ELECTVIZ - ELECTION DATA VISUALIZATION FOR MEDIA

INFOSYS SPRINGBOARD INTERNSHIP 6.0

Milestones Summary Report

By – Ashish Ransing (Team C : Group Leader)

Milestone – 1

Requirement Gathering & Planning

Table of Content

Project Overview

❖ **Executve summary:**

The objective of this project is to develop a broadcast-quality **Election Intelligence Dashboard** tailored for media use. Unlike standard corporate reports, this "Command Center" focuses on speed, storytelling, and high-impact visualization.

Using a **"Macro-to-Micro"** analytical approach, the dashboard transforms 40 years of raw historical data (Lok Sabha & Vidhan Sabha) into actionable insights, enabling users to instantly track national trends, identify swing seats, and analyze party efficiency.

❖ **Project objectives:**

1. **Historical Archiving:** To standardize and visualize over 400,000 election records from 1977 to 2014.
2. **Metric Innovation:** To engineer "hidden" metrics not found in raw data, such as **Strike Rate**, **Winning Margin**, and **Voter Turnout %**.
3. **Media-Ready Storytelling:** To create a visual narrative that supports live reporting, featuring "Nail-Biter" contests, "Landslide" victories, and demographic shifts.

❖ **Teck Stack:**

Layer	Technology Selected	Purpose
ETL & Data Engineering	Python	Core language for data processing and logic.
Data Manipulation	Pandas, NumPy	Used for cleaning, merging, and calculating complex metrics (Margins/Ranks).
Exploratory Analysis	Matplotlib, Seaborn	Used for generating initial statistical visualisations (Box Plots, Histograms).
Visualization / BI	Microsoft Power BI	The primary dashboarding tool for the final user interface.
Data Storage	CSV (Flat Files) / Excel files	Storage for the 400,000+ raw and processed election records.
Asset Management	Microsoft Excel	Used to create the Party_Master dimension table (New data consisting Logos of each Party).
IDE / Environment	Google collab/ Jupyter Notebook	Development environment for writing and testing Python scripts.

Product Backlog / Requirements

Theme: Creating a broadcast-ready dashboard for media analysts to visualize 40 years of Indian Election history.

Sr.no	Requirements	Acceptance Criteria	Priority Level
1	**Data Standardization -** As for the initiation of project , we need to clean the raw CSVs so that party names and years are consistent across 40 years.	All variations (e.g., "INC(I)", "Congress") mapped to "INC". No missing values in critical columns.	High
2	**National Overview (Macro) -** As Analysts, we want to see a national heat map and seat share to understand the ruling party's dominance.	Interactive Map of India. Donut chart showing "Seats Won" > 272 (Majority).	High
3	**State Drill-Down (Micro) -** We have to filter data by specific states (e.g., UP, Bihar) to see local trends.	Slicer for State selection. Trend line showing State Assembly results over time.	Medium
4	**Candidate Efficiency –** We have to analyze "Strike Rates" and "Turnout rates" to see which party fights the most efficient elections.	Scatter plot: Seats Contested vs. Seats Won. Metrics for Candidate Win %.	Medium
5	**The "War Room" -** As per the theme media we assume a TV Anchor needs to instantly identify "Nail-Biter" contests (Low Margin) and "Landslides."So we will include this page as well in our dashboard	Filter for Winning Margin < 1,000 votes. Highlight swing constituencies.	High

Requirement Analysis

A. Data Analysis & Feasibility:

We analyzed two primary datasets-

1. indian-national-level-election.csv (Lok Sabha Data: 1977–2014)
2. indian-state-level-election.csv (Vidhan Sabha Data)

- **Key Data Gaps Identified:**
  - **Missing Metrics:** The raw data contains *Votes Polled* but lacks derived metrics like *Voter Turnout %*, *Winning Margin*, and *Strike Rate*.
  - **Inconsistency:** Party names vary (e.g., "BJP" vs "Bharatiya Janata Party").
  - **No Media Assets:** The data lacks visual elements like Party Logos or Hex Colors.

B. Functional Requirements (The Logic and Formulas):

- **Winning Margin Calculation:**
  - Margin = (Votes{Winner} - Votes{RunnerUp})
- **Voter Turnout %:**
  - Turnout = ({Total Valid Votes}/ {Electors}) x 100
- **Winner Identification:**
  - Rank 1 = Winner, Rank > 1 = Loser. (Hard-coded during ETL).

C. Non-Functional Requirements:

- **Performance:** Dashboard must load 40 years of data in less time depending upon PC specifications.
- **Usability:** "Macro-to-Micro" navigation flow (National -> State -> Constituency).
- **Aesthetics:** Strict color coding (According to theme and Party color) for media consistency.

Backlog Grooming / Finding Solutions

Requirement Question	Discussion/Challenge	Resolution (Outcomes)
How do we handle missing gender data?	~500 candidates have Null gender. Dropping them affects total vote counts.	**Decision:** Filling Nulls with "Unknown" to preserve the Total Votes sum for Turnout calculations.
How do we visualize "Close Contests"?	The raw data only shows the winner, not the runner-up's proximity.	**Decision:** Created a calculated column Margin_Votes in Python. Define "Close Contest" as < 5,000 votes.
How do we handle By-Elections?	State data has years like 2005.1 indicating by-elections.	**Decision:** Casting all years to Integer (Int) to group them into the main election cycle for cleaner trending.
Where do we get Party Logos?	Raw data doesn't include party logos or images.	**Decision:** We will create an external Party_Master.xlsx file containing Logo URLs and Hex Codes to link in Power BI. (Taking logos/images from web)

Sprints / Execution Plan

Sprint 1: Data Engineering:

- **Goal:** Clean, enriched datasets ready for import.
- **Tasks:**
  - Writing Python script to standardize Party Names.
  - Calculating Ranks and Margins using Pandas.
  - Performing EDA (Histograms/Box Plots) to check outliers.

Sprint 2: Core Dashboarding:

- **Goal:** Pages 1 and 2 (Macro View).
- **Tasks:**
  - Importing Data & Build Data Model (Schema).
  - Creating DAX Measures (Total Seats, Vote Share %).
  - Designing National Heat Map & Parliament Chart.
  - Designing State Slicers & Trend Lines.

Sprint 3: Advanced Analytics & UI:

- **Goal:** Pages 3, 4, 5 (Micro View & Polish).
- **Tasks:**
  - Building "Strike Rate" Scatter Plot and other useful charts for making dashboard effective.
  - Design "War Room" (Page 5) with Margin Filters.
  - Integrate Party Logos & Color Formatting.
  - Final QA , Presentation & Documentation.

Team Meetings

1. **Till Date - 2 Official Meetings (Introduction, Discussing Approach of the Project)**
2. **Several One-on-One Calls and Discussions (During Data cleaning , visualizations, roles, etc)**

Milestone – 2

Development Phase – Part 1

Table of Content

Data Modelling Architecture

❖ **Objective:** To design and implement a scalable schema facilitating seamless cross-filtering between National (Lok Sabha) and State (Vidhan Sabha) datasets.

❖ **Schema Topology:** Galaxy Schema (Multi-Fact Architecture).

❖ **Fact Tables (Transactional Data):**

  1. **national_cleaned_enriched:** Contains granular Lok Sabha election results.
  2. **state_cleaned_enriched:** Contains granular Vidhan Sabha election results.

❖ **Dimension Tables (Lookup Masters):**

  3. **Dim_State:** Standardized state list ensuring accurate geolocating and filtering across maps.
  4. **Dim_Year:** Unified timeline dimension for temporal analysis.
  5. **Party_Master:** Centralized branding repository containing Party Codes, Names, Logos, and standardized HEX Colors.

❖ **Relationship Logic:** One-to-Many relationships established between Dimension and Fact tables, ensuring that slicers (Year, State) operate dynamically across all report pages.
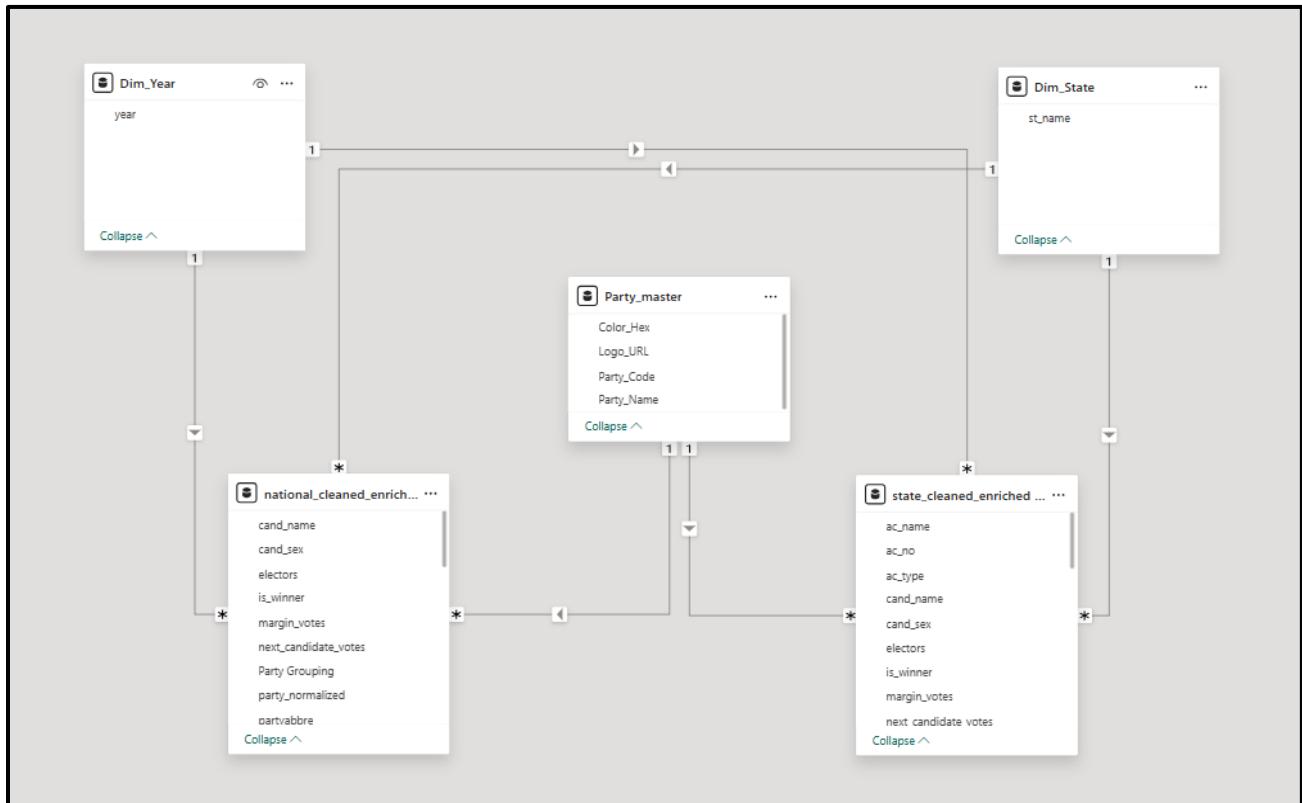

Fig 1 : Data Model Architecture

Key Dax Measures (Initial Implementation)

❖ **Development Status:** 28 Core Measures have been implemented. Below are the critical formulas powering the visuals for Pages 1 and 2.

A. Headline KPIs (Aggregation Logic):

- **Total Seats Won:**
  - Calculates the total constituencies secured by a party.
  - Nat Total Seats Won =
    CALCULATE(COUNTROWS('national_cleaned_enriched'),'national_cleaned_enriched'[is_winner] = 1)

- **Total Votes Polled:**
  - Aggregates the total valid votes cast in the election.
  - Nat Total Votes = SUM('national_cleaned_enriched'[totvotpoll])

- **Democratic Turnout %:**
  - Computes the average voter turnout percentage.
  - Nat Avg Turnout % = AVERAGE('national_cleaned_enriched'[turnout_pct])

B. Geospatial Logic (Smart Formatting):

- **Winner Color (Dynamic HEX Assignment):**
  - Retrieves the official party color from the Master table; defaults to Grey (#808080) for independent/minor parties to ensure visual integrity.
  - Nat Winner Color =
    VAR TopPartyTable = TOPN(1, VALUES('national_cleaned_enriched'[party_normalized]), [Nat Total Seats Won], DESC)
    VAR TopPartyName =
    MAXX(TopPartyTable,'national_cleaned_enriched'[party_normalized])
    VAR RealColor = LOOKUPVALUE('Party_Master'[Color_Hex], 'Party_Master'[Party_Code], TopPartyName)
    RETURN
    COALESCE(RealColor, "#808080")

C. Advanced Analytics (Context Aware):

- **Vote Share % (Context Fixed):**
  - Calculates a party's popularity against the total national vote, ignoring specific party filters to provide accurate context.
  - Nat Vote Share % =
    VAR PartyVotes = [Nat Total Votes]
    VAR AllVotes = CALCULATE([Nat Total Votes], REMOVEFILTERS('Party_Master'))
    RETURN
    DIVIDE(PartyVotes, AllVotes, 0)

Dashboard Visualization

Page 1: National Election Command Center

- ❖ **Theme:** "The Macro View"
- ❖ **Strategic Purpose:** A high-level executive dashboard designed to instantly visualize the status of the central government, parliamentary majority, and geographic dominance.
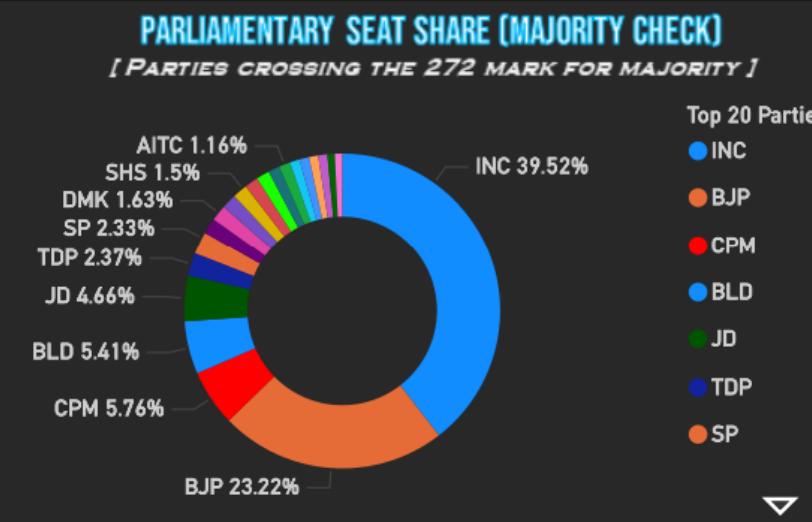
Visual Component	Type	Configuration & Logic	Strategic Rationale
Global Controller	Slicer	Field: Dim_Year[year]	Acts as a "Time Machine," allowing stakeholders to switch context between election cycles (e.g., 1984 vs. 2014) to analyze historical shifts.

YEAR
All ⌄

Headline KPIs	5-Card Layout	Metrics: Seats Declared, Total Votes, Voter Turnout, Total Parties, Total Candidates.	Provides an immediate "Health Check" of the election scale and participation levels before users navigate to complex analytics.

SEATS DECLARED	TOTAL PARTIES	TOTAL CANDIDATES
6K	1K	73K

TOTAL VOTES POLLED	VOTER TURNOUT %
4bn	57.85

Majority Gauge	Donut Chart	Legend: Party_Code Values: Nat Total Seats Won	Instantly answers the critical question: "*Did any party cross the majority mark (272)?*" visually representing the parliament floor.
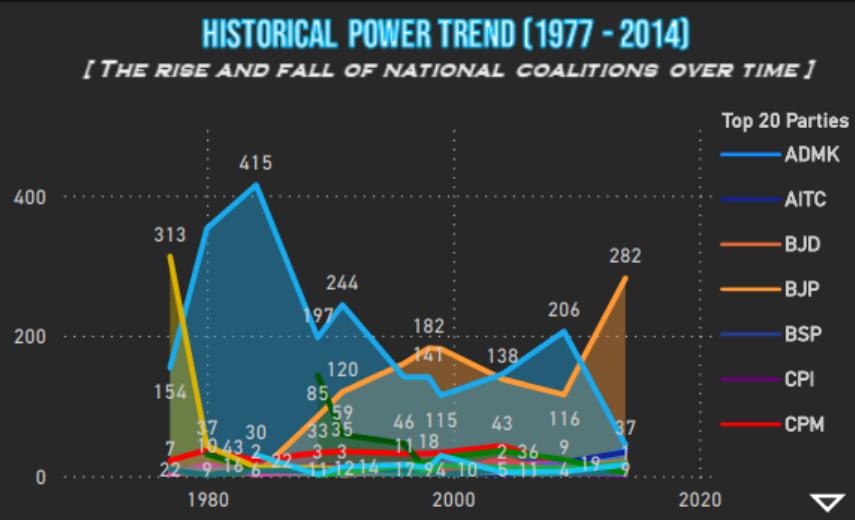
| Trend Analysis	Line Chart	Axis: Year Values: Seats Won	Contextualizes the current result, distinguishing between sudden "Waves" and long-term organic growth patterns since 1977.
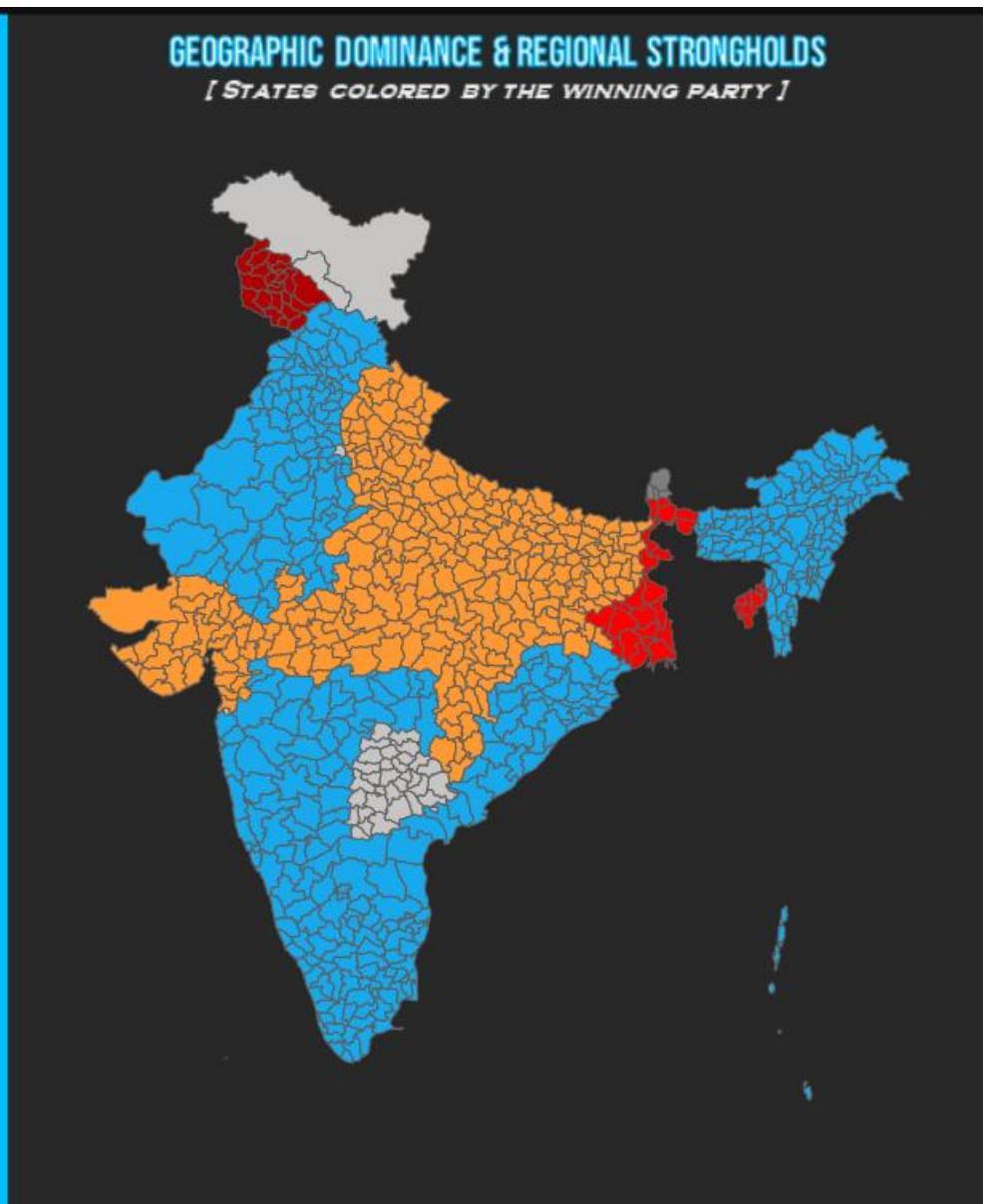
Geographic Map	Shape Map	Loc: Dim_State Color: Nat Winner Color	Visualizes "Regional Strongholds," revealing North-South or East-West political divides using official party colors.
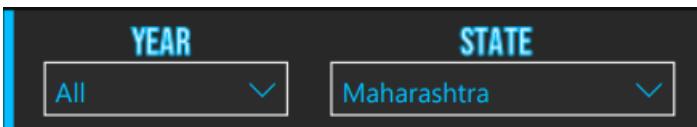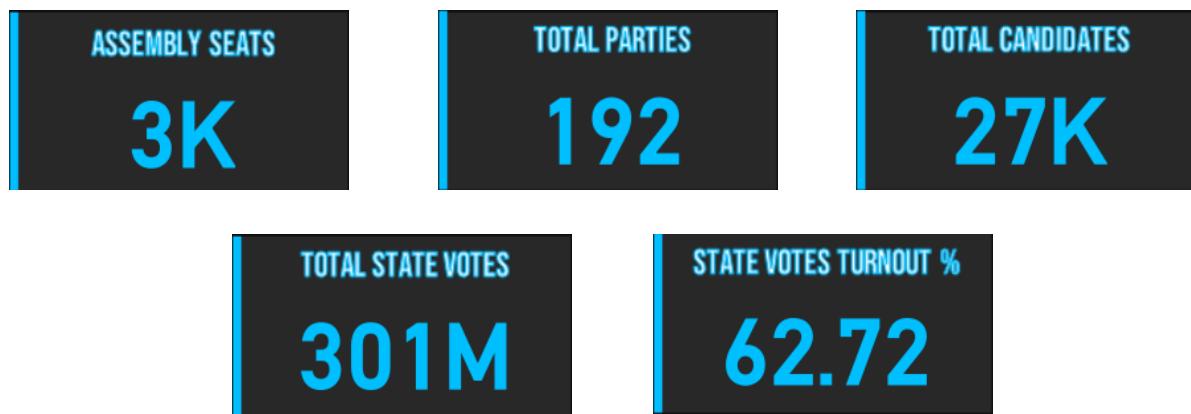
GEOGRAPHIC DOMINANCE & REGIONAL STRONGHOLDS

[STATES COLORED BY THE WINNING PARTY]
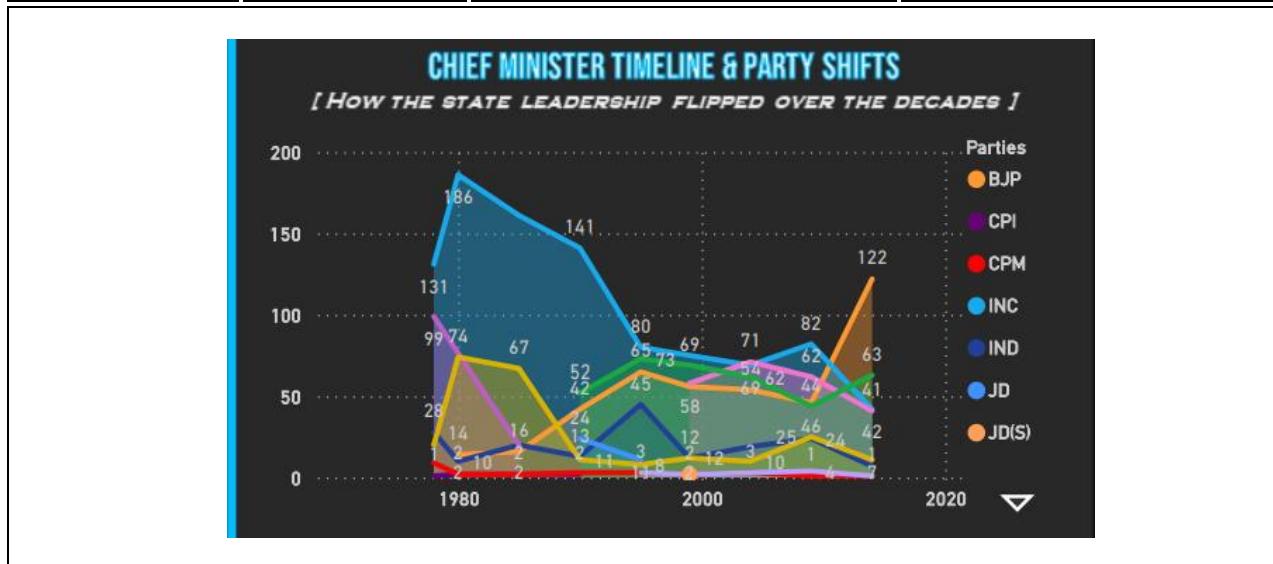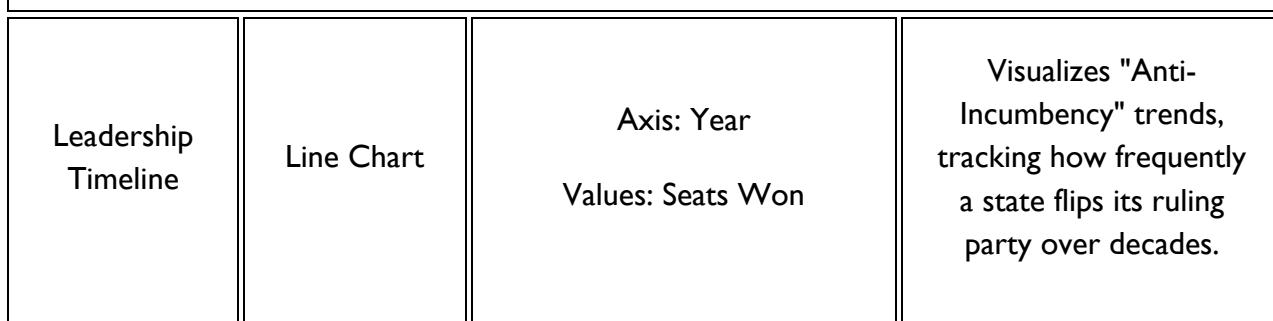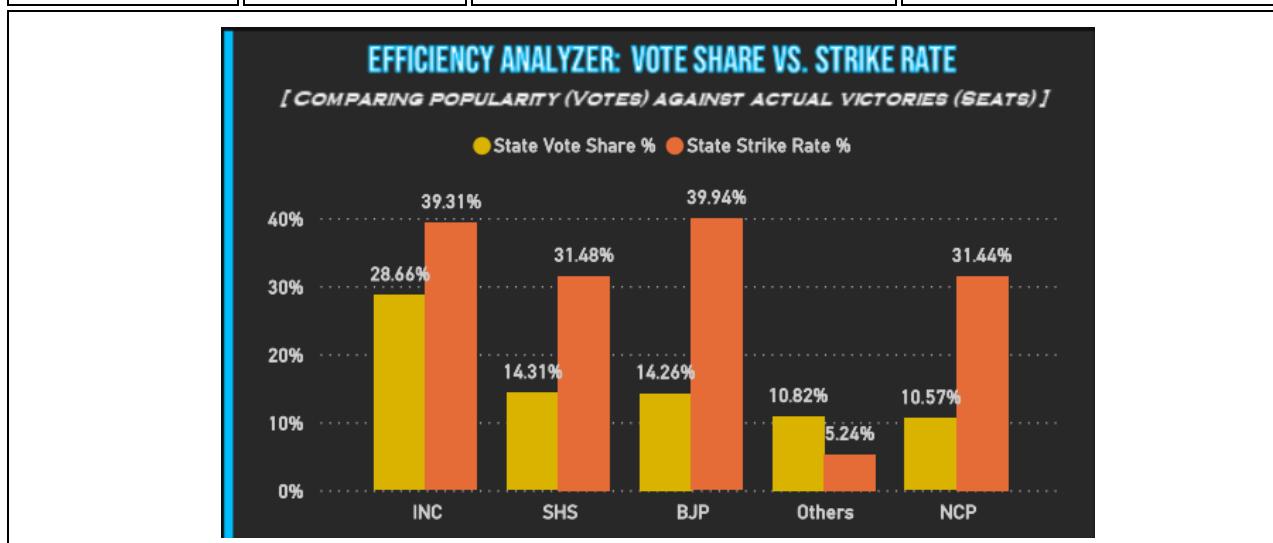
Page 2: State Battlegrounds

❖ **Theme:** "The Regional Deep Dive"
Strategic Purpose: A granular analysis tool for specific Legislative Assembly (Vidhan Sabha) results, focusing on local efficiency and leadership stability.

Visual Component	Type	Configuration & Logic	Strategic Rationale
Dual Controllers	Slicer	Fields: Dim_State, Dim_Year	Enables drill-down capabilities, allowing users to isolate specific regions (e.g., Uttar Pradesh) and specific assembly terms.

YEAR	STATE
All ⌄	Maharashtra ⌄

Visual Component	Type	Configuration & Logic	Strategic Rationale
Regional KPIs	5-Card Layout	Metrics: Assembly Seats, State Votes, State Turnout, Local Parties, Candidates.	Resets the analytical context from "National" to "Local" metrics immediately upon state selection.

ASSEMBLY SEATS
3K

TOTAL PARTIES
192

TOTAL CANDIDATES
27K

TOTAL STATE VOTES
301M

STATE VOTES TURNOUT %
62.72

Efficiency Analyzer	Clustered Column	Y-Axis: Vote Share % vs. Strike Rate %	Exposes "First Past the Post" realities by comparing Popularity (Votes) against Efficiency (Seat Conversion Rate).

Leadership Timeline	Line Chart	Axis: Year Values: Seats Won	Visualizes "Anti-Incumbency" trends, tracking how frequently a state flips its ruling party over decades.
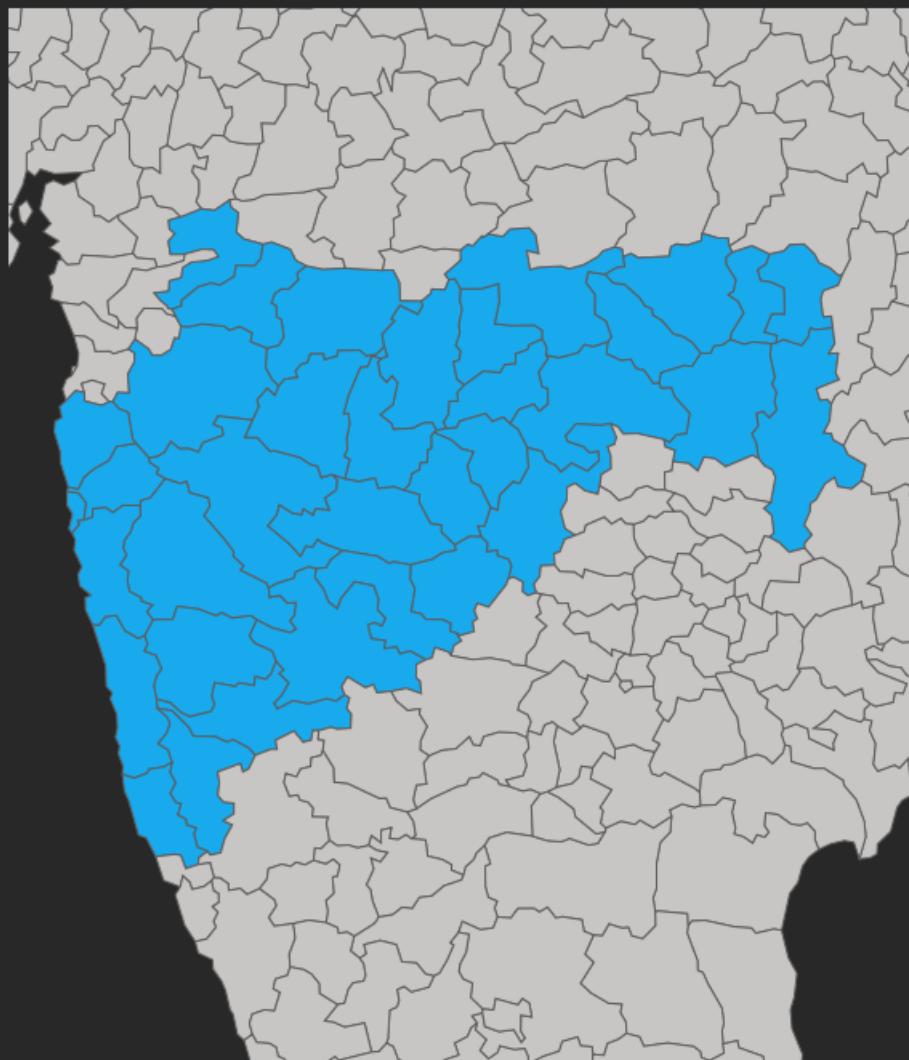
Assembly Map	Shape Map	Loc: Dim_State Color: State Winner Color	Displays "Micro-Level" dominance, revealing distinct voting belts (e.g., rural vs. urban) within the selected state.

ASSEMBLY CONSTITUENCY MAP
[Detailed constituency-level winners]

Sprint Review & Validation

A. Visual Verification:

- Color Consistency: Verified that major parties retain consistent branding (BJP: Saffron #FF9933, INC: Blue #19AAED) across all pages. Confirmed that independent/minor parties correctly default to Grey (#808080) via COALESCE logic.

- Map Integrity: Validated the National Map for the 2014 dataset. Resolved geospatial mismatches in the State Battlegrounds map (specifically the "Orissa/Odisha" naming convention) to ensure 100% data visibility.

B. Data Validation:

- Aggregation Check: Performed cross-verification where the sum of State Total Votes in granular views exactly matched the Nat Total Votes KPI, confirming accurate row-level aggregation.

- Filter Logic: Verified that the "State Slicer" automatically filters out Union Territories without legislative assemblies (e.g., Chandigarh) by applying visual-level filters (Seats > 0).

C. Feedback Implementation:

- Layout Optimization: Expanded the KPI layout from 4 to 5 cards (adding "Total Parties") to improve visual symmetry and provide better context on the political scale.

- Typography Upgrade: Standardized typography to "Bebas Neue" for all headline metrics to achieve a professional "Command Center" aesthetic.

Team Stand-Up Record

❖ **Sprint Cycle: 1-2:**

1. **Kick-off Meeting:** Defined the Galaxy Schema architecture and visualized the requirement for "Safe Mode" DAX to handle missing party colors.
2. **Implementation Reviews:** Conducted 5-6 one-on-one sessions during the development phase to address:
   a. DAX Context Transition issues in Vote Share calculation.
   b. Visual formatting of the custom India TopoJSON map.
   c. Data standardization (State name cleaning) via Python.

Next Sprint Goals

(Sprint 3)

❖ **Focus: Advanced Analytics & UI Polish (Pages 3, 4, & 5)**

- **Advanced Charts:** Implementation of "Strike Rate" Scatter Plots for strategic analysis.

- **The War Room:** Development of Page 5 featuring Margin Filters to isolate "Close Contests" (<5,000 votes).

- **UI/UX Finalization:** Integration of high-resolution Party Logos and final color formatting.

- **Submission:** Final Quality Assurance (QA) and project documentation.

Milestone – 3

Development Phase – Part 2 + Testing

Table of Content

Advanced Dax Logic & Measure Definitions

Development Status: Completion of 28+ complex measures enabling the "War Room" logic and "Efficiency" analysis. Below are all the advanced formulas implemented during this sprint.

MEASURE NAME	DAX FORMULAS
Nat Total Seats Won	Nat Total Seats Won = CALCULATE(COUNTROWS('national_cleaned_enriched'), 'national_cleaned_enriched'[is_winner] = 1)
Nat Total Votes	Nat Total Votes = SUM('national_cleaned_enriched'[totvotpoll])
Nat Avg Turnout %	Nat Avg Turnout % = AVERAGE('national_cleaned_enriched'[turnout_pct])
Nat Total Candidates	Nat Total Candidates = COUNTROWS('national_cleaned_enriched')
Nat Total Parties	Nat Total Parties = DISTINCTCOUNT('national_cleaned_enriched'[party_normalized])
State Total Seats Won	State Total Seats Won = CALCULATE(COUNTROWS('state_cleaned_enriched'), 'state_cleaned_enriched'[is_winner] = 1)
State Total Votes	State Total Votes = SUM('state_cleaned_enriched'[totvotpoll])

|	
State Avg Turnout %	State Avg Turnout % = AVERAGE('state_cleaned_enriched'[turnout_pct])
State Total Candidates	State Total Candidates = COUNTROWS('state_cleaned_enriched')
State Total Parties	State Total Parties = DISTINCTCOUNT('state_cleaned_enriched'[party_normalized])
Nat Seats Contested	Nat Seats Contested = COUNTROWS('national_cleaned_enriched')
Nat Strike Rate %	Nat Strike Rate % = DIVIDE([Nat Total Seats Won], [Nat Seats Contested], 0)
Nat Vote Share %	Nat Vote Share % = VAR PartyVotes = [Nat Total Votes] VAR AllVotes = CALCULATE([Nat Total Votes], ALL('national_cleaned_enriched'[party_normalized])) RETURN DIVIDE(PartyVotes, AllVotes, 0)
State Seats Contested	State Seats Contested = COUNTROWS('state_cleaned_enriched')
State Strike Rate %	State Strike Rate % = DIVIDE([State Total Seats Won], [State Seats Contested], 0)
State Vote Share %	State Vote Share % = VAR PartyVotes = [State Total Votes] VAR AllVotes = CALCULATE([State Total Votes], ALL('state_cleaned_enriched'[party_normalized])) RETURN DIVIDE(PartyVotes, AllVotes, 0)
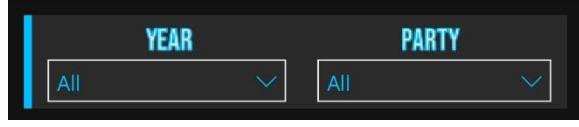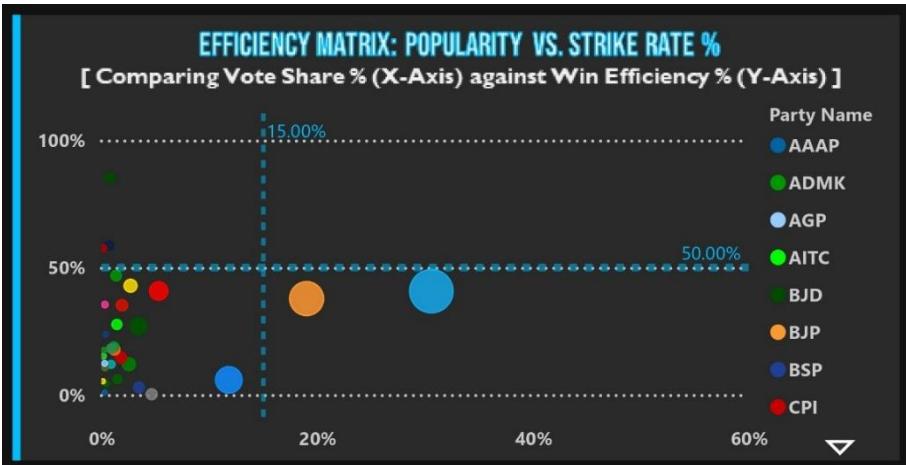
State Avg Win Margin	State Avg Win Margin = CALCULATE(AVERAGE('state_cleaned_enriched'[margin_votes]), 'state_cleaned_enriched'[is_winner] = 1)
Nat Avg Win Margin	Nat Avg Win Margin = CALCULATE(AVERAGE('national_cleaned_enriched'[margin_votes]), 'national_cleaned_enriched'[is_winner] = 1)
Nat Close Contests	Nat Close Contests = CALCULATE([Nat Total Seats Won], 'national_cleaned_enriched'[margin_votes] < 5000, 'national_cleaned_enriched'[is_winner] = 1, 'national_cleaned_enriched'[margin_votes] > 0)
Nat Landslide Victories	Nat Landslide Victories = CALCULATE([Nat Total Seats Won], 'national_cleaned_enriched'[margin_votes] >= 100000, 'national_cleaned_enriched'[is_winner] = 1) + 0
Margin Category	Margin Category = SWITCH(TRUE(), 'national_cleaned_enriched'[margin_votes] < 5000, "1. Nail Biter", 'national_cleaned_enriched'[margin_votes] < 20000, "2. Close Fight", 'national_cleaned_enriched'[margin_votes] < 100000, "3. Safe Seat", 'national_cleaned_enriched'[margin_votes] >= 100000, "4. Landslide", "Other")
Nat Female Candidates	Nat Female Candidates = CALCULATE(COUNTROWS('national_cleaned_enriched'), 'national_cleaned_enriched'[cand_sex] = "F")
Nat Female Winners	Nat Female Winners = CALCULATE([Nat Total Seats Won], 'national_cleaned_enriched'[cand_sex] = "F")
Nat Female Success Rate	Nat Female Success Rate = VAR Candidates = [Nat Female Candidates] VAR Winners = [Nat Female Winners] RETURN IF(Candidates = 0, 0, DIVIDE(Winners, Candidates, 0))
Nat Winner Name	Nat Winner Name = VAR TopParty = TOPN(1, VALUES('national_cleaned_enriched'[party_normalized]), [Nat Total Seats Won], DESC) RETURN CONCATENATEX(TopParty, 'national_cleaned_enriched'[party_normalized], ", ")

|	
Nat Winner Color	Nat Winner Color = VAR TopPartyTable = TOPN(1, VALUES('national_cleaned_enriched'[party_normalized]), [Nat Total Seats Won], DESC) VAR TopPartyName = MAXX(TopPartyTable, 'national_cleaned_enriched'[party_normalized]) VAR RealColor = LOOKUPVALUE('Party_Master'[Color_Hex], 'Party_Master'[Party_Code], TopPartyName) RETURN COALESCE(RealColor, "#808080")
State Winner Name	State Winner Name = VAR TopParty = TOPN(1, VALUES('state_cleaned_enriched'[party_normalized]), [State Total Seats Won], DESC) RETURN CONCATENATEX(TopParty, 'state_cleaned_enriched'[party_normalized], ", ")
State Winner Color	State Winner Color = VAR TopPartyTable = TOPN(1, VALUES('state_cleaned_enriched'[party_normalized]), [State Total Seats Won], DESC) VAR TopPartyName = MAXX(TopPartyTable, 'state_cleaned_enriched'[party_normalized]) VAR RealColor = LOOKUPVALUE('Party_Master'[Color_Hex], 'Party_Master'[Party_Code], TopPartyName) RETURN COALESCE(RealColor, "#808080")
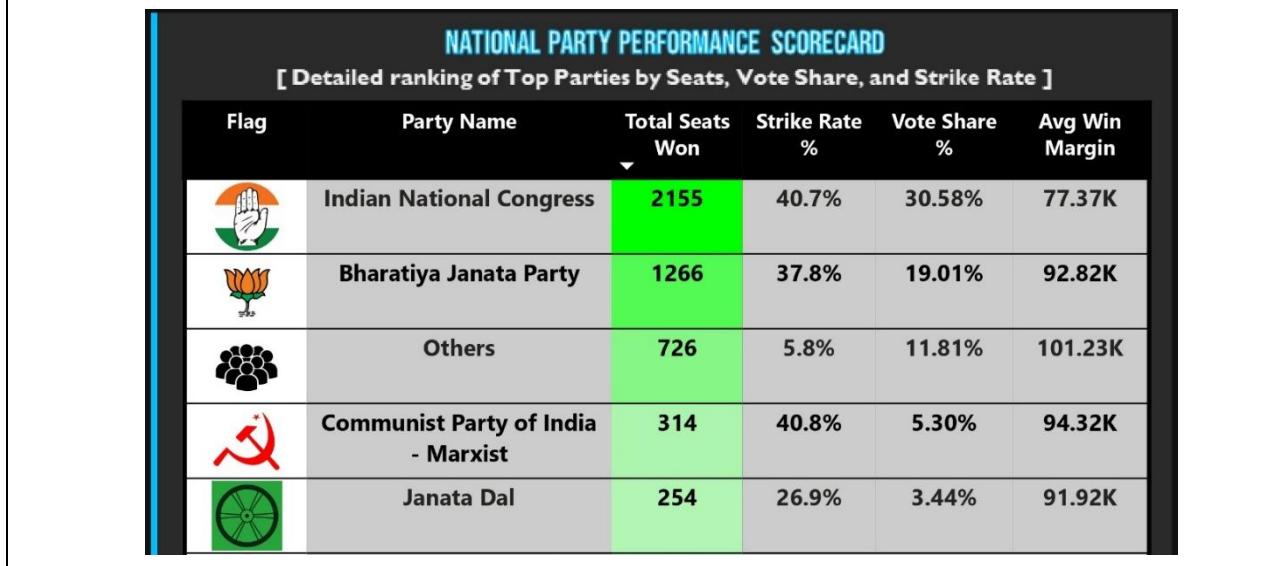
DASHBOARD VISUALIZATION

PAGE 3: PARTY STRATEGY & PERFORMANCE ANALYSIS
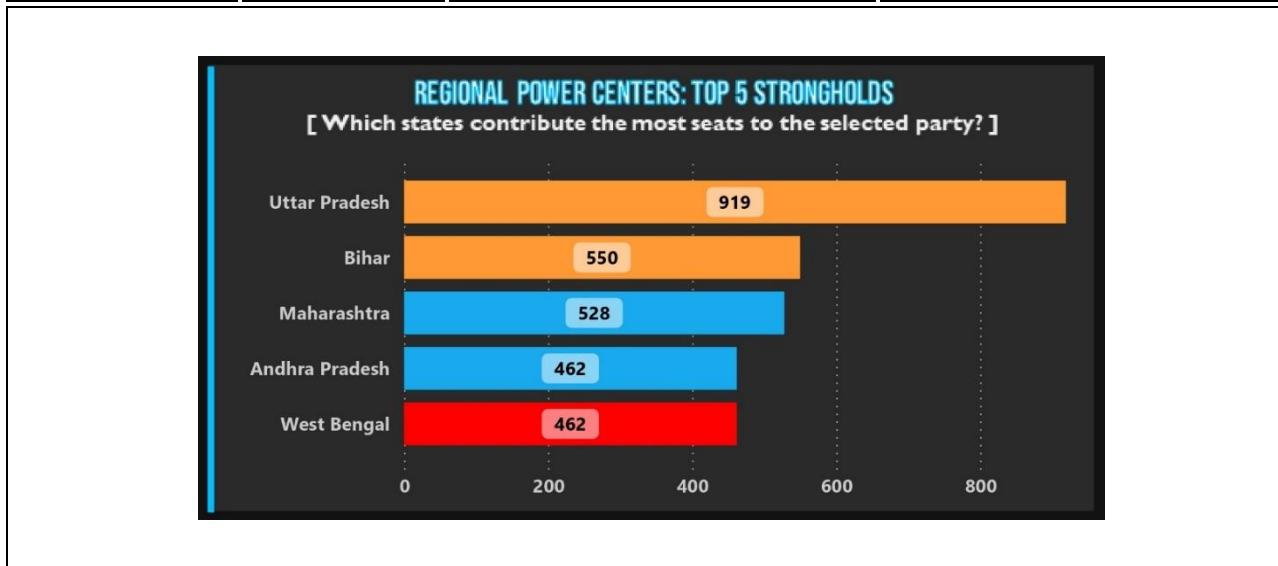
❖ **Theme:** "The ROI of Politics"

❖ **Strategic Purpose:** To move beyond simple seat counts and evaluate the "efficiency" of political campaigns, helping analysts distinguish between mass popularity and actual electoral convertibility.

Visual Component	Type	Configuration & Logic	Strategic Rationale
Global Controller	Slicer	Fields: Dim_Year, Party_Master[Party_Name]	Enables deep-dive analysis into specific political entities, filtering out noise from hundreds of smaller parties.

Efficiency Matrix	Scatter Plot	X-Axis: Vote Share % Y-Axis: Strike Rate % Values: Party Name	Visualizes the "Conversion Rate" of votes to seats. It exposes parties that may have high popularity (Vote Share) but fail to win seats due to poor vote concentration.

Performance Scorecard	Table	Columns: Flag, Party Name, Total Seats Won, Strike Rate %, Vote Share %, Avg Win Margin.	Provides a "Hard Data" lookup for journalists who need precise numbers (e.g., exact Strike Rate percentages) rather than just visual trends.

NATIONAL PARTY PERFORMANCE SCORECARD
[Detailed ranking of Top Parties by Seats, Vote Share, and Strike Rate]

| Flag	Party Name	Total Seats Won ▾	Strike Rate %	Vote Share %	Avg Win Margin
	Indian National Congress	2155	40.7%	30.58%	77.37K
	Bharatiya Janata Party	1266	37.8%	19.01%	92.82K
	Others	726	5.8%	11.81%	101.23K
	Communist Party of India - Marxist	314	40.8%	5.30%	94.32K
	Janata Dal	254	26.9%	3.44%	91.92K

Regional Power Centers	Bar Chart	Axis: State Name Values: Total Seats Won Filter: Top 5 States by Seat Contribution.	Identifies the geographic "Base" of a party. It answers whether a party is a true national force or heavily reliant on a few specific states.

REGIONAL POWER CENTERS: TOP 5 STRONGHOLDS
[Which states contribute the most seats to the selected party?]

| State	Seats
Uttar Pradesh	919
Bihar	550
Maharashtra	528
Andhra Pradesh	462
West Bengal	462

PAGE 4: DEMOGRAPHIC ANALYSIS: GENDER & TURNOUT
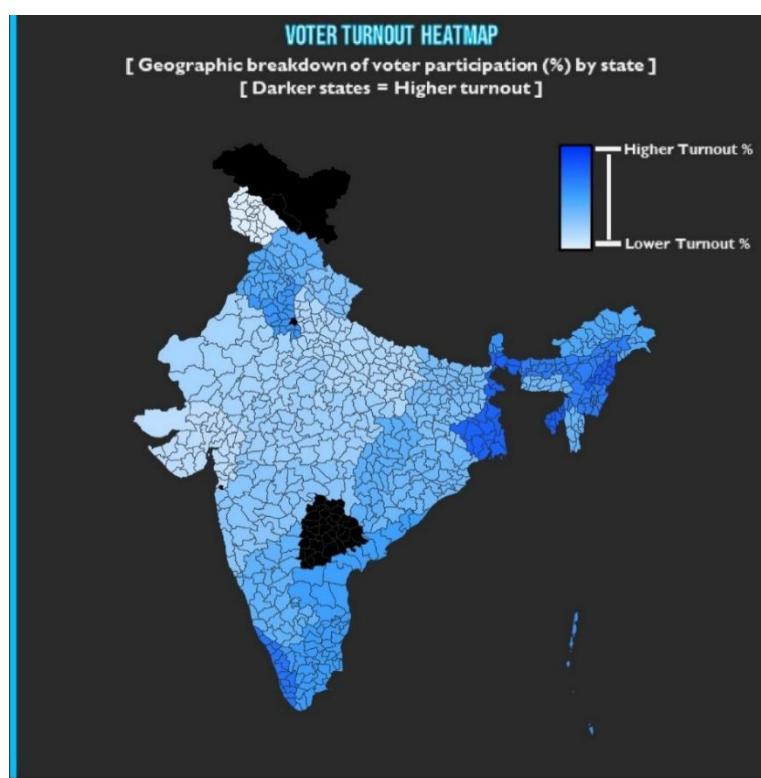
❖ **Theme:** "The Social Health Check"
❖ **Strategic Purpose:** To analyze the human element of the election, specifically correlating voter participation with outcomes and measuring the progress of gender inclusivity in Indian politics.

| Visual Component	Type	Configuration & Logic	Strategic Rationale
Context Controllers	Slicer	Fields: Dim_Year, Dim_State	Facilitates comparison between different states or time periods to track demographic shifts.
		YEAR — All ⌄ **STATE** — Maharashtra ⌄	
The Gender Gap	Clustered Bar	X-Axis: Year Y-Axis: Count Legend: Female Candidates Contested vs. Female Winners.	Reveals the "Drop-off Effect." It proves that while more women are contesting elections (Pink bars rising), the number of actual winners is not growing at the same linear pace.

THE GENDER GAP: CANDIDATES VS. WINNERS
[A historical comparison of Female Candidates fielded vs. Female Winners elected]
● Female Candidates Contested ● Female Candidate Winners

Year	Female Candidates Contested	Female Candidate Winners
1977	70	19
1980	143	28
1984	171	43
1989	198	29
1998	274	43
1999	284	49
1991	330	39
2004	355	45
2009	556	58
1996	599	40
2014	668	62

Female Success Rate	Gauge Chart	Value: [Female Success Rate %] Target: 33% (labeled "Nari Shakti Adhiniyam").	Sets a clear policy benchmark, highlighting the gap between current female representation and the proposed 33% legislative target.
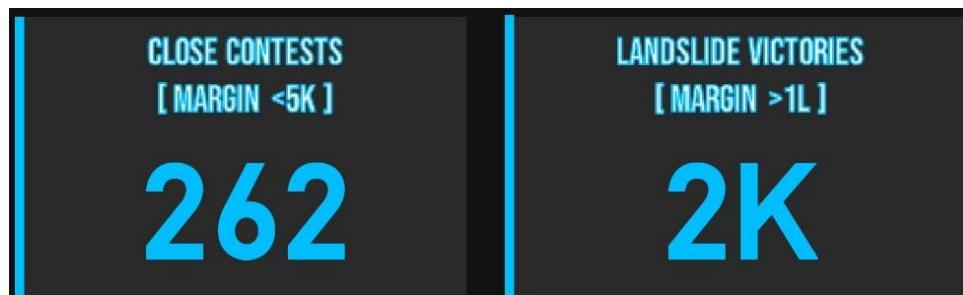
| Turnout Heatmap	Shape Map	Loc: Dim_State Saturation: [Nat Avg Turnout %] Gradient: Light Blue (Low) to Dark Blue (High)..	Instantly highlights regions with high civic engagement. Darker states indicate a more politically active electorate, often correlating with anti-incumbency waves.

PAGE 5: ELECTION WAR ROOM: MARGIN ANALYSIS

❖ **Theme:** "Critical Alerts & Volatility"
❖ **Strategic Purpose:** A high-stakes command center designed for live reporting. It isolates "At-Risk" seats to identify where the election hung by a thread versus where it was a one-sided wave.

Visual Component	Type	Configuration & Logic	Strategic Rationale
Global Controller	Slicer	Field: Dim_Year[year]	Acts as a "Time Machine," allowing stakeholders to switch context between election cycles

YEAR
All

Visual Component	Type	Configuration & Logic	Strategic Rationale
Volatility Alerts	KPI Cards	Metrics: [Close Contests (<5k)] vs [Landslide Victories (>1L)].	The Headline Stat: These big numbers instantly segregate "Safe" seats from "Nail Biters."

CLOSE CONTESTS
[MARGIN <5K]
262

LANDSLIDE VICTORIES
[MARGIN >1L]
2K
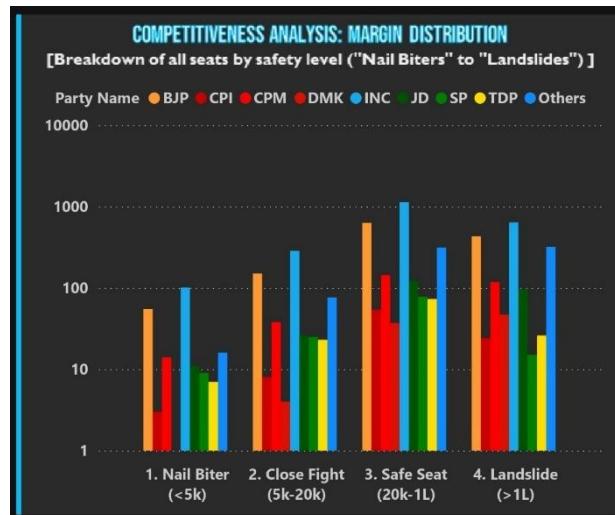
Competitiveness Analysis	Clustered Bar	Axis: Safety Category (Nail Biter, Close Fight, Safe Seat, Landslide) Values: Seat Count.	Profiles the "Intensity" of the mandate. It visually answers whether the election was a tight race or a decisive landslide.

The Nail Biters	Table	Filter: Margin < 5,000 votes. Formatting: Red background for Margin Votes column.	The Watchlist: Lists specific constituencies that are vulnerable to flipping. Essential for identifying swing seats..

The Fortresses	Table	Filter: Margin > 100,000 votes. Formatting: Green background for Margin Votes column.	Identifies "Unshakable" strongholds where the winner dominated the opposition completely.

THE FORTRESSES: MARGINS > 1L VOTES

List of decisive landslide victories where the opposition was crushed.

Flag	Party	Constituency	State	Margin Votes ▼
	CPM	Jhargram	West Bengal	1991032
	CPM	Burdwan	West Bengal	1902834
	BJP	Gandhinagar	Gujarat	1870694
	CPM	Tripura East	Tripura	1603199
	CPM	Arambagh	West Bengal	1594335
	CPM	Tripura West	Tripura	1493622
	NPF	Nagaland	Nagaland	1335265
	NCP	Baramati	Maharashtra	1128428
	INC	Nagaland	Nagaland	1103568
	JD	Hajipur	Bihar	1095992

Testing & Validation

Overview: Given the complexity of the "War Room" logic and "Efficiency" metrics, a rigorous testing phase was conducted to ensure system integrity.

A. Data Accuracy & Integrity (The "Zero-Loss" Check):

- **Technique:** Aggregate Reconciliation
- **Method:** We implemented a custom "Integrity Check" flag that calculates the absolute difference between the *Sum of Candidate Votes* and the *Total Valid Votes* reported in the source file.
- **Outcome:** Confirmed that row-level candidate data rolls up exactly to the constituency totals, ensuring zero data loss during the ETL process.

B. Logical & Scenario Testing:

- **Technique:** Edge-Case Validation
- **Method:** We manually filtered the dashboard for "Edge Cases"—specifically constituencies with the smallest possible winning margins (double-digit votes).
- **Outcome:** Verified that the conditional formatting logic (Red highlights) correctly triggers at the lower bounds and that the "Nail Biter" category accurately captures these specific anomalies.

C. Interactivity & Context Transition:

- **Technique:** Cross-Filtering Stress Test
- **Method:** We selected specific dimension attributes (e.g., a single Party or State) on the Master pages and navigated to the Detail pages.
- **Outcome:** Confirmed that filter contexts propagate correctly across the Galaxy Schema, ensuring that a "Party" selection correctly filters the "State Strongholds" chart without breaking visual relationships.

D. Temporal Logic Validation:

- **Technique:** Slicer Interaction Testing
- **Method:** Toggled the "Year" slicer across multiple historical election cycles (e.g., comparing 1984 vs. 2014).
- **Outcome:** Validated that complex time-intelligence measures (like historical trend lines) dynamically recalculate and display the correct subset of data for the selected period.

Following are the test cases which were executed during the completion of the dashboard pages –

Test ID	Visual Component	Action Performed	Expected Result	Status
D-01	Donut Chart	Select Year 2014.	BJP slice (Saffron) takes > 50% majority.	Pass
D-02	Line Chart	Clear filters.	Chart shows INC decline & BJP rise (1977-2014).	Pass
F-01	Slicer Logic	Select 2009.	Map updates to Blue (INC); All KPIs refresh.	Pass
F-02	Map Tooltips	Hover on the Map	Tooltip correctly shows data and logos of BJP, INC, Others.	Pass
D-03	KPI Cards	Select Uttar Pradesh.	"Assembly Seats" card shows 80 (Lok Sabha).	Pass
F-03	Map Zoom	Select Kerala.	Map auto-zooms to show only Kerala state shape.	Pass
F-04	Slicer Control	Try selecting 2 states.	System blocks selection (Single Select enforced).	Pass
D-04	Trend Line	Select West Bengal.	Line chart shows CPM dominance until 2009, then AITC.	Pass
D-05	Scatter Plot	Select BJP (2014).	Single dot in Top-Right quadrant (High Efficiency).	Pass

ID	Feature	Action	Expected Result	Status
D-06	Matrix Table	Check Columns.	Shows Party, Seats, Strike Rate, Vote Share %, Avg Win Margin	Pass
F-05	Regional Bar Chart	Select Shiv Sena.	Top bar is Maharashtra (Stronghold).	Pass
F-06	Filtering	Select 2004.	Chart filters out parties with 0 seats (Noise reduction).	Pass
D-07	Gender Gap Chart	Select Year 2014.	Female Candidates bar (Attempt) >> Winners bar.	Pass
D-08	Success Gauge	Check Gauge Value.	Shows approx 9-12% (Low female success rate).	Pass
F-07	Turnout Map	Hover over Nagaland.	Custom Tooltip appears showing the zoomed-in map.	Pass
F-08	State Filter	Select Kerala.	Gender chart updates to show only Kerala's history.	Pass
D-09	"Nail Biters" Table	Check Margin Column.	All displayed values must be strictly < 5,000 Votes.	Pass
D-10	"Fortresses" Table	Check Margin Column.	All displayed values must be strictly > 1,00,000 Votes.	Pass

F-09	Sorting Logic	Check "Nail Biters" List.	Rows are sorted Ascending (Smallest margin at top).	Pass
F-10	Margin Chart	Click "Landslide" Bar.	The "Fortresses" table filters to show those specific seats.	Pass

NEXT SPRINT GOALS (MILESTONE 4)

Focus: Deployment, Final Documentation & Submission.

- **Deployment Strategy:**

  - **Submission:** Submit the final .pbix file, dataset, and Python source code to the Infosys team via a **GitHub Link**.

- **Comprehensive Documentation:**

  - **Report Compilation:** Consolidate all previous reports (Milestone 1, 2, & 3) into a single master repository.

  - **Final Project Report:** Create a concluding document summarizing the full development lifecycle, technical challenges faced, and the solutions implemented.

- **Final Review & Presentation:**

  - **Sign-Off:** Conduct a final presentation rehearsal and prepare for the project Q&A session.