Lab: Data Exploration using Python

Name: Myo Myint Aung Jimmy

Part 1: Exploration

Introduction

This project analyzes a Portuguese bank's marketing dataset to explore customer behavior and identify factors that influence whether a client subscribes to a term deposit.
In **Part 1**, I performed exploratory data analysis (EDA) to understand the dataset structure, summarize key features, and identify variables relevant for predictive modeling. These findings guided the model-building phase in **Part 2 – Modeling**.

(1) Results of descriptive statistics

To begin, I generated descriptive statistics for all numerical and categorical variables to understand their central tendency, spread, and potential outliers.
Key Observations:

- **Age:**
  The dataset contains 41,188 customers with an average age of **40 years** (ranging from 17 to 98). Most customers fall between **32–47 years old**, indicating a mid-career population targeted by the bank.
- **Default, Housing, and Loan:**
  - Only **0.09%** of records show credit default, meaning almost all customers are in good credit standing.
  - About **54%** of customers have a housing loan, while **15%** hold a personal loan.
- **Campaign (Number of Contacts):**
  The average campaign involved about **2.6 calls per client**, but this number varies widely, with some clients being contacted more than **50 times**, suggesting potential outliers or persistent follow-up efforts.
- **Previous and Pdays (Previous Contacts):**
  - Most customers had **no prior contact** before the current campaign (pdays = 999 for 96% of cases).
  - The average number of previous contacts is **0.17**, indicating that most were first-time calls.
- **Macroeconomic Indicators:**
  - **Employment Variation Rate (emp.var.rate)** averages **0.08**, ranging from –3.4 to 1.4, reflecting mixed labor market trends during the campaign period.
  - **Consumer Price Index (cons.price.idx)** centers around **93.6**, while **Consumer Confidence Index (cons.conf.idx)** averages **–40.5**, showing generally pessimistic consumer sentiment.
  - **Euribor 3-month rate (euribor3m)** averages **3.62**, and **number of employees (nr.employed)** averages **5167**, both reflecting macroeconomic conditions in Portugal during 2008–2013.
- **Target Variable (y):**
  About **11.3%** of customers subscribed to the term deposit (y = 1), while **88.7%** did not (y = 0). This confirms a **strong class imbalance**, which later influenced how I improved the model's fairness in Part 2.

Lab: Data Exploration using Python

These descriptive insights provide an overview of both customer demographics and the broader economic context in which the marketing campaign was conducted. They also highlight key challenges, such as class imbalance and skewed distributions (e.g., campaign, pdays), that could affect model training and interpretation.

(2) Your investigation into missing values and how you dealt with them. (Remember: leaving them alone is a valid option if it's justified!)

During data inspection, I found that most columns were complete, but three categorical fields — default, housing, and loan — contained blank entries. Since these blanks represented unrecorded responses rather than true data loss, I replaced them with the label "unknown" to maintain data consistency and avoid dropping useful records.

In addition, the pdays column contained the value 999 to indicate that a client had not been previously contacted. This was kept as-is because it provides meaningful information about client history rather than missing data.

After these adjustments, the dataset had no missing values, allowing me to continue with feature exploration and modeling confidently, without imputation or row removal.

(3) Your investigation into outliers and how you dealt with them. (Remember: leaving them alone is a valid option if it's justified!)

I examined all numerical variables using boxplots and summary statistics to identify potential outliers. The main features showing outliers were campaign, pdays, and, to a smaller extent, cons.conf.idx.

campaign contained a few clients contacted more than 30 times, suggesting over-contacting cases.

pdays had a large number of 999 values, which indicate no previous contact rather than true outliers, so these were kept intentionally.

cons.conf.idx showed minor deviations at the lower tail, but since the values reflected real-world consumer sentiment rather than data entry errors, these were also retained.

Because all outlier values represented genuine variation in marketing behavior or economic conditions, I decided not to remove or cap them. Keeping them preserves the real distribution of customer responses, which helps the model learn more accurate patterns.

(4) The exploration of the relationship between your potential features and the target, e.g. answering questions like, "How did the percentage of people who bought the product vary with the age of customers?" and "Are older or younger customers more likely to buy?" Based on these answers, which features did you choose for modeling?

To understand which factors influenced customers' likelihood of subscribing to a term deposit, I analysed the relationship between key **categorical variables** and the target variable **y** ("Yes" = subscribed).

Key observations:

- **Job:** Customers with management, technician, or blue-collar jobs formed the largest groups. However, the **highest subscription rates** were observed among students and retired clients, likely because they have more time or stable savings habits.
- **Marital status:** Married clients made up the majority of contacts but had lower subscription rates compared to single clients, suggesting that singles may be more open to new financial products.
- **Education:** Clients with tertiary education showed slightly higher interest in term deposits than those with primary or secondary education levels.
- **Loan and housing:** Clients **without housing or personal loans** tended to subscribe more, implying that lower financial burden correlates with a greater willingness to invest.
- **Previous campaign outcome (poutcome):** Those previously marked as "success" were far more likely to subscribe again, confirming the predictive value of past positive responses.

These insights suggest that **demographics, financial obligations, and past campaign interactions** all influence marketing success. Such patterns can guide the marketing team to **prioritize contact with financially stable, educated, and previously responsive clients** in future campaigns.

Part 2: Modeling

(1) Describe the features you chose for each model.

Based on the findings from Part 1, I selected features that were most relevant to predicting a customer's likelihood of subscribing to a term deposit. These include a mix of **demographic**, **financial**, and **campaign-related** variables:
age, job, marital, education, default, housing, loan, contact, month, day_of_week, duration, campaign, pdays, previous, poutcome, and macroeconomic indicators (emp.var.rate, cons.price.idx, cons.conf.idx, euribor3m, nr.employed).
Categorical variables were converted into dummy variables for modeling, while numerical variables were kept as-is. The **target variable (y)** indicates whether a client subscribed to a term deposit (1 = Yes, 0 = No).

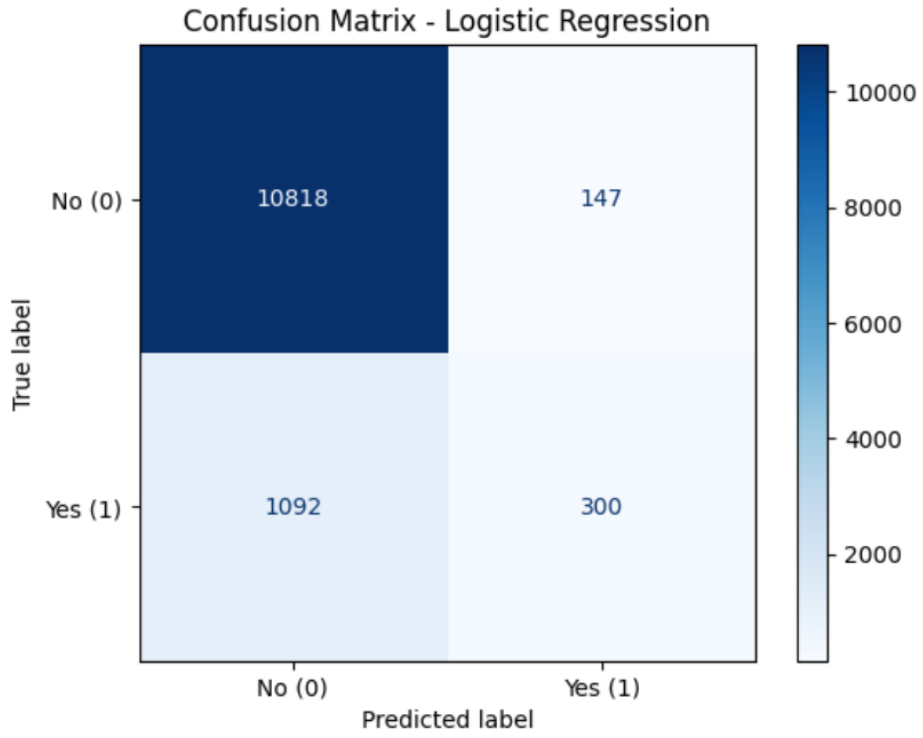
(2) Describe the model you used for each model.

Model 1 – Baseline Logistic Regression
For the first model, I used a **standard Logistic Regression** as a baseline classifier. This model estimates the probability that a client subscribes (y=1) based on the input features.
Results summary:

- **Accuracy:** 0.8997
- **Precision (Yes):** 0.67
- **Recall (Yes):** 0.22
- **F1-score (Yes):** 0.33

Although the overall accuracy is high, the model performs poorly on the minority class ("Yes"), indicating **class imbalance** — it correctly predicts most "No" cases but misses many potential subscribers.

Lab: Data Exploration using Python




```
Confusion Matrix and Classification Metrics:

[[10818   147]
 [ 1092   300]]

Classification Report:

              precision    recall  f1-score   support

      No (0)       0.91      0.99      0.95     10965
     Yes (1)       0.67      0.22      0.33      1392

    accuracy                           0.90     12357
   macro avg       0.79      0.60      0.64     12357
weighted avg       0.88      0.90      0.88     12357
```

Figure 1: Confusion Matrix – Logistic Regression

Lab: Data Exploration using Python

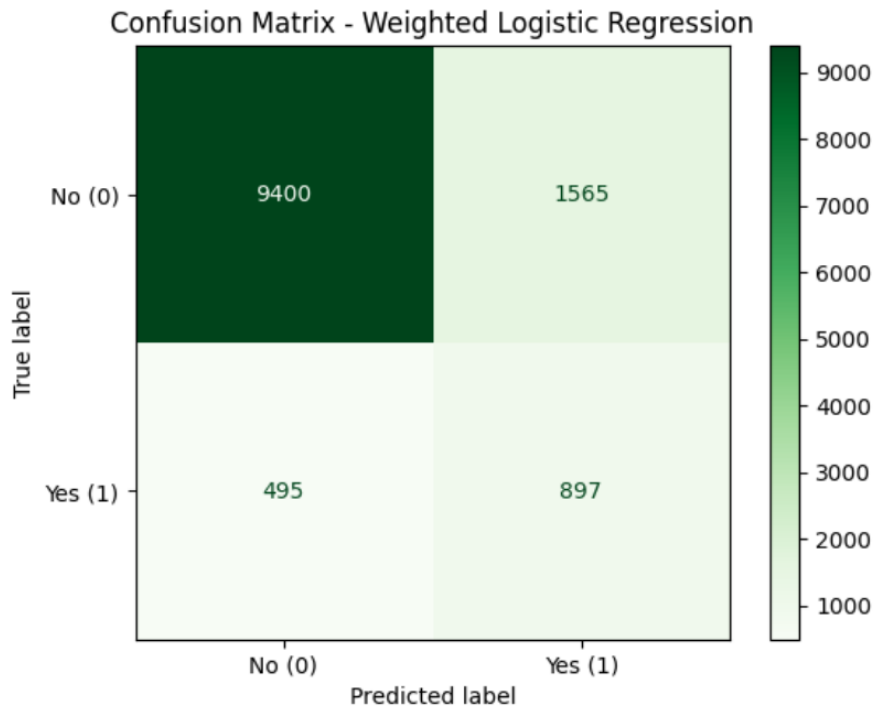
Model 2 – Weighted Logistic Regression

To handle class imbalance, I trained a **Weighted Logistic Regression** model by setting class_weight='balanced'. This approach penalizes misclassifications of the minority class, giving more weight to the underrepresented "Yes" cases.

Results summary:

- **Accuracy:** 0.8333
- **Precision (Yes):** 0.36
- **Recall (Yes):** 0.64
- **F1-score (Yes):** 0.47

The confusion matrix and heatmap (Figure 1) show that this model identifies significantly more true positives (actual subscribers) compared to the baseline model, although it sacrifices some overall accuracy.




```
Confusion Matrix (Weighted Logistic Regression):
[[9400 1565]
 [ 495  897]]

Classification Report (for cross-check):
              precision    recall  f1-score   support

      No (0)       0.95      0.86      0.90     10965
     Yes (1)       0.36      0.64      0.47      1392

    accuracy                           0.83     12357
   macro avg       0.66      0.75      0.68     12357
weighted avg       0.88      0.83      0.85     12357
```

Figure 2. Confusion Matrix – Weighted Logistic Regression

Lab: Data Exploration using Python

(3) Detail the results of both models.

- **What was their accuracy score?**
- **What did the confusion matrix reveal? Include some discussion about false positives and false negatives.**

A detailed comparison shows how both models perform differently depending on whether the focus is on **overall accuracy** or **identifying subscribers**.

Model	Accuracy	Precision (Yes)	Recall (Yes)	F1-score (Yes)
Baseline Logistic Regression	0.8997	0.67	0.22	0.33
Weighted Logistic Regression	0.8333	0.36	0.64	0.47

Accuracy:
The baseline model achieves slightly higher overall accuracy (90%) compared to the weighted version (83%). However, this comes mainly from predicting the majority class ("No") correctly — not from identifying the actual subscribers.

Confusion Matrix Insights:

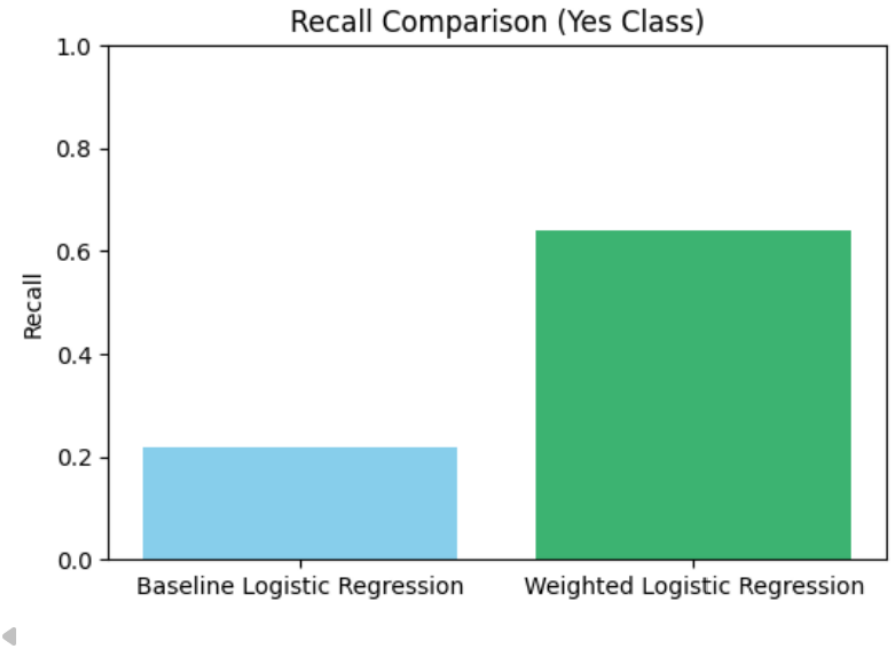
- **Baseline Logistic Regression:** The model correctly predicts most "No" cases but fails to capture many true "Yes" cases. It produces few **false positives** (predicting "Yes" when the customer didn't subscribe) but many **false negatives** (predicting "No" for customers who actually subscribed).
- **Weighted Logistic Regression:** By rebalancing class weights, this model detects far more true positives (897 "Yes" predictions) but also introduces more false positives. While overall accuracy decreases slightly, recall improves from **0.22 → 0.64**, meaning it identifies nearly three times more actual subscribers.

Interpretation:
From a marketing perspective, the **weighted model** is more practical. In this context, a **false positive** means contacting a customer who ultimately won't subscribe — which costs a small amount of time or resources. A **false negative**, however, represents a **missed opportunity** to engage someone who might have subscribed. Thus, it's better to tolerate more false positives if it helps capture a higher proportion of true subscribers.

Overall, the **Weighted Logistic Regression** strikes a better balance between precision and recall, improving campaign effectiveness even if it sacrifices a small portion of accuracy.

```
=== Model Comparison ===
```

	Model	Accuracy	Precision (Yes)	Recall (Yes)	F1-score (Yes)
0	Baseline Logistic Regression	0.8997	0.67	0.22	0.33
1	Weighted Logistic Regression	0.8333	0.36	0.64	0.47



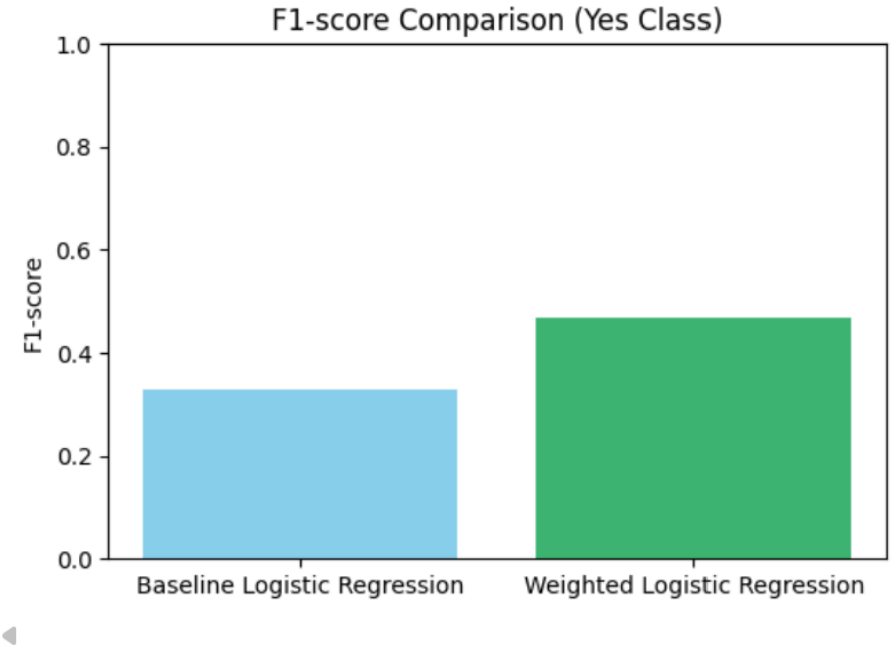


Figure 3. Recall and F1-score Comparison Between Both Models

Lab: Data Exploration using Python

(4) Decide which model performed better overall, and justify your decision. Is it because one has a higher accuracy, or is it the makeup of the confusion matrices?

After comparing both models, I conclude that the **Weighted Logistic Regression** performed better overall.
While the **Baseline Logistic Regression** achieved higher overall accuracy (≈ 90%), it failed to correctly identify many true subscribers (low recall = 0.22). This means that in a real marketing campaign, most potential customers who were likely to subscribe would have been missed.

In contrast, the **Weighted Logistic Regression** achieved a more balanced trade-off between precision (0.36) and recall (0.64), producing an F1-score of 0.47 — a significant improvement over 0.33 from the baseline model. The confusion matrix confirmed this: the weighted model correctly captured far more true positives (actual subscribers) at the expense of a few extra false positives.

From a business standpoint, this trade-off is acceptable because the cost of contacting a few uninterested customers is far smaller than the loss of missing genuinely interested clients. Therefore, the weighted model would lead to **better marketing outcomes**, enabling the bank to reach a larger share of potential subscribers and improve campaign efficiency.

Part 2 Summary – Overall Reflection

Across Part 2, I explored the full modelling process — from feature selection and encoding to model building, evaluation, and improvement.
Starting with a baseline logistic regression provided a foundation for understanding how the data behaved under a simple, interpretable model. Recognizing class imbalance, I then applied a weighted logistic regression to rebalance the training process, which successfully increased the model's ability to detect real subscribers.

Although the weighted model slightly reduced accuracy, its improvement in recall and F1-score demonstrated that a model's success depends not only on accuracy but also on its ability to capture the right outcomes for the business objective.

In summary, the Weighted Logistic Regression is the more effective and practical model for predicting term-deposit subscriptions. This concludes Part 2 and transitions to Part 3, where I will reflect on lessons learned, model limitations, and recommendations for future improvement.