ElectViz Election Data Visualization for Media

Infosys Springboard 6.0
Milestone -1

Project Overview:

The **ElectViz** project focuses on analysing and visualizing data from the Indian General Election **It** is a data visualization project designed to convert raw election data into meaningful and interactive visual dashboards. By using structured datasets from the Indian Genearl elections The dataset contains information like state name, constituency, candidate, party, votes secured, winner details, and winning margin.
The main aim of ElectViz is to take this large election data and turn it into **easy dashboards, charts, and maps** that can help media channels quickly understand and explain the results.

Objective:

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

The main goals of this project are:

1. To consolidate and analyze key election metrics, including total votes, party

performance, seats won by party, and winning margin.

2. To identify historical voting trends by visualizing data across different election

years.

3. To provide a comparative analysis of party performance at state 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

Project description in detail:

The ElectViz project is develope to explore how data visualization can enhance political and electoral reporting. Our team used the Eldataset from Kaggle to create a Power BI dashboard that highlights national and regional election results, party performance comparisons, alliance analytics, and voter behavior patterns.

Requirement Analysis:

The ElectViz project aims to take the **Indian General Election dataset**, clean it, analyze it, and create simple visualizations suitable for media reporting. Before starting the analysis, the following requirements were identified:

1. 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.

2. 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

c) Analysis Requirements

further steps:

● Total Votes

● Total Parties

● Total Seats Won

● Basic visualization (bar/histogram)

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 program

Data set Exploration and Understanding

The dataset includes the following key:

Column Name Description

State Name Name of the Indian State

Constituency Constituency where election occurred

Candidate Name Name of contesting candidate

Gender Male/Female

Age Age of the candidate

Category General, SC, ST

Party Political Party Name

Party Symbol Symbol of the political party

party Symbol Symbol of the political party

Total_Votes_Polled Total votes polled in the constituency

Votes_Secured Votes secured by that candidate

Pct_of_Polled Percentage of polled votes

Pct_of_Valid Percentage of valid votes

Votes_Secured_fill Cleaned/processed vote count

Winner_Name Name of winning candidate

Winner_Party Winning candidate’s political party

Winner_Votes Total votes secured by the winner

Product backlog creation

Download dataset :

1.Uplode the election dataset for analysis the data

2. clean and transform the dataset so the dashboard works accurately.

3 view a state-wise summary of total votes.

4. want to view party-wise seat count.

5. check gender/category representation.

6. view winning margin top 10.

7. filters by party, state, and category.

8. interactive visuals and tooltips.

Sprint Planning:

1.Dataset collection & understanding of columns

2. Cleaning data: remove nulls, formatting columns

3. Create calculated fields (Winning Margin, Winner etc.)

4. Create Power BI model & relationships

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 asTotal Votes, Total Parties, Total Seats, and Voter Turnout %.

4. Microsoft Excel: Likely used as the initial data source for collecting and formattingthe raw election data before importing it into Power BI.

5. Canva & PowerPoint – for presentation design and visual enhancement.

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:

2. Data Type Issues

● Columns like **totvotpoll** sometimes contained commas or text that prevented numeric conversion

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

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.

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 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

Milestone -2 :Development Phase

Tool And Technology :

- Microsoft power bi
- Indian general election 2024
- Power query
- Dax
- Pie chat bar chat column chat map treemap slicer table

Objectives:

- analyze the seats won and total votes secured by political parties
- identify winning margins at constituency and state levels
- evaluate demographic characteristics of candidates (age, gender, category)
- provide a comparative study among states, parties, and candidates
- create user-friendly dashboards for quick decision-making and reporting

Data Modelling :

After cleaning the data set we move for modelling part.we create a fact table for storing the numerical data such as

Total vote

Total seat

Total vote by party

Seat own

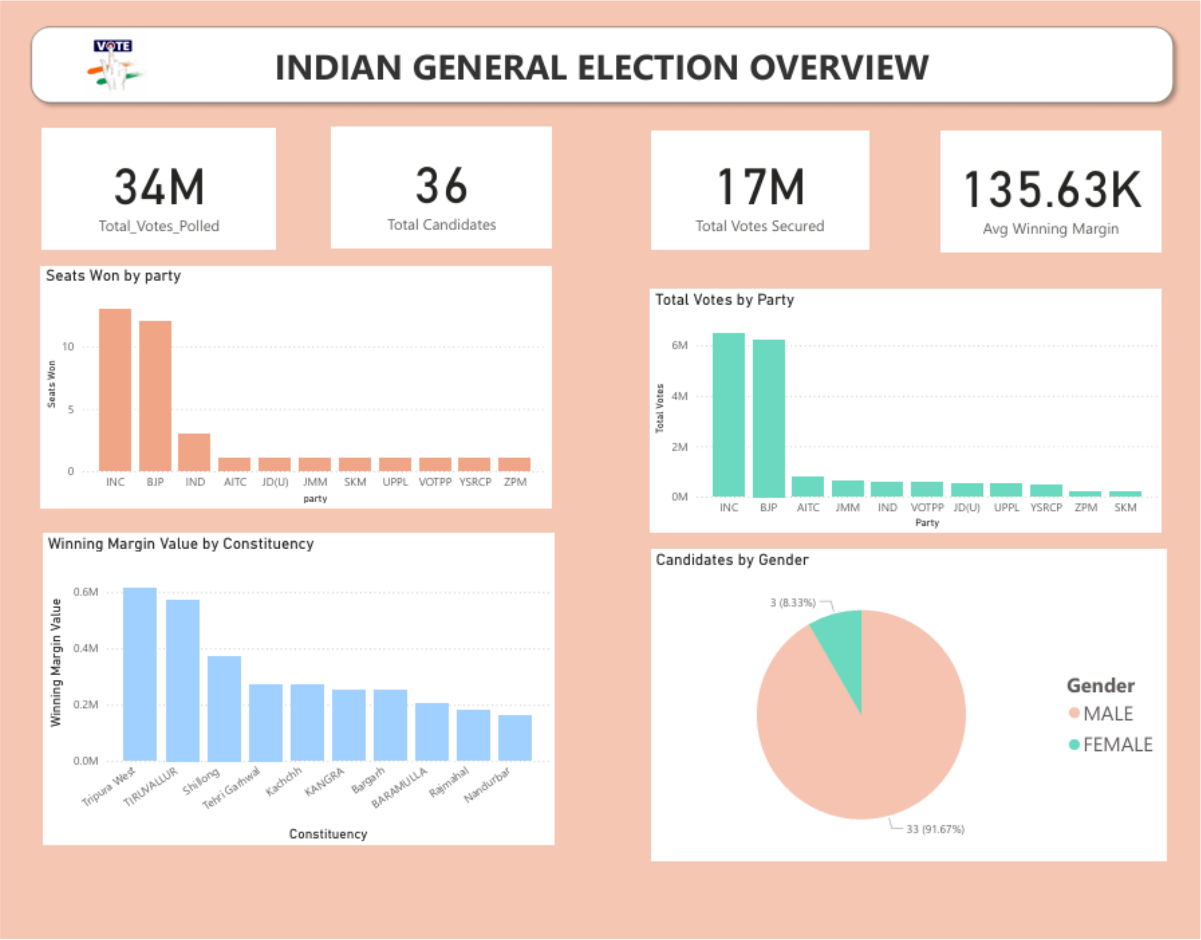
Winning margin

Visualization design :

There are four major dashboard pages each have a specific analytical purpose

Page 1 – Election over view

This dashboard provides a high level national summary of elections.It provides a high level national summary of elections.This page users to understand overall performance of election. Like seat own by party,constituency wise winning margin, gender participation and total vote gained by party



INDIAN GENERAL ELECTION OVERVIEW

34M Total_Votes_Polled

36 Total Candidates

17M Total Votes Secured

135.63K Avg Winning Margin

Key Visuals:

- Bar Chart – Seats Won by Party
- Column Chart – Total Votes by Party
- Donut Chart – candidate by gender

KPI card show:

- Total vote polled - 34M
- Total candidates - 60
- Total vote secured17 M
- Avg winning margin 135.63K

DAX measurement:

Total Votes Polled =

SUM('IndianGE2024_Cleaned_Constituency_Results'[Total_Votes_Polled])

Total Candidates =

DISTINCTCOUNT('IndianGE2024_Cleaned_Constituency_Results'[Candidate Name])

Total Votes Secured =

SUM('IndianGE2024_Cleaned_Constituency_Results'[Votes_Secured])

Winning Margin =

MAX('IndianGE2024_Cleaned_Constituency_Results'[Winning_Margin])

ALL('IndianGE2024_Cleaned_Constituency_Results')

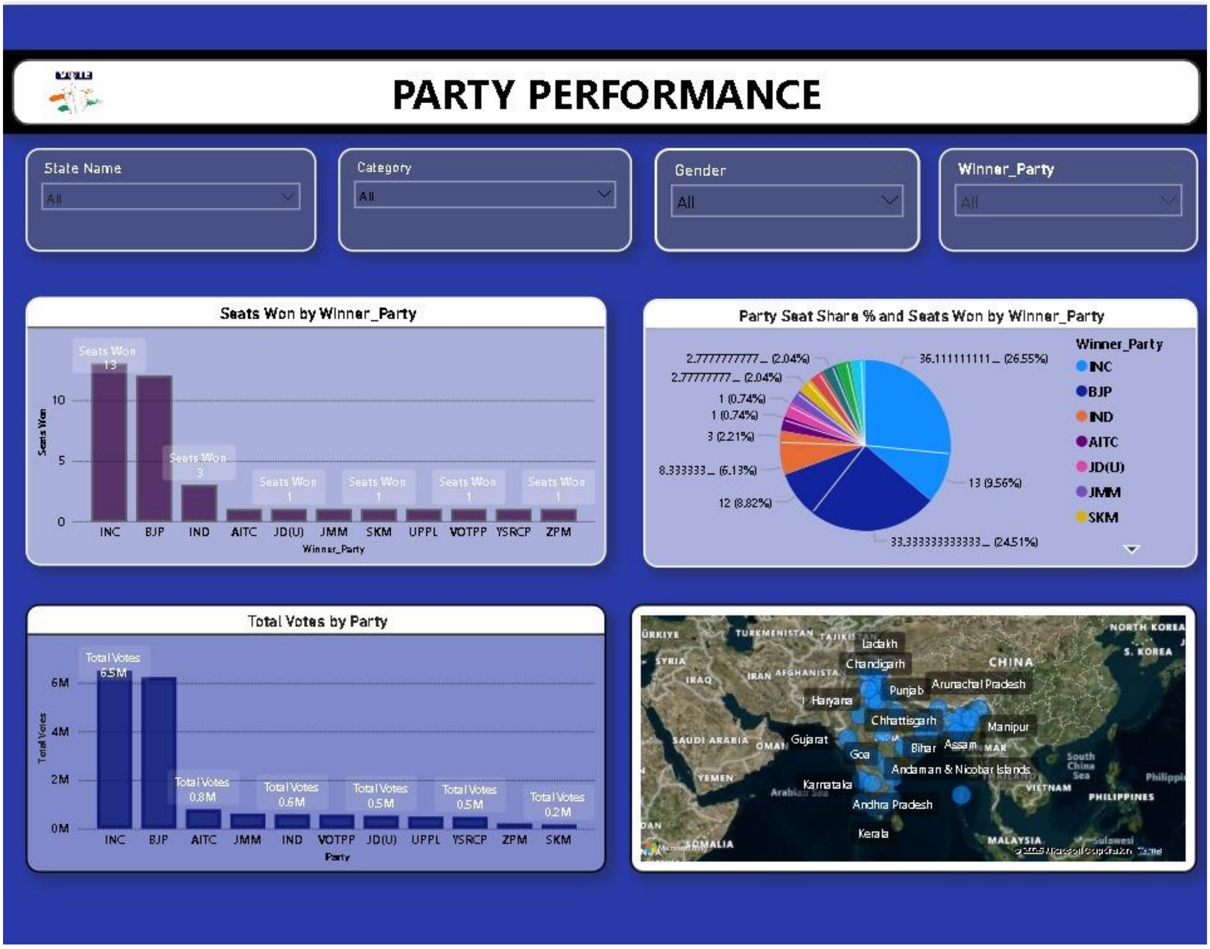
)

Page 2 – Party Performance Dashboard

This page focuses on measuring political party performance and vote distribution across India.

Key Visuals:

- Bar Chart – Seats Won by Party
- Column Chart – Total Votes by Party
- Donut Chart – Party Seat Share %
- Map – Seats won by party across states



State Name: All

Category: All

Gender: All

Winner_Party: All

DAX Measurement:

Total Seats =

CALCULATE(

COUNT('IndianGE2024_Cleaned_Constituency_Results'[Winner_Name]),

ALL('IndianGE2024_Cleaned_Constituency_Results')

)

Seats Won by Party =

CALCULATE(COUNTROWS('IndianGE2024_Cleaned_Constituency_Results'),

'IndianGE2024_Cleaned_Constituency_Results'[Is_Winner] = TRUE)

Seats Won by Party =

CALCULATE(COUNTROWS('IndianGE2024_Cleaned_Constituency_Results'), 'IndianGE2024_Cleaned_Constituency_Results'[Is_Winner] = TRUE)

Total Votes Secured =

SUM('IndianGE2024_Cleaned_Constituency_Results'[Votes_Secured])

Seats Won by State =

CALCULATE(

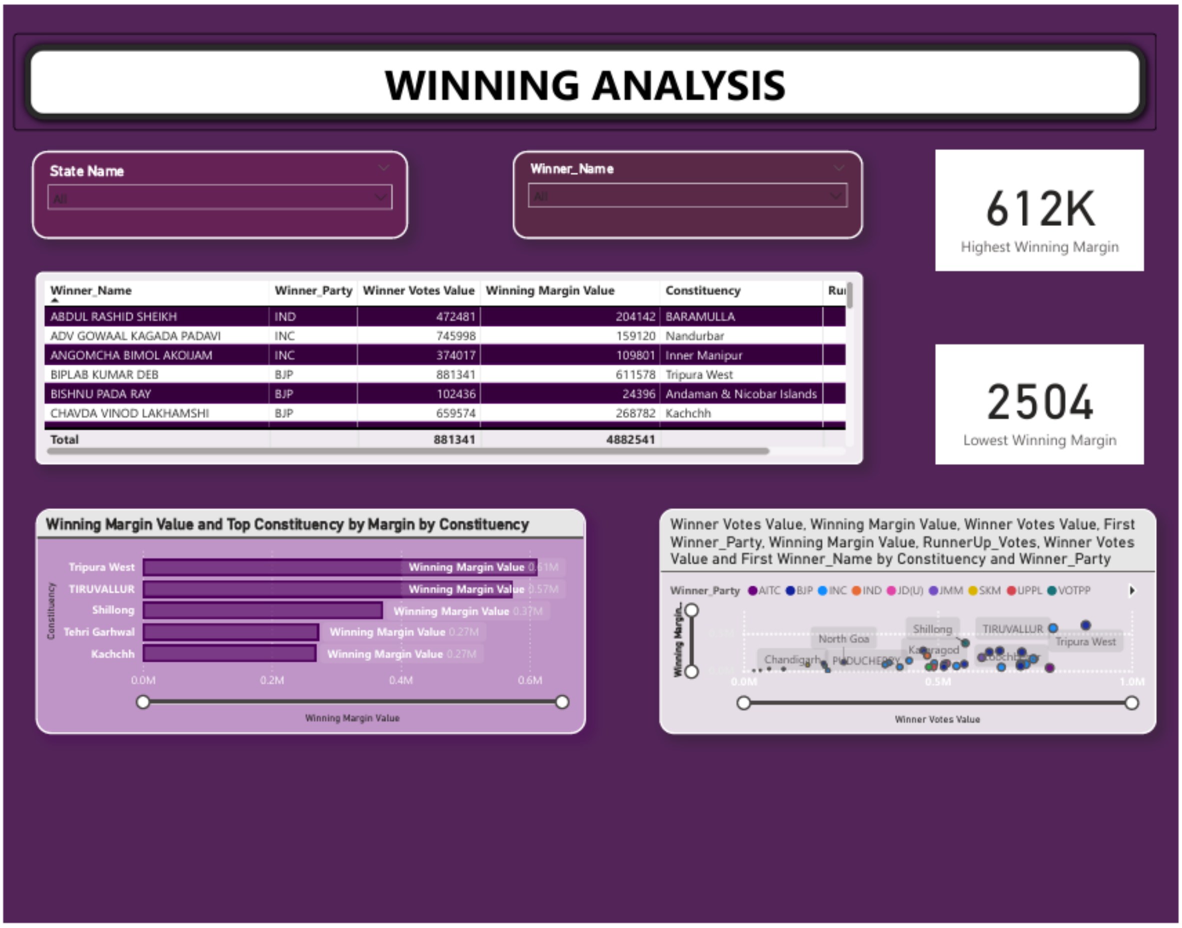
COUNTROWS('IndianGE2024_Cleaned_Constituency_Results'),

'IndianGE2024_Cleaned_Constituency_Results'[Is_Winner] = TRUE()

)ERAGE('IndianGE2024_Cleaned_Constituency_Results'[Winning_Margin])

Page-3 Winning Analysis Dashboard:

This page analyze top winning constituencies



WINNING ANALYSIS

State Name: All

Winner_Name: All

612K
Highest Winning Margin

2504
Lowest Winning Margin

Winner_Name	Winner_Party	Winner Votes Value	Winning Margin Value	Constituency	Ru
ABDUL RASHID SHEIKH	IND	472481	204142	BARAMULLA	
ADV GOWAAL KAGADA PADAVI	INC	745998	159120	Nandurbar	
ANGOMCHA BIMOL AKOIJAM	INC	374017	109801	Inner Manipur	
BIPLAB KUMAR DEB	BJP	881341	611578	Tripura West	
BISHNU PADA RAY	BJP	102436	24396	Andaman & Nicobar Islands	
CHAVDA VINOD LAKHAMSHI	BJP	659574	268782	Kachchh	
Total		881341	4882541		

Winning Margin Value and Top Constituency by Margin by Constituency

Tripura West
TIRUVALLUR
Shillong
Tehri Garhwal
Kachchh
Winning Margin Value 0.61M
Winning Margin Value 0.57M
Winning Margin Value 0.37M
Winning Margin Value 0.27M
Winning Margin Value 0.27M
Constituency
0.0M
0.2M
0.4M
0.6M
Winning Margin Value

Winner Votes Value, Winning Margin Value, Winner Votes Value, First Winner_Party, Winning Margin Value, RunnerUp_Votes, Winner Votes Value and First Winner_Name by Constituency and Winner_Party

Key Visuals:

- Bar chart for Winning Margin Value by Constituency
- Comparision chart for Winner Votes vs Runner-up Votes
- Table for Winner Name & Party-wise Winning Margin

KPI Cards:

- Highest Winning Margin – 612K
- Lowest Winning Margin – 2540

DAX Measurement:

Winning Margin Value =

SUM('IndianGE2024_Cleaned_Constituency_Results'[Winning_Margin])

Winner Votes Value =

MAX(IndianGE2024_Cleaned_Constituency_Results[Winner_Votes])

Winner Votes Value =

MAX(IndianGE2024_Cleaned_Constituency_Results[Winner_Votes])

Lowest Winning Margin =

MIN(IndianGE2024_Cleaned_Constituency_Results[Winning_Margin])

Lowest Winning Margin =

MIN(IndianGE2024_Cleaned_Constituency_Results[Winning_Margin])

Winner Display =

CONCATENATEX(

VALUES(IndianGE2024_Cleaned_Constituency_Results[Winner_Name]),

IndianGE2024_Cleaned_Constituency_Results[Winner_Name],

", "

)

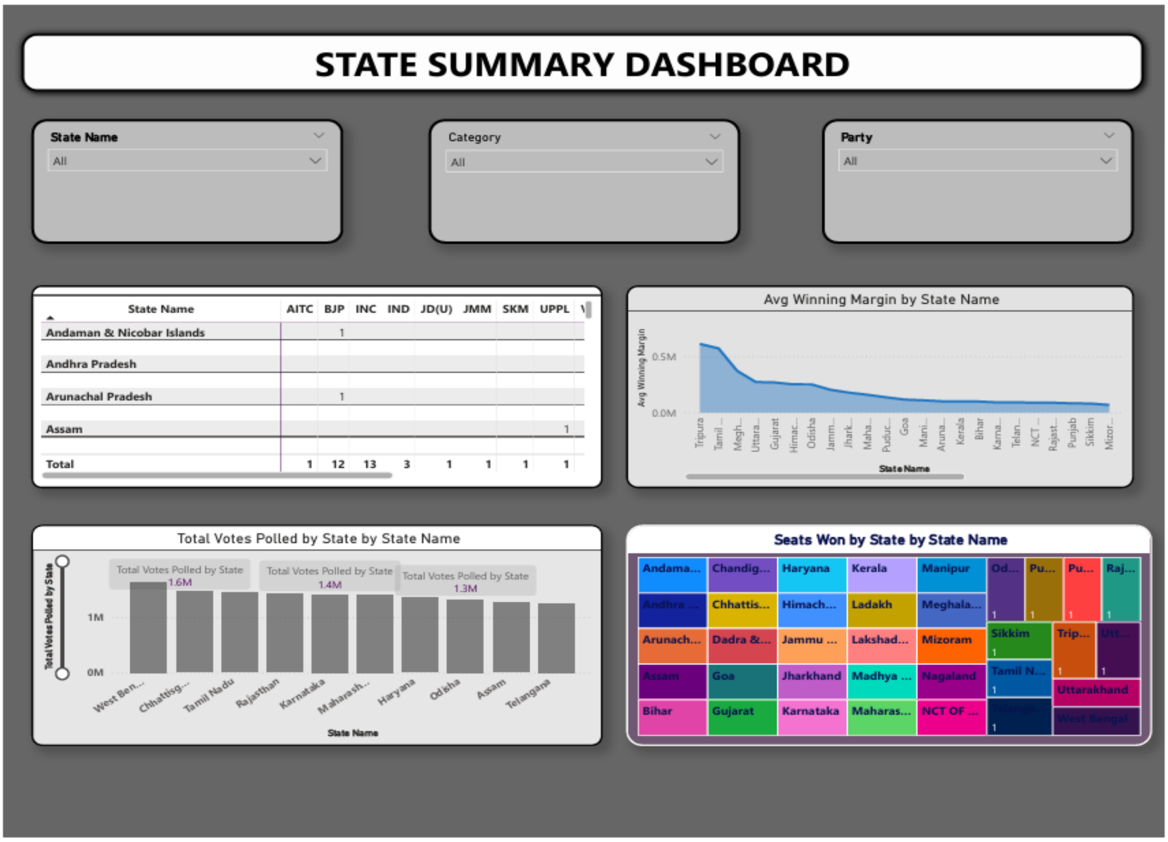
Page 4 – State-Wise summary

It Focuses on analyzing performance across states.

Key Visuals:

- Clustered Bar – Seats won by state

- ❖ Column – Total votes polled by state
- ❖ Heat Map – Average winning margin by state
- ❖ Table – Candidate count per state



STATE SUMMARY DASHBOARD

State Name: All

Category: All

Party: All

State Name	AITC	BJP	INC	IND	JD(U)	JMM	SKM	UPPL
Andaman & Nicobar Islands		1						
Andhra Pradesh								
Arunachal Pradesh		1						
Assam								1
Total	1	12	13	3	1	1	1	1

DAX Measurement:

Seats Won by State =

```
CALCULATE(
COUNTROWS('IndianGE2024_Cleaned_Constituency_Results'),
'IndianGE2024_Cleaned_Constituency_Results'[Is_Winner] = TRUE()
)
```

Total Votes Polled by State =

```
SUM( IndianGE2024_Cleaned_Constituency_Results[Total_Votes_Polled] )
```

Avg Winning Margin =

```
AVERAGE('IndianGE2024_Cleaned_Constituency_Results'[Winning_Margin])
```

Seats by State and Party =

```
CALCULATE(

DISTINCTCOUNT( IndianGE2024_Cleaned_Constituency_Results[Constituency] ),

IndianGE2024_Cleaned_Constituency_Results[Is_Winner] = TRUE()

)
```

Future Scope:

This dashboard can be extended by adding:

- Year-wise election comparison
- Voter turnout patterns
- Urban vs rural constituency analysis
- Integration with live election API sources

Milestone -3: Development And Testing Phase

Description :

Milestone 3 was dedicated to the comprehensive development and implementation of all required DAX measures, their systematic integration across respective dashboard pages, and rigorous validation of analytical accuracy. Each dashboard page was enhanced with tailored, page-specific DAX measures to ensure the delivery of reliable KPIs, in-depth comparative analysis, and data-driven insights pertaining to Indian General Election data

DAX Complection:

Page 1 – Election Overview Dashboard:

This dashboard provides a high level national summary of elections.This page users to understand overall performance of election. Like seat own by party,constituency wise winning margin, gender participation and total vote gained by party

DAX Measurement:

Total Candidates =

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

Total Votes Polled =

SUM('Indian General Election'[Total_Votes_Polled])

Total States =

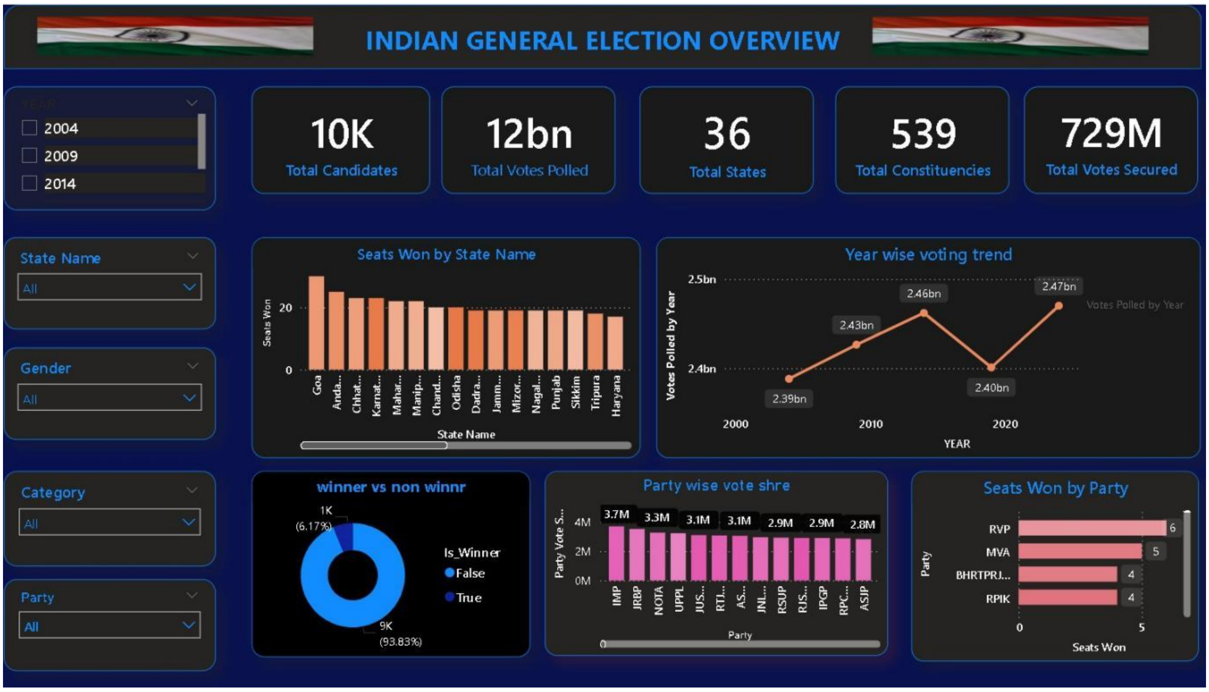
DISTINCTCOUNT('Indian General Election'[State Name])

Total Constituencies =

DISTINCTCOUNT('Indian General Election'[Constituency])

Total Votes Secured =

SUM('Indian General Election'[Votes_Secured])



Votes Polled by Year =

SUM('Indian General Election'[Total_Votes_Polled])

Candidate Count =

COUNT('Indian General Election'[Candidate Name])

Seats Won =

CALCULATE(

DISTINCTCOUNT('Indian General Election'[Constituency]),

'Indian General Election'[Is_Winner] = TRUE()

)

Party Vote Share =

SUM('Indian General Election'[Votes_Secured])

Seats Won =

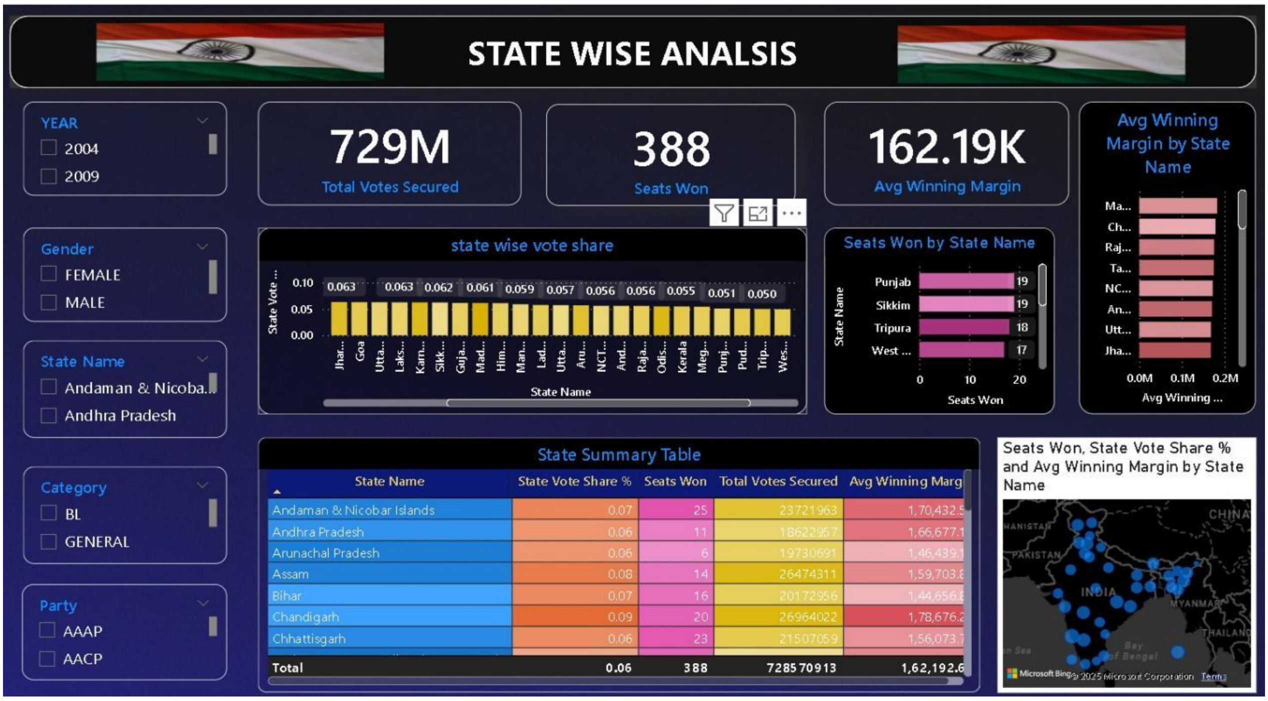
CALCULATE(

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

)

Page 2 – State-Wise Analysis Dashboard:

Provides a comprehensive analysis of election performance across different states to identify voting patterns, seat distribution, and winning margins.



STATE WISE ANALISIS

YEAR
□ 2004
□ 2009

Gender
□ FEMALE
□ MALE

State Name
□ Andaman & Nicoba...
□ Andhra Pradesh

Category
□ BL
□ GENERAL

Party
□ AAAP
□ AACP

729M
Total Votes Secured

388
Seats Won

162.19K
Avg Winning Margin

State Summary Table

State Name	State Vote Share %	Seats Won	Total Votes Secured	Avg Winning Marg
Andaman & Nicobar Islands	0.07	25	23721963	1,70,432.5
Andhra Pradesh	0.06	11	18622957	1,66,677.1
Arunachal Pradesh	0.06	6	19730691	1,46,439.1
Assam	0.08	14	26474311	1,59,703.8
Bihar	0.07	16	20172956	1,44,656.8
Chandigarh	0.09	20	26964022	1,78,676.2
Chhattisgarh	0.06	23	21507059	1,56,073.7
Total	0.06	388	728570913	1,62,192.6

DAX Measurement:

Seats Won =

CALCULATE(

DISTINCTCOUNT('Indian General Election'[Constituency]),

'Indian General Election'[Is_Winner] = TRUE()

)

Total Votes Secured =

SUM('Indian General Election'[Votes_Secured])

Vote Share % =

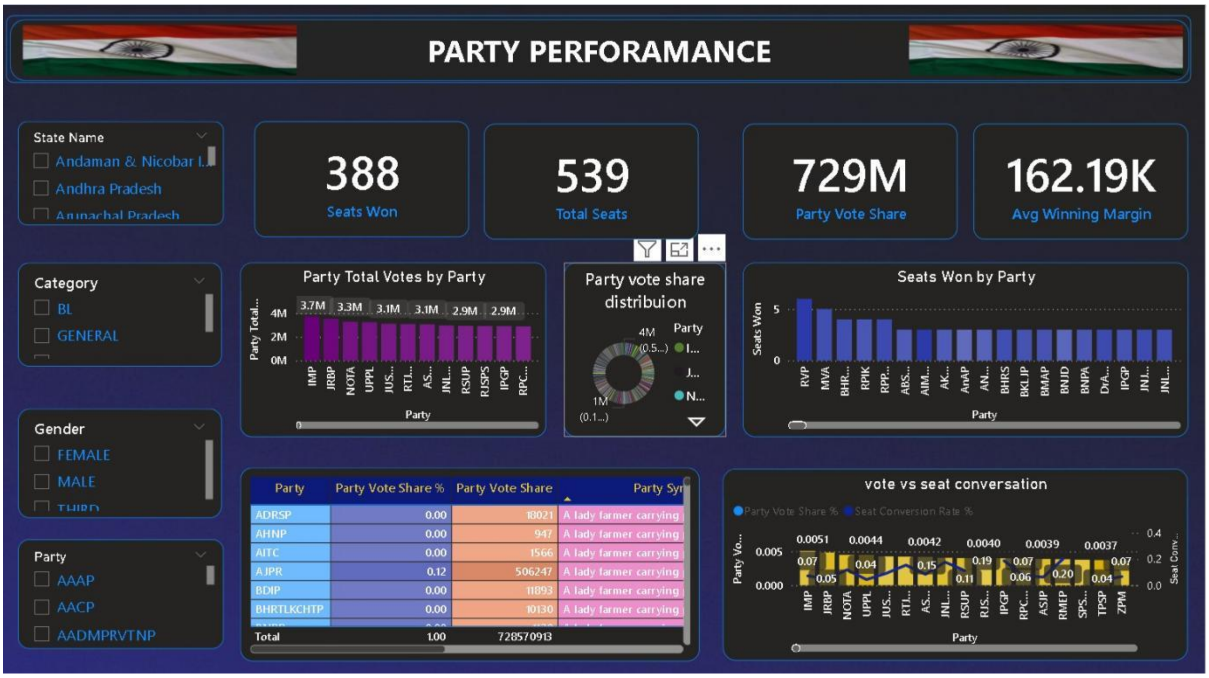
AVERAGE('Indian General Election'[Pct_of_Valid])

Avg Winning Margin =

AVERAGE('Indian General Election'[Winning_Margin])

Page 3 – Party Performance Dashboard :

To analyze party-wise vote share, seats won, and winning margins.Analyzes the performance of political parties by comparing votes, vote share, seats won, and winning margins to evaluate party effectiveness.



PARTY PERFORAMANCE

State Name
□ Andaman & Nicobar I...
□ Andhra Pradesh
□ Arunachal Pradesh

Category
□ BL
□ GENERAL

Gender
□ FEMALE
□ MALE
□ THIRD

Party
□ AAAP
□ AACP
□ AADMPRVTNP

388 Seats Won

539 Total Seats

729M Party Vote Share

162.19K Avg Winning Margin

Party	Party Vote Share %	Party Vote Share	Party Sy
ADRSP	0.00	18021	A lady farmer carrying
AHNP	0.00	947	A lady farmer carrying
AITC	0.00	1566	A lady farmer carrying
AJPR	0.12	506247	A lady farmer carrying
BDIP	0.00	11893	A lady farmer carrying
BHRTLKCHTP	0.00	10130	A lady farmer carrying
Total	1.00	728570913	

DAX Measurement:

Seats Won =

CALCULATE(

DISTINCTCOUNT('Indian General Election'[Constituency]),

'Indian General Election'[Is_Winner] = TRUE()

)

Total Seats =

DISTINCTCOUNT('Indian General Election'[Constituency])

Party Vote Share =

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

Avg Winning Margin =

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

Party Total Votes =

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

Party Seats Won =

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

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
)
```

Seat Conversion Rate % =

```
DIVIDE(
    [Party Seats Won],
```

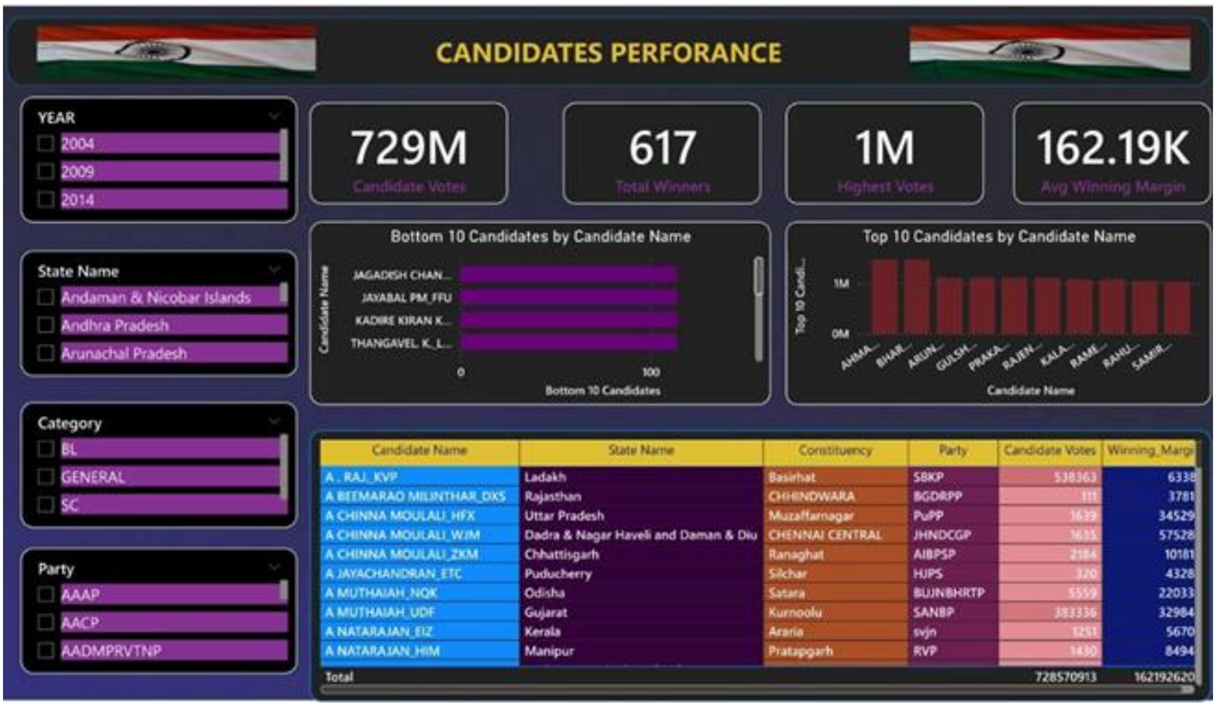
    [Total Seats],

    0

)

Page 4 – Candidate Analysis Dashboard

To analyze and compare the performance of individual candidates across elections.And understand voting patterns and competitiveness at the candidate level.



Candidate Name	State Name	Constituency	Party	Candidate Votes	Winning_Margi
A . RAJ_KVP	Ladakh	Basirhat	SBKP	538363	6338
A BEEMARAO MILINTHAR_DXS	Rajasthan	CHHINDWARA	BGDRPP	111	3781
A CHINNA MOULALI_HFX	Uttar Pradesh	Muzaffarnagar	PuPP	1639	34529
A CHINNA MOULALI_WJM	Dadra & Nagar Haveli and Daman & Diu	CHENNAI CENTRAL	JHNDCGP	1635	57528
A CHINNA MOULALI_ZKM	Chhattisgarh	Ranaghat	AIBPSP	2184	10181
A JAYACHANDRAN_ETC	Puducherry	Silchar	HJPS	320	4328
A MUTHAIAH_NQK	Odisha	Satara	BUJNBHRTP	5559	22033
A MUTHAIAH_UDF	Gujarat	Kurnoolu	SANBP	383336	32984
A NATARAJAN_EIZ	Kerala	Araria	svjn	1251	5670
A NATARAJAN_HIM	Manipur	Pratapgarh	RVP	1430	8494
Total				728570913	162192620

Candidate Votes =

SUM('Indian General Election'[Votes_Secured])

Total Winners =

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

Highest Votes =

MAX('Indian General Election'[Votes_Secured])

Avg Winning Margin =

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

Bottom 10 Candidates =

```
IF(
    RANKX(
        ALL('Indian General Election'[Candidate Name]),
        [Candidate Votes],
        ,
        ASC
    ) <= 10,
    [Candidate Votes]
)
```

Candidate Rank =

```
RANKX(
    ALL('Indian General Election'[Candidate Name]),
    [Candidate Votes],
    ,
    DESC
)
```

Page 5 – Year wise Trend Dashboard :

It provide the year wise voting Trend and analyze the performance across different election years. And understand changes in voter participation, seat distribution, and candidate performance over time.

DAX Measurement:

Total Votes Polled =

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

Total Votes Secured =

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

Total Seats =

```
DISTINCTCOUNT('Indian General Election'[Constituency])
```

Total Winners =

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

Avg Winning Margin =

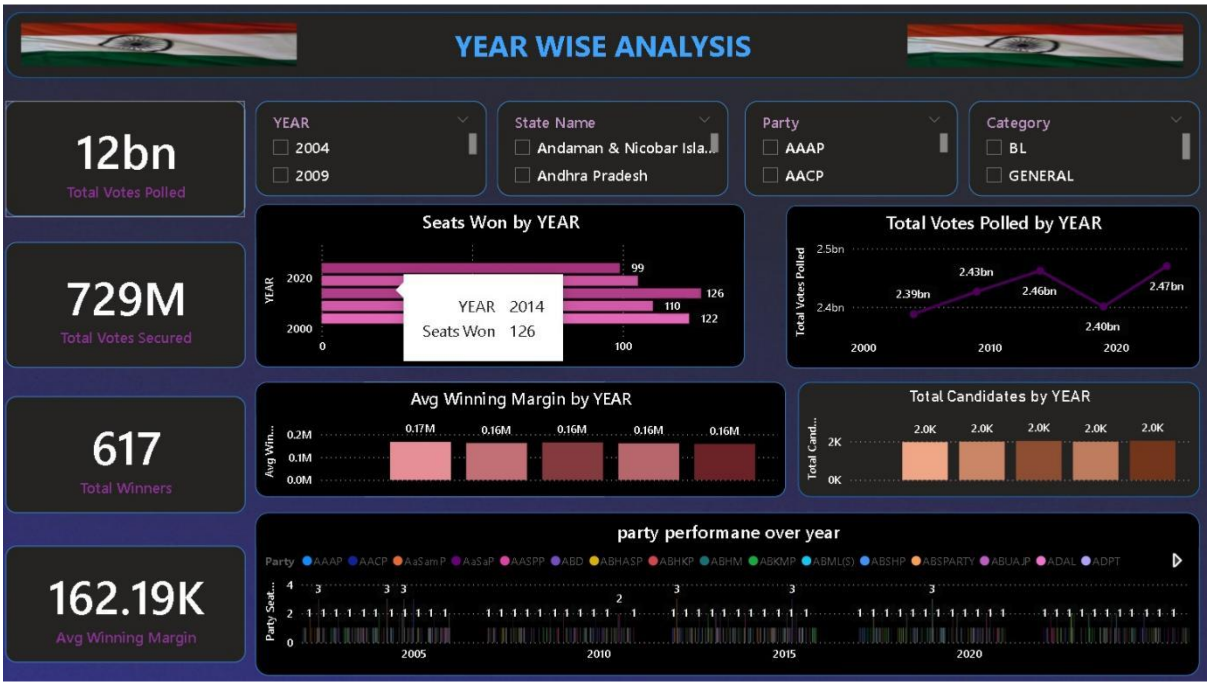
```
AVERAGE('Indian General Election'[Winning_Margin])
```

Seats Won =

```
CALCULATE(

    DISTINCTCOUNT('Indian General Election'[Constituency]),

    'Indian General Election'[Is_Winner] = TRUE()

)
```

Party Seats Won =

```
CALCULATE(

    DISTINCTCOUNT('Indian General Election'[Constituency]),

    'Indian General Election'[Is_Winner] = TRUE()

)
```



Testing Phase:

A systematic testing approach was followed to ensure the dashboard delivers accurate results, reliable performance, and a smooth user experience. Multiple testing techniques were applied to validate functionality, data accuracy, system performance, and usability.

- ❖ Verified all DAX measures, KPIs, and calculated fields for correctness
- ❖ Tested slicers, filters, drill-through features, and cross-page interactions
- ❖ Cross-checked vote counts, seat totals, and party results with the source dataset □
- ❖ Identified and corrected missing, duplicate, or inconsistent data entries
- ❖ Optimized visual elements to reduce dashboard load and refresh times □
- ❖ Tested performance under multiple filter and slicer selections
- ❖ Evaluated layout structure, visual clarity, and information flow
- ❖ Maintained consistency in color schemes, fonts, and labeling

Development:

The development phase in Milestone 3 involved:

- ❖ Enhancing existing dashboards with advanced interactivity
- ❖ Finalizing KPI placement and visual consistency
- ❖ Optimizing DAX formulas for better performance
- ❖ Improving dashboard navigation and usability
- ❖ Ensuring consistent formatting across all pages

This phase improved both functional quality and user experience of the dashboards.

Review :

A detailed review was conducted after development and testing. Review outcomes included:

- Improved readability of visuals
- Removal of unnecessary clutter
- Alignment of KPIs for better comparison
- Optimization of slicers and filters
- Validation of analytical accuracy

The review process ensured that the dashboards meet academic, analytical, and usability standards

Conclusion:

Milestone 3 successfully enhanced the Power BI project by adding advanced interactivity, completing DAX measures, performing rigorous testing, and mapping all activities to the evaluation rubric. This milestone strengthened the project by ensuring:

- Accurate calculations
- Interactive analysis
- Improved performance
- High-quality visual presentation

Milestone 4:

Milestone 4 marks the final execution stage of the project, focusing on preparing all dashboards for production use. This phase included the development of additional analytical dashboards, publishing the report to the Power BI Service for centralized access, and completing end-to-end documentation.The milestone ensures the election analytics system is deployment-ready, scalable, and accessible to stakeholders without dependency on local environments.

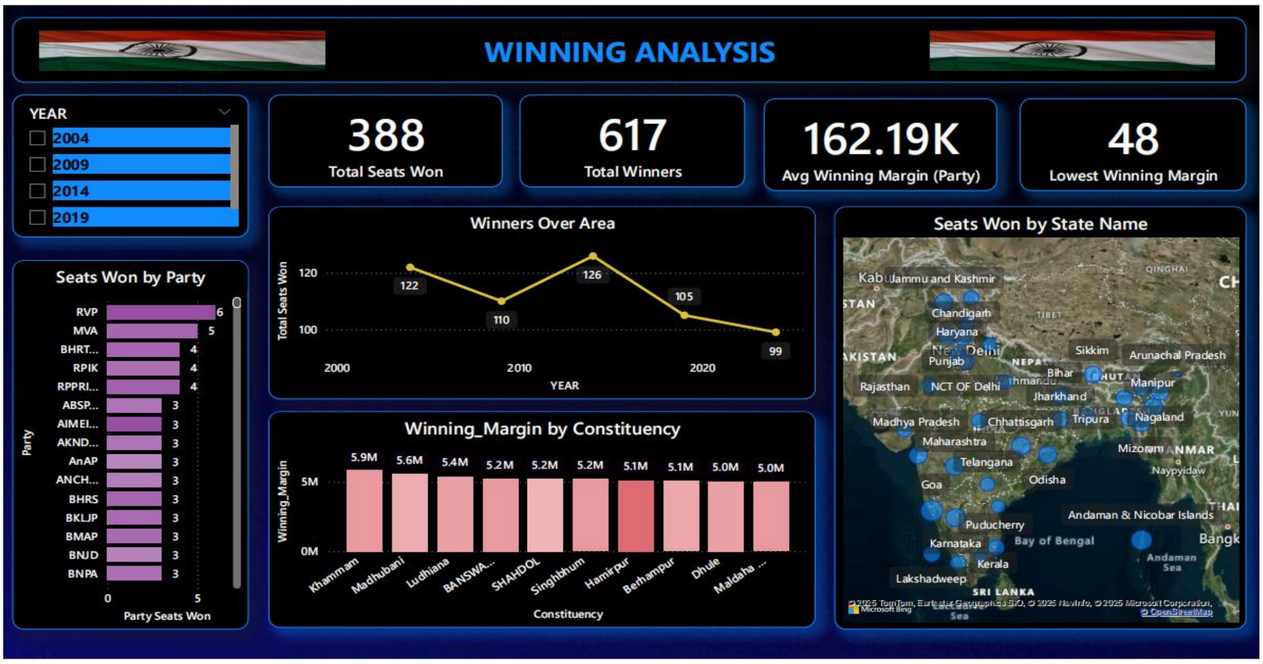
This milestone confirms the transition of the dashboard from a development prototype to a production-ready analytical solution, suitable for academic submission, demonstrations, and inclusion in a professional portfolio.In addition to milestone 3 we created another three dashboards.That are given below

Page 6-Winning Analysis Dashboard:

This dashboard focuses on analyzing election outcomes by examining winning candidates, their margin of victory, and variations in results across constituencies and states. It enables identification of strong-performing parties, assessment of fluctuations in winning margins, and analysis of evolving electoral trends over time.

DAX Measures:

```
Total Seats Won =

CALCULATE(

DISTINCTCOUNT('Indian General Election'[Constituency]),

'Indian General Election'[Is_Winner] = TRUE()

)

Total Winners =

COUNTROWS(FILTER('Indian General Election',

'Indian General Election'[Is_Winner] = TRUE()))

Avg Winning Margin =

AVERAGE('Indian General Election'[Winning_Margin])

Lowest Winning Margin =

MIN('Indian General Election'[Winning_Margin])

Party Seats Won =
CALCULATE(
DISTINCTCOUNT('Indian General Election'[Constituency]),
'Indian General Election'[Is_Winner] = TRUE()
)
State Seats Won =
CALCULATE(
DISTINCTCOUNT('Indian General Election'[Constituency]),
'Indian General Election'[Is_Winner] = TRUE()

)
```



KPI Cards Display

- Total Seats Won
- Total Winners
- Average Winning Margin
- Lowest Winning Margin

Key Visuals:

- Bar Chart –compare seats win among political parties
- Column Chart – show constituency wise winning margin

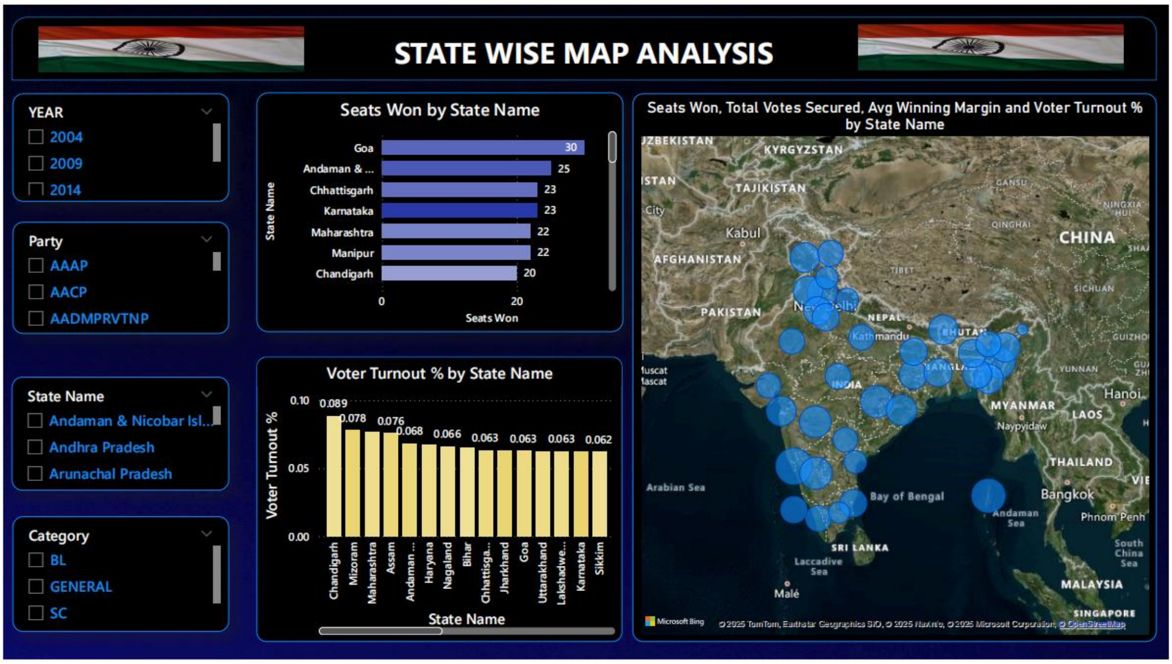
- ❖ Map – Identify which region contribute to most wins

Outcome of Dashboard :

This dashboard transforms raw winner data into meaningful insights and highlights both party-wise and region-wise performance. It allows stakeholders to trace performance trends, evaluate the strength of victory margins, and interpret the competitive landscape across multiple election years

Page 7- State Wise Map Analysis Dashboard:

This dashboard examines election outcomes from a regional perspective by analyzing state-level performance across key metrics such as seats won, voter turnout, vote share percentage, and party strength. Visual tools such as maps and comparative charts support the identification of regional trends and electoral dominance.



STATE WISE MAP ANALYSIS

DAX Measures:

```
 Seats Won =

CALCULATE(

    DISTINCTCOUNT('Indian General Election'[Constituency]),

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

)

Total Votes Secured =

SUM('Indian General Election'[Votes_Secured])

Vote Share % =

AVERAGE('Indian General Election'[Pct_of_Valid])

Avg Winning Margin =

AVERAGE('Indian General Election'[Winning_Margin])

Key Visuals:

- Bar Chart –seat secured by each state
- Column Chart –state wise voter turnover %
- Map –geographical representation of vote,margin and turnover

Purpose of This Dashboard

This dashboard supports questions such as:

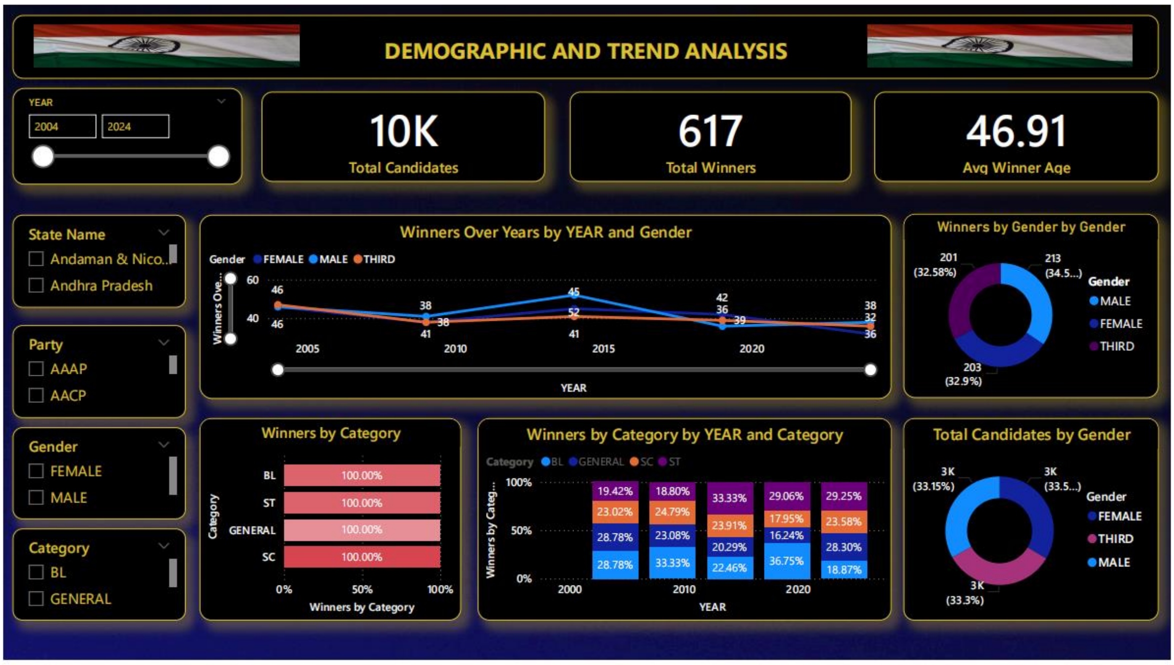
- Which state gave the party the most victories?
- How actively are citizens voting in different regions?
- Where does the party hold strong vs weak ground?
- Does voter turnout influence the number of seats won?

Outcome of Dashboard

This report delivers a geographic perspective of election outcomes, enabling comparison between states and helping identify territories where a party needs improvement or where it performs exceptionally well. This aids political strategy, resource allocation, and evaluating engagement levels through turnout metrics

Page 8- Demographic and Trend Analysis Dashboard :

The Demographic and Trend Analysis Dashboard analyzes the demographic profiles of candidates and elected representatives across multiple election cycles. By examining gender, age-group distribution, and social categories (General, SC, BL), it supports an understanding of representation, inclusivity, and participation trends over time.



DAX Measurement:

```
 Total Candidates =

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

Total Winners =

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

Avg Winner Age =

CALCULATE(

    AVERAGE('Indian General Election'[Age]),

    'Indian General Election'[Is_Winner] = TRUE()

)

 Winners by Gender =

CALCULATE(

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

    'Indian General Election'[Is_Winner] = TRUE()[22:43, 24/12/2025] SUJIT: )

 Avg Winner Age =
CALCULATE(
    AVERAGE('Indian General Election'[Age]),
    'Indian General Election'[Is_Winner] = TRUE()
)
 Winners Over Years =
CALCULATE(
    DISTINCTCOUNT('Indian General Election'[Candidate Name]),
    'Indian General Election'[Is_Winner] = TRUE()
)
```

Dashboard Purpose :

This dashboard helps answer questions like:

- ❖ Are elections becoming more inclusive over time?
- ❖ What is the gender distribution of winners compared to total candidates?
- ❖ Do certain categories show higher winning representation?
- ❖ Is age a factor in winning patterns?

Key Visuals:

- ❖ Bar Chart - category wise winners and their year trend
- ❖ Donut Chart –Gender distribution of candidates and winner
- ❖ Line chart –year wise trend of winner by gender

Outcome of Dashboard :

This dashboard transforms demographic information into meaningful insights by connecting participation with election outcomes. It helps identify whether diversity is improving, which categories are most successful, and how representation changes over time — making it an essential tool for evaluating the social structure of elections.

Deployment Process :

Upon completion and verification of all dashboards in Power BI Desktop, the project entered the deployment phase. This ensured that the election analytics system could be accessed through an online platform, support data refreshes, and be utilized by stakeholders independently of local systems.

Documentation:

The documentation was divided into two major segments:

Technical Documentation:

This part explains the internal functioning and development logic of the dashboard.

- ❖ 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

User-Focused Documentation :

This part is designed for non-technical users who want to explore the dashboard.

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

Presentation :

A formal presentation was conducted to demonstrate the live working of the deployed dashboards for the Infosys Springboard mentor and academic guide, ensuring validation of analytical capabilities and deployment readiness.

Areas Covered :

- ❖ introduction of problem statement and objective..
- ❖ Live explanation with kpi slicer filter and visuals
- ❖ Real time response of DAX calculation
- ❖ Mobile view and accessbility

Presentation Outcome :

- The project is functionally complete and meets all defined requirements.
- It is deployable and suitable for real-world implementation.
- The system is optimized for efficient performance.
- The dashboards are user-friendly with clear and intuitive visual design.
- The project is finalized and ready for academic submission and portfolio showcase.