**Technical Report: COVID-19 Hospital Utilization & Mortality Trends**

**1. Introduction and Problem Statement**

This project was done to better understand how hospitals coped with COVID-19 using data from the U.S. Department of Health and Human Services. We focused on things like which hospitals handled the most patients, which conditions had higher in-hospital death rates, and how patient visits changed over time. The idea was to build easy-to-read dashboards that could help spot trends in hospital usage and outcomes without getting into overly complicated analysis.

The pandemic exposed weaknesses in hospital systems, including overloaded emergency departments, unbalanced patient distribution across hospitals, and insufficient early warning metrics for high-risk conditions. Through this project, we tried to use publicly available data to create meaningful visualizations that provide a clearer picture of hospital usage and mortality during COVID-19, especially across the peak years of 2020 and 2021. By analyzing mortality trends, care-type demand, and condition-specific impacts, our dashboards aim to help hospitals prepare for future public health emergencies.

**2. Data Sources and Collection Methodology**

We used the publicly available dataset titled **"Effects of COVID-19 on Hospital Utilization Trends"** from **data.gov**. It came in multiple CSVs, and we only used the ones relevant to our analysis:

* hospital-utilization-trends.csv
* in-hospital-mortality-trends-by-diagnosis-type.csv
* in-hospital-mortality-trends-by-health-category.csv
* in-hospital-mortality-trends-by-secondary-diagnosis.csv
* utilization-trends-by-health-category.csv

The files were extracted and cleaned locally. We used Python to:

* Drop rows with junk like ########
* Parse date and create year, month, and quarter
* Standardize column names

We handled each file independently and avoided merging unrelated datasets. This helped us keep the data model clean and lightweight, especially for Power BI performance. After cleaning, we verified that each table had usable entries for count, date, category, and setting where applicable.

Data cleaning was a time-consuming step because some of the files were large and had inconsistent formatting. We also had to convert string-based month-year fields into proper date formats to enable accurate time-based filtering in Power BI. After ensuring all tables had unique keys or clear joins, we exported the final clean files to CSV and imported them into Power BI for modeling.

**3. Data Model Design and Implementation**

We followed a star schema model in Power BI with a few fact and dimension tables.

**Fact Tables:**

* Fact\_Utilization: Count of visits by facility, care setting, and date
* Mortality\_By\_Category: In-hospital deaths by health category and time

**Dimension Tables:**

* Dim\_Date: Auto-generated date table with year, month, quarter
* Dim\_Condition: Health category listing
* Dim\_Facility: Unique hospital names (from utilization table only)

The relationships were made on date and category fields only. We didn’t force connections between unrelated tables (like deaths and facilities) since there was no direct key.

We also created a few calculated columns in the Dim\_Date table to simplify time-based filtering and trend charts. These included MonthName, QuarterLabel, and Year-Month for clean X-axis labels in visuals. All relationships were many-to-one, and referential integrity was maintained. Our semantic layer supported filters on all visuals using shared keys across tables.

The model was kept intentionally simple due to time and complexity limits. No bridge tables or bi-directional relationships were used. The goal was to ensure stable visuals and performance even with slicers and drilldowns enabled.

**4. Visualization Approach and Tool Justification**

We used **Power BI** because it allowed us to build interactive dashboards with slicers and custom measures. It was also easy to format, design, and test visuals quickly.

We created **three dashboards**:

1. **Hospital Usage and Mortality Overview**: Showed total visits, death rates, and risky conditions using KPIs, stacked bar charts, donut charts, and a trend line.
2. **COVID-19 Condition Trends**: Focused just on COVID and how it compared to other health issues, using line charts, pie charts, and category-based filters.
3. **Facility-Level Analysis**: Highlighted which hospitals had the highest patient volumes over time using area charts, treemaps, and a breakdown table.

Color consistency was maintained throughout. Filters included year, condition category, care setting, and facility name. We avoided overloading any single dashboard with too many visuals and made sure that each one focused on a distinct analytical theme.

We also paid attention to layout structure by grouping KPIs on top, placing trend charts in the middle, and using filters or summaries at the bottom. Each visual was designed to support a specific question (e.g., Which conditions have the most deaths? Which hospitals had the most visits?) instead of being there just for display.

**5. Documentation of Advanced Calculations**

These are the actual DAX measures we used in our dashboards:

**1. Mortality Rate (%)**

Mortality Rate (%) =

DIVIDE(

SUM(Mortality\_By\_Category[count]),

SUM(Fact\_Utilization[count])

) \* 100

Used in KPI card on Dashboard 1.

**2. Total Visits**

Total Visits = SUM(Fact\_Utilization[count])

Used in most charts and cards.

**3. YoY Encounter Growth (%)**

YoY Encounter Growth (%) =

DIVIDE(

[Total Visits] - CALCULATE([Total Visits], DATEADD(Dim\_Date[Date], -1, YEAR)),

CALCULATE([Total Visits], DATEADD(Dim\_Date[Date], -1, YEAR))

) \* 100

Used as KPI in Dashboard 1 (e.g., 832.8%).

**4. Max Visits by a Hospital**

Max Visits by a Hospital =

CALCULATE(

MAXX(

SUMMARIZE(Fact\_Utilization, Fact\_Utilization[facility\_name], "VisitCount", SUM(Fact\_Utilization[count])),

[VisitCount]

)

)

Used in a card on Dashboard 3.

**5. Facility Load Icon**

Facility Load Icon =

SWITCH(TRUE(),

[Total Visits] >= 1000000, "🟢 High Volume",

[Total Visits] >= 500000, "🟠 Moderate",

"🔴 Low Volume"

)

This was planned for risk flagging but replaced with better visuals later.

**6. Analysis of Findings and Insights**

**From Dashboard 1:**

* Conditions like Stroke and Substance Use Disorders were leading in death counts.
* Emergency Departments and Ambulatory Surgery had higher mortality rates (~3%).
* Overall visits dropped significantly in 2020 but bounced back in 2021.

**From Dashboard 2:**

* COVID-19 completely dominated hospital visits in 2021.
* Inpatient Discharges carried most of the COVID load, meaning these patients were more severe.
* Visit spikes matched real-world COVID waves, especially in Q1 and Q3 2021.

**From Dashboard 3:**

* Kaiser Fontana had the highest total visits (~201K), far above the median.
* A few hospital groups consistently appeared in the top 10 across months.
* Monthly visits by setting showed Emergency Department traffic rising steadily during recovery phases.

We made sure each dashboard helped answer a different question. Dashboard 1 gave the high-level overview, Dashboard 2 zoomed in on a single condition, and Dashboard 3 gave a bottom-up look at individual hospital volume

**7. Challenges Encountered and Solutions Implemented**

|  |  |
| --- | --- |
| **Challenge** | **What We Did** |
| Dataset had ######## cells | Cleaned with Python before import |
| No link between death data and facilities | Avoided fake joins; analyzed at category/setting level only |
| Slicer responsiveness | Kept slicers minimal: year, category, setting |
| Blanks in KPI visuals | Rebuilt logic using valid DAX scope and tables |
| Too much clutter in one dashboard | Split the visuals across 3 dashboards with distinct themes |
| Lack of patient-level data | Focused on trends and aggregates instead of per-patient analysis |

We learned how important it is to design the model first before jumping into visuals. Our early versions had performance problems due to unnecessary relationships and filters.

**8. Future Enhancements**

* Add a geographic view (e.g., map) to show state or region-level distribution
* Include patient demographics like age or insurance type for deeper analysis
* Explore more advanced forecasting like ARIMA or exponential smoothing for trends
* Connect with live data using Power BI Service and schedule refreshes
* Add calculated columns for day-of-week and weekday/weekend traffic split
* Add tooltips with explanations for KPIs and rate formulas for better interpretation