Data Science

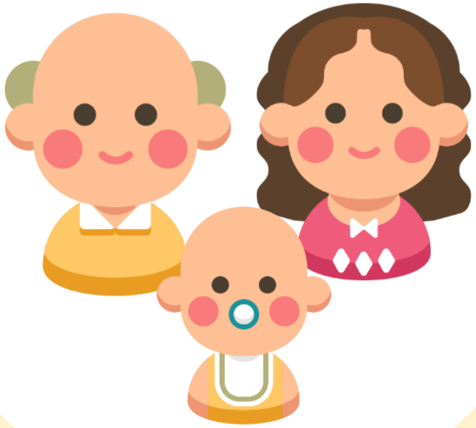
Term project Final Presentation

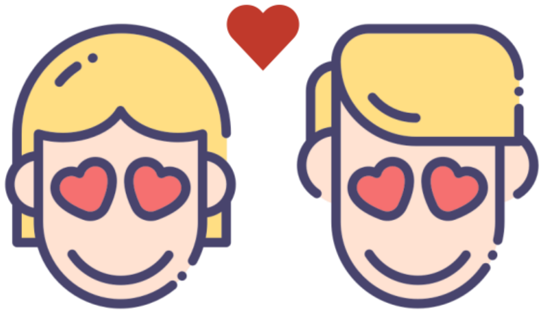
최준헌 김지현 양희림

Contents

Objective Setting

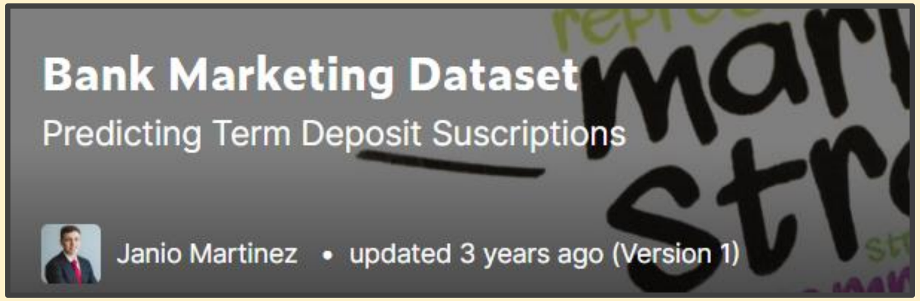
Is the client subscribed a term deposit?





Data Curation

kaggle



bank.csv (897.42 KB)

Detail Compact Column 10 of 17 columns

age	job	marital	education	default
18 – 95	management 23% blue-collar 17% Other (6652) 60%	married 57% single 32% Other (1293) 12%	secondary 49% tertiary 33% Other (1997) 18%	true 0 0% false 0 0%
31	technician	single	tertiary	no
35	management	divorced	tertiary	no
32	blue-collar	single	primary	no
49	services	married	secondary	no
41	admin.	married	secondary	no
49	admin.	divorced	secondary	no
28	admin.	divorced	secondary	no
43	management	single	tertiary	no
43	management	divorced	tertiary	no
43	blue-collar	married	primary	no

Data Inspection

Data Inspection

Head data

```
    age            job  marital  education  ... pdays  previous poutcome deposit
0  38.0     management  married   tertiary  ...    -1         0  unknown      no
1  50.0    blue-collar   single    primary  ...    -1         0  unknown      no
2  40.0  self-employed   single   tertiary  ...    -1         0  unknown     yes
3  38.0     technician  married  secondary  ...    -1         0  unknown     yes
4  55.0    blue-collar  married  secondary  ...    -1         0  unknown     yes

[5 rows x 17 columns]
```

Numerical column Info

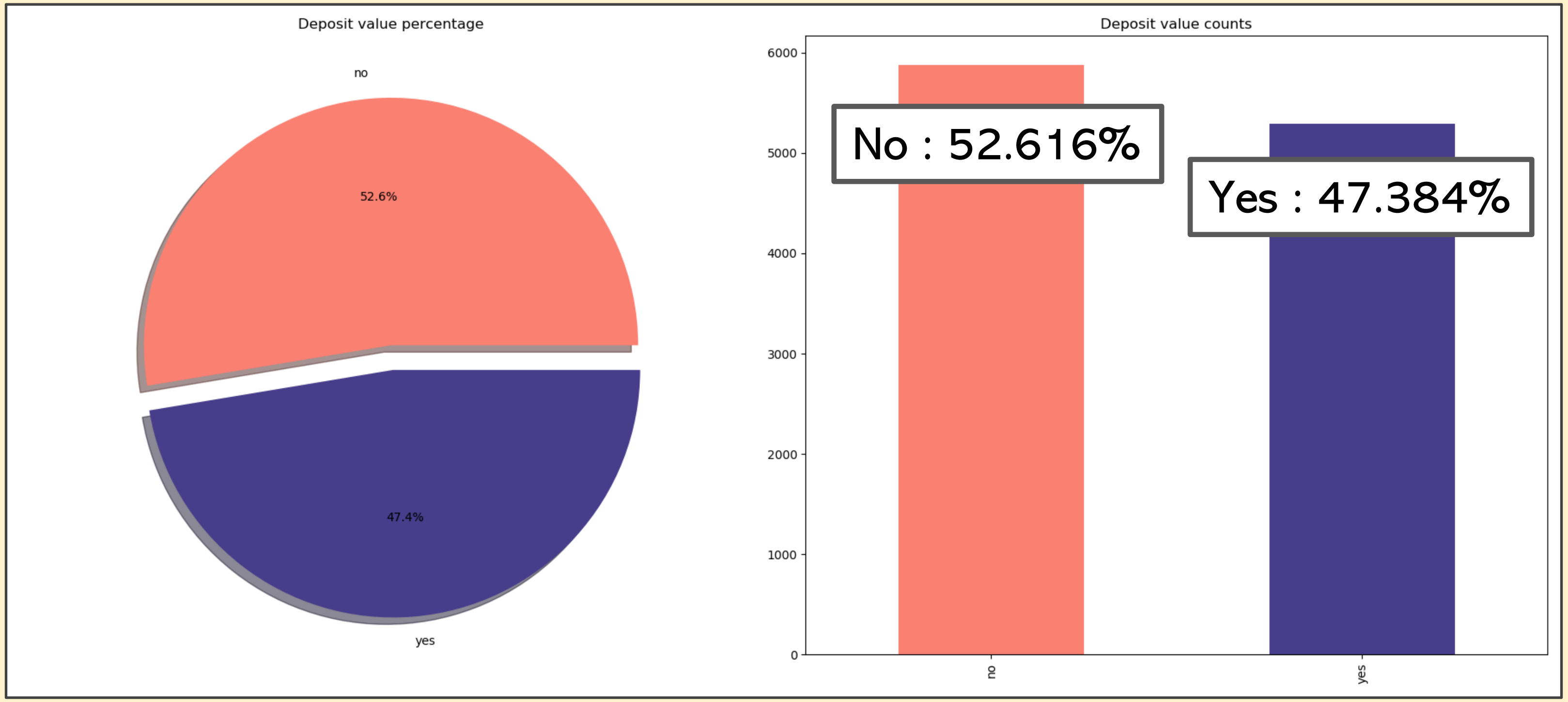
```
                age       balance  ...         pdays      previous
count  11109.000000  11043.000000  ...  11162.000000  11162.000000
mean      44.260059   1530.081409  ...     51.330407      0.832557
std       76.330654   3233.456493  ...    108.758282      2.292007
min    -1000.000000  -6847.000000  ...     -1.000000      0.000000
25%       32.000000    122.000000  ...     -1.000000      0.000000
50%       39.000000    551.000000  ...     -1.000000      0.000000
75%       49.000000   1711.000000  ...     20.750000      1.000000
max     2000.000000  81204.000000  ...    854.000000     58.000000

[8 rows x 7 columns]
```

Column & Data type

```
feature names & data types
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11162 entries, 0 to 11161
Data columns (total 17 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   age        11109 non-null  float64
 1   job        11162 non-null  object
 2   marital    11097 non-null  object
 3   education  11101 non-null  object
 4   default    11162 non-null  object
 5   balance    11043 non-null  float64
 6   housing    11162 non-null  object
 7   loan       11162 non-null  object
 8   contact    11101 non-null  object
 9   day        11162 non-null  int64
 10  month      11162 non-null  object
 11  duration   11110 non-null  float64
 12  campaign   11162 non-null  int64
 13  pdays      11162 non-null  int64
 14  previous   11162 non-null  int64
 15  poutcome   11162 non-null  object
 16  deposit    11162 non-null  object
dtypes: float64(3), int64(4), object(10)
memory usage: 1.4+ MB
```

Data Inspection

Percentage of target value




```
target value count
no      5873
yes     5289
Name: deposit, dtype: int64


target value percentage
Percentage of "No": 52.616%
Percentage of "Yes": 47.384%
```

Data Inspection

Missing data

age	job	marital	educat	default	balance	housin	loan	contac	day	month	duratic	campai
56	services	married	primary	no	486	no	yes	cEllULAr	21	jul	1877	1
41	blue-colla	divorced	primary	no	285	yes	no	cellular	20	apr	1272	2
45	admin.	married	secondary	no	236	no	no	cellular	20	aug	703	2
48	management		AAA	no		no	yes		8	jul	-10	1
59	managem	married	unknown	no	3534	no	no	cellular	21	nov	216	4
39	managem	married	tertiary	no	22	yes	no	unknown	2	jun	493	1
44	managem	married	tertiary	no	70	no	no	cellular	20	aug	165	3
48	unemploy	divorced	secondary	no	201	no	no	cellular	11	aug	140	1
48	self-empl	married	secondary	no	1559	no	no	cellular	4	feb	130	2
2000	blue-colla	single	secondary	no	953	yes	no	cellular	14	may	479	1
42	technicia	single	secondary	no	49	no	no	cellular	9	feb	7	10
56	blue-colla	married	secondary	no	1210	no	no	unknown	11	jun	935	1
95	retired	divorced	primary	no	2282	no	no	telephone	21	apr	207	17
34	blue-colla	married	secondary	no	577	no	no	unknown	14	may	337	1
60	entrepren	divorced	secondary	no	80	yes	no	unknown	15	may	397	1
59	housemaid		secondary	no	1040	no	no	cellular	5	aug	123	2
39	unemploy	single	tertiary	no	7	yes	no	cellular	20	nov	931	4
48	managem	single	tertiary	no	86	no	no	cellular	28	jun	281	3

Data Inspection

Wrong data

age	job	marital	educat	default	balanc	housin	loan	contac	day	month	duratic	campai
56	services	married	primary	no	486	no	yes	cEllULAr	21	jul	1877	1
41	blue-colla	divorced	primary	no	285	yes	no	cellular	20	apr	1272	2
45	admin.	married	secondary	no	236	no	no	cellular	20	aug	703	2
48	management		AAA	no		no	yes		8	jul	-10	1
59	managem	married	unknown	no	3534	no	no	cellular	21	nov	216	4
39	managem	married	tertiary	no	22	yes	no	unknown	2	jun	493	1
44	managem	married	tertiary	no	70	no	no	cellular	20	aug	165	3
48	unemploy	divorced	secondary	no	201	no	no	cellular	11	aug	140	1
48	self-empl	married	secondary	no	1559	no	no	cellular	4	feb	130	2
2000	blue-colla	single	secondary	no	953	yes	no	cellular	14	may	479	1
42	technician	single	secondary	no	49	no	no	cellular	9	feb	7	10
56	blue-colla	married	secondary	no	1210	no	no	unknown	11	jun	935	1
95	retired	divorced	primary	no	2282	no	no	telephone	21	apr	207	17
34	blue-colla	married	secondary	no	577	no	no	unknown	14	may	337	1
60	entrepren	divorced	secondary	no	80	yes	no	unknown	15	may	397	1
59	housemaid		secondary	no	1040	no	no	cellular	5	aug	123	2
39	unemploy	single	tertiary	no	7	yes	no	cellular	20	nov	931	4
48	managem	single	tertiary	no	86	no	no	cellular	28	jun	281	3

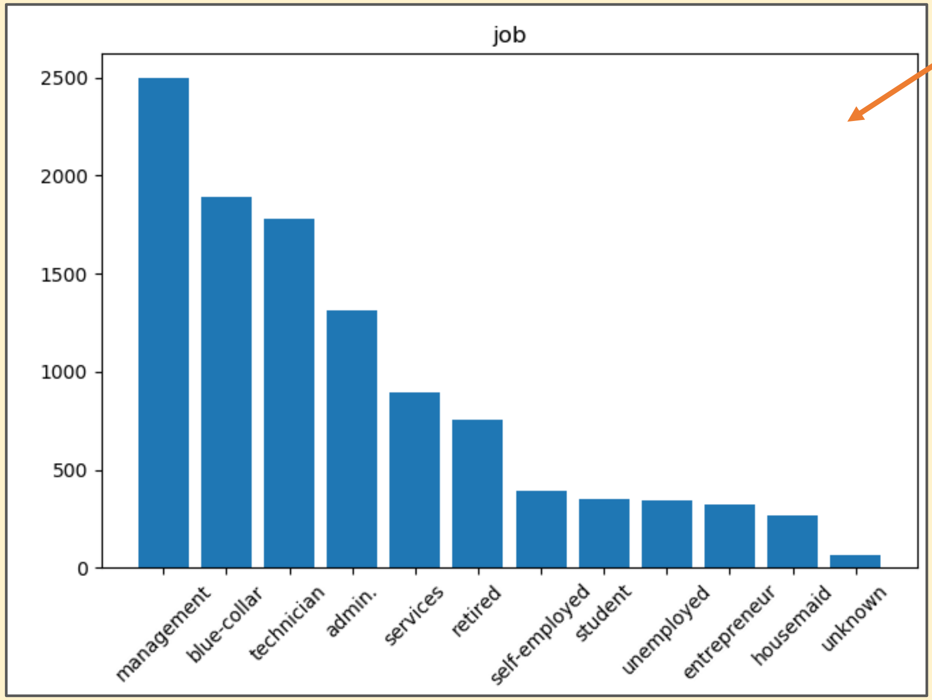
Data Inspection

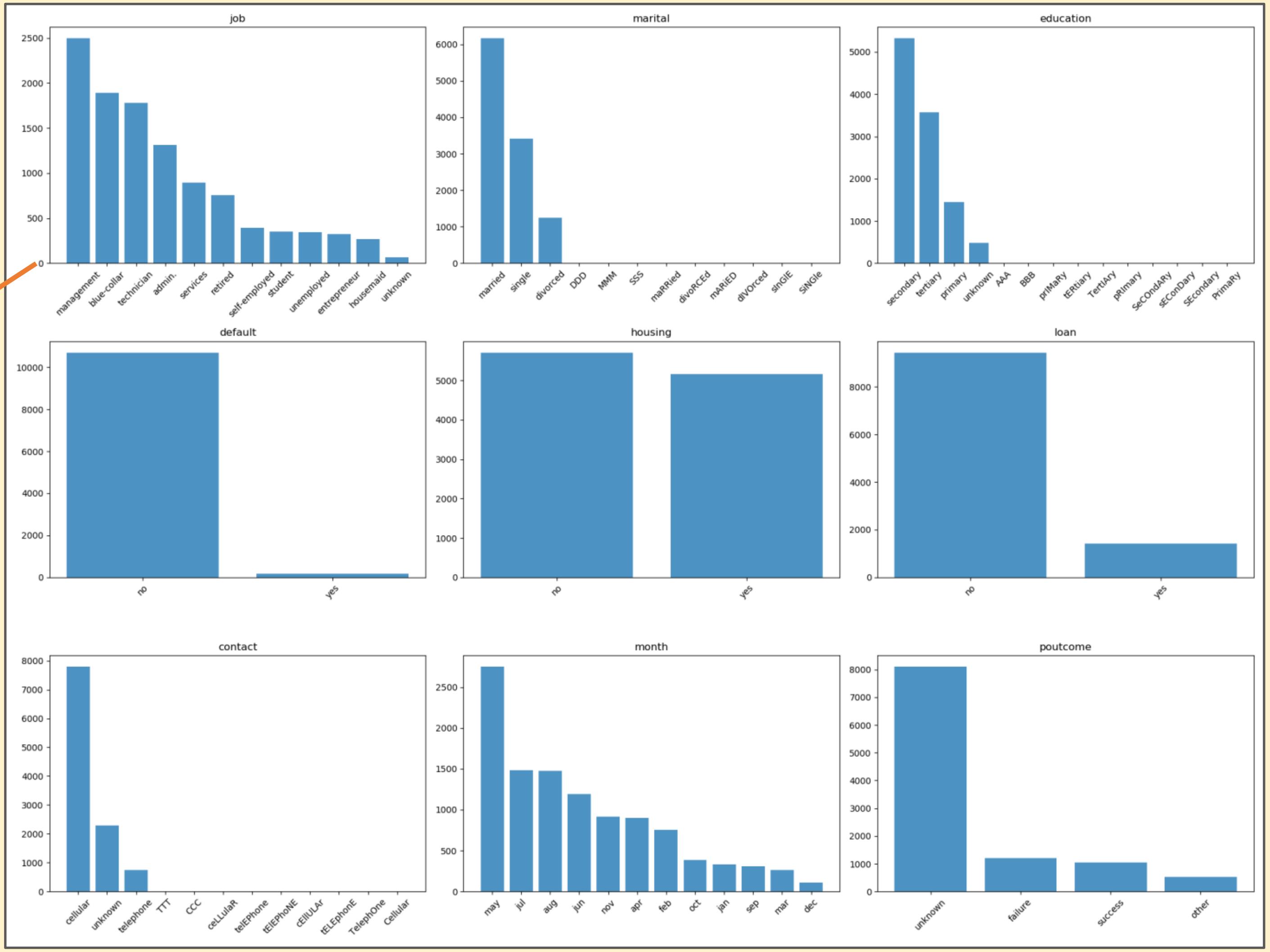
Wrong data, But Usable Data

age	job	marital	educat	default	balanc	housin	loan	contac	day	month	duratic	campai
56	services	married	primary	no	486	no	yes	cEllULAr	21	jul	1877	1
41	blue-colla	divorced	primary	no	285	yes	no	cellular	20	apr	1272	2
45	admin.	married	secondary	no	236	no	no	cellular	20	aug	703	2
48	management		AAA	no		no	yes		8	jul	-10	1
59	managem	married	unknown	no	3534	no	no	cellular	21	nov	216	4
39	managem	married	tertiary	no	22	yes	no	unknown	2	jun	493	1
44	managem	married	tertiary	no	70	no	no	cellular	20	aug	165	3
48	unemploy	divorced	secondary	no	201	no	no	cellular	11	aug	140	1
48	self-empl	married	secondary	no	1559	no	no	cellular	4	feb	130	2
2000	blue-colla	single	secondary	no	953	yes	no	cellular	14	may	479	1
42	techniciar	single	secondary	no	49	no	no	cellular	9	feb	7	10
56	blue-colla	married	secondary	no	1210	no	no	unknown	11	jun	935	1
95	retired	divorced	primary	no	2282	no	no	telephone	21	apr	207	17
34	blue-colla	married	secondary	no	577	no	no	unknown	14	may	337	1
60	entrepren	divorced	secondary	no	80	yes	no	unknown	15	may	397	1
59	housemaid		secondary	no	1040	no	no	cellular	5	aug	123	2
39	unemploy	single	tertiary	no	7	yes	no	cellular	20	nov	931	4
48	managem	single	tertiary	no	86	no	no	cellular	28	jun	281	3

Data Inspection

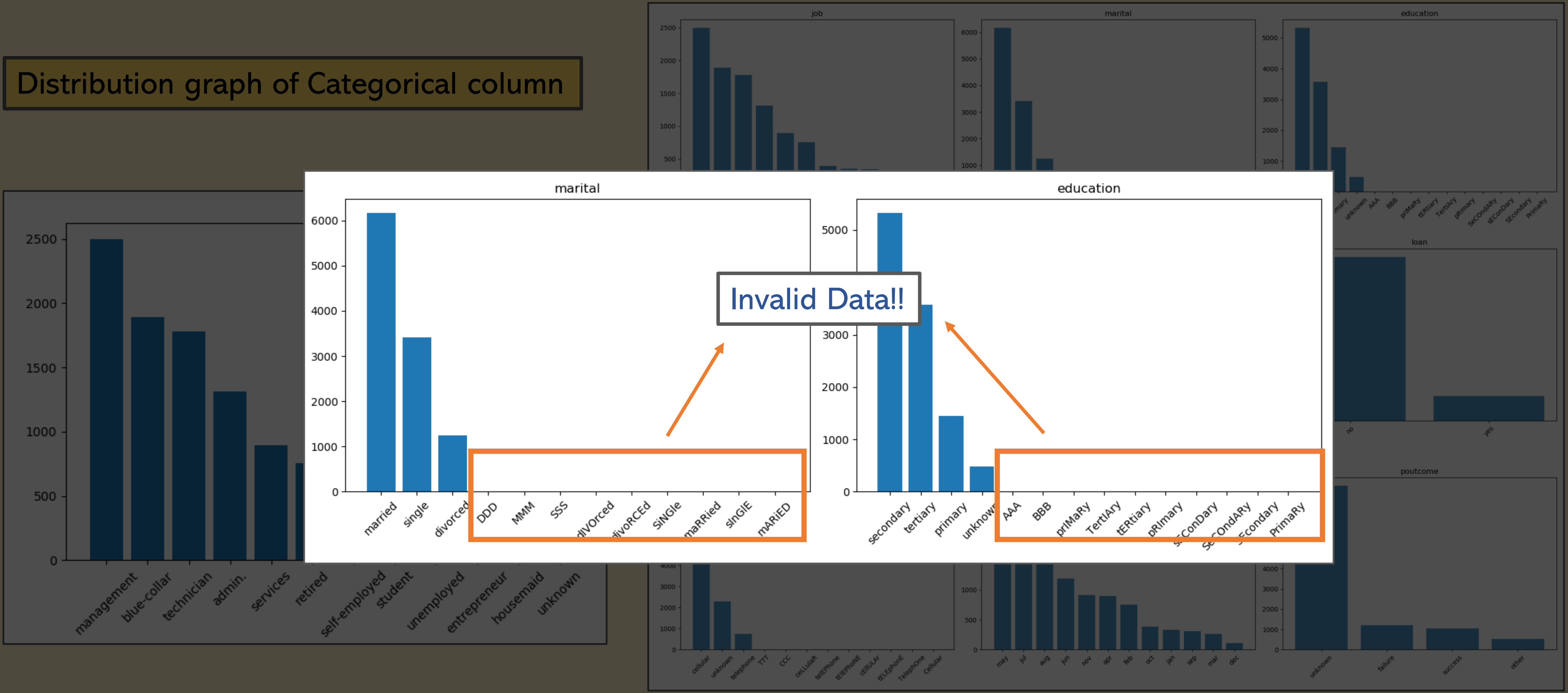
Distribution graph of Categorical column





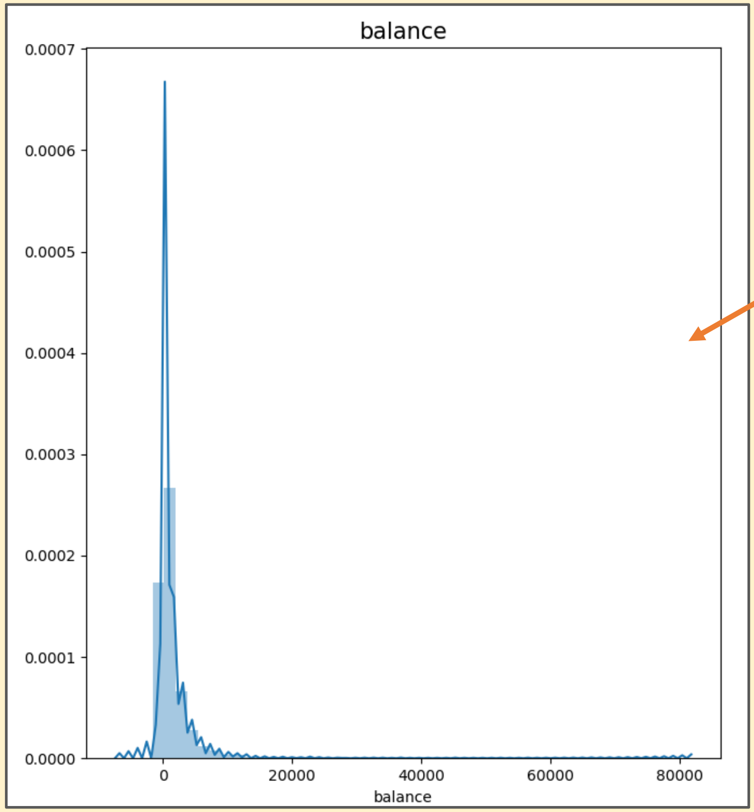
Data Inspection

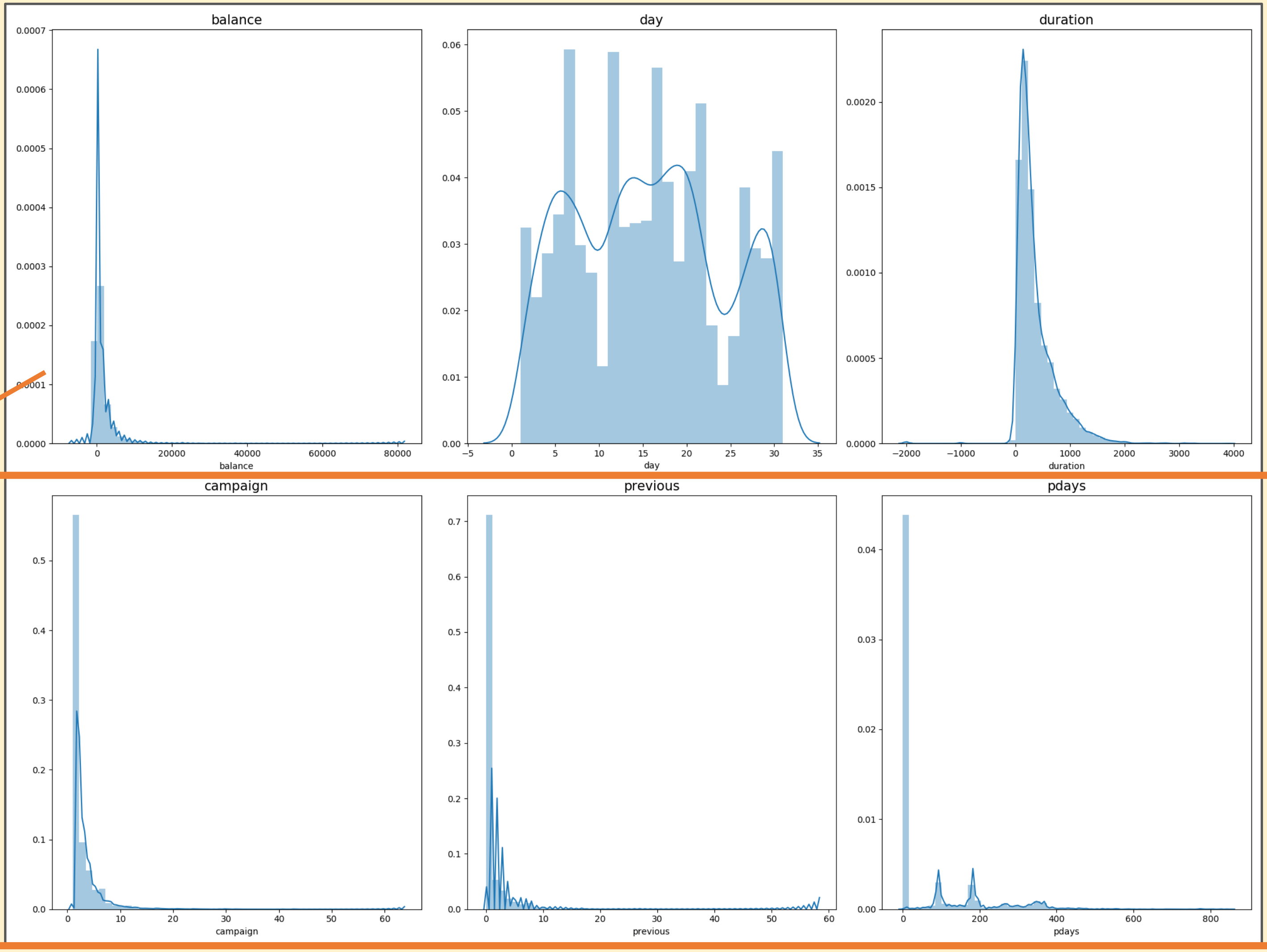
Distribution graph of Categorical column



Data Inspection

Distribution graph of Numeric column





Data Preprocessing

Data Preprocessing

What we preprocess?

- Missing data
- Wrong data
- Wrong data, But Usable Data
- Unusable Data
- Outlier Data

How preprocessing?

Drop | Fill | Replace

1. By target feature value,
   - Numeric → Median
   - Categorical → Mode
2. By related feature value
   - Numeric → Linear Regression / Median
   - Categorical → Mode

Data Preprocessing

Change uppercase to lowercase

age	job	marital	educatio
39	blue-collar	sInGIE	tertiary
19	student	single	prIMaRy
30	manageme	dIVOrced	tertiary

age	job	marita	educati
39	blue-collar	single	tertiary
19	student	single	primary
30	manageme	divorced	tertiary

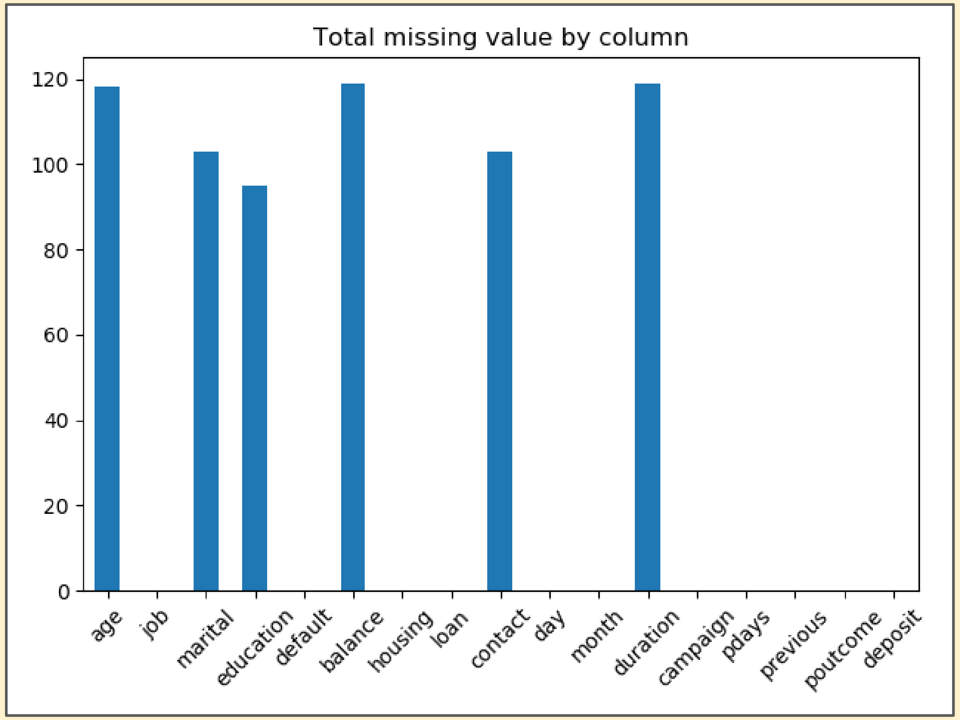
Data Preprocessing

Replace wrong data to NA

age	job	marita	educati	contac
-100	managem	mmm	aaa	ccc
64	retired	sss	secondary	cellular
48	management		aaa	
55	services		bbb	
-100	blue-collar			ccc
36	managem	maried	tertiary	cellular
43	blue-collar	sss	secondary	unknown
35	technician	ddd	secondary	unknown
36	blue-collar	married	secondary	ccc
34	managem	single	bbb	cellular
53	managem	divorced	tertiary	ttt
60	retired	married	bbb	cellular
	housemaic	mmm	primary	ttt
-1	management			ttt
-100	blue-collar			ttt
	managem	sss		cellular

age	job	marita	educati	contac
	management			
64	retired		secondary	cellular
48	management			
55	services			
	blue-collar			
36	management		tertiary	cellular
43	blue-collar		secondary	unknown
35	technician		secondary	unknown
36	blue-collar	married	secondary	
34	managem	single		cellular
53	managem	divorced	tertiary	
60	retired	married		cellular
	housemaid		primary	
	management			
	blue-collar			
	management			cellular

Data Preprocessing



Percentage of that is missing : 0.34%

Missing value existence status

age	True
job	False
marital	True
education	True
default	False
balance	True
housing	False
loan	False
contact	True
day	False
month	False
duration	True
campaign	False
pdays	False
previous	False
poutcome	False
deposit	False
dtype: bool	

How many missing value?

age	118
job	0
marital	103
education	95
default	0
balance	119
housing	0
loan	0
contact	103
day	0
month	0
duration	119
campaign	0
pdays	0
previous	0
poutcome	0
deposit	0
dtype: int64	

Percentage of missing value

age	1.057158
job	0.000000
marital	0.922774
education	0.851102
default	0.000000
balance	1.066117
housing	0.000000
loan	0.000000
contact	0.922774
day	0.000000
month	0.000000
duration	1.066117
campaign	0.000000
pdays	0.000000
previous	0.000000
poutcome	0.000000
deposit	0.000000
dtype: float64	

Data Preprocessing

Delete row with more than 5 missing values

age	job	marita	educati	defaul	balanc	housin	loan	contac	day	montl
	management			no		no	no		2	mar

1 2 3 4 5

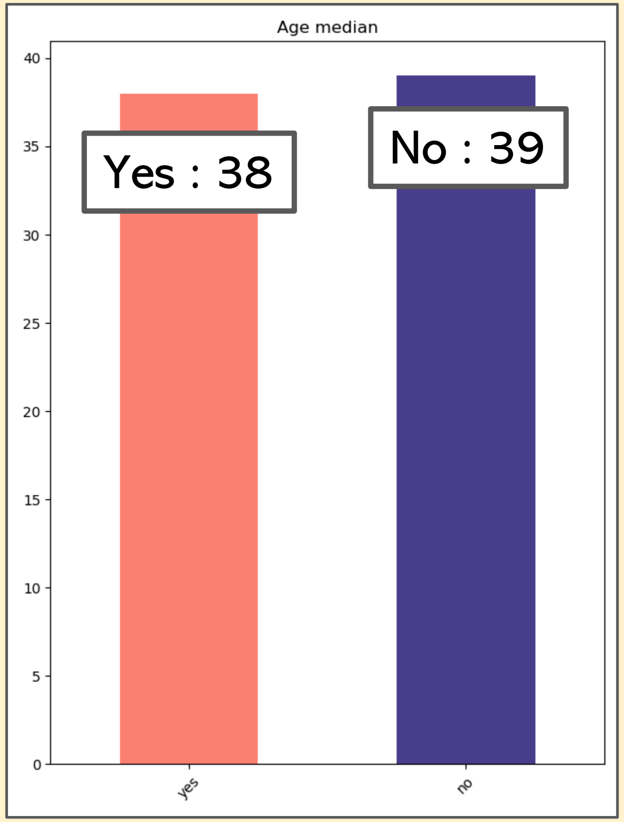
Data Preprocessing

1 By target feature value,

Numeric → Median

Categorical → Mode

Fill missing data to specific values [Numeric]



age	job	marita	educati	deposi
38	manageme			yes
38	blue-colla	single	secondary	yes
39	managem	married	primary	no
38	blue-colla			yes
38	blue-colla	single	secondary	yes
38	unemploy			yes
39	managem	married	tertiary	no
39	managem	married	tertiary	no
39	managem	married	tertiary	no
39	services			no
38	retired	divorced	secondary	yes
38	managem	married	tertiary	yes

If deposit value is "Yes"
→ Fill age to 38

If deposit value is "No"
→ Fill age to 39

Data Preprocessing

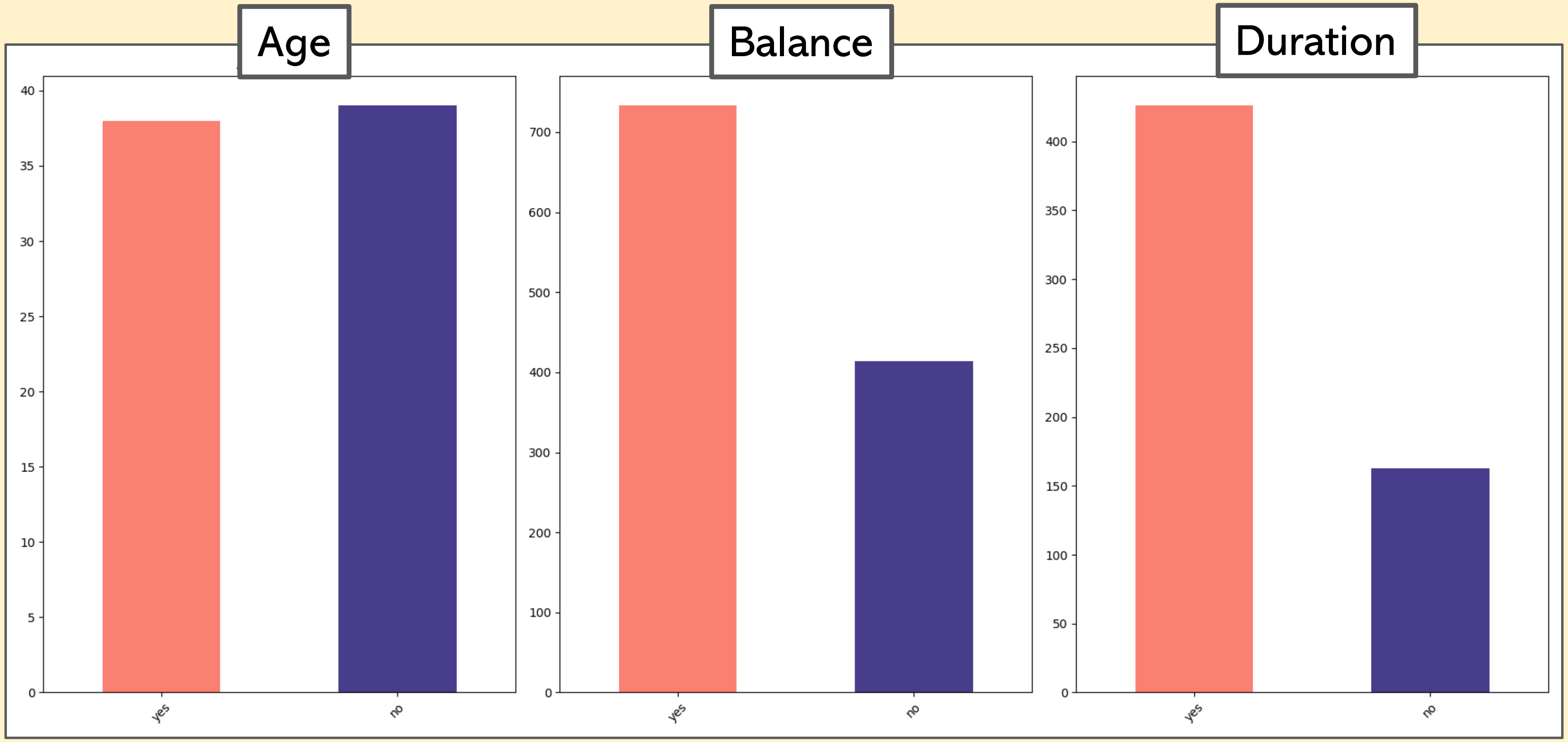
1

By target feature value,

Numeric → Median

Categorical → Mode

Fill missing data to specific values [Numeric]



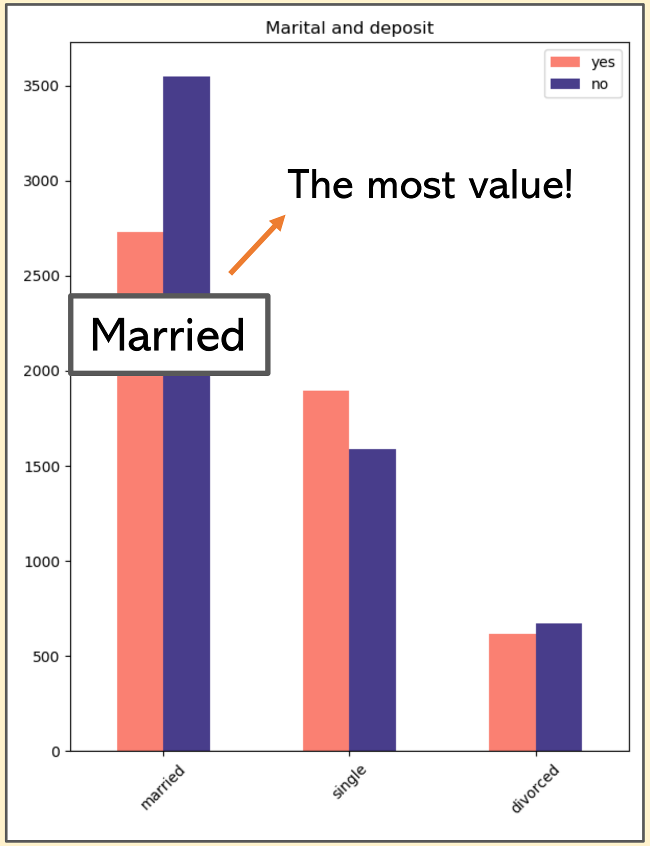
Data Preprocessing

1 By target feature value,

Numeric → Median

Categorical → Mode

Fill missing data to specific values [Categorical]



age	job	marital	education	deposit
	managem	married		yes
64	retired	married	secondary	yes
48	managem	married		no
59	housemaid	married	secondary	no
55	services	married		no
	blue-collar	married		yes
36	managem	married	tertiary	no
	unemploy	married		yes
60	retired	married	secondary	no
43	blue-collar	married	secondary	yes
35	technician	married	secondary	yes
	services	married		no
	housemaid	married	primary	no
	managem	married		no

If deposit value is "Yes"
→ Fill to 'married'

If deposit value is "No"
→ Fill to 'married'

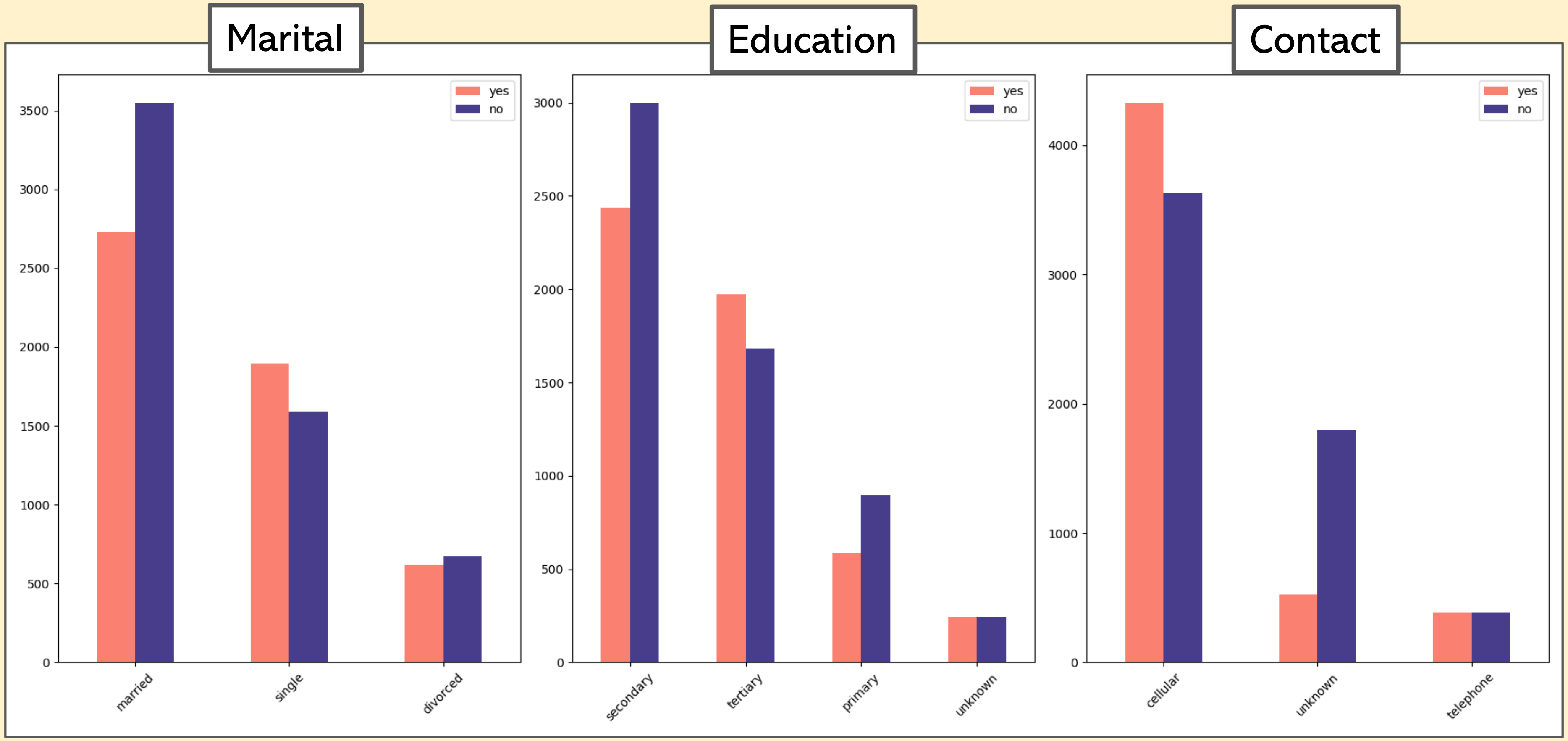
Data Preprocessing

1 By target feature value,

Numeric → Median

Categorical → Mode

Fill missing data to specific values [Categorical]



Data Preprocessing

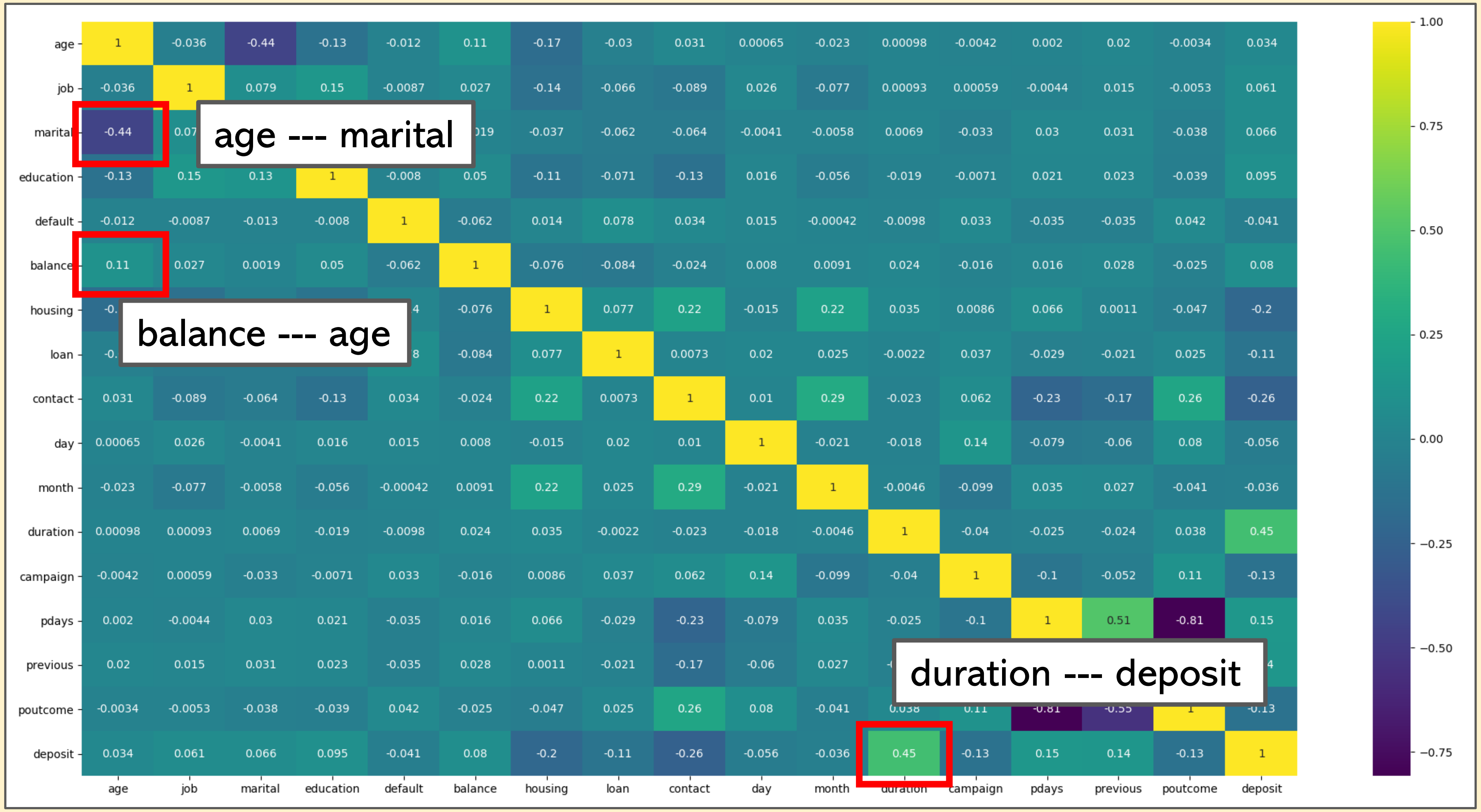
2

By related feature value

Numeric → Linear Regression / Median

Categorical → Mode

Look correlation heatmap and find relational column



Data Preprocessing

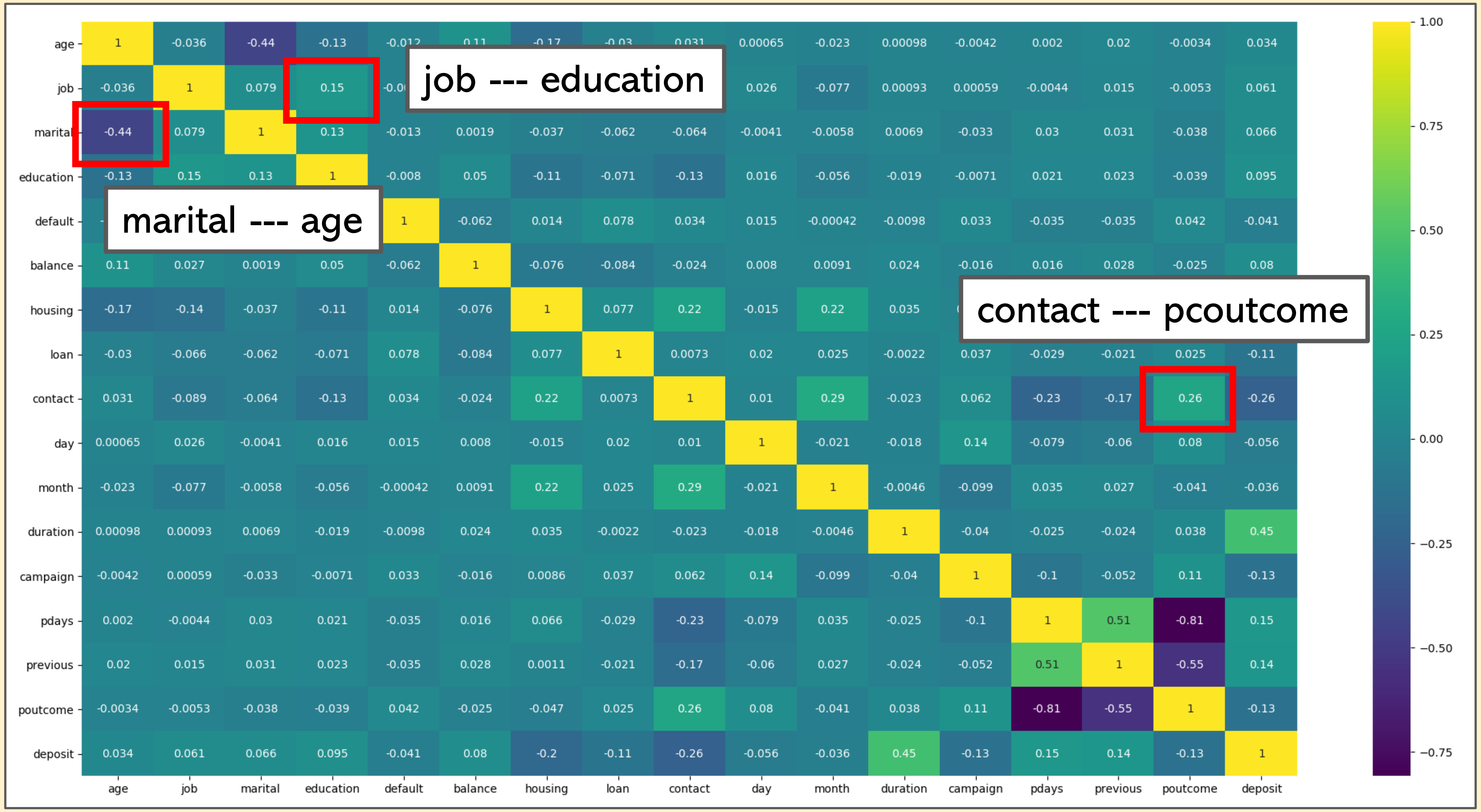
2

By related feature value

Numeric → Linear Regression / Median

Categorical → Mode

Look correlation heatmap and find relational column



Data Preprocessing

2

By **related feature** value

Numeric → **Linear Regression / Median**

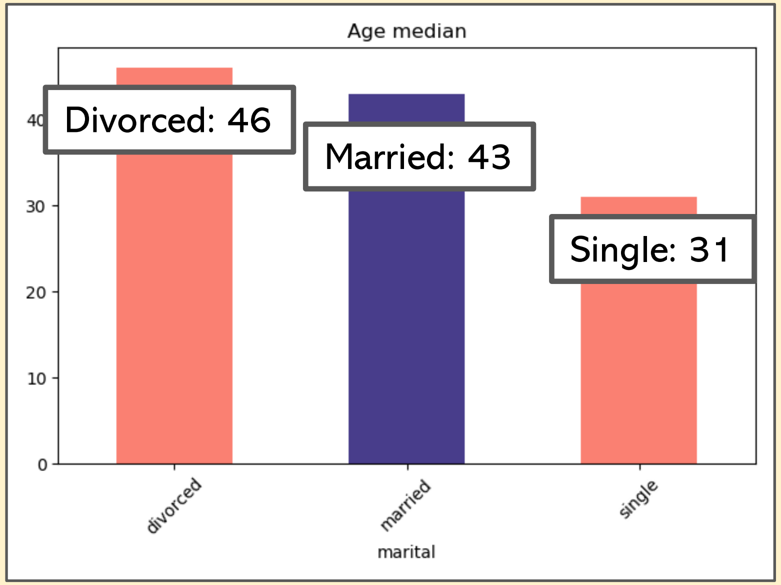
Categorical → Mode

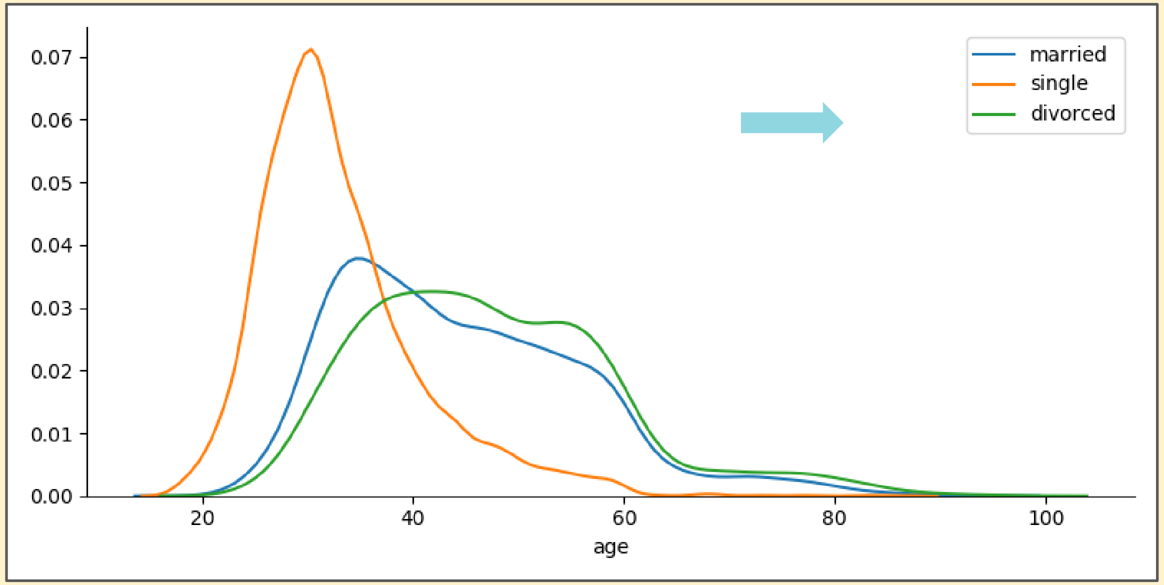
Fill missing data to specific values [Numeric]

If marital value is "divorced" → Fill age to '46'

If marital value is "married" → Fill age to '43'

If marital value is "single" → Fill age to '31'





Data Preprocessing

2 By related feature value

Numeric → Linear Regression / Median

Categorical → Mode

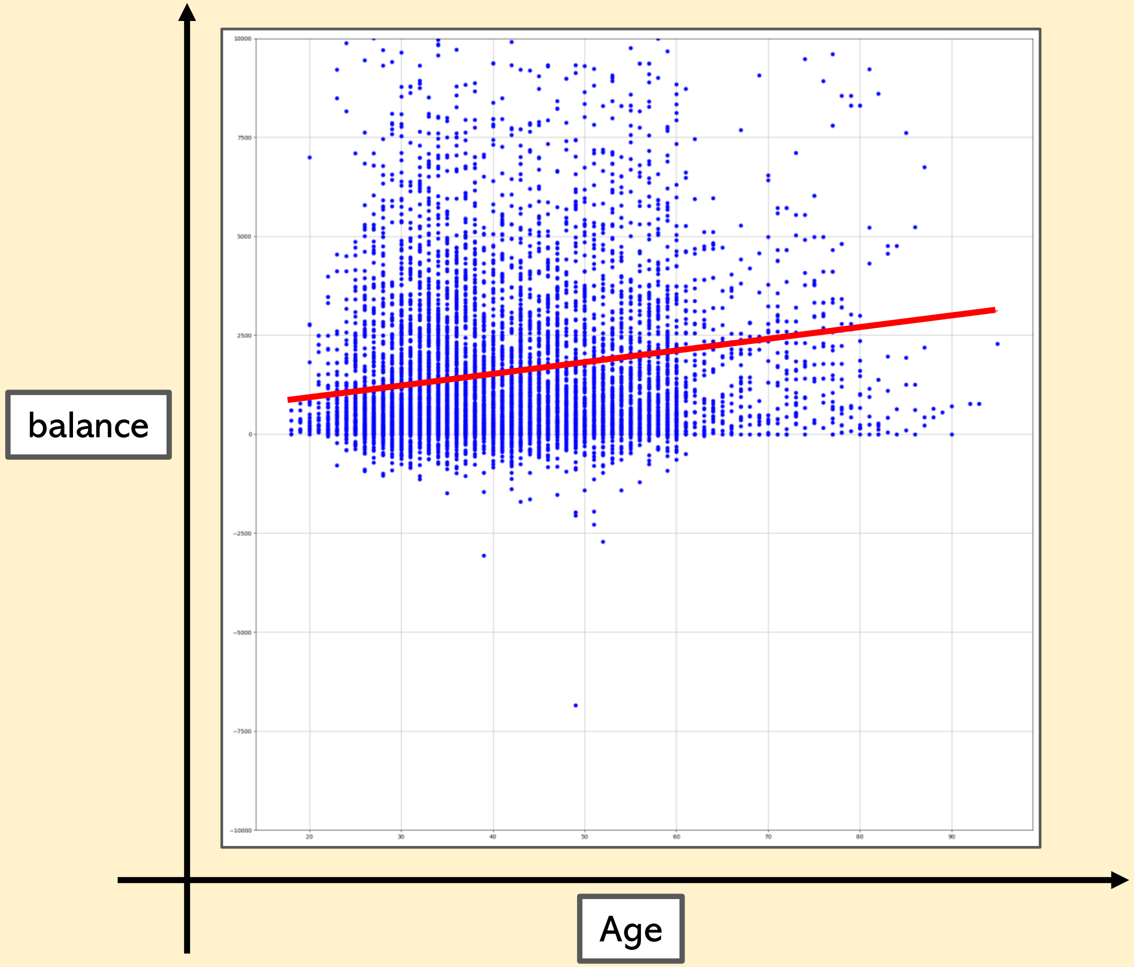
Fill missing data to specific values [Numeric]

Linear Regression

$$Y = 29.679 * X + 302.99$$

If age value is "20"

→ '29.679*20 + 302.99'



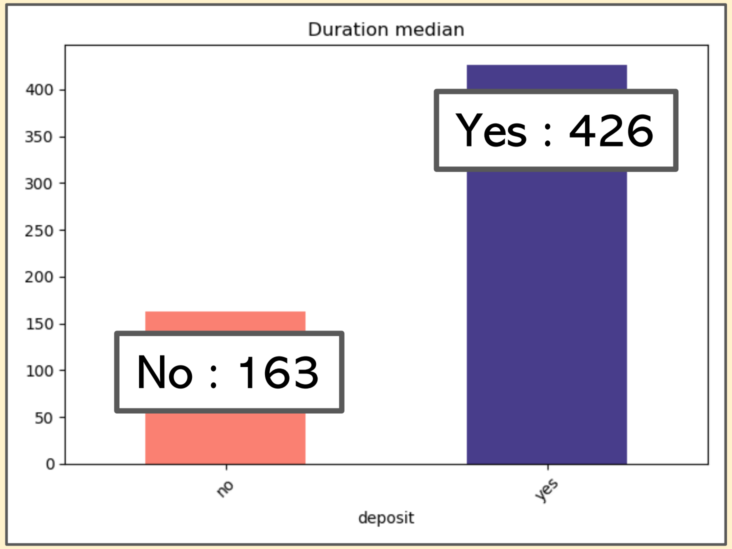
Data Preprocessing

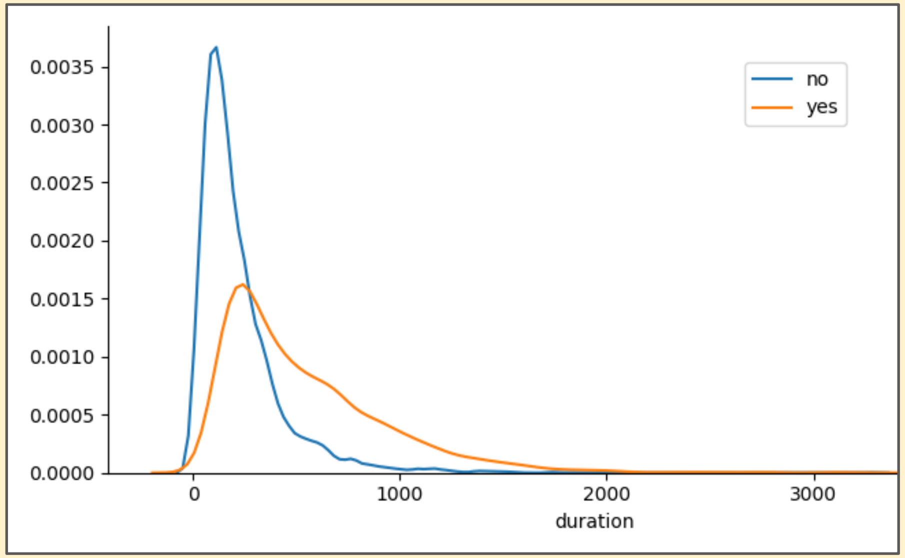
2 By related feature value

Numeric → Linear Regression / Median

Categorical → Mode

Fill missing data to specific values [Numeric]





If deposit value is "Yes"
➡Fill duration to '163'

If deposit value is "No"
➡Fill duration to '426'

Data Preprocessing

2 By related feature value

Numeric → Linear Regression / Median

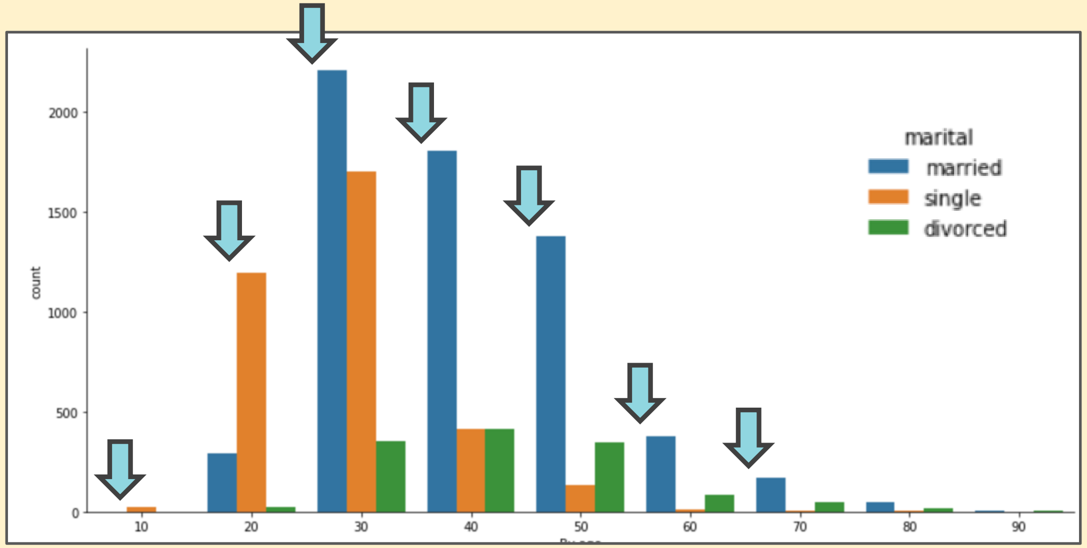
Categorical → Mode

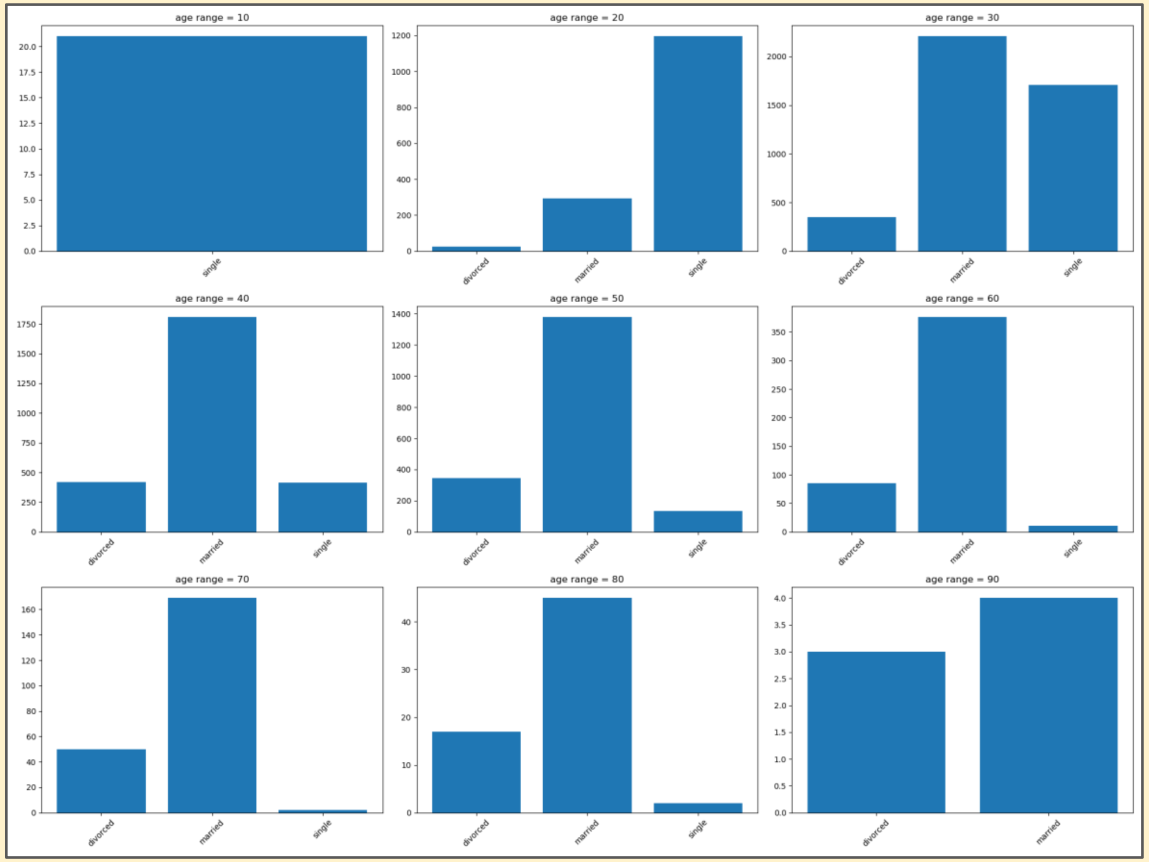
Fill missing data to specific values [Categorical]

If 20's → "Single"

If 30's → "Married"

If 40's → "Married"





Data Preprocessing

2 **By related feature value**

Numeric → Linear Regression / Median

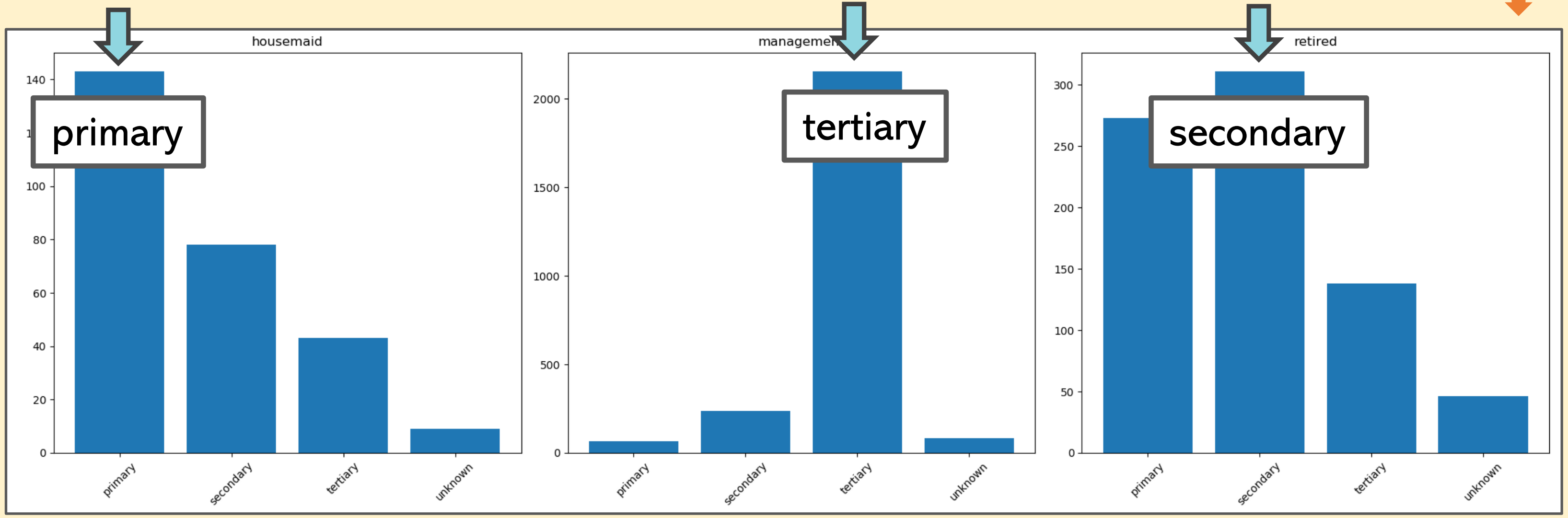
Categorical → Mode

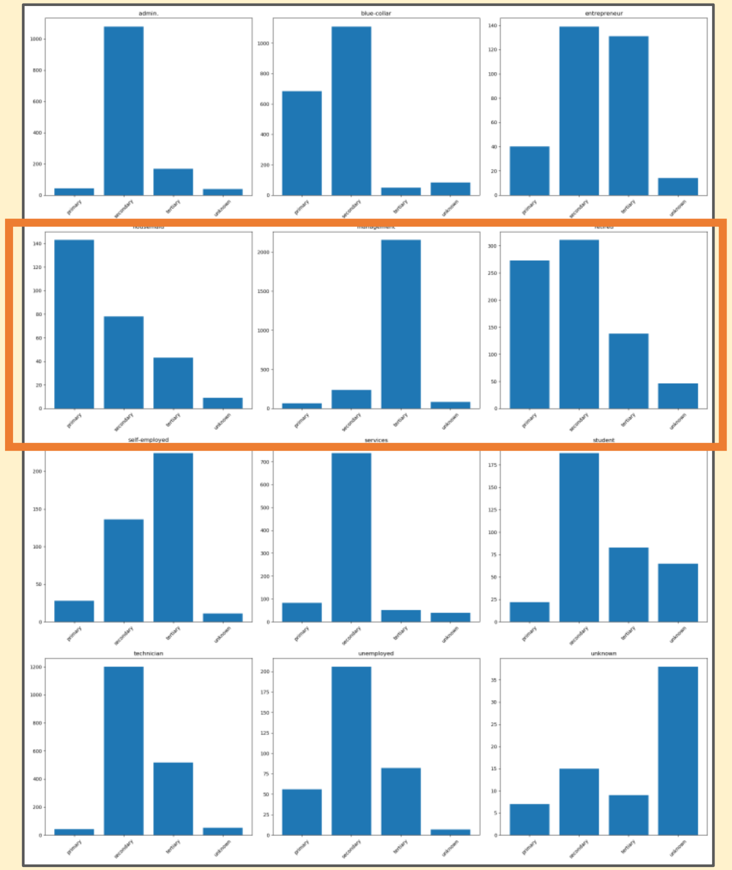
Fill missing data to specific values [Categorical]

If the job is "housemaid" → "primary"

If the job is "management" → "tertiary"

If "retired" → "secondary"





Data Preprocessing

2 By related feature value

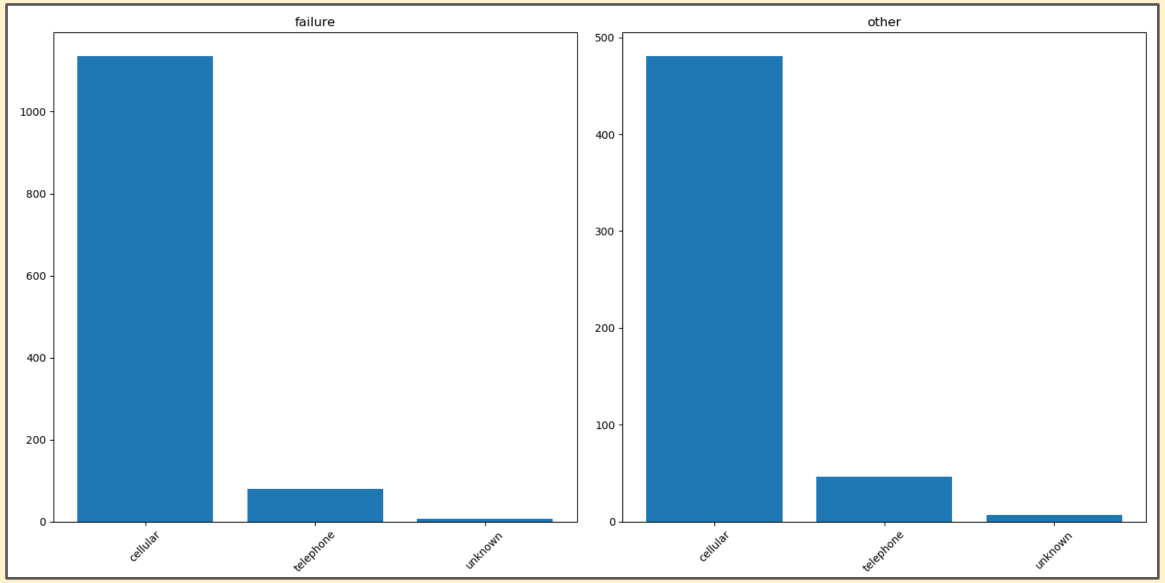
Numeric → Linear Regression / Median

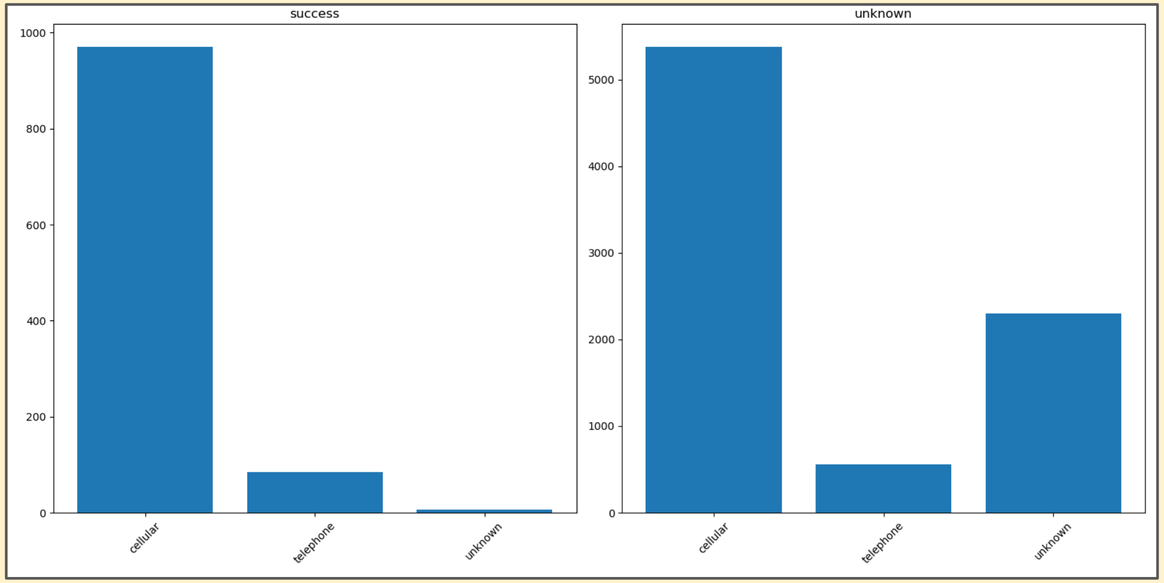
Categorical → Mode

Fill missing data to specific values [Categorical]

If contact is "success"
➡ Fill contact to 'cellular'

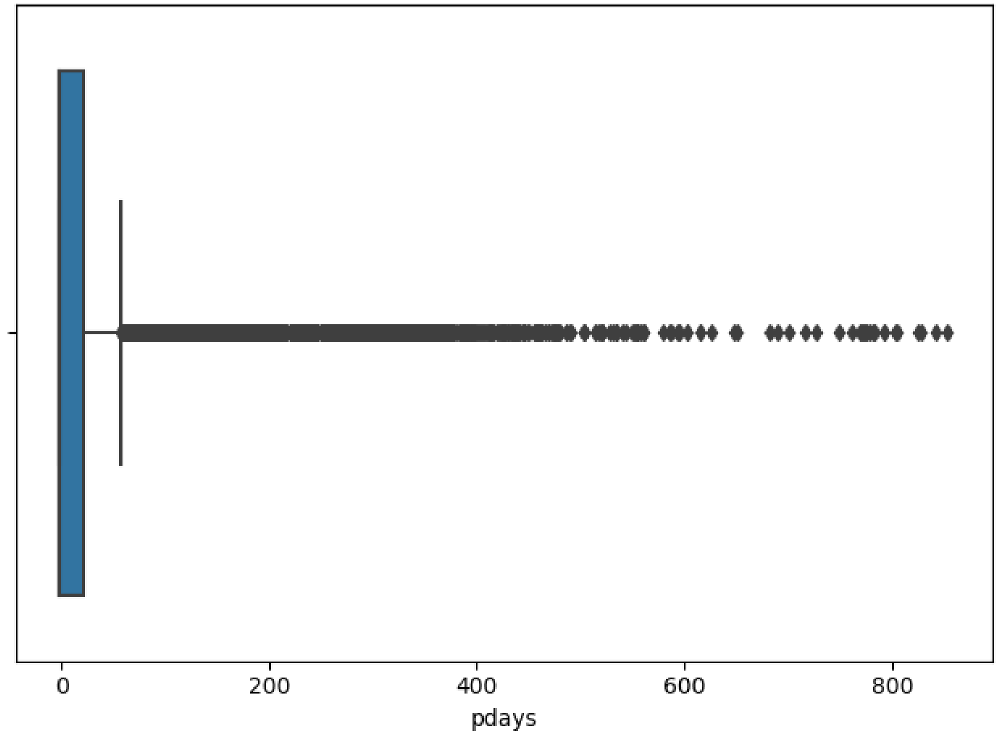
If contact is "failure"
➡ Fill contact to 'cellular'





Data Preprocessing

Outlier Data



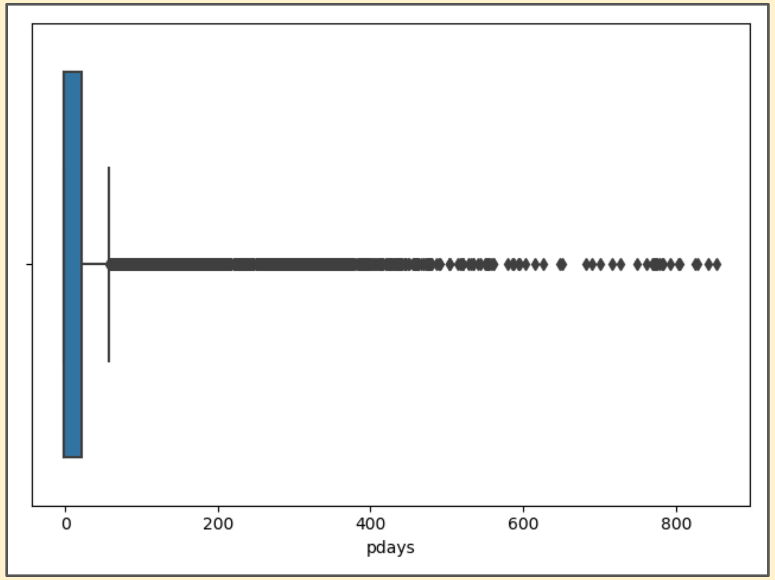
Percentage of -1 of 'pdays' column : 74.5%

day	month	duration	campaign	pdays	previous	poutcome	deposit
9	jul	426	3	-1	0	unknown	yes
3	jun	406	2	-1	0	unknown	yes
29	sep	316	1	119	1	other	yes
30	apr	121	1	63	2	failure	no
7	jul	480	1	-1	0	unknown	no
18	nov	66	1	-1	0	unknown	no
22	aug	1123	4	-1	0	unknown	yes
2	jul	217	3	-1	0	unknown	yes
3	nov	412	1	-1	0	unknown	yes
29	may	814	2	-1	0	unknown	yes
22	oct	554	3	-1	0	unknown	yes
18	feb	386	1	-1	0	unknown	yes
12	aug	768	2	-1	0	unknown	yes
18	feb	332	2	-1	0	unknown	yes
28	aug	195	6	-1	0	unknown	no
18	aug	194	2	-1	0	unknown	yes
20	jun	42	7	-1	0	unknown	no
7	may	28	2	289	5	failure	no
28	jan	111	1	-1	0	unknown	no
30	oct	373	3	-1	0	unknown	yes
21	nov	135	2	-1	0	unknown	no
16	may	38	3	-1	0	unknown	no
8	sep	261	1	98	1	success	yes

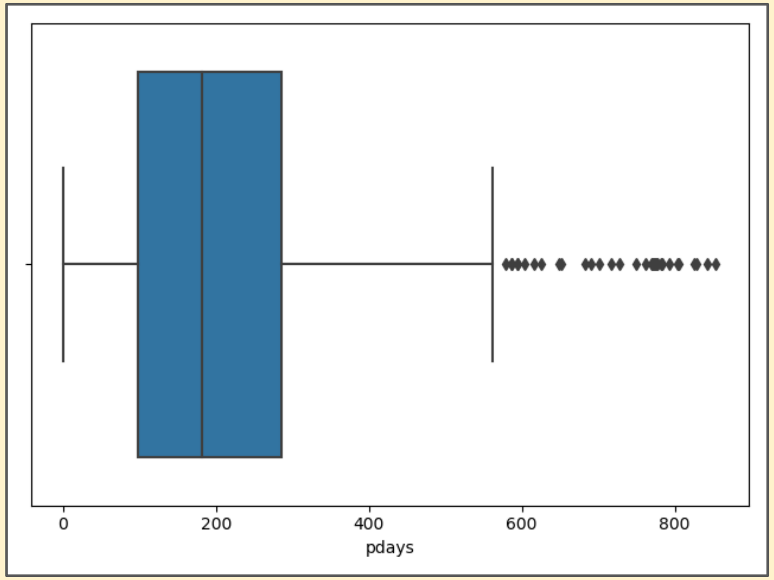
Data Preprocessing

Handle outlier data

Percentage of -1 of ‘pdays’ column : 74.5%



Before remove -1

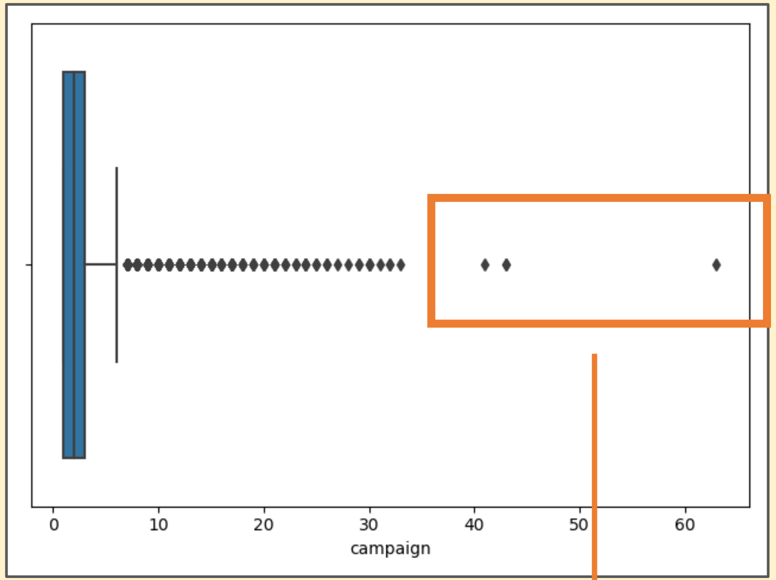


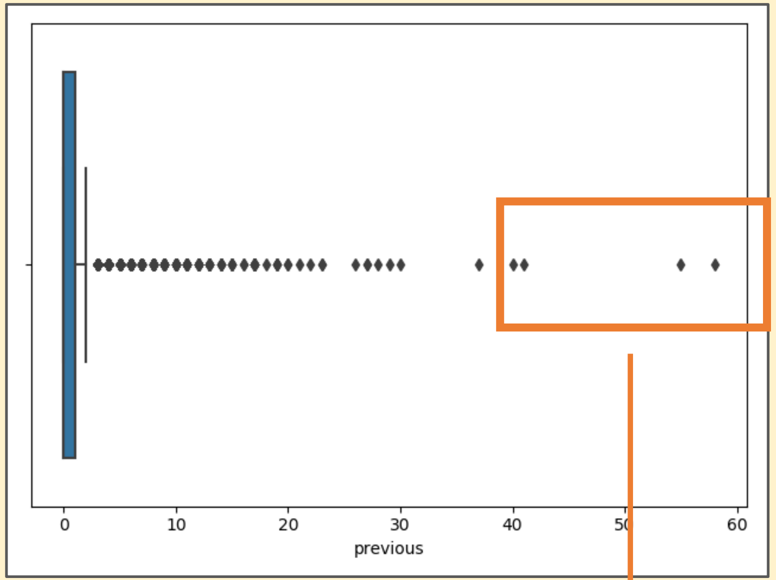
After remove -1

Drop ‘pdays’ feature

Data Preprocessing

Handle outlier data





Above 35

Above 40

Replace to median value !

Data Preprocessing

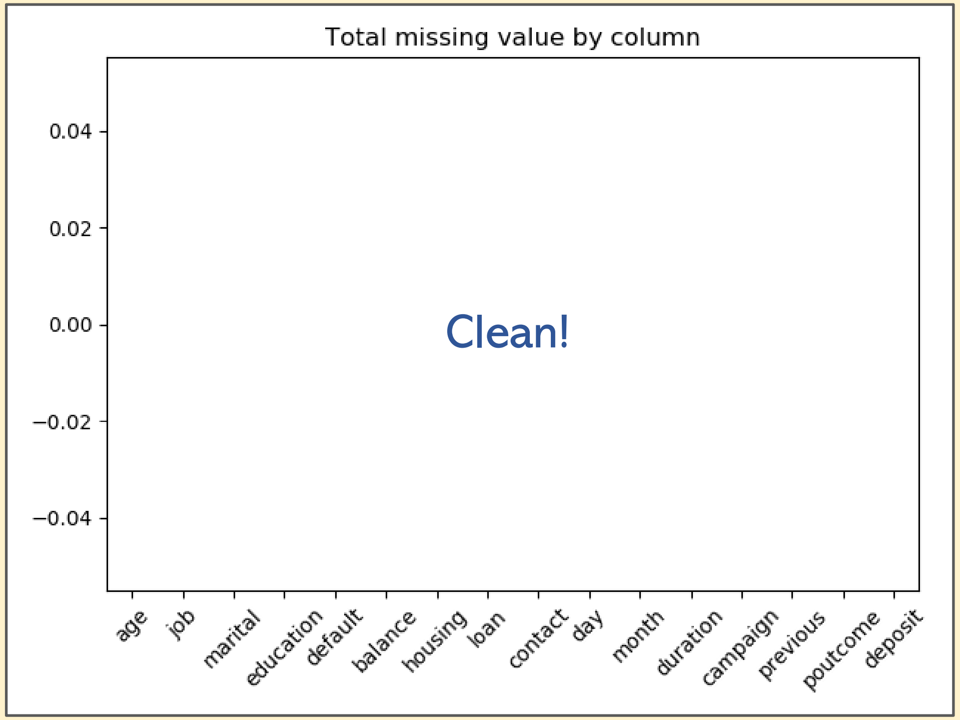
Encoding

age	job	marital	education	default	balance	housing
38	managem	married	tertiary	no	2278	yes
50	blue-collar	single	primary	no	1743	yes
40	self-emplc	single	tertiary	no	1616	no
38	technician	married	secondary	no	205	no
55	blue-collar	married	secondary	no	49	yes
33	technician	married	secondary	no	806	no
36	technician	married	secondary	no	4613	no
30	blue-collar	single	secondary	no	155	yes
32	admin.	married	secondary	no	995	no
29	managem	single	tertiary	no	0	yes
46	blue-collar	married	secondary	no	1723	no
26	blue-collar	single	primary	no	-887	yes
36	managem	married	secondary	no	565	no
56	technician	married	secondary	no	147	no
26	blue-collar	single	secondary	no	-46	yes
39	blue-collar	married	primary	no	-50	yes
51	managem	married	tertiary	no	-321	no

age	job	marital	education	default	balance	housing
38	4	1	2	0	2278	1
50	1	2	0	0	1743	1
40	6	2	2	0	1616	0
38	9	1	1	0	205	0
55	1	1	1	0	49	1
33	9	1	1	0	806	0
36	9	1	1	0	4613	0
30	1	2	1	0	155	1
32	0	1	1	0	995	0
29	4	2	2	0	0	1
46	1	1	1	0	1723	0
26	1	2	0	0	-887	1
36	4	1	1	0	565	0
56	9	1	1	0	147	0
26	1	2	1	0	-46	1
39	1	1	0	0	-50	1
51	4	1	2	0	-321	0

Data Preprocessing

Complete cleaning!



Percentage of that is missing : 0 %

Missing value existence status

age	False
job	False
marital	False
education	False
default	False
balance	False
housing	False
loan	False
contact	False
day	False
month	False
duration	False
campaign	False
previous	False
poutcome	False
deposit	False
dtype: bool	

How many missing value?

age	0
job	0
marital	0
education	0
default	0
balance	0
housing	0
loan	0
contact	0
day	0
month	0
duration	0
campaign	0
previous	0
poutcome	0
deposit	0
dtype: int64	

Percentage of missing value

age	0.0
job	0.0
marital	0.0
education	0.0
default	0.0
balance	0.0
housing	0.0
loan	0.0
contact	0.0
day	0.0
month	0.0
duration	0.0
campaign	0.0
previous	0.0
poutcome	0.0
deposit	0.0
dtype: float64	

Data Analysis

Data Analysis

Data Scaling

```
X = processed_df.drop(columns='deposit')
y = processed_df['deposit']
X = scaling(X)
```

before scaling

	age	job	marital	education	...	duration	campaign	previous	poutcome
0	38.0	4	1	2	...	244.0	1.0	0.0	3
1	50.0	1	2	0	...	49.0	5.0	0.0	3
2	40.0	6	2	2	...	1009.0	7.0	0.0	3
3	38.0	9	1	1	...	332.0	1.0	0.0	3
4	55.0	1	1	1	...	494.0	4.0	0.0	3
...	...	...	...	...	...	...	...	...	...
11157	26.0	6	2	2	...	446.0	1.0	0.0	3
11158	41.0	9	2	1	...	2420.0	3.0	0.0	3
11159	29.0	4	2	2	...	963.0	2.0	0.0	3
11160	31.0	1	1	1	...	295.0	1.0	0.0	3
11161	68.0	5	1	1	...	318.0	2.0	0.0	3

[11112 rows x 15 columns]

After scaling

	age	job	marital	...	campaign	previous	poutcome
0	0.259740	0.363636	0.5	...	0.00000	0.0	1.0
1	0.415584	0.090909	1.0	...	0.12500	0.0	1.0
2	0.285714	0.545455	1.0	...	0.18750	0.0	1.0
3	0.259740	0.818182	0.5	...	0.00000	0.0	1.0
4	0.480519	0.090909	0.5	...	0.09375	0.0	1.0
...	...	...	...	...	...	...	...
11157	0.103896	0.545455	1.0	...	0.00000	0.0	1.0
11158	0.298701	0.818182	1.0	...	0.06250	0.0	1.0
11159	0.142857	0.363636	1.0	...	0.03125	0.0	1.0
11160	0.168831	0.090909	0.5	...	0.00000	0.0	1.0
11161	0.649351	0.454545	0.5	...	0.03125	0.0	1.0

[11112 rows x 15 columns]

Data Analysis

Analysis Algorithm

K-Nearest Neighbors

```
classifier1 = KNeighborsClassifier()
classifier1.fit(X_train, y_train)
classifier1.predict(X_test)
```

Decision Tree

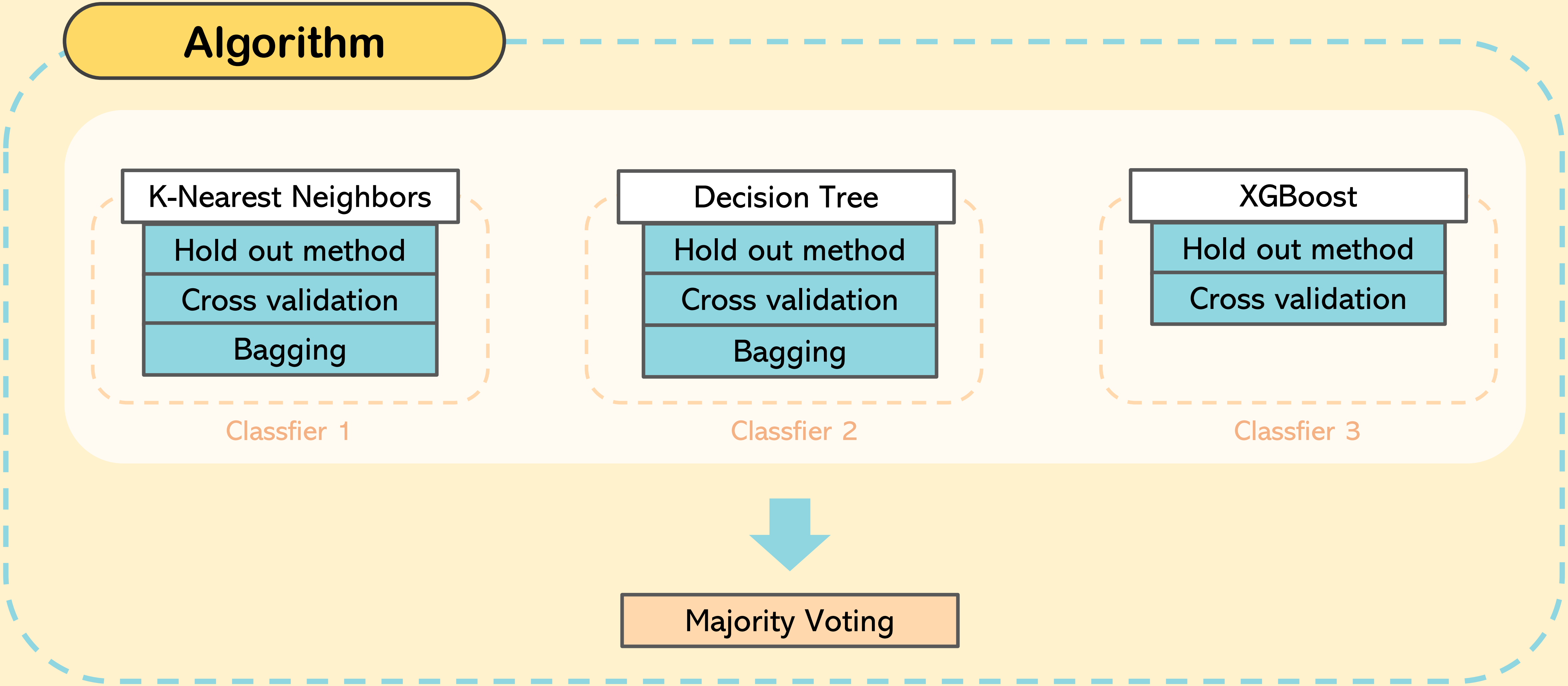
```
classifier1 = DecisionTreeClassifier(criterion='entropy', random_state=0)
classifier1.fit(X_train, y_train)
classifier1.predict(X_test)
```

XGBoost

```
classifier1 = xgb.XGBClassifier()
classifier1.fit(X_train, y_train.squeeze().values)
classifier1.predict(X_test)
```

Data Evaluation

Data Evaluation



Data Evaluation

Hypertuning By GridSearchCV !!

Algorithm

K-Nearest Neighbors	Decision Tree	XGBoost
Hold out method	Hold out method	Hold out method
Cross validation	Cross validation	Cross validation
Bagging	Bagging	
Classfier 1	Classfier 2	Classfier 3

```
knn_gscv = GridSearchCV(classifier2, param_grid, cv=cv)
knn_gscv.fit(X, y)
bestParams = knn_gscv.best_params_
bestEstimator = knn_gscv.best_estimator_
```

Get best parameter

Data Evaluation

Example – Decision Tree

1. Separate from the train set to the training set and the test set by hold-out method
2. Perform default parameters without Hyper-parameter tuning.
3. Tuning Hyper-parameter Using GridSearchCV
4. Using tuned Hyper-parameter, refit and predict
5. Compare prediction of holdout method with defaultparameter and with tuned hyper-parameter
6. Cross validation with tuned hyper-parameter
7. Use tuned hyper-parameter for bagging algorithm

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
classifier1 = DecisionTreeClassifier(criterion='entropy', random_state=0)
classifier1.fit(X_train, y_train)
# Predicting the best set result
y_pred = classifier1.predict(X_test)
```

Data Evaluation

Example – Decision Tree

1. Separate from the train set to the training set and the test set by hold-out method
2. Perform default parameters without Hyper-parameter tuning.
3. Tuning Hyper-parameter Using GridSearchCV
4. Using tuned Hyper-parameter, refit and predict
5. Compare prediction of holdout method with defaultparameter and with tuned hyper-parameter
6. Cross validation with tuned hyper-parameter
7. Use tuned hyper-parameter for bagging algorithm

```
classifier2 = DecisionTreeClassifier(criterion='entropy', random_state=0)
param_grid = {'max_depth': np.arange(1, 30)}

cv = KFold(n_splits=5, shuffle=True, random_state=1)
dt_gscv = GridSearchCV(classifier2, param_grid, cv=cv)
dt_gscv.fit(X, y)
bestParams = dt_gscv.best_params_
```

Data Evaluation

Example – Decision Tree

1. Separate from the train set to the training set and the test set by hold-out method
2. Perform default parameters without Hyper-parameter tuning.
3. Tuning Hyper-parameter Using GridSearchCV
4. Using tuned Hyper-parameter, refit and predict
5. Compare prediction of holdout method with defaultparameter and with tuned hyper-parameter
6. Cross validation with tuned hyper-parameter
7. Use tuned hyper-parameter for bagging algorithm

```
classifier2 = DecisionTreeClassifier(max_depth=bestParams['max_depth'])
classifier2.fit(X_train, y_train)

# Predicting the best set result
y_pred = classifier2.predict(X_test)
```

```
========== Holdout method(Decision Tree) ==========
Best Parameter:  {'max_depth': 8}
```

	Model	Accuracy	Precision	Recall	F1 Score
0	Decision Tree (default = None)	0.796671	0.804209	0.786929	0.795475
1	Decision Tree (Best max depth)	0.821413	0.821429	0.823635	0.822530

Data Evaluation

Example – Decision Tree

1. Separate from the train set to the training set and the test set by hold-out method
2. Perform default parameters without Hyper-parameter tuning.
3. Tuning Hyper-parameter Using GridSearchCV
4. Using tuned Hyper-parameter, refit and predict
5. Compare prediction of holdout method with defaultparameter and with tuned hyper-parameter
6. Cross validation with tuned hyper-parameter
7. Use tuned hyper-parameter for bagging algorithm

```
cvResults = predictCVResult(X, y, classifier2, 'Decision Tree', cvResults)
```

```
baggingResults = predictBaggingResult(X_train, X_test, y_train, y_test, bestEstimator, 'Decision Tree',
                                       baggingResults)
```

Data Evaluation

Result – Parameter tuning

Accuracy increases ↑ with parameter tuning.

```
========== Holdout method(K-Nearest Neighbors) ==========
Best Parameter:  {'n_neighbors': 3}
                          Model  Accuracy  Precision    Recall  F1 Score
0  K-Nearest Neighbors (default = 5)  0.706253  0.752174  0.619517  0.679431
1       K-Nearest Neighbors (Best k)  0.716149  0.756329  0.641898  0.694431
```

```
========== Holdout method(Decision Tree) ==========
Best Parameter:  {'max_depth': 8}
                          Model  Accuracy  Precision    Recall  F1 Score
0  Decision Tree (default = None)  0.796671  0.804209  0.786929  0.795475
1  Decision Tree (Best max depth)  0.821413  0.821429  0.823635  0.822530
```

```
==========  Holdout method(XGBoost) ==========
Best Parameter:  {'max_depth': 6, 'min_child_weight': 1, 'n_estimators': 200}
                          Model  Accuracy  Precision    Recall  F1 Score
0             XGBoost (default)  0.835358  0.827376  0.849597  0.838339
1  XGBoost (best parameters)  0.853351  0.837746  0.878245  0.857517
```

Data Evaluation

Result – Comparison of preprocessing method

In the second cleaning method,
the decision tree and XGBoost accuracy are better.

Cleaning 1

========== Holdout method(K-Nearest Neighbors) ==========

Best Parameter: {'n_neighbors': 5}

	Model	Accuracy	Precision	Recall	F1 Score
0	K-Nearest Neighbors (default = 5)	0.717049	0.745516	0.623243	0.678918
1	K-Nearest Neighbors (Best k)	0.717049	0.745516	0.623243	0.678918

========== Holdout method(Decision Tree) ==========

Best Parameter: {'max_depth': 9}

	Model	Accuracy	Precision	Recall	F1 Score
0	Decision Tree (default = None)	0.776878	0.777994	0.748828	0.763133
1	Decision Tree (Best max depth)	0.805668	0.793167	0.805061	0.799070

========== Holdout method(XGBoost) ==========

Best Parameter: {'max_depth': 6, 'min_child_weight': 1, 'n_estimators': 100}

	Model	Accuracy	Precision	Recall	F1 Score
0	XGBoost (default)	0.839856	0.817128	0.858482	0.837294
1	XGBoost (best parameters)	0.848403	0.825893	0.866917	0.845908

Cleaning 2

========== Holdout method(K-Nearest Neighbors) ==========

Best Parameter: {'n_neighbors': 3}

	Model	Accuracy	Precision	Recall	F1 Score
0	K-Nearest Neighbors (default = 5)	0.706253	0.752174	0.619517	0.679431
1	K-Nearest Neighbors (Best k)	0.716149	0.756329	0.641898	0.694431

========== Holdout method(Decision Tree) ==========

Best Parameter: {'max_depth': 8}

	Model	Accuracy	Precision	Recall	F1 Score
0	Decision Tree (default = None)	0.796671	0.804209	0.786929	0.795475
1	Decision Tree (Best max depth)	0.821413	0.821429	0.823635	0.822530

========== Holdout method(XGBoost) ==========

Best Parameter: {'max_depth': 6, 'min_child_weight': 1, 'n_estimators': 200}

	Model	Accuracy	Precision	Recall	F1 Score
0	XGBoost (default)	0.835358	0.827376	0.849597	0.838339
1	XGBoost (best parameters)	0.853351	0.837746	0.878245	0.857517

Data Evaluation

Result - Comparison of three algorithms

========== Holdout method(Parameter tuning) ==========

	Model	Accuracy	Precision	Recall	F1 Score
0	K-Nearest Neighbors	0.716149	0.756329	0.641898	0.694431
1	Decision Tree	0.821413	0.821429	0.823635	0.822530
2	XGBoost	0.853351	0.837746	0.878245	0.857517

========== Cross validation ==========

	Model	Mean accuracy
0	K-Nearest Neighbors	0.717963
1	Decision Tree	0.818844
2	XGBoost	0.858982

XGBoost has the highest accuracy !

========== Bagging Method ==========

	Model	Accuracy	Precision	Recall	F1 Score
0	K-Nearest Neighbors	0.707602	0.745531	0.634736	0.685687
1	Decision Tree	0.827710	0.821930	0.838854	0.830306

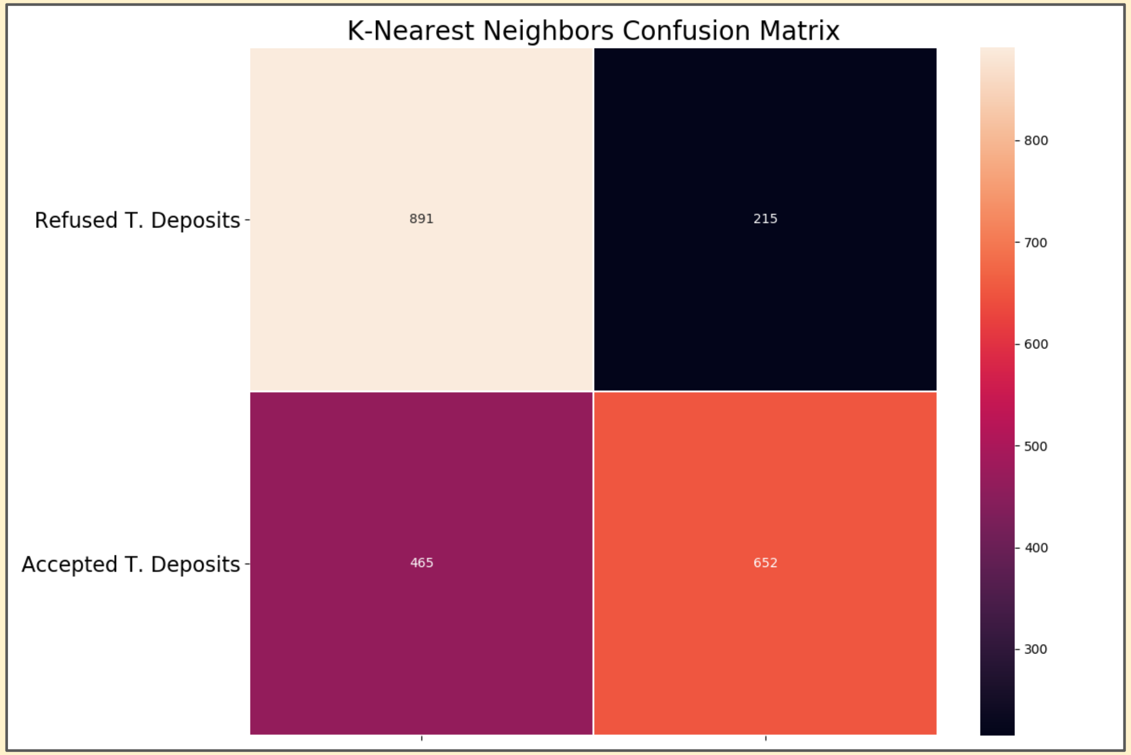
========== Majority voting ==========

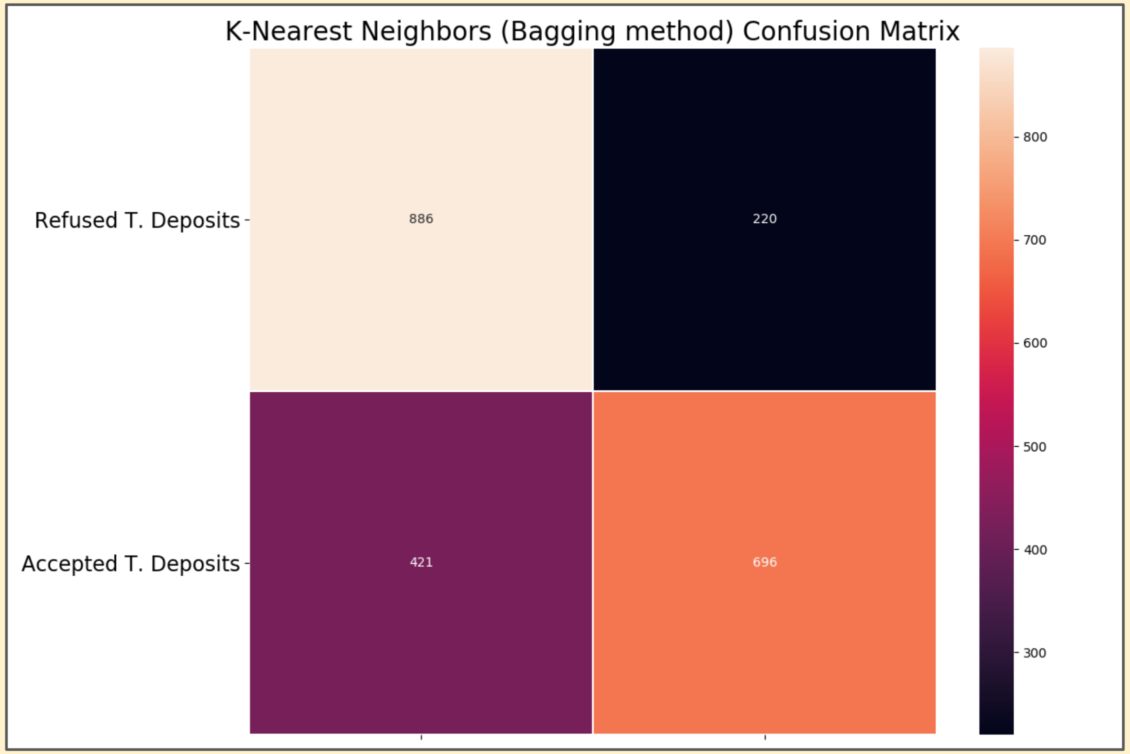
	Accuracy	Precision	Recall	F1 Score
0	0.835807	0.838129	0.834378	0.836249

But did not produce better results in bagging and voting.
When using the bagging method,
the decision tree rose slightly, but the KNN fell slightly.

Data Evaluation

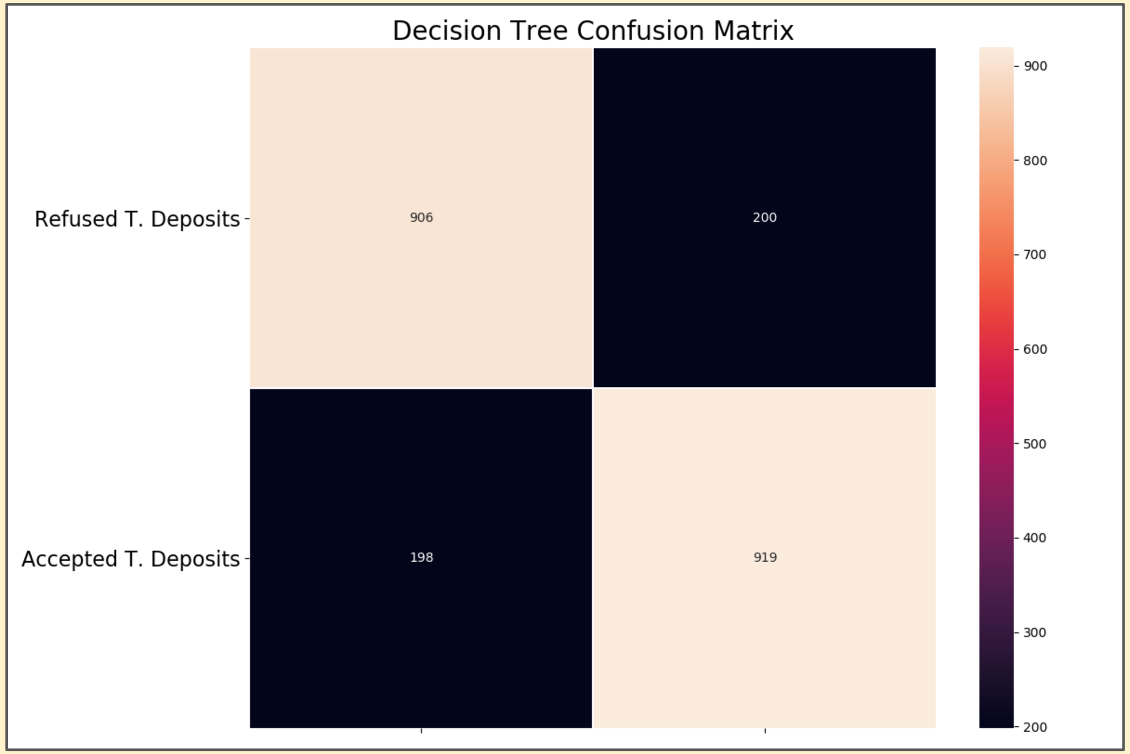
Confusion Matrix : K-Nearest Neighbors

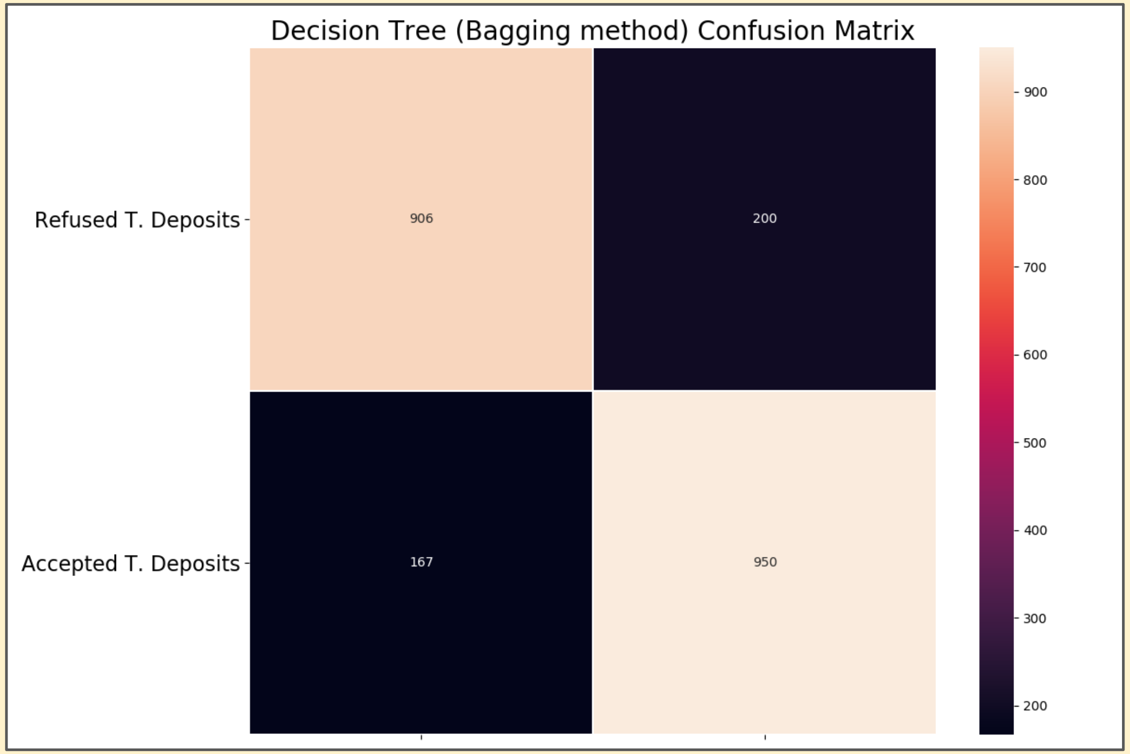




Data Evaluation

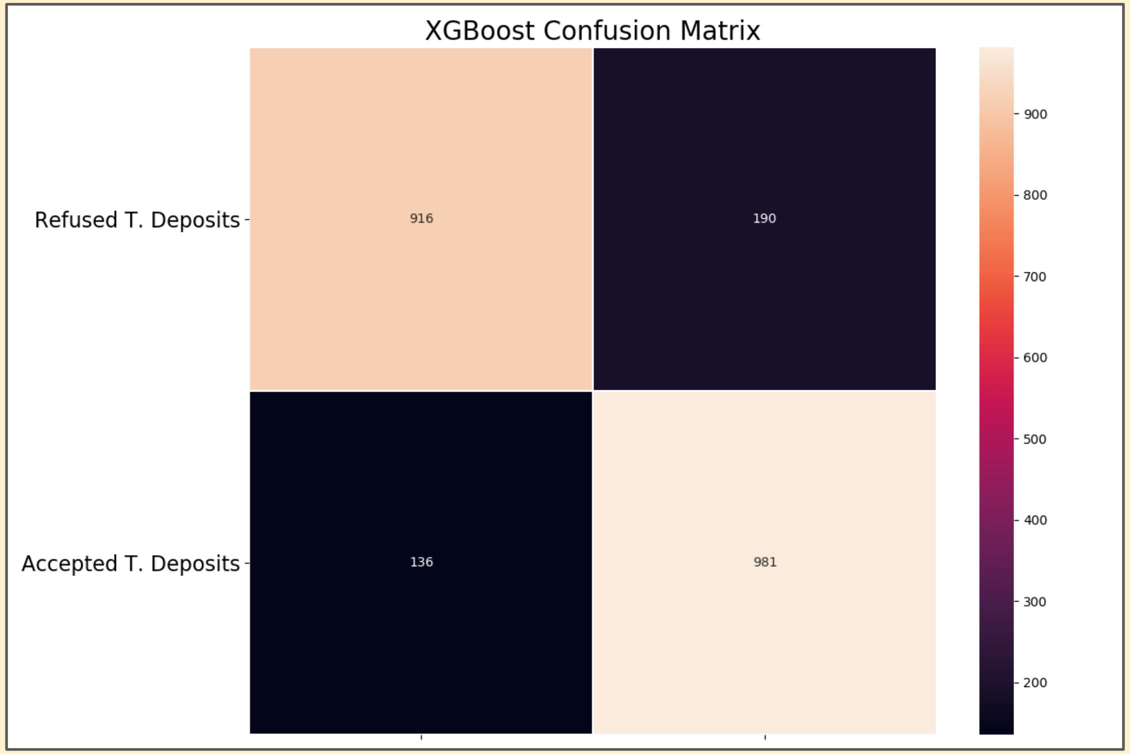
Confusion Matrix : Decision Tree





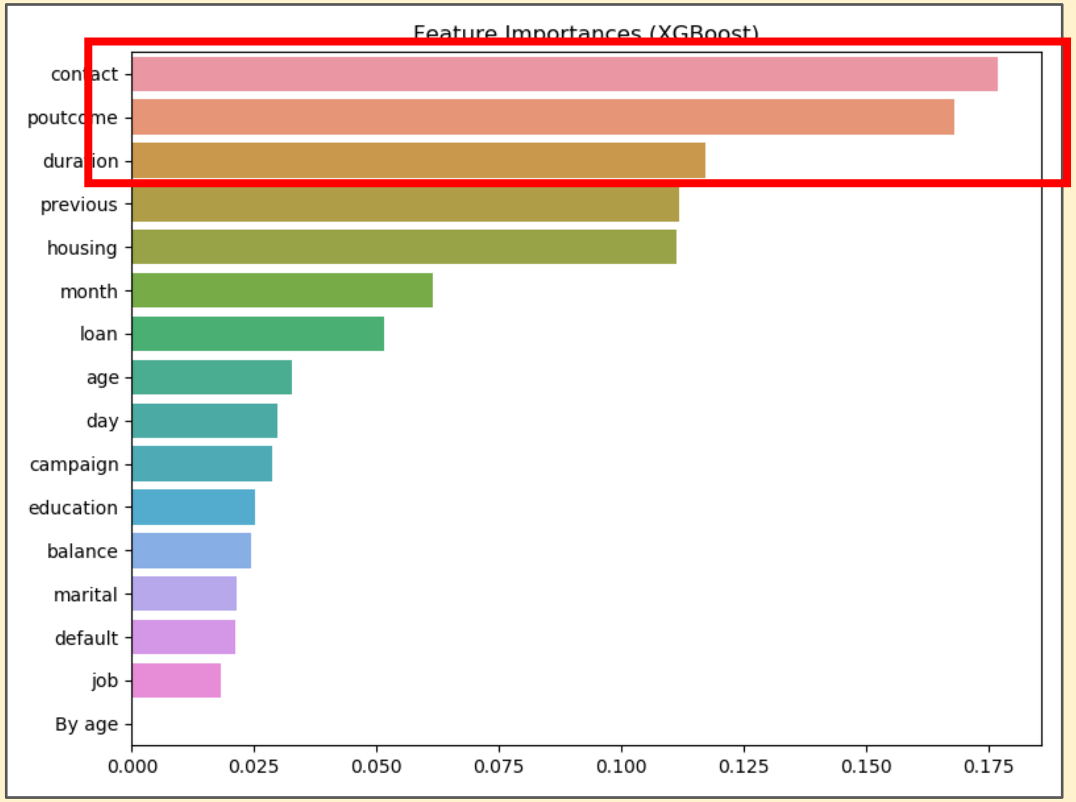
Data Evaluation

Confusion Matrix : XGBoost



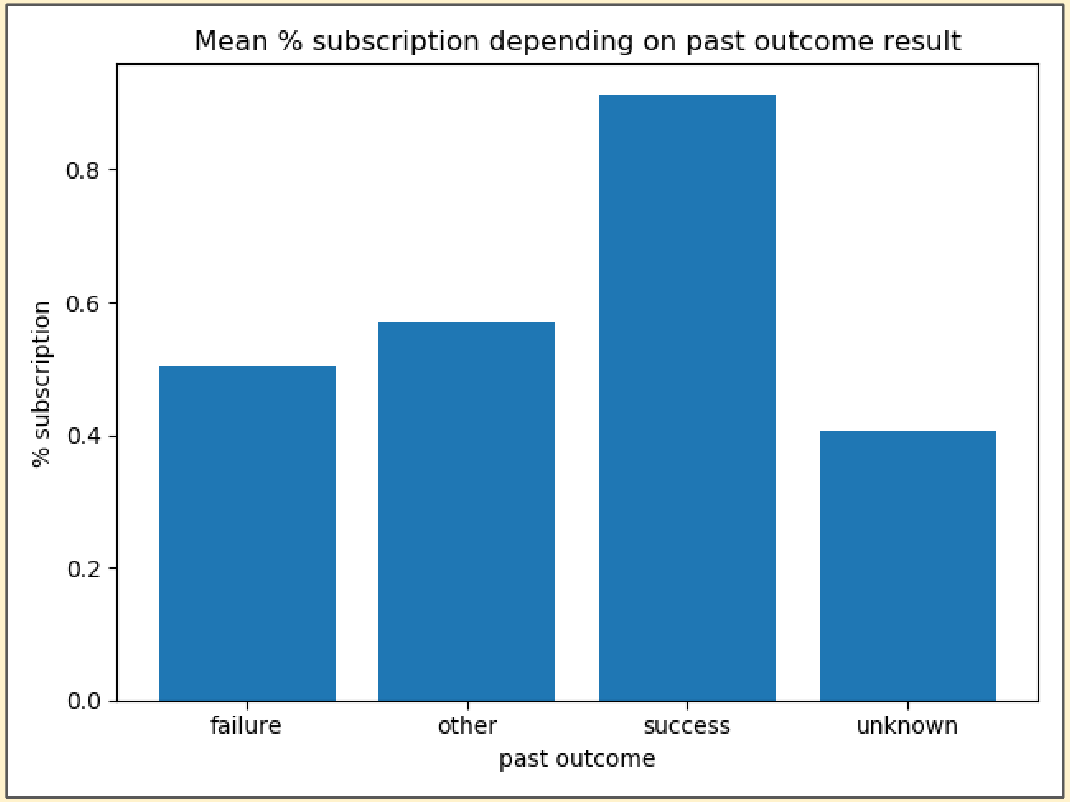
Conclusion

Conclusion

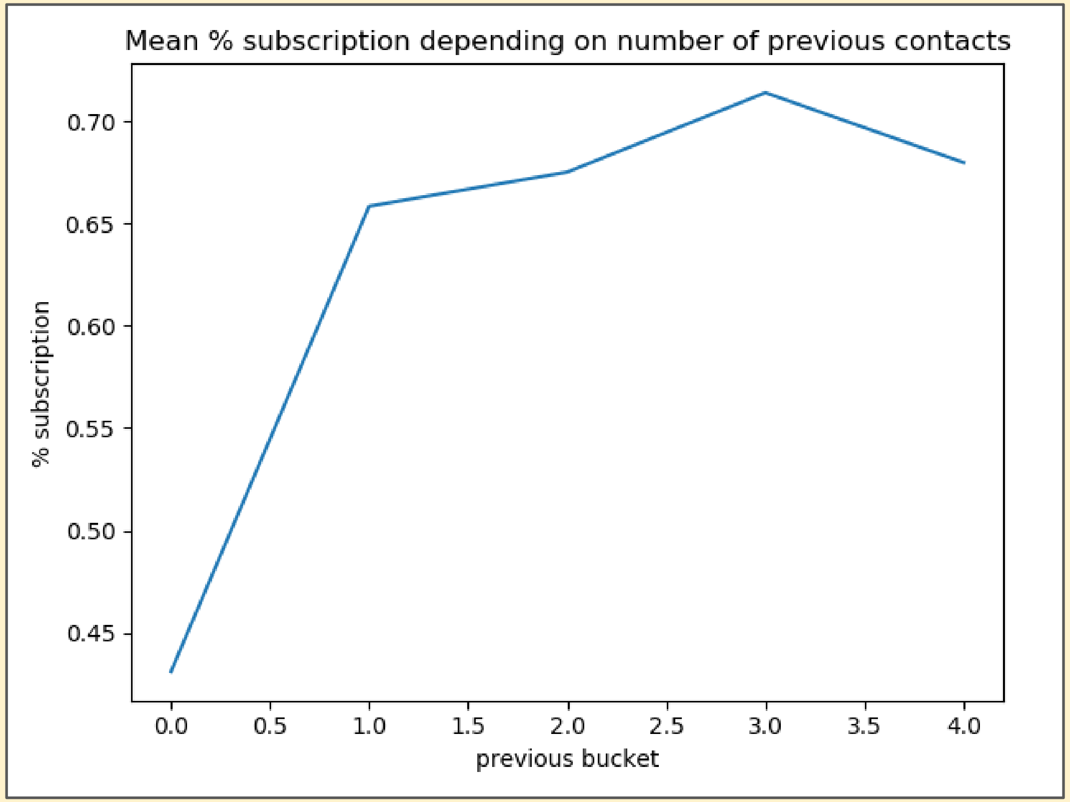


contact, poutcome, duration → important!

Conclusion

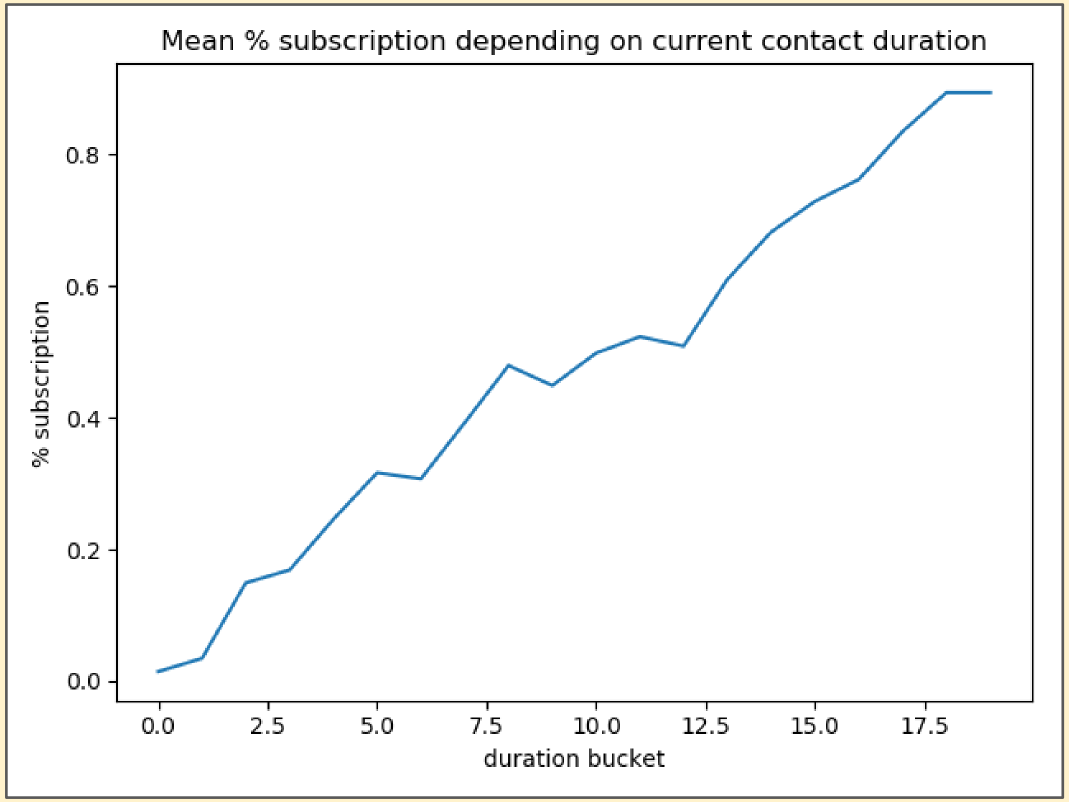


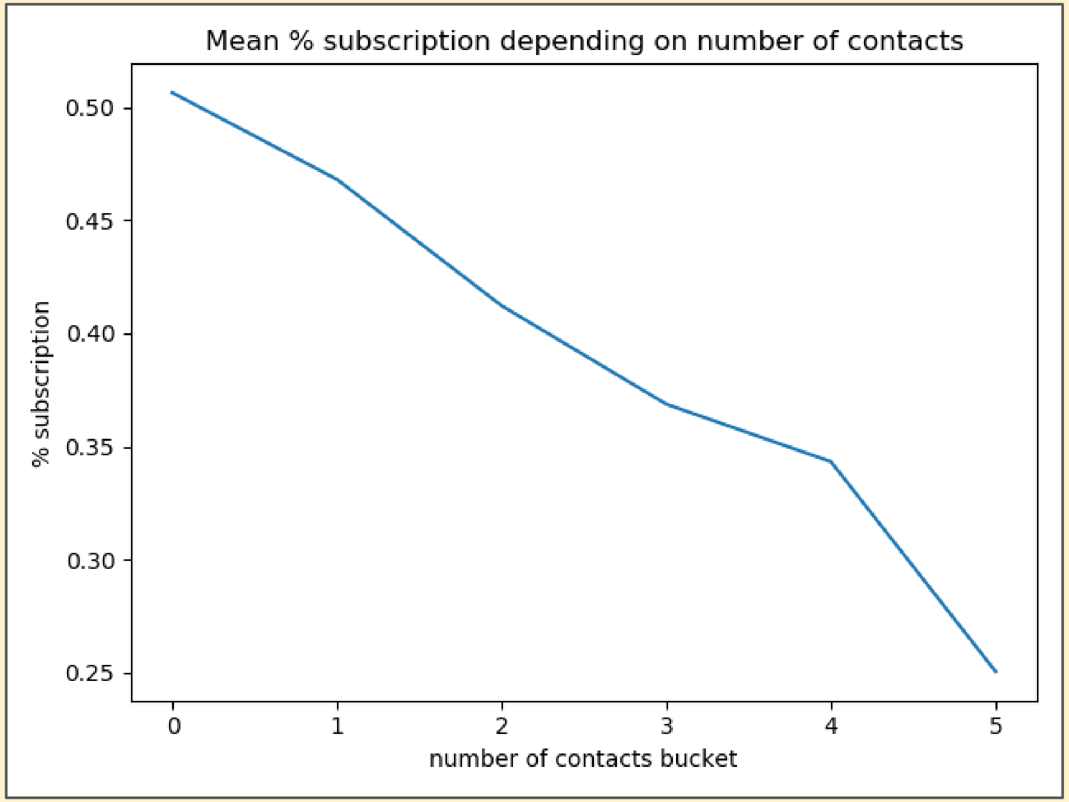
High possibility that people who have subscribed in the last campaign will subscribe in this campaign.



The higher the number of contacts in the previous campaign, the higher the average subscription rate.

Conclusion





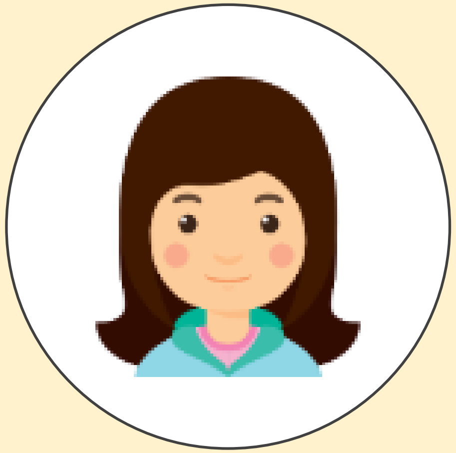
The longer you keep in touch during the campaign, the more likely you are to subscribe.

On the other hand, too much contact during the campaign leads to a lower subscription rate.

Team Member

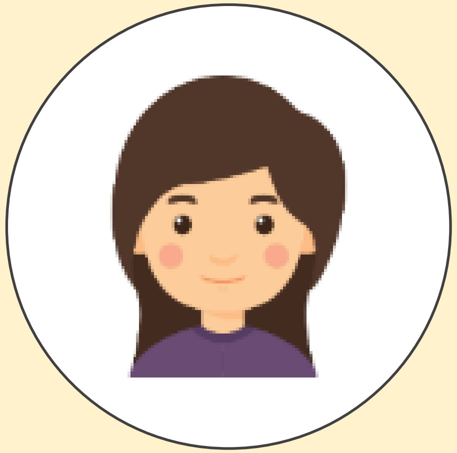
최준헌
201533673
chjh121@gmail.com

Data preprocessing
Data Analysis
Data Evalution
Conclusion



김지현
201633310
zizi39028@gmail.com

Proposal ppt
Graph visualization
Data preprocessing
Final presentation



양희림
201735853
yanghl1998@gmail.com

Data preprocessing
Dataset management
Generate invalid data
Final ppt

THANK YOU