भारतीय प्रबंध संस्थान कोषिक्कोड
Indian Institute of Management Kozhikode
Globalizing Indian Thought

THE RED BRICK SUMMIT

Predicting driving factors for term deposit

Submission for
IIM Ahmedabad's Blitzkrieg challenge (TRBS)

Team BLabber

Apoorv Gupta

METHODOLOGY

MISSING VALUE IMPUTATION

- As the given data has missing value >20%, removing the rows will lead to data loss. To avoid the same, lets impute data.
- 'default' is a categorical variable hence **imputing Mode** of the column, which is 'No'.
- **As fields have missing value <5% dropping the rows seems reasonable.**

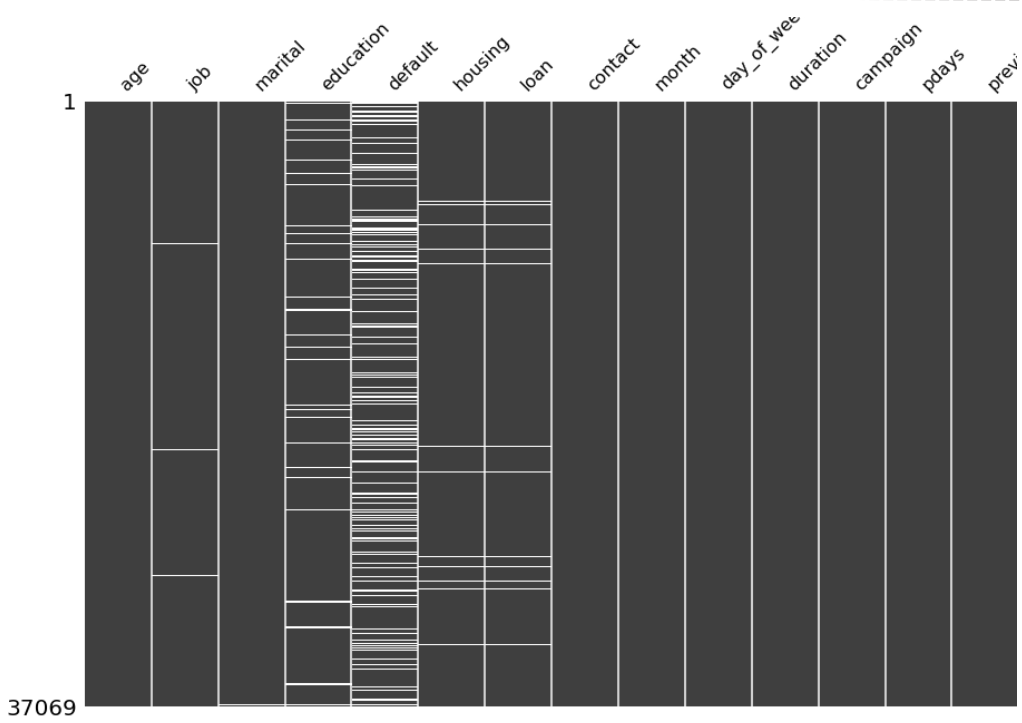
ANALYSIS

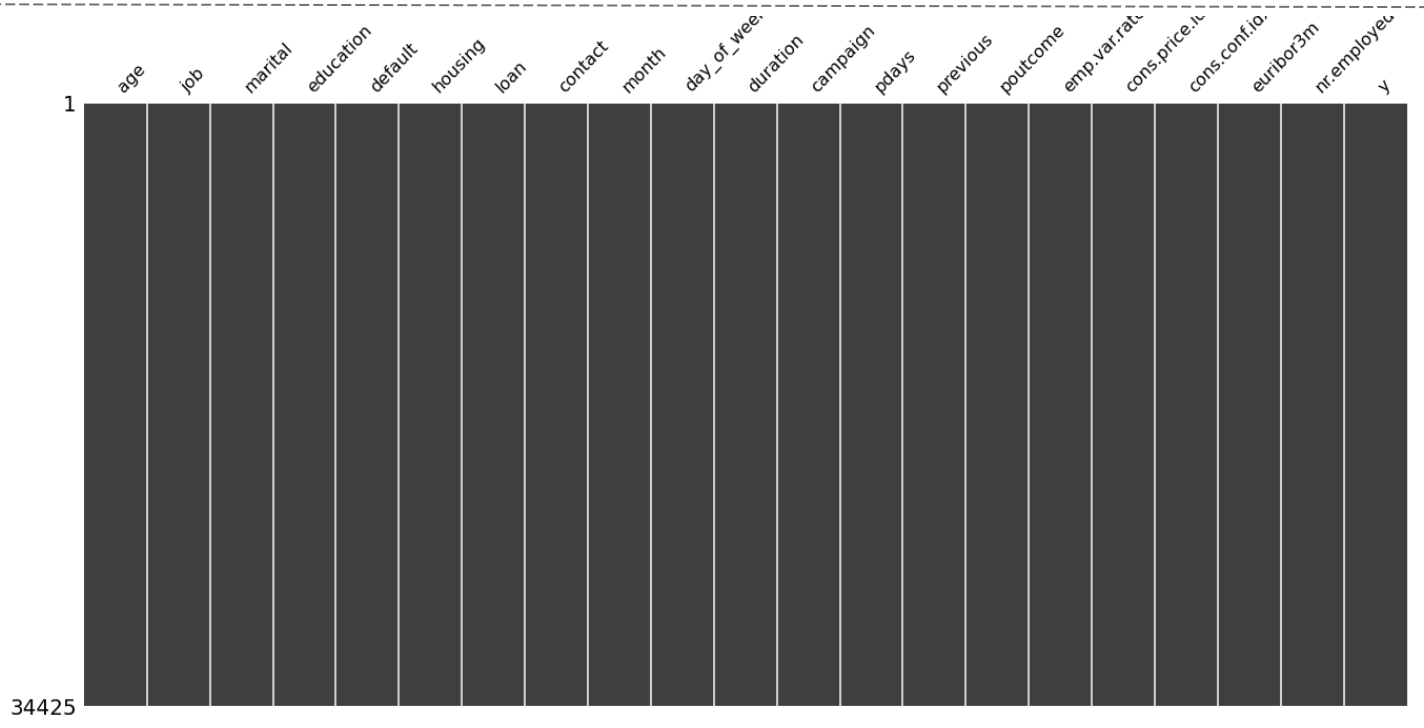
- The given data contains missing value, termed as 'unknown' in multiple parameters.
- Field **'Default' has highest missing value percentage with 21%,** followed by **'education' at 4.19%, 'loan' and 'housing' at 2.37%, 'job' and 'marital' at < 1%**

RESULTING DATA SET

- The Resulting data set post the imputations consists of **34425 rows and 21 columns with 0 missing correlation in nullity between data columns.**

RESULT AND TREATMENT





METHODOLOGY

OUTLIER ANALYSIS

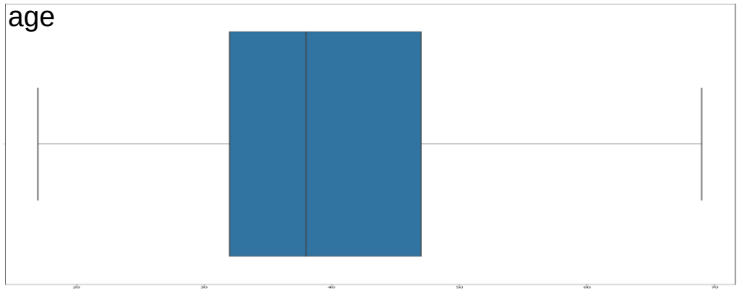
- Variable ‘cons.conf.idx’ has 1.1% outlier with count of 380
- Variable duration has 7.13% outlier with count of 2454
- Variable age has 1.05% outlier with count of 362
- Variable 'campaign' has 5.75% outlier with count of 1979

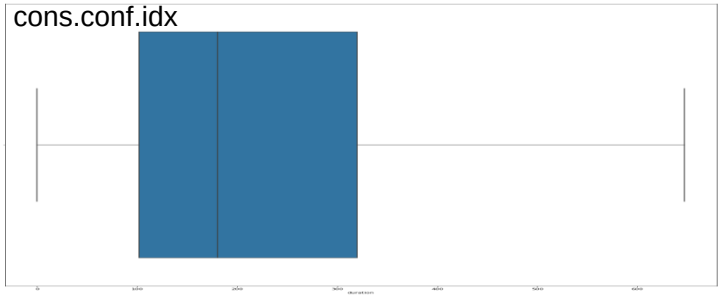
TREATMENT

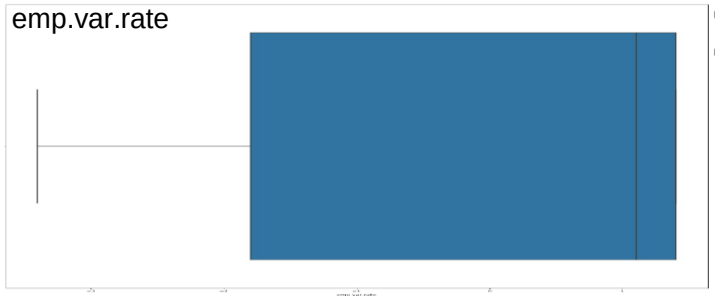
- We have to either Cap the outlier or remove the outlier
- Removing the outliers may cause loss of critical information
- Capping the outlier will cause information loss as target variable may be correlated with higher values.

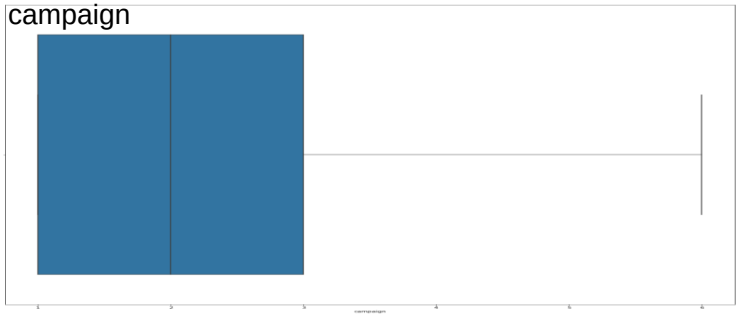
ANALYSIS		
Factor	**Skewness**	**Median**
Age	Not Skewed	38
duration	Positively skewed	180
emp.var.rate	Negatively skewed	1.1
cons.conf.idx	Positively skewed	-41.8
euribor3m	Negatively skewed	4.9
nr.employed	Negatively skewed	5191
Campaign	Positively Skewed	2

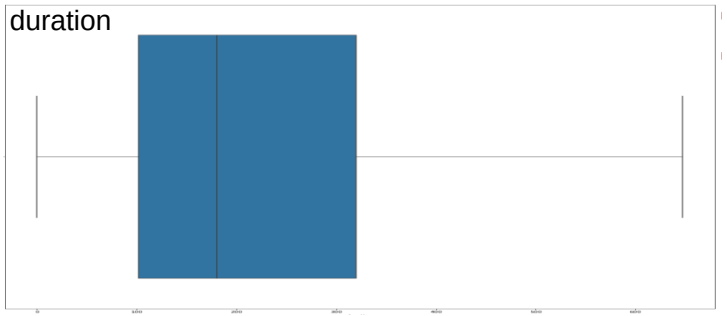
OBSERVATION



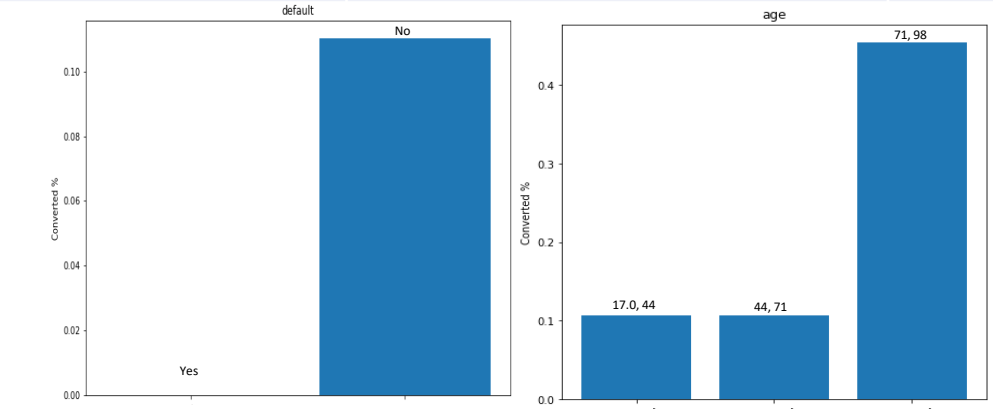


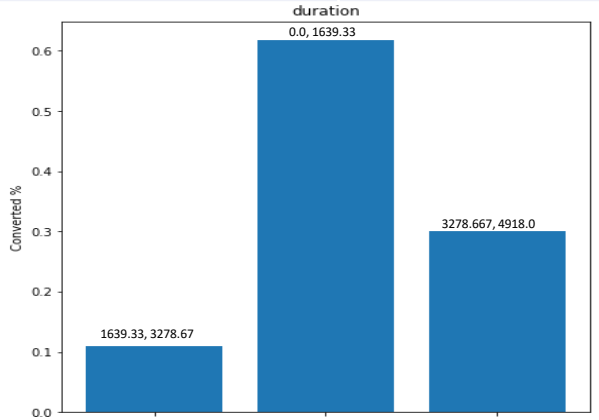


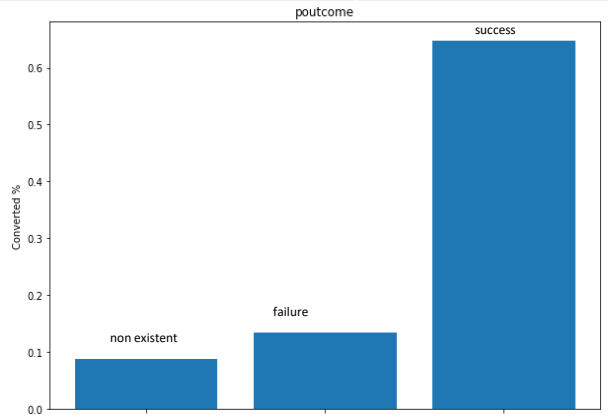


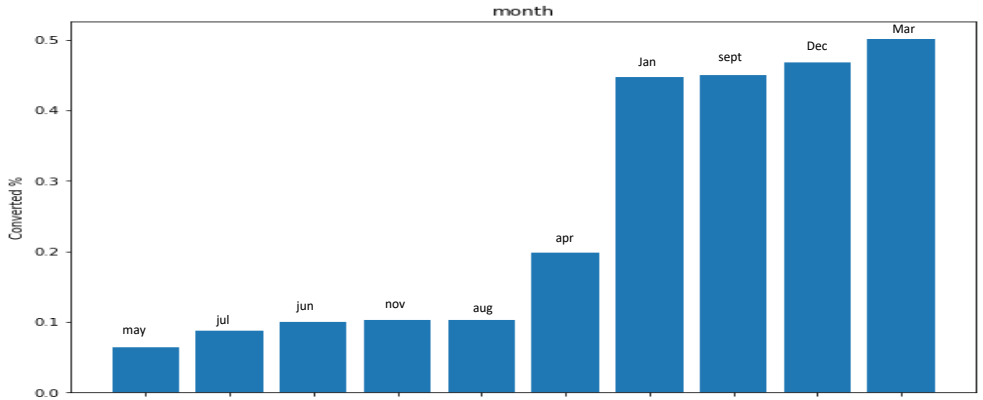


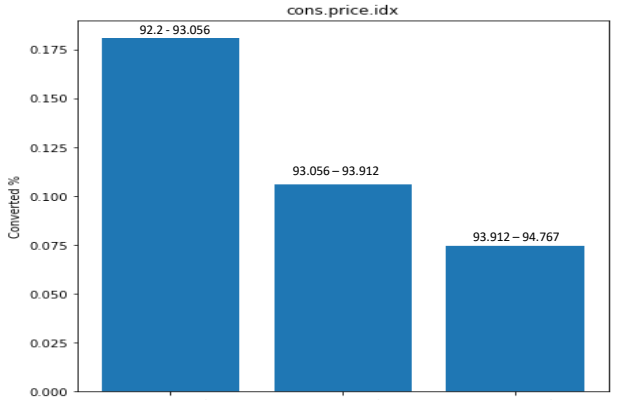
METHODOLOGY

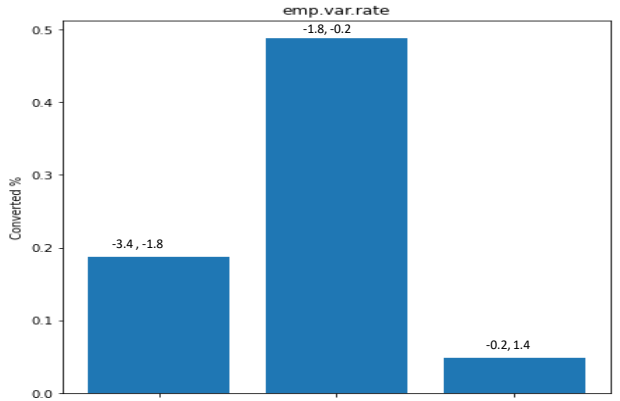


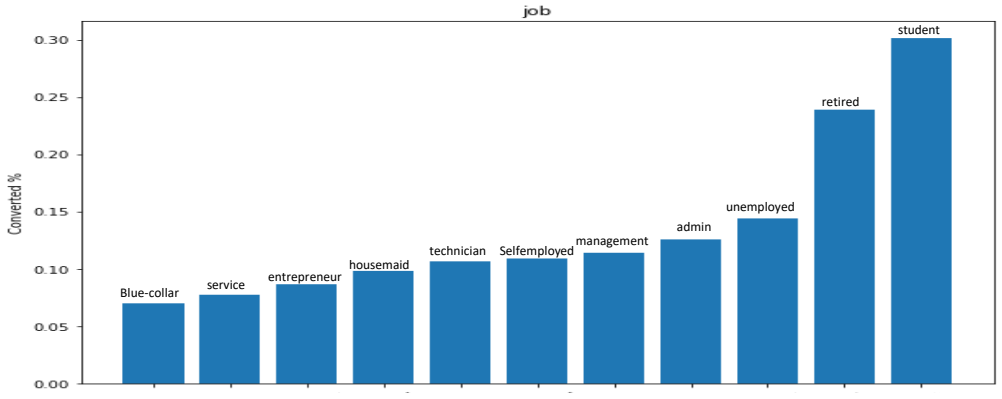


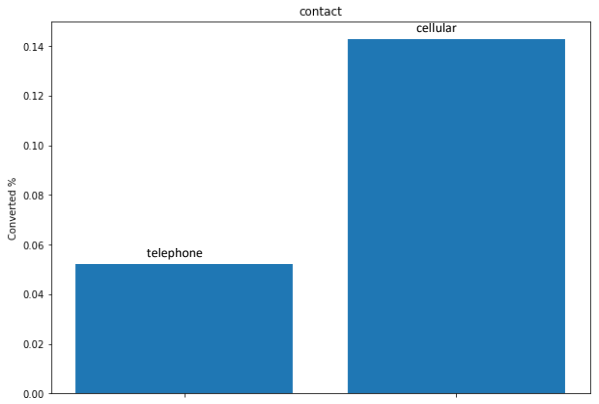


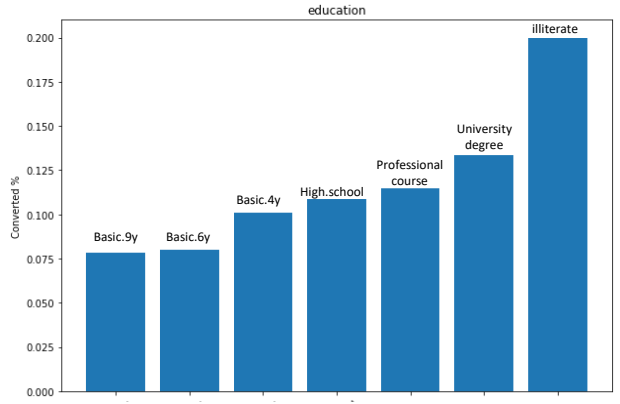












METHODOLOGY

CORRELATION

- To omit multicollinearity and reduce VIF, we remove variables that are highly correlated with each other.

VARIABLES TO REMOVE

- Removing the following variables as they are highly correlated 'euribor3m', 'nr.employed'

	age	duration	campaign	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed
age	1.000000	0.003177	0.006918	-0.002165	-0.000580	0.126671	0.009308	-0.018985
duration	0.003177	1.000000	-0.072128	-0.024083	0.006165	-0.004551	-0.026915	-0.038705
campaign	0.006918	-0.072128	1.000000	0.150361	0.129535	-0.011349	0.134271	0.142576
emp.var.rate	-0.002165	-0.024083	0.150361	1.000000	0.776617	0.210748	0.972409	0.907924
cons.price.idx	-0.000580	0.006165	0.129535	0.776617	1.000000	0.070214	0.690966	0.525887
cons.conf.idx	0.126671	-0.004551	-0.011349	0.210748	0.070214	1.000000	0.292026	0.114619
euribor3m	0.009308	-0.026915	0.134271	0.972409	0.690966	0.292026	1.000000	0.945276
nr.employed	-0.018985	-0.038705	0.142576	0.907924	0.525887	0.114619	0.945276	1.000000

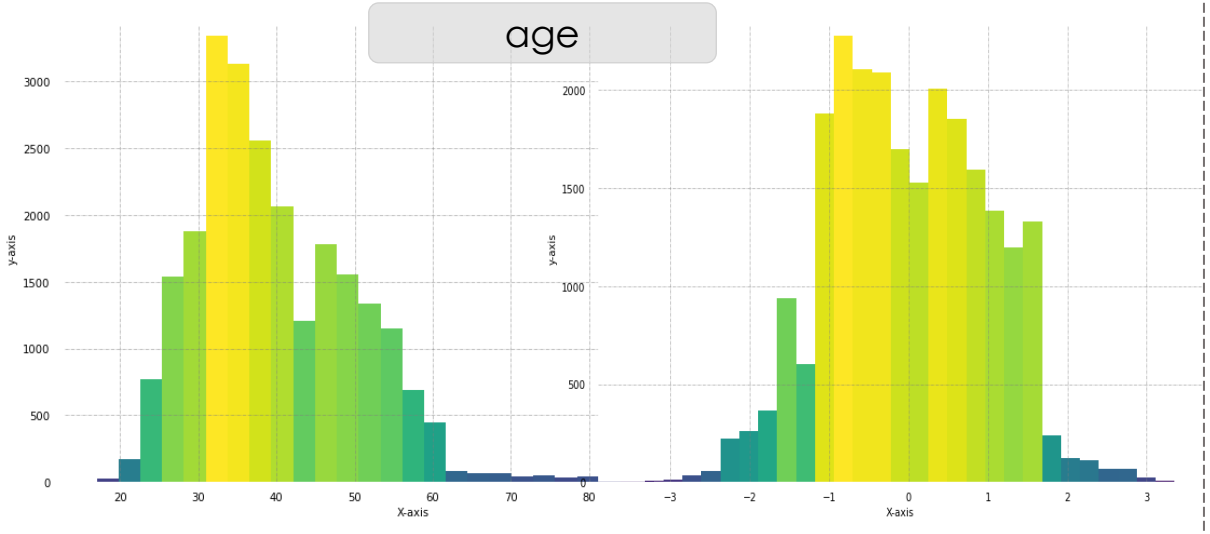
LIST OF VARIABLES

```
'age', 'job', 'marital',
'education', 'default', 'housing',
'loan', 'contact', 'month',
'day_of_week', 'duration',
'campaign', 'pdays', 'previous',
'poutcome', 'emp.var.rate',
'cons.price.idx', 'cons.conf.idx'
```

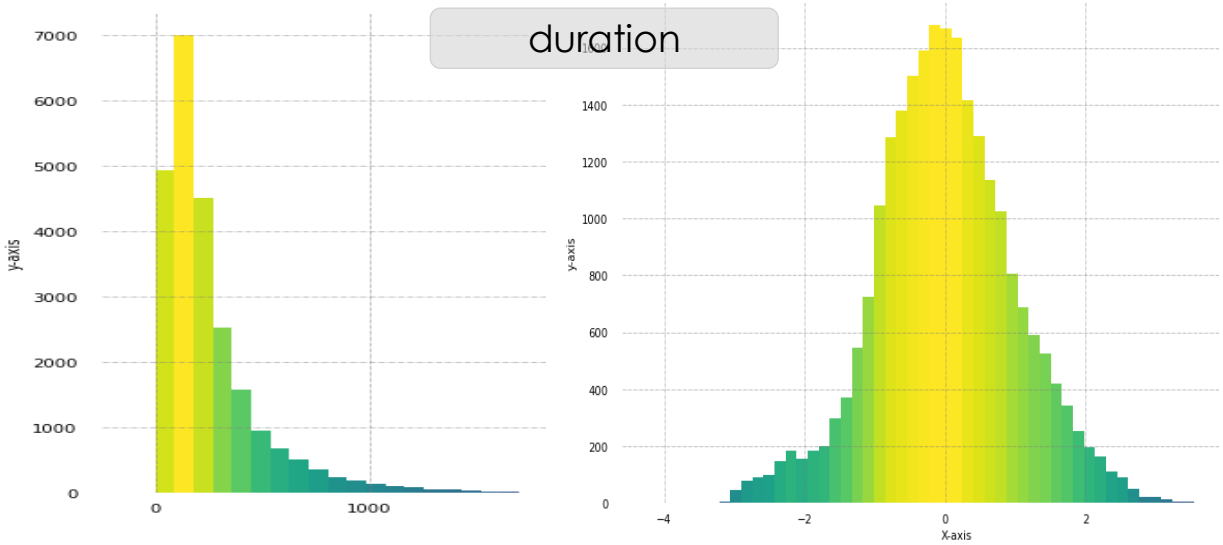
CORRELATION HEATMAP

CONVERSION TO GAUSSIAN CURVE

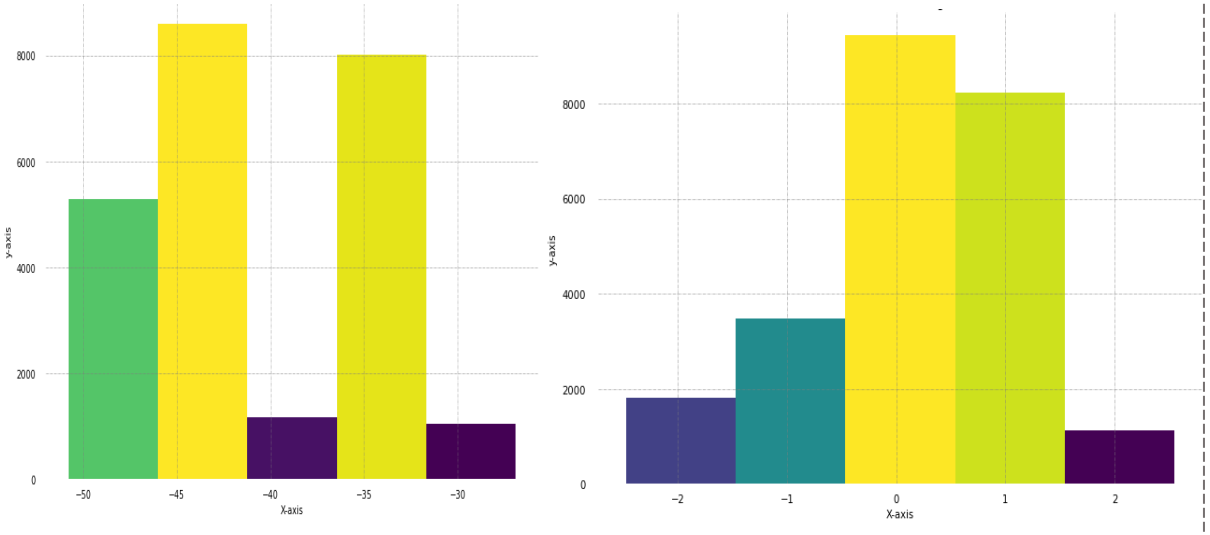
age



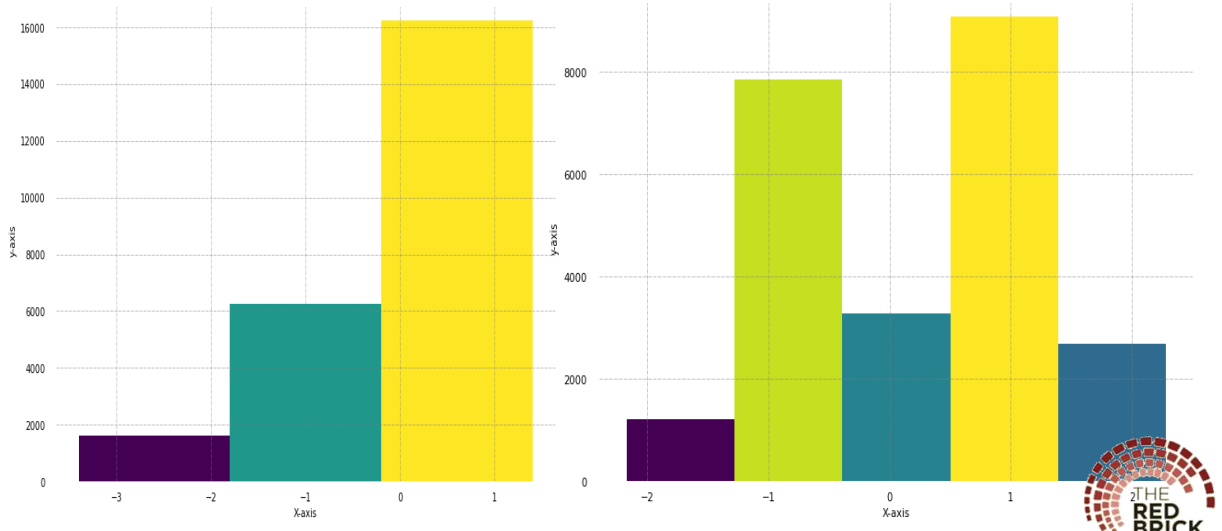
duration



Cons.conf.idx



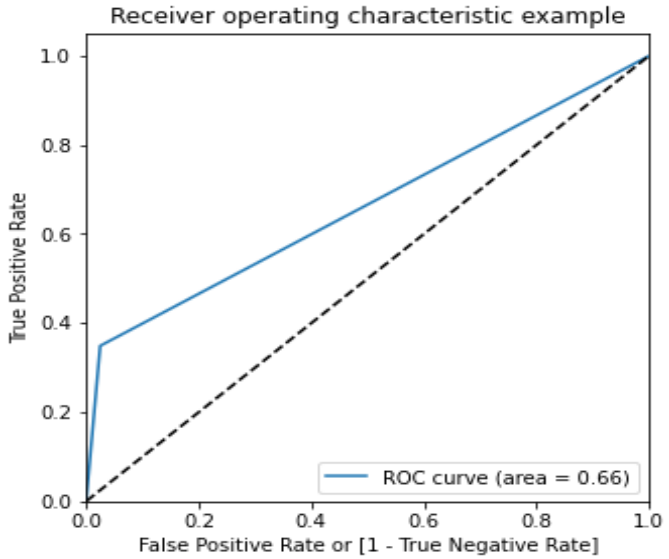
cons.price.idx

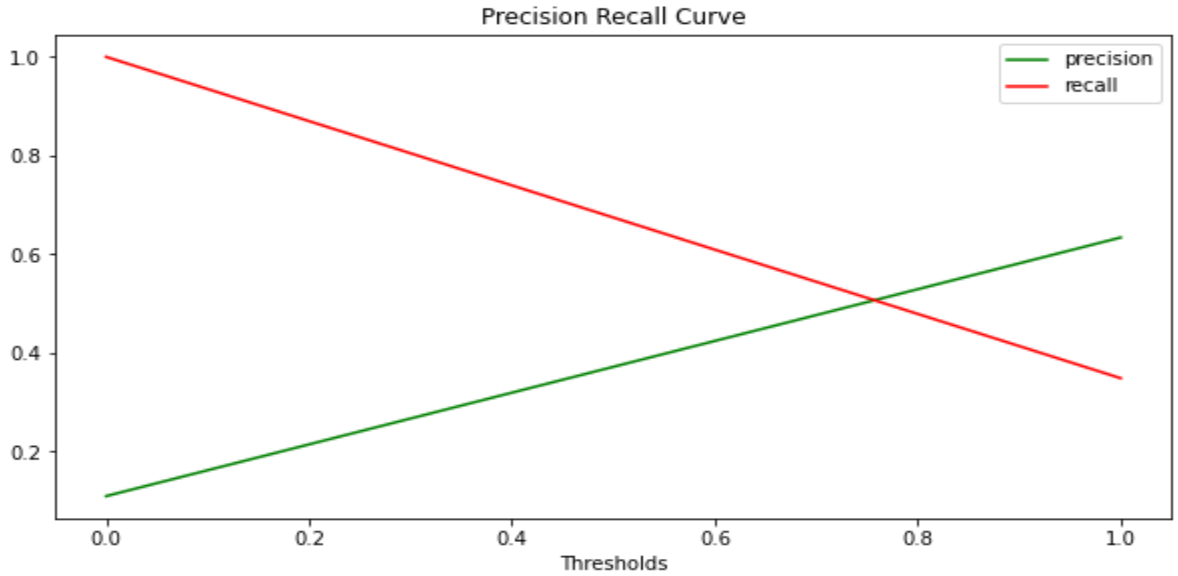


PERFORMANCE METRICS

MODEL DETAILS

- As the Target feature 'Y' is a binary variable, we use **'Logistic Regression'** Model
- **Accuracy = 0.91**
- Sensitivity = 0.35
- **Specificity = 0.98**
- False Positive Rate = 0.02
- Precision = 0.63
- Recall = 0.35



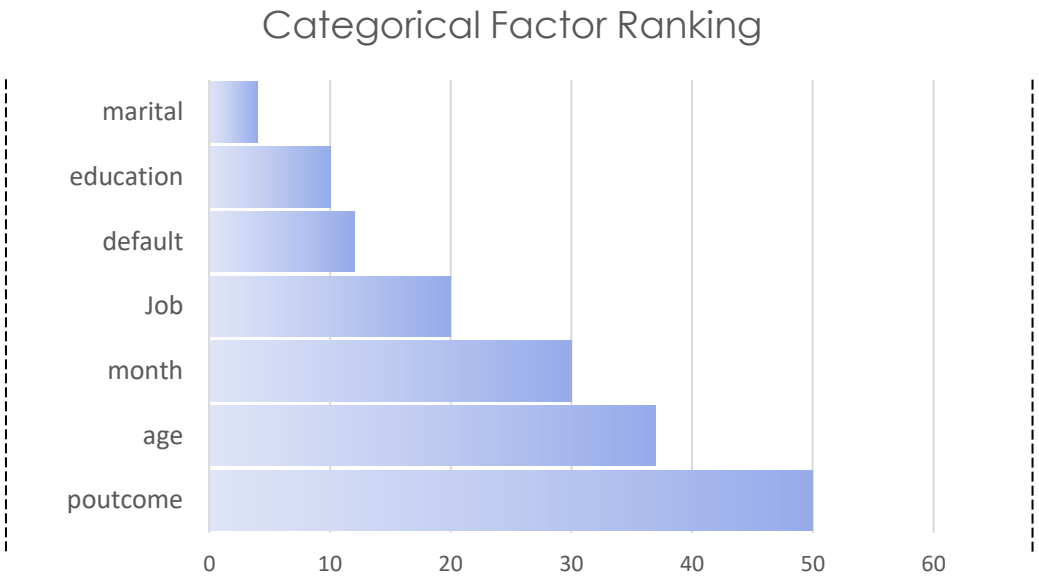


Dep. Variable:	y	**No. Observations:**	24097
Model:	GLM	**Df Residuals:**	24092
Model Family:	Binomial	**Df Model:**	4
Link Function:	logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-5226.4
Date:	Tue, 27 Sep 2022	**Deviance:**	10453.
Time:	23:01:01	**Pearson chi2:**	1.67e+04
No. Iterations:	7		
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	-3.4714	0.046	-75.406	0.000	-3.562	-3.381
cons.price.idx	0.6167	0.033	18.504	0.000	0.551	0.682
emp.var.rate	-1.7070	0.041	-41.739	0.000	-1.787	-1.627
duration	1.7341	0.034	50.478	0.000	1.667	1.801
cons.conf.idx	0.4408	0.022	20.028	0.000	0.398	0.484

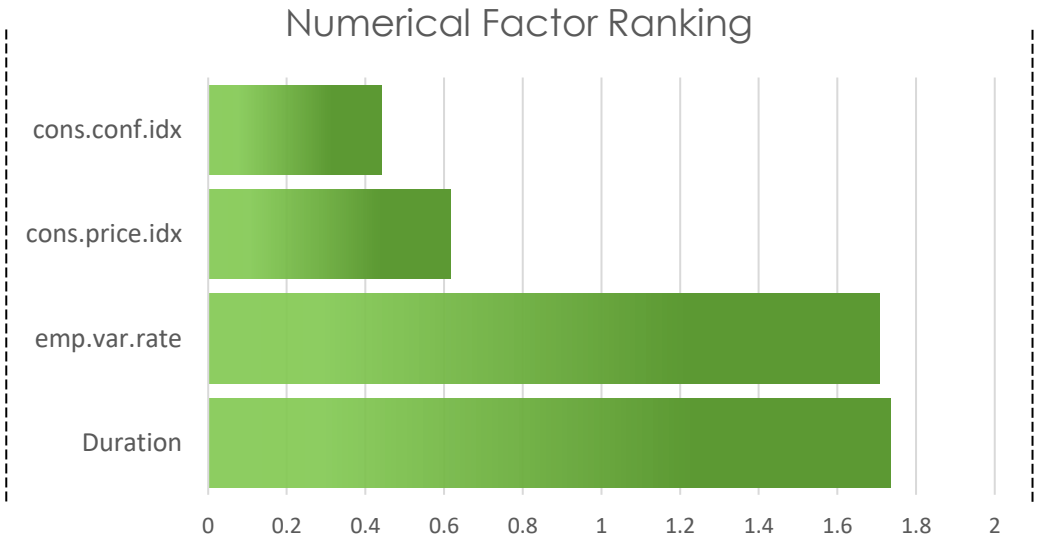
LIST OF VARIABLES

Numerical Factor Weights	
emp.var.rate	1.7341
cons.conf.idx	1.707
cons.price.idx	0.6167
duration	0.4408



- No person who has defaulted earlier has taken term Deposit.
- People who have taken housing loan or not does not effect their chances of taking Term Deposit.
- emp.var.rate shows unemployment rate and is inversely correlated to subscribing to Term deposit.
- Duration has a sweet spot. More or less call duration decreases the subscription chances.

Advantage(%) Categorical Factors	
poutcome	50
age	37
month	30
Job	20
default	12
education	10
marital	4



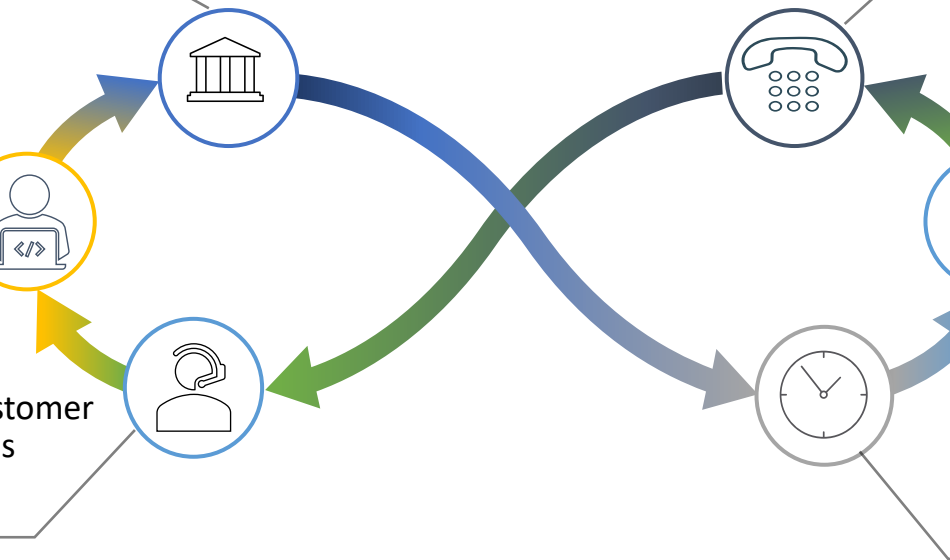
- Job-'Student' has a conversion rate is ~30%, 'retired' has 24% conversion rate.
- Education-'Illiterate' has the highest conversion rate of 20%. This maybe due to lack of data or government schemes.
- Poutcome-'Success' shows a significant conversion rate of 65%.
- The Later months in the year shows a significant increase in conversion rate – Due to taxes & fiscal year cycle.

- Student Awareness Programs (CFA institutes etc)
- Student oriented programs for Short-term FD
- Influencer Marketing

- Regular feedback post campaigns
- Bundled products: Loan + term Deposit

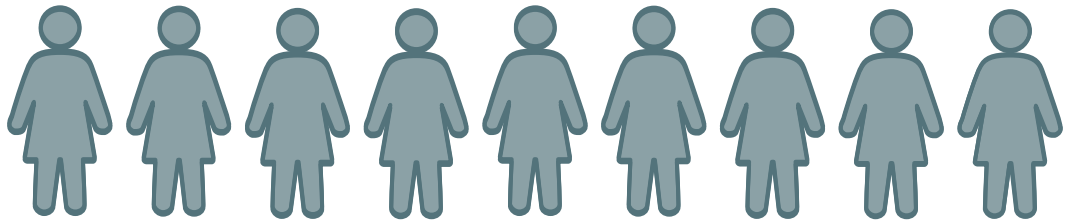
- Develop dedicated profile of hot leads based on consumer confidence using ML.

- Partner With ClearTax, ET Money.
- Heavy Marketing (Newspaper, digital)



- Regular training of customer service representatives
- Regular feedback

- Omni Channel Approach
- Provide door-step service
- Fully digitized term deposit process
- Customer Representatives for old age homes



भारतीय प्रबंध संस्थान कोषिक्कोड
Indian Institute of Management Kozhikode
Globalizing Indian Thought

THE RED BRICK SUMMIT

Thank You!

Submission for
IIM Ahmedabad's Blitzkrieg challenge (TRBS)

Team BLabber

Apoorv Gupta

APPENDIX: Python Code

jupyter Blitzkreig Last Checkpoint: Yesterday at 1:22 AM (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

```
    plt.xlabel('Thresholds')
    plt.title('Precision Recall Curve')
    plt.legend()
    plt.show()
```

In [4]:
```
#data = pandas.read_csv(r"C:\Users\Gupta\Downloads\bank data final question.xlsx",encoding = "1250")
data=pd.read_excel(r"C:\Users\Gupta\Downloads\bank data final question.xlsx", sheet_name="Training Data")
```

In [5]:
```
data.head()
```

Out[5]:

	age	job	marital	education	default	housing	loan	contact	month	day_of_week	duration	campaign	pdays	previous	poutcome	emp.v
0	26	admin.	single	university.degree	no	no	no	telephone	aug	mon	1087	1	3	1	success	
1	49	unknown	married	high.school	unknown	no	yes	cellular	aug	tue	144	2	999	0	nonexistent	
2	40	entrepreneur	married	basic.9y	no	no	no	telephone	nov	tue	40	1	999	0	nonexistent	
3	39	self-employed	married	basic.4y	unknown	no	no	telephone	may	mon	179	4	999	0	nonexistent	
4	38	housemaid	married	high.school	no	yes	no	cellular	jul	wed	335	5	999	0	nonexistent	

In [6]:
```
print(data.columns)
print(data.shape)
```

```
Index(['age', 'job', 'marital', 'education', 'default', 'housing', 'loan',
       'contact', 'month', 'day_of_week', 'duration', 'campaign', 'pdays',
       'previous', 'poutcome', 'emp.var.rate', 'cons.price.idx',
       'cons.conf.idx', 'euribor3m', 'nr.employed', 'y', 'Unnamed: 21'],
      dtype='object')
(37069, 22)
```

In [7]:
```
data = data.replace({'unknown':numpy.nan});data = data.replace({'Unknown':numpy.nan});
```

In [8]:
```
data.head()
```

jupyter Blitzkreig Last Checkpoint: Yesterday at 1:22 AM (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

```
       'previous', 'poutcome', 'emp.var.rate', 'cons.price.idx',
       'cons.conf.idx', 'Unnamed: 21'],
      dtype='object')
```

In [44]:
```
print(numpy.sum(y))
print(numpy.sum(y_train))
print(numpy.sum(y_test))

print('Train subscriber Rate', numpy.sum(y_train)/len(y_train))
print('Test subscriber Rate', numpy.sum(y_test)/len(y_test))
```

```
3795
2651
1144
Train subscriber Rate 0.1100136946507864
Test subscriber Rate 0.11076684740511232
```

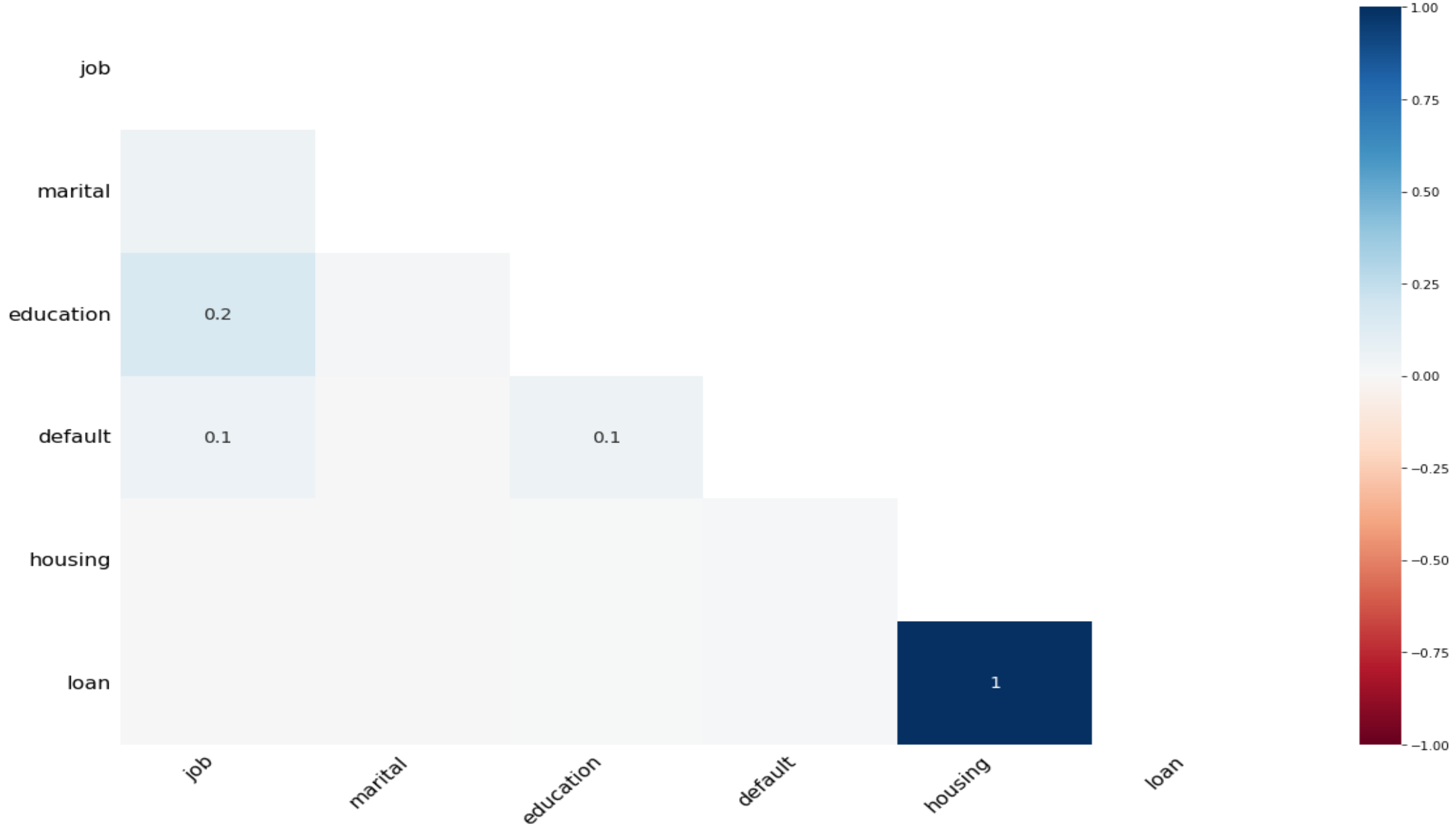
Observation:

The train and test sets both have similar event rates.

Plot the histogram of a variable from the dataset to see the skewness

In [45]:
```
n_bins = 29
fig, axs = plt.subplots(1, 1, figsize =(10, 7),tight_layout = True)
for s in ['top', 'bottom', 'left', 'right']:
    axs.spines[s].set_visible(False)
axs.xaxis.set_ticks_position('none')
axs.yaxis.set_ticks_position('none')
axs.xaxis.set_tick_params(pad = 5)
axs.yaxis.set_tick_params(pad = 10)
axs.grid(b = True, color ='grey',
        linestyle ='-.', linewidth = 0.5,
        alpha = 0.9)
N, bins, patches = axs.hist(X_train['age'], bins = n_bins)
```

APPENDIX: Correlation with Nullity



APPENDIX: Standard Scaler

jupyter Blitzkreig Last Checkpoint: Yesterday at 1:22 AM (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

In [70]:
```
data_test.shape
```

Out[70]: (4119, 21)

In [71]:
```
X_test=data_test[features]
```

In [72]:
```
X_test
```

Out[72]:

	cons.price.idx	emp.var.rate	duration	cons.conf.idx
0	93.444	1.4	122	-36.1
1	93.444	1.4	331	-36.1
2	93.994	1.1	339	-36.4
3	93.200	-0.1	310	-42.0
4	93.444	1.4	207	-36.1
...	...	...	...	...
4114	93.918	1.4	147	-42.7
4115	93.444	1.4	507	-36.1
4116	92.893	-1.8	402	-46.2
4117	93.075	-1.8	397	-47.1
4118	93.994	1.1	197	-36.4

4119 rows × 4 columns

In [73]:
```
# - Apply : preprocessing.PowerTransformer(copy=False) to fit & transform the train & test data
pt = preprocessing.PowerTransformer()
X_test[features] = pt.fit_transform(X_test[features])
```

In [74]:
```
scaler = preprocessing.StandardScaler()
X_test[features] = scaler.fit_transform(X_test[features])
X_test[features] = scaler.transform(X_test[features])
```

In [75]:
```
ans=list(logreg.predict(X_test))
```