Bank Marketing case

Maria Ovchinnikova

DataSet

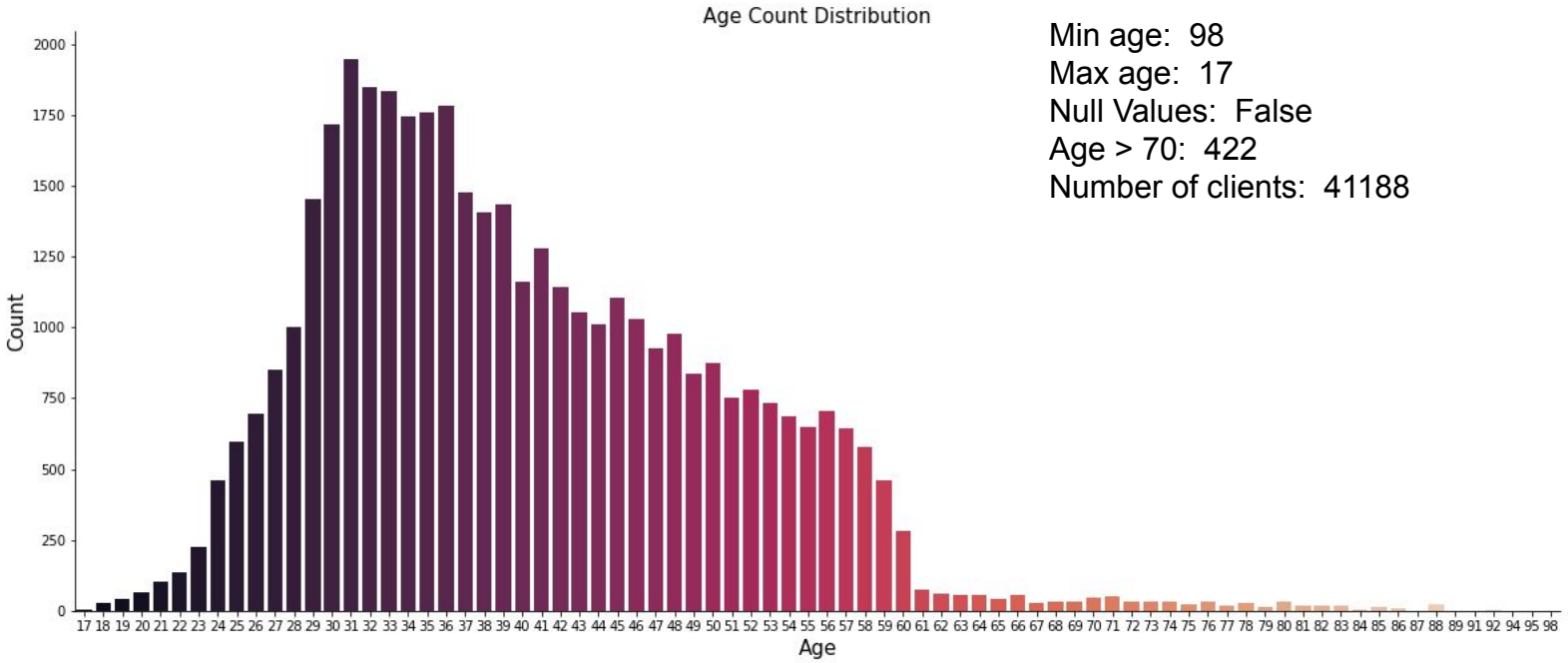
Bank clients - Age, Jobs, Marital, Education, Default, Housing, Loan

Campaign data - Call duration, Contact, Month, Day of Week

Other - …

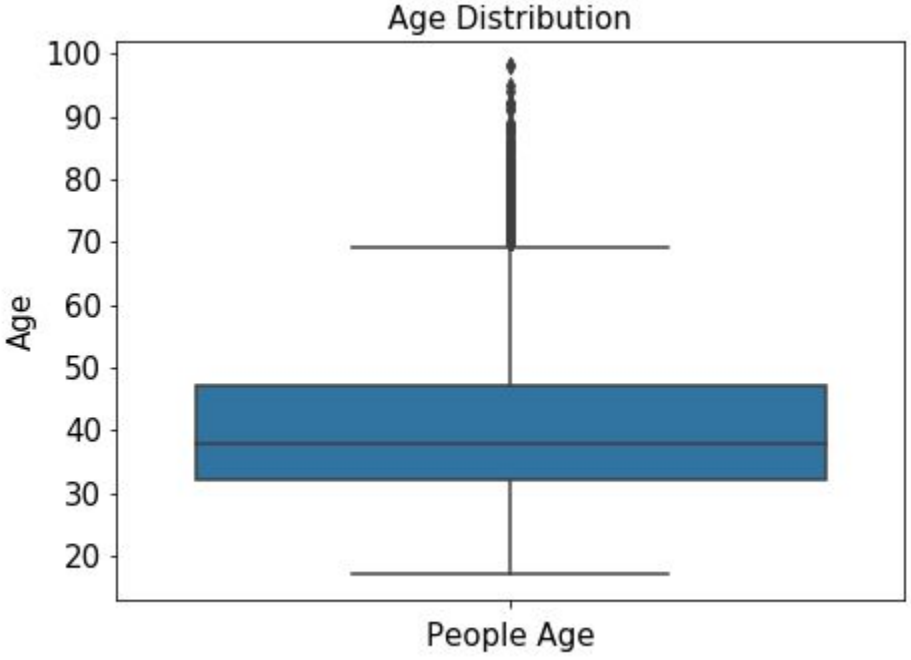
```
RangeIndex: 41188 entries, 0 to 41187
Data columns (total 21 columns):
age             41188 non-null int64
job             41188 non-null object
marital         41188 non-null object
education       41188 non-null object
default         41188 non-null object
housing         41188 non-null object
loan            41188 non-null object
contact         41188 non-null object
month           41188 non-null object
day_of_week     41188 non-null object
duration        41188 non-null int64
campaign        41188 non-null int64
pdays           41188 non-null int64
previous        41188 non-null int64
poutcome        41188 non-null object
emp.var.rate    41188 non-null float64
cons.price.idx  41188 non-null float64
cons.conf.idx   41188 non-null float64
euribor3m       41188 non-null float64
nr.employed     41188 non-null float64
y               41188 non-null object
dtypes: float64(5), int64(5), object(11)
```

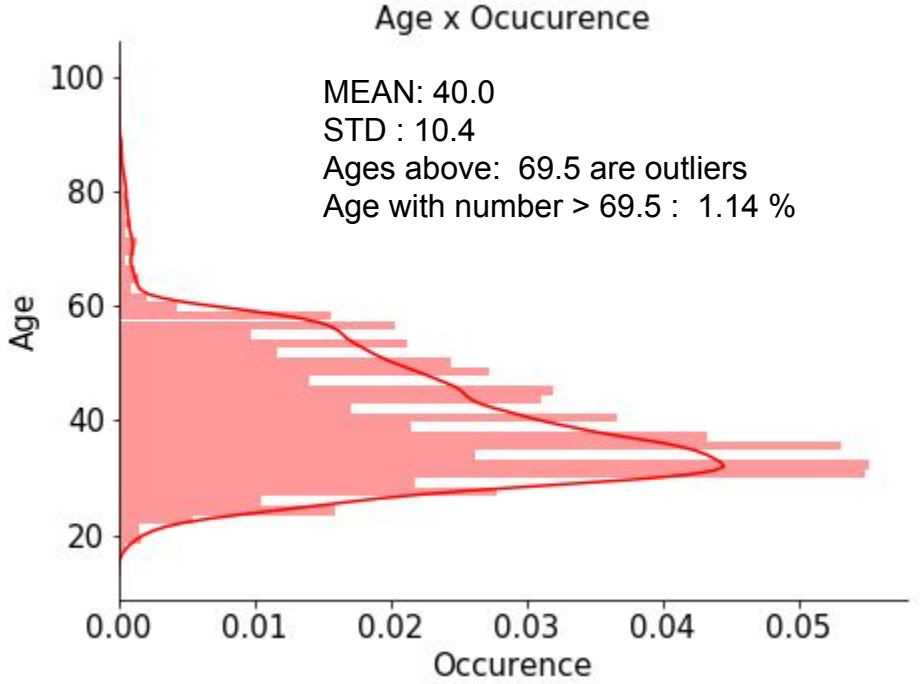
Client data - Age



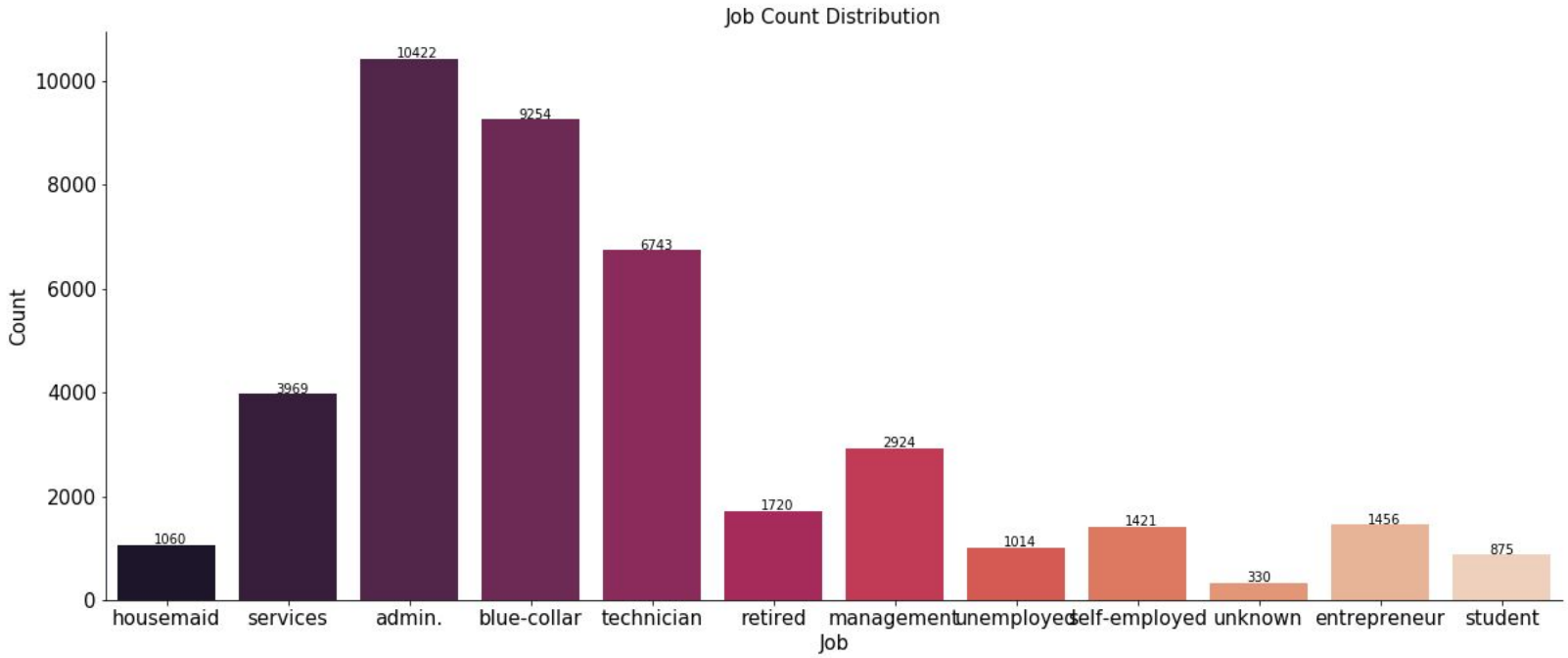
Min age: 98
Max age: 17
Null Values: False
Age > 70: 422
Number of clients: 41188

Age

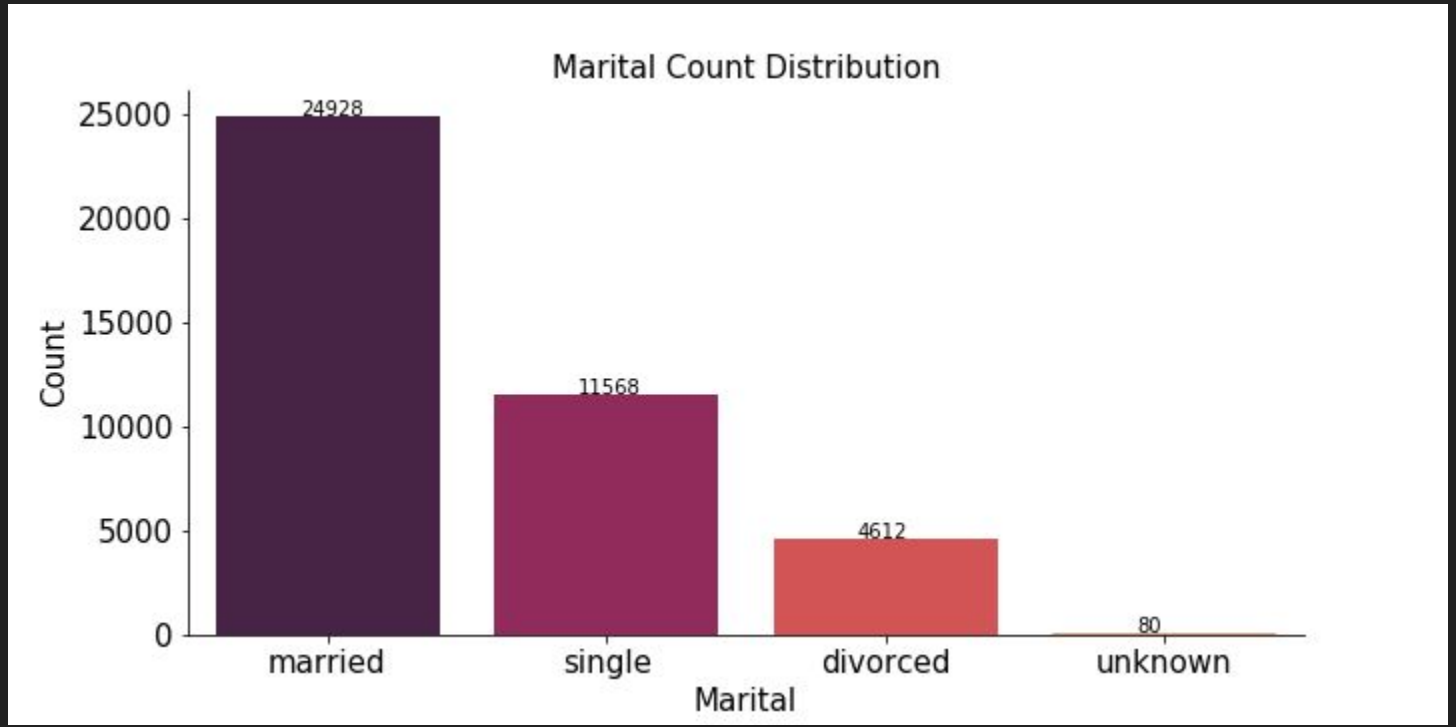

Age Distribution
100
90
80
70
60
50
40
30
20
Age
People Age


Age x Ocucurence
MEAN: 40.0
STD : 10.4
Ages above: 69.5 are outliers
Age with number > 69.5 : 1.14 %
100
80
60
40
20
Age
0.00
0.01
0.02
0.03
0.04
0.05
Occurence

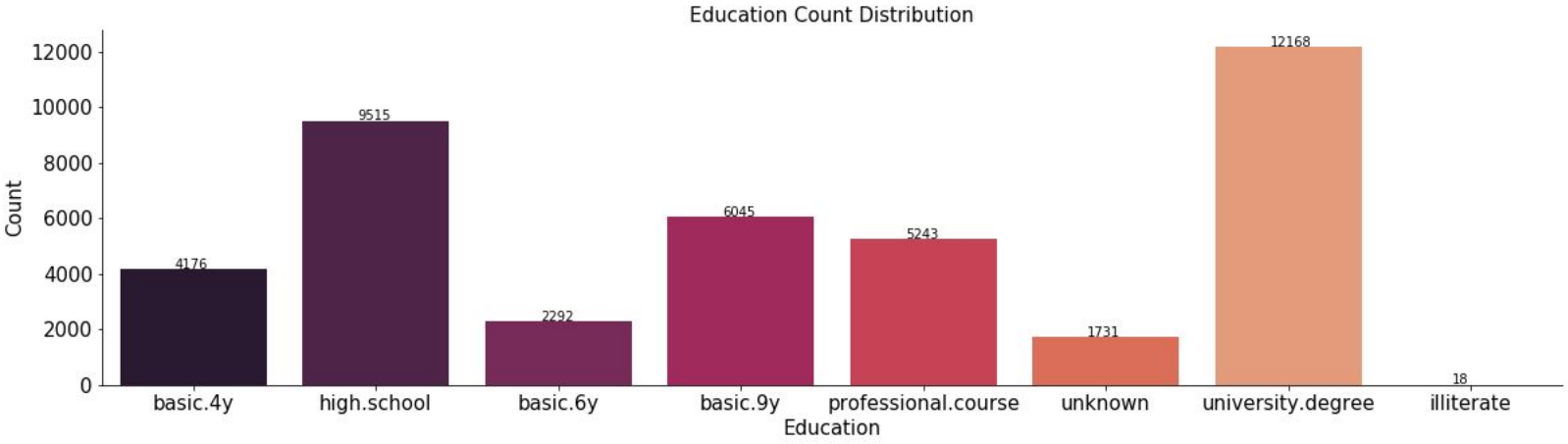
Jobs


Job Count Distribution
Count
Job
0
2000
4000
6000
8000
10000
housemaid
services
admin.
blue-collar
technician
retired
management
unemployed
self-employed
unknown
entrepreneur
student
1060
3969
10422
9254
6743
1720
2924
1014
1421
330
1456
875

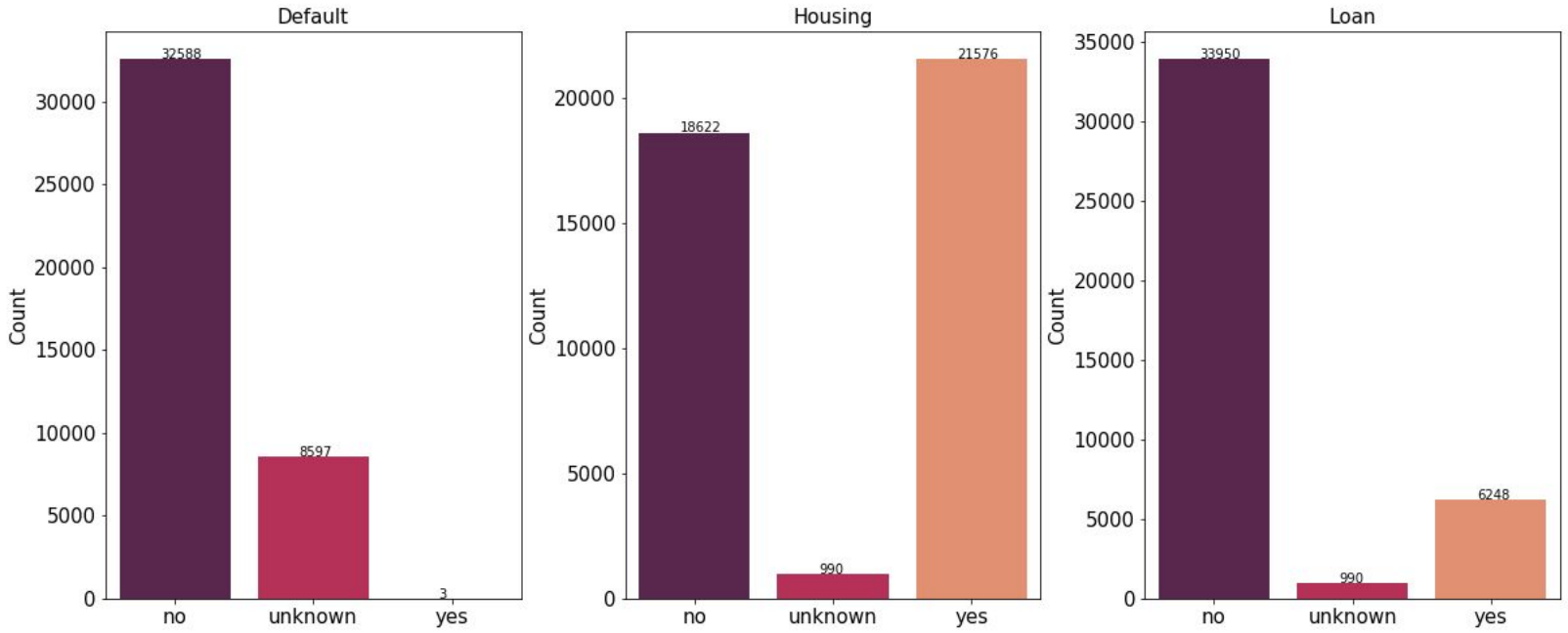
Marital


Marital Count Distribution
25000
20000
15000
10000
5000
0
Count
24928
11568
4612
80
married
single
divorced
unknown
Marital

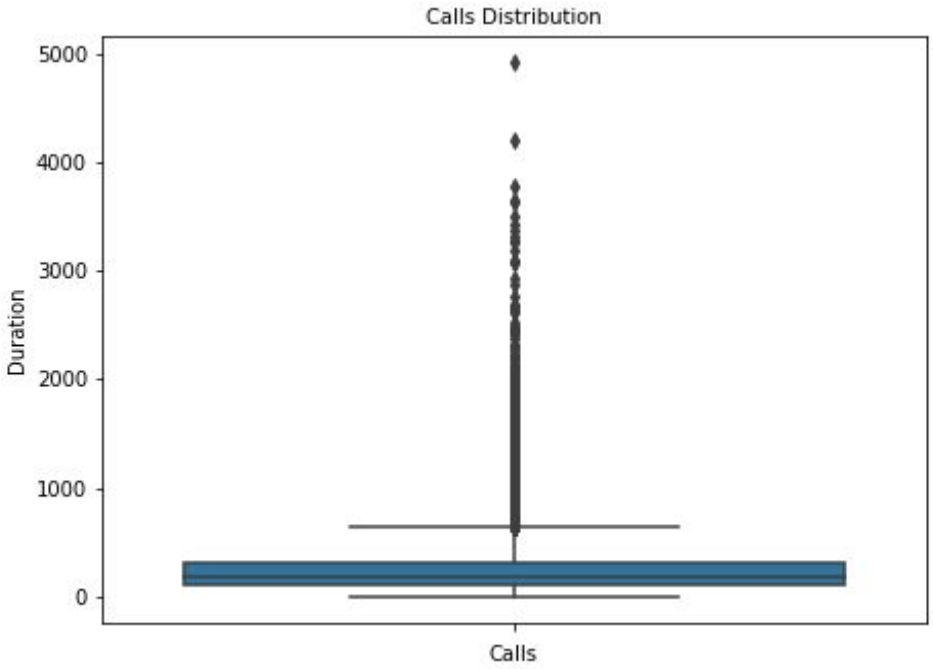
Education

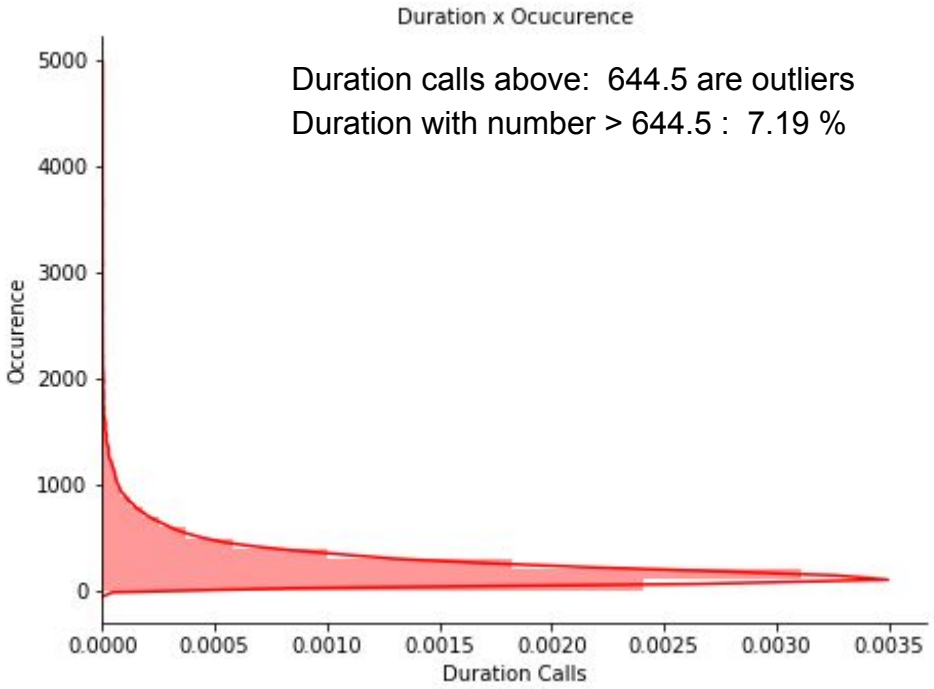

Education Count Distribution
Count
12000
10000
8000
6000
4000
2000
0
4176
9515
2292
6045
5243
1731
12168
18
basic.4y
high.school
basic.6y
basic.9y
professional.course
unknown
university.degree
illiterate
Education

Default, Housing and Loan



Campaign Data - Call duration

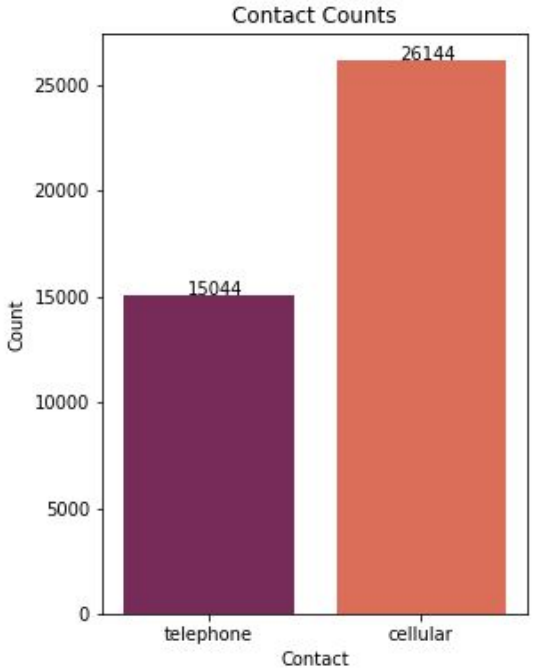


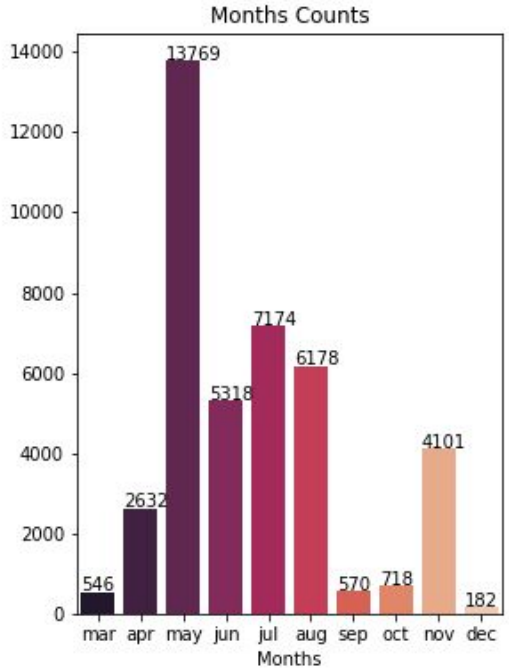


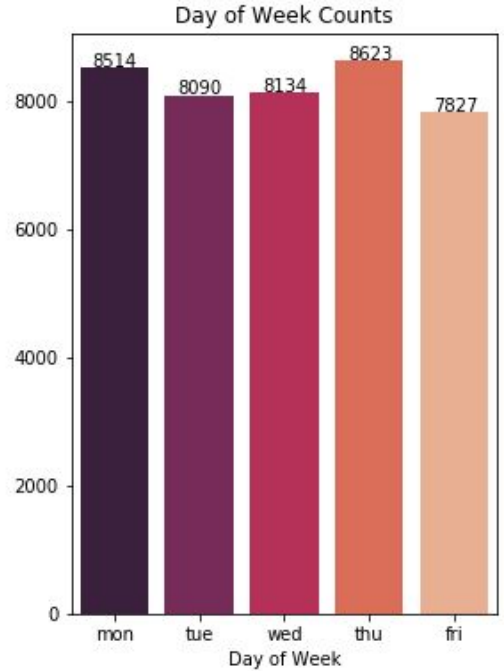
Duration calls above: 644.5 are outliers

Duration with number > 644.5 : 7.19 %

Contacts


Contact Counts
Count
25000
20000
15000
10000
5000
0
15044
26144
telephone
cellular
Contact


Months Counts
14000
12000
10000
8000
6000
4000
2000
0
546
2632
13769
5318
7174
6178
570
718
4101
182
mar
apr
may
jun
jul
aug
sep
oct
nov
dec
Months


Day of Week Counts
8000
6000
4000
2000
0
8514
8090
8134
8623
7827
mon
tue
wed
thu
fri
Day of Week

Rest

Index	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed	campaign	pdays	previous	poutcome
0	1.1	93.994	-36.4	4.857	5191	1	999	0	nonexistent
1	1.1	93.994	-36.4	4.857	5191	1	999	0	nonexistent
2	1.1	93.994	-36.4	4.857	5191	1	999	0	nonexistent
3	1.1	93.994	-36.4	4.857	5191	1	999	0	nonexistent

Index	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed	campaign ▲	pdays	previous	poutcome
0	1.1	93.994	-36.4	4.857	5191	1	999	0	1
1	1.1	93.994	-36.4	4.857	5191	1	999	0	1
2	1.1	93.994	-36.4	4.857	5191	1	999	0	1
3	1.1	93.994	-36.4	4.857	5191	1	999	0	1
4	1.1	93.994	-36.4	4.857	5191	1	999	0	1
5	1.1	93.994	-36.4	4.857	5191	1	999	0	1
6	1.1	93.994	-36.4	4.857	5191	1	999	0	1
7	1.1	93.994	-36.4	4.857	5191	1	999	0	1
8	1.1	93.994	-36.4	4.857	5191	1	999	0	1
9	1.1	93.994	-36.4	4.857	5191	1	999	0	1

Data Preparation

```
def age(dataframe):
    dataframe.loc[dataframe['age'] <= bank_client['age'].quantile(q = 0.25), 'age'] = 1
    dataframe.loc[(dataframe['age'] > bank_client['age'].quantile(q = 0.25)) & (dataframe['age']
    dataframe.loc[(dataframe['age'] > bank_client['age'].quantile(q = 0.50)) & (dataframe['age']
    dataframe.loc[(dataframe['age'] > bank_client['age'].quantile(q = 0.75)) & (dataframe['age']
    dataframe.loc[dataframe['age'] > age_outliers, 'age'] = 5
    return dataframe

age(bank_client);
```

```
# Label encoder order is alphabetical
from sklearn.preprocessing import LabelEncoder
labelencoder_X = LabelEncoder()
bank_related['contact']     = labelencoder_X.fit_transform(bank_related['contact'])
bank_related['month']       = labelencoder_X.fit_transform(bank_related['month'])
bank_related['day_of_week'] = labelencoder_X.fit_transform(bank_related['day_of_week'])
```

```
bank_final = bank_final.drop(bank_final[(bank_final['age']==5)].index)
bank_final = bank_final.drop(bank_final[(bank_final['duration'] == 5)].index)
bank_final = bank_final.drop(bank_final[(bank_final['duration'] == 0)].index)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(bank_final, y,
                                                    test_size = 0.2, random_state = 101)

from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.metrics import confusion_matrix, accuracy_score
k_fold = KFold(n_splits=10, shuffle=True, random_state=0)

from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.transform(X_test)
```

7929		2	3	0	0	0	1	4	1	4	1.4	94.465	−41.8	4.865	5228.1	3	999	0	1

−0.393213	0.912752	1.3597	−0.3616	−0.515443	−1.08872	−0.450061	1.30312	−0.0996452	−0.715684	1.498	0.830303	1.53978	−0.27718

Models

GradientBoostingClassifier
[[6849 133]
 [315 260]]
94.0

DecisionTreeClassifier
[[6639 343]
 [308 267]]
91.0

RandomForestClassifier
[[6844 138]
 [316 259]]
94.0

KNeighborsClassifier
[[6885   97]
 [391 184]]
94.0

RandomForestClassifier

	precision	recall	f1-score	support
0	0.96	0.98	0.97	6982
1	0.65	0.45	0.53	575
micro avg	0.94	0.94	0.94	7557
macro avg	0.80	0.72	0.75	7557
weighted avg	0.93	0.94	0.93	7557

API

GitLab Dockerfile API for standalone resources

Triggered docker image on Azure DevOps