UBER SUPPLY-DEMAND GAP ANALYSIS

BUSINESS INSIGHTS ON UBER DATA | BY SANJANA KUMARI

PROBLEM STATEMENT

- Uber has been experiencing a significant number of ride failures, particularly during peak hours and at key pickup locations like the Airport and City center. These failures, which include cancellations due to driver unavailability and long wait times, indicate potential inefficiencies in demand forecasting, driver allocation, and service reliability.
- This project aims to analyze historical ride request data to identify patterns in demand and supply, uncover the underlying causes of trip failures, and provide actionable insights. The goal is to support data-driven decisions that improve operational efficiency, enhance customer satisfaction, and reduce failure rates across the platform.

PROJECT OBJECTIVES

- Analyze ride request data to identify trends in demand and supply.
- Determine time slots and locations with high failure rates.
- Categorize failure reasons and understand their root causes.
- Evaluate trip success patterns based on hour, weekday, and pickup point.
- Measure average trip durations and assess driver performance.
- Provide actionable insights to improve fleet availability and reduce cancellations.
- Support strategic decisions through data-backed recommendations.

TOOLS & TECHNIQUES USED

- **Excel**: Cleaned raw data, fixed timestamps, removed duplicates, and restructured columns to extract useful features for analysis.
- **Feature Engineering:** Enriched the dataset by generating new columns, including Time Slot (based on request hour), Trip Duration (for completed rides), Ride Outcome (Success/Failure), and Request Weekday – to support granular and insightful analysis
- **SQL**(MySQL Workbench): Derived core KPIs, explored request patterns by hour, weekday, and pickup location, and identified high-failure zones and demand-supply mismatches
- **Python**: Conducted detailed visual analysis using pandas, matplotlib, and seaborn to uncover temporal trends, trip duration patterns, and failure reasons; created diverse charts to support business insights and validate hypotheses.
- **Excel Dashboard**: Designed interactive visuals with slicers for Pickup Point, Ride Outcome, and Time Slot to enable dynamic filtering and support quick, data-driven decision-making.
- **PowerPoint**: Compiled insights into a presentation-ready format for stakeholders.

EXCEL PHASE

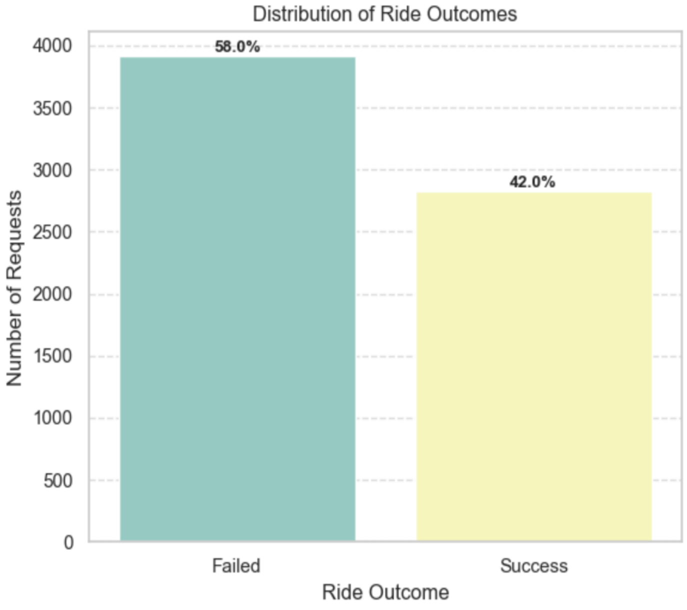
- Imported the raw Uber dataset and performed structural cleaning: removed duplicates, handled blank timestamps, and ensured format consistency for datetime fields
- Engineered key analytical features directly in Excel: calculated Trip Duration for completed rides, derived Ride Outcome based on status, categorized requests into Time Slots, and extracted Request Weekday
- Built supporting pivot tables to summarize key metrics: total requests, ride outcomes, success/failure rate, and average trip duration
- Used these pivot outputs to create early-stage visuals that shaped the analytical direction for SQL and Python phases
- Established the first layer of business insights, such as high-failure zones, hourly request surges, and pickup-location dynamics, which were later validated and explored further using SQL and Python

SQL PHASE

- Imported the cleaned dataset into MySQL Workbench and structured it for analysis.
- Derived core KPIs: 6,745 total requests, with 2,831 successful and 3,914 failed rides; average trip duration stood at 52.4 minutes.
- Identified peak demand during evening hours (5–9 PM) and on Fridays, confirming surge patterns.
- Analyzed pickup point performance: City had higher request volume (3,507), but Airport showed higher failure rates, suggesting driver unavailability or long wait times.
- Segmented failures by time slot and weekday; Early Morning and Evening slots showed critical shortages, especially at the Airport.
- Evaluated driver-level data: a large pool of drivers were active, but few handled majority of successful trips, indicating uneven distribution.
- SQL insights validated Excel findings and formed a robust base for deeper EDA and visual storytelling in Python.

VISUAL VALIDATION OF SQL FINDINGS

This chart illustrates the percentage split between successful and failed ride requests, highlighting that a majority (58%) of rides did not complete successfully.



This chart shows ride success vs failure by pickup point. While the City had more total ride requests, the Airport faced a higher failure rate.

PYTHON EDA PHASE

- Imported the cleaned dataset into Jupyter Notebook; used Pandas, Matplotlib, and Seaborn for in-depth analysis.
- Ride Outcome Analysis: Only 42% of rides were successful. Failures were higher from the City in the morning and Airport in the evening, aligning with SQL observations.
- Hourly Trends: Peak demand occurred between 5 PM and 9 PM, especially at the Airport, but with a sharp increase in failed rides due to no car availability.
- Weekday Trends: Thursday and Friday saw the highest failure rates, likely due to pre-weekend rush, indicating a need for better supply planning.
- Driver Analysis: Identified top 10 drivers contributing the most successful rides; majority had consistent performance patterns.
- Trip Duration: Average trip lasted ~52 minutes; duration was slightly longer for Airport pickups, with wide variance seen in City pickups.
- Failure Reason Breakdown: "No Cars Available" dominated Airport failures, while "Driver Cancelled" was more frequent in City during early hours.
- Visualizations confirmed key demand-supply patterns, aligning closely with SQL findings and enhancing stakeholder understanding through visual storytelling.

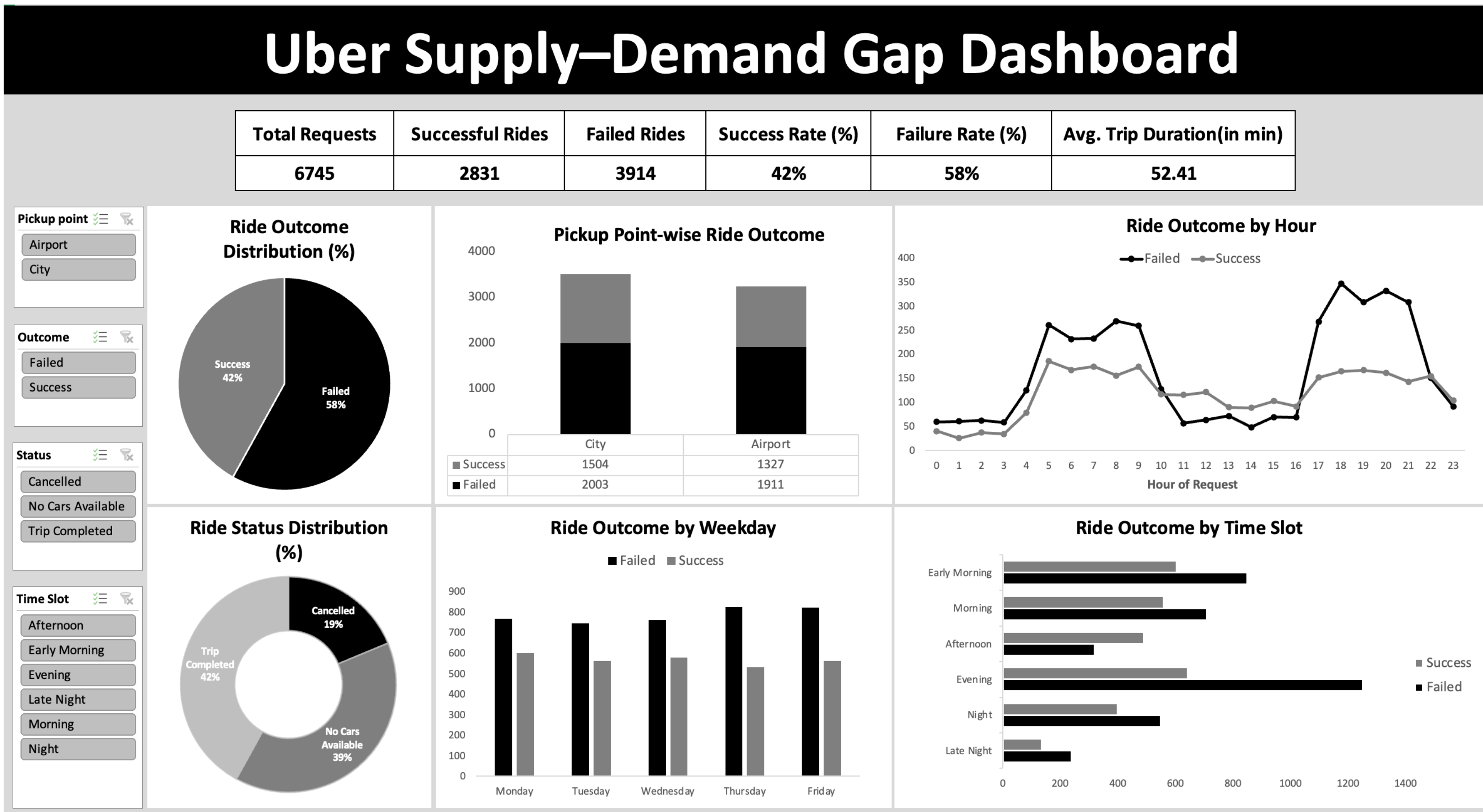
VISUAL DIAGNOSIS OF FAILURE CAUSES

KEY INSIGHTS

- 58% ride failure rate, mainly due to no cars available at high-demand hours.
- City requests fail more often in the morning, while Airport failures peak in the evening.
- 5 PM to 9 PM is the most critical window for demand-supply imbalance.
- Thursday and Friday see the highest ride failure rates.
- No Cars Available is the top failure reason, especially at the Airport.
- Top drivers contribute significantly to success rates; driver allocation impacts outcomes.

EXCEL DASHBOARD SNAPSHOT

Contains KPIs, slicers, 6 diverse charts to monitor Uber's performance visually.



RECOMMENDATIONS

- **Rebalance Fleet Supply**: Allocate more drivers to high-demand hours (especially 5–9 PM) and locations with recurring failures, like the **Airport during evenings** and **City during mornings**.
- **Driver Incentives**: Implement performance-based incentives during peak failure windows to improve availability.
- **Predictive Driver Deployment**: Use historical ride data to forecast demand by hour and pickup point for smarter fleet positioning.
- **Real-time Monitoring Dashboard**: Enable live tracking of request patterns to act quickly on emerging supply gaps.
- **Rider Communication**: Alert users during peak hours with expected wait times or surge pricing to manage expectations and reduce cancellations.
- **Optimize Onboarding**: Target driver recruitment in failure-prone zones to reduce 'No Cars Available' occurrences.

PROJECT FLOW SUMMARY

- Challenge Identified: Surge in unfulfilled ride requests across specific hours and zones.
- Data Preparation (Excel): Removed inconsistencies, engineered contextual fields (e.g., Time Slot, Duration).
- Exploratory Queries (SQL): Computed performance metrics, exposed temporal and spatial inefficiencies.
- Insight Discovery (Python): Visualized ride patterns and failure triggers through statistical plots.
- Interactive Reporting (Excel Dashboard): Enabled real-time filtering and strategic drilldowns.
- Final Documentation (PowerPoint): Synthesized findings into actionable insights for business impact.

CONCLUSION & NEXT STEPS

Conclusion :

- The project uncovered critical supply-demand gaps, pinpointed failure hotspots, and explained driver availability issues using structured data analysis and visual storytelling.

Next Steps :

- Redistribute drivers during peak failure time slots and high-demand zones
- Introduce real-time alerts for demand surges at the Airport
- Monitor KPIs regularly using the Excel Dashboard
- Expand the model to include weather, traffic, and holiday variables for deeper insight