DSC 789-001 Strategic Capstone Project

By Zahra Sedighi Maman

Data-Driven Bank Marketing: A CRISP-DM Approach to Analyzing Customer Engagement for Term Deposit Subscription

Team Members

Riya Shah

Hemanth Meka

Mamatha Jala

Sharanya Dulam

Introduction:

This project applies the CRISP-DM framework to enhance bank marketing strategies for term deposit subscriptions—a critical driver of customer loyalty and long-term profitability. With traditional marketing methods losing impact, we adopt a data-driven approach using predictive analytics and machine learning to deliver more personalized and effective campaigns.

We begin by understanding the business problem and defining clear objectives, guided by insights from recent research on modeling techniques and key performance indicators. Next, we explore the Bank Marketing dataset through descriptive analysis and visualizations to uncover data patterns and characteristics. In the Data Preparation phase, we clean and transform the dataset, encode categorical variables, handle class imbalance, and select relevant features to ensure high-quality inputs for modeling.

For modeling, we build and compare Logistic Regression and Random Forest classifiers across multiple scenarios involving resampling techniques (SMOTE, ADASYN) and feature selection methods (LASSO, Random Forest importance). Models are evaluated using accuracy, sensitivity, specificity, precision, AUC, and geometric mean to identify the most effective approach.

Finally, we interpret model outcomes, highlight key predictors of customer subscription, and translate these insights into actionable recommendations for targeted marketing strategies. This end-to-end analysis demonstrates how data-driven decision-making can significantly improve campaign performance and customer engagement in the banking sector.

To begin with, the first phase of CRISP – DM framework consists of **Business Understanding**:

Description of the Problem:

This project addresses the challenge of improving the effectiveness of bank marketing campaigns by identifying the factors that influence a customer's decision to subscribe to a term deposit. Through data analysis, the goal is to uncover actionable insights that can help the bank better target potential customers and enhance overall campaign performance.

Background:

The dataset, sourced from Kaggle and the UCI Machine Learning Repository, includes over 41,000 observations from phone-based marketing campaigns conducted by a Portuguese bank. It captures a wide range of variables such as customer demographics (e.g., age, job, marital status, education), loan and credit status, details of previous campaign interactions (e.g., number of contacts, outcomes), and information related to the most recent contact attempt. The target variable indicates whether the customer subscribed to a term deposit. By analyzing these features, the project aims to identify the most influential factors driving customer response and support the development of data-informed, targeted marketing strategies.

Research Questions:

1. Can we predict the variable of interest (response variable) using data analytical techniques?
2. What are the important features in order to predict the response variable?
3. What is the importance of the features?
4. Can we predict the probability of each class (classification problem) for each sample using data analytical techniques?
5. How do the different sampling techniques affect the prediction of the response variable?
6. How do the different feature selection methods affect the prediction of the response variable?
7. How does the prediction performance change for more complex analytical models?
8. What strategy outline do you propose to increase the performance of the specific problem (healthcare, banking, service) you are working on? Or how do you apply the data analysis that you have into practice to make data-driven decisions for the business you are studying?

Research Overview:

Moving further, will explore the research paper where we found out why analytics is important specially in Bank Marketing**.**

1. **Predictive Analytics and Machine Learning in Direct Marketing for Anticipating Bank Term Deposit Subscriptions**
   This study used the Portuguese Bank Marketing dataset and applied machine learning models—including SGD classifier, k-Nearest Neighbours, and Random Forest—to predict term deposit subscriptions. After careful feature engineering and data visualization, the Random Forest model came out on top with 87.5% accuracy**,** a Positive Predictive Value of 87.8%**,** and a Negative Predictive Value of 93%**,** demonstrating its strong predictive power for identifying subscription-ready customers [1].

2. **Bank Direct Marketing Analysis of Asymmetric Information Based on Machine Learning**

This paper tackles the issue of asymmetric information in banking marketing, where customer data may be incomplete or imbalanced. By leveraging machine learning approaches—likely decision trees or logistic regression—it shows that predictive models can effectively infer hidden variables and improve customer segmentation, even under limited data conditions [2].

3. **A Banking Platform to Leverage Data-Driven Marketing with Machine Learning**

   Focusing on real-world implementation, this paper highlights a banking platform that integrates transactional data and machine learning to enable personalized marketing within an online banking environment. While it didn't report specific accuracy metrics, it emphasizes that leveraging customer spending behaviour ("you are what you spend") allows for more targeted offers and enhanced customer engagement, supporting smarter, in-app recommendations [3].

4. **Enhancing Bank Marketing Strategies with Ensemble Learning: Empirical Analysis**

   Using financial dataset analysis and a focus on e-commerce banking, this paper compares Random Forest and SVM models for customer segmentation. The ensemble Random Forest model achieved 92% accuracy versus 87% for SVM and translated these gains into impressive real-world results—a 20% boost in sales and a 30% increase in customer satisfaction [4].

Project Plan:

Here, we are going over the project plan of Bank Marketing.

Phase	Time	Resources	Risk
Business Understanding	Week 1	CRISP -DM framework, stakeholder input	Misalignment of business goals
Data Understanding	Week 1	Python, R, Tableau	Data quality, missing values
Data Preparation	Week 2	Python, R, Feature Selection Techniques	Data leakage or loss of important information
Modelling	Week 3	Python, R, Different Models	Overfitting or underfitting due to improper model selection
Evaluation	Week 4	Feature importance plots using Tableau, Python, R	Inability to implement results

Deployment	Week 5	Power Point, word document and visualization tools	Inability to implement results

CRISP - DM Questions:

And the last by least the important part of business understanding, the CRISP- DM questions which are as follows:

1. Describe the type of problem?

This is a binary classification problem because the goal is to predict whether a customer will subscribe to a term deposit or not, based on their personal, financial, and campaign-related information. Since the outcome has two possible values— 'yes' or 'no'—classification models like Logistic Regression, Random Forest, or Decision Trees are typically used to solve this type of problem.

2. Document Technical Goal?

To improve term deposit marketing campaigns, this project focuses on building a binary classification model to predict whether customers will subscribe. Models such as Logistic Regression, Decision Trees, Random Forest, and SVM are suitable, but Random Forest is recommended due to its high accuracy and ability to handle complex data. The required time to execute models like Random Forest is moderate—typically a few minutes—making it practical for this dataset size. Since this is for strategic decision-making, the analysis speed needs to be timely but not real-time, allowing enough flexibility for marketing planning.

3. What are the computing and data storage needs and/or requirements?

For this project, the computing and data storage needs are moderate**.** The analysis can be comfortably handled on a single machine with standard processing power since the dataset size (around 41,000 records) is manageable. However, for larger datasets, frequent model updates, or collaborative work, using a server or cloud platform could offer better scalability, storage, and processing speed.

4. What are the success criteria?

The success of this project will be measured by the model's evaluation metrics, primarily focusing on accuracy, precision, recall, and AUC to ensure reliable predictions. Additionally, subjective criteria such as the model's ease of interpretation, alignment with business goals, and its ability to support decision-making will also be important. While deployment success is not the immediate focus, it can be considered part of the broader Data Mining (DM) process if the goal is to integrate the model into regular marketing operations.

The second phase of CRISP – DM Framework is about **Data Understanding** where we first went over the dataset overview.

Dataset Overview:

The Bank Marketing Dataset is about phone marketing campaign for Portuguese banking institution, where the goal is to ask customers to subscribe for a bank term deposit. The dataset includes various client data, last contact of the current campaign, social and economic attributes, and targeted variables. The dataset is extracted from Kaggle [5].

No. of Observations: 41188 Observations
No. of Variables: 21 Variables

Types of the dataset:

Variable Name	Type	Description
Age	Integer	Age of the Customer
Job	Categorical	Type of Job
Marital	Categorical	Marital Status
Education	Categorical	Education Level
Default	Categorical	Has credit in default?
Housing	Categorical	Has housing loan?
Loan	Categorical	Has personal loan?
Contact	Categorical	Contact Communication type
Month	Categorical	Last contact month of year
Day_of_week	Categorical	Last contact day of the week
Duration	Numeric	Last contact duration, in seconds
Campaign	Numeric	Number of contacts performed during this campaign and for this client
Pdays	Numeric	Number of days that passed by after the client was last contacted from a previous campaign
Pervious	Numeric	Number of contacts performed before this campaign and for this client
Poutcome	Categorical	Outcome of the previous marketing Campaign
Emp.var.rate	Numeric	Employment variation rate
Cons.price.idx	Numeric	Consumer price index
Cons.conf.idx	Numeric	Consumer confidence index
Euribor3m	Numeric	Euribor 3-month rate
Nr. employed	Numeric	Number of employees
Y	Binary	Has the client subscribed a term deposit?

Descriptive Statistics for each Numerical Variable:

Moving further we have the descriptive statistics for numerical variables.

Index/variable	Age	Duration	Campaign	Pdays	Previous
Count	41188	41188	41188	41188	41188
Mean	40.02	258.29	2.57	962.48	0.17
Std	10.42	259.28	2.77	186.91	0.49
Min	17	0	1	0	0
25%	32	102	1	999	0
50%	38	180	2	999	0
75%	47	319	3	999	0
Max	98	4918	56	999	7

Notable findings:

- Age: Most clients are around 40, with a wide age range.
- Duration: Call lengths vary greatly, averaging about 4 minutes.
- Campaign: Clients were typically contacted 1–3 times; some much more.
- Pdays: Most clients had no prior contact (pdays = 999).
- Previous: Few clients had previous campaign interactions.

Value Counts for each Categorical Variables:

Furthermore, we are looking into the value counts of the categorical variables.

Variable	Values	Count
	admin.	10422
	blue-collar	9254
	technician	6743
	services	3969
	management	2924
Job	retired	1720
	entrepreneur	1456
	self-employed	1421
	housemaid	1060
	unemployed	1014
	student	875
	unknown	330
	married	24928
Marital	single	11568
	divorced	4612
	unknown	80
	university.degree	12168
	high.school	9515
	basic.9y	6045
Education	professional.course	5243
	basic.4y	4176
	basic.6y	2292
	unknown	1731
	illiterate	18
	no	32588
Default	unknown	8597
	yes	3
	yes	21576
Housing	no	18622
	unknown	990
	no	33950
Loan	yes	6248

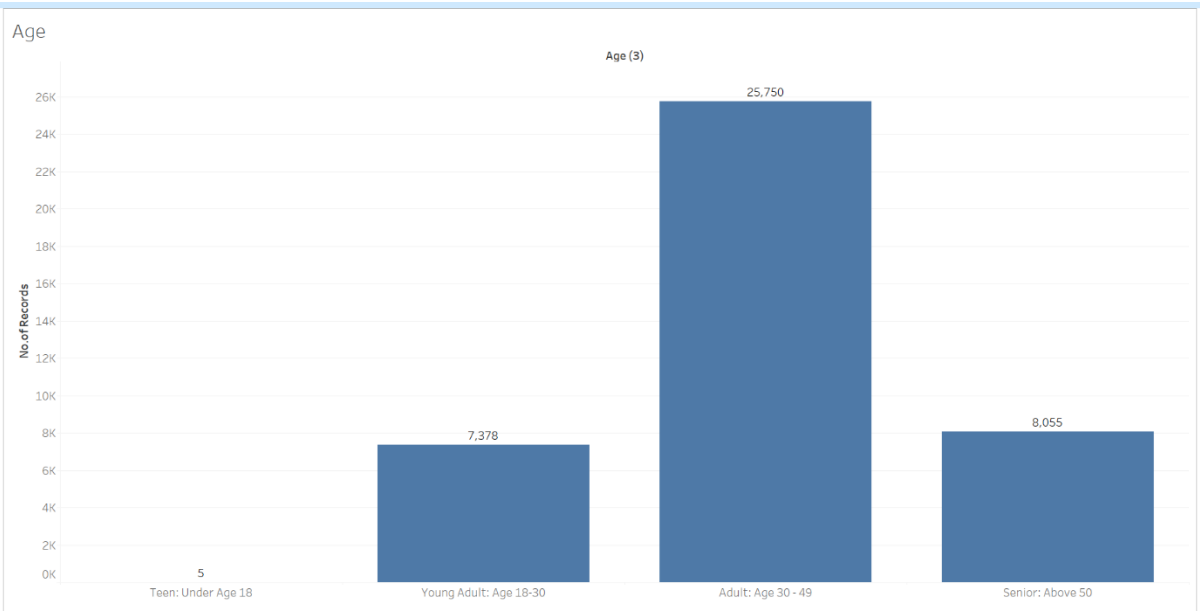
	unknown	990
	cellular	26144
Contact	telephone	15044
	may	13769
	jul	7174
	aug	6178
	jun	5318
	nov	4101
Month	apr	2632
	oct	718
	sep	570
	mar	546
	dec	182
	Thu	8623
	Mon	8514
Day of the Week	Wed	8134
	Tue	8090
	Fri	7827
	non-existent	35563
Poutcome	failure	4252
	success	1373
y	no	36548
	yes	4640

Moreover, we performed EDA to look into the descriptive statistics of the dataset.

Performing Exploratory Data Analysis using Tableau

Visualization for each Variable:

Age:

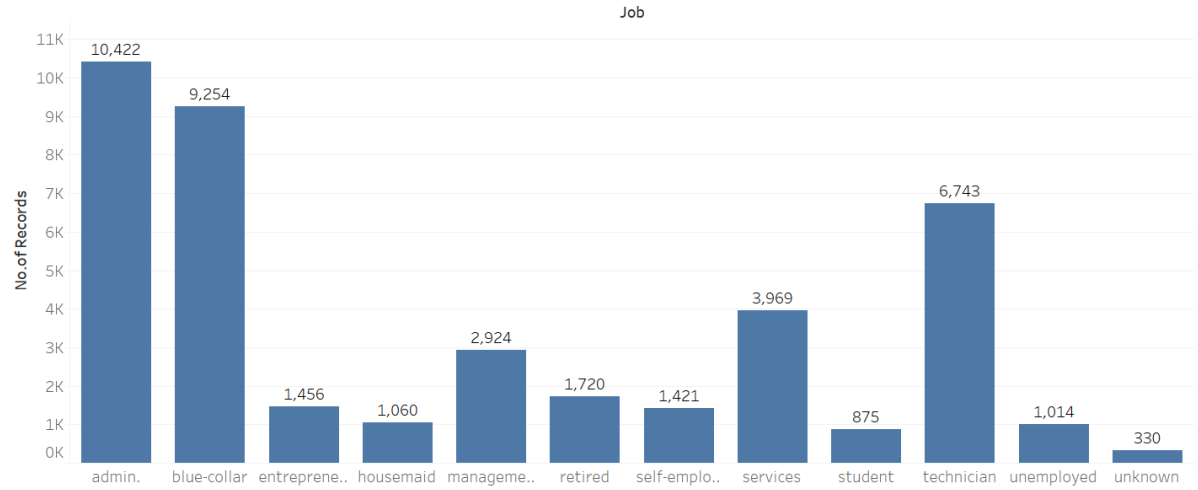


Insights:

The majority of clients fall within the 30 to 49 age group, forming the largest segment of the dataset. Young adults (18–30) and seniors (50 and above) are represented in smaller numbers, with teenagers under 18 being almost non-existent. This indicates the bank's marketing campaign primarily targeted individuals in their prime working years, likely due to their stable incomes and financial planning needs.

Job:

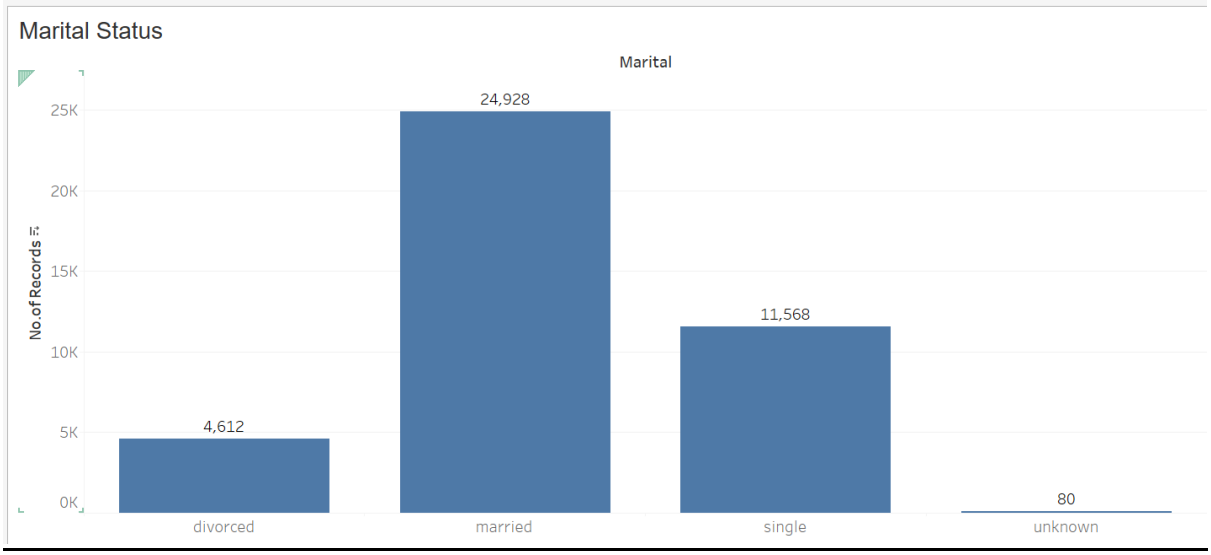
Job Type



Insights:

Most clients work in **administrative**, **blue-collar**, or **technical** roles. Other occupations like **services**, **management**, and **retired** are moderately represented, while roles such as **students** and the **unemployed** appear less frequently. This suggests the campaign focused mainly on working professionals.

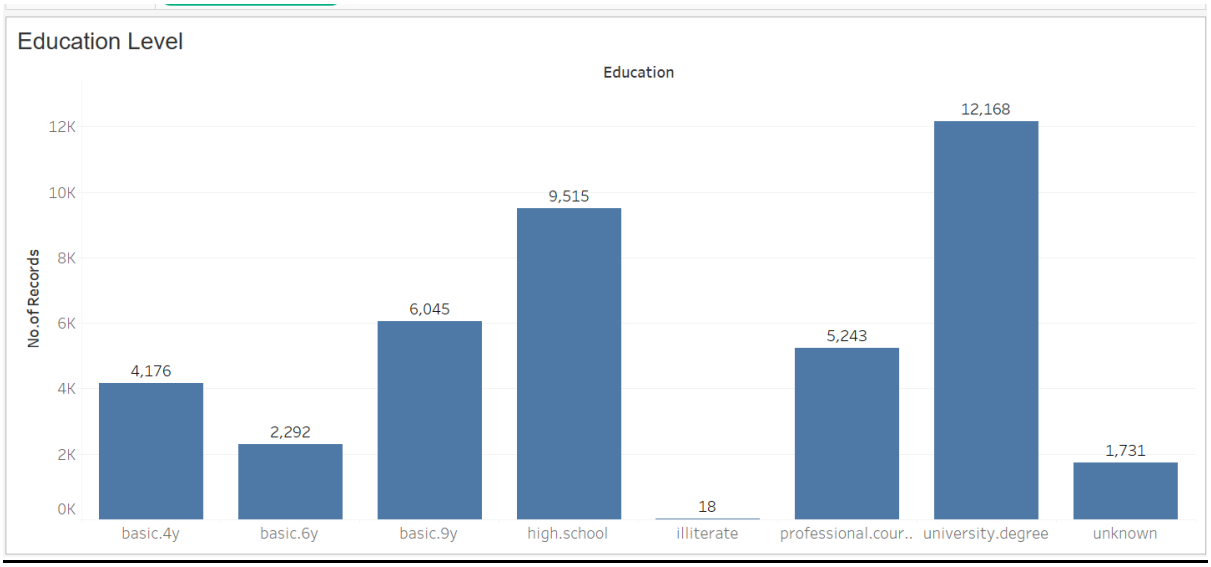
Marital Status:



Insights:

The majority of clients are **married**, followed by **single** individuals, with a smaller portion are **divorced**, and only a few records have unknown marital status. This suggests the campaign primarily engaged with clients in stable household situations.

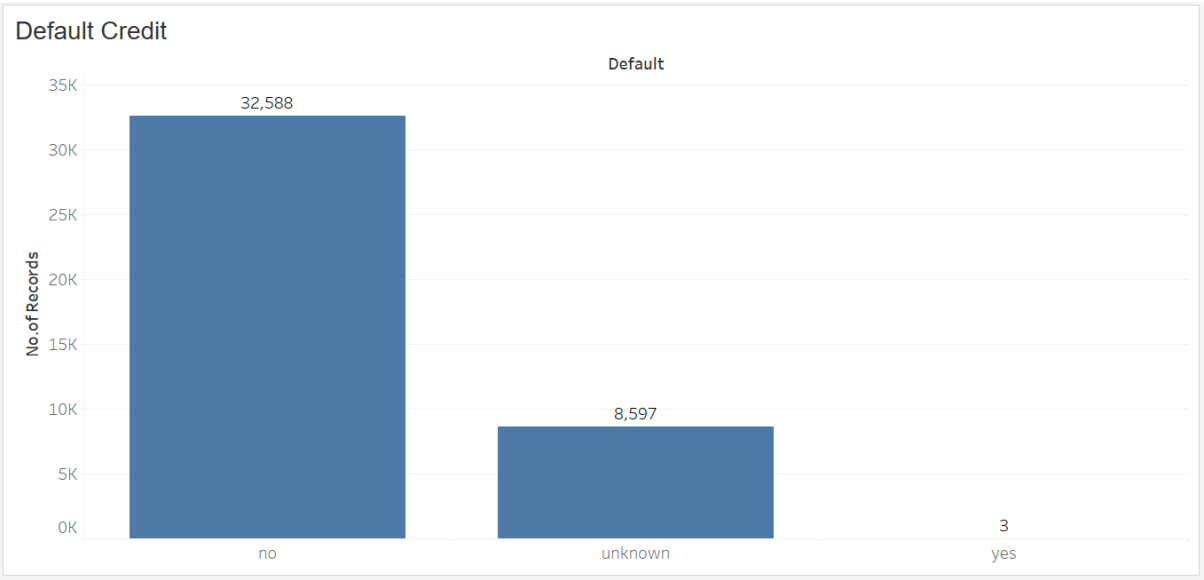
Education:



Insights:

The chart indicates that "university. Degree" is the most common education level, followed closely by "high. School." There's also a considerable number of individuals with "basic.9y" education. The categories "basic.4y" and "professional. Course" also represent significant portions of the data. Notably, the "illiterate" category is extremely rare, and there's a small segment of "unknown" education levels. This distribution provides a good overview of the educational background within the dataset.

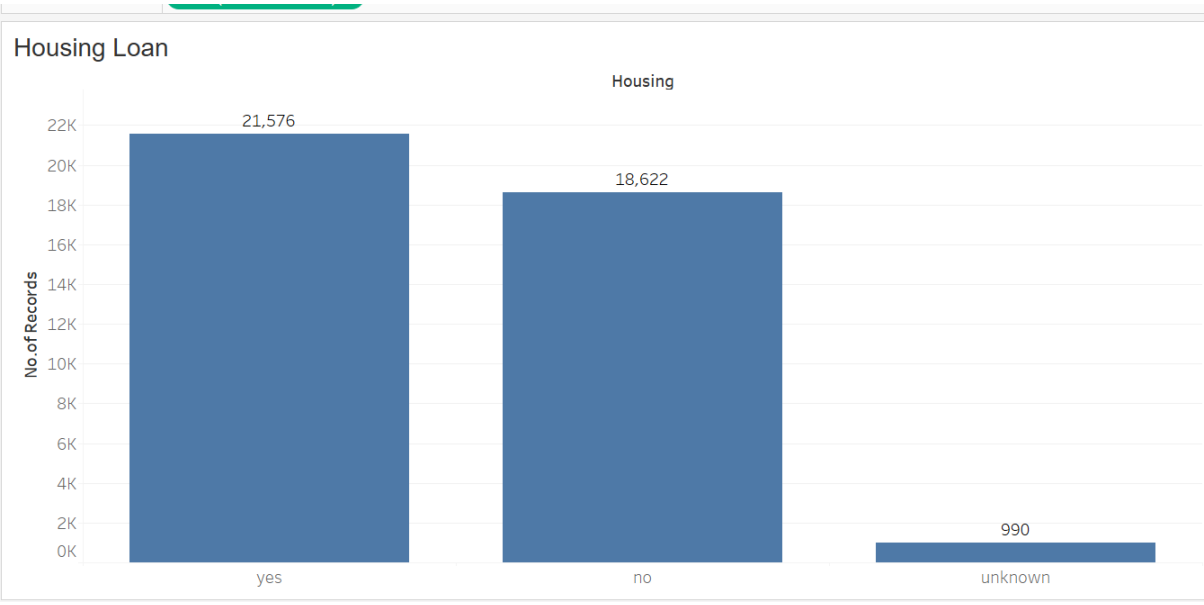
Default Credit:



Insights:

The graph reveals a highly imbalanced distribution. A vast majority of records, over 32,500, indicate "no" default credit. There's a notable segment of "unknown" records (around 8,500), but critically, only a negligible number of records (3) are associated with a "yes" for default credit. This extreme imbalance suggests that default credit is an infrequent occurrence in this dataset.

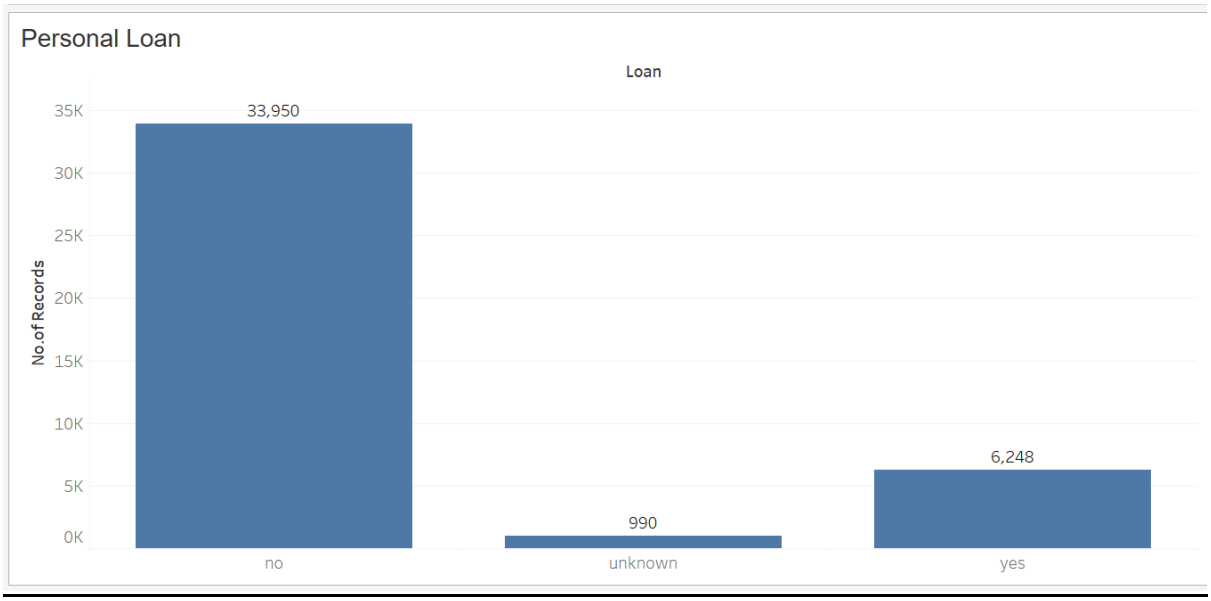
Housing Loan:



Insights:

The graph shows a fairly even split between individuals who have a housing loan ("yes," over 21,500 records) and those who do not ("no," over 18,600 records). A smaller proportion of records, under 1,000, have an "unknown" status regarding housing loans. This balanced distribution suggests that housing loan status is a common and varied attribute within the dataset.

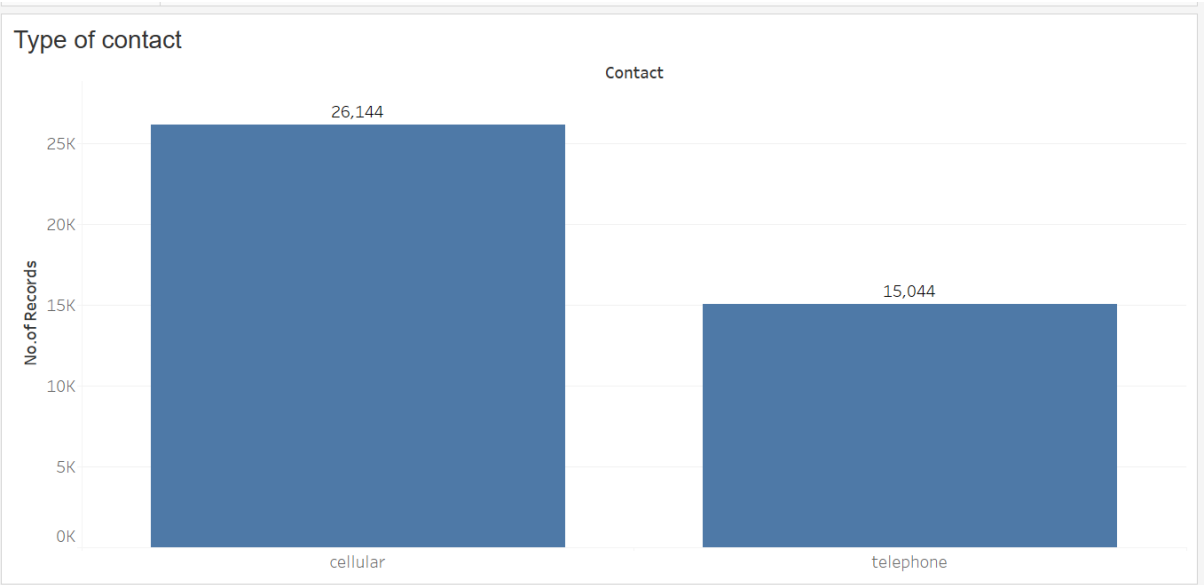
Personal Loan:



Insights:

This chart clearly indicates that the vast majority of records (nearly 34,000) show "no" personal loan. While there's a smaller but still significant group with "yes" to a personal loan (over 6,200 records), the proportion of "unknown" records is minimal. This suggests that personal loans are not as common as housing loans in this dataset.

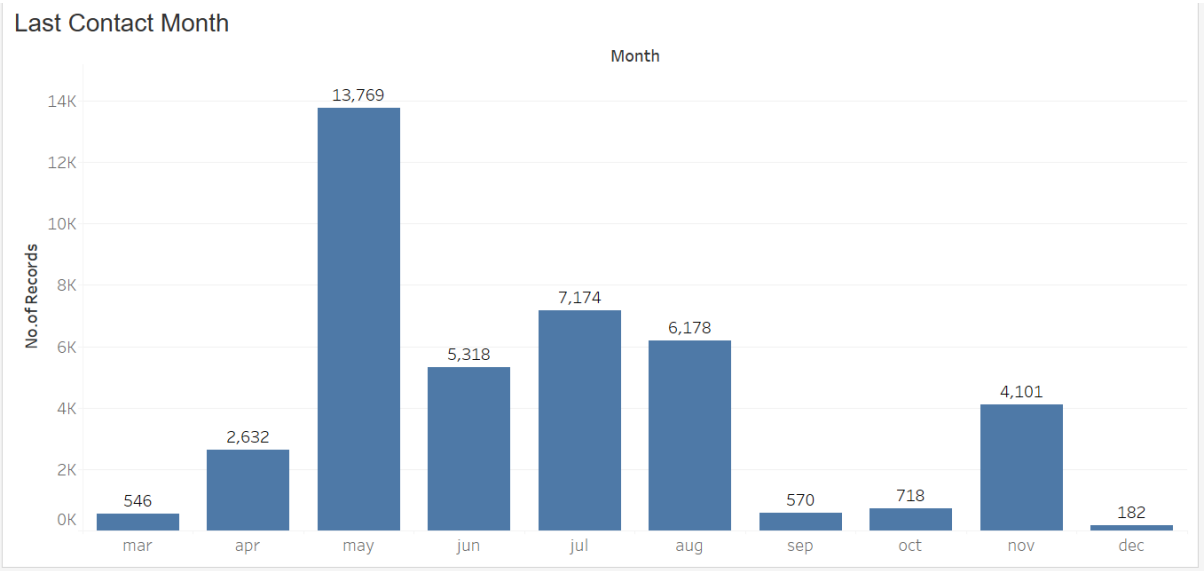
Contact Type:



Insights:

This chart demonstrates a clear preference or higher volume of interactions via "cellular" contact, which accounts for over 26,000 records. "Telephone" contacts, while still substantial at just over 15,000 records, are notably less frequent. This suggests that cellular communication is the primary mode of contact in our dataset.

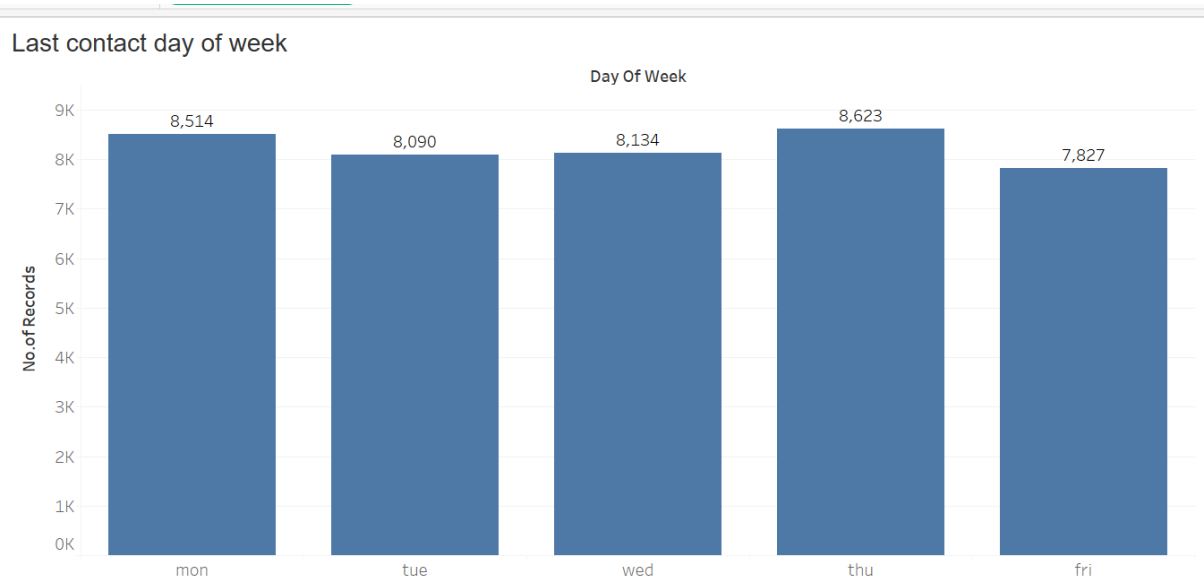
Contact Month:



Insights:

The graph highlights significant seasonality in contact activity. May clearly stands out as the peak month for contacts, with over 13,700 records. July and August also show strong contact volumes, while the activity significantly dips in September, October, and December, suggesting a concentrated effort or higher engagement during specific periods of the year.

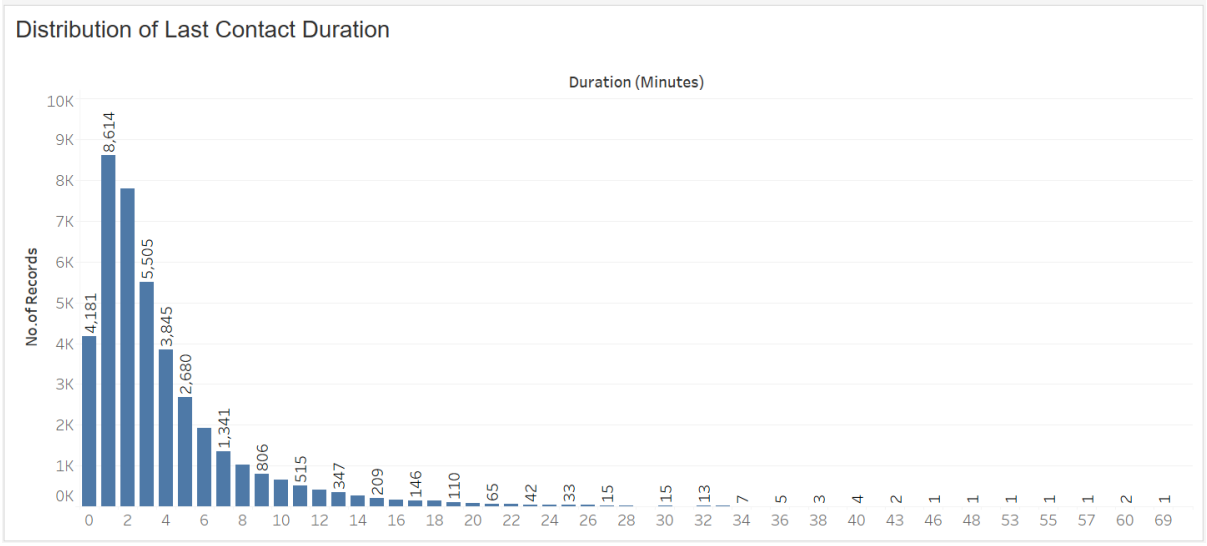
Contact day of the week:



Insights:

The graph demonstrates a relatively uniform distribution of contacts across the working week. While Thursday has a slightly higher number of contacts (over 8,600), and Friday is marginally lower, there isn't a dramatic variation between Monday, Tuesday, and Wednesday. This indicates that contact efforts are spread fairly evenly throughout the typical business week.

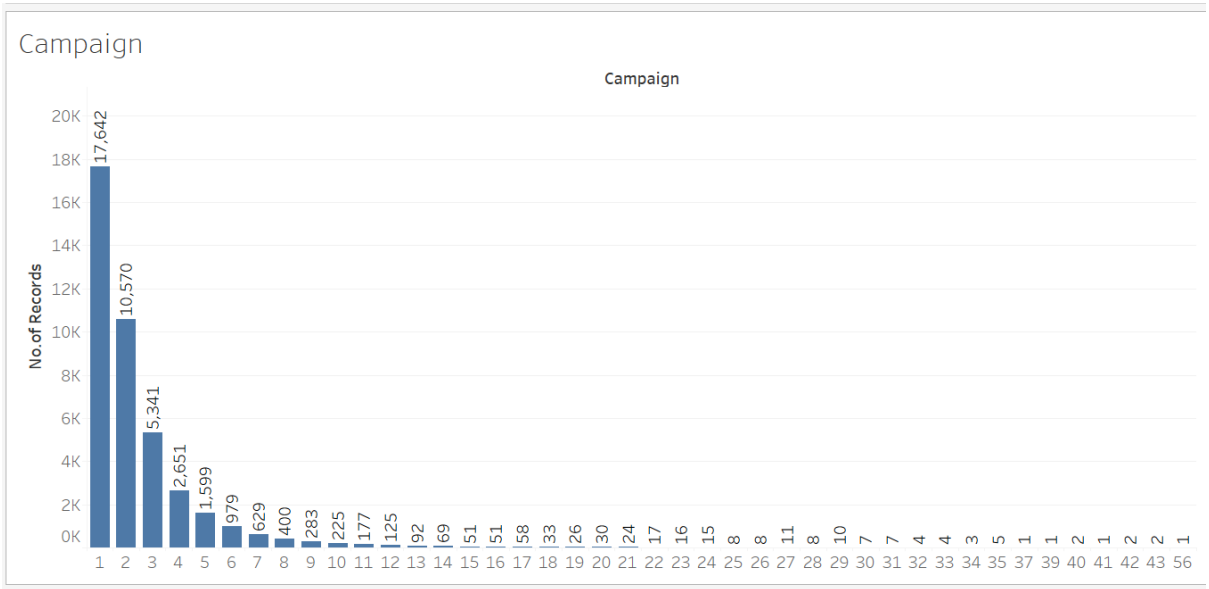
Duration:



Insights:

The graph reveals that most contacts are very brief, heavily concentrated within the 0 to 6-minute range, with a sharp decline in records for longer durations, indicating that prolonged customer engagement is rare. Despite the prevalence of short calls, it's crucial to note that longer durations are strongly correlated with the client subscribing to a term deposit. This suggests that while brief interactions are common, the fewer, extended conversations are significantly more valuable for achieving a 'yes' for term deposit subscriptions.

Campaign:

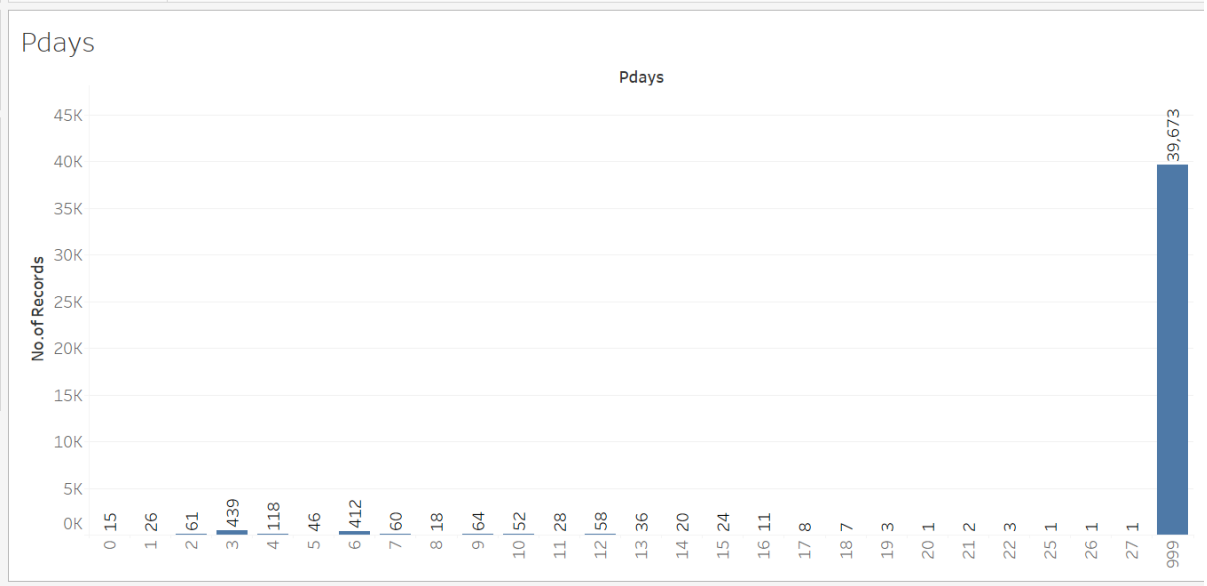


Insights:

The graph clearly indicates a highly skewed distribution where a significant majority of records (over 17,000) are associated with Campaign '1'. The frequency drops off sharply for subsequent campaigns, with only a small number of records beyond Campaign '7'. This

suggests that most of our activity or data points are concentrated in the initial campaigns, with diminishing returns or participation in later ones.

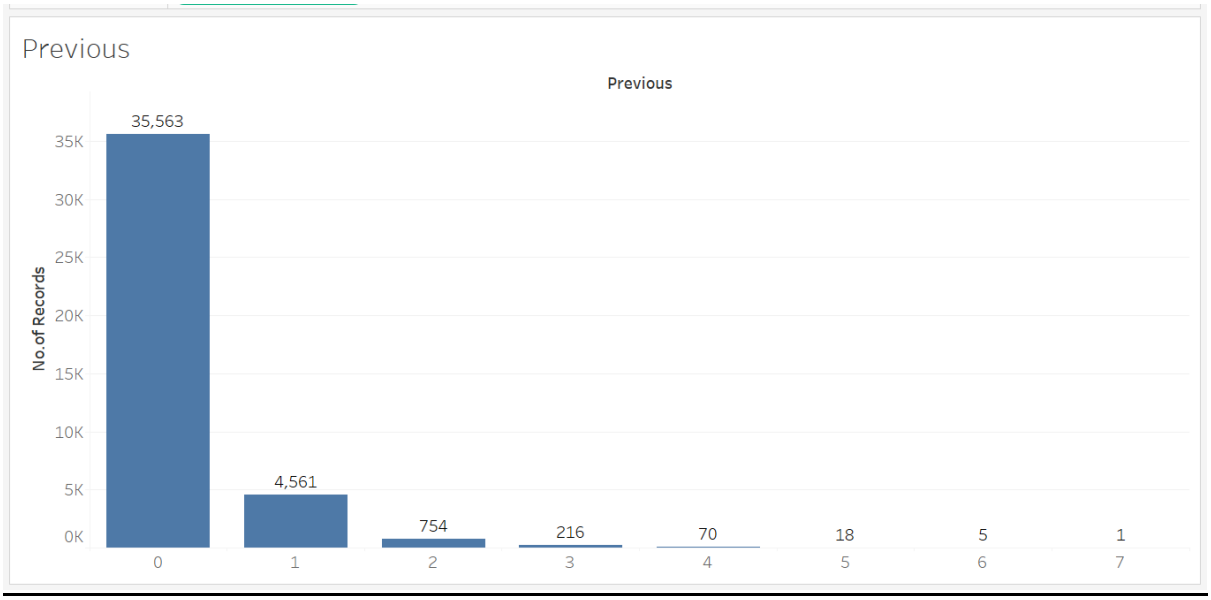
Pdays:



Insights:

An overwhelming majority of records are marked with '999', indicating that the 'Pdays' (days since last contact) value is not applicable or that no previous contact was made. Instances where previous contact occurred represent a small fraction of the data, with a spread distribution of days.

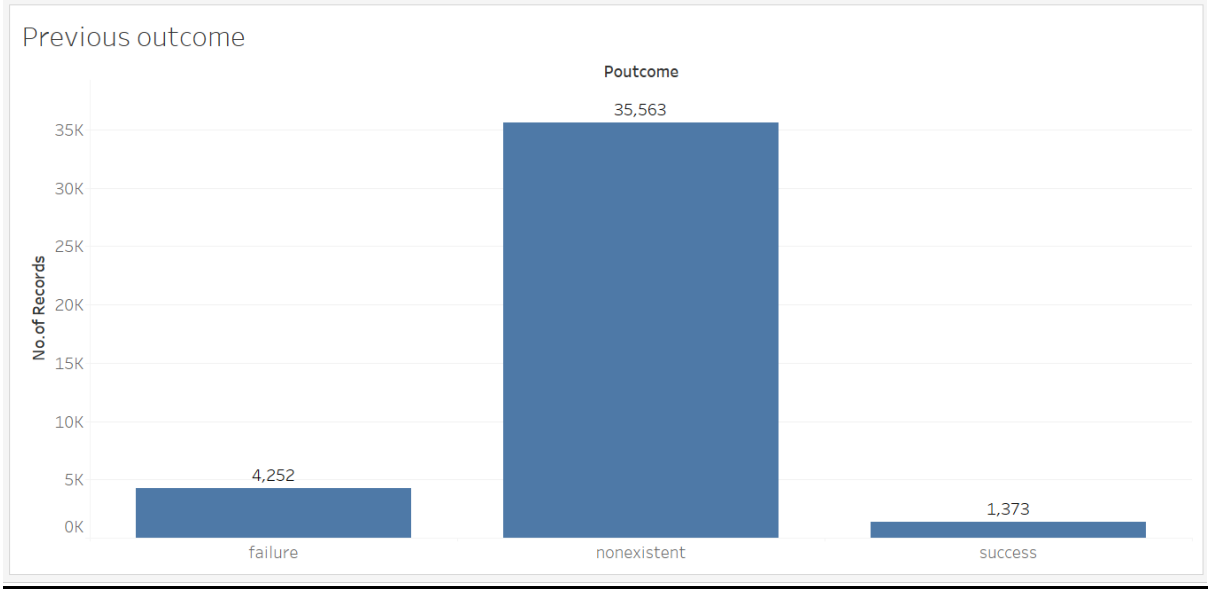
Previous:



Insights:

The vast majority of records indicate '0' previous contacts, suggesting no prior interaction for most clients. When previous contacts did occur, the count significantly decreases with each additional contact, with very few records showing 5 or more interactions.

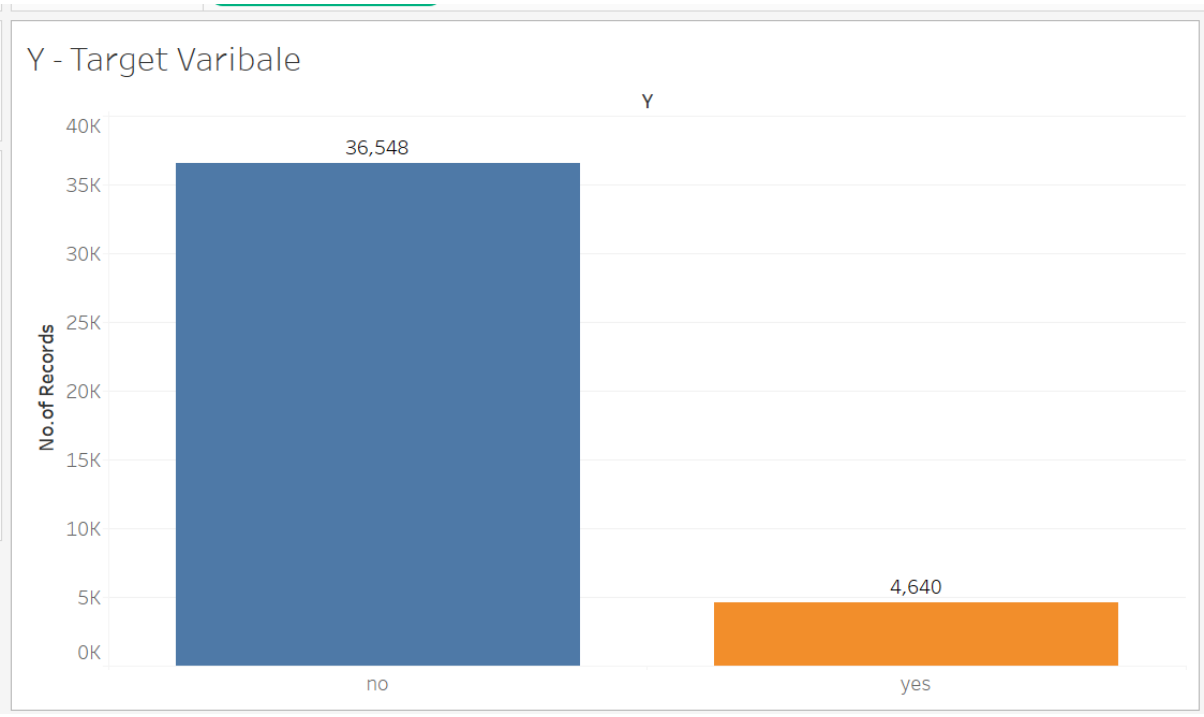
Poutcome:



Insights:

The 'non-existent' category accounts for the largest proportion of previous outcomes, aligning with the "Previous" graph and reinforcing the lack of defined previous outcomes for a substantial number of records. Among those with a previous outcome, 'failure' is more prevalent than 'success', though both are significantly less frequent than 'non-existent'.

Y-Target:



Insights:

The "Y - Target Variable" shows an imbalanced distribution, where the 'no' category (client has not subscribed to a term deposit) accounts for a significantly larger number of records compared to the 'yes' category (client has subscribed). This imbalance indicates that the target variable is heavily weighted towards a negative outcome, which is a crucial consideration for any modelling or analysis.

We also dive deeper into the bivariate variables considering the target variable i.e., Client Subscribed term deposit.

Based on the patterns identified during EDA, including the strong class imbalance and outliers in call duration and age, we moved to the **Data Preparation phase**. This step included importing the dataset into Google Colab, cleaning and transforming the data, encoding categorical variables, and applying techniques like SMOTE and ADASYN to balance the target classes.

Fetching the dataset:

We began by importing the "Bank Marketing – separated columns.csv" dataset into Google Colab using the pandas library. This dataset, sourced from Kaggle and the UCI Machine Learning Repository, contains 21 variables and over 41,000 observations. An initial exploration provided insights into the structure of both numerical and categorical features.

```
# Loading the dataset
df = pd.read_csv("/content/Bank Marketing-separeted columns.csv")
```

Dropping the variables that are not required:

Next, we removed variables that were deemed less relevant to predicting term deposit subscriptions. These variables represent macroeconomic indicators and do not directly describe customer behaviour.

The following columns were dropped:

- **emp.var.rate –** Employment variation rate (quarterly indicator)
- **cons.price.idx –** Consumer Price Index (monthly indicator)
- **cons.conf.idx –** Consumer Confidence Index (monthly indicator)
- **euribor3m –** Euribor 3-month rate (daily indicator)
- **nr.employed –** Number of employees (quarterly indicator)

Removing these variables reduces noise and helps the model focus on meaningful customer-related features.

```
# Removing Redudant variables.
data = bank.drop(['emp.var.rate', 'cons.price.idx', 'cons.conf.idx', 'euribor3m', 'nr.employed'], axis=1)
```

After Dropping the variables this is how the dataset looks.

	age	job	marital	education	default	housing	loan	contact	month	day_of_week	duration	campaign	pdays	previous	poutcome	y
0	56	housemaid	married	basic.4y	no	no	no	telephone	may	mon	261	1	999	0	nonexistent	no
1	57	services	married	high.school	unknown	no	no	telephone	may	mon	149	1	999	0	nonexistent	no
2	37	services	married	high.school	no	yes	no	telephone	may	mon	226	1	999	0	nonexistent	no
3	40	admin.	married	basic.6y	no	no	no	telephone	may	mon	151	1	999	0	nonexistent	no
4	56	services	married	high.school	no	no	yes	telephone	may	mon	307	1	999	0	nonexistent	no
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
41183	73	retired	married	professional.course	no	yes	no	cellular	nov	fri	334	1	999	0	nonexistent	yes
41184	46	blue-collar	married	professional.course	no	no	no	cellular	nov	fri	383	1	999	0	nonexistent	no
41185	56	retired	married	university.degree	no	yes	no	cellular	nov	fri	189	2	999	0	nonexistent	no
41186	44	technician	married	professional.course	no	no	no	cellular	nov	fri	442	1	999	0	nonexistent	yes
41187	74	retired	married	professional.course	no	yes	no	cellular	nov	fri	239	3	999	1	failure	no

41188 rows × 16 columns

Checking for the Missing Values:

We examined the dataset using .info () and .describe() functions to detect any missing values. The results confirmed that the dataset contained **no missing entries**, allowing us to proceed without the need for imputation.

```
# Checking for the missing values
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41188 entries, 0 to 41187
Data columns (total 16 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   age          41188 non-null  int64
 1   job          41188 non-null  object
 2   marital      41188 non-null  object
 3   education    41188 non-null  object
 4   default      41188 non-null  object
 5   housing      41188 non-null  object
 6   loan         41188 non-null  object
 7   contact      41188 non-null  object
 8   month        41188 non-null  object
 9   day_of_week  41188 non-null  object
 10  duration     41188 non-null  int64
 11  campaign     41188 non-null  int64
 12  pdays        41188 non-null  int64
 13  previous     41188 non-null  int64
 14  poutcome     41188 non-null  object
 15  y            41188 non-null  object
dtypes: int64(5), object(11)
memory usage: 5.0+ MB
```

```
data.describe()
```

	age	duration	campaign	pdays	previous
count	41188.00000	41188.000000	41188.000000	41188.000000	41188.000000
mean	40.02406	258.285010	2.567593	962.475454	0.172963
std	10.42125	259.279249	2.770014	186.910907	0.494901
min	17.00000	0.000000	1.000000	0.000000	0.000000
25%	32.00000	102.000000	1.000000	999.000000	0.000000
50%	38.00000	180.000000	2.000000	999.000000	0.000000
75%	47.00000	319.000000	3.000000	999.000000	0.000000
max	98.00000	4918.000000	56.000000	999.000000	7.000000

Outlier and Noise Detection

We separated categorical and numerical variables, then addressed outliers—particularly in age and call duration—using the Interquartile Range (IQR) method (values beyond Q1 – 1.5×IQR or Q3 + 1.5×IQR). Instead of removing these outliers, we applied binning and capping to reduce their impact while preserving data integrity:

- **Age:** <18, 18–24, 25–34, 35–50, 50–60, 60+
- **Call Duration (min):** 0, 1–2, 3–5, 6–10, 11–20, 21+

This approach effectively reduces skewness without losing valuable information.

```
# Numeric columns
columns = ['age', 'duration']

# Dictionary to store number of outliers
outlier_counts = {}

for col in columns:
    Q1 = numericData[col].quantile(0.25)
    Q3 = numericData[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # Count outliers
    outliers = numericData[(numericData[col] < lower_bound) | (numericData[col] > upper_bound)]
    outlier_counts[col] = outliers.shape[0]

# Display the number of outliers per column
print(outlier_counts)
```

```
{'age': 469, 'duration': 2963}
```

```
# Handling Outliers and Noise
# Binning Age with the given ranges
bins_age = [0, 18, 25, 35, 50, 60, 100]
abels_age = ['<18', '18-24', '25-34', '35-50', '50-60', '60+']
numericData['age_bin'] = pd.cut(data['age'], bins=bins_age, labels=labels_age, right=False)


# Binning Call Duration
bins_duration = [0, 60, 120, 300, 600, 1200, data['duration'].max()]
labels_duration = ['0 min', '1-2 min', '3-5 min', '6-10 min', '11-20 min', '21+ min']
numericData['duration_bin'] = pd.cut(data['duration'], bins=bins_duration, labels=labels_duration, right=False)

# Dropping the age and duration
numeric_Data = numericData.drop(['age', 'duration'], axis=1)
```

Combining categorical data with numeric data

The categorical features were then combined with the transformed numeric dataset (with binned age and call duration variables) to create a single unified dataset, forming the foundation for encoding.

```
data_1 = numeric_Data.join(categoricalData)
```

```
[ ] data_1
```

	campaign	pdays	previous	age_bin	duration_bin	job	marital	education	default	housing	loan	contact	month	day_of_week	poutcome	y
0	1	999	0	50-60	3-5 min	housemaid	married	basic.4y	no	no	no	telephone	may	mon	nonexistent	no
1	1	999	0	50-60	3-5 min	services	married	high.school	unknown	no	no	telephone	may	mon	nonexistent	no
2	1	999	0	35-50	3-5 min	services	married	high.school	no	yes	no	telephone	may	mon	nonexistent	no
3	1	999	0	35-50	3-5 min	admin.	married	basic.6y	no	no	no	telephone	may	mon	nonexistent	no
4	1	999	0	50-60	6-10 min	services	married	high.school	no	no	yes	telephone	may	mon	nonexistent	no
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
41183	1	999	0	60+	6-10 min	retired	married	professional.course	no	yes	no	cellular	nov	fri	nonexistent	yes
41184	1	999	0	35-50	6-10 min	blue-collar	married	professional.course	no	no	no	cellular	nov	fri	nonexistent	no
41185	2	999	0	50-60	3-5 min	retired	married	university.degree	no	yes	no	cellular	nov	fri	nonexistent	no
41186	1	999	0	35-50	6-10 min	technician	married	professional.course	no	no	no	cellular	nov	fri	nonexistent	yes
41187	3	999	1	60+	3-5 min	retired	married	professional.course	no	yes	no	cellular	nov	fri	failure	no

41188 rows × 16 columns

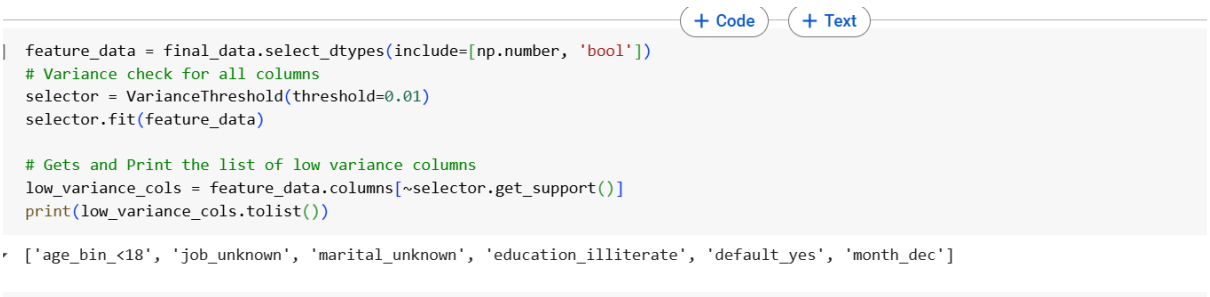
Encoding Categorical Variables

To prepare the data for modelling, categorical variables were encoded using one-hot encoding, and the target variable (client subscribed term deposit) was converted into a binary format (yes → 1, no → 0). The numerical, categorical, and target variables were then combined using pd.concat(), resulting in a clean and structured dataset (final_data) for the subsequent CRISP-DM phases.

```
# Extract categorical column names (excluding 'y')
categorical_cols = data_1.select_dtypes(include=['object']).columns.drop('y')

# One-hot encode categorical columns
categoriesEncoded = pd.get_dummies(data_1[categorical_cols], drop_first=False)

# Extract numerical columns (excluding categorical columns and 'y')
numerical_cols = data_1.drop(columns=list(categorical_cols) + ['y'])

# Combine numerical and encoded categorical columns
features = pd.concat([numerical_cols, categoriesEncoded], axis=1)

# Now encode the target variable 'y' and add it back
target = data_1['y'].replace({'yes': 1, 'no': 0})

# Combine features and target
final_data = pd.concat([features, target], axis=1)

final_data
```

Addressing Near-Zero Variance Variables

We applied scikit-learn's Variance Threshold to detect features with very low variance, which typically contribute little to predictive performance and may introduce noise. Since this method can be less reliable for categorical variables, we carefully reviewed the results and excluded low-variance categories—such as job_unknown, marital_unknown, education_illiterate, default_yes, and month_dec—to improve model efficiency and reduce redundancy.



After preparing a clean and structured dataset through encoding, feature selection, and balancing, we advanced to the **Modeling phase**. Here, we split the data into training and testing sets and applied classification models such as Logistic Regression and Random Forest to evaluate their ability to predict term deposit subscriptions.

Data Preparation and Splitting:

First, the dataset was split into features (X) and target variable (y), where the target y indicates whether the client subscribed to a term deposit.

The data was then divided into training and testing sets with a 70%-30% split using stratified sampling (via random_state=42 to ensure reproducibility). The training set contains approximately 70% of the data to train the models, while the test set contains 30% of the data to evaluate model performance on unseen data.

Class distributions in the target variable were checked for both training and test sets. This is important to understand the extent of class imbalance, which is common in marketing datasets where the positive class (subscribed) is often a minority.

```
[ ]  X = bank_data.drop('y', axis=1)
     y = bank_data['y']
```

Split into train and test sets

```
[ ]  from sklearn.model_selection import train_test_split

     X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
print(X_train.shape)
print(X_test.shape)
```

```
(28831, 62)
(12357, 62)
```

Handling Class Imbalance Using SMOTE and ADASYN

The training data showed class imbalance, which can negatively impact model performance, especially for the minority class. To address this, two synthetic oversampling techniques were applied:

- **SMOTE (Synthetic Minority Oversampling Technique):** SMOTE balances the dataset by generating synthetic samples for the minority class. In our case, the minority class (subscribers) increased from **3,251** to **25,580**, matching the majority class (non-subscribers), as shown in the output of the resampled training data. This interpolation-based approach ensures a more balanced class distribution, improving the model's ability to learn patterns from both classes.

```
Before SMOTE:
y
0    25580
1     3251
Name: count, dtype: int64

After SMOTE:
y
0    25580
1    25580
Name: count, dtype: int64
```

- **ADASYN (Adaptive Synthetic Sampling):** Similar to SMOTE, ADASYN generates synthetic samples for the minority class but focuses on harder-to-learn cases by adaptively creating more data where needed. In our case, subscribers increased from 3,251 to 25,447, nearly matching the 25,580 non-subscribers. This adaptive balancing enhances the model's ability to detect the minority class during prediction.

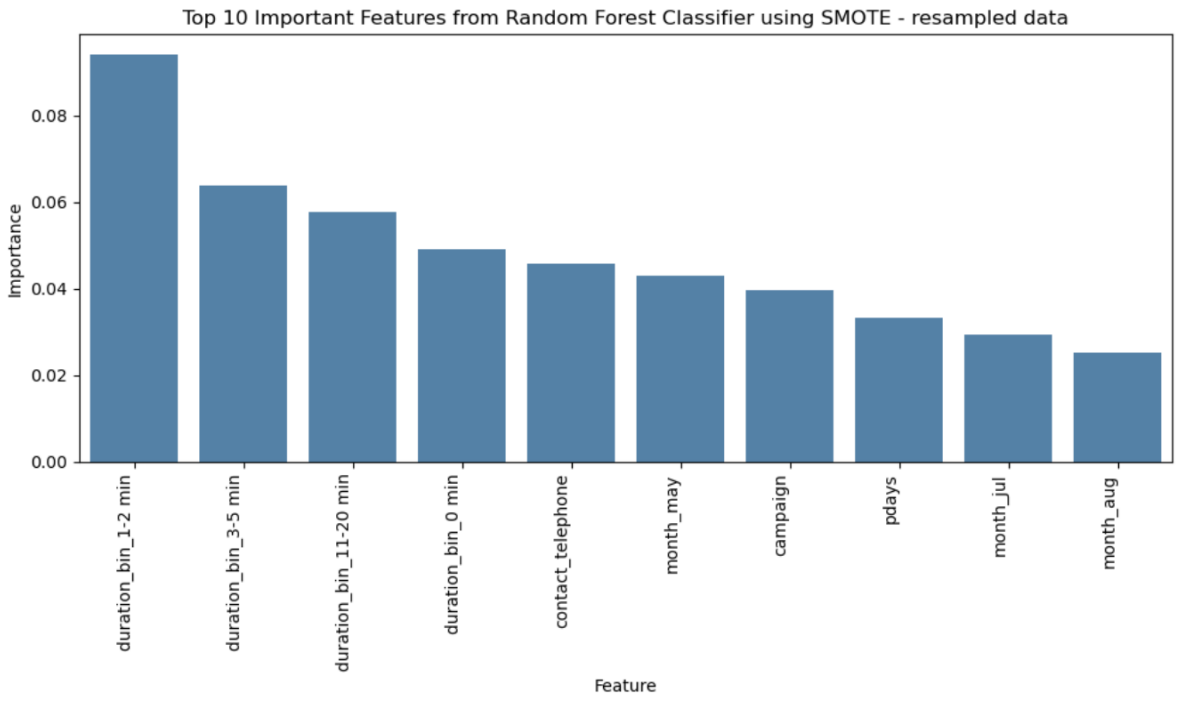
```
Before ADASYN:
y
0    25580
1     3251
Name: count, dtype: int64

After ADASYN:
y
0    25580
1    25447
Name: count, dtype: int64
```

The class distributions before and after applying SMOTE and ADASYN confirmed that the minority class was successfully balanced in the training data, which helps improve the model's ability to detect the minority class during prediction.

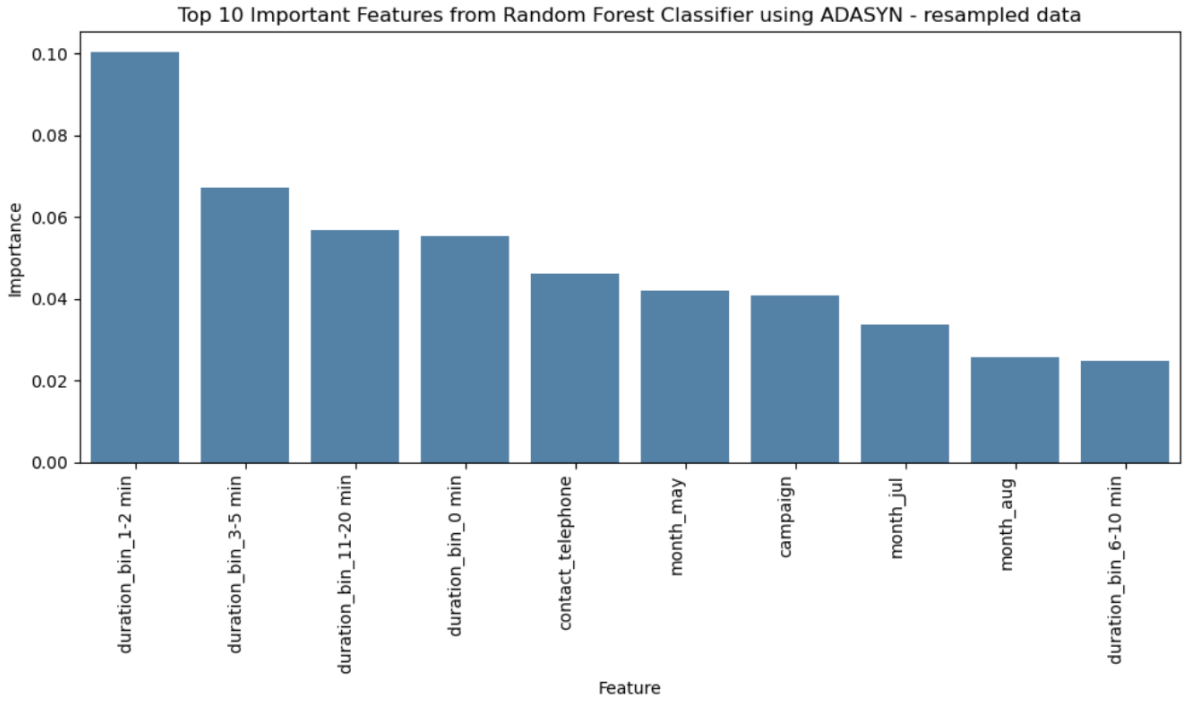
Feature Importance Analysis with Different Models

Feature selection was performed using three different methods to identify the most important predictors:

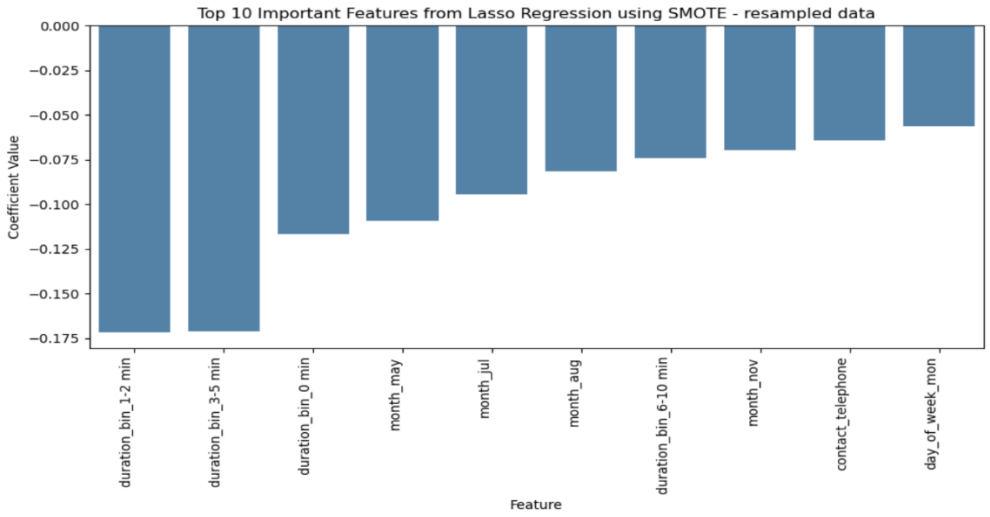
1. Random Forest with SMOTE Balanced Data: A Random Forest algorithm was applied as a feature selection method on the SMOTE-resampled dataset to identify the most influential variables. The feature_importances_ attribute was used to rank features based on their contribution to the model's decision splits. Call duration-related variables, such as duration_bin_1-2 min, duration_bin_3-5 min, and duration_bin_11-20 min, emerged as the most important features.



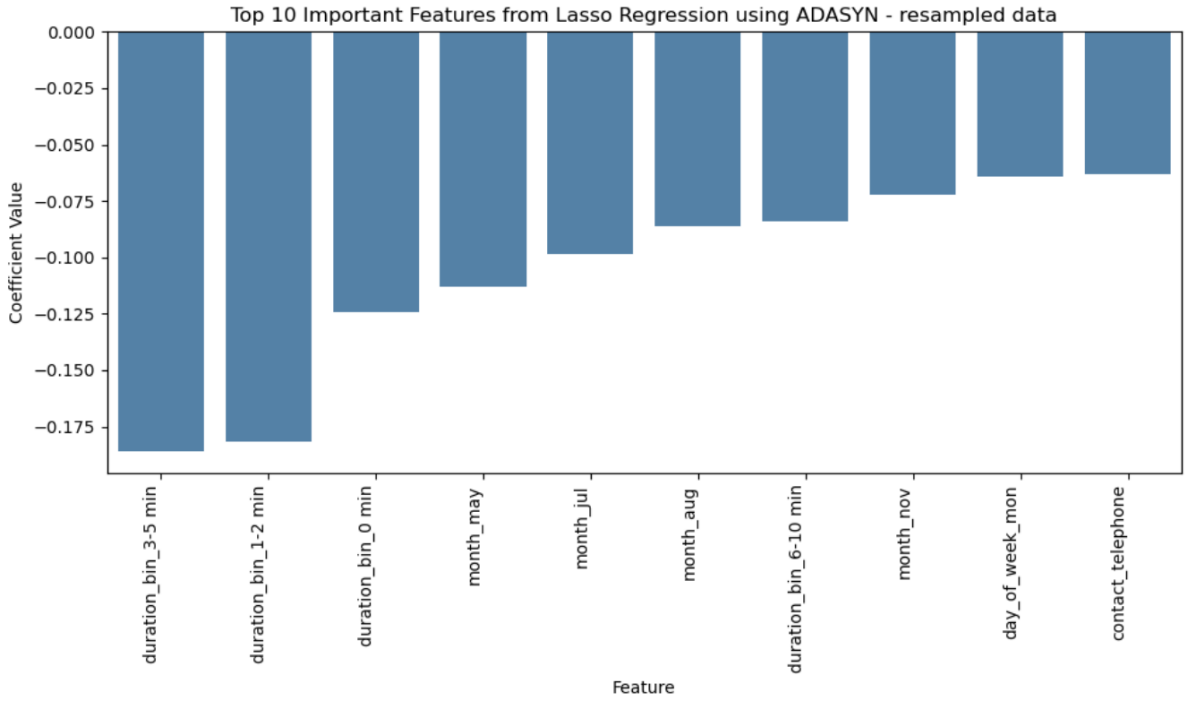
2. Random Forest with ADASYN Balanced Data: A Random Forest was applied to the ADASYN-resampled data to determine feature importance using the feature_importances_ attribute. The top features include duration_bin_1-2 min (0.100), along with other duration bins like duration_bin_3-5 min and duration_bin_11-20 min, highlighting the strong influence of call duration.



3. LASSO with SMOTE Balanced Data: A Lasso regression was applied to the SMOTE-resampled dataset after scaling the features using StandardScaler to identify the most impactful variables. The Lasso model, with an alpha of 0.01, selected the top 10 features based on non-zero coefficients. The results show that call duration bins such as duration_bin_1-2 min and duration_bin_3-5 min have the strongest influence, followed by specific months like May, July, and August, as well as contact type (contact_telephone) and certain days of the week.



4. LASSO with ADAYSN Balanced Data: A Lasso regression was performed on the ADASYN-resampled dataset after scaling the features with StandardScaler to identify the most significant variables. Using an alpha of 0.01, the Lasso model highlighted call duration bins such as duration_bin_3-5 min and duration_bin_1-2 min as the strongest predictors. Other influential features include specific months like May, July, and August, as well as contact_telephone and day_of_week_mon, indicating their notable impact on the dataset.



Model Training and Evaluation Scenarios

Eight different modelling scenarios were tested to evaluate the impact of oversampling techniques and feature selection methods on model performance. The target is to predict whether a client will subscribe to a term deposit.

Random Forest Scenarios (1–4):
Random Forest models were evaluated using two oversampling methods—SMOTE and ADASYN—combined with two feature selection strategies—Lasso and Random Forest feature importance. SMOTE balances the classes by generating synthetic minority samples, while ADASYN (an extension of SMOTE) generates more synthetic data for harder-to-classify cases near the decision boundary. Lasso performs feature selection by shrinking less relevant coefficients to zero, whereas Random Forest importance selects features based on their contribution to reducing model impurity. These scenarios explored how different balancing and feature selection techniques affect Random Forest's ability to detect minority classes and capture nonlinear relationships.

Logistic Regression Scenarios (5–8):
Logistic Regression models were tested with the same balancing and feature selection methods. While SMOTE and ADASYN addressed class imbalance, Lasso emphasized the most significant linear predictors, and Random Forest importance leveraged features ranked by a more complex, nonlinear model. These scenarios assessed how class balancing and refined feature sets impact the predictive performance and interpretability of a simpler linear classifier.

Evaluation Metrics Used:

Following the implementation of various modeling scenarios with different feature selection and sampling techniques, we evaluated their performance using multiple metrics. The next section outlines these evaluation measures, including sensitivity, specificity, accuracy, G-Mean, and AUC.

- **Sensitivity (Recall):** Ability of the model to correctly identify positive cases (subscribers).
- **Specificity:** Ability of the model to correctly identify negative cases (non-subscribers).
- **Precision:** Proportion of predicted positive cases that are actually positive.
- **G-Mean:** Geometric mean of sensitivity and specificity, balancing performance on both classes.
- **Accuracy:** Overall correctness of the model.
- **AUC (Area Under ROC Curve):** Measure of model's ability to discriminate between classes across thresholds.

Confusion matrices were plotted for each scenario to visually assess true positives, false positives, true negatives, and false negatives.

Model Performance Across Feature Selection and Balancing Strategies:

Scenario	Model	Encoding Approach	Imputation Method	Outlier Handling Method	Balancing Approach	Feature Selection	Sensitivity	Specificity	Precision	G-Mean	Accuracy	AUC
1	Random Forest	One Hot Encoding	N/A	Binning	SMOTE	Lasso-selected	0.676	0.877	0.411	0.770	0.854	0.875
2	Random Forest	One Hot Encoding	N/A	Binning	ADASYN	Lasso-selected	0.713	0.856	0.386	0.782	0.840	0.875
3	Random Forest	One Hot Encoding	N/A	Binning	SMOTE	RF-selected	0.826	0.757	0.301	0.791	0.764	0.870
4	Random Forest	One Hot Encoding	N/A	Binning	ADASYN	RF-selected	0.728	0.819	0.338	0.772	0.809	0.861
5	Logistic Regression	One Hot Encoding	N/A	Binning	SMOTE	Lasso-selected	0.678	0.866	0.391	0.767	0.845	0.867
6	Logistic Regression	One Hot Encoding	N/A	Binning	ADASYN	Lasso-selected	0.642	0.881	0.406	0.756	0.854	0.867
7	Logistic Regression	One Hot Encoding	N/A	Binning	SMOTE	RF-selected	0.837	0.759	0.306	0.780	0.767	0.874
8	Logistic Regression	One Hot Encoding	N/A	Binning	ADASYN	RF-selected	0.705	0.830	0.344	0.765	0.816	0.860

Insights:

- Sensitivity (Recall) is prioritized due to the imbalanced nature of the dataset, as it reflects the model's ability to correctly identify positive cases (i.e., customers who subscribe).

- Scenario 7 (Logistic Regression + SMOTE + Random Forest-selected features) achieved the highest sensitivity (0.8373) and G-Mean (0.7972), indicating strong performance in identifying true positives while maintaining balanced classification.

- Scenarios 1 and 2 (Random Forest + SMOTE/ADASYN + Lasso-selected features) recorded the highest AUC scores (0.8752 and 0.8750), showing strong overall discriminative power between the classes.

- Scenario 6 (Logistic Regression + ADASYN + Lasso) achieved the highest specificity (0.8808), making it most effective in minimizing false positives.

- Models using Random Forest for feature selection (Scenarios 3, 4, 7, 8) generally resulted in higher sensitivity and G-Mean, while those using Lasso selection showed relatively better specificity and AUC.

- SMOTE generally outperformed ADASYN in sensitivity across both Random Forest and Logistic Regression models, suggesting it may be more suitable for this dataset.

Top 2 Scenarios:

- **Scenario 7 (Logistic Regression + SMOTE + RF-selected):**
  This scenario stands out with the highest sensitivity (0.8373) and G-Mean (0.7972)**,** ensuring that the majority of subscribers are correctly identified. Its strong AUC (0.8744) confirms its ability to distinguish between classes effectively. By using SMOTE to handle class imbalance and RF-selected features to remove irrelevant variables, the model achieves high recall and balanced performance, which is crucial for imbalanced datasets.
- **Scenario 3 (Random Forest + SMOTE + RF-selected):**
  With a sensitivity of 0.8265, a G-Mean of 0.7910, and a high AUC (0.8702), this scenario demonstrates excellent capability in detecting positive cases while maintaining good class balance. The combination of Random Forest's strength in modelling complex relationships and SMOTE's oversampling creates a robust approach for improving predictions on minority class outcomes.

Conclusion

Our analysis shows that the Logistic Regression model using SMOTE and Random Forest-selected features (Scenario 7) best identifies potential subscribers with the highest sensitivity and balanced accuracy. Random Forest models with SMOTE or ADASYN combined with Lasso feature selection (Scenarios 1 and 2) provide the strongest overall prediction accuracy (AUC). The Logistic Regression model with ADASYN and Lasso features (Scenario 6) is most effective at reducing false positives. Depending on business goals, Scenario 7 is ideal for maximizing subscriber detection, while Scenarios 1 and 2 offer the most reliable overall predictions.

With the best-performing model determined, the final step focuses on translating these findings into actionable strategies. The **Evaluation and Deployment phase** details how these insights can enhance bank marketing campaigns through targeted and data-driven decision-making.

1. Business Problem and Best Model

Business Problem:

The bank seeks to improve the effectiveness of its telemarketing campaigns by identifying customers most likely to subscribe to term deposit accounts. Given the significant class imbalance in the dataset (only ~11% of customers subscribe), it's critical to build a model that can effectively detect potential subscribers without overfitting to the majority class.

Best Performing Model:

In our Bank Marketing Project, the top – performing model is Scenario 7.

- **Model:** Logistic Regression
- **Balancing Technique:** SMOTE
- **Feature Selection:** Random Forest Importance
- **Sensitivity (Recall):** 0.8373
- **G-Mean:** 0.7972
- **AUC**: 0.8744

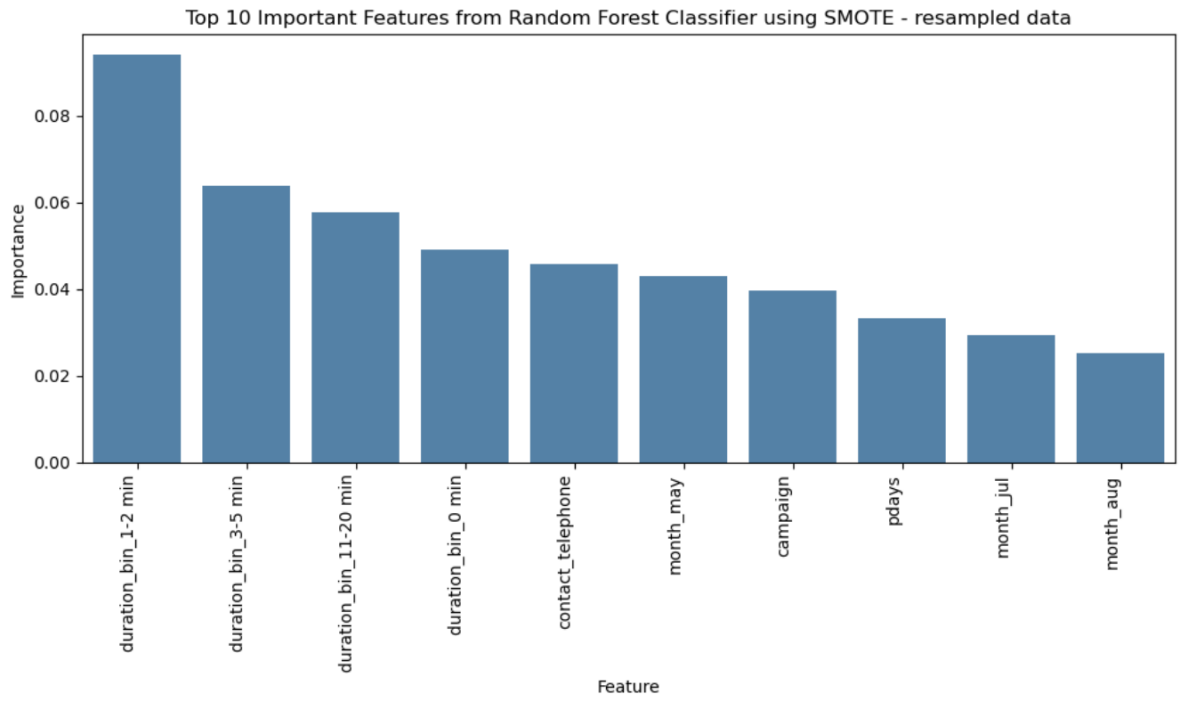
This model offers the best balance between high recall (to catch most subscribers) and overall discriminative power, making it ideal for maximizing campaign success.

2. Feature Summary

Below is a table of the top features selected by Random Forest Importance (used in Scenario 7), along with their corresponding importance scores:

Features Name	Importance Score
duration_bin_1-2 min	0.094006
duration_bin_3-5 min	0.063861
duration_bin_11-20 min	0.057763
duration_bin_0 min	0.049117
contact_telephone	0.045780
month_may	0.042859
campaign	0.039535
pdays	0.033338
month_jul	0.029443
month_aug	0.025249

Bar Plot:



Higher or Lower value implications for the outcome variable:

- **duration_bin_1-2 min**
  Higher value (i.e., calls lasting 1–2 minutes) suggests very brief interactions, which are generally ineffective for converting customers. Lower value indicates calls that are either longer or not in this short range, which tend to perform better.
- **duration_bin_3-5 min**
  Higher value indicates calls lasted 3–5 minutes, which usually shows better engagement than shorter calls. Lower values imply shorter conversations that fail to build rapport.
- **duration_bin_11-20 min**
  Higher value reflects long, detailed conversations, which are highly associated with successful subscriptions. Lower value suggests minimal engagement.
- **duration_bin_0 min**
  Higher value means no conversation occurred (failed or missed call), which leaves no chance to convert the customer. Lower value means actual engagement took place.
- **contact_telephone**
  Higher value indicates the customer was contacted on a landline rather than mobile, which tends to be less effective. Lower values (cellular contact) often perform better due to easier accessibility.
- **month_may**
  Higher value means many calls occurred in May, but this month has historically

underperformed. Lower value implies distribution across other months, which may yield better results.

- **campaign**
  Higher values represent multiple contact attempts within the same campaign. While repeated attempts can help, too many can annoy customers and reduce effectiveness. Lower values imply fewer, more strategic contacts.
- **pdays**
  Higher values indicate a long gap since the last contact, which reduces success probability. Lower values (recent follow-ups) significantly improve chances.
- **month_jul**
  Higher value means contacts were made in July, a generally less responsive month. Lower value indicates calls in other months that may work better.
- **month_aug**
  Similar to July, higher values indicate calls in August, which tend to perform modestly. Lower value suggests calls in more favourable months.

Overall, Longer calls, timely follow-ups, mobile contact, and focusing on better-performing months drive higher subscription success.

3. Detailed Feature Analysis

Summary Statistics of top features:

Binary Features Summary Statistics

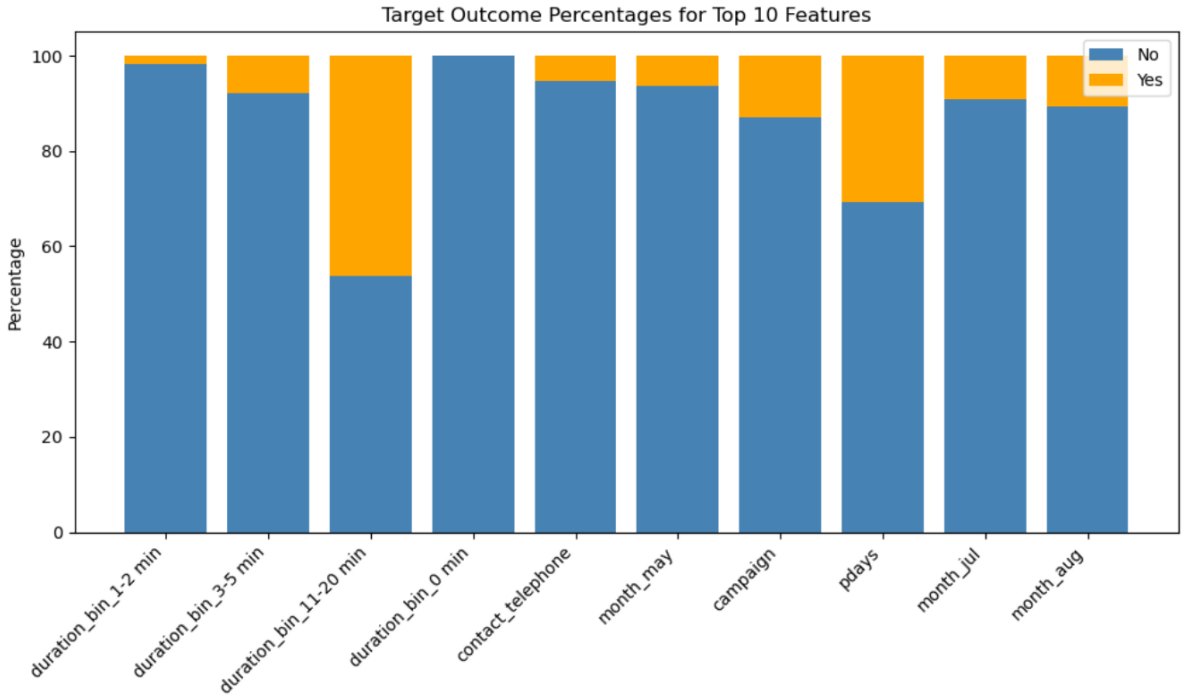
	count	unique	top	freq
duration_bin_1–2 min	41188	2	False	32574
duration_bin_3–5 min	41188	2	False	24045
duration_bin_11–20 min	41188	2	False	38225
duration_bin_0 min	41188	2	False	37007
contact_telephone	41188	2	False	26144
month_may	41188	2	False	27419
month_jul	41188	2	False	34014
month_aug	41188	2	False	35010

Continuous Feature Summary Statistics

	count	mean	std	min	25%	50%	75%	max
campaign	41188.0	2.567593	2.770014	1.0	1.0	2.0	3.0	56.0
pdays	41188.0	962.475454	186.910907	0.0	999.0	999.0	999.0	999.0

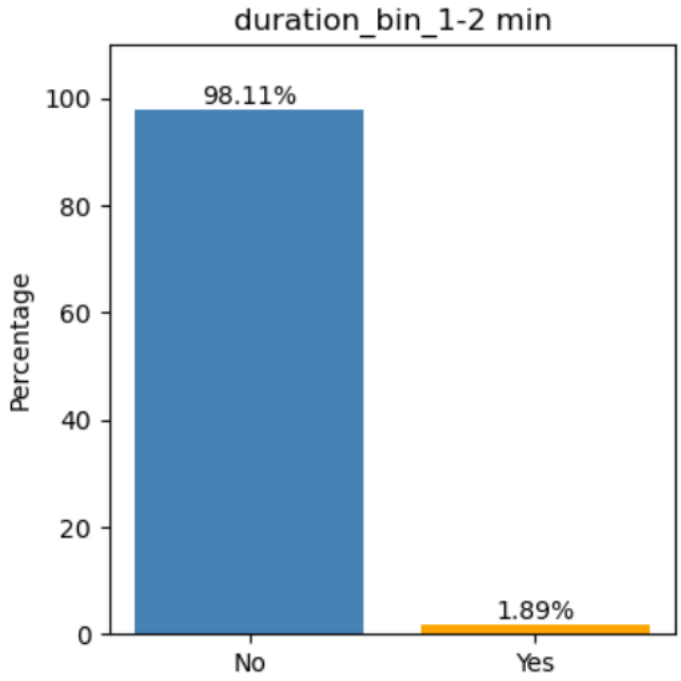
Top 10 features to the target variable:

Features Name	No %	Yes %
duration_bin_1-2 min	98.11	1.89
duration_bin_3-5 min	92.15	7.85
duration_bin_11-20 min	53.80	46.20
duration_bin_0 min	99.98	0.02
contact_telephone	94.77	5.23
month_may	93.57	6.43
campaign	86.96	13.04
pdays	69.23	30.77
month_jul	90.95	9.05
month_aug	89.40	10.60



sIndividual Features distribution:

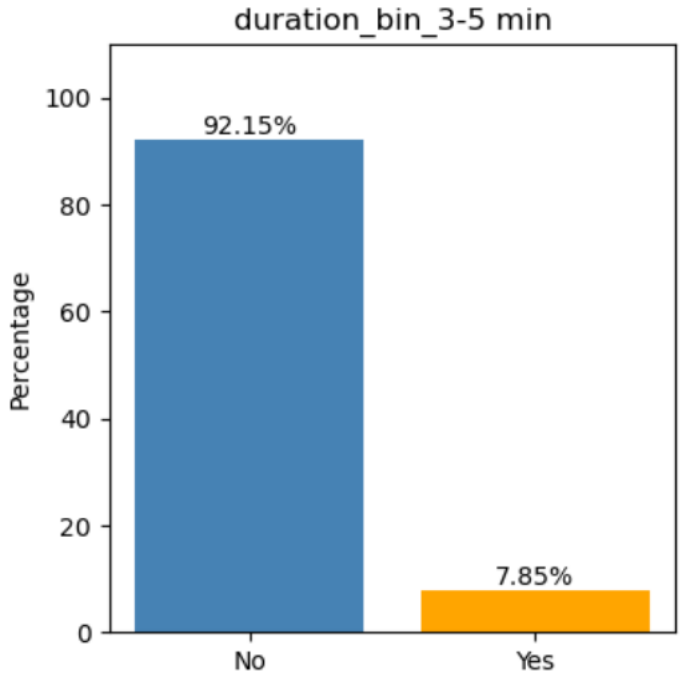
duration_bin_1-2 min:



Insights:

Very short calls (1–2 minutes) rarely lead to customers signing up for a term deposit. These calls likely end before agents can explain the offer or build interest. To improve success, calls must be long enough to establish value and answer questions.

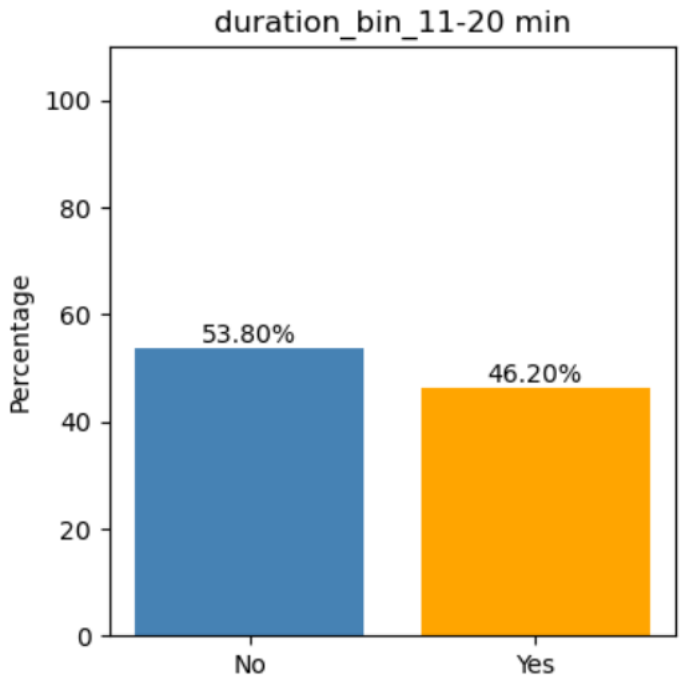
duration_bin_3-5 min:



Insights:

Slightly longer calls show improved results, but success is still low. This suggests that 3–5 minutes might allow for basic information sharing, yet not enough depth. Agents may need more time or better scripting to persuade customers.

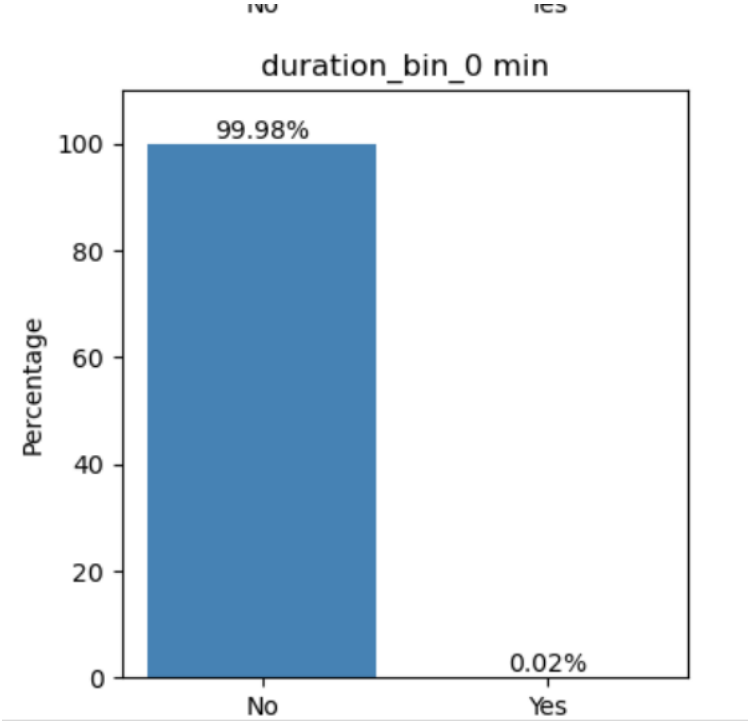
duration_bin_11-20 min:



Insights:

These calls have the highest success rate, with nearly half resulting in sign-ups. This shows that detailed, engaging conversations are key. Campaigns should focus on fewer but longer calls where agents can address concerns and build trust.

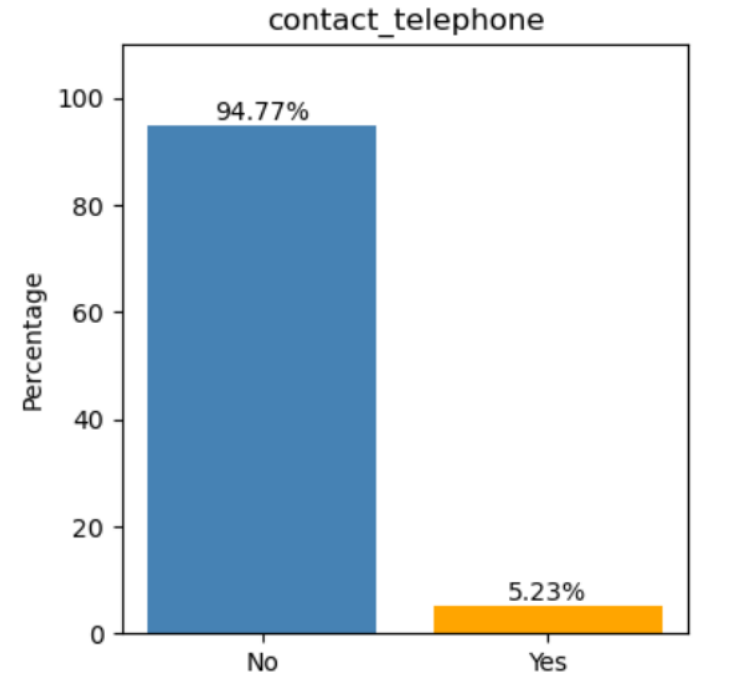
duration_bin_0 min:



Insights:

Calls with 0 duration almost never succeed—these are likely missed, dropped, or instantly rejected. They reflect inefficiencies in dialling or targeting. Reducing such calls will save time and increase focus on reachable leads.

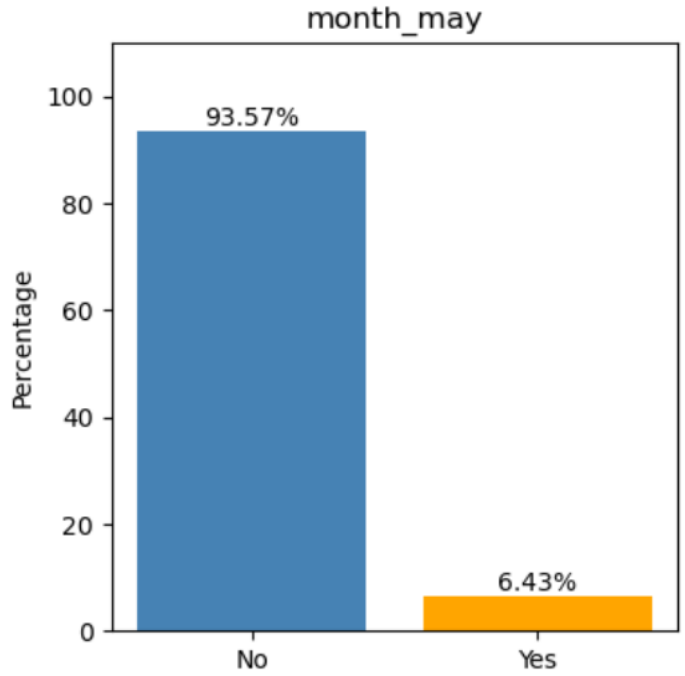
contact_telephone:



Insights:

Using traditional telephone numbers results in lower sign-up rates compared to mobile contacts. This may be due to outdated contact info or customer preferences. Shifting focus to mobile outreach could significantly improve engagement.

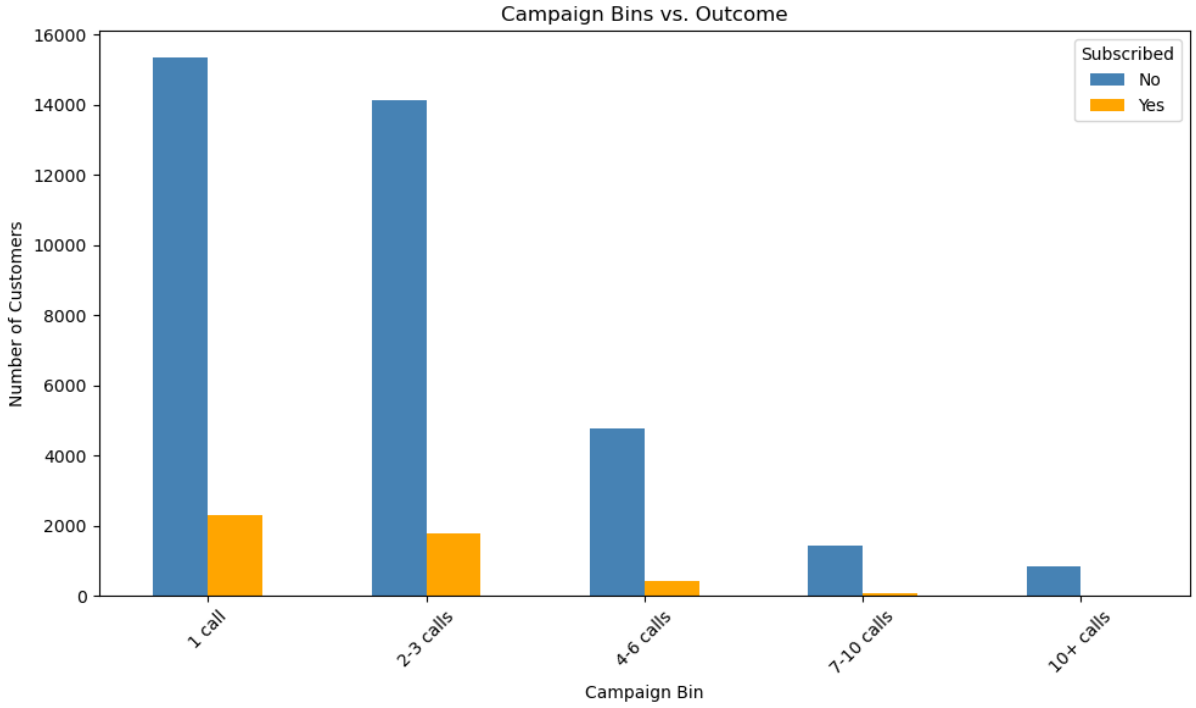
month_may:



Insights:

May shows low success despite likely being a peak outreach period. This could be due to campaign fatigue or oversaturation. Rethinking message timing or diversifying outreach across other months might increase effectiveness.

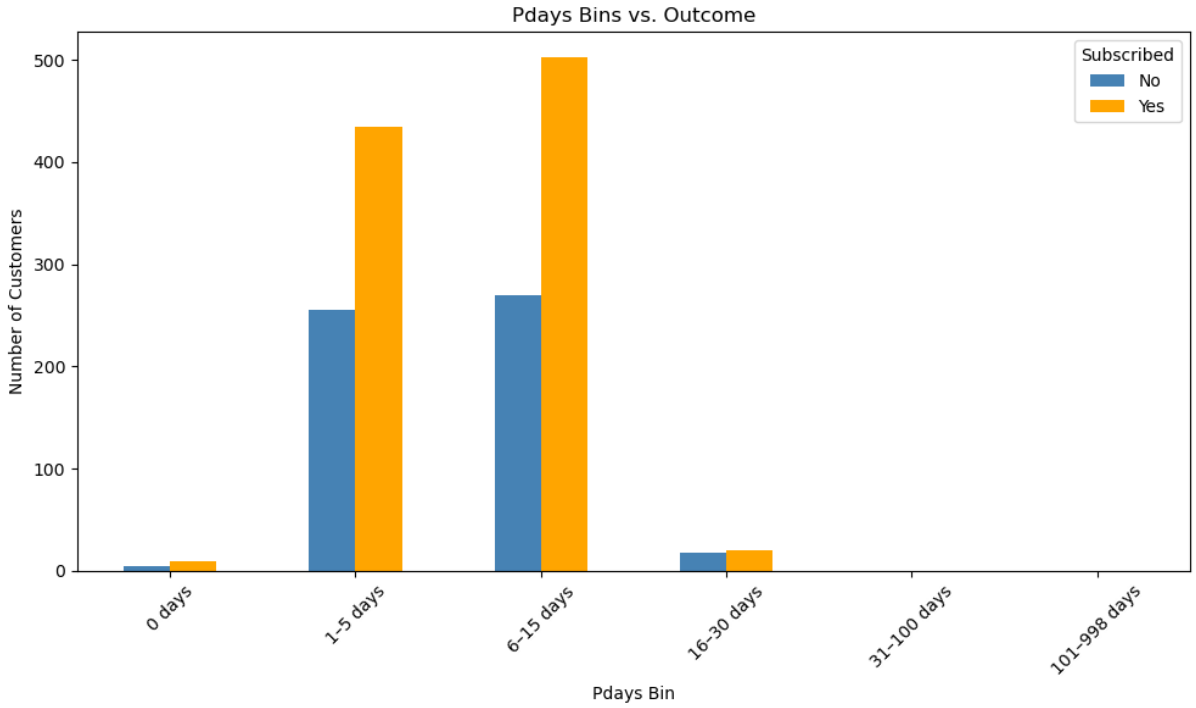
campaign:



Insights:

This graph shows that most subscriptions happen within first few calls, with a sharp decline as the number of calls increases. Beyond 4-6 calls the likelihood of getting 'YES' drops significantly, suggesting that repeated calls may be ineffective and even counterproductive.

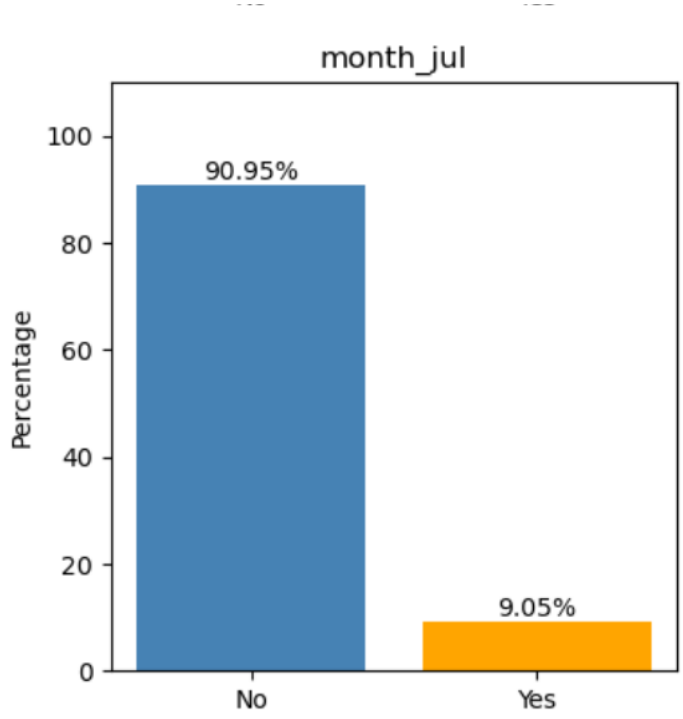
pdays:



Insights:

Most subscriptions happen when customers are contacted within 1-15 days, with success peaking in this window. Beyond 15 days, responses drop sharply, showing that timely follow-ups are key to higher conversions.

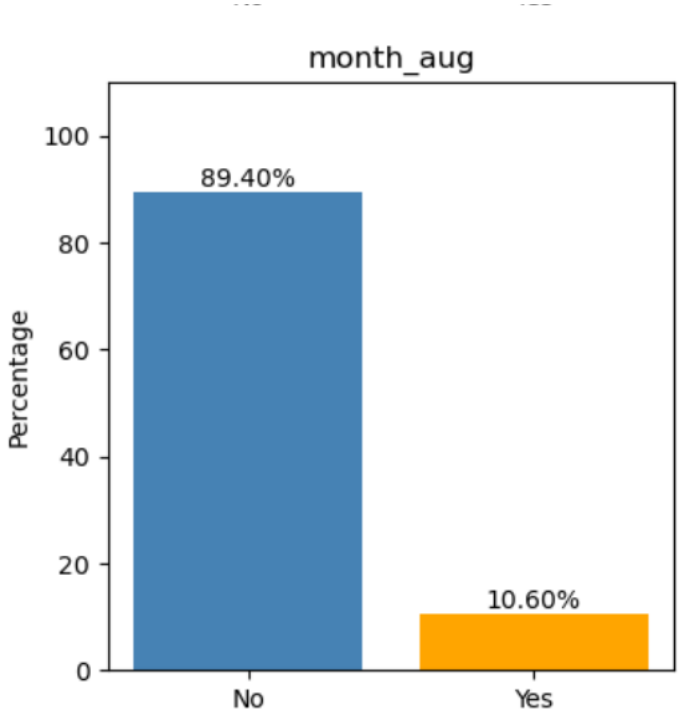
month_jul:



Insights:

Campaigns in July show below-average success, possibly due to summer holidays when people are less responsive. Avoid heavy outreach during low-attention periods or adjust messaging to account for seasonal factors.

month_aug:



Insights:

August performs slightly better than July, but still trails behind other months. Continued low performance may be tied to vacation season. Spacing out campaigns or using alternative channels could help increase impact.

4. Actionable Recommendations:

- Target customers likely to engage in longer conversations, as calls lasting 11–20 minutes show the highest conversion rates.
- Enhance call scripts to make shorter calls (1–5 minutes) more impactful by including concise value propositions and stronger opening lines.
- Reduce zero-duration calls by improving lead validation, optimizing dialing systems, and scheduling calls when customers are most likely to answer.
- Prioritize mobile contacts over landlines to increase reach and improve conversion potential.
- Schedule follow-ups within 1–15 days of the previous contact to maximize engagement, as success rates decline sharply after this window.
- Set a reasonable cap on follow-up attempts, as conversions drop significantly after 4–6 calls, and excessive attempts may create customer fatigue.

- Rethink outreach strategies for May by reducing oversaturation and incorporating more personalized or targeted messaging.
- Adjust campaigns in July and August by lowering call intensity and supplementing with digital channels or seasonal incentives to maintain engagement during low-response periods.
- Implement predictive lead scoring using key features such as call duration, contact type, recent interaction, and campaign timing to allocate resources effectively.
- Continuously monitor performance metrics and adapt strategies based on data-driven insights to improve campaign outcomes over time.

References:

1. Zaki, A. M., Khodadadi, N., Hong Lim, W., & Towfek, S. K. (2024). Predictive Analytics and Machine Learning in Direct Marketing for Anticipating Bank Term Deposit Subscriptions. *American Journal of Business & Operations Research*, *11*(1). https://www.researchgate.net/profile/Ahmed-Mohamed-Zaki/publication/376585755_Predictive_Analytics_and_Machine_Learning_in_Direct_Marketing_for_Anticipating_Bank_Term_Deposit_Subscriptions/links/658a883d0bb2c7472b105cae/Predictive-Analytics-and-Machine-Learning-in-Direct-Marketing-for-Anticipating-Bank-Term-Deposit-Subscriptions.pdf
2. Ruangthong, P., & Jaiyen, S. (2015, July). Bank direct marketing analysis of asymmetric information based on machine learning. In *2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE)* (pp. 93-96). IEEE.https://ieeexplore.ieee.org/abstract/document/7219777/
3. Tang, X., & Zhu, Y. (2024). Enhancing bank marketing strategies with ensemble learning: Empirical analysis. *Plos one*, *19*(1), e0294759. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0294759
4. Torrens, M., & Tabakovic, A. (2022). A banking platform to leverage data driven marketing with machine learning. *Entropy*, *24*(3), 347. https://www.mdpi.com/1099-4300/24/3/347
5. A data-driven approach to predict the success of bank telemarketing By Sérgio Moro, P. Cortez, P. Rita. 2014 Published in Decision Support Systems
6. Tableau: Tableau – 2025. https://www.tableau.com/community/public
7. ChatGPT: Open AI. ChatGPT (version 4) [Large Language Model]