

# International Journal Research Publication Analysis

Page: 01-17

---

## ELECTRIC VEHICLE SALES AND MARKET TREND ANALYSIS USING PYTHON AND POWER BI

---

**\*Lavudi Naveen, Dr. Ramchandar (Guide)**

---

*Department of MCA (Master of Computer Applications) Chaitanya Bharathi Institute of  
Technology (CBIT), Gandipet, Hyderabad, Telangana, India.*

---

**Article Received: 05 September 2026**

**Article Revised: 25 September 2026**

**Published on: 15 October 2026**

**\*Corresponding Author: Lavudi Naveen**

Department of MCA (Master of Computer Applications) Chaitanya Bharathi  
Institute of Technology (CBIT), Gandipet, Hyderabad, Telangana, India.

DOI: <https://doi-org/101555/ijrpa.4632>

---

### ABSTRACT

Sales records for electric vehicles (EVs) are often examined in spreadsheets, or in forecasting studies that say little about how the data were prepared and queried. This paper documents a small, reproducible analytics workflow for EV sales and applies it to a demonstration dataset of 440 records covering ten Indian states, eleven models from five brands, and the years 2022 to 2025. The dataset is synthetic and ships with the project, so the findings describe how the workflow behaves and not how the real market behaves. Python and pandas clean the data and derive revenue, efficiency and value indicators. The cleaned table is stored in a SQLite database and explored with eight Structured Query Language (SQL) queries, a Streamlit dashboard and an Excel workbook present the results, and a Power BI dashboard is specified with Data Analysis Expressions (DAX) measures and visuals. In the demonstration data, yearly sales rise from 14,201 to 32,304 units between 2022 and 2025, while year-over-year growth eases from 36.6% to 25.3%. One brand holds 43.5% of units, sport utility vehicles (SUVs) account for 72.3%, and a least-squares line projects 38,090 units for 2026. The workflow gives students and analysts a transparent route from a raw sales table to interpretable market indicators, and it can be pointed at a real dataset that follows the same column layout.

**INDEX TERMS:** Electric Vehicles, Ev Sales, Market Trend Analysis, Python, Sql, Sqlite, Power Bi, Data Visualization.

## 1. INTRODUCTION

Electric vehicles (EVs) have gone from a niche product to a visible share of new car sales in a short time. The International Energy Agency (IEA) reports that global electric car sales grew by about 20% in 2025 to pass 20 million, so that roughly one in four new cars sold was electric [1]. India took part in this rise: the IEA's 2026 energy review notes that electric car sales in the country grew by more than 75% in 2025 [2].

Growth of this kind raises practical questions for manufacturers, dealers and planners. Which brands are gaining ground? Do all states move at the same pace? Do buyers favour a particular body type or price band? A sales table broken down by year, state and model can answer such questions only after it has been aggregated, and a few hundred rows are already hard to read as they stand. Summary indicators and charts make the patterns visible, provided the numbers behind them come from a preparation process that can be repeated and checked. Python suits the preparation stage. The pandas library offers labelled tabular structures for cleaning and aggregation [3], and NumPy supplies array-based numerical routines [4]. For storage, SQLite is a serverless SQL engine that keeps a complete database in one file [5], which is enough for a small project that still benefits from structured queries. For presentation, business intelligence (BI) tools such as Microsoft Power BI let users build interactive dashboards with filters and key performance indicators (KPIs) on top of tabular sources [6].

The work reported here began as a Master of Computer Applications (MCA) internship data analytics project on EV sales. Its purpose was practical: to bring Python, SQL, Excel and a BI tool into one workflow, and to test that workflow on a dataset small enough to inspect by hand. The research problem has two parts. How can a model-level EV sales table be cleaned, stored, queried and visualised so that others can reproduce the steps? And which market indicators can be read from such a table once the workflow is in place?

The contributions are modest but concrete. (i) A documented pipeline turns a raw comma-separated values (CSV) file into a SQLite database and an Excel report. (ii) Eight SQL queries and three views cover yearly, brand, state, model, vehicle-type, growth, efficiency and value questions. (iii) A Streamlit dashboard offers four filters, five KPI cards and five tabs, including a simple linear-trend projection. (iv) A Power BI specification lists five DAX measures, ten visual types and four slicers. (v) A worked analysis of the demonstration dataset is reported together with the limits of what that dataset can support.

The rest of the paper is organised as follows. Section 2 reviews related work, and Section 3 states the research gap and problem. Section 4 lists the objectives. Sections 5 to 8 describe the

dataset, the methodology, the architecture and the implementation. Results and dashboard views are presented in Sections 9 and 10, followed by discussion, limitations, future work and conclusions.

## 2. RELATED WORK

Studies of the Indian EV market often start from policy and market structure. Singh *et al.* reviewed the policies, strategies and technical considerations behind EV deployment in India and used a strengths, weaknesses, opportunities and challenges (SWOC) analysis to summarise the sector [7]. Gujarathi *et al.* described the main two- and four-wheeler players in the Indian market, discussed government initiatives and charging tariffs, and added a consumer case study [8]. Both offer useful background. They are review-style works, however, and neither publishes a data pipeline that a reader could rerun on new sales figures.

A larger group of papers treats EV sales as a time series to be forecast. Zhang *et al.* compared a univariate singular spectrum analysis model with a multivariate vector autoregressive model on monthly Chinese data, and found that market-related exogenous variables improved accuracy significantly [9]. Long and Liu fitted seasonal ARIMA (SARIMA) and exponential smoothing models to new energy vehicle sales for 2011–2019 in Python, chose SARIMA for a 2025 projection, and then applied LASSO (least absolute shrinkage and selection operator) regression, K-means clustering and multiple linear regression to look at factors related to sales [10]. Dhankhar *et al.* analysed monthly Indian EV sales from January 2018 to August 2023 with autoregressive integrated moving average (ARIMA) and exponential smoothing models, and reported that the ARIMA and Holt models pointed in opposite directions for the following two years [11].

Machine learning has been applied to the same problem. Afandizadeh *et al.* combined monthly sales and registration records of 357 vehicles in the United States (2014–2020) with web-crawled attributes, trained long short-term memory (LSTM) and ConvLSTM networks and a hybrid attention model, and studied which factors influenced the sales of each vehicle [12]. Mishra *et al.* compared LSTM, support vector regression, ARIMA, linear regression and a hybrid LSTM–gradient boosted regression trees (GBRT) model for Europe and India, and found the hybrid the most predictive [13]. At city level, Li tuned a bidirectional gated recurrent unit with the whale optimisation algorithm to forecast monthly new energy vehicle sales in Chinese cities, and observed that the cities with the highest sales were concentrated in four economic regions [14].

On the visualisation side, Gonçalves *et al.* built Power BI dashboards on a star-schema data warehouse for a public retail sales dataset, following the Vercellis methodology, and noted that the data did not come from a real organisation [6]. Their subject is sales in general, not EVs.

Taken together, these papers show a split. Forecasting studies concentrate on model accuracy for aggregated monthly series and usually say little about how the data were cleaned, stored or explored beforehand. The Indian market studies concentrate on policy. The BI study offers a dashboard method for a different domain. Table 1 summarises the comparison.

**Table 1: Comparison with Related Approaches.**

| Work      | Data and method                                                                            | Main output                                    |
|-----------|--------------------------------------------------------------------------------------------|------------------------------------------------|
| [9]       | China, monthly EV sales; singular spectrum analysis, vector autoregression                 | Forecast accuracy                              |
| [10]      | China, new energy vehicle sales 2011–2019; SARIMA, smoothing, LASSO, K-means (Python)      | Forecast and factors                           |
| [11]      | India, monthly EV sales 2018–2023; ARIMA, Holt                                             | Two-year forecast                              |
| [12]      | US, monthly sales of 357 vehicles; LSTM variants                                           | Vehicle-level forecast                         |
| [13]      | Europe and India; LSTM, support vector regression, ARIMA, LSTM–GBRT                        | Model comparison                               |
| [6]       | Retail sales (not EV); Power BI, star schema                                               | KPI dashboards                                 |
| This work | Model-level EV sales (synthetic); pandas, SQLite, Streamlit, Excel, Power BI specification | Documented workflow and descriptive indicators |

### 3. RESEARCH GAP AND PROBLEM STATEMENT

Among the sources examined for this review, none described a compact workflow that starts from a raw model-level EV sales table with brand, state, vehicle-type and specification attributes, passes it through documented cleaning and a SQL database, and ends in an interactive dashboard. The observation is limited to the literature located here, which was not a systematic search, so other work may exist. It is also a statement about scope and not about quality: the forecasting papers above answer a different question.

**Problem statement.** Given a table of EV sales by year, state, brand and model, design and document a reproducible workflow that cleans the data, stores it in a database, answers standard market questions with SQL and Python, and presents the answers interactively; then demonstrate the workflow on the available dataset and state plainly what that dataset can and cannot show.

## 4. OBJECTIVES

The work pursued six objectives. (i) Clean and validate a raw EV sales table and store it in a SQLite database. (ii) Derive revenue, efficiency and value indicators from the recorded attributes. (iii) Write SQL queries for yearly totals, brand, state and model rankings, year-over-year (YoY) growth and vehicle-type share. (iv) Build an interactive dashboard with filters, KPI cards and trend, share and ranking views. (v) Provide a simple trend projection, clearly labelled as a demonstration. (vi) Specify a Power BI dashboard (measures, visuals, slicers) that reads the exported workbook or the database.

## 5. DATASET AND DATA PREPROCESSING

### 5.1 DATASET DESCRIPTION

The project ships with data/ev\_sales.csv, a table of 440 records and ten attributes (Tables 2 and 3). Each record gives the units of one model sold in one state during one year, together with the technical and price attributes of that model. The README describes the file as synthetic demonstration data, and the project files record no original source, collection method or licence. The dataset should therefore be read as a test bed for the workflow, not as evidence about the Indian market. Prices are recorded in lakh of Indian rupees (INR). The table holds aggregate counts only and contains no personal information.

The table is a complete panel. Four years, ten states and eleven models give  $4 \times 10 \times 11 = 440$  combinations, and each occurs exactly once. The eleven models belong to five brands (BYD, Hyundai, MG, Mahindra and Tata) and three vehicle types: hatchback, multi-purpose vehicle (MPV) and sport utility vehicle (SUV). Battery capacity, range, price and charging time are fixed for each model across all years and states.

**Table 2: Dataset Characteristics.**

| Property        | Value                                                                                                                                        |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| Records         | 440 (before and after cleaning)                                                                                                              |
| Attributes      | 10 raw; 3 derived in the pipeline                                                                                                            |
| Period          | 2022–2025, annual                                                                                                                            |
| Geography       | 10 values in the State column (Delhi, Gujarat, Karnataka, Kerala, Maharashtra, Rajasthan, Tamil Nadu, Telangana, Uttar Pradesh, West Bengal) |
| Brands / models | 5 / 11                                                                                                                                       |
| Vehicle types   | 3 (hatchback, MPV, SUV)                                                                                                                      |
| Missing values  | 0                                                                                                                                            |

| Property       | Value                                       |
|----------------|---------------------------------------------|
| Duplicate rows | 0                                           |
| Source         | Not stated; README calls the data synthetic |

**Table 3: Attributes of the EV Sales Table.**

| Attribute              | Type  | Range / values | Role                 |
|------------------------|-------|----------------|----------------------|
| Year                   | int   | 2022–2025      | Time                 |
| State                  | text  | 10 values      | Geography            |
| Brand, Model           | text  | 5; 11          | Product              |
| Vehicle_Type           | text  | 3 values       | Segment              |
| Battery_Capacity_kWh   | float | 17.3–72.6      | Specification        |
| Range_KM               | int   | 230–631        | Specification        |
| Price_Lakh             | float | 8.0–46.0       | Price (INR lakh)     |
| Charging_Time_Hours    | float | 4.0–10.0       | Specification        |
| Units_Sold             | int   | 47–654         | Sales volume         |
| Revenue_Crore*         | float | derived        | Units $\times$ price |
| Efficiency_KM_per_kWh* | float | derived        | Range / battery      |
| Value_KM_per_Lakh*     | float | derived        | Range / price        |

\* Derived attribute, added by build\_database.py. 1 lakh =  $10^5$  and 1 crore =  $10^7$  INR.

## 5.2 DATA QUALITY

Quality checks on the raw file found no missing values, no fully duplicated rows, no value that failed numeric conversion, and no stray leading or trailing spaces in the four text columns. Units sold per record range from 47 to 654, with a median of 181. The project does not include an outlier analysis, so none is reported. The balanced panel and the fixed specifications per model are unusual for registration data collected in the field, which is consistent with the README's description of the file as synthetic. One consequence is that all variation in the table comes from the sales column: prices, ranges and battery sizes vary only between models, never within a model.

## 5.3 DATA PREPROCESSING

Preprocessing lives in load\_and\_clean() in build\_database.py and is summarised in Table 4. No rows were removed on this dataset, because the checks above found nothing to remove; the steps are nonetheless part of the pipeline and would act on a dirtier file. Three derived attributes are then added:

$$R_i = U_i \times P_i / 100 \quad (1)$$

$$E_i = D_i / B_i \quad (2)$$

$$V_i = D_i / P_i \quad (3)$$

where  $U$  is units sold,  $P$  the price in INR lakh,  $D$  the range in km and  $B$  the battery capacity in kWh. Revenue  $R$  is in INR crore and is computed from list price and units, so it is not an observed revenue figure.  $E$  (km/kWh) is the efficiency and  $V$  (km per INR lakh) the value indicator.

**Table 4: Data Preprocessing Operations.**

| Operation         | Implementation                                    | Effect here     |
|-------------------|---------------------------------------------------|-----------------|
| Type coercion     | pd.to_numeric, invalid values to NaN, six columns | None needed     |
| Duplicate removal | drop_duplicates()                                 | 0 rows          |
| Missing values    | dropna() on numeric columns                       | 0 rows          |
| Derived columns   | Eqs. (1)–(3), rounded to two decimals             | 3 columns added |
| Loading           | DataFrame.to_sql into SQLite, plus Excel export   | 440 rows stored |

## 6. PROPOSED METHODOLOGY

The workflow has seven stages: data collection, cleaning, transformation, database storage, SQL analysis, Python analysis and visualisation, and interpretation. The CSV file is the entry point, and the README recommends replacing it with a real public dataset before any formal submission. Cleaning and transformation follow Section 5. Storage is a single SQLite file whose main table, `ev_sales`, feeds every later stage.

SQL analysis answers fixed questions through eight reference queries. Python analysis happens inside the dashboard after the table is loaded: pandas groupby operations build each chart, and YoY growth is computed from yearly totals  $S_t$  as

$$G_t = (S_t - S_{t-1}) / S_{t-1} \times 100 \quad (4)$$

The dashboard also reports the fastest-growing state and brand by comparing the first and last selected year. For the projection, a straight line is fitted to the yearly totals by least squares (numpy.polyfit, degree one),

$$\hat{S}_t = at + b \quad (5)$$

and evaluated for the year after the last one selected, with negative results set to zero. The final stage is interpretation, in which rankings, shares and growth rates are read against the coverage of the data (Sections 9 and 11).

## 7. SYSTEM ARCHITECTURE

Fig. 1 shows how the parts connect. `build_database.py` reads the raw CSV once, cleans it, and writes two artefacts: a SQLite database and an Excel workbook. The database is the single source for the Streamlit application, which loads it with `SELECT * FROM ev_sales` and caches the result; if the database file is missing, the application falls back to cleaning the CSV itself. The reference queries can be run in any SQLite client against the same file. The workbook holds the cleaned records and four summary sheets, and it is the source named in the Power BI guide, which adds that Power BI could read the database through an ODBC (Open Database Connectivity) driver. The Power BI box is dashed because the dashboard is specified in the project guide and not delivered as a saved file (Section 8.4). Runtime dependencies are Streamlit, pandas, NumPy, PyArrow, Plotly and openpyxl.

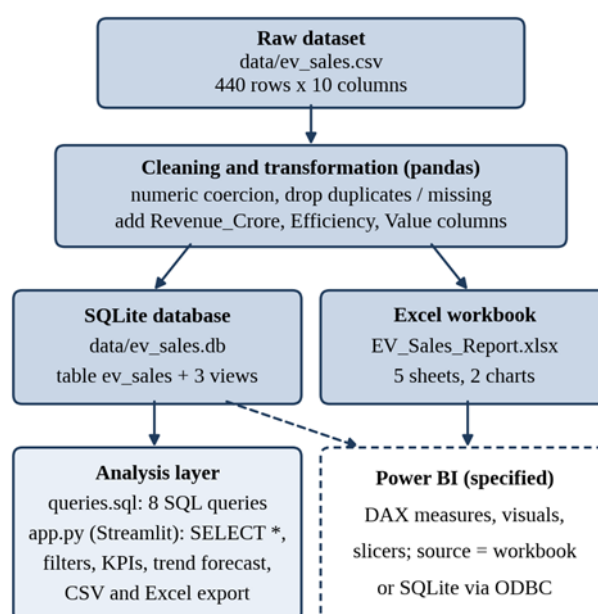


**Fig. 1. Architecture of the EV sales analytics workflow. The dashed box marks the Power BI layer, which is specified in the project guide but not delivered as a dashboard file.**

## 8. IMPLEMENTATION

### 8.1 PYTHON IMPLEMENTATION

Two scripts hold the code. `build_database.py` defines three functions: `load_and_clean` reads the CSV, coerces the numeric columns, removes duplicate and incomplete rows and adds the derived columns; `build_sqlite` writes the table and creates three views; `build_excel` writes a five-sheet workbook (Raw\_Data, Yearly\_Trend, Brand\_Summary, State\_Summary,

Top\_10\_Models) with a line chart of yearly units and a bar chart of brand units, using openpyxl. app.py is the Streamlit dashboard. Table 5 lists the technology stack. pandas performs the aggregations behind every chart, NumPy supplies the least-squares fit, Plotly draws the donut and projection charts, and the remaining charts use Streamlit's built-in chart elements.

**Table 5: Technology Stack.**

| Layer         | Technology           | Role in the project                |
|---------------|----------------------|------------------------------------|
| Language      | Python 3.10 or newer | Pipeline and dashboard logic       |
| Data handling | pandas, NumPy        | Cleaning, aggregation, trend fit   |
| Database      | SQLite (sqlite3)     | Table, views, reference queries    |
| Spreadsheet   | openpyxl             | Excel report, in-app download      |
| Dashboard     | Streamlit, Plotly    | Filters, KPI cards, charts         |
| BI tool       | Power BI             | Specified dashboard (DAX, visuals) |

## 8.2 DATABASE IMPLEMENTATION

The table `ev_sales` is created with `DataFrame.to_sql` and `if_exists="replace"`, so the pipeline can be rerun whenever the CSV changes. SQLite stores `Year`, `Range_KM` and `Units_Sold` as `INTEGER`, the decimal columns as `REAL` and the four text columns as `TEXT`. No primary key or index is defined. Three views are recreated on each run: `yearly_sales` (units and revenue by year), `brand_sales` (units and mean price by brand) and `state_sales` (units by state). The dashboard executes only the single `SELECT *` statement and does its own aggregation in pandas; the SQL file documents the equivalent queries for anyone who wants to work in a SQL client.

## 8.3 SQL ANALYSIS

The file `sql/queries.sql` holds eight queries. Queries 1 to 4 are grouped aggregations: yearly units and revenue, brand units with mean price, state units, and the ten best-selling models. Queries 5 and 6 average the derived columns per model to rank the five most efficient and the five best-value models. Query 7 computes YoY growth by joining the `yearly_sales` view to itself on consecutive years. Query 8 gives each vehicle type's share of units, using a scalar subquery for the total. All eight ran without error against the supplied database, and their output supplies the results in Section 9.

## 8.4 POWER BI IMPLEMENTATION

The guide in `powerbi/PowerBI_Setup_Guide.md` takes the Excel workbook as the data source (with the `Raw_Data` sheet as the main table) and suggests an ODBC connection to the SQLite

file as an alternative. It asks the analyst to confirm that numeric columns are typed as numbers in Power Query, and then defines five DAX measures: total units, total revenue, average price, average range and a YoY growth measure that compares the latest year in the filter context with the year before it. The suggested report page has ten visual specifications: KPI cards, a yearly line chart, brand and state bar charts, a top-ten model chart, two donut charts, a battery-versus-range scatter plot and two top-five tables, controlled by four slicers (Year, State, Brand, Vehicle\_Type) that mirror the Streamlit filters. At the time of writing the repository contains the guide but no saved dashboard file, screenshot or PDF export. The Power BI layer is therefore reported as a specification, and the interactive views that were actually exercised are those of the Streamlit dashboard.

## 9. EXPERIMENTAL RESULTS AND ANALYSIS

The results below come from running the project's SQL queries on data/ev\_sales.db, with pandas used to cross-check them and to compute a few supplementary quantities. Quantities that the delivered code does not produce (compound growth, correlations, per-model means and revenue growth) are marked as supplementary.

### 9.1 YEARLY SALES AND GROWTH

Table 6 reports total units and revenue by year (Query 1) and YoY growth (Query 7, Eq. 4). Fig. 2 plots the same totals. Units rose from 14,201 in 2022 to 32,304 in 2025, an increase of 127.5% and a compound annual rate of 31.5% (supplementary). Revenue rose from INR 2,673.56 crore to INR 6,125.61 crore. YoY growth in units fell from 36.6% to 32.8% and then to 25.3%, while the absolute yearly increment grew from 5,202 to 6,370 and 6,531 units.

**Table 6: Yearly Sales, Revenue and Growth.**

| Year | Units  | Revenue (INR crore) | YoY units (%) | Added units |
|------|--------|---------------------|---------------|-------------|
| 2022 | 14,201 | 2,673.56            | —             | —           |
| 2023 | 19,403 | 3,661.16            | 36.6          | 5,202       |
| 2024 | 25,773 | 4,917.45            | 32.8          | 6,370       |
| 2025 | 32,304 | 6,125.61            | 25.3          | 6,531       |

The reading is that the series is close to linear. A shrinking percentage on a growing base is what a roughly constant yearly increment looks like. This matters for dashboard design: a KPI card that shows only the latest YoY figure would suggest a slowdown, while the increments show that each year still adds more units than the last. Revenue growth followed the same pattern (36.9%, 34.3% and 24.6%, supplementary). Across the four years the

dashboard's KPI cards show 91,681 units, INR 17,377.78 crore of revenue, a mean price of INR 21.68 lakh and a mean range of 452 km.

The mean price deserves a note. The card averages over records, so each model counts equally. Weighting by units gives INR 18.95 lakh (supplementary), lower because the cheaper models sell in larger numbers. Which of the two is appropriate depends on the question being asked, and the dashboard should say which one it shows.

## 9.2 BRAND AND MODEL PERFORMANCE

Brand units come from Query 2, model units from Query 4, and unit share is brand units divided by the total (Table 7). Tata leads with 43.5% of units, and its three models are the three best sellers: Tiago EV (13,892), Nexon EV (13,603) and Punch EV (12,391) (Fig. 3b). Mahindra (17.8%), MG (15.7%), Hyundai (13.3%) and BYD (9.7%) follow. Tata also has three models where every other brand has two, so per-model averages give a fairer picture: 13,295 units per Tata model against 8,151 for Mahindra and 4,444 for BYD (supplementary). Brand shares hardly move between years; Tata stays between 42.9% and 44.1%.

**Table 7: Brand Summary, 2022–2025.**

| Brand    | Units  | Share (%) | Mean price (lakh) | Growth (%) |
|----------|--------|-----------|-------------------|------------|
| Tata     | 39,886 | 43.5      | 12.67             | 128.0      |
| Mahindra | 16,302 | 17.8      | 18.75             | 127.6      |
| MG       | 14,370 | 15.7      | 15.00             | 116.7      |
| Hyundai  | 12,236 | 13.3      | 35.00             | 131.6      |
| BYD      | 8,887  | 9.7       | 31.50             | 137.3      |

Growth in Table 7 is the change from 2022 to 2025 (supplementary; the dashboard computes the same first-versus-last-year figure for its fastest-growing-brand callout, which would name BYD). BYD grows fastest but from the smallest base. Price changes the ranking once revenue replaces units: Tata's revenue share is 29.0% against a unit share of 43.5%, while Hyundai, with a mean price of INR 35.00 lakh, earns 24.8% of revenue from 13.3% of units (supplementary).

## 9.3 STATE-WISE TRENDS

Query 3 (Fig. 3a) ranks Maharashtra first with 11,785 units (12.9%), followed by Karnataka (11,509; 12.6%) and Delhi (10,387; 11.3%). Rajasthan is last with 6,709 units (7.3%). The three leading states hold 36.7% of all units, and the ratio between the largest and smallest state is only 1.76. The order is almost fixed over time: Maharashtra, Karnataka and Delhi occupy the top three places and Rajasthan the last one in every year, and only mid-table

neighbours trade places (Gujarat and Uttar Pradesh in 2024; Kerala and West Bengal in 2023 and 2025). Growth from 2022 to 2025 runs from 115.2% in Rajasthan to 139.3% in Kerala, which is the state the fastest-growing callout would report (supplementary).

#### **9.4 VEHICLE CATEGORIES**

Query 8 (Fig. 4) gives SUVs 66,279 units (72.3%), hatchbacks 20,971 (22.9%) and MPVs 4,431 (4.8%). The SUV share stays between 72.1% and 72.7% in every year. The share is influenced by portfolio composition, though: eight of the eleven models are SUVs, two are hatchbacks and one is an MPV. Averaged over model-state-year records, a hatchback sells 262 units, an SUV 207 and an MPV 111 (supplementary), so the dominance of SUVs reflects how many SUV models the dataset contains more than how strongly each one sells.

#### **9.5 EFFICIENCY, VALUE AND SPECIFICATIONS**

By Eq. (2), the most efficient models (Query 5) are Comet EV (13.29 km/kWh), Tiago EV (13.12), Punch EV (12.03), XUV400 EV (11.57) and Kona Electric (11.53); the lowest is e6 at 6.75 (supplementary). By Eq. (3), the best-value models (Query 6) are Punch EV (33.68 km per INR lakh), Tiago EV (31.50), Nexon EV (30.00), Comet EV (28.75) and BE 6 (26.75). Both rankings depend only on model specifications, which are constant across years and states, so they do not change with the dashboard filters.

Across the eleven models, battery capacity and range are strongly correlated (Pearson  $r = 0.892$ , supplementary; Fig. 5). Price and total units are negatively correlated ( $r = -0.646$ , Spearman  $-0.727$ , supplementary), which suggests that cheaper models sell more in this dataset. With eleven points and synthetic data this is a description of the table and not a finding about buyers.

#### **9.6 TREND PROJECTION**

For the full dataset, the least-squares line of Eq. (5) has a slope of 6,068 units per year and gives 38,090 units for 2026, about 17.9% above 2025. The line passes close to the four observed totals ( $R^2 = 0.997$ , supplementary), but four points say very little about predictive quality. The dashboard itself calls the result a demonstration trendline, and it is treated that way here.

### **10. VISUALIZATION AND DASHBOARD ANALYSIS**

#### **10.1 STREAMLIT DASHBOARD**

The dashboard has a sidebar with four multi-select filters (Year, State, Brand, Vehicle Type) and a row of five KPI cards, followed by five tabs. Every chart is computed from the filtered data frame, so a single change in the sidebar updates all of them together. The lookup tab

searches the full dataset, independent of the filters, for a model or brand name and returns its units, revenue, mean price, growth and a year-by-year line. Figs. 2 to 5 were regenerated from the database with the same aggregations as the dashboard; they are not screenshots of the running interface.

**Trend line (Fig. 2).** Year is on the horizontal axis and total units on the vertical axis, since the question is how a quantity changes in time. The line rises almost steadily, and the YoY labels make the falling percentages visible next to the rising totals.

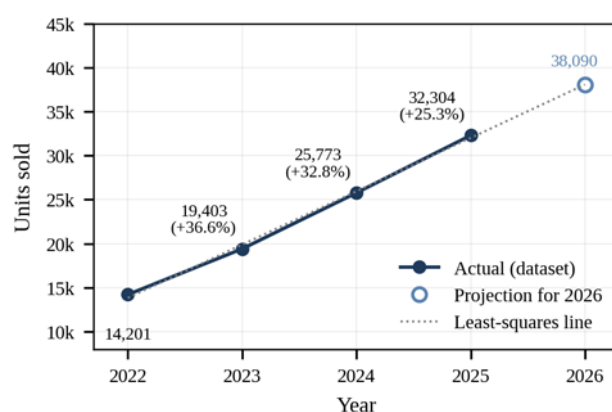


**Fig. 2.** Yearly units sold with YoY growth and the least-squares projection for 2026.

**Ranked bars (Fig. 3).** Horizontal bars suit categorical names and ordered comparison. Panel (a) shows a modest spread between states, and panel (b) shows a sharp drop after the three Tata models, from 12,391 units to 8,260 for the fourth.

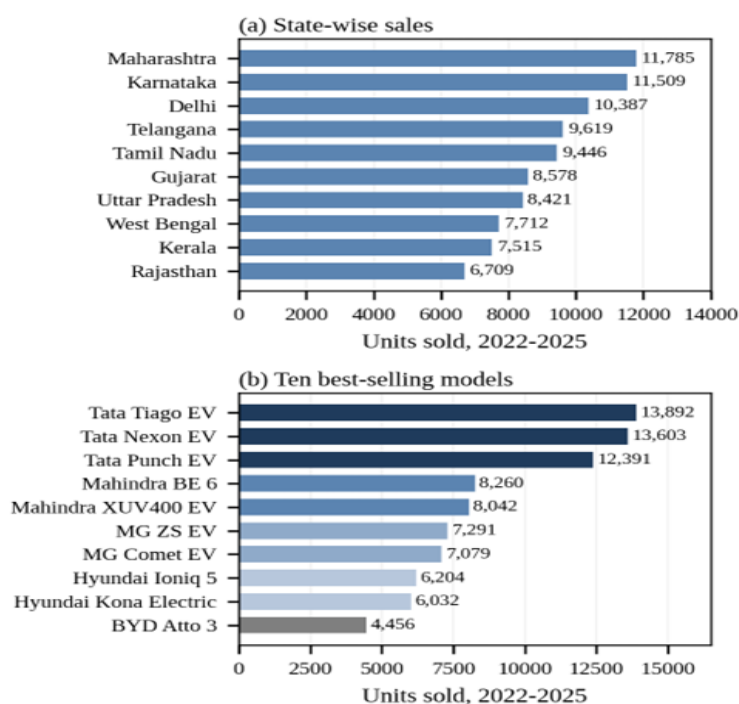


**Fig. 3.** (a) Units sold by state and (b) the ten best-selling models, 2022–2025.

**Share charts (Fig. 4).** Donut charts show the part-to-whole split for the five brands and the three vehicle types. They make the size of the Tata and SUV slices obvious, but they are poor at showing small differences, so Table 7 carries the exact values.

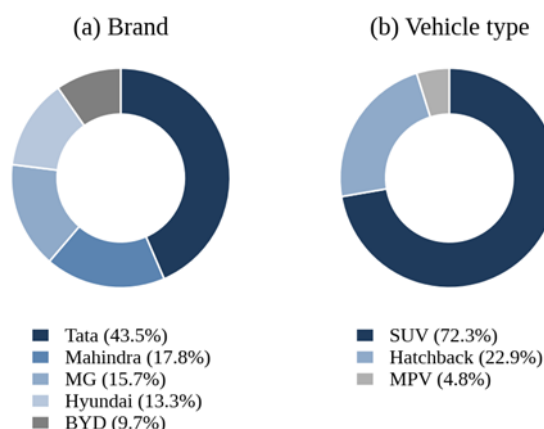


**Fig. 4. Share of units by (a) brand and (b) vehicle type, 2022–2025.**

**Scatter plot (Fig. 5).** Battery capacity against range compares two continuous variables. The points lie near a straight line, with e6 standing out: it offers 484 km from a 71.7 kWh battery, the lowest efficiency in the table. Because specifications are fixed per model, the plot contains eleven distinct points however many records are selected.

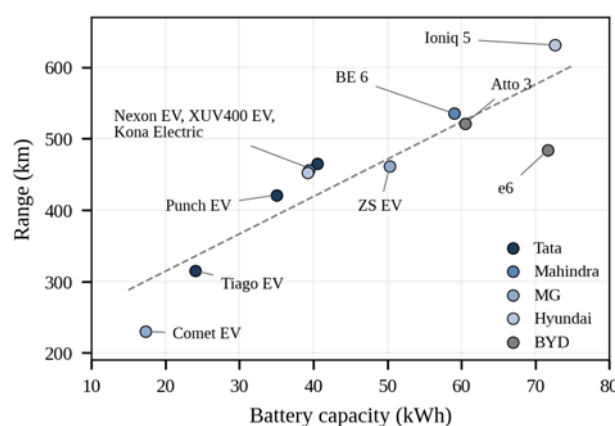


**Fig. 5. Battery capacity versus range for the eleven models; the dashed line is a least-squares fit.**

## 10.2 POWER BI VIEWS

The specified Power BI page repeats these views with a BI tool: cards for the five KPIs, a line chart for the yearly trend, bar charts for brand, state and model, donut charts for brand and vehicle-type share, a scatter plot, two top-five tables and the four slicers. The reasoning

for each visual type is the same as above. Since no dashboard file is available, no Power BI screenshot is shown, and no claim is made about how the visuals render.

## 11. DISCUSSION

**What was observed.** Growth is nearly the same everywhere in the dataset: 116.7% to 137.3% across brands, 115.2% to 139.3% across states and 126.3% to 130.8% across vehicle types between 2022 and 2025. Brand, state and vehicle-type shares barely change over time.

**Interpretation.** Such uniformity suggests that the data leave little room for segment-specific dynamics. Whether real records would behave the same way is an empirical question, and it is one the workflow can answer once real data are loaded. The finding that the dataset is uniform is itself a useful outcome of the analysis, because it reveals what kind of data was supplied.

**Relation to earlier work.** The projection here is a straight line on four annual points. The studies in Section 2 fit models to monthly series covering several years [9]–[11], and one of them reports that two standard models can disagree on the direction of the forecast [11]. A four-point line offers no basis for that kind of comparison and should not be read as a forecast in the same sense. What the workflow contributes is traceability: every number in Section 9 can be reproduced from the CSV with the supplied scripts. On geography, Li found strong concentration of sales in a few economic regions of China [14]. The state shares here span only 7.3% to 12.9%, but the units are different (states against cities) and the data are synthetic, so the two results cannot be compared.

**External context.** The IEA reports growth of more than 75% in Indian electric car sales for 2025 [2]. The dataset's 25.3% YoY growth for the same year is far lower, and it would be a mistake to read the difference as market evidence: the dataset covers eleven models and is synthetic. The gap is a reminder that KPIs on a dashboard describe only the data behind them.

## 12. LIMITATIONS

The main limitation is the data. The dataset is synthetic, no source is recorded, and results such as brand shares or state rankings must not be quoted as facts about the EV market. The panel is perfectly balanced and each model's specification is fixed, which removes much of the variation that real data would contain. Sales are annual, so there is no seasonality or monthly detail, and four time points support only very simple trend statements. Only passenger cars are covered, and revenue is derived from list price and units, not observed. No outlier analysis, significance testing or causal inference was carried out, and the projection

has no error measure or hold-out validation. The Power BI dashboard exists only as a specification. Finally, the SQLite file and Excel workbook are static: they change only when the pipeline is rerun, and no performance testing was done on larger tables.

### 13. FUTURE WORK

The following items are proposed and have not been implemented. The first priority is to replace the synthetic file with real registration or sales data in the same column layout; the IEA names India's Vahan database among the sources behind its EV statistics [1], and data of that kind could also bring monthly granularity. With monthly data, ARIMA-type and machine learning models could be compared against the linear baseline using hold-out errors, as done in [9]–[13]. Further steps include price and incentive analysis, charging-infrastructure data, consumer-behaviour variables, automatic dashboard refresh, cloud deployment, and a completed Power BI file with published screenshots.

### 14. CONCLUSION

This paper described a compact workflow for EV sales analysis in which pandas cleans a raw CSV file, SQLite stores the result, SQL queries and a Streamlit dashboard answer standard market questions, an Excel workbook carries the summaries, and a Power BI dashboard is specified on top. On the supplied synthetic dataset of 440 records, the workflow reported yearly sales rising from 14,201 to 32,304 units, YoY growth easing from 36.6% to 25.3% while yearly increments grew, a 43.5% unit share for Tata, a 72.3% share for SUVs, and a linear projection of 38,090 units for 2026. These values describe the demonstration data only. The practical value lies in the structure: the steps are short, documented and repeatable, and they separate cleaning, storage, querying and presentation so that each can be checked. Applying the same workflow to real data, adding monthly records and validating the projection are the natural next steps.

### REFERENCES

1. IEA, *Global EV Outlook 2026*. Paris, France: International Energy Agency, 2026. [Online]. Available: <https://www.iea.org/reports/global-ev-outlook-2026>
2. IEA, "Technology: Electric vehicles," in *Global Energy Review 2026*. Paris, France: International Energy Agency, 2026. [Online]. Available: <https://www.iea.org/reports/global-energy-review-2026/technology-electric-vehicles>
3. W. McKinney, "Data structures for statistical computing in Python," in *Proc. 9th Python in Science Conf. (SciPy)*, 2010, pp. 56–61, doi: 10.25080/Majora-92bf1922-00a.

4. C. R. Harris *et al.*, “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, pp. 357–362, 2020, doi: 10.1038/s41586-020-2649-2.
5. SQLite, “About SQLite.” [Online]. Available: <https://www.sqlite.org/about.html> (accessed Sep. 19, 2026).
6. C. T. Gonçalves, M. J. A. Gonçalves, and M. I. Campante, “Developing integrated performance dashboards visualisations using Power BI as a platform,” *Information*, vol. 14, no. 11, Art. no. 614, 2023, doi: 10.3390/info14110614.
7. V. Singh, V. Singh, and S. Vaibhav, “Analysis of electric vehicle trends, development and policies in India,” *Case Stud. Transp. Policy*, vol. 9, no. 3, pp. 1180–1197, 2021, doi: 10.1016/j.cstp.2021.06.006.
8. P. K. Gujarathi, V. A. Shah, and M. M. Lokhande, “Electric vehicles in India: Market analysis with consumer perspective, policies and issues,” *J. Green Eng.*, vol. 8, no. 1, pp. 17–36, 2018, doi: 10.13052/jge1904-4720.813.
9. Y. Zhang, M. Zhong, N. Geng, and Y. Jiang, “Forecasting electric vehicles sales with univariate and multivariate time series models: The case of China,” *PLoS ONE*, vol. 12, no. 5, Art. no. e0176729, 2017, doi: 10.1371/journal.pone.0176729.
10. S. Long and Q. Liu, “Research on new energy vehicle sales forecast and product optimization based on data mining,” in *Proc. 2nd Int. Conf. Electron., Commun. Inf. Technol. (CECIT)*, 2021, pp. 1019–1024.
11. S. Dhankhar, N. Dhankhar, V. Sandhu, and S. Mehla, “Forecasting electric vehicle sales with ARIMA and exponential smoothing method: The case of India,” *Transp. Dev. Econ.*, vol. 10, Art. no. 32, 2024, doi: 10.1007/s40890-024-00216-y.
12. S. Afandizadeh, D. Sharifi, N. Kalantari, and H. Mirzahosseini, “Using machine learning methods to predict electric vehicles penetration in the automotive market,” *Sci. Rep.*, vol. 13, Art. no. 8345, 2023, doi: 10.1038/s41598-023-35366-3.
13. R. D. Mishra *et al.*, “Hybrid LSTM-GBRT based machine learning technique implementation for electric vehicle sales prediction analysis,” *Int. J. Sustain. Eng.*, vol. 17, no. 1, pp. 843–858, 2024, doi: 10.1080/19397038.2024.2411284.
14. X. Li, “A study on monthly sales forecasting of new energy vehicles in urban areas using the WOA-BiGRU model,” *PLoS ONE*, vol. 20, no. 4, Art. no. e0320962, 2025, doi: 10.1371/journal.pone.0320962.