- Bộ dữ liệu ghi lại các cuộc gọi điện thoại dựa trên các chiến dịch Marketing. Dữ liệu ghi lại khách hàng có hay không đăng ký dịch vụ tiền gửi có kì hạn ngân hàng;
- Bộ dữ liệu gồm 1 file CSV với 41188 bản ghi và 21 cột (20 cột tham số đầu vào và 1 cột y là dự đoán kết quả đầu ra);
- Bộ dữ liệu từ tháng 5/2008 đến tháng 11/2010;
- Mục tiêu phân loại là dự đoán liệu khách hàng có đăng ký (yes/no) một khoản tiền gửi có kỳ hạn (y) hay không.

Metadata

STT	Cột	Dạng dữ liệu	Mô tả
1	age	numeric	Tuổi
2	job	categorical	admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown'
3	marital	categorical	divorced','married','single','unknown'; note: 'divorced' means divorced or widowed
4	education	categorical	'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course', 'university.degree','unknown'
5	default	categorical	'no','yes','unknown'
6	housing	categorical	'no','yes','unknown'
7	loan	categorical	'no','yes','unknown'
8	contact	categorical	cellular','telephone'
9	month	categorical	'jan', 'feb', 'mar', ..., 'nov', 'dec'
10	day_of_week	categorical	mon','tue','wed','thu','fri'
11	duration	numeric	Thời gian gọi điện tính theo giây
12	campaign	numeric	số lượng liên hệ được thực hiện trong chiến dịch này và cho khách hàng này

13	pdays	numeric	số ngày trôi qua sau khi khách hàng được liên hệ lần cuối từ một chiến dịch trước đó (số; 999 có nghĩa là khách hàng chưa được liên hệ trước đó)
14	previous	numeric	số lượng địa chỉ liên hệ được thực hiện trước chiến dịch này và cho khách hàng này
15	poutcome	categorical	'failure','nonexistent','success'
16	emp.var.rate	numeric	tỷ lệ thay đổi việc làm - chỉ báo hàng quý
17	cons.price.idx	numeric	chỉ số giá tiêu dùng - chỉ báo hàng tháng (số)
18	cons.conf.idx	numeric	chỉ số niềm tin của người tiêu dùng - chỉ số hàng tháng (số)
19	euribor3m	numeric	lãi suất 3 tháng của euribor
20	nr.employed	numeric	số lượng nhân viên - chỉ số hàng quý (số)
21	y	binary	khách hàng đã đăng ký tiền gửi có kỳ hạn chưa? (y/n)

Tool

- Ta tiến hành load dữ liệu vào **Python** để phân tích
- Bộ dữ liệu gồm có 5 cột dạng **int64**, 11 cột có giá trị dạng **object**, 5 cột giá trị dạng **float64**;

- Bộ dữ liệu gồm có 5 cột dạng **int64**, 11 cột có giá trị dạng **object**, 5 cột giá trị dạng **float64**;

```
In [3]: #thông tin chung
        bank_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41188 entries, 0 to 41187
Data columns (total 21 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  emp.var.rate    41188 non-null  float64
 16  cons.price.idx  41188 non-null  float64
 17  cons.conf.idx   41188 non-null  float64
 18  euribor3m       41188 non-null  float64
 19  nr.employed     41188 non-null  float64
 20  y               41188 non-null  object 
dtypes: float64(5), int64(5), object(11)
memory usage: 6.6+ MB
```

- Không có cột nào bị thiếu dữ liệu. Không có dữ liệu null/nan.

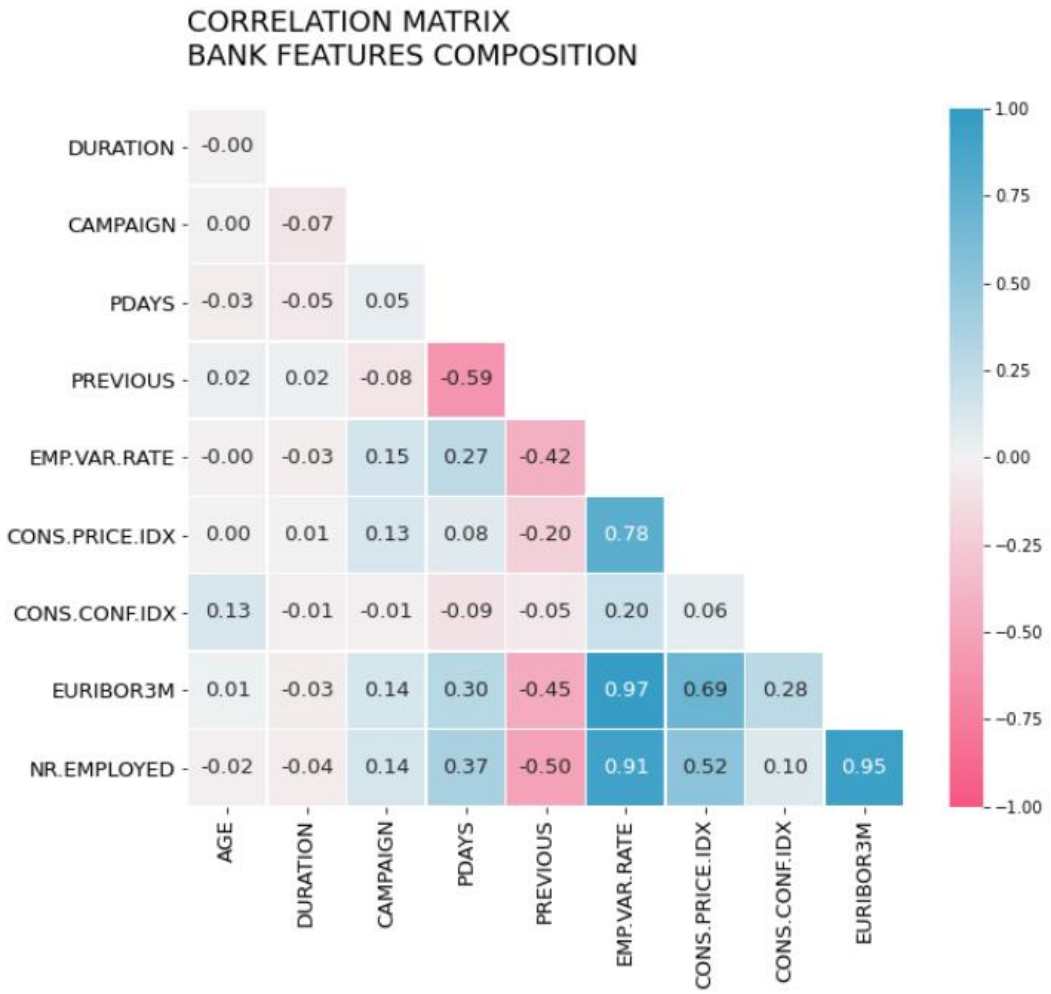
```
In [5]: # thống kê các cột thiếu bao nhiêu giá trị
        for col in bank_df.columns:
            missing_data = bank_df[col].isna().sum()
            missing_percent = missing_data/len(bank_df)*100
            print(f"Column {col}: has {missing_percent} % missing_data")
```

```
Column age: has 0.0 % missing_data
Column job: has 0.0 % missing_data
Column marital: has 0.0 % missing_data
Column education: has 0.0 % missing_data
Column default: has 0.0 % missing_data
Column housing: has 0.0 % missing_data
Column loan: has 0.0 % missing_data
Column contact: has 0.0 % missing_data
Column month: has 0.0 % missing_data
Column day_of_week: has 0.0 % missing_data
Column duration: has 0.0 % missing_data
Column campaign: has 0.0 % missing_data
Column pdays: has 0.0 % missing_data
Column previous: has 0.0 % missing_data
Column poutcome: has 0.0 % missing_data
Column emp.var.rate: has 0.0 % missing_data
Column cons.price.idx: has 0.0 % missing_data
Column cons.conf.idx: has 0.0 % missing_data
Column euribor3m: has 0.0 % missing_data
Column nr.employed: has 0.0 % missing_data
Column y: has 0.0 % missing_data
```

EDA: Exploratory Data Analysis

Numberic Correlation

- Các biến tương quan dưới đây thuộc các biến trong nhóm bối cảnh kinh tế xã hội.



Qua biểu đồ ở trên ta có thể thấy các features thuộc bối cảnh kinh tế xã hội có liên quan trực tiếp đến nhau. Ví dụ như:

- `emp.var.rate` (tỷ lệ biến động việc làm) tương quan với `cons.price.idx` (chỉ số giá tiêu dùng), `euribor3m` (lãi suất chào bán liên ngân hàng euribor) và nr.employed (số lượng nhân viên).

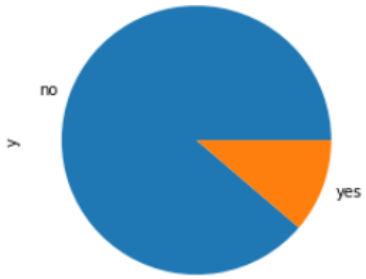
- `cons.price.idx` (chỉ số giá tiêu dùng) cũng tương quan với `euribor3m` (lãi suất chào bán liên ngân hàng euribor) và `nr.employed` (số lượng nhân viên).
- `euribor3m` (lãi suất chào bán liên ngân hàng euribor) có sự tương quan lớn với `nr.employed` (số lượng nhân viên).

Bank client data

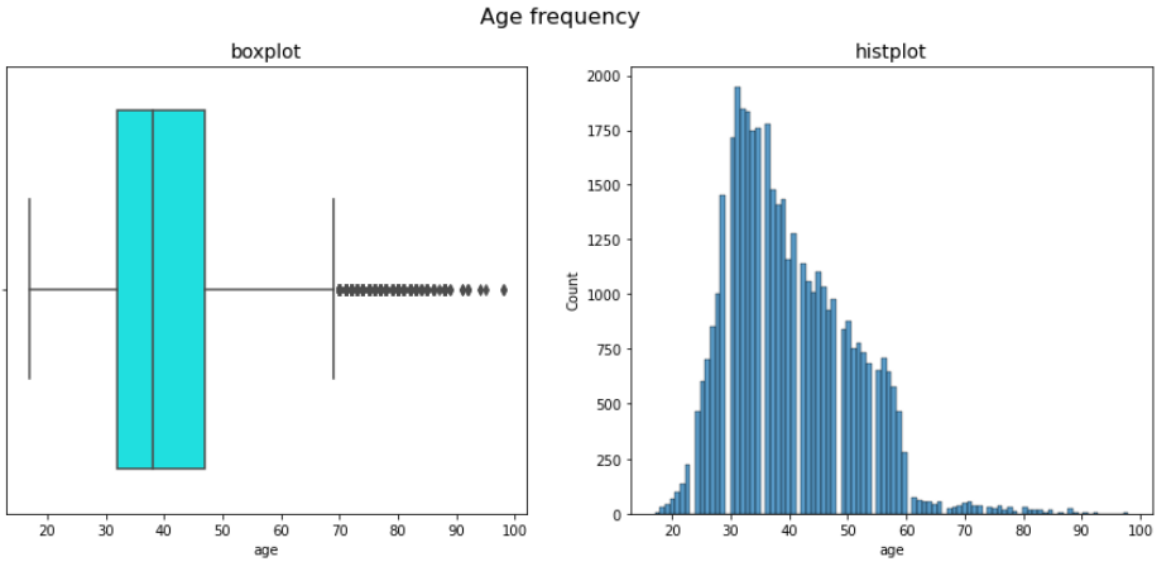
1. Biến y

- Số khách hàng đồng ý ít hơn số khách hàng từ chối dịch vụ tiền gửi.

```
In [23]: bank_df['y'].value_counts().plot.pie()
Out[23]: <AxesSubplot:ylabel='y'>
```



2. Age

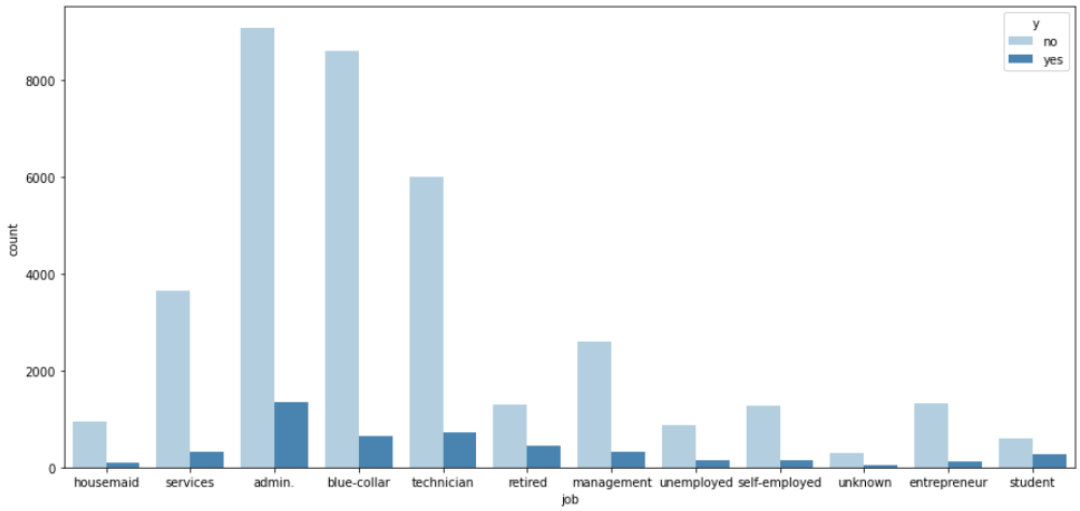


- Ta có thể thấy độ tuổi phổ biến trong data chương trình tiền gửi có kỳ hạn ngân hàng nhiều nhất trong khoảng từ 30 đến 48 tuổi - ta có thể thấy đây là

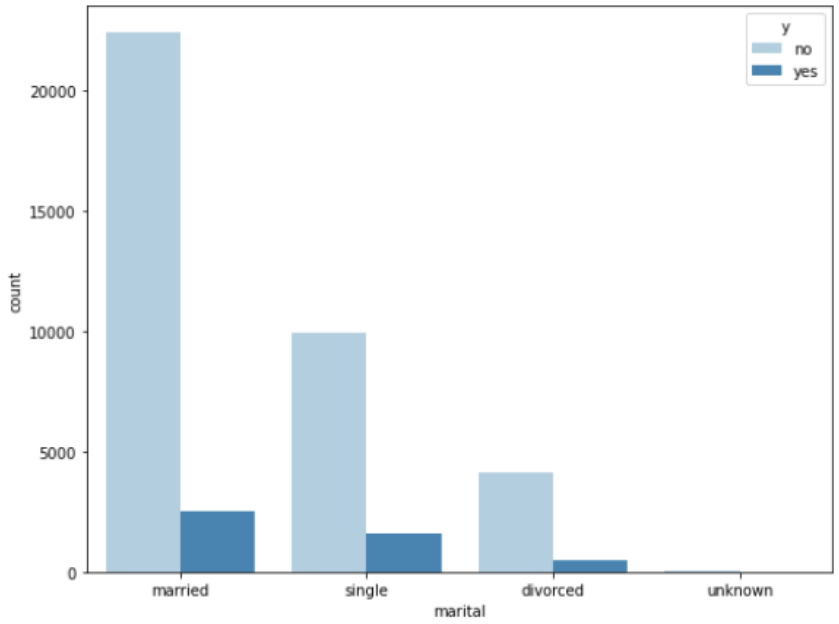
độ tuổi lao động chính trong xã hội. Ngoài ra có các trường hợp đặc biệt trong data là trên 70 tuổi.

3. Job

- **admin.** là công việc phổ biến nhất trong tập dữ liệu khách hàng và cũng tham gia vào chương trình tiền gửi nhiều nhất. Thấp nhất là **học sinh, sinh viên** và **các ngành nghề khác.**



4. Marital




```
In [29]: # % tình trạng hôn nhân
         bank_df["marital"].value_counts(normalize = True)

Out[29]: married     0.605225
         single      0.280859
         divorced    0.111974
         unknown     0.001942
         Name: marital, dtype: float64
```

- Tỉ lệ đã có gia đình trong tập khách hàng khá cao chiếm khoảng 60%. Tập hợp khách hàng này thường hướng đến trách nhiệm và tính ổn định lâu dài. Vì vậy nhu cầu của nhưng khách hàng này là quan tâm đến các khoảng tiết kiệm và lãi suất tiền gửi trong kỳ.

5. Education

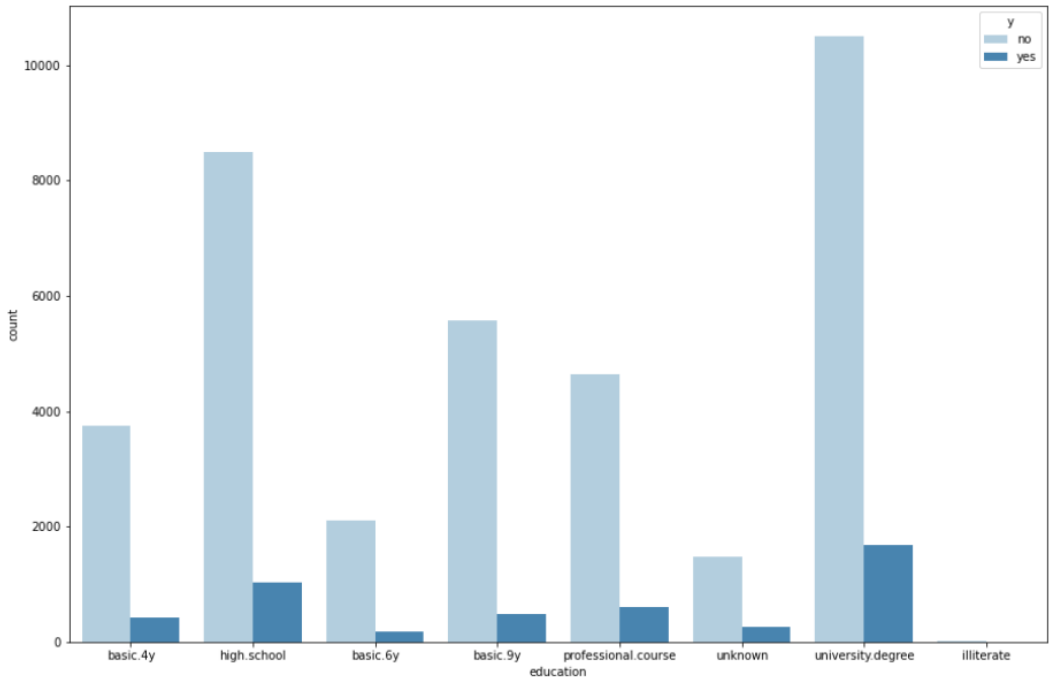
- Tình trạng học vấn của khách hàng trong bộ data set đa phần đã có bằng đại học.

In [30]:
```
# Tình trạng học vấn
bank_df['education'].value_counts().to_frame()
```

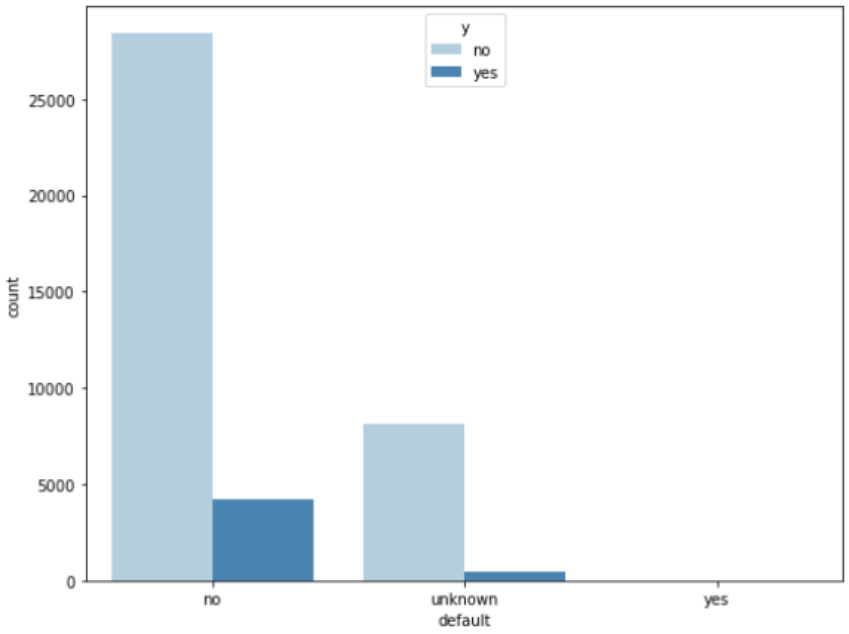
Out[30]:

	education
university.degree	12168
high.school	9515
basic.9y	6045
professional.course	5243
basic.4y	4176
basic.6y	2292
unknown	1731
illiterate	18

Out[31]: <AxesSubplot:xlabel='education', ylabel='count'>

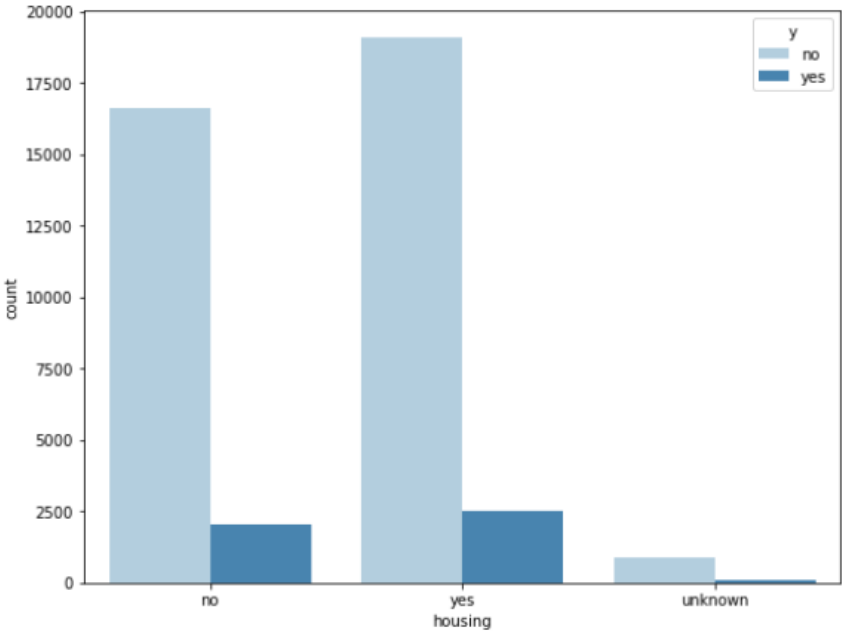


6. Default



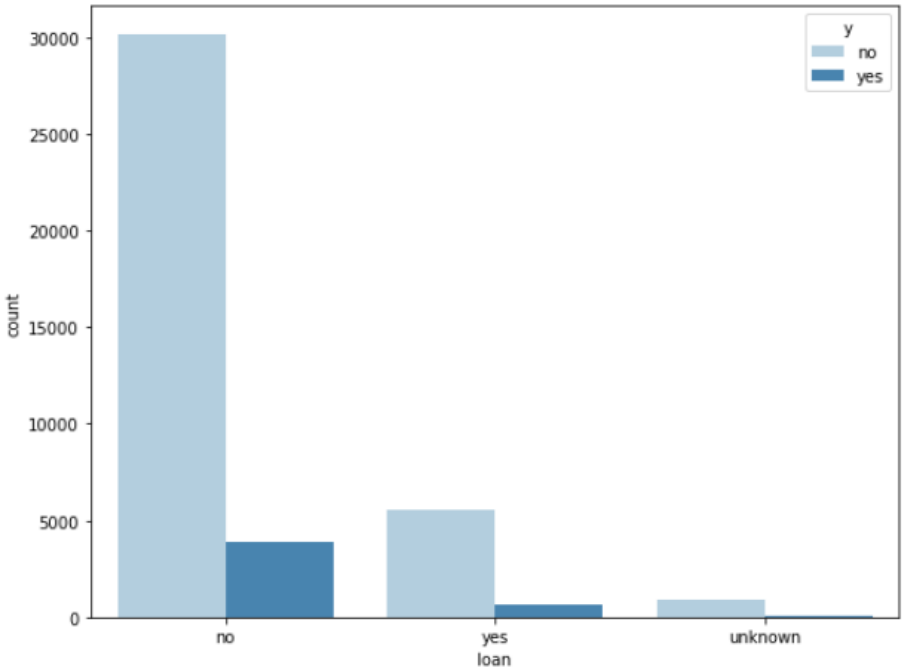
- Tệp khách hàng đã được lọc kĩ từ đầu vào nên gần như ko có khách hàng vỡ nợ tín dụng.

7. Housing



- Tỉ lệ khách hàng tham gia chương trình hoặc không khi có khoảng vay mua nhà có tỉ lệ tương đương với nhau.

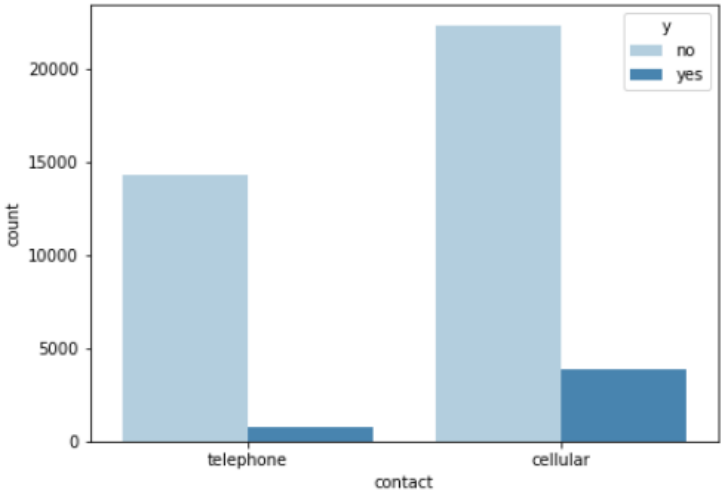
8. Loan



- Các khách hàng tham gia chương trình thường không có các khoảng vay các nhân do họ đang tìm đến những chương trình gửi tiết kiệm

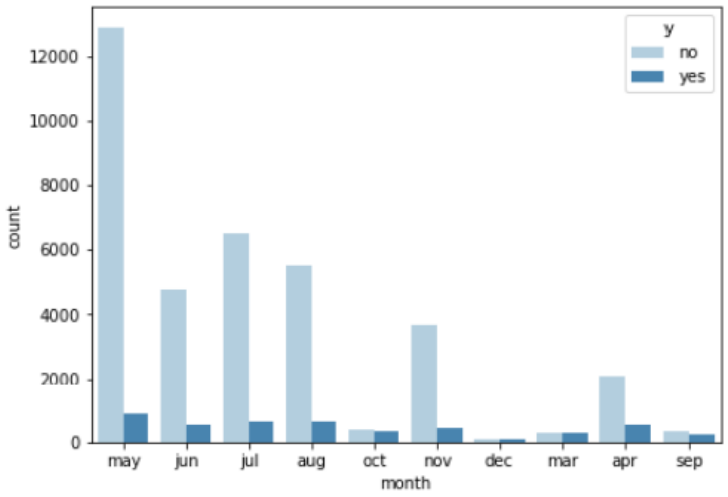
Related with the last contact of the current campaign

9. Contact



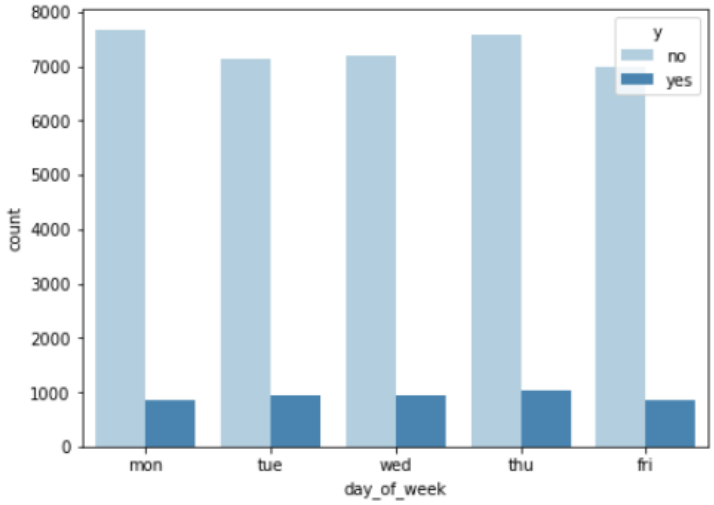
- Số lượng khách hàng liên lạc bằng điện thoại di động nhiều hơn lượt khác hàng sử dụng máy cố định.

10. Month



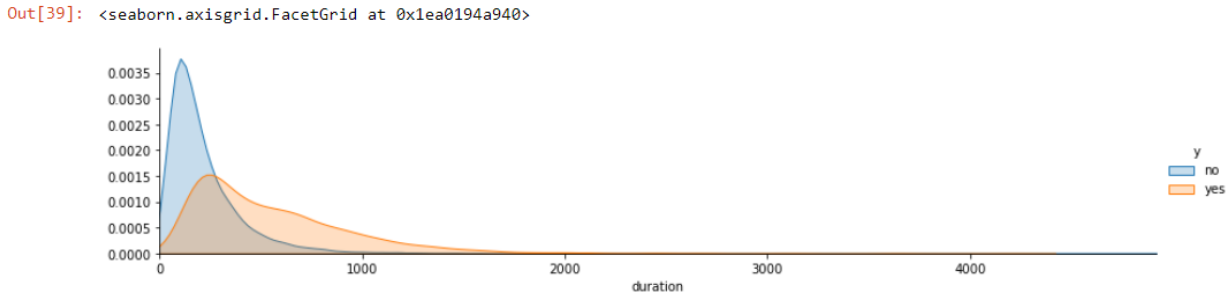
- Người dùng có xu hướng đăng kí tham gia nhiều nhất vào tháng 4 đến tháng 8.

11. Day_of_week



- Thời gian người dùng đăng kí dịch vụ tiền gửi kì hạn ngân hàng trong 1 tuần là như nhau.

12. Duration

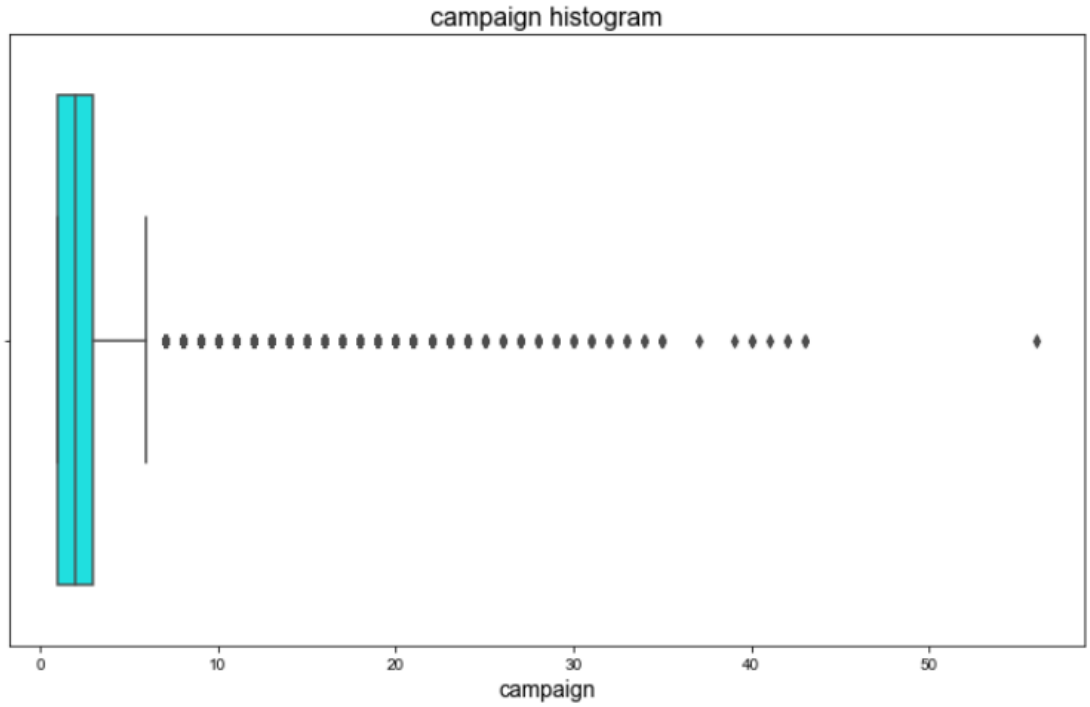


Thời lượng liên lạc:

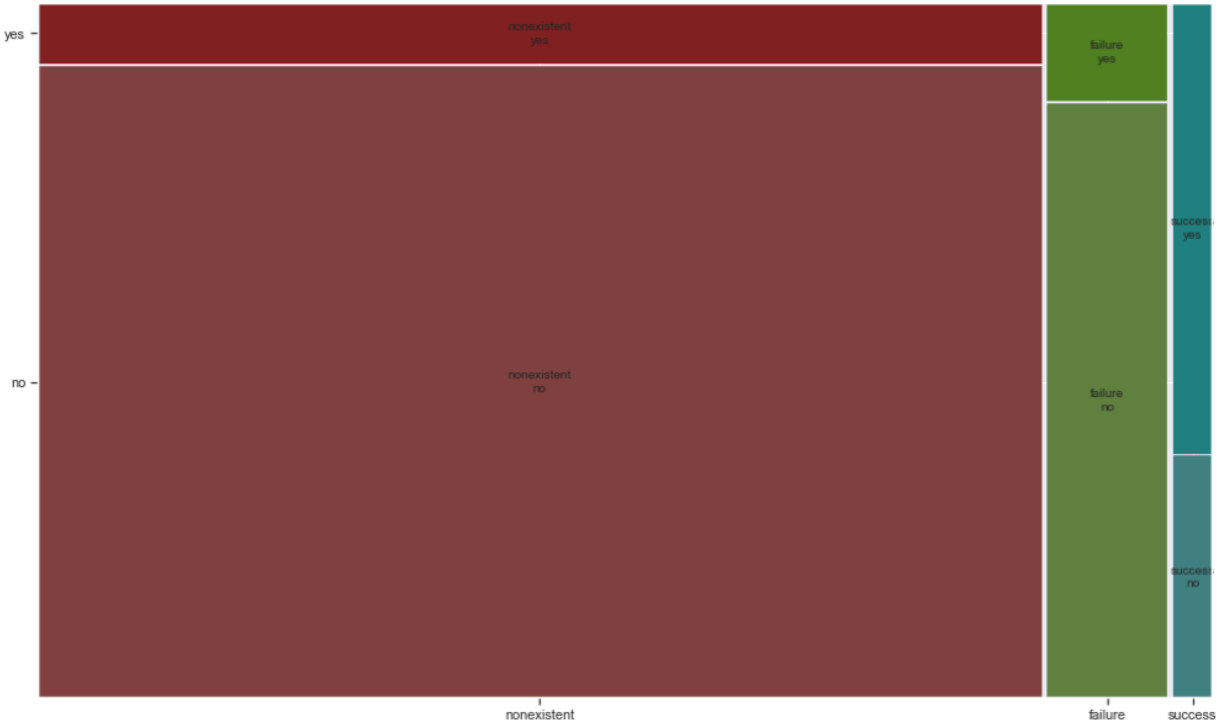
- duration: thời lượng liên lạc cuối cùng, tính bằng giây (số). Lưu ý quan trọng: thuộc tính này ảnh hưởng nhiều đến mục tiêu đầu ra (ví dụ: nếu thời lượng = 0 thì y = 'không'). Tuy nhiên, thời lượng không được biết trước khi thực hiện cuộc gọi. Ngoài ra, sau khi kết thúc cuộc gọi, y hiển nhiên được biết đến. Do đó, đầu vào này chỉ nên được đưa vào cho mục đích chuẩn và nên bị loại bỏ nếu mục đích là có một mô hình dự đoán thực tế.

Other attributes

13. Campaign



14. Poutcome



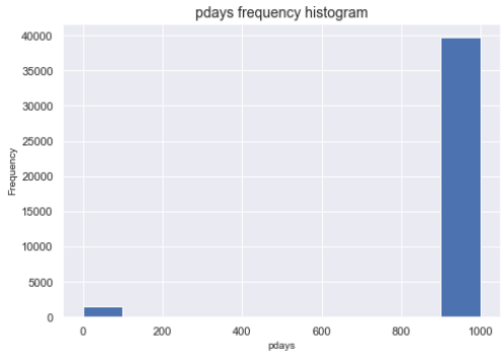
- Các chiến dịch marketing thành công mang đến số khách hàng đồng ý đăng kí tham gia dịch vụ.

15. Pdays

```
In [43]: bank_df['pdays'].value_counts().to_frame()
```

Out[43]:

	pdays
999	39673
3	439
6	412
4	118
9	64
2	61
7	60
12	58
10	52
5	46
13	36
11	28
1	26
15	24
14	20
8	18
0	15
16	11
17	8
18	7
19	3
22	3
21	2
20	1
25	1
26	1
27	1



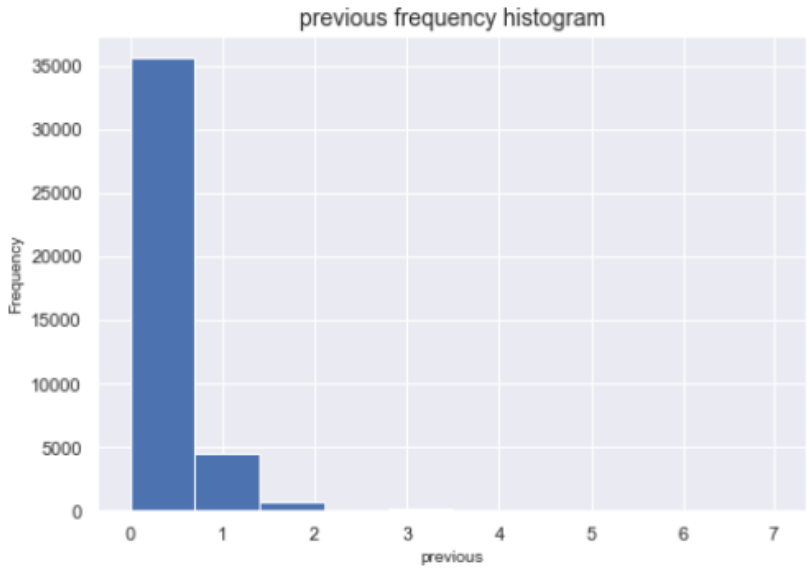
- Số khách hàng trong bộ dữ liệu đa phần chưa được liên hệ trước đó và đều là các data mới.

16. Previous

```
In [44]: bank_df['previous'].value_counts().to_frame()
```

Out[44]:

	previous
0	35563
1	4561
2	754
3	216
4	70
5	18
6	5
7	1



Data Pre-processing

- Bước này ta sẽ xử lý dữ liệu để tiến hành đưa vào mô hình

In [46]:
```
# lọc trùng dữ liệu
bank_df.duplicated().sum()
```

Out[46]: 12

In [47]:
```
# Bỏ các dòng dữ liệu trùng
bank_df = bank_df.drop_duplicates()
```

In [48]:
```
# Kiểm tra dữ liệu
bank_df.duplicated().sum()
```

Out[48]: 0

In [49]:
```
# Data binning
d = {"no": 0, "yes": 1}
bank_df["y"] = bank_df["y"].map(d)
bank_df.head()
```

In [54]:
```
import warnings
warnings.filterwarnings('ignore')
#xóa các cột không ảnh hưởng đến việc dự đoán
bank_df=bank_df.drop(["month","day_of_week","contact"],axis=1)

#Thay các giá trị trong cột education thành dạng số
bank_df["education"]=bank_df["education"].replace(['basic.4y','high.school','basic.6y','basic.9y','professional.course','universi

#Thay các giá trị trong cột housing thành dạng số
bank_df.housing[bank_df['housing']=='no']=0
bank_df.housing[bank_df['housing']=='yes']=1
bank_df.housing[bank_df['housing']=='unknown']=-1

#Thay các giá trị trong cột loan thành dạng số
bank_df.loan[bank_df['loan']=='no']=0
bank_df.loan[bank_df['loan']=='yes']=1
bank_df.loan[bank_df['loan']=='unknown']=-1
# bank_df["Loan"]=bank_df["Loan"]+bank_df["housing"]
#bank_df.drop("housing",axis=1)

#Job
bank_df["job"]=bank_df["job"].replace(['unknown','unemployed','entrepreneur','blue-collar','technician','services','admin.','mana

bank_df["default"]=bank_df["default"].replace(['no','yes','unknown'],[0,1,-1])

bank_df["poutcome"]=bank_df["poutcome"].replace(['nonexistent','failure','success'],[0,-1,1])

bank_df['pdays'] = bank_df['pdays'].apply(lambda x: 0 if x==999 else(20 if x<=10 else(6 if x<=20 else 3)))
```

Outlier

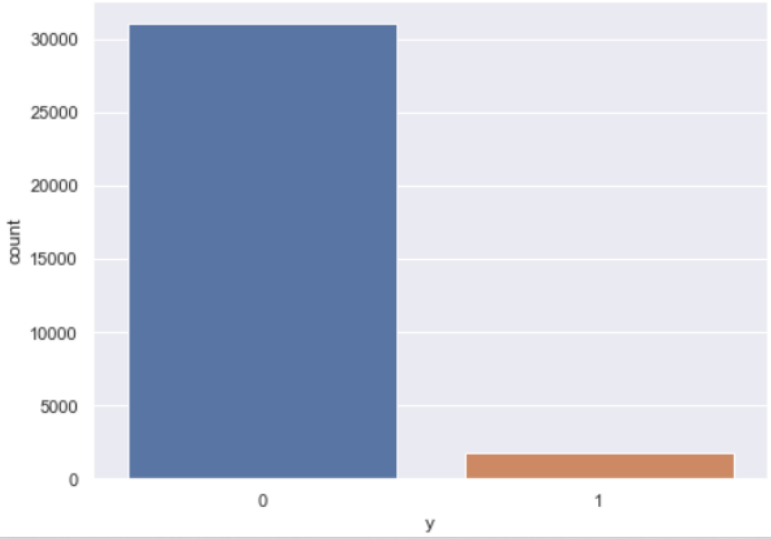
In [57]:
```
# xử lý outlier để tăng độ chính xác của mô hình
plt.figure(figsize=(14,6))
bank_df.boxplot()
print()
```

Data Balancing

Cân bằng dữ liệu để tăng độ chính xác cho mô hình phân loại với biến Y có 2 giá trị là yes: 1 và no: 0

In [63]:
```
# Data Balancing
fig, ax = plt.subplots(figsize=(7,5))
sns.countplot(bank_df['y'])
```

Out[63]:
```
<AxesSubplot:xlabel='y', ylabel='count'>
```



Business question

- Qua tính toán ta thấy tỷ lệ thu hút khách hàng của các chiến dịch MKT là 11.3%. Đây là một con số khá thấp so với một ngân hàng.

In [51]:
```
# Điều gì thu hút khách hàng
bank_df[bank_df["y"] == 1].mean()
```

Out[51]:
```
age               40.912266
duration         553.256090
campaign           2.051951
pdays            791.990946
previous           0.492779
emp.var.rate      -1.233089
cons.price.idx    93.354577
cons.conf.idx    -39.791119
euribor3m          2.123362
nr.employed     5095.120069
y                  1.000000
dtype: float64
```

- Độ tuổi trung bình của khác hàng đồng ý đăng kí dịch vụ là 40 và cần ít nhất là 2 cuộc gọi để có thể thuyết phục họ.

- Thời gian trung bình một cuộc gọi thu hút được khác hàng đăng kí dịch vụ là 9.0 phút 13 giây.

```
In [53]: #Bảng tổng hợp độ tuổi, công việc và thời gian đăng kí dịch vụ của khách hàng
bank_df.pivot_table(
    ["age", "duration"],
    ["job"],
    aggfunc = "mean",
).head(10)
```

Out[53]:

job	age	duration
admin.	38.186870	254.315961
blue-collar	39.555820	264.557549
entrepreneur	41.723214	263.267857
housemaid	45.500000	250.454717
management	42.362859	257.058140
retired	62.037253	273.909779
self-employed	39.949331	264.142153
services	37.925637	258.491303
student	25.894857	283.683429
technician	38.508681	250.287431

- Khách hàng là học sinh, sinh viên và người nghỉ hưu thường có thời gian nghe điện thoại lâu nhất.

Data Model

- Bộ dataset có đầu ra là biến y với kết quả là yes/no có thể chuyển đổi thành hệ nhị phân ta lựa chọn mô hình ML classification là Decision Tree để dự đoán kết quả đầu ra. Ngoài ra trong bài làm cũng sử dụng mô hình K-Nearest Neighbors với K bằng 2 để dự đoán biến y đầu ra.
- Ở mô hình K-NN với K bằng 2 --> Mô hình đạt chính xác 97%
- Ở mô hình Decision Tree cũng cho ra độ chính xác 97%, Confusion matrix cũng cho thấy kết quả dự đoán khá chính xác với tỉ lệ dự đoán sai các khác hàng đã chọn đăng kí dịch vụ gần như bằng không.

```
Confusion Matrix
[[7783    0]
 [ 316 7424]]

Accuracy: 0.9796

Classification Report
              precision    recall  f1-score   support

           1       0.96      1.00      0.98      7783
           0       1.00      0.96      0.98      7740

    accuracy                           0.98     15523
   macro avg       0.98      0.98      0.98     15523
weighted avg       0.98      0.98      0.98     15523
```

- Chỉ số F1 – score và Avg accuracy cũng cho kết quả gấn đến 1 ☮Mô hình có thể sử dụng để dự đoán khách hàng tham gia dịch vụ tiền gửi có kỳ hạn của ngân hàng.

Em xin chân thành cảm ơn!

namthanhpham111@gmail.com|0964774296