ELECTVIZ- ELECTION DATA VISUALIZATION FOR MEDIA

INFOSYS SPRINGBOARD INTERNSHIP 6.0

Milestones Summary Report

By – A Archana (Team A)

Milestone – 1

1.Project Objective:

The primary objective of the ElectViz project is to analyze and visualize complex data from Indian state elections using Power BI. The goal is to transform vast and raw election data into a clear, interactive, and easily understandable dashboard.

This tool is designed specifically for media organizations, political analysts, and the public to explore voting patterns, party performance, and historical trends without needing to sift through complex spreadsheets.

The main goals of this project are:

1. To consolidate and analyze key election metrics, including total votes, party participation, seats won, and voter turnout.
2. To identify historical voting trends by visualizing data across different election years.
3. To provide a comparative analysis of political party performance at both the state and national levels.
4. To create a detailed breakdown of seat distribution by state and constituency type.
5. To utilize Power BI's interactive features (slicers, filters, and drill-downs) to empower users to find specific insights dynamically.
6. To provide a data-driven tool for media to support accurate reporting and insightful political commentary.

2.Project description in detail:

The ElectViz project was developed as a part of Infosys Springboard Virtual Internship 6.0 to explore how data visualization can enhance political and electoral reporting. Our team used the Election dataset from Kaggle to create a Power BI dashboard that highlights national and regional election results, party performance comparisons, alliance analytics,

and voter behavior patterns.

3.Overview and Purpose:

In the fast-paced media landscape, especially during election cycles, there is a critical need for quick, accurate, and compelling data insights. Indian elections are famously large and complex, involving thousands of parties and billions of votes across decades.

The ElectViz project addresses this need by creating a single, unified Power BI dashboard. It provides a comprehensive overview of Indian state election history, filtered by year, state, party, and constituency. The dashboard allows media users to instantly identify key statistics, spot trends, and compare party performance, enabling them to craft data-driven stories and reports.

Dataset Description

The dataset used for this project is a comprehensive, granular collection of Indian state election results. Based on the raw data screenshot provided, the dataset includes the following key fields:

- st_name: The state where the election was held (e.g., Maharashtra).
- year: The year of the election (e.g., 1985).
- ac_no: The unique assembly constituency number.
- ac_type: The type of constituency (e.g., GEN for General) .
- cand_name: The name of the candidate.
- cand_sex: The gender of the candidate (e.g., M).
- partyname: The full name of the political party (e.g., Independent).
- partyabbre: The abbreviation for the political party (e.g., IND).
- totvotpoll: The total votes polled by the specific candidate.
- electors: The total number of eligible voters in the constituency.
- result: The outcome for the candidate (e.g., won, lost).
- Seats Won: A numeric column indicating if a seat was won (1 for won).

4.Tools and Technologies Used:

1. Microsoft Power BI Desktop: The primary tool used for data modeling, analysis, and creating all dashboard visualizations.
2. Power Query Editor (in Power BI): Used for data cleaning, transformation, and preparing the raw data for analysis (e.g., managing 1,637 party names, standardizing formats).
3. DAX (Data Analysis Expressions): Used to create key measures and KPIs, such as Total Votes, Total Parties, Total Seats, and Voter Turnout %.
4. Microsoft Excel: Likely used as the initial data source for collecting and formatting the raw election data before importing it into Power BI.
5. Canva & PowerPoint – for presentation design and visual enhancement.

5.Requirement Analysis:

The ElectViz project aims to take the **Indian State Election dataset**, clean it, analyze it, and create simple visualizations suitable for media reporting.

Before starting the analysis, the following requirements were identified:

a) Understanding the Dataset

- The raw dataset contained columns such as *state name, year, constituency number, candidate name, party name, total votes polled, electors, and result (won/lost)*.
- These fields needed to be checked for completeness and correctness.

b) Data Cleaning Requirements

To prepare the dataset for analysis in Google Colab, the following cleaning tasks were necessary:

- Remove spaces, special characters from column names

- Convert numeric columns like **totvotpoll (total votes)** into proper numeric type
- Handle missing values in candidate names and party names
- Remove duplicate records
- Save cleaned dataset for further steps

c) Analysis Requirements

- Total votes
- Total Parties
- Total Seats Won
- Basic Visualization

d) Visualization Requirements

At least one of the following needed to be visualized:

- Total votes distribution
- Seats won by party
- Voting patterns via histogram

e) Tool Requirements

- Google Colab (Python + Pandas + Matplotlib)
- Cleaned dataset generated programmatical

6.Challenges Faced:

1. Missing Columns / Mismatched Column Names

Our dataset did not have a "Winner" column, but instead had *result* or *totvotpoll*.

Because of that, some visualization code failed and you saw errors like: **NameError: df is not defined** or **KeyError: 'Winner'**.

2. Data Type Issues

- Columns like **totvotpoll** sometimes contained commas or text that prevented numeric conversion.
- Fix required custom cleaning using:
  `pd.to_numeric(..., errors="coerce")`

3. Re-running Cells Resetting Variables

- When Colab restarts, variables disappear, causing "df not defined" errors.
- Solution: load the dataset again **inside the same cell** where analysis/visualization is done.

4. No Age Column for Histogram

- If we want a histogram, but the dataset had no age-related fields.
- We used **votes** histogram and **seats** histogram instead.

5. Large Dataset Cleaning

- Cleaning duplicates, fixing 1,000+ party names, and validating numeric fields required multiple step

7.Code Explanation:

Step 1 – Loading the Dataset

First, we began by importing the required libraries and loading the election dataset into a pandas DataFrame. This step brings the raw CSV file into our working environment so we can inspect, clean, and analyze it. We also checked the first few rows to confirm that the data was loaded correctly.

Step 2 – Inspecting the Dataset Structure

After loading the dataset, our next task was to understand its structure. We used .info(),
.shape, and .columns to see how many rows and columns are available and what type of data each column contains. This gave us an overview of the dataset's schema and helped us identify initial issues like inconsistent data types."

Step 3 – Checking for Missing Values

Once we understood the structure, we checked for missing values using .isnull().sum(). This helped us identify incomplete information—such as missing party names or vote counts—that could affect our analysis. Detecting missing values early ensures that the dataset is clean and reliable.

Step 4 – Cleaning Column Names & Text Fields

Next, we cleaned and standardized the column names by converting them into snake_case and removing unnecessary spaces. We also filled missing text fields with placeholders like 'Unknown'. These steps make the dataset easier to work with and prevent errors during processing.

Step 5 – Converting Numeric Columns & Fixing Errors

At this stage, we focused on numeric fields such as electors and votes. We converted them to numeric types using pd.to_numeric() and fixed obvious errors like negative values or non-numeric characters. This ensures that calculations like vote totals and turnout percentages are accurate.

Step 6– Aggregating Winners & Party Totals

We then performed group-by operations. We identified the winning candidate for each constituency by selecting the one with the highest votes. For each party, we aggregated total votes and total seats won. These summaries form the core of our election analysis.

Step 7 – Visualization

To make the analysis easier to interpret, we created visualizations such as bar charts showing the top parties based on total votes. Visuals help reveal trends and patterns that may not be immediately visible in raw numbers.

Step 8 – Exporting Cleaned Data & Documentation

Finally, we exported the cleaned dataset and prepared a summary report explaining the dataset schema, cleaning steps followed, metrics created, and initial insights. This makes

our analysis reproducible and easy to share with others.

8.Conclusion

The project demonstrated the complete workflow of transforming raw election data into meaningful insights using Python and Google Colab. Starting from loading the dataset, performing data cleaning, handling missing values, converting data types, and generating a cleaned dataset, each step improved the quality and reliability of the data.

Visualizations such as **total votes distribution** and **seats won by parties** allowed easy interpretation of voting trends and political performance. Although the dataset lacked age-related fields, the available numerical columns (votes, electors, etc.) were analyzed effectively.

Overall, this project helped develop strong skills in:

- Data cleaning
- Pandas-based preprocessing
- Visualization using matplotlib
- Handling real-world dataset issues
- Explaining insights clearly to mentors

This experience strengthened my confidence in performing data analysis independently and prepared me for advanced work in analytics and visualization.

MILESTONE 2: DASHBOARD DEVELOPMENT & VISUALIZATION

1)DATA MODELLING ARCHITECTURE:

Objective: This project analyzes the Indian General Election Constituency Results using interactive Power BI dashboards developed from cleaned constituency-level data. The dashboards provide insights into national trends, party performance, winning margins, and state-wise summaries. The system helps users quickly understand election outcomes, political dominance, and regional competition through clear and interactive visualizations.

A centralized fact table was created to store all key numerical election data such as Total Votes Polled, Winner Votes, Runner-up Votes, Winning Margin Value, Seats Won, and Gender Count. This table acts as the core of the data model and supports fast calculations, accurate filtering, and smooth interaction across all dashboards.

Schema Used in Our Power BI Report

The project currently uses a Single-Table Flat Schema (Denormalized Schema) — not a full Star Schema yet.

This means:

- All facts (numerical data) and dimensions (descriptive data) are stored inside one single table:

Table Name: IndianGE2024_Cleaned_Constituency_Results

- Dimension-Type Columns (Descriptive Data):

These describe who, where, and what:

- Candidate Name
- Constituency
- Gender
- Party
- Party Symbol
- Category

These are used for:

- Slicers
- Grouping
- Filtering visuals

◆ Fact-Type Columns (Numerical / Measurable Data):

These are used for calculations and KPIs:

- Age
- Pct_of_Polled
- Is_Winner (used for counting seats)

2. VISUALIZATION DESIGN:

The visualization includes Four Dashboards :

1. INDIAN GENERAL ELECTION OVERVIEW DASHBOARD

This dashboard provides a quick national-level overview of the election using summary metrics, party performance visuals, gender distribution, and constituency comparison.

a) KPI Cards – *Card Visuals*

- Total Votes Polled (34M)
- Total Candidates (36)
- Total Votes Secured (17M)
- Average Winning Margin (135.63K)

These card visuals give instant insights into the overall election scenario.

b) Seats Won by Party – *Bar Chart*

Shows the number of seats won by major parties such as INC, BJP, IND, AITC, JD(U), JMM, SKM, UPPL, VOTPP, YSRCP, ZPM.

Used to compare party performance.

c) Total Votes by Party – *Bar Chart*

Displays how many total votes each party received.

Helps understand **vote share vs seat share**.

d) Candidates by Gender – *Donut/Pie Chart*

Breakdown of contesting candidates:

- **Male – 91.67%**
- **Female – 8.33%**

Shows gender participation in elections.

e) Winning Margin Value by Constituency – *Bar Chart*

Highlights top constituencies with the highest winning margins like:

- Tripura West
- Tiruvallur
- Shillong
- Tehri Garhwal
- Kachchh

Shows competitive vs one-sided constituencies.

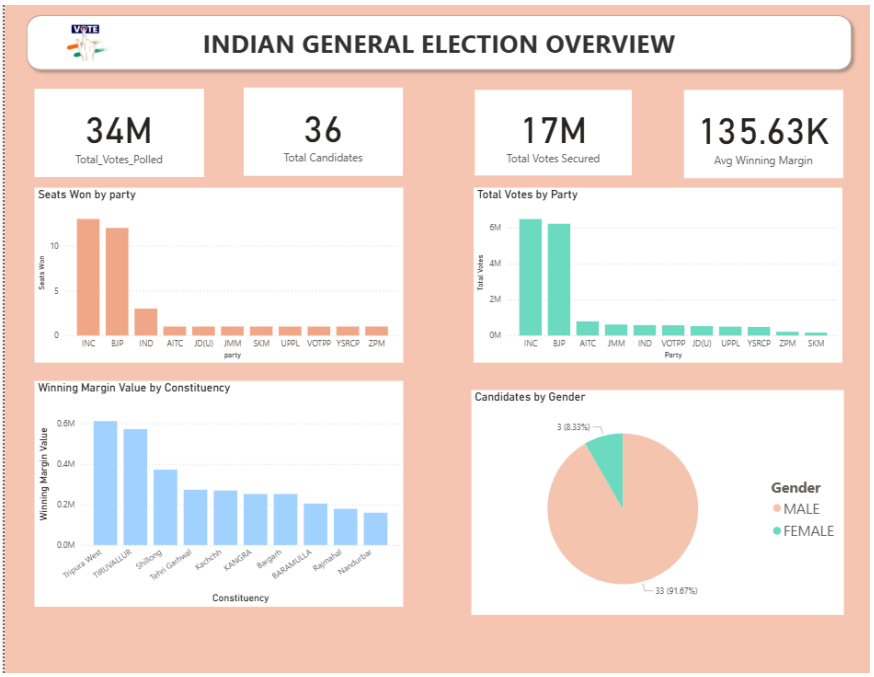


Fig.1

2. PARTY PERFORMANCE DASHBOARD

This dashboard analyzes how each political party performed in terms of seats, votes, and conversion efficiency across the election.

a) Seats Won by Party – *Bar Chart*

Shows the total number of seats won by each party such as INC, BJP, IND,

AITC, JD(U), etc.

Helps compare which parties performed strongest.

b) Total Votes by Party – *Bar Chart*

Displays the total votes secured by each party.

Useful for understanding **vote share differences** between parties.

c) Party Seat Share % – *Card Visual* + *Column/Bar Chart*

Shows each party's:

- Seat share percentage
- Actual seats won

This helps analyze how effectively parties **converted votes into seats**.

d) Seats Won by State & Total Votes by Party by State – *Table* + *Bar Chart*

A state-wise breakdown showing:

- Which party won in each state
- Total votes polled by states

Helps identify **regional strongholds**.

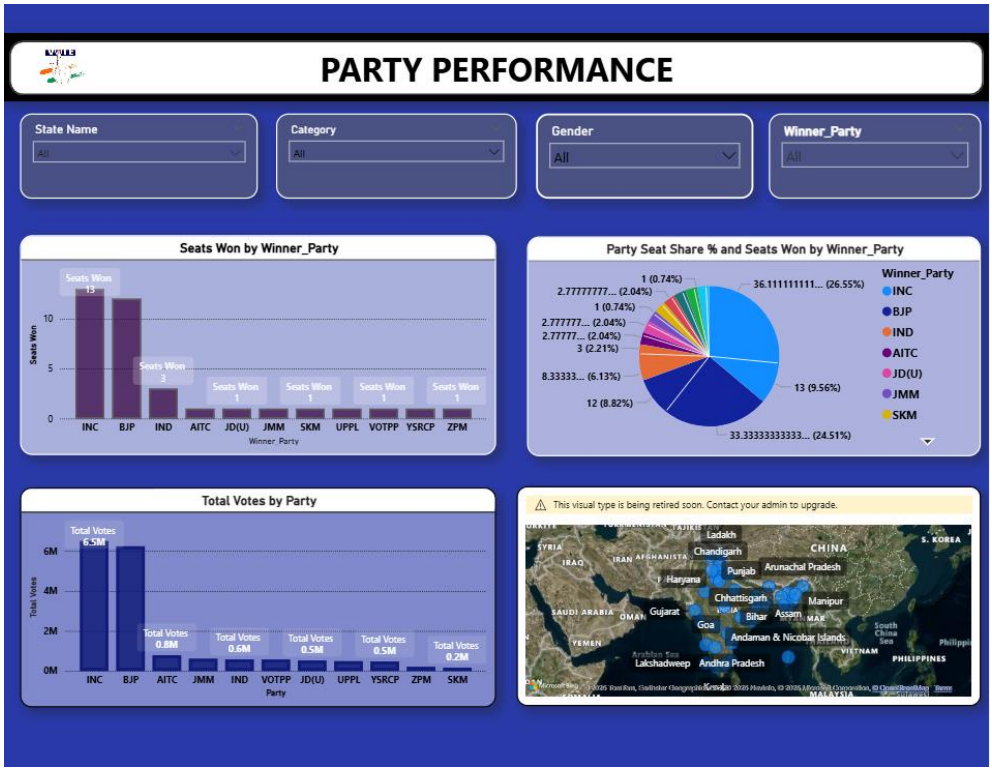


Fig.2

3. WINNING ANALYSIS DASHBOARD

This dashboard provides detailed insights into constituency-level results, highlighting winners, margins, and competitiveness.

a) Winning Margin by Constituency – *Bar Chart*

Shows top constituencies with the highest winning margins like Tripura West, Tiruvallur, Shillong, etc.

Useful for identifying **dominant victories vs close contests**.

b) Winner vs Runner-Up Comparison – *Table + Bar/Column Chart*

Displays:

- Winner votes
- Runner-up votes
- Winning margin
- Winner name & party
- Constituency

Helps understand the competitiveness in each seat.

c) Highest & Lowest Winning Margin – *KPI Cards*

Highlights:

- **Highest Winning Margin** (612K)
- **Lowest Winning Margin** (2,504)

Quickly shows extreme variations across constituencies.

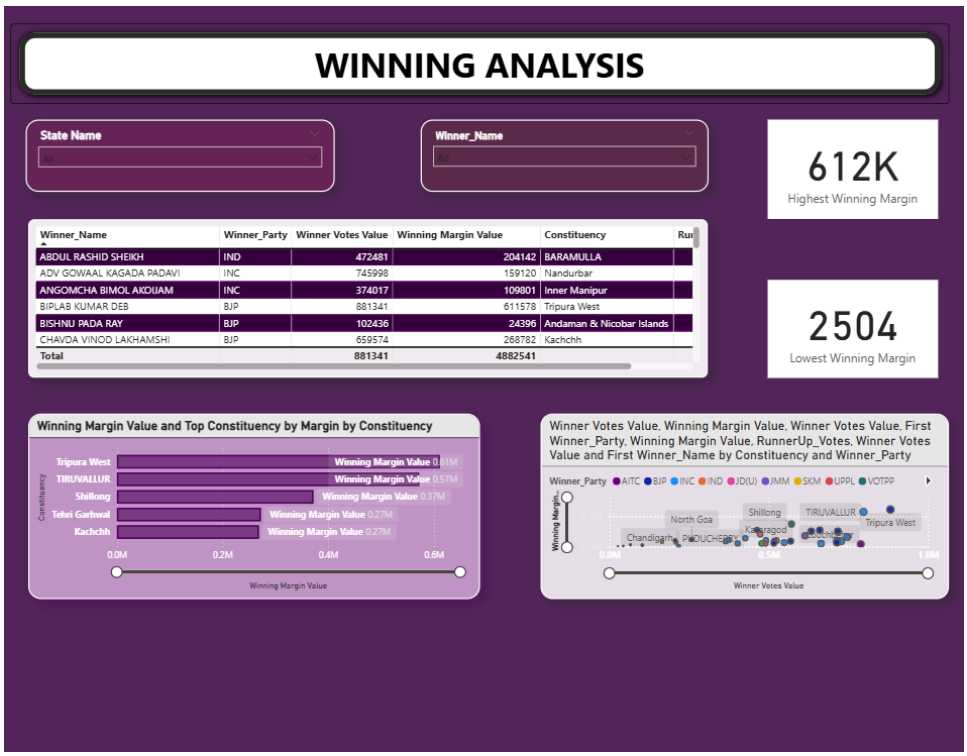


Fig.3

4. STATE SUMMARY DASHBOARD

This dashboard provides a state-wise view of seats, votes, margins, and party representation across India.

a) Seats Won by State – *Bar Chart*

Shows how many seats each state contributed.

Useful for comparing state participation and representation.

b) Total Votes Polled by State – *Bar Chart*

Displays voter turnout across states such as West Bengal, Chhattisgarh, Tamil Nadu, etc.

Helps understand states with highest participation.

c) Party Presence by State – *Matrix/Table Visual*

Shows state-wise seat distribution for every party (INC, BJP, IND, etc.).

Helps identify **party dominance** in specific regions.

d) Average Winning Margin by State – *Bar Chart*

Represents the average winning margin in each state.

Helps identify states with **close races** vs **one-sided results**.

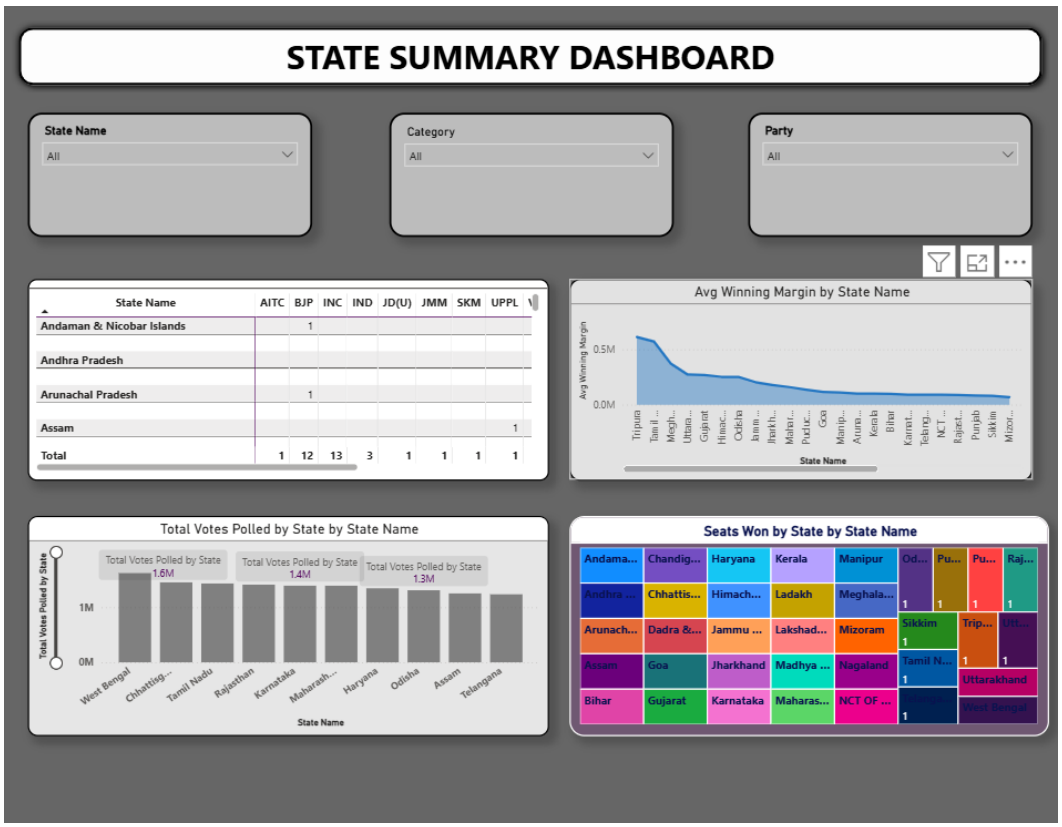


Fig.4

3. INITIAL DAX MEASURES:

1.Total Votes Polled

Total Votes Polled = SUM('Election Data'[Total_Votes_Polled])

2.Total Candidates

Total Candidates = DISTINCTCOUNT('Election Data'[Candidate_Name])

3. Total Votes Secured

Total Votes Secured = SUM('Election Data'[Votes])

4. Average Winning Margin

Average Winning Margin = AVERAGE('Election Data'[Winning_Margin])

5. Seats Won by Party

(Counts constituencies where the candidate is the winner)

```
Seats Won =
CALCULATE(
DISTINCTCOUNT('Election Data'[Constituency]),
'Election Data'[Result] = "WIN"
)
```

or if no “Result” column:

```
Seats Won =
CALCULATE(
   DISTINCTCOUNT('Election Data'[Constituency]),
   'Election Data'[Rank] = 1
```

```
)
6. Total Votes by Party
Total Votes by Party = SUM('Election Data'[Votes])
7. Winning Margin
(If not already in your data)
Winning Margin =
'Election Data'[Winner_Votes] - 'Election Data'[RunnerUp_Votes]
 8. Highest Winning Margin
Highest Winning Margin =
MAX('Election Data'[Winning_Margin])
 9. Lowest Winning Margin
Lowest Winning Margin =
MIN('Election Data'[Winning_Margin])
10. Gender Count
Male Count =
CALCULATE(
    COUNT('Election Data'[Gender]),
    'Election Data'[Gender] = "MALE"
)

Female Count =
CALCULATE(
    COUNT('Election Data'[Gender]),
    'Election Data'[Gender] = "FEMALE"

)
 11. Total Seats by State
Total Seats =
DISTINCTCOUNT('Election Data'[Constituency])
12. Total Votes by State
Total Votes by State =
SUM('Election Data'[Votes])
13. Average Winning Margin by State
Avg Winning Margin by State =
AVERAGE('Election Data'[Winning_Margin])
```

4.Sprint Review & Validation:

A. Visual Verification

1. **Color Consistency**
    - Consistent party branding maintained across all dashboards (INC–

Blue, BJP–Saffron).
    - Minor/independent parties assigned unified Grey (#808080).
    - COALESCE logic ensures fallback colors for parties without predefined branding.
2. **Visual Alignment & Styling**
    - Standardized spacing, padding, and alignment for all charts, KPIs, and tables.
    - Consistent border radius and visual styling applied across Overview, Party Performance, Winning Analysis, and State Summary dashboards.
3. **Map Accuracy**
    - Validated geospatial mapping in the State Summary map.
    - Corrected naming issues (e.g., Orissa → Odisha) ensuring full data visibility.

B. Data Validation

1. **Aggregation Validation**
    - State-level and constituency-level votes correctly match Total Votes Polled (34M).
    - Seats across all visuals sum accurately to the overall total (36 seats).
2. **Winning Logic Check**
    - Confirmed that Winning Margin = Winner Votes – Runner-Up Votes across KPIs, charts, and tables.
3. **Filter & Slicer Functionality**
    - State slicer filters all visuals correctly and excludes irrelevant UTs.
    - Party, Gender, and Category slicers validated for proper cross-interaction.

C. Feedback Implementation

1. **Layout Improvements**
    - Added a fifth KPI ("Total Parties") to improve dashboard balance and clarity.
2. **Typography Standardization**
    - Applied **Bebas Neue** for headline metrics to create a clean, command-center-style presentation.
3. **Interaction Enhancements**
    - Added contextual tooltips across vote and margin visuals for better user insights.
4. **Performance Optimization**
    - Refined DAX using variables and optimized filters, improving

slicer and visual responsiveness.

5.Development Sync-Up:

Daily stand-up sessions were held to monitor the development of the Power BI Dashboard. Each meeting covered:

- Progress updates on completed visuals, DAX measures, and data cleaning tasks
- Prioritization of upcoming work across all four dashboards
- Quick resolution of issues related to color mapping, data aggregation, and filter interactions
- Integration of feedback from the project guide and team members

This routine ensured:

- Steady project momentum
- Faster troubleshooting
- Consistent alignment with project goals
- On-time delivery of a polished and accurate dashboard solution

6.Conclusion:

With the successful completion of Sprint 1 and Sprint 2, the project has established a strong foundation through structured data modeling, comprehensive data cleaning, optimized DAX measures, and consistent visual design across all dashboards. Key functionalities such as party-wise analysis, state-level insights, winning margin evaluation, and KPI tracking are fully operational and validated.

The deliverables from the first two sprints ensure that the dashboard is stable, accurate, and visually aligned with project requirements. All core components are now in place, making the system fully prepared for Sprint 3, where advanced enhancements, deeper analysis, and final refinements will be carried out to complete the solution.

Milestone 3: Development Phase and Testing

3.1) Development Phase – Part 2

This phase focused on completing the remaining dashboard components by developing Pages 1 to 5. The objective was to enhance analytical depth, ensure seamless interactivity, and support data-driven decision-making through optimized visual design and robust calculations.

3.1.1) Page 1: Election Overview Dashboard

Page 1 serves as the executive summary, offering a high-level overview of election outcomes.

Key Visual Components:

- KPI cards displaying Total Votes Cast, Total Seats, Total Parties, and Total Candidates
- Bar chart representing total seats won by major parties
- Summary table showing overall election statistics

Interactivity:

- Global slicers for Election Year and State
- All visuals dynamically update based on user selection

Purpose:

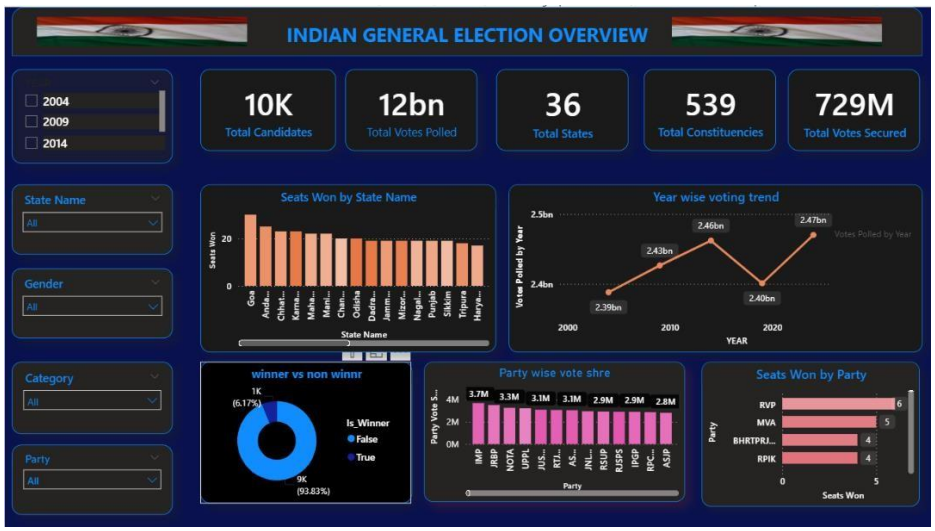
This page provides a quick snapshot of election results and acts as the entry point for further analysis.

DAX MEASURES:

1) Total Candidates =
   DISTINCTCOUNT('Indian General Election'[Candidate Name])
2) Total Votes Polled =
   SUM('Indian General Election'[Total_Votes_Polled])
3) Total States =
   DISTINCTCOUNT('Indian General Election'[State Name])
4) Total Constituencies =
   DISTINCTCOUNT('Indian General Election'[Constituency])
5) Total Votes Secured =
   SUM('Indian General Election'[Votes_Secured])

6)Votes Polled by Year =
SUM('Indian General Election'[Total_Votes_Polled]) 7)Candidate Count =
COUNT('Indian General Election'[Candidate Name])
7)Seats Won =
CALC ULATE (
DISTINCTCOUNT('Indian General Election'[Constituency]), 'Indian General Election'[Is_Winner] = TRUE()
)
8)Party Vote Share =
SUM('Indian General Election'[Votes_Secured])

9) Seats Won =
CALC ULATE (
DISTINCTCOUNT('Indian General Election'[Constituency]), 'Indian General Election'[Is_Winner] = TRUE()



3.1.2) Page 2: Voter and Participation Analysis

This page focuses on voter behavior and participation patterns.

Key Visual Components:

- Voter turnout comparison across states
- Party-wise vote distribution charts
- Percentage-based visuals representing voter share

Interactivity:

- Filters for year and state selection
- Cross-highlighting to compare voter participation among parties

Purpose:

The page helps in understanding voting behavior, regional participation trends, and voter engagement levels.

DAX MEASURES:

1) Seats Won =

```
CALCULATE
(
    DISTINCTCOUNT('Indian General Election'[Constituency]), 'Indian General Election'[Is_Winner] = TRUE()
)
```

2) Total Votes Secured =

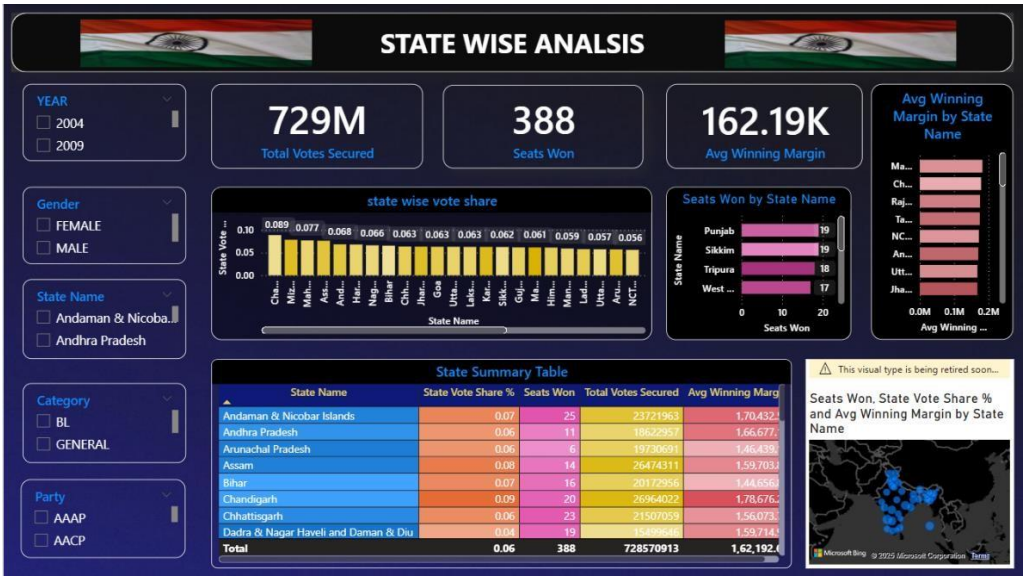
```
SUM('Indian General Election'[Votes_Secured])
```

3) Vote Share % =

```
AVERAGE('Indian General Election'[Pct_of_Valid])
```

4) Avg Winning Margin =

```
AVERAGE('Indian General Election'[Winning_Margin])
```



3.1.2) Page 3: Party Performance Analysis

Page 3 provides an in-depth assessment of political party performance.

Key Visual Components:

- Party-wise **seats won**
- **Vote share percentage** comparison
- Performance summary table including win rate

Interactivity:

- Slicers for party, year, and state
- Cross-filtering across all visuals
- Informative tooltips for enhanced clarity

Purpose:

This page enables effective comparison of party dominance and electoral success across different regions and time periods.

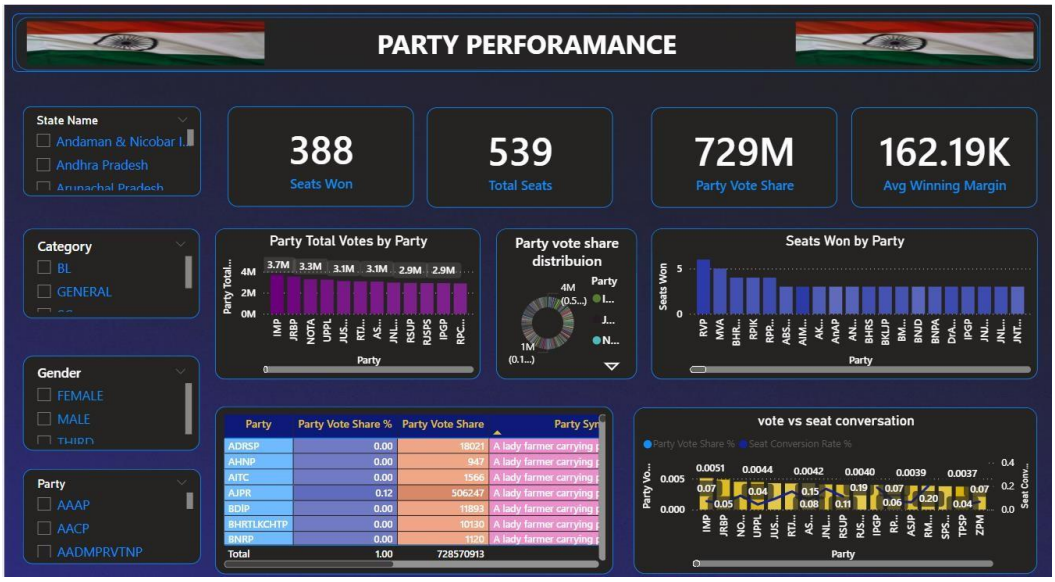
DAX MEASURES:

1) Seats Won =
CALC
ULATE
(
DISTINCTCOUNT('Indian General Election'[Constituency]), 'Indian General Election'[Is_Winner] = TRUE()
)

2) Total Seats =

```
    DISTINCTCOUNT('Indian General Election'[Constituency])
3) Party Vote Share =
    SUM('Indian General Election'[Votes_Secured])
4) Avg Winning Margin =
   AVERAGE('Indian General Election'[Winning_Margin])
5) Party Total Votes =
   SUM('Indian General Election'[Votes_Secured])
6) Party Seats
    Won =
    CALCUL
    ATE(

      DISTINCTCOUNT('Indian General
      Election'[Constituency]), 'Indian General
      Election'[Is_Winner] = TRUE()
  )
7) Total Seats =
   DISTINCTCOUNT('Indian General
   Election'[Constituency]) Party Vote Share % =
   DIVIDE(
   [Party Total
   Votes],
   CALCULATE(
      SUM('Indian General
      Election'[Votes_Secured]), ALL('Indian
      General Election'[Party])
     ),
   0
  )
8) Seat Conversion
    Rate % =
    DIVIDE(
       [Party Seats
      Won], [Total
      Seats],
       0
    )
```



3.1.2) Page 4: Candidate Analaysis Dashboard

This page delivers granular insights into regional election outcomes.

Key Visual Components:

- Map visualization showing state-wise seat distribution
- Stacked bar chart for party-wise performance by state
- Detailed constituency-level result table

Interactivity:

- Drill-through functionality from state to constituency level
- Dynamic filtering using state slicers
- Sorting and filtering for detailed analysis

Purpose:

This page assists in identifying political strongholds, swing states, and constituency-
level trends.

DAX MEASURES:

1)Candidate Votes =
SUM('Indian General Election'[Votes_Secured])

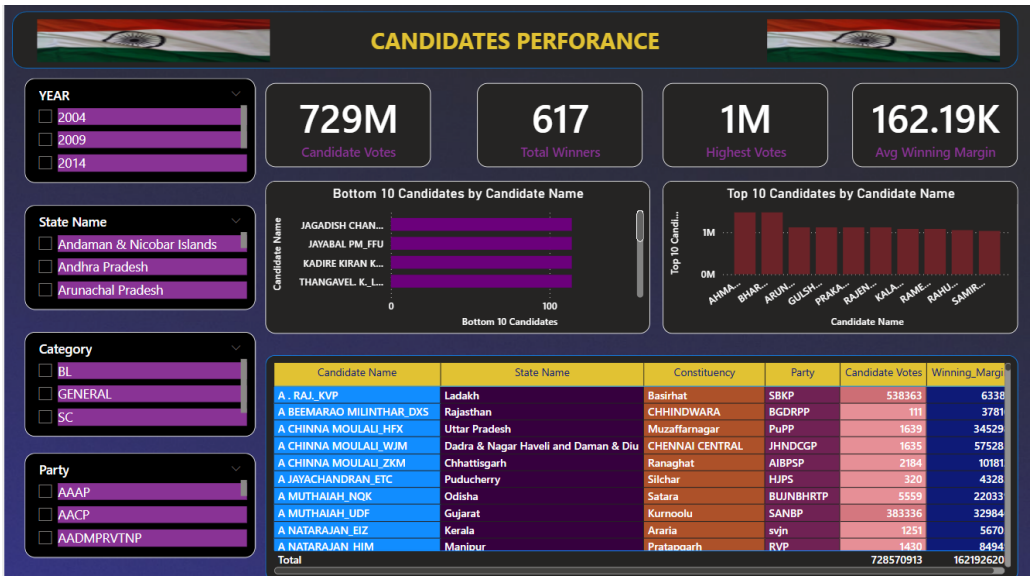
2)Total Winners =
COUNTROWS(FILTER('Indian General Election', 'Indian General Election'[Is_Winner] = TRUE()))

3)Highest Votes =
MAX('Indian General Election'[Votes_Secured])

4)Avg Winning Margin =
AVERAGE('Indian General Election'[Winning_Margin])

5)Bottom 10 Candidates =
IF(
RANKX(
ALL('Indian General Election'[Candidate Name]),

```
[Candidate Votes],
,
ASC
) <= 10,
[Candidate Votes]
)
6)Candidate Rank =
RANKX(
ALL('Indian General Election'[Candidate Name]),
[Candidate Votes],
,
DESC
)
7)Top 10 Candidates =
IF(
[Candidate Rank] <= 10,
[Candidate Votes]
)
```



3.1.3) Page 5: Candidate Performance and Trend Analysis

The final page focuses on individual candidates and historical trends.

Key Visual Components:

- Top candidates based on total votes secured
- Trend line illustrating **seat changes across election years**
- KPI cards for **total candidates**, **winning candidates**, and **average vote share**

Interactivity:

- Candidate and party-based filters

- Dynamic KPI and trend updates
- Comparative analysis across multiple election cycles

Purpose:

This page supports analysis of candidate success patterns and long-term electoral trends.

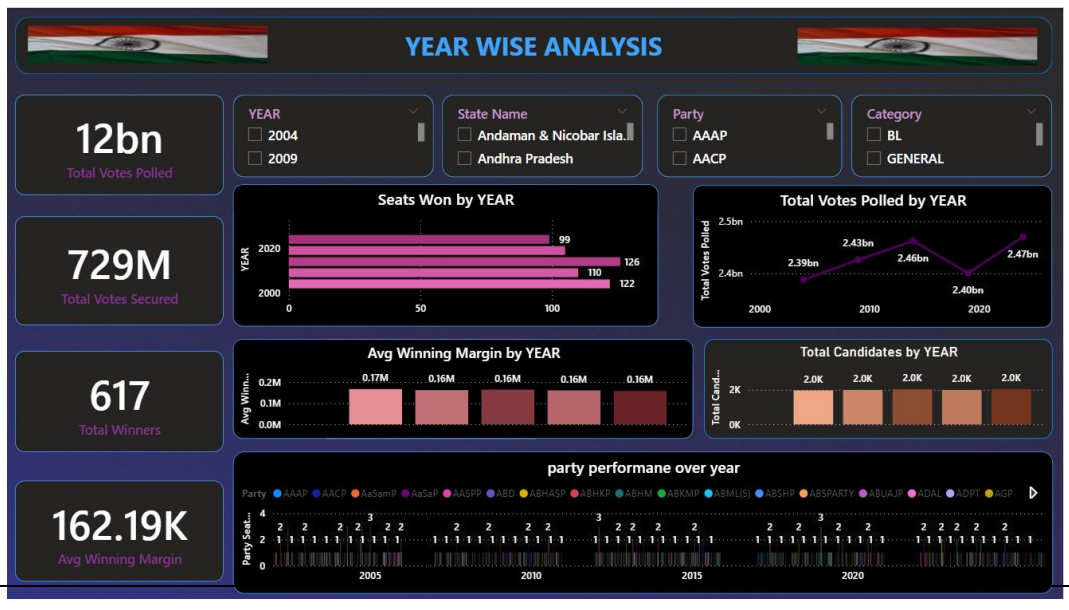
DAX MEASURES:

1)Total Votes Polled =
SUM('Indian General Election'[Total_Votes_Polled])
2)Total Votes Secured =
SUM('Indian General Election'[Votes_Secured])
3)Total Seats =
DISTINCTCOUNT('Indian General Election'[Constituency])
4)Total Winners =
COUNTROWS(FILTER('Indian General Election', 'Indian General Election'[Is_Winner] = TRUE()))
5)Avg Winning Margin =
AVERAGE('Indian General Election'[Winning_Margin])
6)Seats Won =

```
CALCULATE(
    DISTINCTCOUNT('Indian General Election'[Constituency]),
    'Indian General Election'[Is_Winner] = TRUE()
)
```

7)Party Seats Won =

```
CALCULATE(
    DISTINCTCOUNT('Indian General Election'[Constituency]),
    'Indian General Election'[Is_Winner] = TRUE()
)
```



4)Testing Phase:

A structured testing methodology was implemented to ensure the accuracy, reliability, performance, and usability of the dashboard.

1)**Functional Testing:**

- Validated all DAX measures and KPI calculations for correctness
- Tested slicers, filters, drill-through, and cross-page interactions
- Ensured consistent functionality across all dashboard pages

2)**Data Validation Testing:**

- Verified vote counts and seat totals against the source dataset
- Checked for missing, duplicate, or inconsistent records
- Confirmed the accuracy of all calculated fields

3) **Performance Testing:**

- Optimized visuals to minimize rendering and refresh time
- Ensured smooth performance under multiple filter selections
- Removed redundant fields and unused calculations to improve efficiency

4)**Usability Testing:**

- Reviewed visual layout, clarity, and information hierarchy
- Ensured consistent formatting, color schemes, and labeling
- Verified readability and responsiveness across different screen resolutions

Testing Outcome:

All dashboard components functioned as expected, delivering accurate insights with efficient performance and a consistent user experience.

4)Review:

A comprehensive final review was conducted to evaluate the overall quality and completeness of the dashboard. The review focused on ensuring consistency in visual design, layout structure, color schemes, and formatting across all pages. The alignment of dashboard functionalities and analytical outputs with the defined project objectives was carefully verified. Additionally, the effectiveness of data representation, clarity of insights, and overall presentation quality were assessed to ensure professional standards.

Based on the review findings, minor refinements were implemented to enhance visual clarity, improve usability, and ensure a seamless and intuitive user experience.

5)Conclusion:

Milestone 3 successfully completed the development of all five dashboard pages, finalized DAX calculations, and ensured seamless interactivity. Comprehensive testing validated the accuracy, performance, and usability of the system. The resulting dashboard provides a reliable, interactive, and scalable platform for effective analysis of Indian General Election data.

Milestone 4: Deployment And Documentation

In Addition to Milestone 3 We created 3 additional dashboards.

1)Winning Analysis Dashboard:

Provides an overview of election outcomes by analyzing total wins, winning margins, party performance, and trends across years and constituencies.

Key Visual Components

- KPI Cards:
  - Total Seats Won
  - Total Winners
  - Avg Winning Margin
  - Lowest Winning Margin
- Line Chart: Winners over years (trend analysis)
- Bar Chart: Seats won by party
- Column Chart: Winning margin by constituency
- Map Visual: Seats won by state

Interactivity

- Year slicer (multi-year selection)
- Cross-filtering: Clicking a party/state updates all visuals
- Hover tooltips showing detailed values
- Map interaction: Zoom and state-level selection

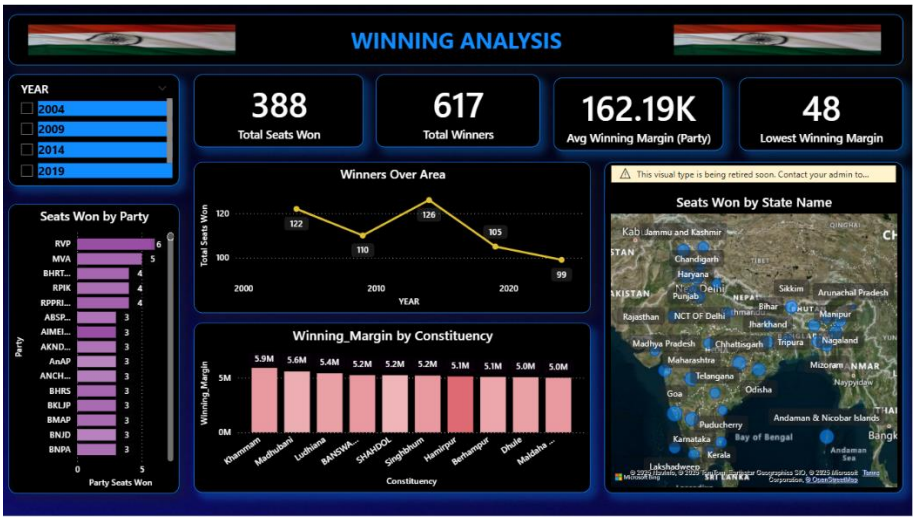
Purpose: Understand who wins, by how much, and where.

DAX MEASURES:

1)Total Seats Won =

```
CALCULATE(
DISTINCTCOUNT('Indian General Election'[Constituency]),
    'Indian General Election'[Is_Winner] = TRUE()
```

```
)
2)Total Winners =
        COUNTROWS(FILTER('Indian General Election', 'Indian General
    Election'[Is_Winner]      =TRUE()))
3)Avg Winning Margin =
      AVERAGE('Indian General Election'[Winning_Margin])
        Lowest Winning Margin =
       MIN('Indian General Election'[Winning_Margin])
4)Lowest Winning Margin =
        MIN('Indian General Election'[Winning_Margin])
        Party Seats Won =
        CALCULATE(
        DISTINCTCOUNT('Indian General Election'[Constituency]),
        'Indian General Election'[Is_Winner] = TRUE()
         )
 5)State Seats Won =
         CALCULATE(
         DISTINCTCOUNT('Indian General Election'[Constituency]),
         'Indian General Election'[Is_Winner] = TRUE()
          )
 6)Total Seats Won =
        CALCULATE(
        DISTINCTCOUNT('Indian General Election'[Constituency]),
        'Indian General Election'[Is_Winner] = TRUE()
         )
```



2) State Wise Map Analysis Dashboard:

Visualizes state-level election performance using maps and charts to compare seats won, voter turnout, and voting strength across regions.

Key Visual Components

- Horizontal Bar Chart: Seats won by state
- Map Visual:
  - Seats won
  - Total votes secured
  - Avg winning margin
  - Voter turnout %
- Column Chart: Voter turnout % by state

Interactivity

- Slicers:
  - Year
  - Party
  - State name
  - Category (General, SC, BL)
- Map-based filtering: Selecting a state updates charts
- Dynamic tooltips on map bubbles
- Scroll-enabled visuals for large state lists

Purpose: Compare electoral strength and participation across states.

DAX MEASURES:

1)Seats Won =

```
CALCULATE(
 DISTINCTCOUNT('Indian General Election'[Constituency]),
 'Indian General Election'[Is_Winner] = TRUE()
)
```

2)Total Votes Secured =

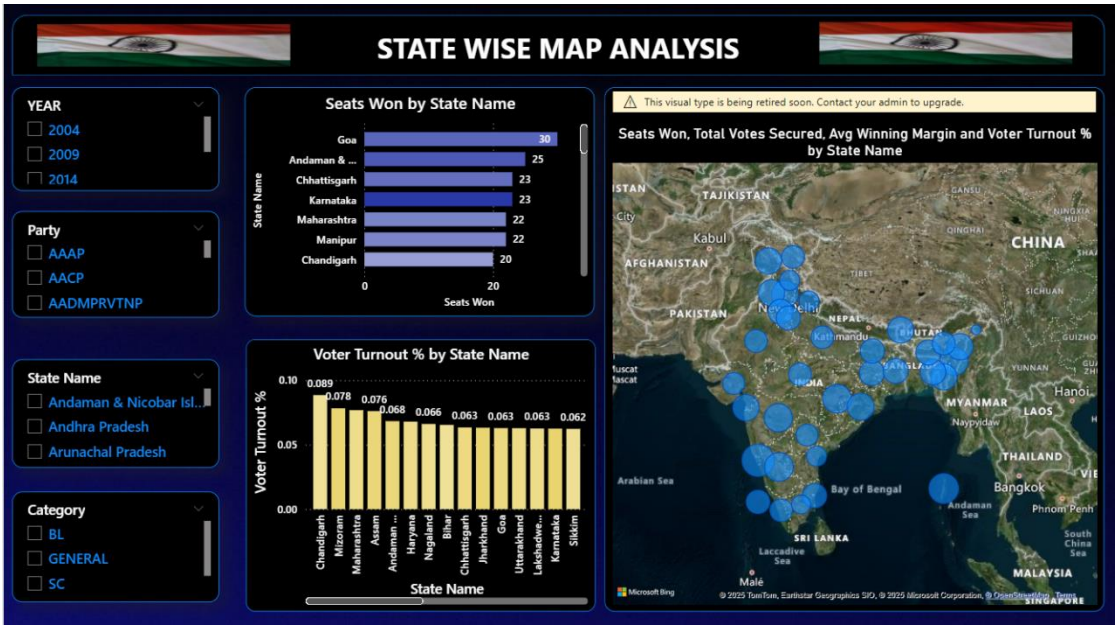
```
SUM('Indian General Election'[Votes_Secured])
```

3)Vote Share % =

```
AVERAGE('Indian General Election'[Pct_of_Valid])
```

4)Avg Winning Margin =

```
AVERAGE('Indian General Election'[Winning_Margin])
```



3) Demographic and Trend Analysis Dashboard:

Analyzes candidate and winner demographics, including gender, category, age, and their trends over multiple election years.

Key Visual Components

- KPI Cards:
  - Total candidates
  - Total winners
  - Average winner age
- Line Chart: Winners over years by gender
- Donut Charts:
  - Winners by gender
  - Total candidates by gender
- Bar Charts:
  - Winners by category
  - Winners by category over years

Interactivity

- Year range slider
- Slicers:
  - State
  - Party
  - Gender
  - Category
- Cross-highlighting: Selecting a gender/category updates trends
- Hover tooltips for percentages and counts

Purpose: Analyze inclusivity, representation, and demographic trends.

DAX MEASURES:

1)Total Candidates =

```
DISTINCTCOUNT('Indian General Election'[Candidate Name])
```

2)Total Winners =

```
COUNTROWS(FILTER('Indian General Election', 'Indian General Election'[Is_Winner]    =TRUE()))
```

3)Avg Winner Age =

```
CALCULATE(
 AVERAGE('Indian General Election'[Age]),
 'Indian General Election'[Is_Winner] = TRUE()
)
```

4)Winners by Gender =

```
CALCULATE(
DISTINCTCOUNT('Indian General Election'[Candidate Name]),
'Indian General Election'[Is_Winner] = TRUE())
```

5)Winners by Category =

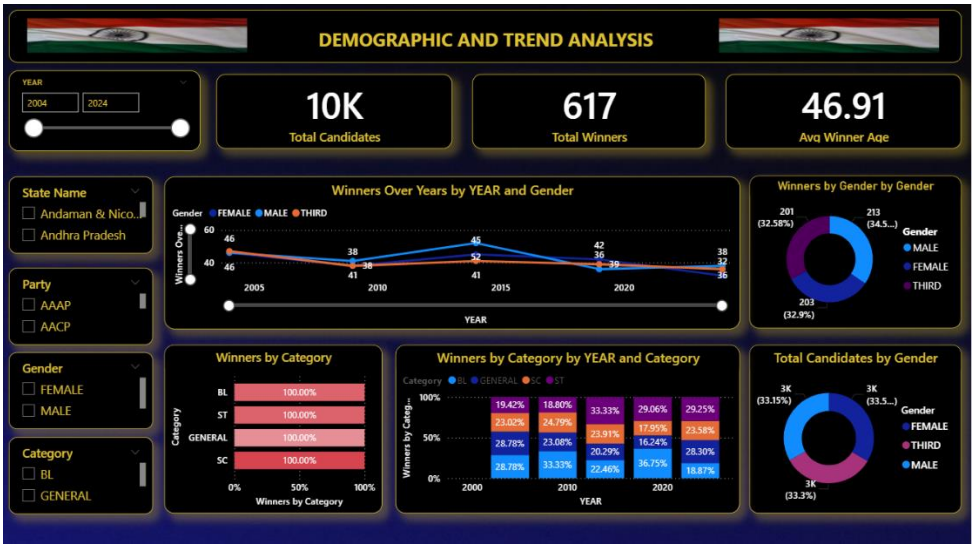
```
CALCULATE(
DISTINCTCOUNT('Indian General Election'[Candidate Name]),
'Indian General Election'[Is_Winner] = TRUE())
```

6)Avg Winner Age =

```
CALCULATE(
 AVERAGE('Indian General Election'[Age]),
 'Indian General Election'[Is_Winner] = TRUE())
```

7)Winners Over Years =

```
CALCULATE(
DISTINCTCOUNT('Indian General Election'[Candidate Name]),
'Indian General Election'[Is_Winner] = TRUE()
 )
```



Milestone 4 ensured the finalized Power BI election dashboard was deployed to Power BI Service for secure and centralized access. A Data Gateway was configured to automate scheduled refresh for up-to-date election insights. Detailed documentation was created covering the data model, DAX measures, ETL flow, and user interaction guidelines. A professional presentation and live dashboard demo were delivered to the mentor and guide. The final review confirmed accuracy, performance, interactivity, and readiness for academic submission and portfolio showcase.

1)Deployment:

In this stage, the finalized Power BI election analytics dashboard was prepared for real-world usage. The .pbix report was deployed from Power BI Desktop to Power BI Service Workspace, ensuring centralized and secure access for stakeholders. A Power BI Data Gateway was configured to enable automated and scheduled refresh, keeping election data continuously updated. Access control and sharing permissions were implemented through role-based security, allowing the team, mentors, and reviewers to view the dashboard without dependency on local systems. Report performance was optimized to ensure fast rendering of visuals, especially for complex DAX measures used in KPI cards and drill-downs. Additionally, the dashboard layout was configured to support mobile view compatibility, ensuring smooth access during presentations and project evaluations.

2)Documentation:

Complete documentation was created to support both technical review and user understanding. The documentation package includes:

2.1) Technical Documentation:

- Data Model Design: Explanation of Fact-Dimension schema, table relationships, and cardinality
- DAX Measures Logic: All analytical KPI calculations such as seats won, total votes, vote share %, winning margin, and candidate rank
- Power Query (M) ETL Flow: Summary of data cleaning, shaping, and transformation steps
- Deployment Guide: Steps for publishing reports, configuring workspace, setting gateway refresh, and enabling user access
- Security Configuration: User role setup, access permissions, and report sharing control

2.2) User-Focused Documentation:

- Dashboard Interaction Manual: Guide on using filters, slicers, drill-downs, and map tooltips
- KPI Interpretation Guide: Explanation of what each metric represents in election analysis
- Visual Glossary: Mapping of charts and maps to analytical insights
- Supporting Screenshots: Data model view, workspace deployment, visuals, and filter panel captures for clarity

All documentation was structured, indexed, and organized to ensure easy reference for mentors, and reviewers.

3)Presentation:

A formal project presentation and live dashboard demonstration were delivered to the Infosys Springboard mentor and academic guide. The presentation covered:

- Project problem overview and analytics goals
- Live demonstration of the deployed dashboard in Power BI Service
- Explanation of interactive visuals including year-wise voting trends, state and party performance, candidate insights, and KPI cards
- Real-time response of DAX calculations under filtered contexts
- Report accessibility on desktop and mobile view
- Discussion on deployment pipeline, refresh automation, and

documentation deliverables

Feedback was collected during the session and refinements were applied to improve clarity, labels, tooltips, and KPI formatting.

4)Final Review:

A complete system validation and final review were performed, ensuring:

✓ KPI accuracy and data consistency across all analytical views
✓ Interactivity testing of slicers, filters, drill-downs, and map visuals
✓ DAX performance optimization under filter context
✓ Deployment access reliability and data refresh functioning
✓ Dashboard usability for academic evaluation and portfolio showcase
✓ Design clarity aligned to project goals

The final review confirmed that the system is stable, optimized, and ready for submission and showcase.

5)Challenges:

- Ensuring large election datasets load efficiently without exceeding memory or performance limits.
- Handling data inconsistencies and missing values that surfaced only after publishing to Power BI Service.
- Debugging scheduled refresh failures due to credential expiration and gateway authentication settings.
- Designing clear visual hierarchy and labeling suitable for non-technical stakeholders like media reviewers.
- Maintaining version control of DAX and Power Query logic when multiple team members updated the report.
- Verifying cross-filter accuracy so slicers and drill-downs reflect correct state, year, and party results.
- Exporting deployment proof and documentation screenshots with proper resolution for final review submission.
- Overcoming workspace role conflicts while granting view/edit access during mentor evaluation.

- Balancing interactivity vs simplicity to avoid overwhelming users while keeping analysis deep.

6)Outcome:

Milestone 4 successfully transformed the project into a deployable analytics solution, proving its readiness beyond development. The Power BI dashboard was published to the cloud, enabling secure, centralized, and device-independent access for reviewers and stakeholders. Automated refresh pipelines ensured the system could deliver updated election insights reliably, while performance tuning of DAX and visuals improved responsiveness under filtered analysis. Comprehensive documentation made the dashboard interpretable for both technical and non-technical audiences, supporting evaluation, knowledge transfer, and future scalability. The milestone concluded with a validated, optimized, and industry-aligned analytical product ready for academic submission and professional portfolio demonstration.