Finlatics Data Science Project

Suryanshu

March 29, 2024

Contents

Data Presets

0.1 PYTHON CODE:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the Dataset
df = pd.read_csv("banking_data.csv")
df.drop(columns = 'marital_status', inplace = True)
df.drop(columns = 'day_month', inplace = True)
df['marital'].replace({"married":1, "single":0, "divorced
    ":2}, inplace = True)
df['education'].replace({"secondary":2, "tertiary":3, "
    primary":1, "unknown":0}, inplace = True)
df['default'].replace({"yes":1, "no":0}, inplace = True)
df['housing'].replace({"yes":1, "no":0}, inplace = True)
df['loan'].replace({"yes":1, "no":0}, inplace = True)
df['contact'].replace({"cellular":2, "telephone":1, "
    unknown": 0}, inplace = True)
df['poutcome'].replace({"success":1, "failure":0, "other"
    :2, "unknown":3}, inplace = True)
df['y'].replace({"yes":1, "no":0}, inplace = True)
```

Question 1

What is the distribution of age among the clients?

1.1 PYTHON CODE:

```
plt.hist(df['age'], color='skyblue', edgecolor='black')
plt.xlabel("Age")
plt.ylabel("Frequency")
plt.title("Distribution of Age amongst clients")
plt.show()
```

1.2 Output

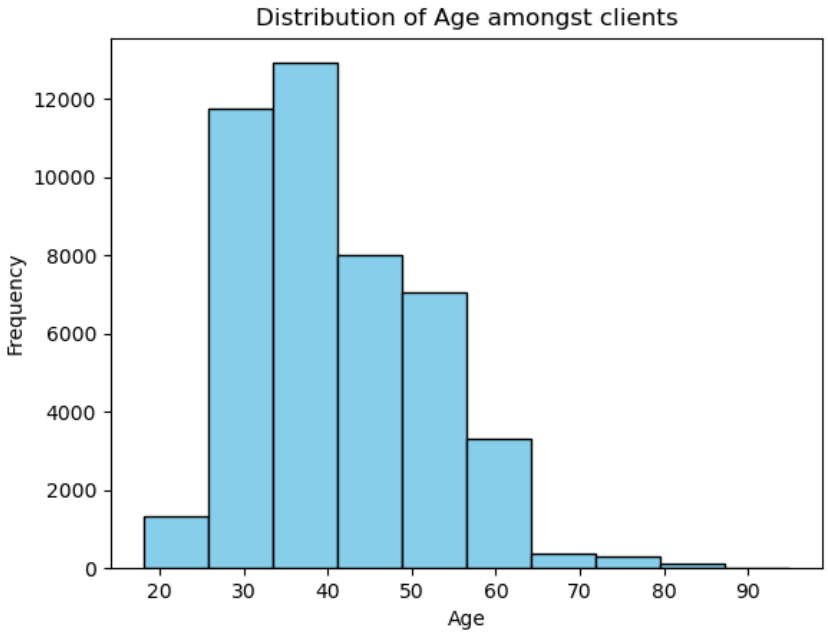

Figure 1.1: Output

Question 2

How does the job type vary among the clients?

2.1 PYTHON CODE:

```
job_type_counts = pd.Series(df['job']).value_counts()
job_type_counts.plot(kind='bar', color='skyblue')
plt.title('Variation of Job Type Among Clients')
plt.xlabel('Job Type')
plt.ylabel('Number of Clients')
plt.show()
```

2.2 Output

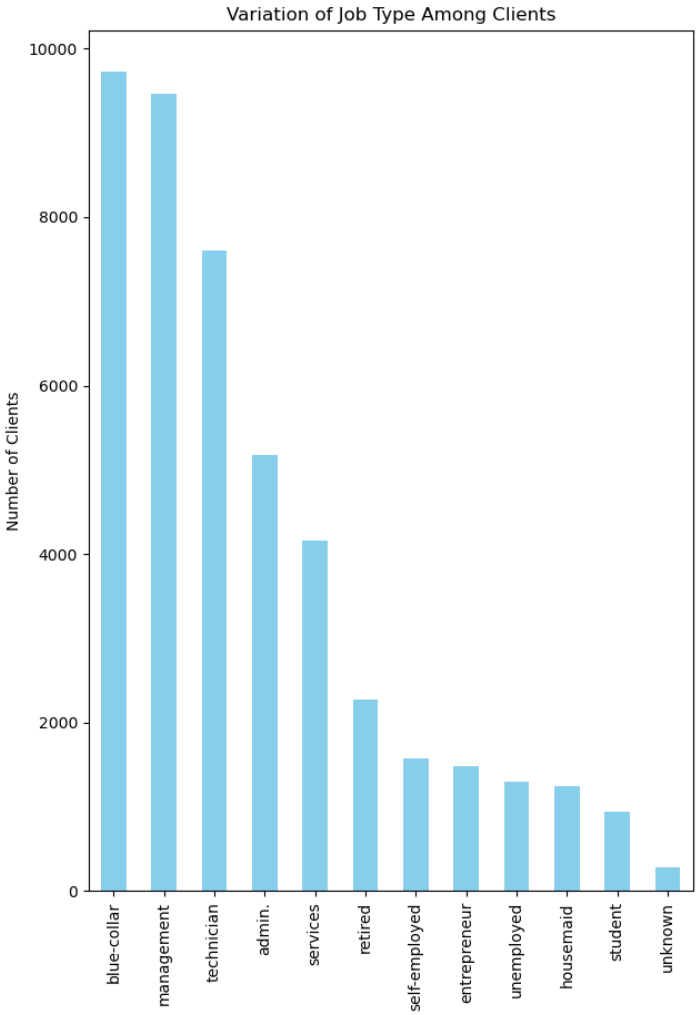

Figure 2.1: Output

Question 3

What is the marital status distribution of the clients?

3.1 PYTHON CODE:

```
marital_status = pd.Series(df['marital']).value_counts()
marital_status.plot(kind='bar', color='skyblue')
plt.title('Marital Status Distribution of Clients')
plt.xlabel('Marital Status')
plt.ylabel('Number of Clients')
plt.show()
df['marital'].replace({"married":1, "single":0, "divorced
    ":2}, inplace = True)
```

3.2 Output

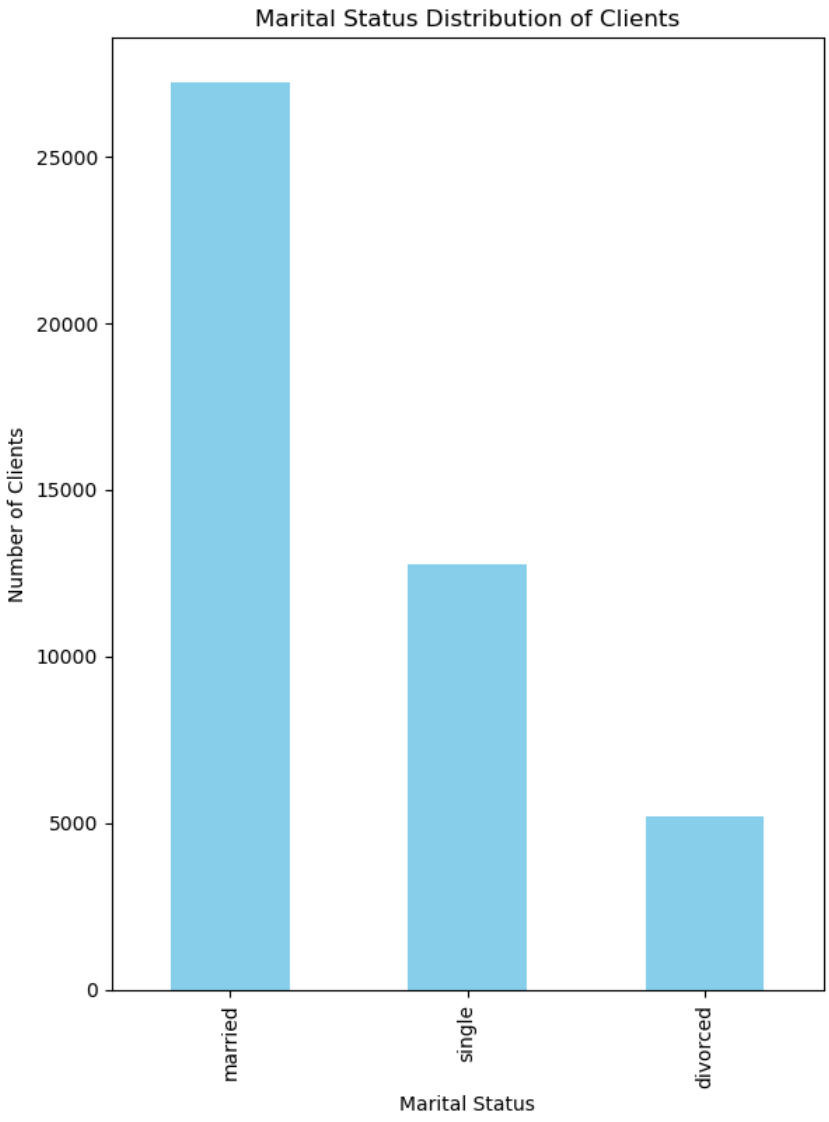

Figure 3.1: Output

Question 4

What is the level of education among the clients?

4.1 PYTHON CODE:

```
education_level = pd.Series(df['education']).value_counts()
education_level.plot(kind='bar', color='skyblue')
plt.title('Education Level of Clients')
plt.xlabel('Education level')
plt.ylabel('Number of Clients')
plt.show()
df['education'].replace({"secondary":2, "tertiary":3, "primary":1, "unknown":0}, inplace = True)
```

4.2 Output

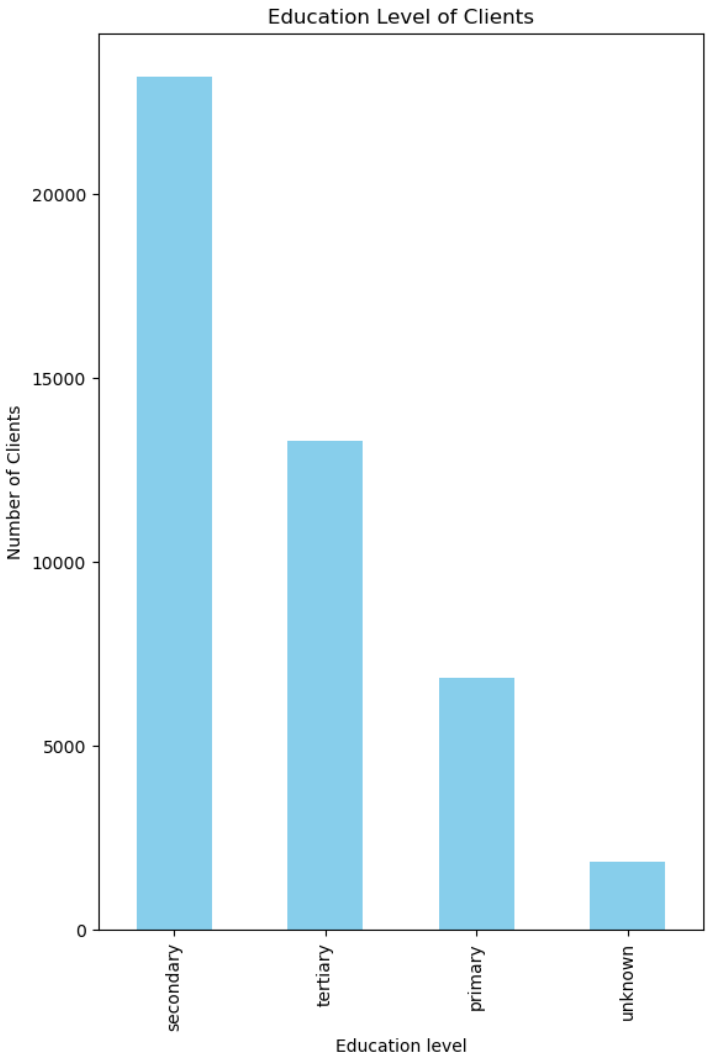

Figure 4.1: Output

Question 5

What proportion of clients have credit in default?

5.1 PYTHON CODE:

```
default_counts = df['default'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(default_counts, labels=default_counts.index,
    autopct='%1.1f%%', startangle=140)
plt.title('Default Credit')
plt.axis('equal')
plt.show()
```

5.2 Output

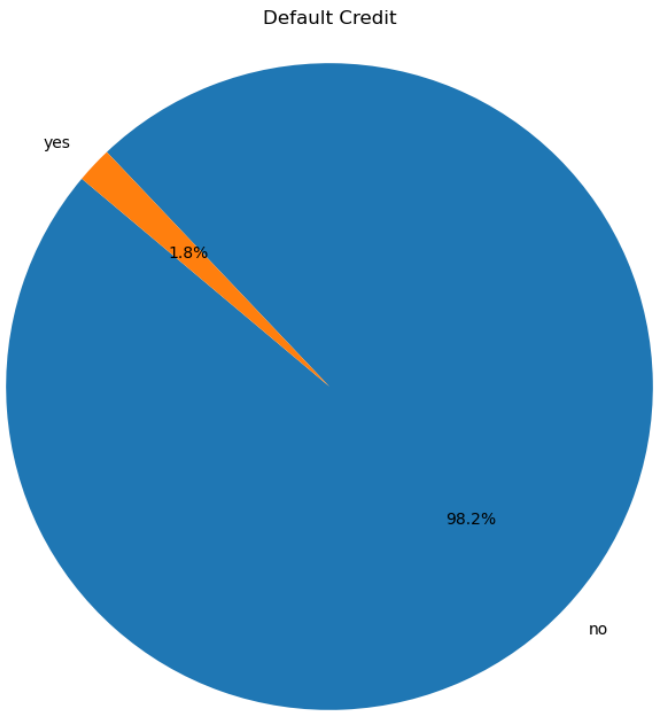

Figure 5.1: Output

Question 6

What is the distribution of average yearly balance among the clients?

6.1 PYTHON CODE:

```
print(df['balance'].median())
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='balance')
plt.title('Box Plot of Account Balances')
plt.xlabel('Account Balance')
plt.ylabel('Density')
plt.show()
```

6.2 Output

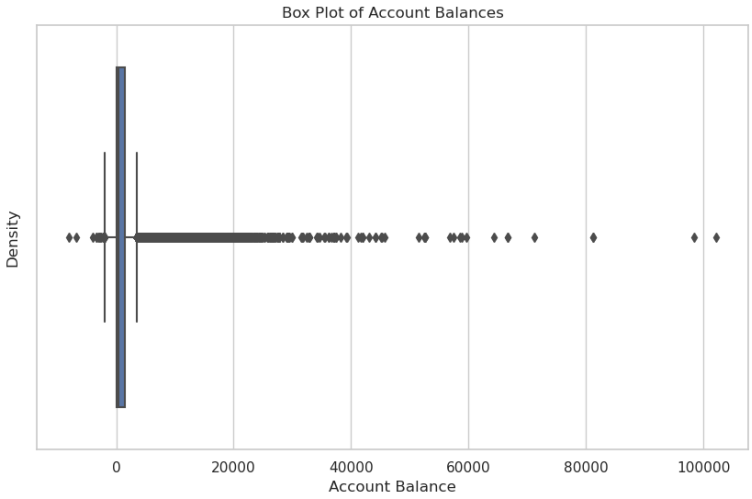

Figure 6.1: Output

Question 7

How many clients have housing loans?

7.1 PYTHON CODE:

```
housing_counts = df['housing'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(housing_counts, labels=housing_counts.index,
    autopct='%1.1f%%', startangle=140)
plt.title('Housing Loans')
plt.axis('equal')
plt.show()
```

7.2 Output

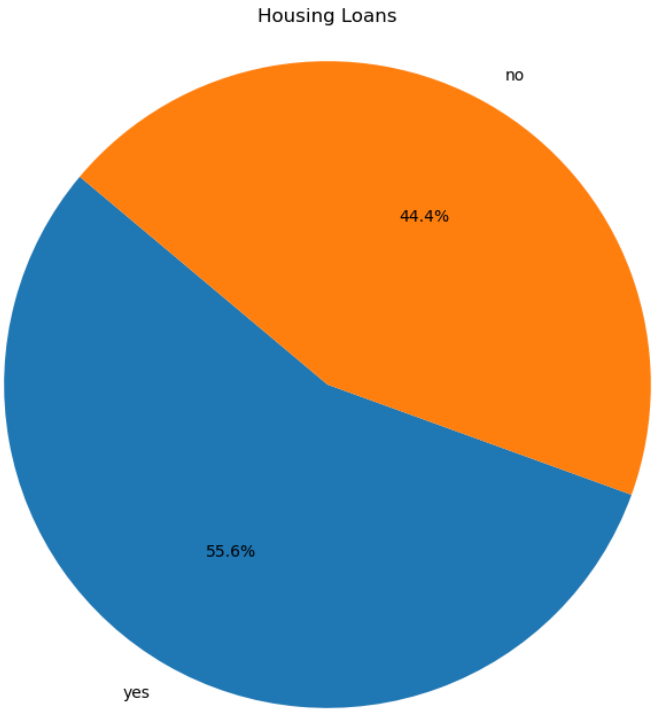

Figure 7.1: Output

Question 8

How many clients have personal loans?

8.1 PYTHON CODE:

```
loan_counts = df['loan'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(loan_counts, labels=loan_counts.index, autopct='
    %1.1f%%', startangle=140)
plt.title('Personal Loans')
plt.axis('equal')
plt.show()
```

8.2 Output

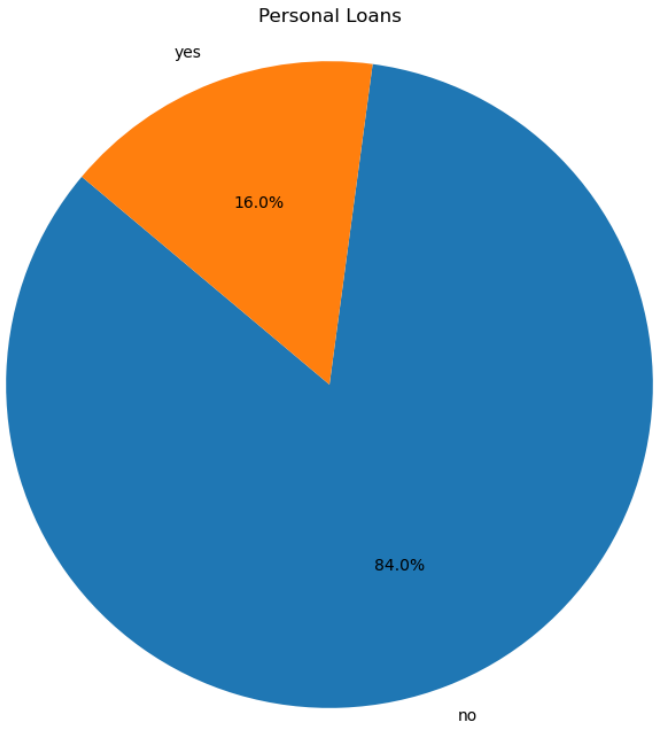

Figure 8.1: Output

Question 9

What are the communication types used for contacting clients during the campaign?

9.1 PYTHON CODE:

```
communication_type = pd.Series(df['contact']).
    value_counts()
communication_type.plot(kind='bar', color='skyblue')
plt.title('Communication types used to contact clients')
plt.xlabel('Type of Communication')
plt.ylabel('Number of Clients')
plt.show()
```

9.2 Output

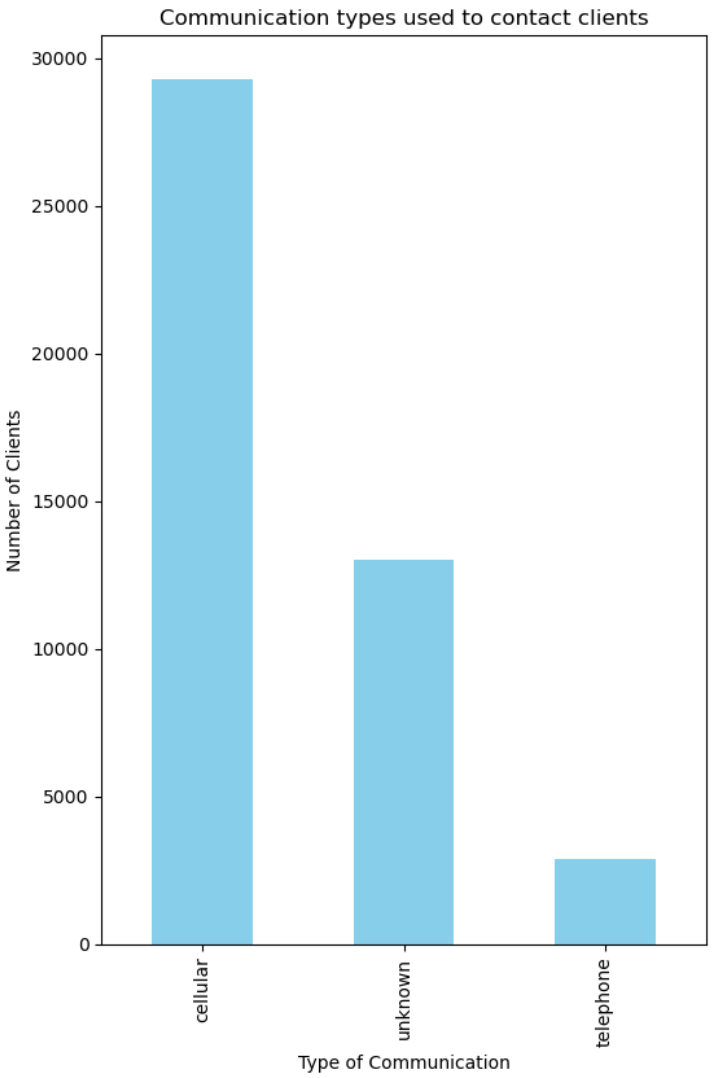

Figure 9.1: Output

Question 10

What is the distribution of the last contact day of the month?

10.1 PYTHON CODE:

```
print(df['day'].mode()[0])
day_counts = df['day'].value_counts().sort_index()
plt.figure(figsize=(10, 6))
plt.plot(day_counts.index, day_counts.values, marker='o',
    linestyle='-')
plt.title('Distribution of Last Contact Day of the Month'
    )
plt.xlabel('Day of the Month')
plt.ylabel('Number of Contacts')
plt.grid(True)
plt.show()
```

10.2 Output

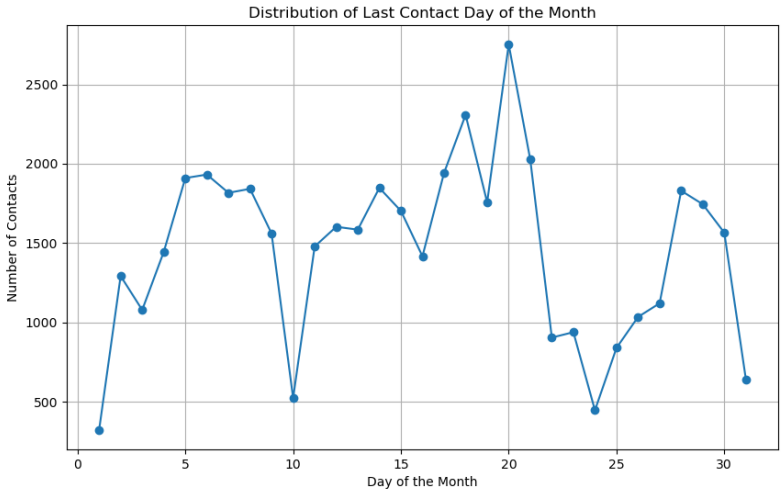

Figure 10.1: Output

Question 11

How does the last contact month vary among the clients?

11.1 PYTHON CODE:

```
print(df['month'].mode()[0])
month_counts = df['month'].value_counts().sort_index()
plt.figure(figsize=(10, 6))
plt.plot(month_counts.index, month_counts.values, marker='o', linestyle='-')
plt.title('Distribution of Last Contact Month')
plt.xlabel('Month of the Year')
plt.ylabel('Number of Contacts')
plt.grid(True)
plt.show()
```

11.2 Output

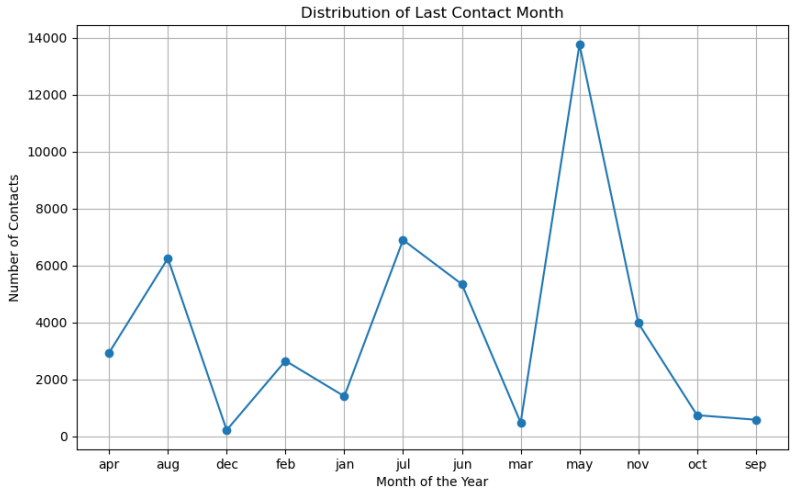

Figure 11.1: Output

Question 12

What is the distribution of the duration of the last contact?

12.1 PYTHON CODE:

```
print(df['duration'].median())
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='duration')
plt.title('Box Plot of Call Duration')
plt.xlabel('Call duration in seconds')
plt.ylabel('Distribution')
plt.show()
```

12.2 Output

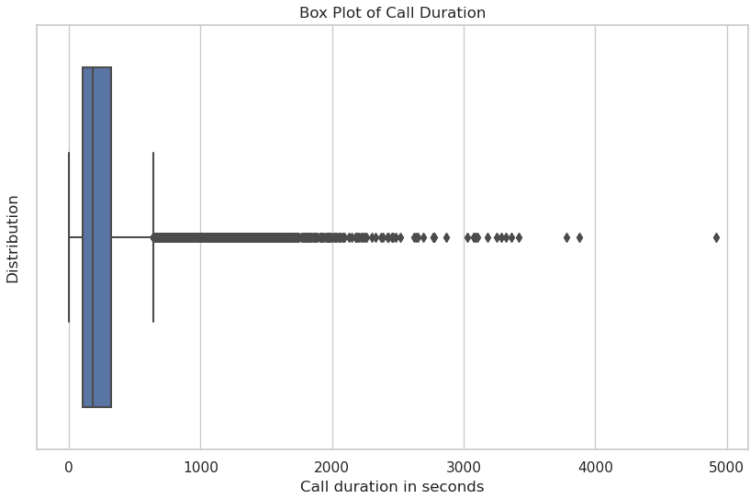

Figure 12.1: Output

Question 13

How many contacts were performed during the campaign for each client?

13.1 PYTHON CODE:

```
print(df['campaign'].median())
campaign_counts = df['campaign'].value_counts().
    sort_index()
plt.figure(figsize=(10, 6))
plt.plot(campaign_counts.index, campaign_counts.values,
    marker='o', linestyle='-')
plt.title('Contacts performed during the campaign')
plt.xlabel('Number of times contacted')
plt.ylabel('Number of clients contacted')
plt.grid(True)
plt.show()
```

13.2 Output

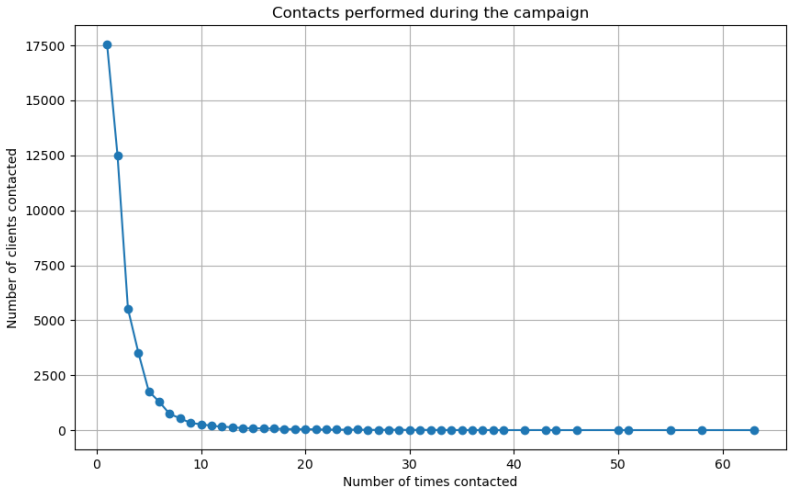

Figure 13.1: Output

Question 14

What is the distribution of the number of days passed since the client was last contacted from a previous campaign?

14.1 PYTHON CODE:

```
print(df['pdays'].median())
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='pdays')
plt.title('Number of days passed since last contact')
plt.xlabel('Number of days')
plt.ylabel('Distribution')
plt.show()
```

14.2 Output

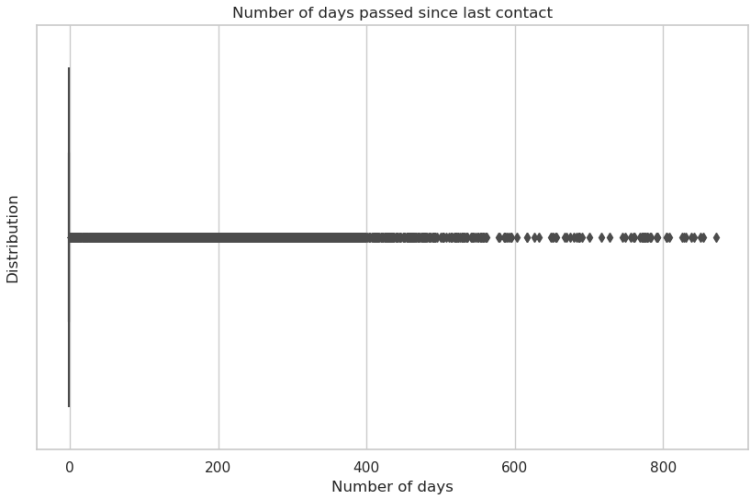

Figure 14.1: Output

Question 15

How many contacts were performed before the current campaign for each client?

15.1 PYTHON CODE:

```
print(df['previous'].median())
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='previous')
plt.title('Number of contacts performed before the
    current campaign')
plt.xlabel('Number of contacts')
plt.ylabel('Distribution')
plt.show()
```

15.2 Output

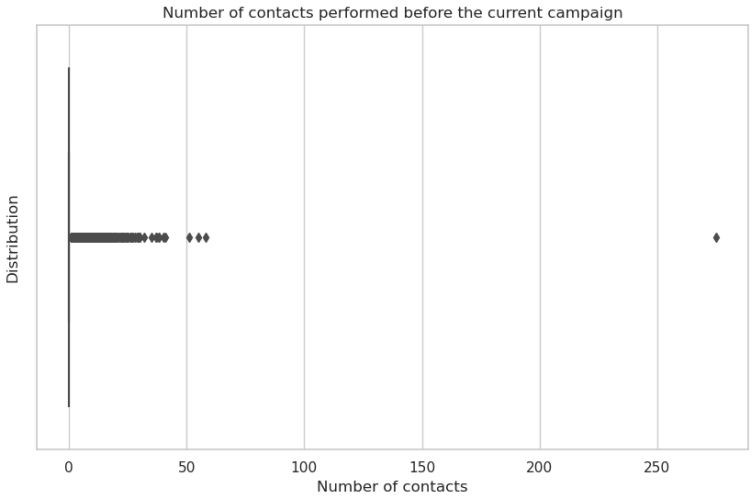

Figure 15.1: Output

Question 16

What were the outcomes of the previous marketing campaigns?

16.1 PYTHON CODE:

```
outcome_type = pd.Series(df['poutcome']).value_counts()
outcome_type.plot(kind='bar', color='skyblue')
plt.title('Outcome of previous marketing campaigns')
plt.xlabel('Outcome')
plt.ylabel('Number of Clients')
plt.show()
```

16.2 Output

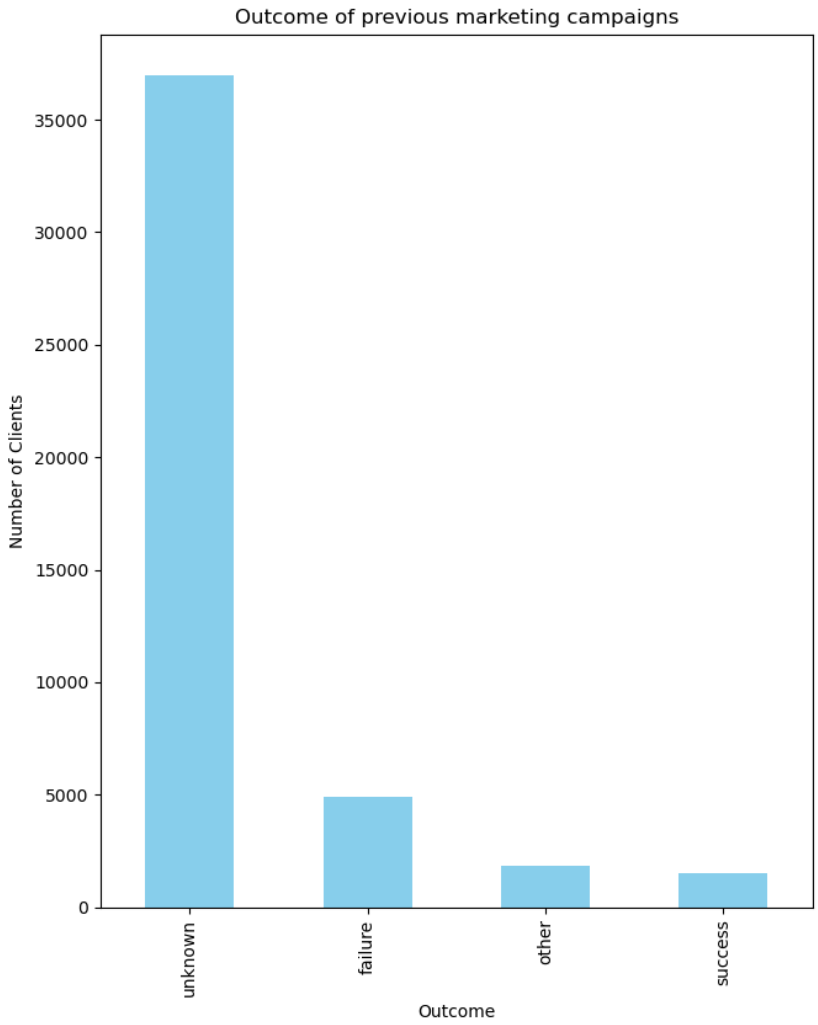

Figure 16.1: Output

Question 17

What is the distribution of clients who subscribed to a term deposit vs. those who did not?

17.1 PYTHON CODE:

```
subscribed_counts = df['y'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(subscribed_counts, labels=subscribed_counts.index
    , autopct='%1.1f%%', startangle=140)
plt.title('Subscribed')
plt.axis('equal')
plt.show()
```

17.2 Output

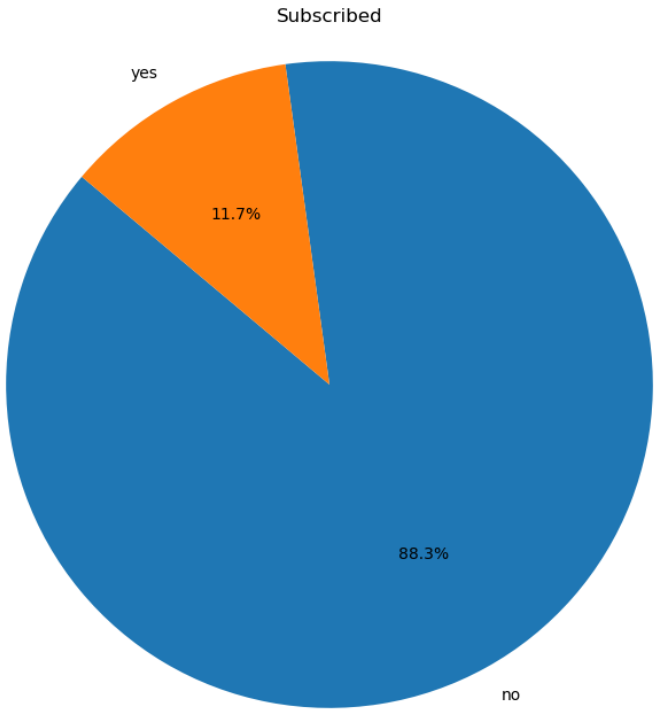

Figure 17.1: Output

Question 18

Are there any correlations between different attributes and the likelihood of subscribing to a term deposit?

18.1 PYTHON CODE:

```
target_variable = 'y'
correlations = df.corr()[target_variable].sort_values(
    ascending=False)
print(correlations)

numeric_df = df.select_dtypes(include='int64')
corr_matrix = numeric_df.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, cmap='PuBuGn', fmt="
    .2f")
plt.title('Correlation Matrix of Banking Dataset wrt
    Subscription')
plt.show()
```

18.2 Output

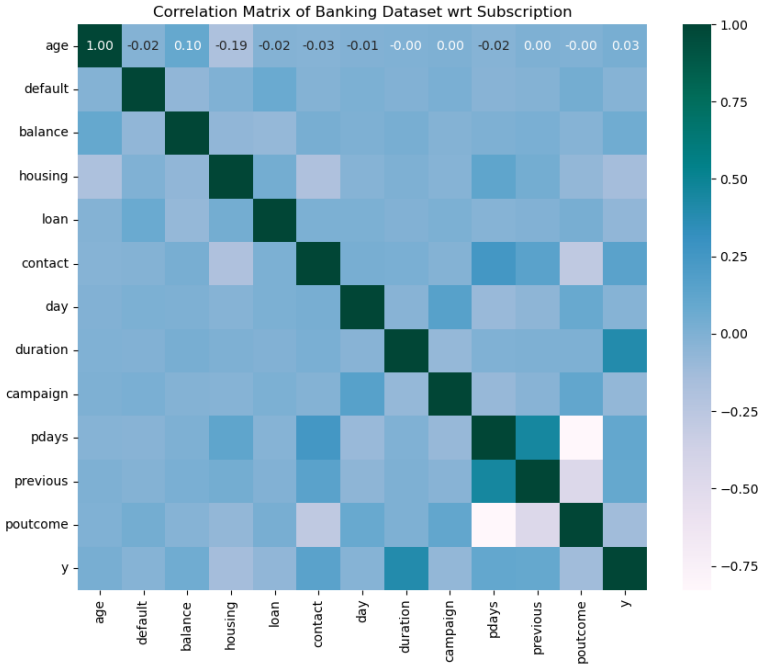

Figure 18.1: Output

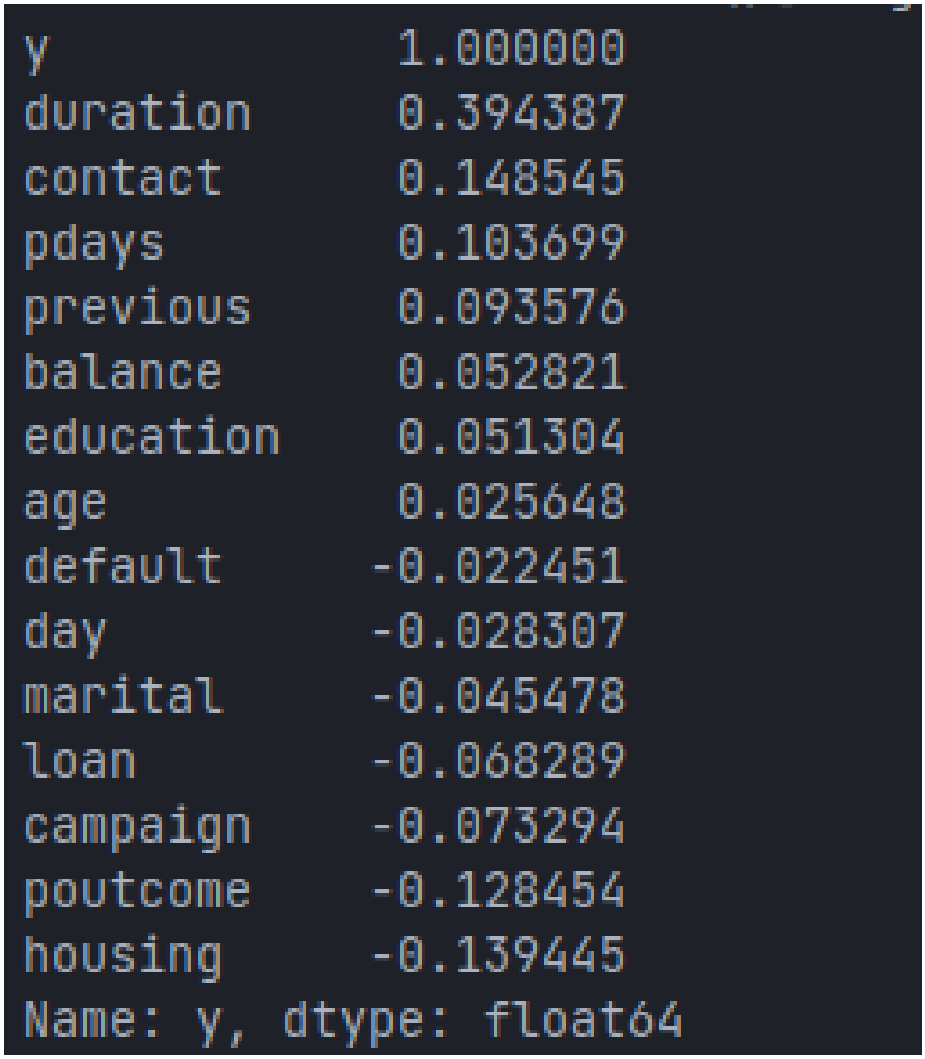

Figure 18.2: Output