



Final Report

Visualizing Divvy Chicago Bike Data: Redshift and Power BI Analytics

Team: Topic 4

Jayanth Chidananda	jchidananda@hawk.iit.edu	A20517012
Devin Wu	dwu38@hawk.iit.edu	A20484154
Harshali Gaikwad	hgaikwad@hawk.iit.edu	A20538035
Kalebu Patan	kpatan@hawk.iit.edu	A20446212

Instructor

Prof. Joseph Rosen

Course

Spring 2024 Big Data Technologies (CSP-554)

CONTENTS

INTRODUCTION.....	3
1.1 Abstract.....	3
1.2 Overview.....	3
1.3 Project Description.....	3
LITERATURE REVIEW.....	4
2.1 Amazon Redshift.....	4
2.2 Azure SQL Data Warehouse.....	4
2.3 Snowflake.....	4
2.4 MemSQL.....	4
DATA COLLECTION.....	7
3.1 Data Attributes.....	7
DATA ANALYSIS.....	8
4.1 Analytical Approaches.....	8
4.2 Infrastructure and Setup.....	8
4.3 Data Management and Querying.....	8
4.4 Analytical Queries.....	10
4.4.1. Popular Routes Analysis:.....	11
4.4.2. Monthly Ride Trends:.....	11
4.4.3. Peak Traffic station:.....	12
4.4.4. Peak Hour Traffic by Station:.....	13
DATA VISUALISATION.....	15
5.1 Performance Metrics Across Different Rideable Types, User Categories, and Days of the Week.....	15
5.2 Performance Metrics for Different Rideable for each Month.....	16
5.3 Performance Metrics for Sum and Average Time Duration by Month and each Weekday for different Bike Types.....	17
5.4 Key Performance Metrics for Total Rides by Started Time and for each Ride Type: Member and Casual.....	18
5.5 Key Performance Metrics for Each Bike Type on every Weekday.....	19
5.6 Key Performance Metrics for Percentage by Rider Type and Bike Type.....	19
5.7 Performance Metrics for Locations of Rides: Starting Position and Ending Position.....	20
FUTURE SCOPE.....	21
6.1 Expanding Service.....	21
6.2 Data-driven Optimization.....	21
CONCLUSION.....	22
REFERENCES.....	23

1.1 Abstract

This project involves examining data from the Divvy Chicago bike system through Amazon Redshift for data processing and Power BI for visualization. The primary goal is to analyze trip details such as start/end times, locations, and distances traveled to gain insights into user behavior and improve the bike-sharing system. Objectives include cleaning the data, smoothly loading it into Amazon Redshift, conducting in-depth analyses to identify popular stations, routes, and peak usage times, and ultimately, using Power BI to communicate actionable insights through engaging visualizations.

1.2 Overview

Since its inception in 2013, the Divvy Chicago bike system has flourished as a public bike-sharing initiative. Boasting over 800 stations and 6,000 bikes, it caters to a daily ridership exceeding 85,000 individuals. Its impact is evident in the reduction of traffic congestion and the enhancement of air quality, both attributed to the increased use of bicycles in Chicago. The extensive data generated by the Divvy system offers valuable insights into user behaviors and highlights opportunities for system enhancement.

1.3 Project Description

The primary goal of this project is to conduct a comprehensive visualization and analysis of Divvy bicycles usage data from the year 2023. The insights garnered from this project will not only serve educational purposes by shedding light on user behaviors but also provide critical data-driven insights that could be leveraged by businesses for strategic decisions.

The data utilized in this project comprises extensive records of Divvy bicycle usage throughout 2023. This data is stored securely in Amazon Cloud Storage, ensuring high availability and integrity. For processing and analysis, the data is then loaded into an Amazon Redshift cluster, a data warehouse solution that offers fast querying capabilities and scalability. The use of Amazon Redshift enables the efficient handling of large volumes of data through complex SQL queries, optimizing the performance and speed of data retrieval and analysis.

2.1 Amazon Redshift

Amazon Redshift is a well-established data warehousing solution within the AWS ecosystem. Its primary strength lies in its ability to handle large datasets and high-concurrency workloads efficiently. This scalability is complemented by flexible pricing models, allowing users to optimize costs based on usage patterns. Redshift's seamless integration with other AWS services simplifies data processing workflows, although it does require some management overhead for infrastructure provisioning and query optimization. Despite this, Redshift benefits from a mature ecosystem with extensive third-party support and resources, making it a popular choice for organizations leveraging AWS.

2.2 Azure SQL Data Warehouse

Azure SQL Data Warehouse is Microsoft's offering for scalable data warehousing in the Azure cloud. While it integrates well with other Azure services, providing a cohesive ecosystem for data processing, it may face limitations in handling extremely large datasets or high-concurrency workloads. However, its flexible pricing options and robust security features make it an attractive choice for organizations invested in the Azure ecosystem. Nevertheless, users may encounter a learning curve in navigating the Azure ecosystem and SQL Server technologies.

2.3 Snowflake

Snowflake stands out for its unique architecture, separating storage and compute to offer elastic scalability and performance. This architecture allows for automatic optimization and minimal management overhead, making it particularly attractive for organizations seeking a low-maintenance solution. Despite this, Snowflake's high performance and versatility make it a strong contender in the data warehousing landscape, although its ecosystem of integrations may not be as extensive as other platforms.

2.4 MemSQL

MemSQL differentiates itself with its in-memory processing capabilities, enabling real-time analytics and high-speed data ingestions. Its horizontal scalability and support for hybrid workloads make it a compelling choice for organizations with time-sensitive applications. However, its architecture and deployment may be complex, requiring expertise in distributed systems and database administration.

Let's dive deeper into the comparisons between Amazon Redshift, Snowflake, MemSQL, and Azure SQL Data Warehouse across various features:

1. Architecture:

1. Both Amazon Redshift and Snowflake employ architectures optimized for scalability and performance.
2. Amazon Redshift utilizes a shared-nothing architecture with separate compute and storage nodes, allowing for parallel processing of queries and seamless scalability.
3. Snowflake's architecture separates storage and compute, enabling on-demand scaling and efficient resource utilization without the need for manual provisioning or management.
4. MemSQL utilizes a distributed, in-memory architecture with both row-based and columnar storage formats, providing high-speed data processing and real-time analytics.
5. Azure SQL Data Warehouse utilizes a Massively Parallel Processing (MPP) architecture with separate compute and storage layers, allowing for scalable query processing and data storage.

2. Scalability:

1. Amazon Redshift and Snowflake offer highly scalable compute and storage resources, allowing organizations to handle large volumes of data and high-concurrency workloads effectively.
2. MemSQL provides horizontal scalability by adding additional nodes to the cluster, enabling linear scaling of performance and capacity.
3. Azure SQL Data Warehouse allows users to scale compute resources up or down based on workload demands, providing flexibility to accommodate changing usage patterns.

3. Performance:

1. Amazon Redshift and Snowflake are optimized for analytical workloads, offering high-performance query execution and efficient resource utilization.
2. MemSQL is designed for real-time analytics and high-speed data ingestions, providing low-latency query responses and high throughput.
3. Azure SQL Data Warehouse is capable of handling complex analytical queries, but its performance may vary compared to specialized solutions for certain workloads.

4. Cost-effectiveness:

1. Amazon Redshift, Snowflake, and Azure SQL Data Warehouse offer flexible pricing models, allowing organizations to pay only for the resources they use without any upfront costs or long-term commitments.
2. MemSQL's pricing may be perceived as higher compared to some other solutions, particularly for organizations with budget constraints or cost-sensitive projects.
3. Redshift, Snowflake, and Azure SQL Data Warehouse offer cost optimization features such as reserved instances and automatic scaling, helping organizations manage costs efficiently.

5. Maturity and Ecosystem:

1. Amazon Redshift and Snowflake benefit from mature ecosystems and support from their respective providers, AWS and Snowflake Inc., ensuring reliability, availability, and comprehensive support for various analytical tasks.
2. MemSQL's ecosystem may be perceived as smaller compared to more established platforms like Redshift and Snowflake, but it offers comprehensive solutions for real-time analytics and high-speed data ingestions.
3. Azure SQL Data Warehouse benefits from Microsoft's extensive ecosystem, including integration with other Azure services and third-party tools, providing a comprehensive platform for data analytics and insights generation.

6. Ease of Management:

1. Amazon Redshift, Snowflake, and MemSQL are fully managed services, handling routine administrative tasks such as hardware provisioning, software patching, and backups automatically.
2. Snowflake's architecture simplifies management by abstracting underlying infrastructure complexities, allowing organizations to focus on data analysis and insights generation.
3. Azure SQL Data Warehouse offers managed service features for automatic scaling and monitoring, reducing operational complexity and administrative overhead.

The project's dataset originates from Divvy's publicly accessible travel data for the year 2023, spanning from January to December. The data was obtained in CSV format to ensure compatibility and ease of use across platforms. Each dataset was meticulously organized using UTF-8 encoding to enhance cross-platform compatibility. With a substantial sample size, the dataset maintains data integrity, ensuring uniqueness, completeness, relevance, and proper citation of the data source.

Monthly datasets were collected as part of the Divvy Chicago data collection process, providing valuable insights into popular stations, usage trends, and key variables related to ride durations and distances. This paper outlines the key aspects of the data collection procedure and presents initial findings derived from the analysis.

3.1 Data Attributes

- Ride Information
- Ride Start and End Timestamps
- Start and End Station Details
- Ride Duration in Seconds
- Trip Distance
- Member Details
- Member Type (Casual, Member)
- Member/Casual Identification
- Geospatial Information
- Start and End Station Coordinates (Latitude, Longitude)

4.1 Analytical Approaches

The project employs a multifaceted analytical approach to dissect the Divvy data:

1. Temporal Analysis

- **Daily, Monthly, and Annual Trends:** This involves examining the frequency and patterns of rides across different times of the day, months, and the year. Such analysis is crucial for identifying peak usage times and seasonal trends, which assists in strategic planning and resource allocation.

2. Route Analysis:

- **High Traffic Routes:** Identifying the most frequented routes to better understand user preferences and potential bottlenecks.
- **Station Utilization:** Analysis of the most commonly used start and stop stations helps in optimizing bike and dock availability to meet user demand effectively.

3. Duration and Ride Type Analysis













- **Ride Duration:** Segmentation of rides into short and long durations to analyze usage intensity and preferences.
- **Rider Demographics:** Differentiating between member and guest usage to tailor marketing and operational strategies accordingly.

4.2 Infrastructure and Setup

Our data analysis is conducted on a provisioned Amazon Redshift cluster, specifically utilizing the '**big-data-cluster**' with dc2.large instances configured in a single-node setup. This configuration was chosen for its balance of performance and cost efficiency, ideal for handling the voluminous Divvy datasets.

4.3 Data Management and Querying

Data ingestion is streamlined via Amazon S3, where all Divvy ride data for 2023 is initially stored. The integration between Amazon S3 and Redshift is highly optimized, allowing for direct data loading from S3 into our Redshift database using the Query Editor v2. This process not only enhances the efficiency of data transfer but also simplifies the data management process.

<input type="checkbox"/>	Name ▲	Type ▼	Last modified ▼
<input type="checkbox"/>	 202301-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202302-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202303-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202304-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202305-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202306-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202307-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202308-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202309-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202310-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202311-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)
<input type="checkbox"/>	 202312-divvy-tripdata.csv	csv	April 1, 2024, 18:36:39 (UTC-05:00)

A dedicated database, named '**divvy-database**', was created within the connected cluster to host the ride data. We established a comprehensive table, '**rides_history**', to consolidate all the pertinent data for the year. This table houses approximately 5.7 million records, providing a robust dataset for detailed analytical operations.

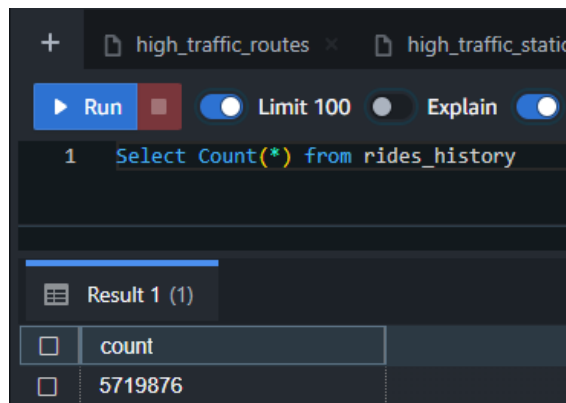


Table Design

The '**rides_history**' table is structured to accommodate detailed analyses of ride patterns, durations, and user demographics. This design facilitates efficient querying and data retrieval, which is pivotal for our subsequent analyses.

+
high_traffic_routes
high_traffic_stations_by_hour
Untitled 1

Run
Limit 100
Explain
Isolated session
big-data-cluster

```

1 SELECT column_name, data_type, is_nullable
2 FROM information_schema.columns
3 WHERE table_name = 'rides_history';
4

```

Result 1 (13)
Result 2 (1)

	column_name	data_type	is_nullable
<input type="checkbox"/>	end_lat	double precision	YES
<input type="checkbox"/>	start_lat	double precision	YES
<input type="checkbox"/>	member_casual	character varying	YES
<input type="checkbox"/>	end_lng	character varying	YES
<input type="checkbox"/>	start_lng	character varying	YES
<input type="checkbox"/>	end_station_id	character varying	YES
<input type="checkbox"/>	end_station_name	character varying	YES
<input type="checkbox"/>	start_station_id	character varying	YES
<input type="checkbox"/>	start_station_name	character varying	YES
<input type="checkbox"/>	rideable_type	character varying	YES
<input type="checkbox"/>	ride_id	character varying	YES
<input type="checkbox"/>	ended_at	timestamp without time zo...	YES
<input type="checkbox"/>	started_at	timestamp without time zo...	YES

4.4 Analytical Queries

The analysis primarily focuses on understanding ride patterns and station performance through a series of SQL queries:

4.4.1. Popular Routes Analysis:

- **Query:** Identifies the most frequented routes based on ride count data.
- **Insight:** The route between Streeter Drive and Grand Avenue, circling back to the starting point, emerges as the most popular, likely due to its proximity to Chicago's bustling pier area.

```
1  SELECT start_station_name, end_station_name, COUNT(*) AS ride_count
2  FROM rides_history
3  WHERE start_station_name IS NOT NULL AND start_station_name <> ''
4  |   AND end_station_name IS NOT NULL AND end_station_name <> ''
5  GROUP BY start_station_name, end_station_name
6  ORDER BY ride_count DESC
7  LIMIT 10;
```

Result 1 (10)

<input type="checkbox"/>	start_station_name	end_station_name	ride_count
<input type="checkbox"/>	Streeter Dr & Grand Ave	Streeter Dr & Grand Ave	10044
<input type="checkbox"/>	DuSable Lake Shore Dr &...	DuSable Lake Shore Dr &...	7572
<input type="checkbox"/>	Ellis Ave & 60th St	Ellis Ave & 55th St	6966
<input type="checkbox"/>	Ellis Ave & 60th St	University Ave & 57th St	6672
<input type="checkbox"/>	Ellis Ave & 55th St	Ellis Ave & 60th St	6405
<input type="checkbox"/>	University Ave & 57th St	Ellis Ave & 60th St	6250
<input type="checkbox"/>	Calumet Ave & 33rd St	State St & 33rd St	5474
<input type="checkbox"/>	State St & 33rd St	Calumet Ave & 33rd St	5374
<input type="checkbox"/>	Michigan Ave & Oak St	Michigan Ave & Oak St	5253
<input type="checkbox"/>	DuSable Lake Shore Dr &...	Streeter Dr & Grand Ave	5136

4.4.2. Monthly Ride Trends:

- **Query:** Aggregates ride counts by month to determine peak and off-peak periods.
- **Insight:** August is identified as the peak month for rides, attributed to favorable weather conditions, while January sees the least activity, reflecting the impact of harsh winter weather on ride frequencies.

+ high_traffic_routes x high_traffic_months x			
<div> <div>▶ Run</div> <div>Limit 100</div> <div>Explain</div> <div>Isolated session</div> <div>big-data-cluster</div> </div>			
<pre> 1 SELECT EXTRACT(YEAR FROM started_at) AS year, EXTRACT(MONTH FROM started_at) 2 FROM rides_history 3 GROUP BY year, month 4 ORDER BY ride_count DESC; </pre>			
Result 1 (12)			
<input type="checkbox"/>	year	month	ride_count
<input type="checkbox"/>	2023	8	771693
<input type="checkbox"/>	2023	7	767650
<input type="checkbox"/>	2023	6	719618
<input type="checkbox"/>	2023	9	666371
<input type="checkbox"/>	2023	5	604827
<input type="checkbox"/>	2023	10	537113
<input type="checkbox"/>	2023	4	426590
<input type="checkbox"/>	2023	11	362518
<input type="checkbox"/>	2023	3	258678
<input type="checkbox"/>	2023	12	224073
<input type="checkbox"/>	2023	2	190444
<input type="checkbox"/>	2023	1	190301

4.4.3. Peak Traffic station:

- Query: Determines the station where the highest number of rides originated.
- Insight: Streeter Dr & Grand Ave is the station from which the highest number of rides were initiated.

```

1  SELECT start_station_name, COUNT(*) AS ride_count
2  FROM rides_history
3  WHERE start_station_name IS NOT NULL AND start_station_name <> ''
4  GROUP BY start_station_name
5  ORDER BY ride_count DESC
6  LIMIT 10;

```

Result 1 (10)

<input type="checkbox"/>	start_station_name	ride_count	
<input type="checkbox"/>	Streeter Dr & Grand Ave	63249	
<input type="checkbox"/>	DuSable Lake Shore Dr &...	40288	
<input type="checkbox"/>	Michigan Ave & Oak St	37383	
<input type="checkbox"/>	DuSable Lake Shore Dr &...	35966	
<input type="checkbox"/>	Clark St & Elm St	35805	
<input type="checkbox"/>	Kingsbury St & Kinzie St	34966	
<input type="checkbox"/>	Wells St & Concord Ln	33590	
<input type="checkbox"/>	Clinton St & Washington ...	32715	
<input type="checkbox"/>	Wells St & Elm St	30407	
<input type="checkbox"/>	Millennium Park	30156	

4.4.4. Peak Hour Traffic by Station:

- **Query:** Determines the hour of the day when each high-traffic station records the most rides.
- **Insight:** The Clark St and Elm St station experiences its highest traffic at 5 PM, indicating a significant uptick in usage likely tied to evening commuting patterns.

▶ Run

Limit 100

Explain

Isolated session ⓘ

big-data-cluster ▾

divvy-database ▾

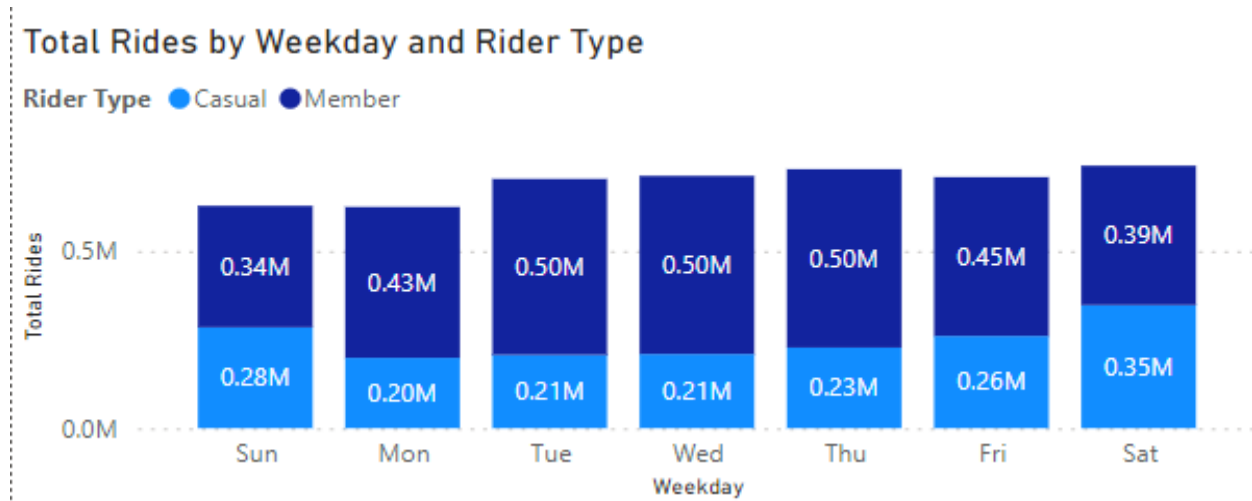
```
1 WITH TopStations AS (
2     SELECT start_station_name
3     FROM rides_history
4     WHERE start_station_name IS NOT NULL AND start_station_name <> ''
5     GROUP BY start_station_name
6     ORDER BY COUNT(*) DESC
7     LIMIT 10
8 ),
9 HourlyRideCounts AS (
10    SELECT ts.start_station_name,
11           EXTRACT(HOUR FROM rh.started_at) AS start_hour,
12           COUNT(*) AS ride_count,
13           ROW_NUMBER() OVER (PARTITION BY ts.start_station_name ORDER BY COUNT(*) DESC) AS rn
14    FROM rides_history rh
15    JOIN TopStations ts ON rh.start_station_name = ts.start_station_name
16    GROUP BY ts.start_station_name, EXTRACT(HOUR FROM rh.started_at)
17 )
18 SELECT start_station_name, start_hour, ride_count
19 FROM HourlyRideCounts
20 WHERE rn = 1
21 ORDER BY start_station_name;
```

Result 1 (10)

<input type="checkbox"/>	start_station_name	start_hour	ride_count
<input type="checkbox"/>	Clark St & Elm St	17	3464
<input type="checkbox"/>	Clinton St & Washington ...	17	6506
<input type="checkbox"/>	DuSable Lake Shore Dr &...	15	3990
<input type="checkbox"/>	DuSable Lake Shore Dr &...	19	3735
<input type="checkbox"/>	Kingsbury St & Kinzie St	17	4526
<input type="checkbox"/>	Michigan Ave & Oak St	17	4043
<input type="checkbox"/>	Millennium Park	17	3678
<input type="checkbox"/>	Streeeter Dr & Grand Ave	15	6802
<input type="checkbox"/>	Wells St & Concord Ln	18	3388
<input type="checkbox"/>	Wells St & Elm St	17	3392

These queries and the resulting insights provide a granular view of user behavior and service utilization, crucial for operational planning and strategic enhancements. The data-driven approach ensures that decisions are informed and targeted, aimed at improving user experience and operational efficiency. The integration of this analysis into visual reports via Power BI will further amplify the value of these insights, making them accessible and actionable for both internal stakeholders and external partners.

5.1 Performance Metrics Across Different Rideable Types, User Categories, and Days of the Week



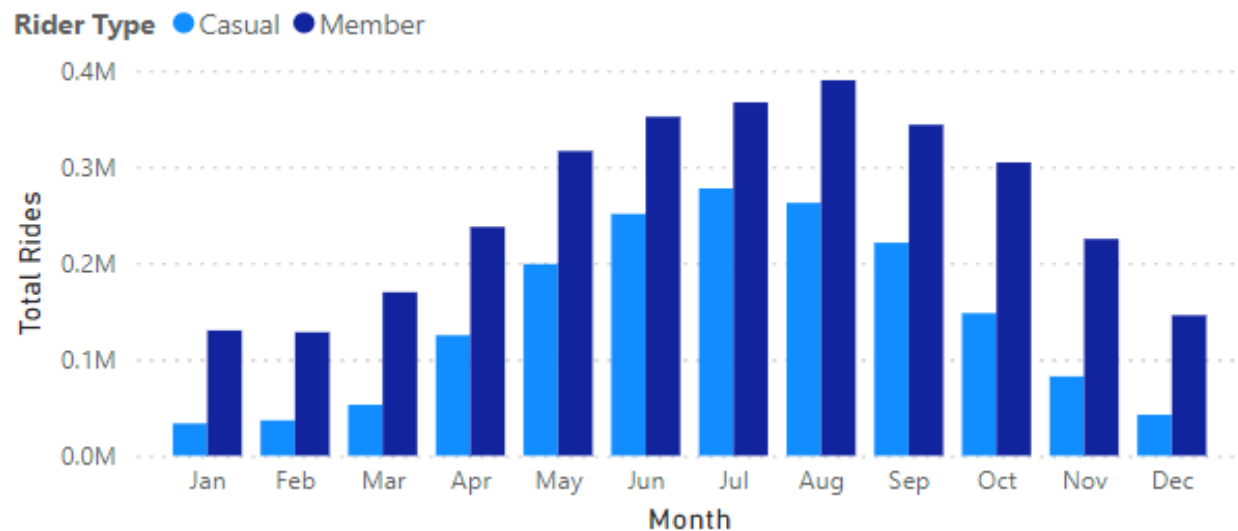
The graph shows the total rides by weekday and rider type for a bike-sharing service for Divvy Chicago. The bars are divided into two colors, blue representing casual riders and dark blue representing members.

A few key observations:

1. For all weekdays, the number of members exceeds the number of casual rides.
2. The busiest day for casual riders appears to be Saturday, while the busiest days for members are Tuesday through Thursday, likely reflecting commuter traffic.
3. Weekend days (Saturday and Sunday) have higher casual rider numbers compared to weekdays, suggesting more recreational usage.
4. The overall ride numbers seem to peak around the middle of the week (Wednesday and Thursday) due to the combination of both casual and member riders.

5.2 Performance Metrics for Different Rideable for each Month

Total Rides by Month and Rider Type



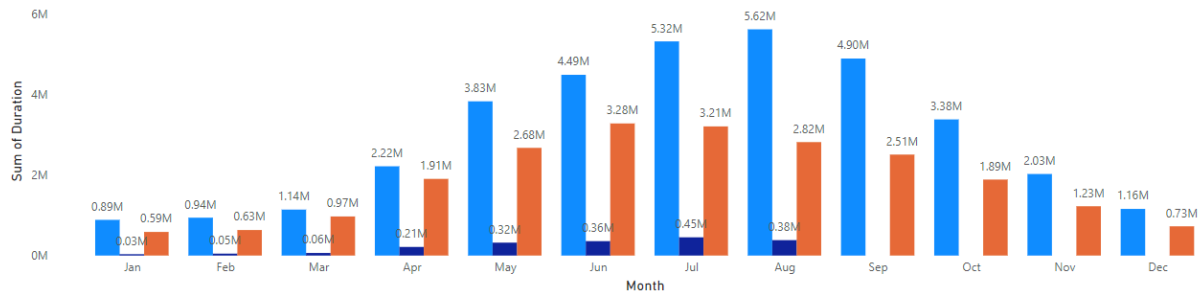
1. The overall ride numbers peak during the summer months (June, July, August), with July having the highest total rides for both casual and member riders. This pattern suggests increased usage during warmer weather.
2. Member/subscriber rides consistently exceed casual rider numbers across all months, indicating a strong subscriber base.
3. Casual rider numbers drop significantly during the colder months (November through March), while member rides remain relatively consistent throughout the year.
4. There appears to be a secondary peak in casual rider numbers during the fall months (September, October), potentially due to mild weather conditions.

The graph effectively illustrates the seasonal trends in bike usage, with a clear surge during the summer months driven by both casual and member riders, as well as the year-round consistent ridership from members/subscribers.

5.3 Performance Metrics for Sum and Average Time Duration by Month and each Weekday for different Bike Types

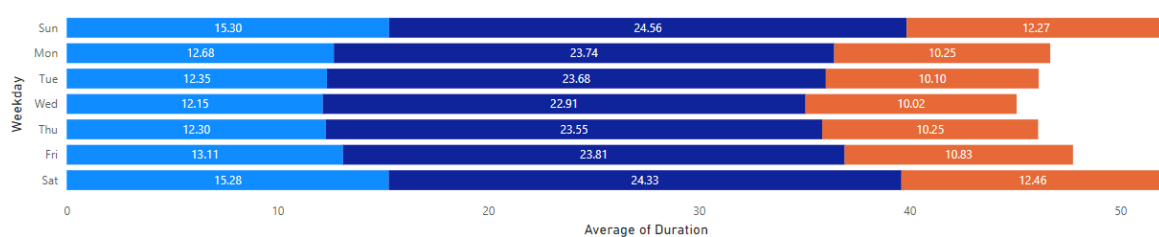
Sum of Duration by Month and Bike Type

Bike Type ● Classic Bike ● Docked Bike ● Electric Bike



Average of Duration by Weekday and Bike Type

Bike Type ● Classic Bike ● Docked Bike ● Electric Bike



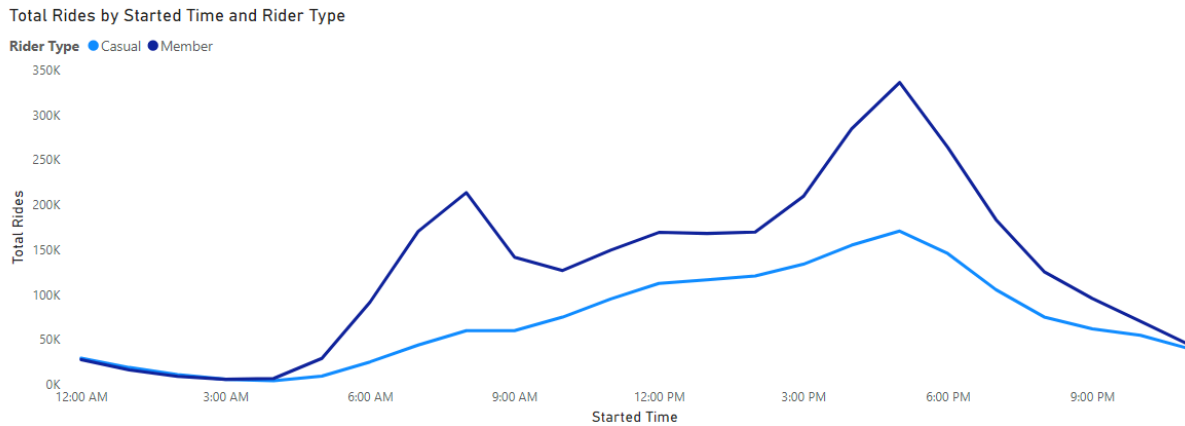
Sum of Duration by Month and Bike Type:

1. The top graph shows the total duration of bike rides by month and bike type: classic bikes, docked bikes, and electric bikes. Classic bikes have the highest total duration across all months, peaking at around 5.65M minutes in July.
2. Docked bike duration peaks in July at 3.21M minutes.
3. Electric bike duration is relatively low but sees a noticeable increase during the summer months, reaching 451K minutes in July.

Average Duration by Weekday and Bike Type:

1. The bottom graph displays the average duration of rides by weekday and bike type.
2. For classic bikes, the average duration is highest on Saturdays at 15.28 minutes and Sundays at 15.30 minutes, likely due to recreational use. Docked bikes have a relatively consistent average duration across weekdays, ranging from around 10-12 minutes.
3. Electric bikes show a longer average duration on weekends, with 24.33 minutes on Saturdays and 24.56 minutes on Sundays, compared to weekdays around 23-24 minutes.

5.4 Key Performance Metrics for Total Rides by Started Time and for each Ride Type: Member and Casual



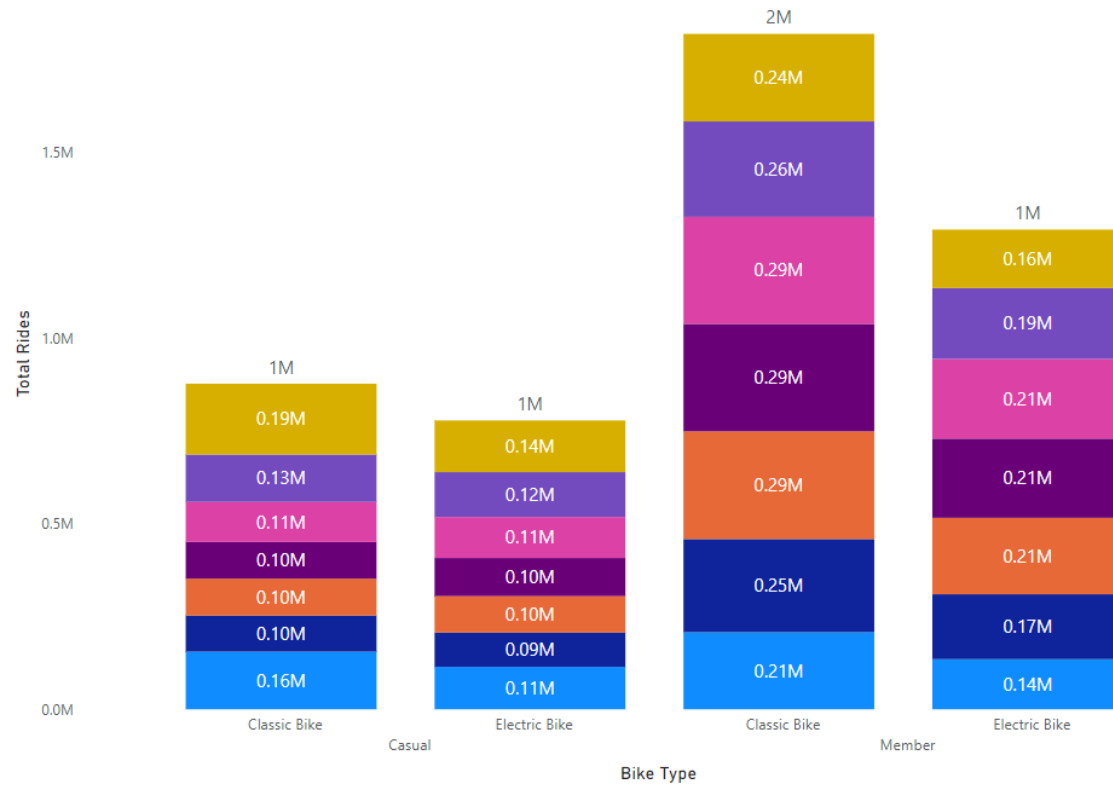
A few key observations

1. The first peak occurs around 8:00 AM, with around 275,000 total rides, likely representing commuters using the service to get to work or school in the morning. The member/subscriber rides (dark blue) substantially outnumber the casual rides (light blue) during this peak.
2. Usage drops after the morning peak but remains relatively steady until around 4:00 PM when it begins to rise again.
3. The highest peak occurs around 5:00 PM, with over 325,000 total rides. This peak is likely due to commuters leaving work or school in the evening. Again, member/subscriber rides dominate this peak.
4. After the evening peak, usage declines rapidly, with a smaller peak around 7:00 PM, potentially representing recreational or social rides.
5. Ride numbers are lowest during the late evening and early morning hours, with the lowest point being around 3:00 AM with only around 10,000 total rides.

5.5 Key Performance Metrics for Each Bike Type on every Weekday

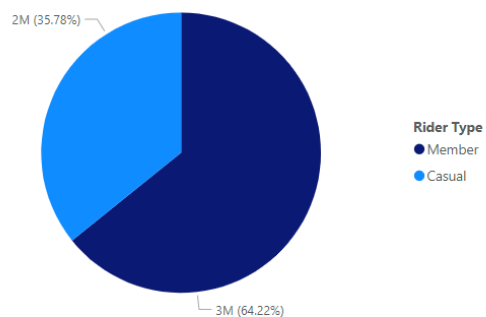
Total Rides by Rider Type, Member_casual and Day_of_Week

Weekday Sun Mon Tue Wed Thu Fri Sat

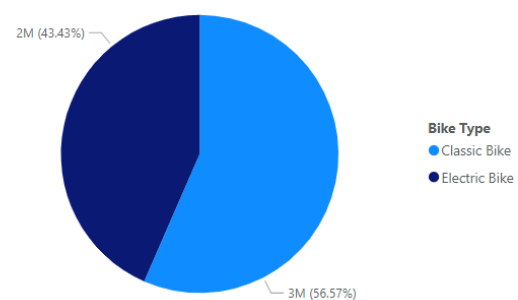


5.6 Key Performance Metrics for Percentage by Rider Type and Bike Type

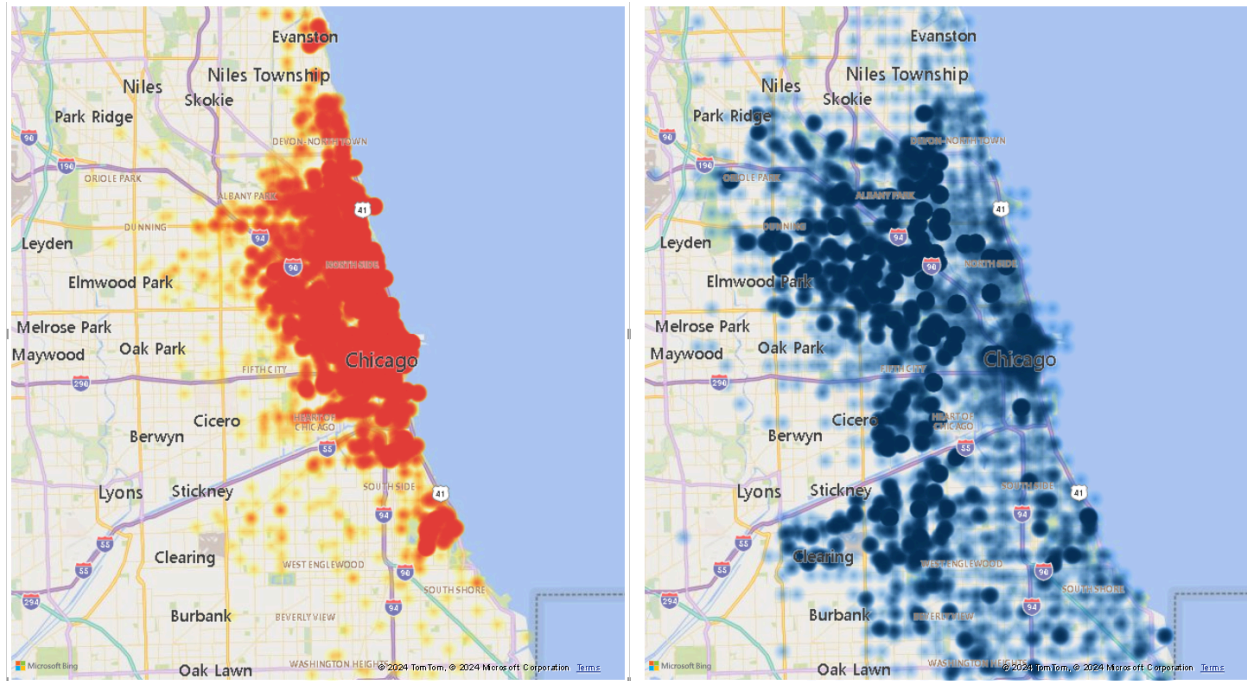
Percentage by Rider Type



Percentage by Bike Type



5.7 Performance Metrics for Locations of Rides: Starting Position and Ending Position



1. The heatmap on the left represents the starting positions of the rides. The heatmap on the right represents the ending positions of the rides.
2. Most bike trips begin within the center of the city as seen in the heatmap on the left but end at the stations further and more spread out from central Chicago.
3. The most popular starting station is Streeter Dr & Grand Ave (a station outside Navy Pier) with 59615 total rides beginning their ride from there.
4. The most popular ending station is also Streeter Dr & Grand Ave with 64197 total rides ending their ride there.
5. The most popular trip between two stations is Streeter Dr & Grand Ave to Streeter Dr & Grand Ave with 10044 total rides.

6.1 Expanding Service

To further enhance the accessibility and convenience of the Divvy bike-sharing service, we propose several strategic expansions.

1. **Increase Station Density:** Boost the number of stations in high-demand locations, major transportation hubs, and underdeveloped areas to improve service coverage and accessibility.
2. **E-bike Exclusive Stations:** In response to the increasing popularity of e-bikes, establish stations dedicated exclusively to these vehicles, ensuring availability and promoting the use of eco-friendly transportation options.
3. **Micro-mobility Integration:** Explore the inclusion of dockless bikes and e-scooters to diversify the micro-mobility offerings, catering to a broader range of urban transportation needs.

6.2 Data-driven Optimization

Leveraging advanced data analytics and machine learning will enable more informed decision-making and operational efficiencies.

1. **Demand Forecasting:** Utilize machine learning algorithms to accurately forecast demand patterns, optimize station placements, and effectively distribute resources across the network.
2. **User Surveys:** Conduct regular surveys to gather feedback from users about their preferences and expectations. This data will be invaluable for tailoring services to meet customer needs more effectively.
3. **Trend Monitoring and Responsive Adjustment:** Maintain vigilant monitoring of usage trends and perform ongoing analysis to adapt strategies dynamically. This will ensure the service remains responsive to user requirements and adaptable to changing urban mobility landscapes.

These recommendations aim to not only expand the service reach but also to refine operational strategies based on data-driven insights, fostering a more responsive and user-centric bike-sharing ecosystem.

This comprehensive study has provided invaluable insights into the usage patterns and operational dynamics of the Divvy bike-sharing system in Chicago. Utilizing Amazon Redshift for high-performance data processing and Microsoft Power BI for advanced data visualization, we have successfully analyzed and interpreted complex datasets to reveal detailed behaviors of Divvy users throughout 2023.

Our analyses uncovered several key trends and patterns:

- Temporal distribution highlighted significant seasonal variances, with **August** witnessing the highest ride volumes due to favorable weather conditions, and **January** the lowest, likely due to harsh winter conditions.
- Spatial analysis revealed that the route between **Streeter Drive** and **Grand Avenue** was particularly popular, underscoring its significance near major attractions like Chicago's pier.
- Peak usage times were strategically identified, aiding in better resource allocation during high-demand periods at specific stations.

These insights not only enhance our understanding of urban mobility but also serve as a vital tool for city planners, the Divvy system administrators, and business strategists looking to optimize operational efficiencies and user engagement.

Furthermore, the project demonstrated the power of integrating cloud storage and data warehousing technologies for scalable, efficient data analysis. Our use of a dedicated Redshift cluster enabled the handling of massive datasets with agility, while Power BI facilitated the translation of these data into actionable insights through compelling visualizations.

As urban mobility continues to evolve, the methodologies and findings from this project will contribute significantly to the ongoing improvement and sustainability of bike-sharing systems. The lessons learned here will also inform future projects that aim to harness big data for urban planning and transportation management. Our continued commitment to data-driven decision-making stands to revolutionize how cities approach transportation and mobility in an increasingly data-centric world.

- [1] Choi, Hanbit. "Divvy Bike Use Data Analysis and Recommendations." *SlideShare*, Slideshare, 15 Dec. 2018, www.slideshare.net/HanbitChoi1/divvy-bike-use-data-analysis-and-recommendations.
- [2] Devisangeetha. "Divvy Bike Share - EDA+ Network Analysis." *Kaggle*, Kaggle, 20 Aug. 2018, www.kaggle.com/code/devisangeetha/divvy-bike-share-eda-network-analysis.
- [3] "Divvy Data." *Home*, divvybikes.com/system-data. Accessed 30 Apr. 2024.
- [4] J. Zhang, X. Pan, M. Li and P. S. Yu, "Bicycle-Sharing System Analysis and Trip Prediction," 2016 17th IEEE International Conference on Mobile Data Management (MDM), Porto, Portugal, 2016, pp. 174-179, doi: 10.1109/MDM.2016.35.
- [5] Khan, Usman Aftab. "Exploratory Data Analysis: Cyclistic Bike-Share Analysis Case Study." *Medium*, CodeX, 15 Aug. 2022, medium.com/codex/exploratory-data-analysis-cyclistic-bike-share-analysis-case-study-1b1a00475a4f.
- [6] Krumlinde, Zakary. "Predicting Hourly Divvy Bike-Sharing Checkouts per Station." *Medium*, Towards Data Science, 31 May 2021, towardsdatascience.com/predicting-hourly-divvy-bike-sharing-checkouts-per-station-65b1d217d8a4.
- [7] Mazarei, Hos. "Data Analysis and Visualizations of Chicago Divvy Bikes Sharing." *LinkedIn*, 28 May 2022, www.linkedin.com/pulse/data-analysis-visualizations-chicago-divvy-bikes-sharing-mazarei/.