

Project Documentation

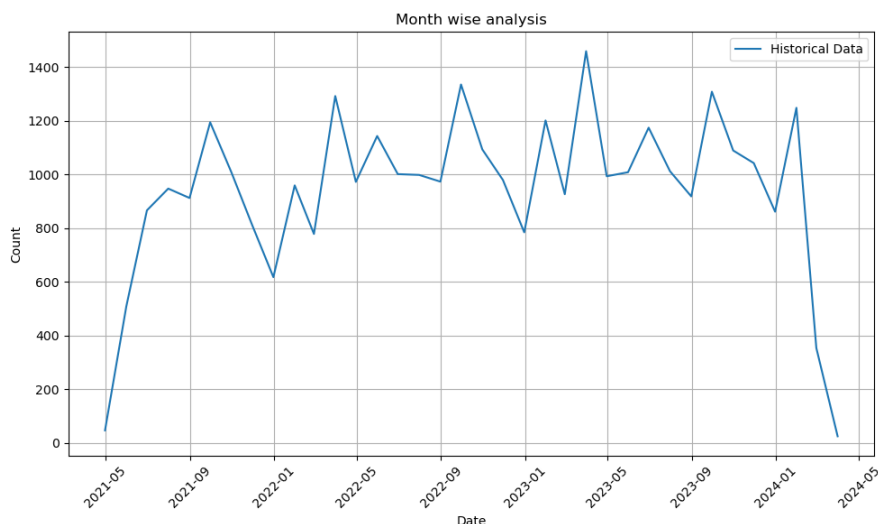
Three-Month Sales Forecasting with Facebook Prophet

Aman Khan | Data Scientists | amancsds@gmail.com | 29/10/2-24

- Stage 1: - Data Collection and Initial Filtering
- Stage 2: - Problem Understanding and Project Structuring
- Stage 3: - Data Loading into MySQL
- Stage 4: - Exploratory Data Analysis (EDA) and Preprocessing
- Stage 5: - Processed Data Storage
- Stage 6: - Model Selection and Testing
- Stage 7: - Final Model Selection with Prophet
- Stage 8: - Model Tuning and Optimization
- Stage 9: - Forecast Results and Post-Processing
- Stage 10: - Results Storage in MySQL
- Stage 11: - PowerBI Dashboard Creation

Data Collection and Initial Filtering

I received the raw data via email, I downloaded and reviewed it in Excel. I noticed some corrupted records and that the data started on April 21, 2021, and ended on March 1, 2024. The first and last months were incomplete, so I removed these partial records to improve accuracy. After cleaning, I uploaded the refined data to MySQL for easier access and analysis.



Problem Understanding and Project Structuring

With a clean dataset ready, I focused on understanding the core objective: to forecast vehicle sales over the next three months with breakdowns by brand and model. This involved defining key project steps, including data exploration, model selection, tuning, and visualization, to ensure the project structure aligned with accurate and actionable insights.

Data Upload and Cleanup

After cleaning in Excel, I uploaded the refined data into MySQL for structured storage and easy access. This allowed for smooth data handling and setup for further processing.

Exploratory Data Analysis (EDA)

In EDA, I analysed the dataset thoroughly, extracting essential features and identifying patterns. I found several issues in the car brand and model columns, with many unique or misspelled values. Using data preprocessing and mining techniques, I corrected these inconsistencies to ensure data accuracy.

Data Storage

Once processed, the cleaned data was stored in a dedicated "Processed Data" folder for seamless retrieval during modeling.

Model Selection

I applied various time-series forecasting models, including ARIMA, auto Arima, PMDARIMA, and Prophet. Each model was evaluated to determine which provided the most accurate results for future sales forecasting.

Model Training and Testing

Using a train-test split technique, I divided the data to evaluate model performance on unseen data. Through this approach, I identified that the Prophet model consistently provided the highest accuracy for forecasting, making it the best fit for the project.



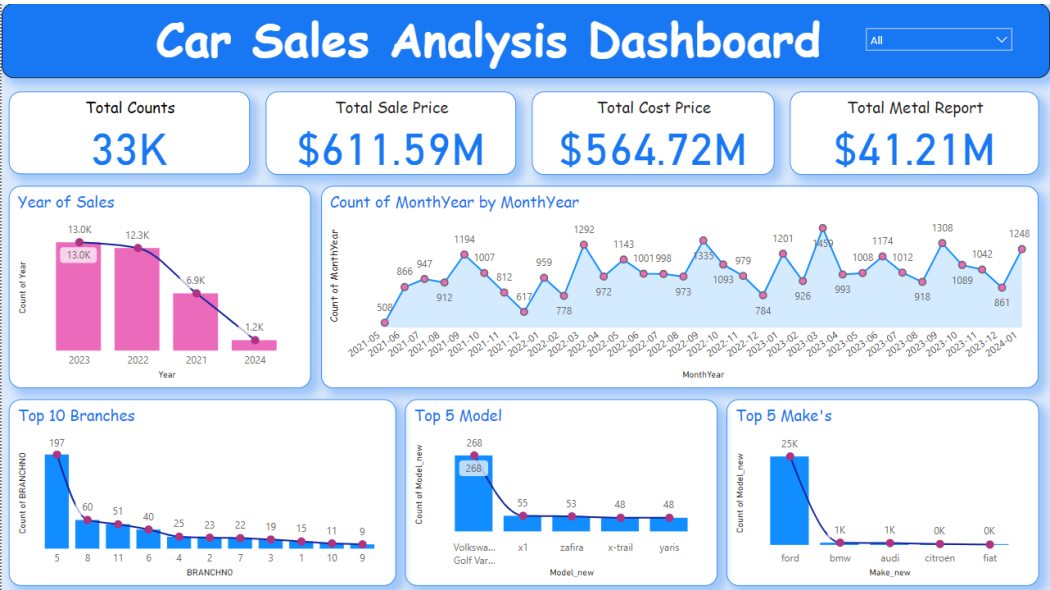
Model Tuning and Optimization

To further improve forecast accuracy, I fine-tuned the Prophet model by adjusting parameters to better capture the data's seasonality and trends. This optimization step helped maximize the model's predictive performance.

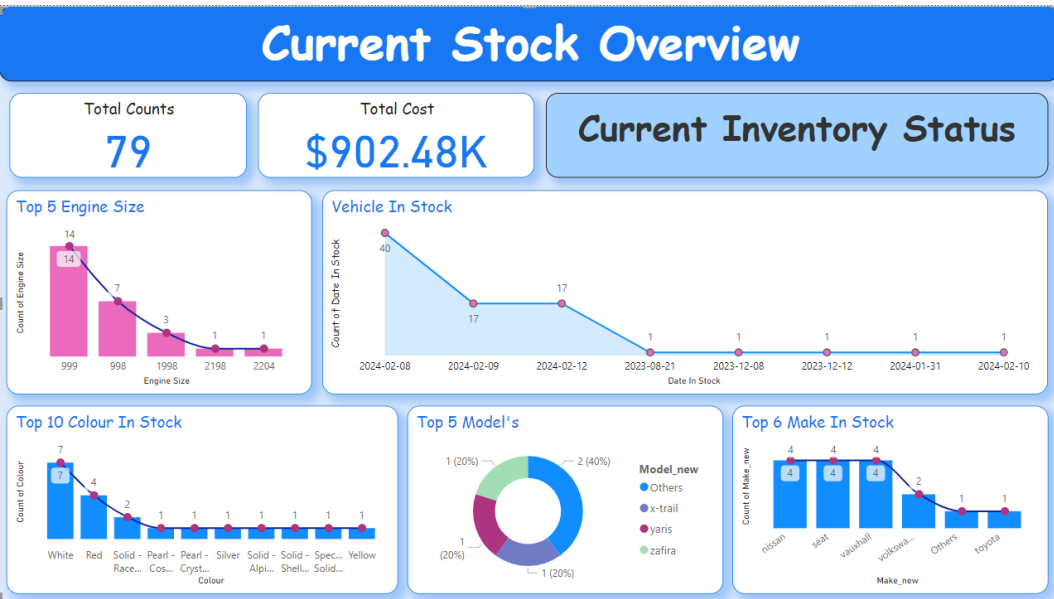
Results Storage and Preparation for Visualization

The final model results were saved in MySQL, ensuring they were easily accessible for visualization. Adjustments were made to format the output for PowerBI dashboards, allowing clear and actionable insights.

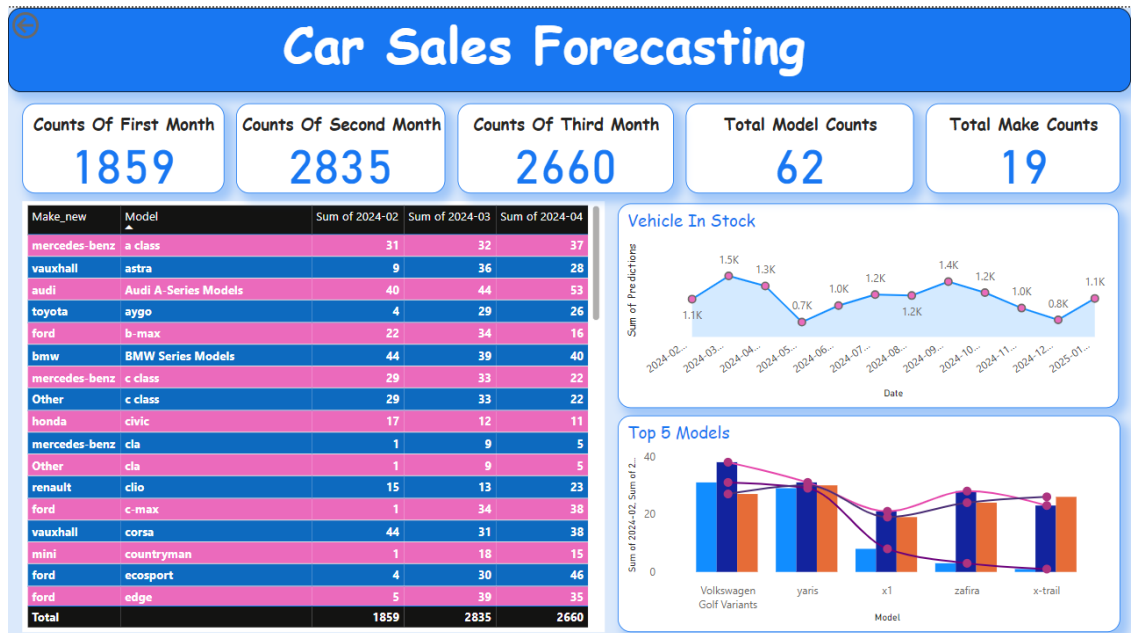
Sales Data Analysis – A breakdown of historical sales trends.



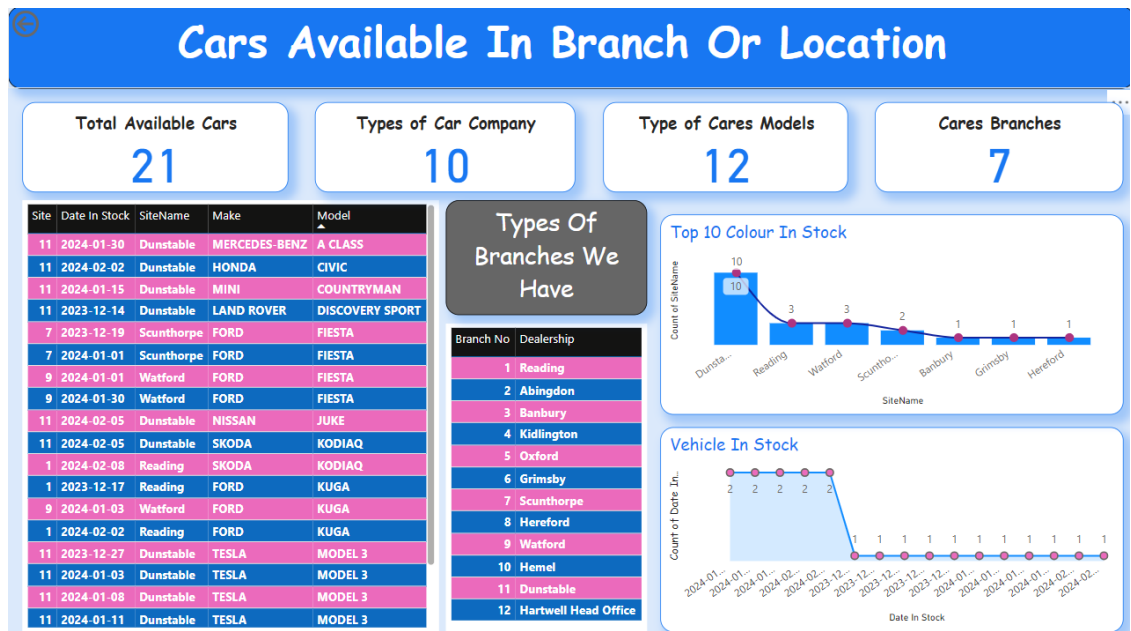
Current Sales Overview – Provides a snapshot of the latest sales figures.



Three-Month Forecast – Presents the forecasted sales counts for the next three months.



Car Location by Site/Branch – Maps car locations across different sites or branches for operational insights.



Project Structure

```
Project_Root/
├── data/
│   ├── raw_data/           # Original dataset files
│   ├── processed_data/     # Data after cleaning/preprocessing
│   ├── mysql_extraction.sql # SQL scripts for data extraction
│   └── mysql_connection.py  # Python script to connect to MySQL and extract data
├── notebooks/
│   ├── EDA.ipynb           # Exploratory Data Analysis notebook
│   ├── data_preprocessing.ipynb # Data preprocessing notebook
│   ├── model_building.ipynb # Model training and initial analysis
│   └── model_tuning.ipynb   # Model tuning and optimization
├── src/
│   ├── data_preprocessing.py # Data cleaning, encoding, and transformation script
│   ├── feature_engineering.py # Feature engineering scripts
│   ├── model_training.py     # Scripts for model training
│   ├── model_tuning.py       # Hyperparameter tuning
│   └── final_model.py        # Script for the final trained model
├── requirements.txt         # List of project dependencies
├── config/
│   ├── config.yaml         # Configurations for database, models, etc.
│   └── db_config.json      # Database connection settings
├── scripts/
│   ├── predictions.py       # Generating predictions for future months
│   └── forecast_visualization.py # Visualization of forecasted results
├── reports/
│   ├── PowerBI_reports.pbix # PowerBI report file or connection info
│   └── results.md           # Summary of results and insights
├── results/
│   ├── model_performance/   # Metrics and validation results
│   └── future_predictions.csv # Future forecast predictions for PowerBI
└── README.md               # Project overview and setup instructions
```