PRESENTATION

# MULTIDOMAIN ANALYTICS
# AI IN BUSINESS

**Course**: MIS 395 - ARTIFICIAL INTELLIGENCE FOR BUSINESS

**Lecturer:** Mr. Dang Thai Doan

**Quarter**: 4/2024-2025

**Presented by**: Team Huy - Thao - Tue

ABOUT TEAM

# TEAM MEMBER

**Bui Gia Tue**

*2132300511*

Is data fun? I don’t think so. Just try it yourself!

**Thai Gia Huy**

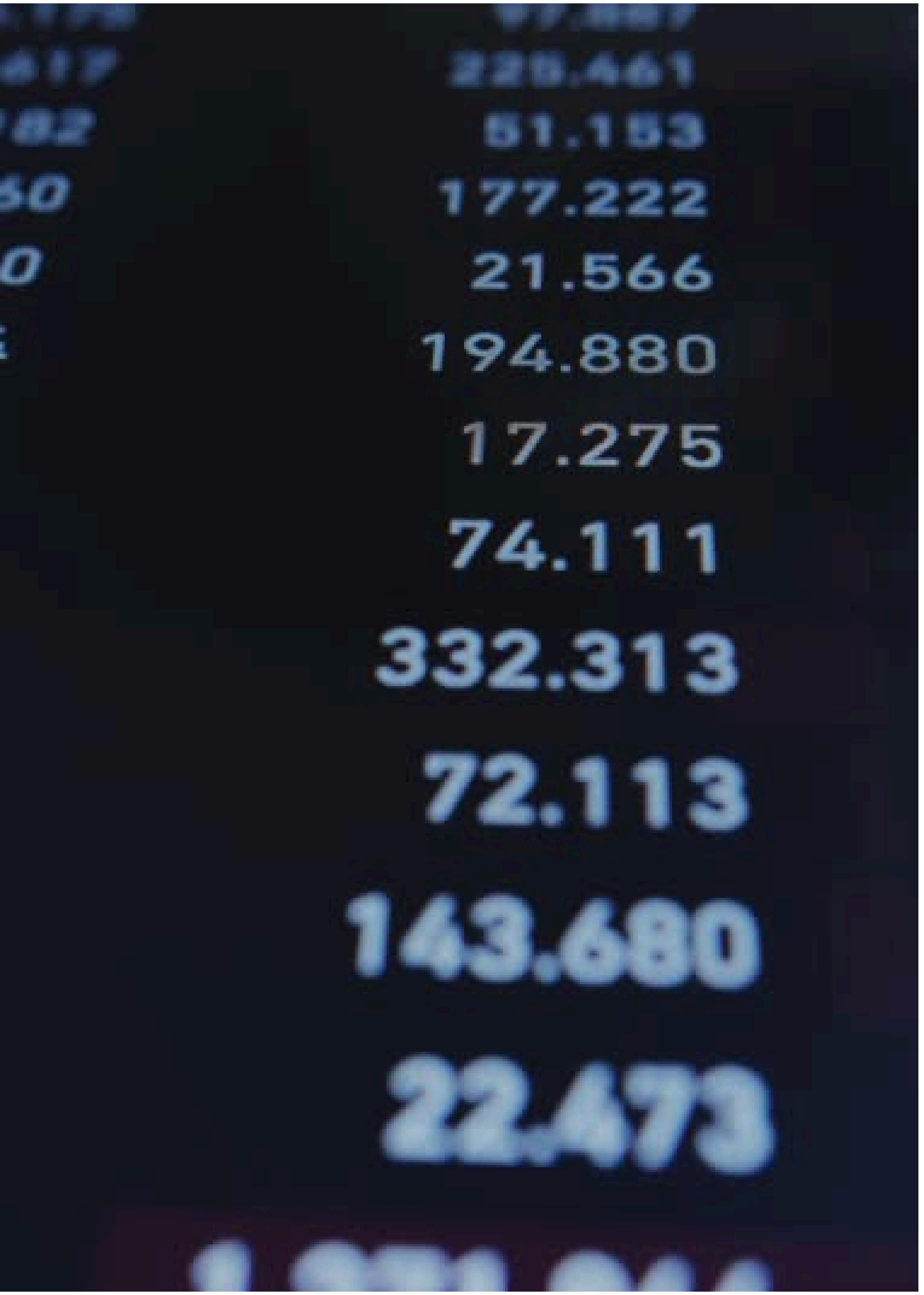
*2132300332*

If you ever need anything, please hesitate to ask, but after I consume all the deadline first.

**Tran Phuong Thao**

*2132300447*

dang suy nghi




# PROJECT MOTIVATION

*We apply AI tools in marketing, retail operations, and HR to uncover insights and improve performance.*

## (1) Marketing (Bank Marketing – UCI) - 2012

We predict customer responses, enhance targeting, and boost campaign effectiveness.

## (2) Retail (Bike Sales – Kaggle) - 2020

We identify sales patterns, forecast demand, and optimize inventory for efficiency and profitability.

## (3) HR (Employee Performance – Kaggle)

We predict employees likely to resign, enabling proactive retention strategies and reducing turnover costs.

HIGHLIGHT

# BANK MARKETING

*We use data from direct phone marketing campaigns of a Portuguese bank. Our goal is to* ***predict whether a client will subscribe to a term deposit*** *(target variable: y)*

**No. of Records**

45,211

**No. of Features**

16

**Model Applied**

Classification

**Missing Value?**

No

ABOUT AI TOOLS

# 3 TOOLS AND WHY

| Tools | Reasons |
| --- | --- |
| Python | We use Python for ***full control*** of data analysis and modeling with rich libraries for EDA and prediction. |
| Graphite Note | We use Graphite Note as a ***no-code, user-friendly tool*** to build models and generate clear business insight reports. |
| PartyRock | We use PartyRock to ***quickly prototype and test scenarios*** using prompts, without traditional model training. |

ABOUT DATASET

# KEY VARIABLES

| No | Variable Name | Type | Description |
|---|---|---|---|
| 1 | y | Target (Binary) | Whether the client subscribed to a term deposit (yes / no). |
| 2 | job | Categorical | Type of job (e.g., admin., blue-collar, student, retired, etc.). |
| 3 | education | Categorical | Education level (e.g., university degree, high school, basic, illiterate). |
| 4 | contact | Categorical | Contact communication type: cellular or telephone. |
| 5 | duration | Integer (seconds) | Length of the last contact call. Strongly linked to “yes”, but only known after the call. |
| 6 | nr_employed | Numeric | Number of employees in the economy – a macroeconomic indicator. |
| 7 | emp_var_rate | Numeric | Quarterly change rate of employment – another macroeconomic indicator. |
| 8 | pdays | Integer | Days since the client was last contacted in a previous campaign (999 = never contacted before). |

KEY FINDINGS

# DESCRIPTIVE BY PYTHON

*Bank Marketing Dataset – Summary*

## Key Demographics

- Average age: 40 (17–98)
- Most common marital status: married (61%)
- Education: university degree most frequent (29%)

## Financial Status

- Majority have no personal loan (82%)
- Housing loan: ~52% “yes”
- Default: mostly “no” (79%)

## Contact Info:

- Contacted mostly via cellular (64%)
- Peak month: May

## Target Variable:

- Positive responses (“yes”): ~11%
- Negative responses (“no”): ~89%

KEY FINDINGS WITH VISUALS

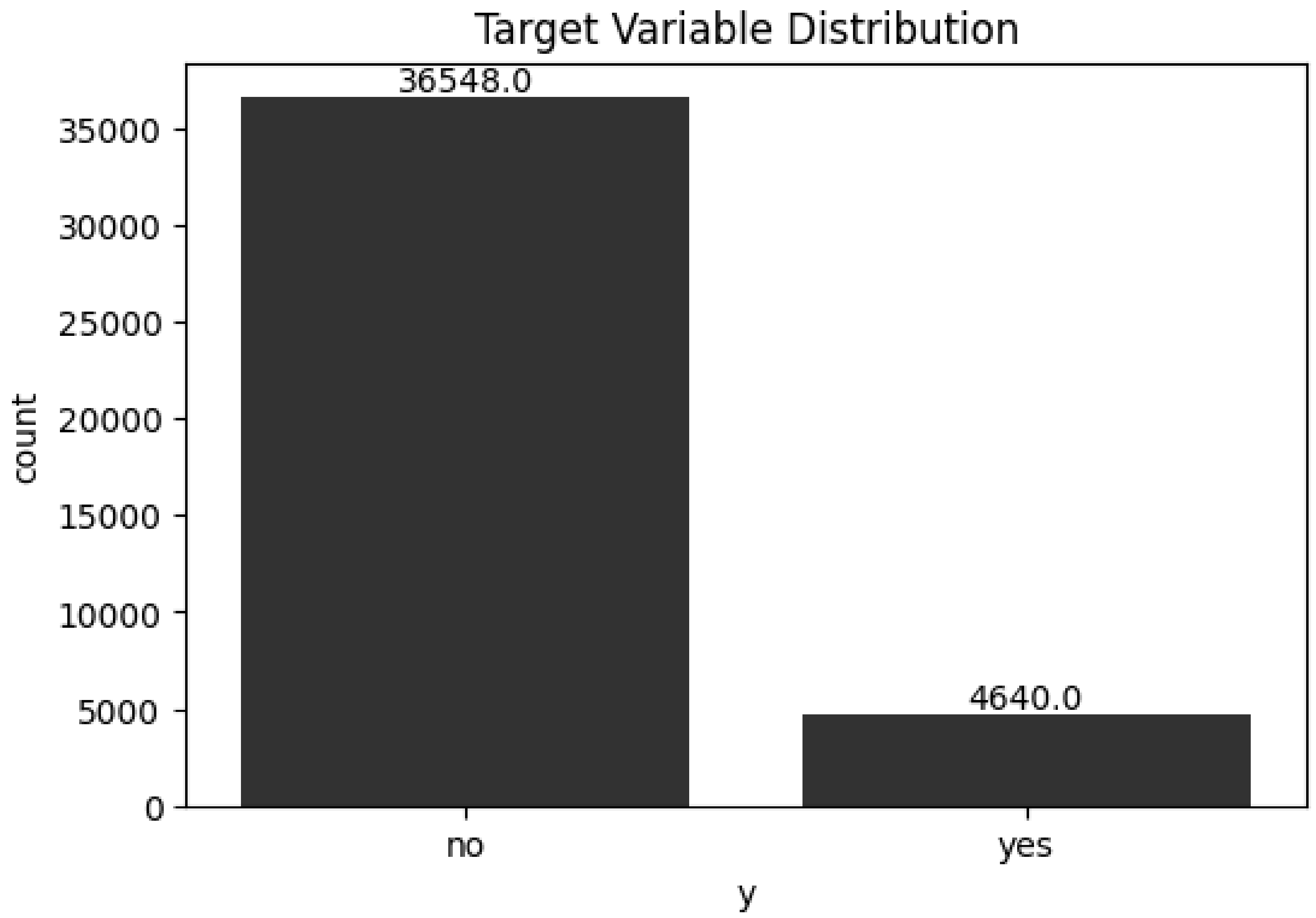
# EDA BY PYTHON

*(1) Target Variable Distribution*

**We observe a strong class imbalance:**

- No: 88.7% (36,548 records)
- Yes: 11.3% (4,640 records)

This imbalance means models may be biased toward predicting “no,” so we need techniques like resampling or class-weight adjustment to improve performance.



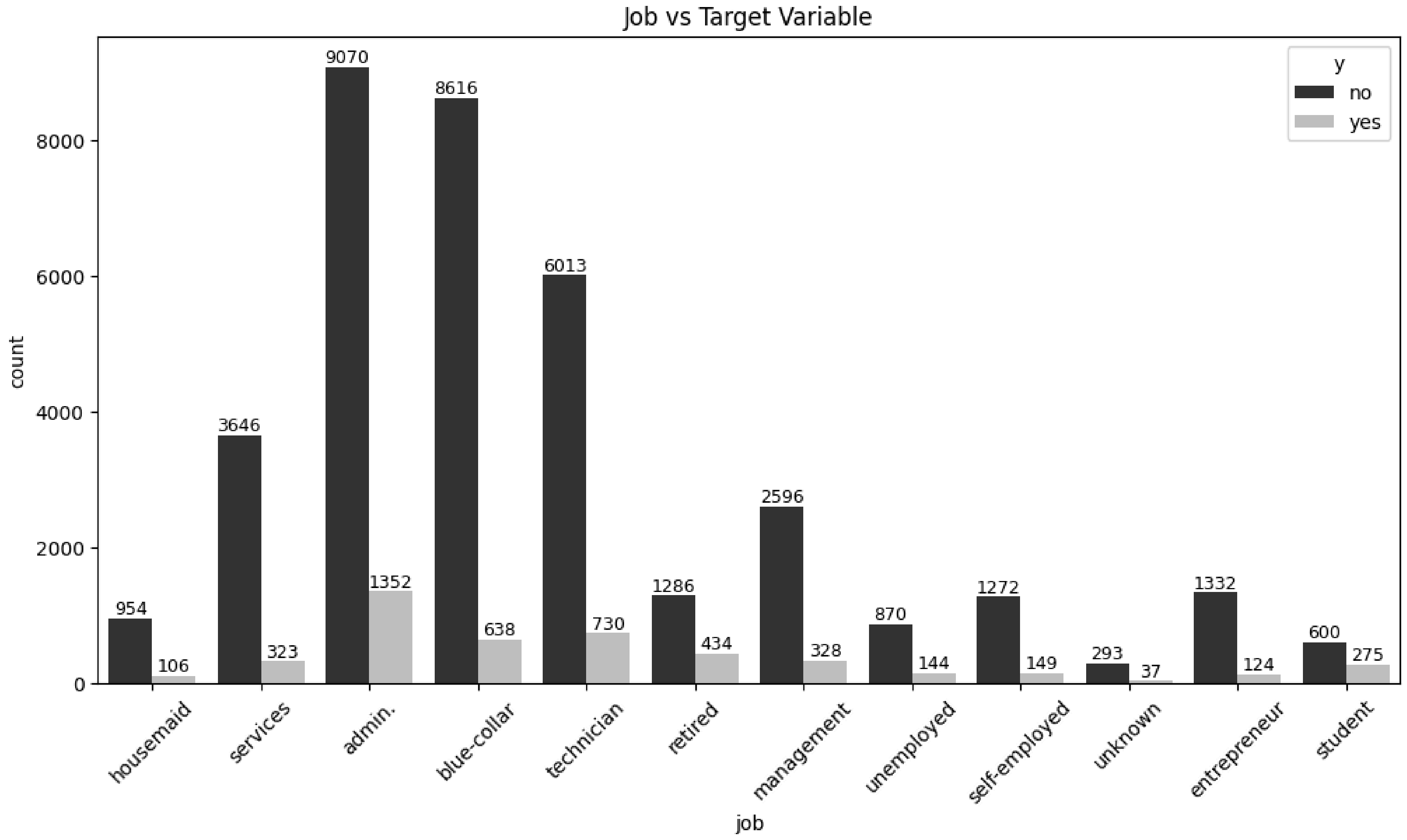
KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*(2) Job vs Subscription Outcome*

**We want to know whether job affects decision:**

- Largest groups: admin, blue-collar, technician → low “yes” rates.
- Smaller groups: student, retired, unemployed → higher “yes” rates.
- Job type clearly impacts subscription likelihood.



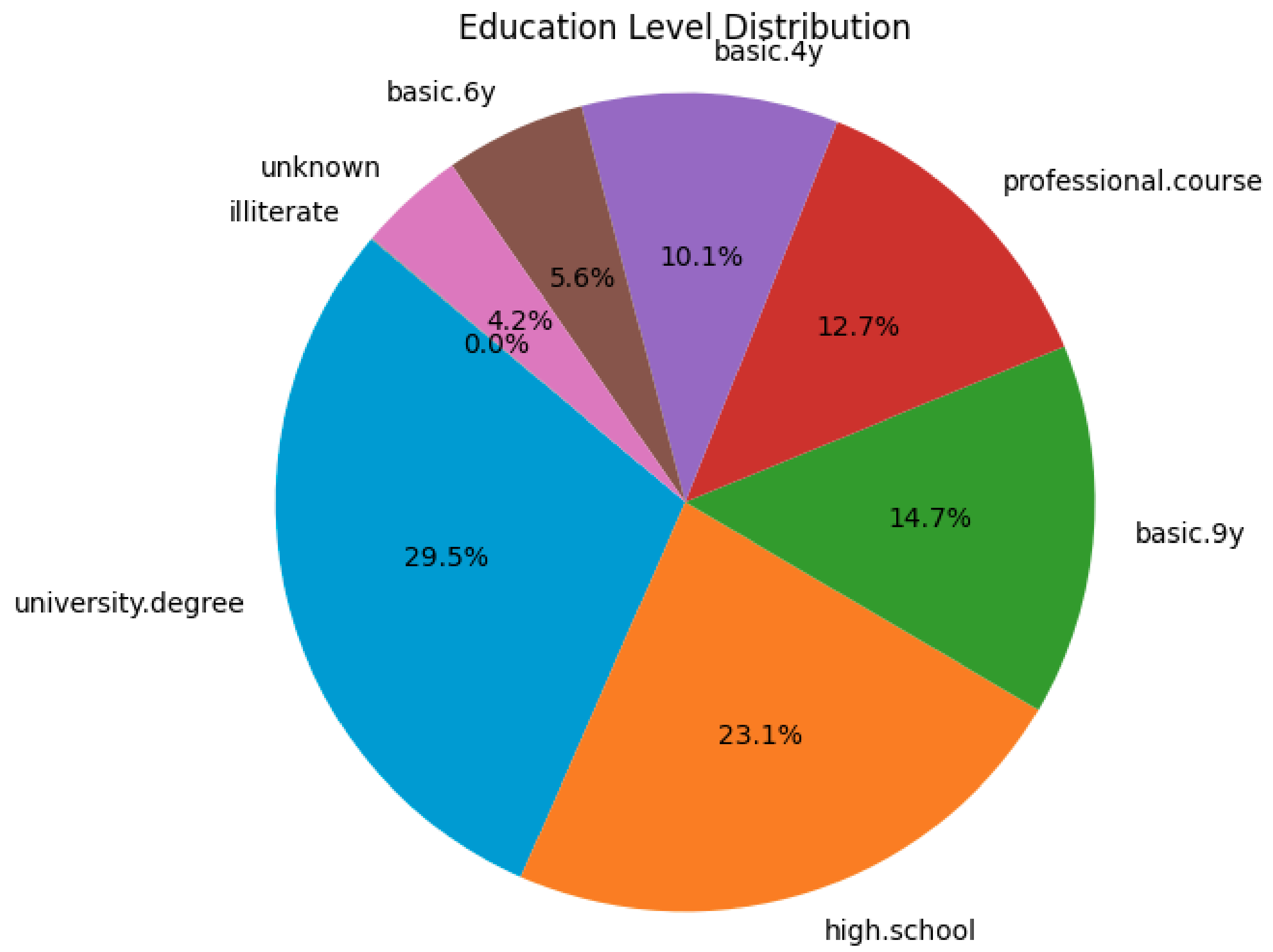
KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*(3) Education Distribution & Yes Rate*

### We want to know whether education affects decision:

- Most common: university degree (29.5%), high school (23.1%).
- Highest “yes” rates: illiterate (22.2%), unknown (14.5%).
- Higher education does not guarantee higher subscription.




| education | Yes Rate (%) |
|---|---|
| illiterate | 22.22% |
| unknown | 14.50% |
| university.degree | 13.72% |
| professional.course | 11.35% |

KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*(4) Contact Method Effectiveness*

**We want to know whether contact methods affects decision:**

- Cellular contact → 14.74% “yes” rate.
- Telephone contact → 5.23% “yes” rate.
- Mobile contact is significantly more effective.

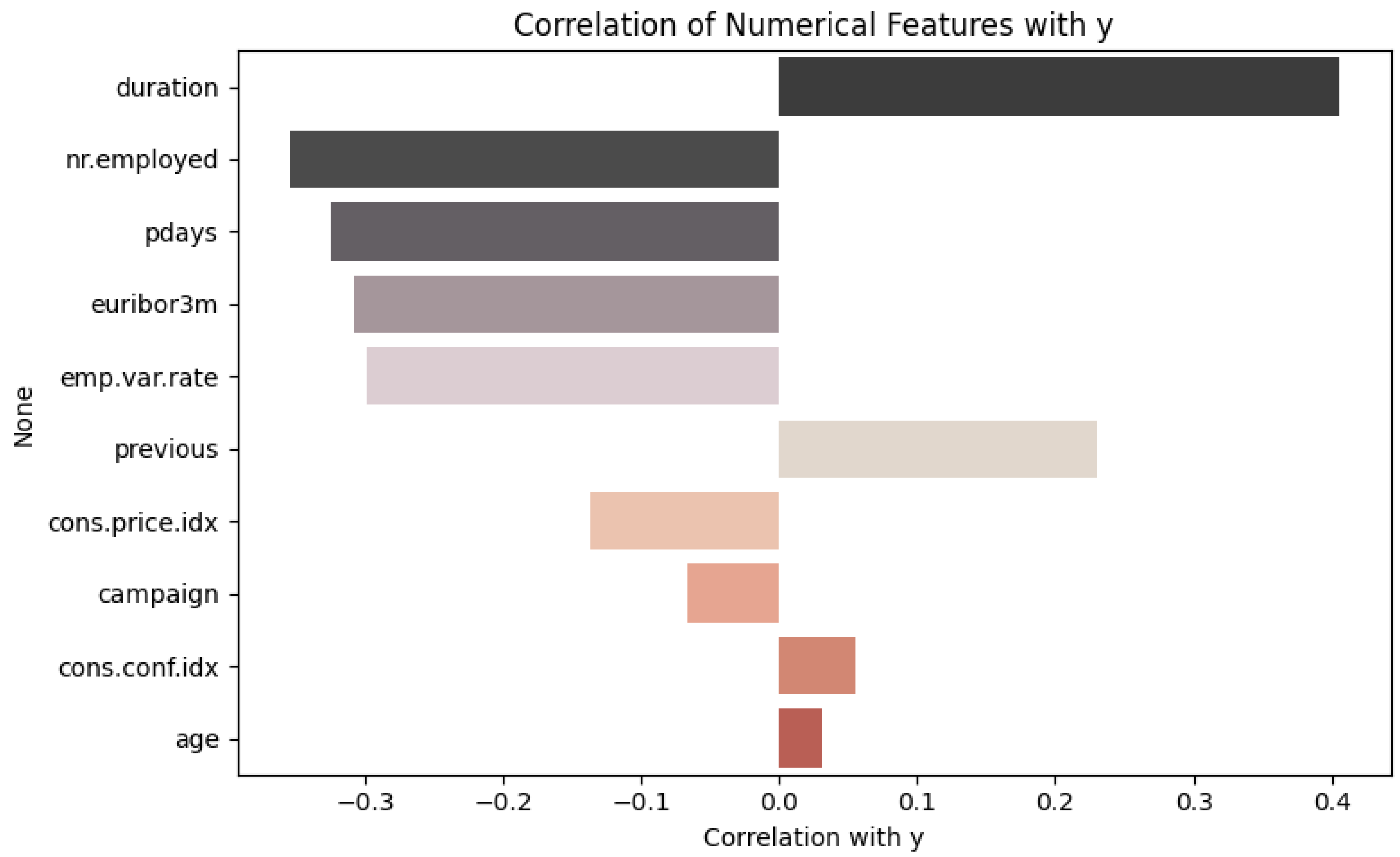
| | Yes Rate (%) |
|---|---|
| contact | |
| cellular | 14.74% |
| telephone | 5.23% |

# EDA BY PYTHON

*(5) Correlation of Numerical Features with **y***

## We observe:

- The strongest positive correlation from duration (~0.4), meaning longer calls tend to end in a subscription — but this feature is ***not usable for pre-contact prediction***.
- ***nr.employed, euribor3m, emp.var.rate*** → weak negative correlations with "yes". Macroeconomic context may slightly influence decisions.
- Other features have minimal correlation with the target.



# PYTHON MODELS OVERVIEW

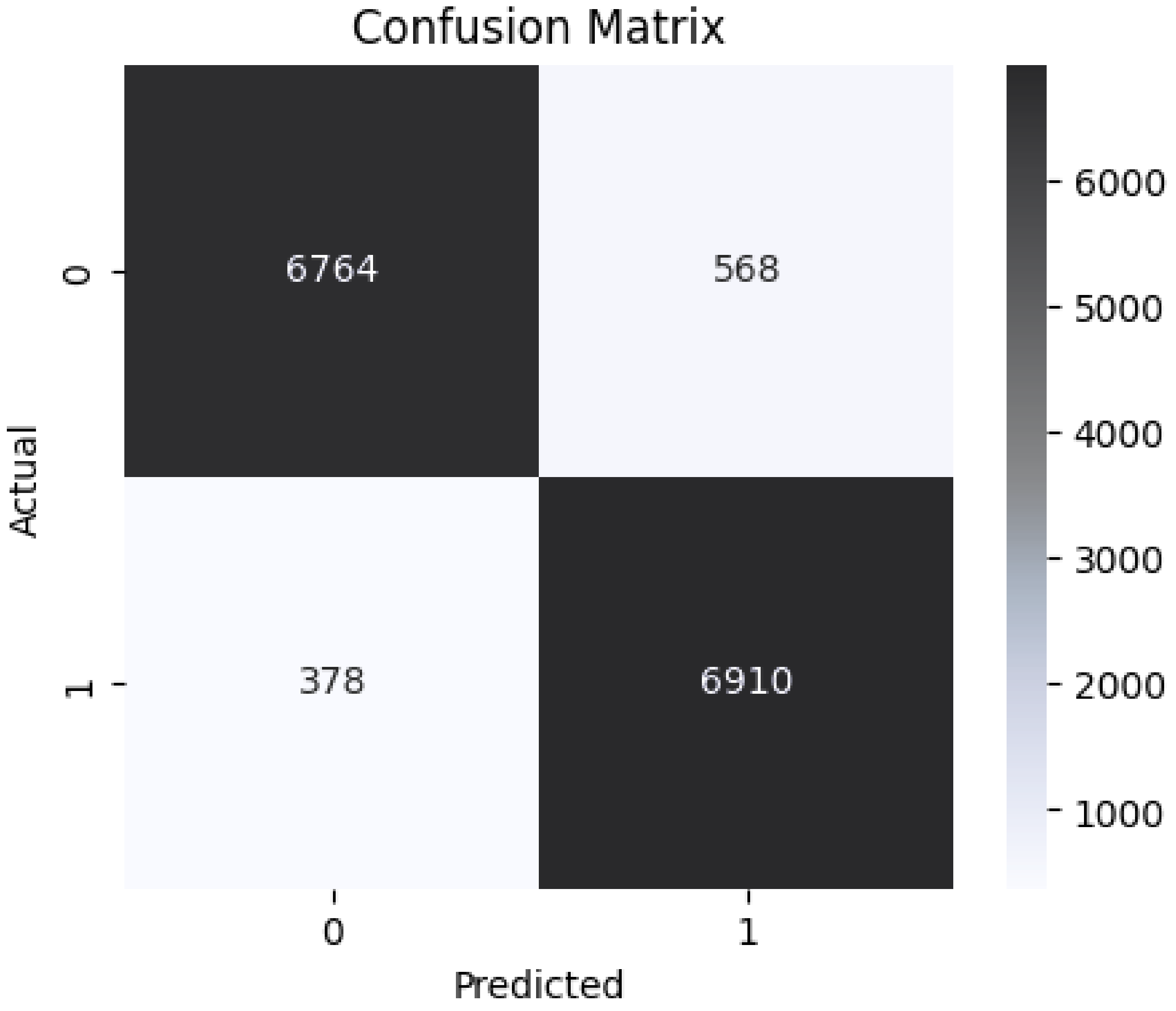
*We tested Logistic Regression & Random Forest on imbalanced data.*

## Outcome:

Initial Logistic Regression: High accuracy (91%) but low recall for “yes” (0.33) - low ability to identify “yes” outcome. Then...

## We applied SMOTE to balance classes:

- **Logistic Regression**: Accuracy ~83%, balanced F1 = 0.83
- **Random Forest**: Accuracy ~93.5%, F1 ≈ 0.93–0.94 (best performance).
- **ROC AUC** for **Random Forest** = 0.98 → strong discrimination between classes.



# GRAPHITE NOTE (LIGHTGBM)

*The best model that Graphite Note suggested is LightGBM*

## Outcome:

- F1-score = 91.23%, Accuracy = 91.56%, AUC = 94.71%.
- Balanced precision (91.02%) & recall (91.56%).

| Actual yes (1392) | Actual no (10965) |
|---|---|
| True Positives (TP) 732 | False Positives (FP) = 383 |
| False Negatives (FN) = 660 | True Negatives (TN) = 10582 |

## Key feature importance:

1. duration (51.4%)
2. nr_employed (14.41%)
3. emp_var_rate (11.77%)

**Calls > 428s**, esp. 469–551s, **greatly boost “yes” likelihood**.

# AI BY PARTYROCK

## What we did

- Built a ***prompt-based app*** to predict bank term deposit subscription ("Yes/No").
- Input: demographics, financial info, campaign details.
- Output: simple Yes/No + short explanation.

## Strengths

- ***Very fast*** to set up.
- User-friendly, no coding required.

## Limitations

- ***Does not train on our dataset*** → uses pre-built AI models.
- Predictions rely on general knowledge, not project data → risk of ***inaccuracy & bias***.
- ***Not recommended*** for data-driven projects needing high precision.

ABOUT INSIGHTS

# STRATEGIC RECOMMENDATIONS

### Leverage Economic Signals

Adjust targeting when consumer confidence is low or employment variation is negative.

### Personalize Scripts

Adapt message to customer's education and job.

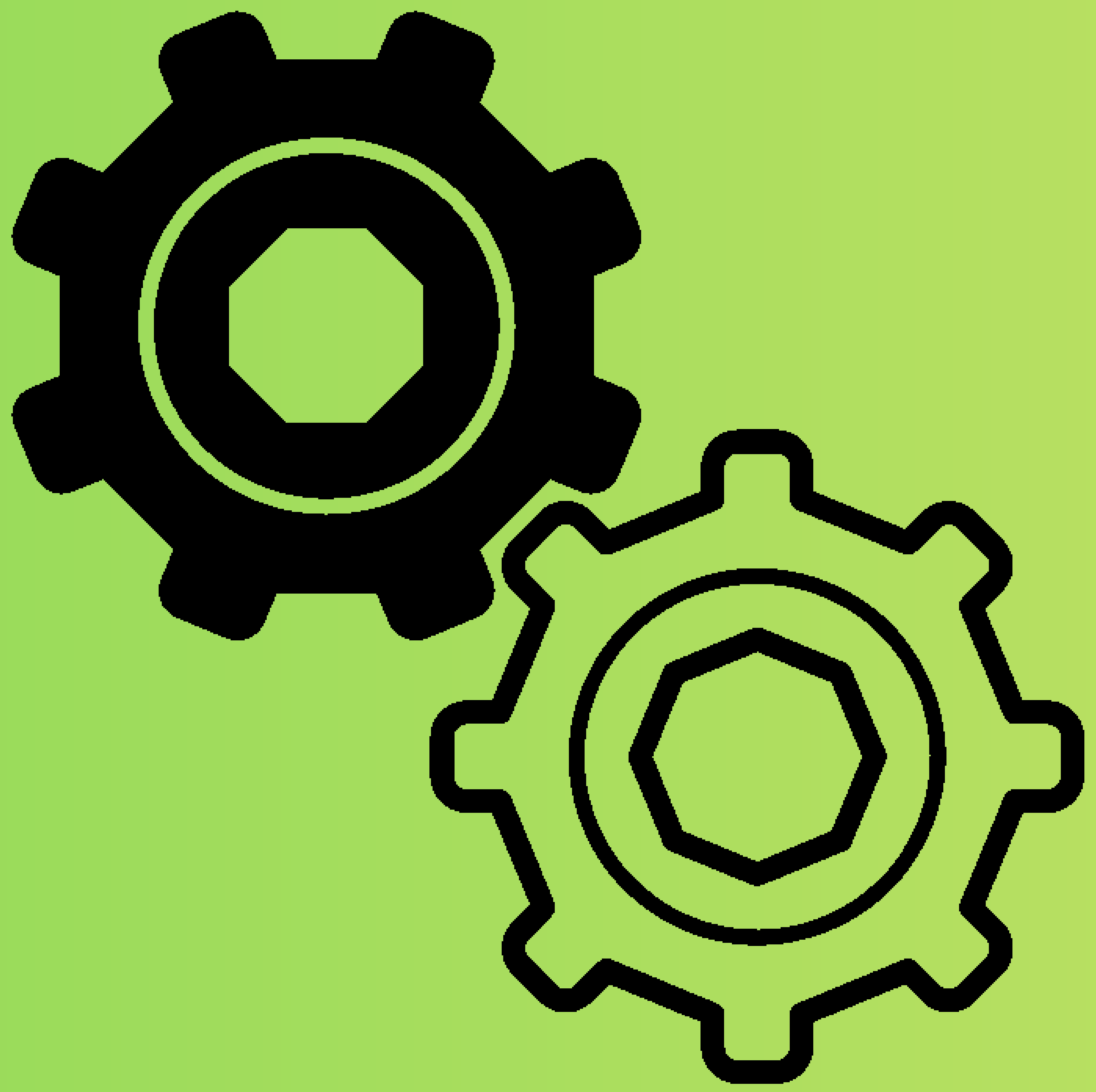
### Optimize Call Strategy

Keep calls > 428s (best: 469–551s), limit follow-ups to max 6.

# TOOL COMPARISON

*When comparing the tools in terms of usability, speed, visualization capacity, and flexibility.*

| | Python | Graphite Note | PartyRock |
|---|---|---|---|
| **Pros** | Full control, customizable pipeline, rich libraries, reproducible. | No-code, fast model building, strong visualization & automated reports. | Easiest to use, fast prototyping via prompts. |
| **Cons** | Steeper learning curve, longer development time. | Less customizable than Python. | No dataset-specific training → lower accuracy & reliability. |
| **Best for** | Technical users needing flexibility & accuracy. | Business analysts, quick iteration without coding. | Idea prototyping, concept demos (not precise modeling). |



HIGHLIGHT

# BIKE SALES

*The dataset can be **used for various analyses**, including sales trends, customer demographics, and performance evaluation of stores and sales personnel.*

**No. of Records**

100,000

**No. of Features**

11

**Model Applied**

Regression

**Missing Value?**

No

ABOUT AI TOOLS

# 3 TOOLS AND WHY

| Tools | Reasons |
| --- | --- |
| Python | We used Python (Google Colab) for ***cleaning and exploring*** 100K bike sales records, spotting trends by model or location. |
| Julius AI | We used Julius AI to quickly create ***ARIMA forecasts and visual*** revenue predictions without heavy coding. |
| Deepnote | We used Deepnote to ***collaborate on ARIMA evaluation***, check model stats, and find key sales drivers. |

KEY FINDINGS

# DESCRIPTIVE BY PYTHON

*Bike Sales Dataset – Summary*

**Price:**

- Average: $2,598 (range: $200–$4,999, SD: $1,385)
- Most sales between $1,400–$3,796

**Customer Age:**

- Average: 44 years (range: 18–70, SD: 15.3)
- Balanced demographic: 50% of customers aged 31–57

**Quantity Sold:**

- Average: 3 units per sale (range: 1–5, SD: 1.41)
- Even distribution:

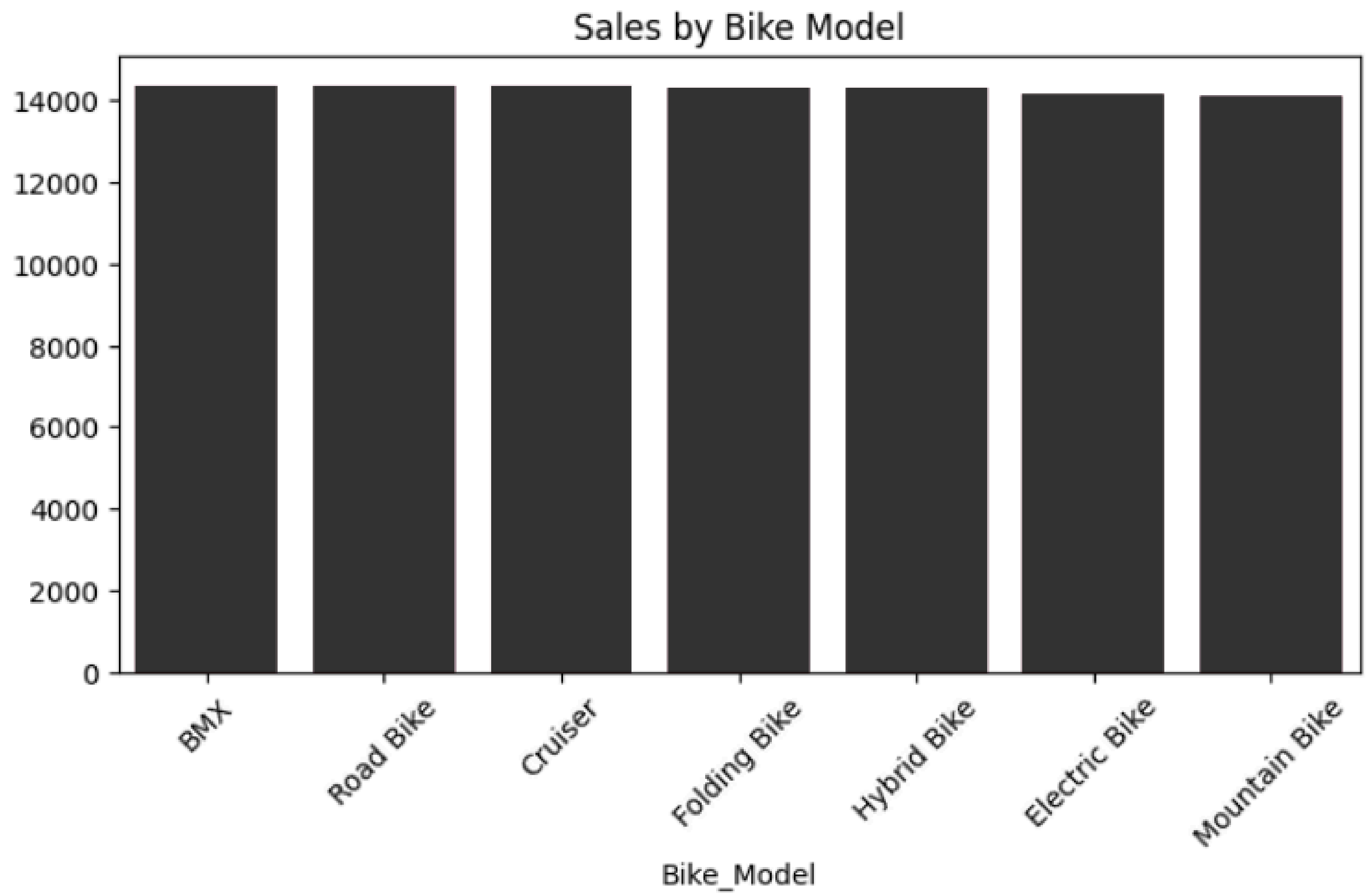
25% = 2 units
50% = 3 units
75% ≥ 4 units

- Indicates bulk purchase tendency

KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*(1) Sales by Bike Model*



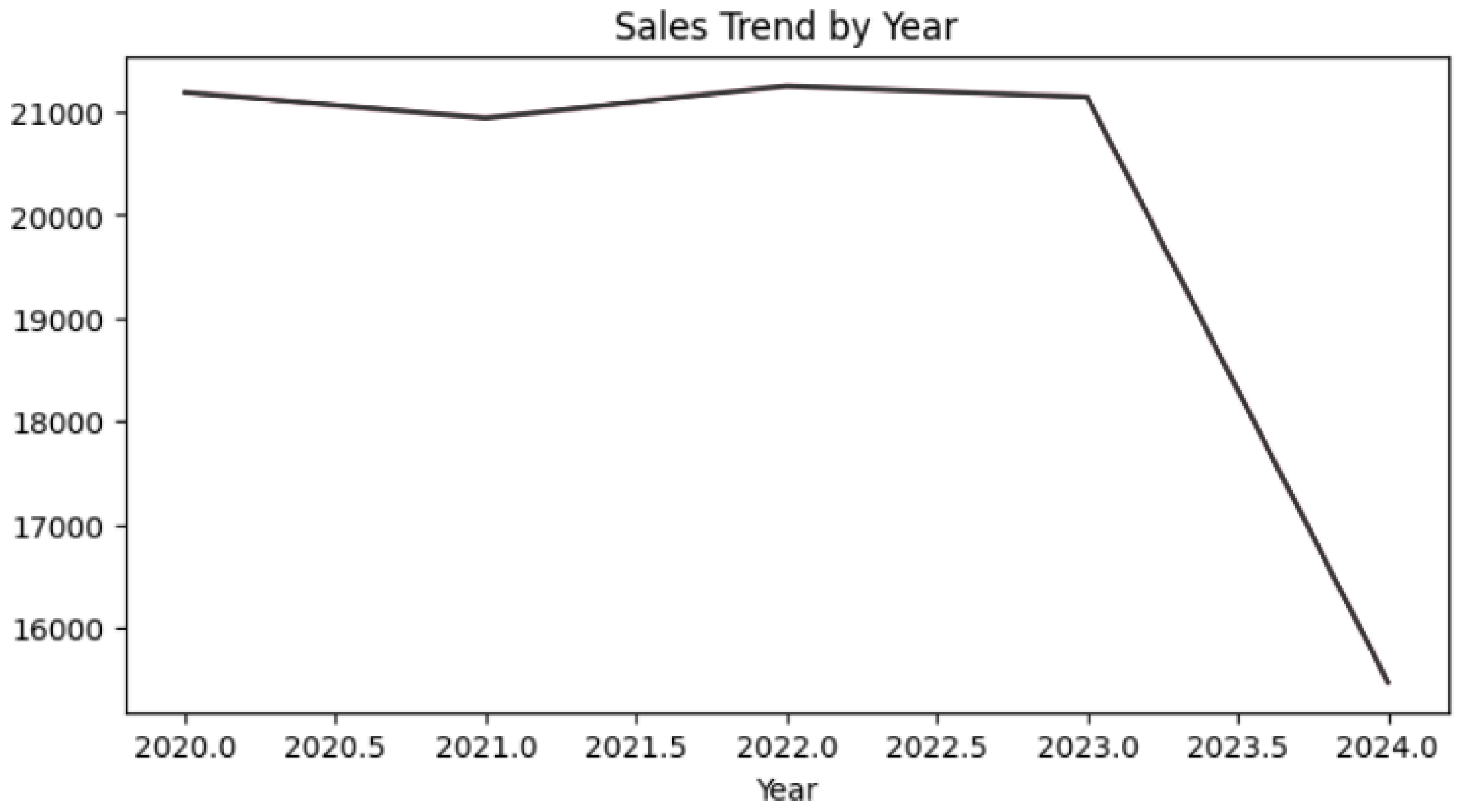

- All models (BMX, Road Bike, Cruiser, Folding, Hybrid, Electric, Mountain) sold ~12K–14K units.
- No single model dominates → ***balanced demand***.
- Recommendation: Allocate inventory & marketing evenly across all models.

KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*(2) Sales Trend (2020–2024)*

- Peak: ~21K in early 2020.
- Stable: ~20K from 2021 to mid-2022.
- Decline: Down to ~16K by mid-2024.
- Insight: Downward trend since 2023 → possible market saturation or external factors.

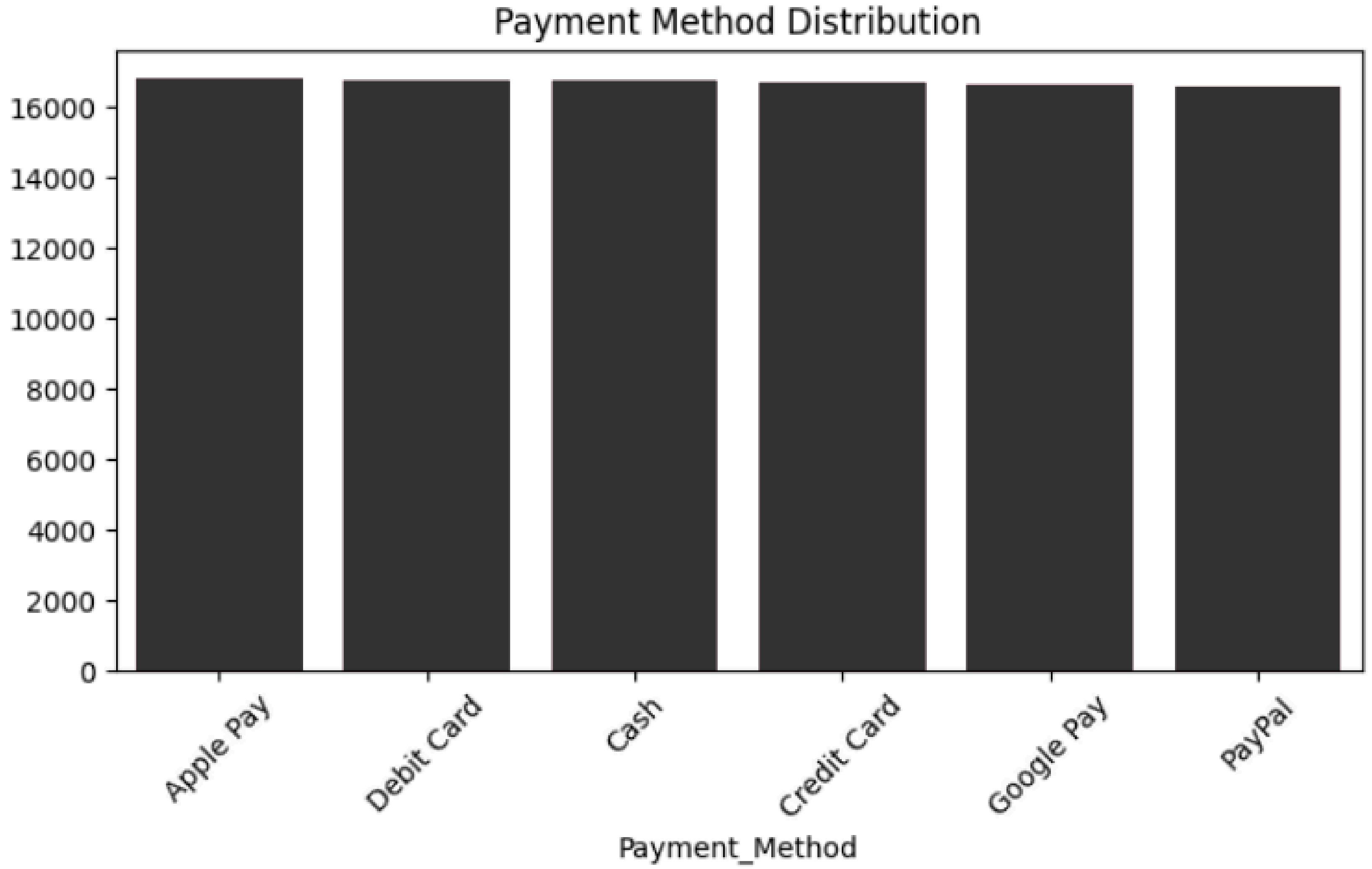


KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*(3) Payment Method Distribution*

- Range: ~14K–16K transactions across all methods (Apple Pay, Debit Card, Cash, Credit Card, Google Pay, PayPal).
- Insight: No strong preference → robust, flexible payment infrastructure.

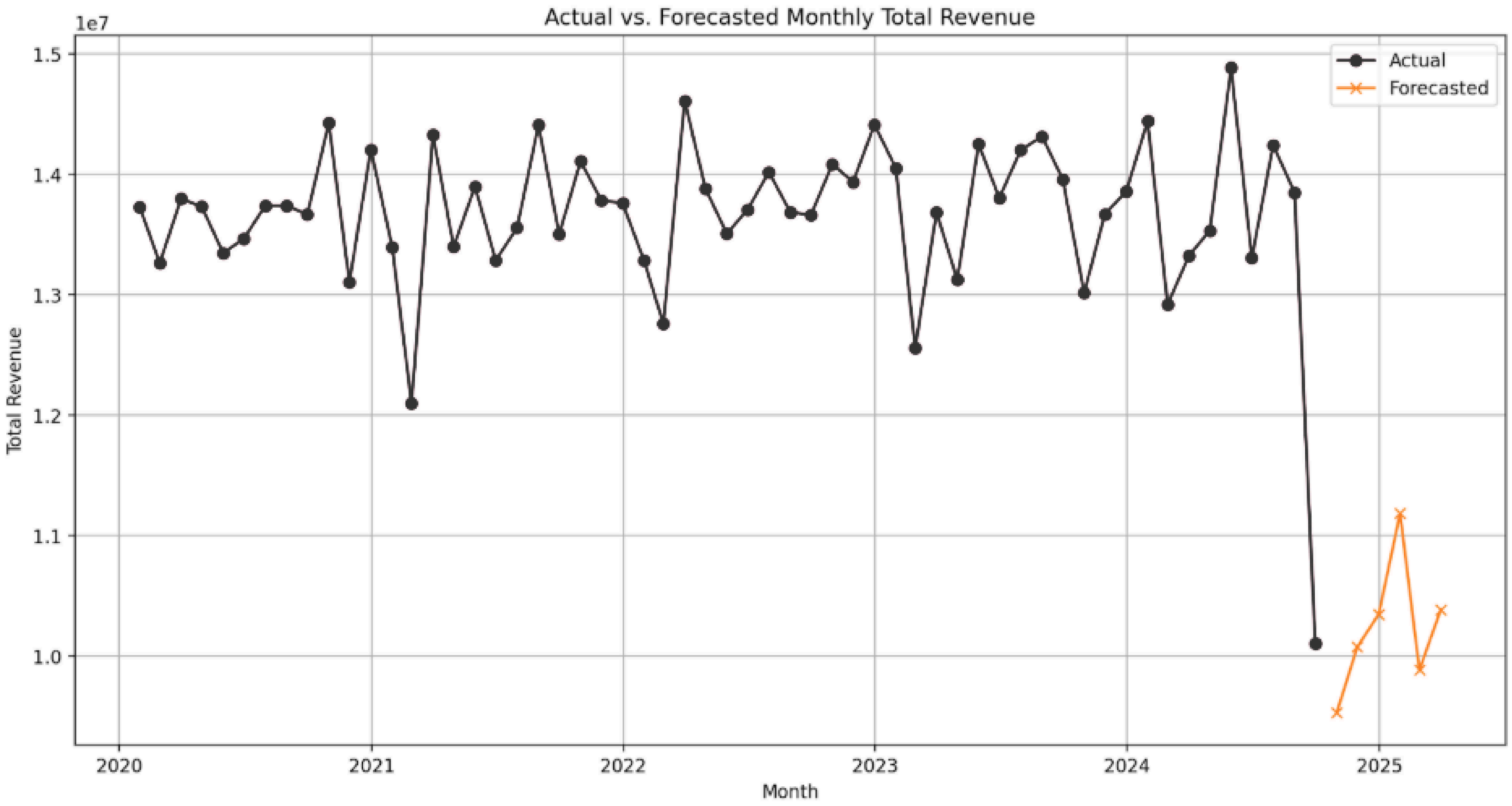


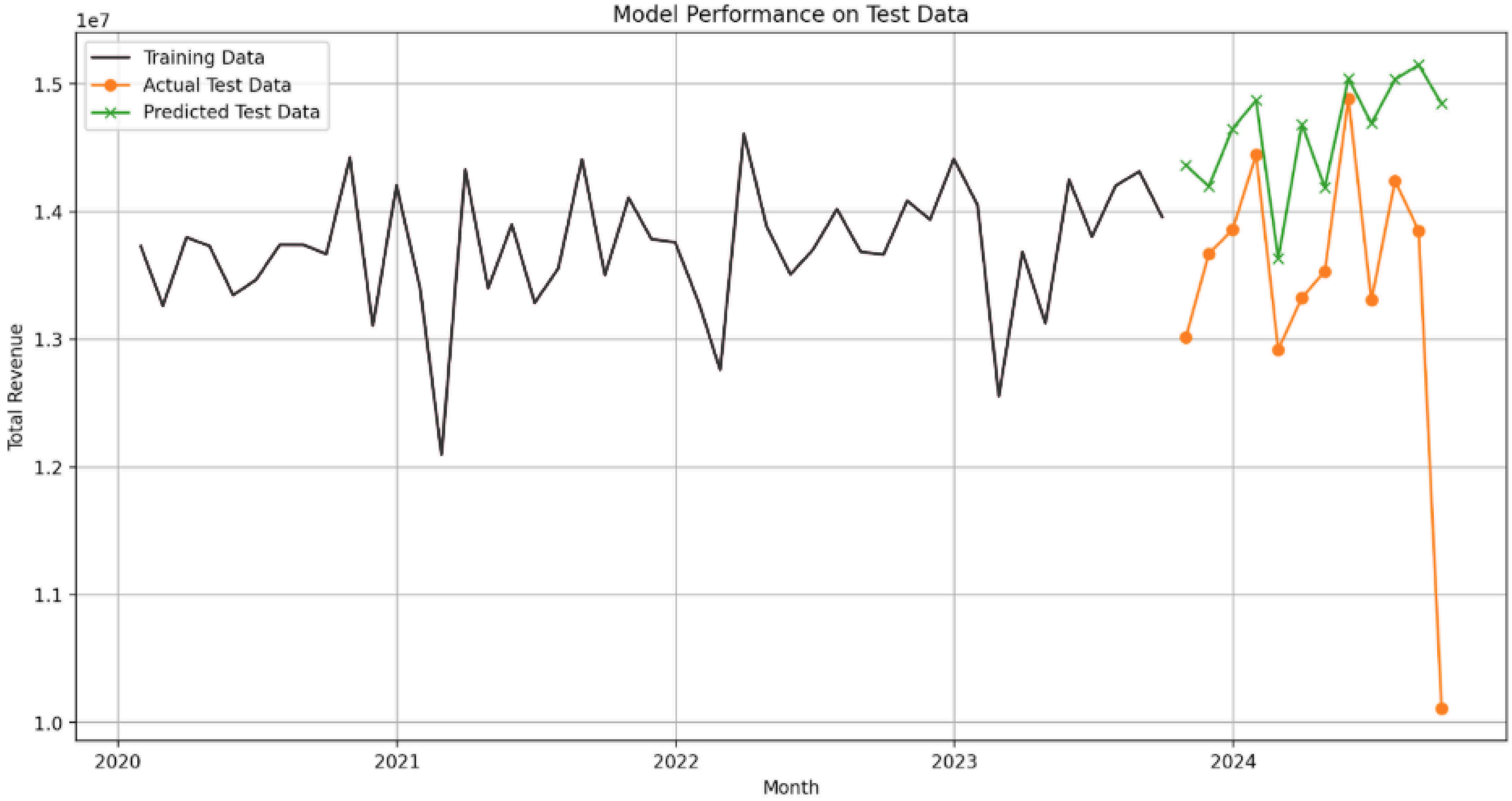
# MODEL BY JULIUS AI

*We do the ARIMA forecastingggg ~*

## ARIMA Forecasting Insights:

- **Historical Trend**: Revenue stayed between $13M–$15M from 2020 to mid-2024, then dropped sharply to ~$10M.
- **Forecast:** Expected rebound to $10M–$11M by mid-2025.
- **Model Accuracy:** Predicted values closely match actuals in test data, indicating reliable short-term forecasting.
- **Note:** Recent decline likely due to external factors beyond historical patterns.


Actual vs. Forecasted Monthly Total Revenue
1e7
1.5
1.4
1.3
1.2
1.1
1.0
Total Revenue
2020
2021
2022
2023
2024
2025
Month
Actual
Forecasted


Model Performance on Test Data
1e7
Training Data
Actual Test Data
Predicted Test Data
1.5
1.4
1.3
1.2
1.1
1.0
Total Revenue
2020
2021
2022
2023
2024
Month

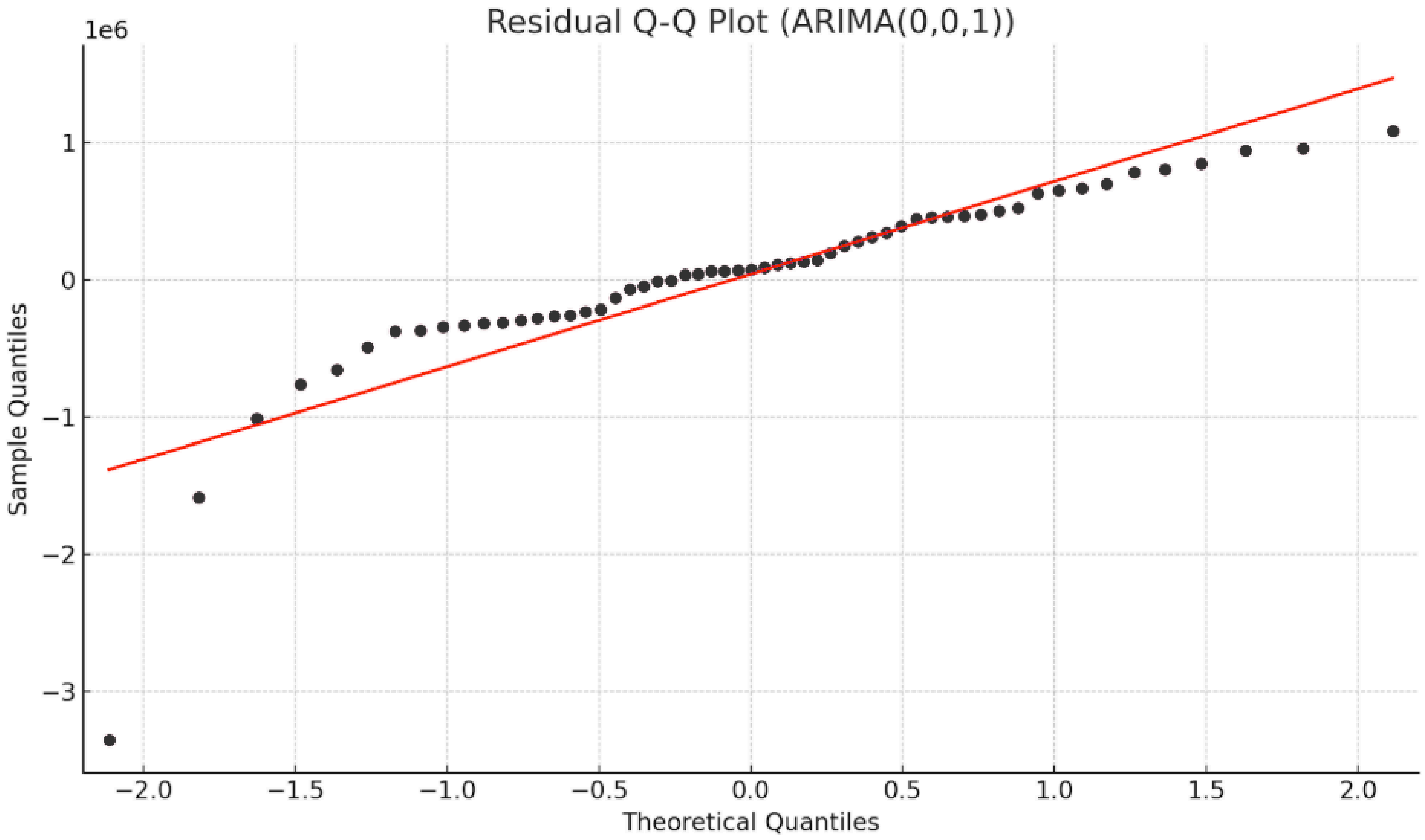
# MODEL EVALUATION BY DEEPNOTE

*We are using simple term to make audiences understand...*

## Outcome:

- **Data is stable:** Our sales numbers don’t change in unpredictable ways, so we can forecast without extra data cleaning (ARIMA d = 0).
- **Sales patterns:** What happens this month is linked to the last few months, but only for a short time.
- **Best choice:** A small, simple forecasting model gives almost the same accuracy as more complicated ones.
- **Reliable results:** Predictions match real sales well, with only a few unusual months.

**Takeaway: We can make trustworthy forecasts using a simple model that’s easy to maintain (in this case is MA(1) model)**


Residual Q-Q Plot (ARIMA(0,0,1))
1e6
Sample Quantiles
Theoretical Quantiles
1
0
−1
−2
−3
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

# TOOL COMPARISON

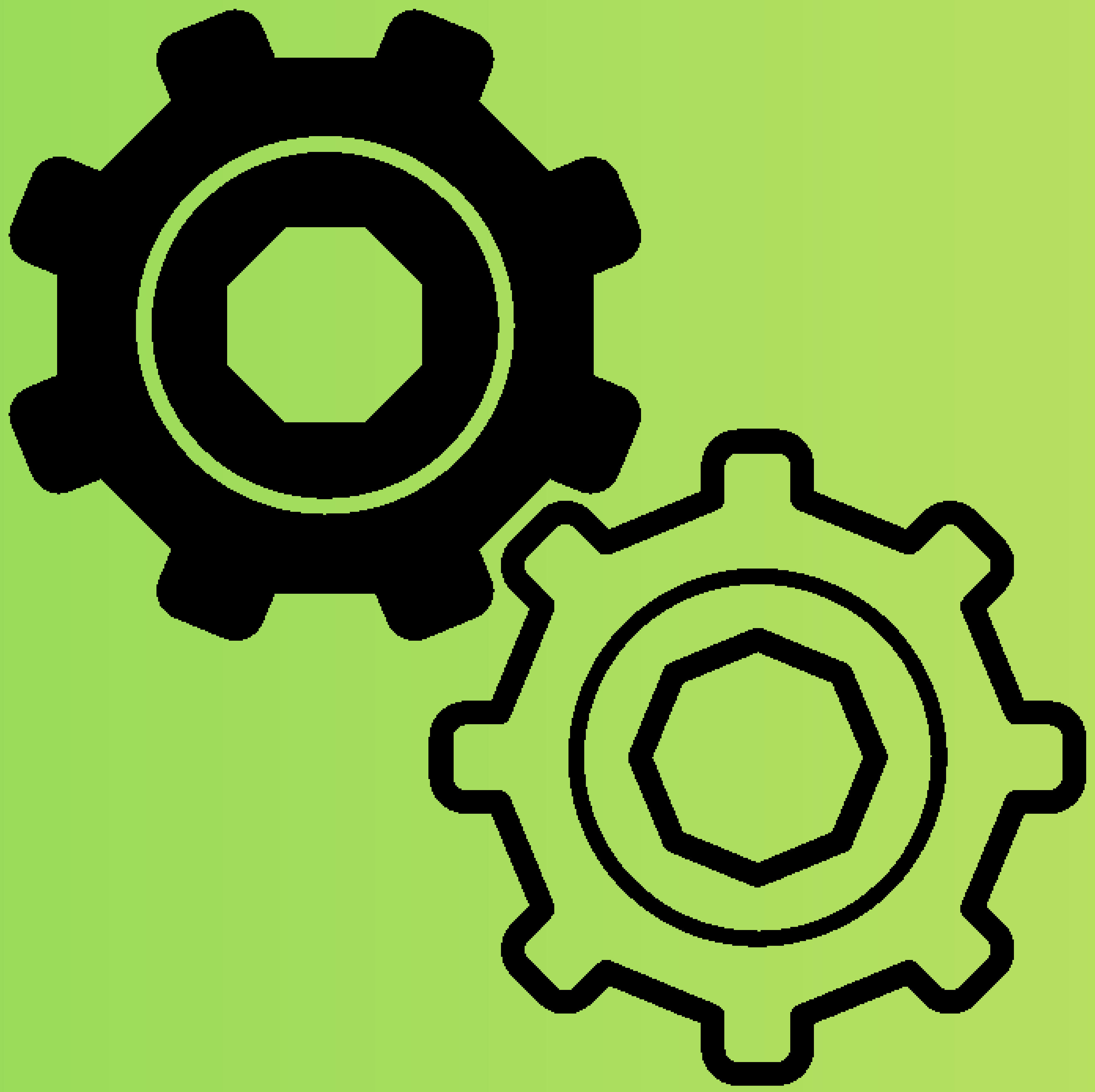
*Comparing the tools in terms of usability, speed, visualization capacity, and flexibility.*

| Python (Colab) | Julius AI | Deepnote |
|---|---|---|
| Free cloud tool for full control of data analysis, but requires coding skills. | No coding needed; quickly makes forecasts and charts from plain English requests. | Great for teamwork, mixing simple coding with easy-to-read visuals. |

Need full control & coding? → Google Colab.

No coding, quick results? → Julius AI.

Teamwork & balanced approach? → Deepnote



OVERVIEW

# EMPLOYEE PERFORMANCE AND PRODUCTIVITY DATA

*This dataset capture key aspects of employee performance, productivity, and demographics in a corporate environment.*

**No. of Records**

100000

**No. of Features**

20

**Model Applied**

Classification

**Missing Value?**

No

OBJECTIVE

# EMPLOYEE PERFORMANCE AND PRODUCTIVITY DATA

***Churn Prediction:*** *Identifying patterns that lead to employee resignation.*

ABOUT AI TOOLS

# 3 TOOLS AND PURPOSE

| Tools | Purpose |
|---|---|
| colab | Data preprocessing, visualization, and building/evaluating classification. |
| Julius | Quick, code-free exploratory analysis and initial data insights. |
| Graphite | No-code model building and clear, narrative-driven presentation of results. |

KEY FINDINGS

# DESCRIPTIVE BY PYTHON

*Employee Performance and Productivity* Dataset

## Demographics

The average employee age is 41 years, with an average tenure of 4.5 years at the company.

## Compensation

The average monthly salary stands at $8,403.

## Workload

On average, employees work 45 hours per week, handle 24 projects, and log an additional 14.5 overtime hours monthly.

## Performance & Satisfaction

Both Performance_Score and Employee_Satisfaction_Score average around 3.0 on a 5-point scale

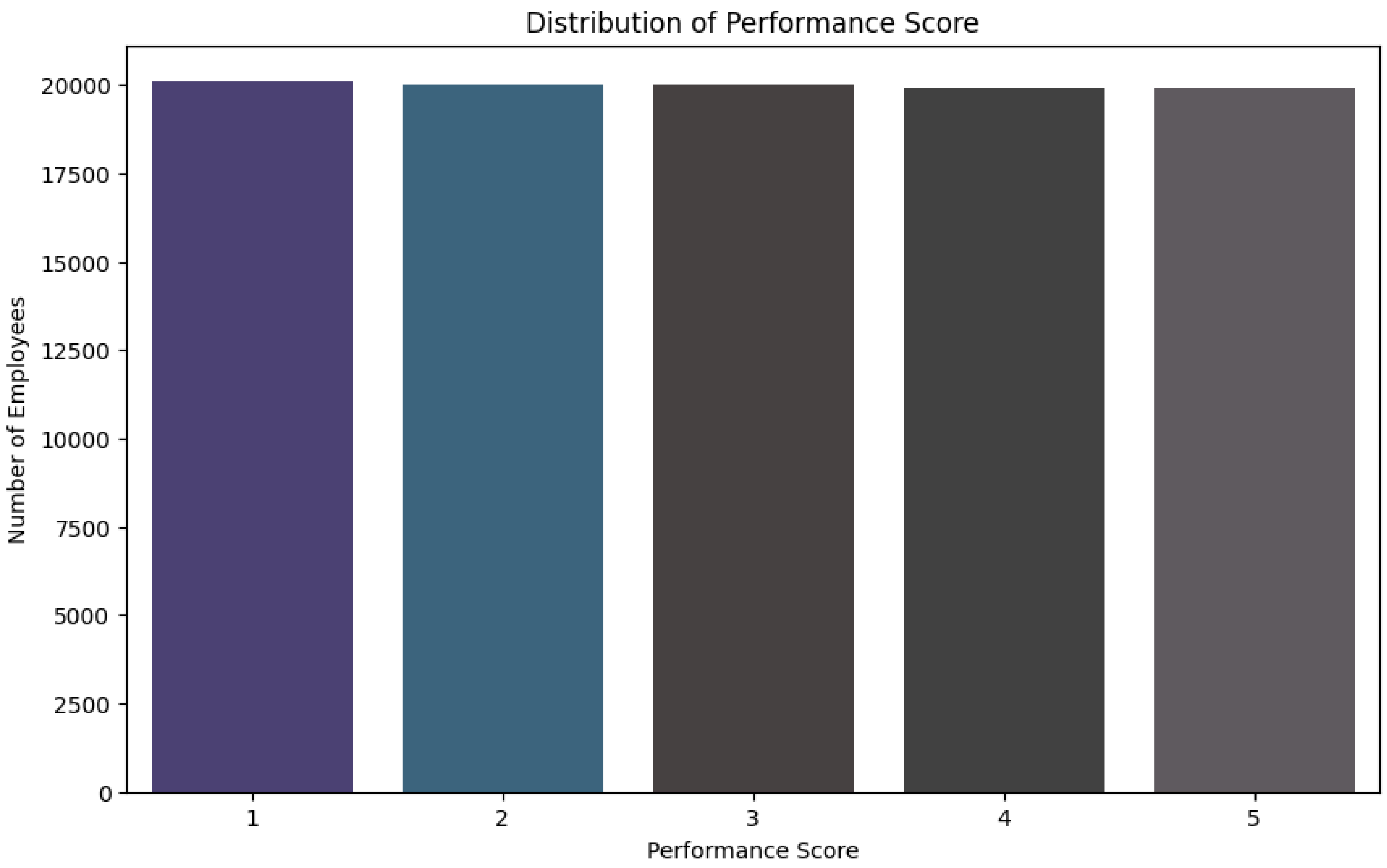
KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*Distribution of Performance Score*

Performance_Score is evenly distributed across its categories, with each score level occurring at roughly similar frequencies.

=> No strong skew toward low or high performance in the dataset.



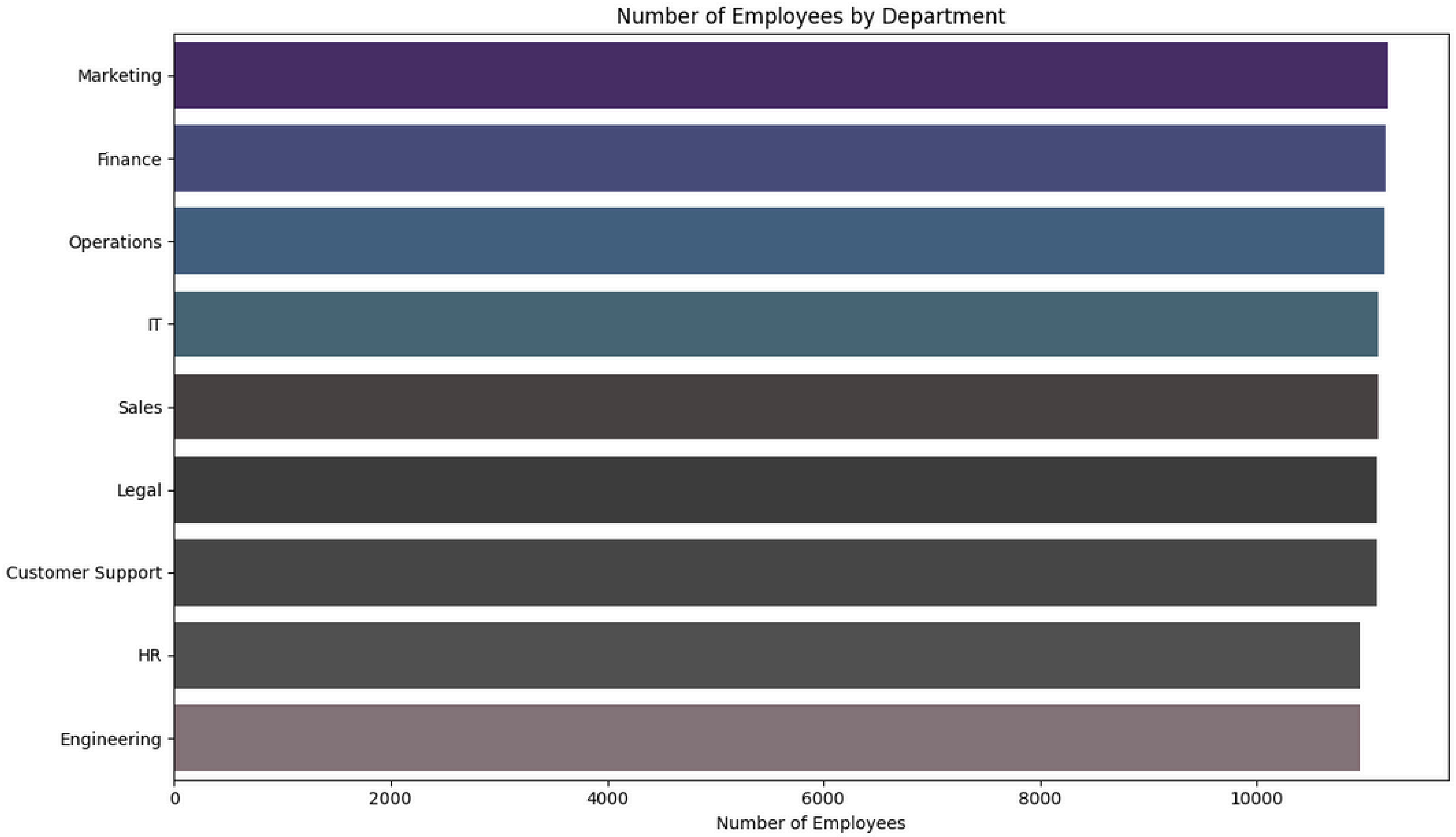
KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*Number of Employees by Department*

**HR and Engineering** have slightly fewer employees compared to other departments, while Marketing, Finance, and Operations maintain similar staffing levels.

=> Workforce allocation is fairly uniform across the organization.



KEY FINDINGS WITH VISUALS

# EDA BY JULIUS AI

*Monthly Salary Distribution by Performance Score*

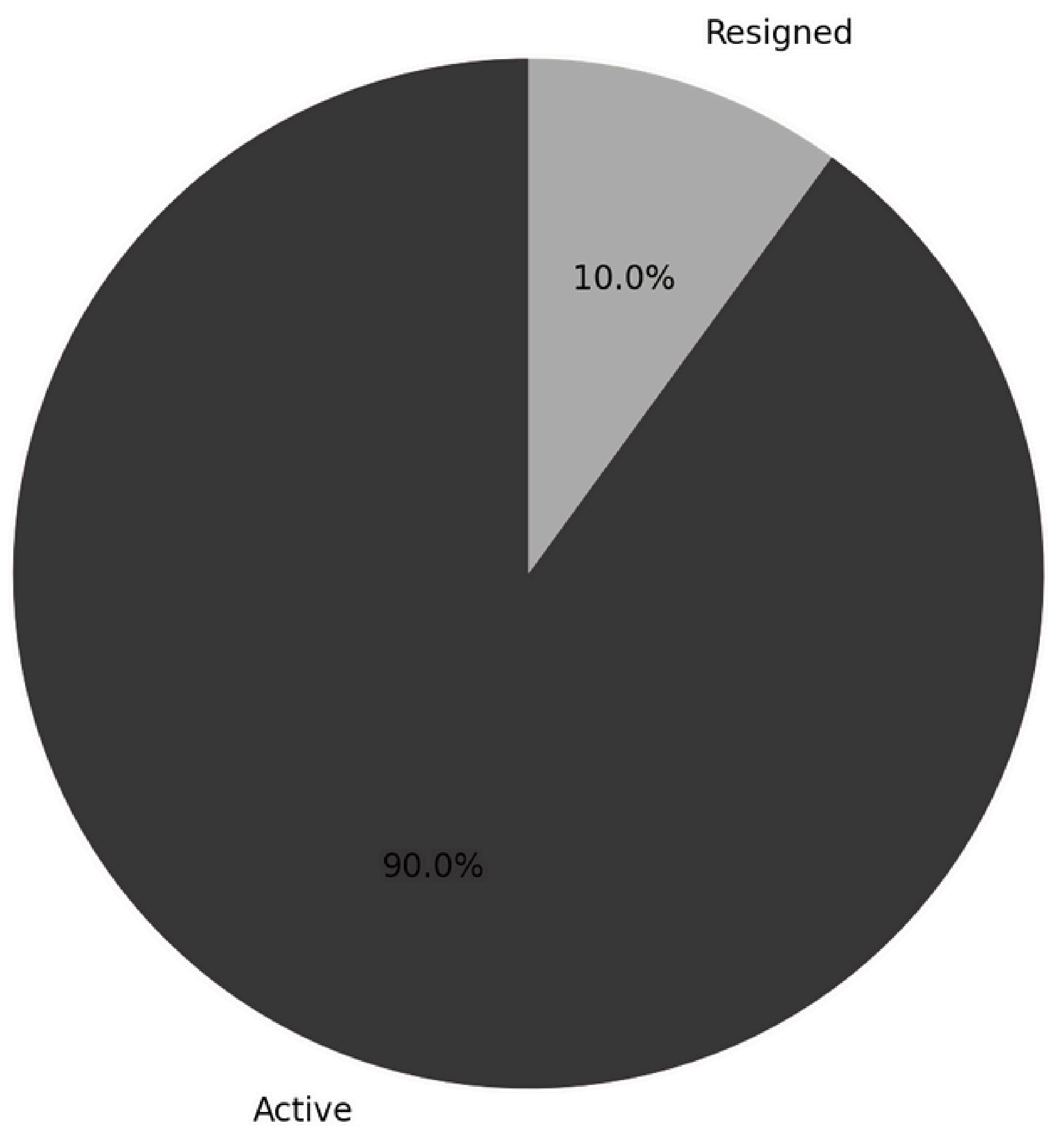
Median monthly salary tends to increase with higher Performance_Score, and variability (IQR and outliers) grows at the top scores

=> Higher performers not only earn more on average but also have a wider pay spread.

# EDA BY JULIUS AI

*Percentage of Employees who Resigned*

**10% of employees have resigned**, indicating relatively low attrition overall. This suggests retention is generally strong, but monitoring trends by department or tenure could reveal targeted risks.



KEY FINDINGS WITH VISUALS

# EDA BY PYTHON

*Employee Count by Department, Segmented by Resignation Status*

The majority of employees are still working, with a smaller % having resigned. The resignation count appears fairly consistent across departments, with Marketing and Finance showing slightly higher resignation numbers compared to HR and Engineering.

# PYTHON MODELS OVERVIEW

*Random Forest*

**Accuracy: 90% →** high overall accuracy, but misleading because the model fails to identify the minority “Resigned” class.

**Precision:**

- **Did Not Resign: 90% →** very accurate when predicting employees who stayed.
- **Resigned: 0% →** model never correctly predicts resignations.
- **Recall:**
- **Did Not Resign: 100% →** captures all employees who actually stayed.
- **Resigned: 0% →** completely misses employees who resigned**.**

**F1-score: ‘Resigned’ = 0.00 →** indicates total inability to detect the resignation class.

```
--- MODEL #2: RANDOM FOREST CLASSIFIER (BALANCED) ---
Accuracy: 0.8999

Classification Report:
                precision    recall  f1-score   support

Did Not Resign       0.90      1.00      0.95     17998
      Resigned       0.00      0.00      0.00      2002

      accuracy                           0.90     20000
     macro avg       0.45      0.50      0.47     20000
  weighted avg       0.81      0.90      0.85     20000
```

# PYTHON MODELS OVERVIEW

*Logistic Regression*

**Accuracy: 50% →** performs almost like random guessing, unable to clearly separate the two classes.

**Precision:**

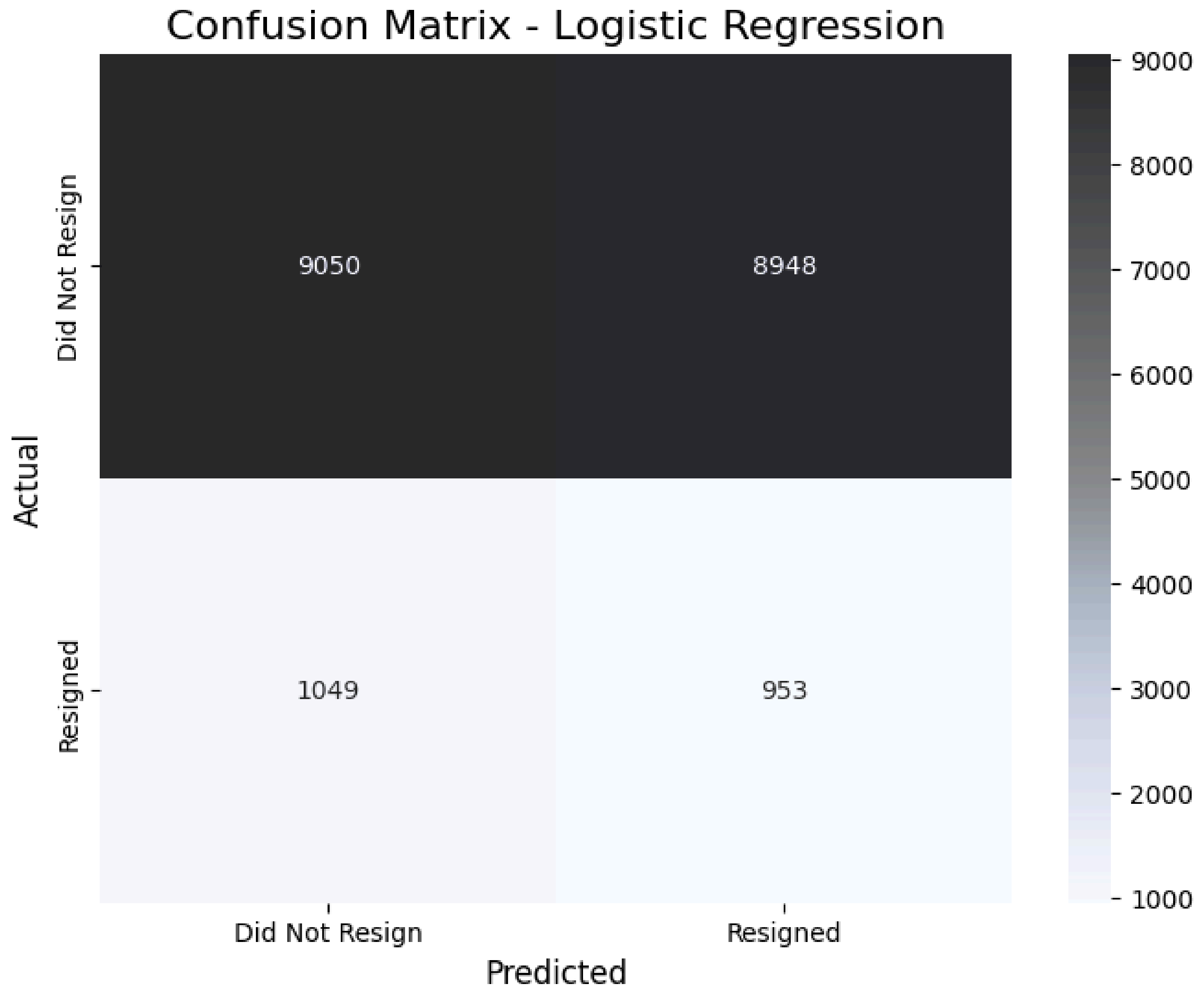
- **Did Not Resign:** 90% (very accurate when predicting employees who stayed).
- **Resigned: 10%** (most predictions for resignation are wrong).

**Recall:**

- **Did Not Resign: 50% →** misses many employees who actually stayed.
- **Resigned: 48% →** captures less than half of those who resigned.

**F1-score for 'Resigned' is very low (0.16)**

=> Logistic Regression shows poor performance



# BINARY CLASSIFICATION

*The best model that Graphite Note suggested is Logistic Regression*

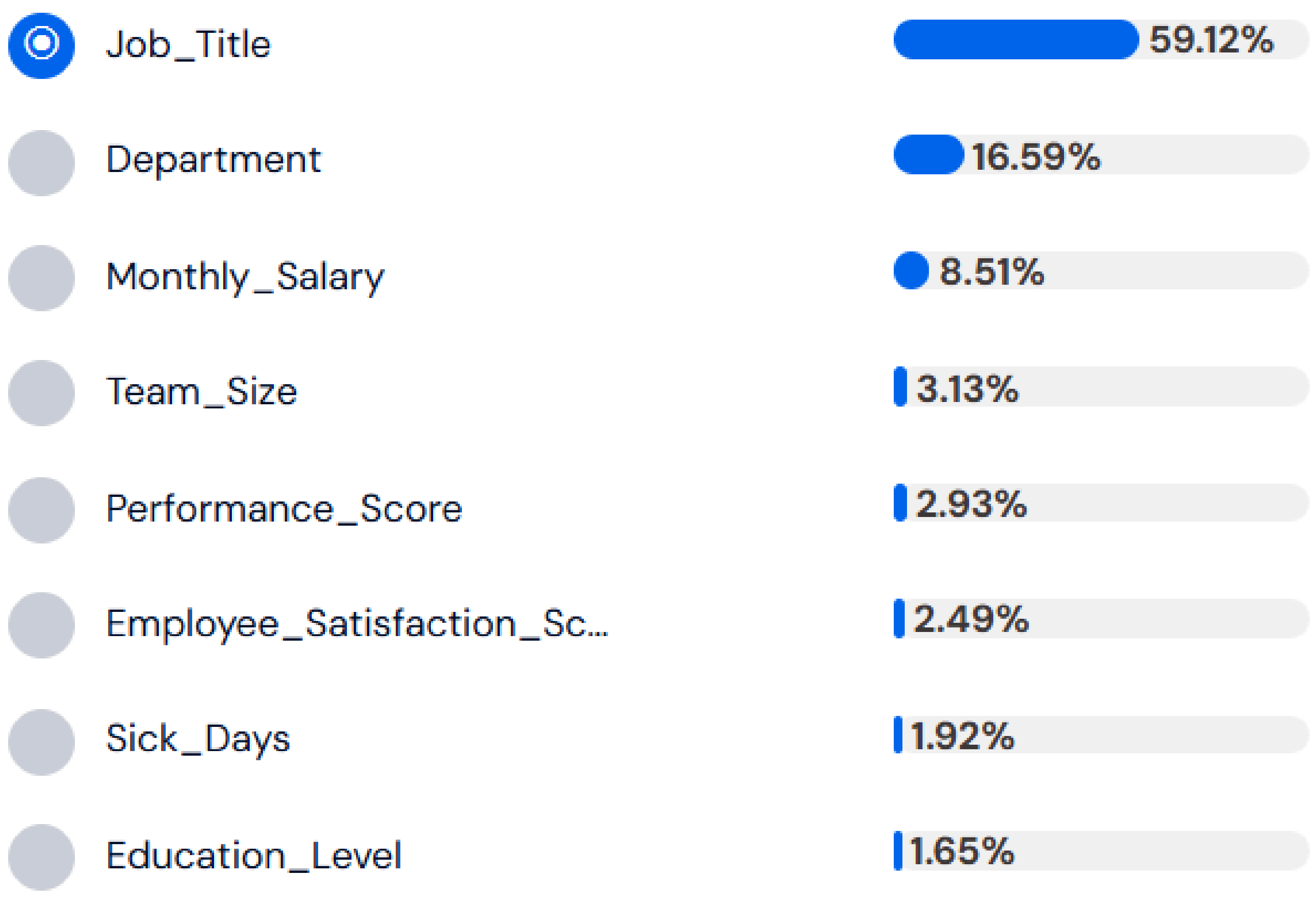
*** training scores based on a training dataset (75000 rows)**

| Model | F1 | Accuracy | AUC | MCC | Precision | Recall | |
|---|---|---|---|---|---|---|---|
| Logistic Regression | 15.91% | 50.20% | 48.45% | -1.430 | 9.570% | 47.07% | |
| K Neighbors | 15.06% | 67.99% | 50.41% | 500.0m | 10.25% | 28.34% | |
| Decision Tree | 11.24% | 79.84% | 50.02% | 50.00m | 10.05% | 12.75% | |
| MLP | 5.170% | 87.42% | 50.65% | 370.0m | 10.68% | 3.440% | |
| Random Forest | 0% | 89.99% | 50.08% | 0 | 0% | 0% | |
| Light GBM | 0% | 89.99% | 49.30% | 0 | 0% | 0% | |
| Gradient Boosting | 0% | 89.99% | 50.00% | 0 | 0% | 0% | |
| Ada Boost | 0% | 89.99% | 50.19% | 0 | 0% | 0% | |

# BINARY CLASSIFICATION

*The best model that Graphite Note suggested is Logistic Regression*

The most important key driver to predict **Resigned** is **Job_Title.**
The least important is **Remote_Work_Frequency.**



# TOOL COMPARISON

*Comparing the tools in terms of usability, speed, visualization capacity, and flexibility.*

| **Criterion** | **Python** | **Julius AI** | **Graphite Note** |
|---|---|---|---|
| **Primary Use** | In-depth analysis, custom modeling, detailed visualization | Quick EDA | No-code modeling & reporting |
| **Speed** | Slow setup, efficient for complex models | Very fast – instant from prompts | Very fast – automated model building |
| **Best For** | Precise, reproducible modeling | Early-stage insights | Polished, narrative-driven reporting |

"

WHAT WE THINK:

**AI won't take your job - but the person who knows how to use AI probably will.**

"

THIS IS THE END

# THANK YOU FOR LISTENING TO OUR TEAM