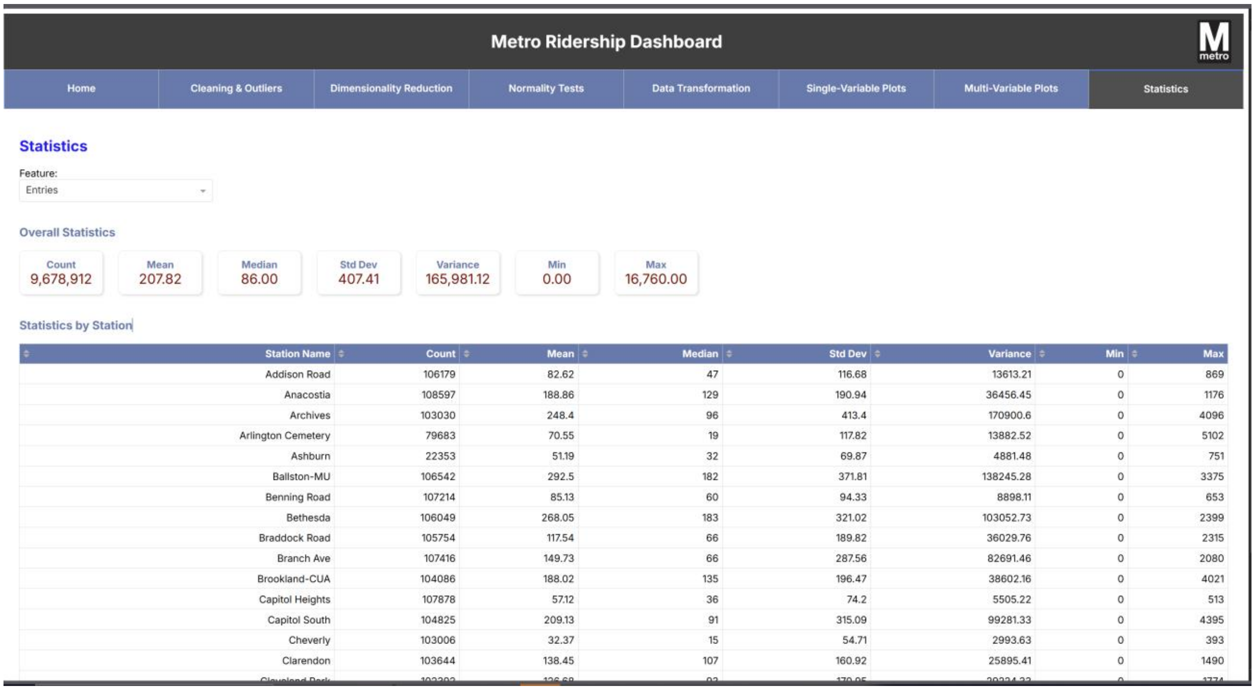


Multi-Variable Plots

Select 2 or more numeric features to explore multi-variable relationships.

Select Features: Entries, Exits, Hour

Color by (optional): Time Period



Statistics

Feature: Entries

Overall Statistics

Count	Mean	Median	Std Dev	Variance	Min	Max
9,678,912	207.82	86.00	407.41	165,981.12	0.00	16,760.00

Statistics by Station

Station Name	Count	Mean	Median	Std Dev	Variance	Min	Max
Addison Road	106179	82.62	47	116.68	13613.21	0	869
Anacostia	108597	188.86	129	190.94	36456.45	0	1176
Archives	103030	248.4	96	413.4	170900.6	0	4096
Arlington Cemetery	79683	70.55	19	117.82	13882.52	0	5102
Ashburn	22353	51.19	32	69.87	4881.48	0	751
Ballston-MU	106542	292.5	182	371.81	138245.28	0	3375
Benning Road	107214	85.13	60	94.33	8898.11	0	653
Bethesda	106049	268.05	183	321.02	103052.73	0	2399
Braddock Road	105754	117.54	66	189.82	36029.76	0	2315
Branch Ave	107416	149.73	66	287.56	82691.46	0	2080
Brookland-CUA	104086	188.02	135	196.47	38602.16	0	4021
Capitol Heights	107878	57.12	36	74.2	5505.22	0	513
Capitol South	104825	209.13	91	315.09	99281.33	0	4395
Cheverly	103006	32.37	15	54.71	2993.63	0	393
Clarendon	103644	138.45	107	160.92	25895.41	0	1490

Conclusion

From this project, I learned a lot about how people move through the metro system during the day — there are clear patterns in when people enter and exit, and each station plays a different role depending on the time and location. The Python plots were super helpful in visualizing this; it was much easier to understand the flow of passengers by seeing it graphically instead of just looking at numbers.

The dashboard is really user-friendly — I designed it so users can explore the dataset however they want, change graphs, and filter data to get their own insights. Building it with Dash and HTML elements made the process straightforward after some trial and error, and it gave me a lot of control over how everything looks and works.