Build a machine learning model to predict if an applicant is 'good' or 'bad' client, different from other tasks, the definition of 'good' or 'bad' is not given. You should use some techique, such as vintage analysis to construct you label. Also, unbalance data problem is a big problem in this task

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

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

```
df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/credit_record.csv')
```

```
df
```

	ID	MONTHS_BALANCE	STATUS
0	5001711	0	X
1	5001711	-1	0
2	5001711	-2	0
3	5001711	-3	0
4	5001712	0	C
...	...	...	...
1048570	5150487	-25	C
1048571	5150487	-26	C
1048572	5150487	-27	C
1048573	5150487	-28	C
1048574	5150487	-29	C

1048575 rows × 3 columns

```
df.describe()
```

	ID	MONTHS_BALANCE
count	1.048575e+06	1.048575e+06
mean	5.068286e+06	-1.913700e+01
std	4.615058e+04	1.402350e+01
min	5.001711e+06	-6.000000e+01
25%	5.023644e+06	-2.900000e+01
50%	5.062104e+06	-1.700000e+01
75%	5.113856e+06	-7.000000e+00
max	5.150487e+06	0.000000e+00

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1048575 entries, 0 to 1048574
Data columns (total 3 columns):
 #   Column          Non-Null Count    Dtype
---  ------          --------------    -----
 0   ID              1048575 non-null  int64
 1   MONTHS_BALANCE  1048575 non-null  int64
 2   STATUS          1048575 non-null  object
dtypes: int64(2), object(1)
memory usage: 24.0+ MB
```

```
df.isnull().count()
```

```
ID                1048575
MONTHS_BALANCE    1048575
STATUS            1048575
dtype: int64
```

```
df = df.dropna()
```

```
df.isnull().sum()
```

```
ID                0
MONTHS_BALANCE    0
STATUS            0
dtype: int64
```

```
data = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/application_record.csv')
```

```
data
```

	ID	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	NAME_INCOME_TYPE	NAME_EDUCATION_TYPE	NAME
0	5008804	M	Y	Y	0	427500.0	Working	Higher education	
1	5008805	M	Y	Y	0	427500.0	Working	Higher education	
2	5008806	M	Y	Y	0	112500.0	Working	Secondary / secondary special	
3	5008808	F	N	Y	0	270000.0	Commercial associate	Secondary / secondary special	Si
4	5008809	F	N	Y	0	270000.0	Commercial associate	Secondary / secondary special	Si
...	...	...	...	...	...	...	...	...	
438552	6840104	M	N	Y	0	135000.0	Pensioner	Secondary / secondary special	
438553	6840222	F	N	N	0	103500.0	Working	Secondary / secondary special	Si
438554	6841878	F	N	N	0	54000.0	Commercial associate	Higher education	Si
438555	6842765	F	N	Y	0	72000.0	Pensioner	Secondary / secondary special	
438556	6842885	F	N	Y	0	121500.0	Working	Secondary / secondary special	

438557 rows × 18 columns

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 438557 entries, 0 to 438556
Data columns (total 18 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   ID                   438557 non-null  int64
 1   CODE_GENDER          438557 non-null  object
 2   FLAG_OWN_CAR         438557 non-null  object
 3   FLAG_OWN_REALTY      438557 non-null  object
 4   CNT_CHILDREN         438557 non-null  int64
 5   AMT_INCOME_TOTAL     438557 non-null  float64
 6   NAME_INCOME_TYPE     438557 non-null  object
 7   NAME_EDUCATION_TYPE  438557 non-null  object
 8   NAME_FAMILY_STATUS   438557 non-null  object
 9   NAME_HOUSING_TYPE    438557 non-null  object
 10  DAYS_BIRTH           438557 non-null  int64
 11  DAYS_EMPLOYED        438557 non-null  int64
 12  FLAG_MOBIL           438557 non-null  int64
 13  FLAG_WORK_PHONE      438557 non-null  int64
 14  FLAG_PHONE           438557 non-null  int64
 15  FLAG_EMAIL           438557 non-null  int64
 16  OCCUPATION_TYPE      304354 non-null  object
 17  CNT_FAM_MEMBERS      438557 non-null  float64
```

```
dtypes: float64(2), int64(8), object(8)
memory usage: 60.2+ MB
```

```
data.isnull().sum()
```

```
ID                          0
CODE_GENDER                 0
FLAG_OWN_CAR                0
FLAG_OWN_REALTY             0
CNT_CHILDREN                0
AMT_INCOME_TOTAL            0
NAME_INCOME_TYPE            0
NAME_EDUCATION_TYPE         0
NAME_FAMILY_STATUS          0
NAME_HOUSING_TYPE           0
DAYS_BIRTH                  0
DAYS_EMPLOYED               0
FLAG_MOBIL                  0
FLAG_WORK_PHONE             0
FLAG_PHONE                  0
FLAG_EMAIL                  0
OCCUPATION_TYPE        134203
CNT_FAM_MEMBERS             0
dtype: int64
```

```
data.dropna()
```

	ID	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TC
2	5008806	M	Y	Y	0	1125
3	5008808	F	N	Y	0	2700
4	5008809	F	N	Y	0	2700
5	5008810	F	N	Y	0	2700
6	5008811	F	N	Y	0	2700
...	...	...	...	...	...	
438541	6837707	M	N	Y	0	2025
438548	6839936	M	Y	Y	1	1350
438553	6840222	F	N	N	0	1035
438554	6841878	F	N	N	0	540
438556	6842885	F	N	Y	0	1215

304354 rows × 18 columns

```
data.isnull().sum()
```

```
ID                          0
CODE_GENDER                 0
FLAG_OWN_CAR                0
FLAG_OWN_REALTY             0
CNT_CHILDREN                0
AMT_INCOME_TOTAL            0
NAME_INCOME_TYPE            0
NAME_EDUCATION_TYPE         0
NAME_FAMILY_STATUS          0
NAME_HOUSING_TYPE           0
DAYS_BIRTH                  0
DAYS_EMPLOYED               0
FLAG_MOBIL                  0
FLAG_WORK_PHONE             0
FLAG_PHONE                  0
```

```
FLAG_EMAIL                 0
OCCUPATION_TYPE       134203
CNT_FAM_MEMBERS            0
dtype: int64
```

```
data.isna().sum()
```

```
ID                     0
CODE_GENDER            0
FLAG_OWN_CAR           0
FLAG_OWN_REALTY        0
CNT_CHILDREN           0
AMT_INCOME_TOTAL       0
NAME_INCOME_TYPE       0
NAME_EDUCATION_TYPE    0
NAME_FAMILY_STATUS     0
NAME_HOUSING_TYPE      0
DAYS_BIRTH             0
DAYS_EMPLOYED          0
FLAG_MOBIL             0
FLAG_WORK_PHONE        0
FLAG_PHONE             0
FLAG_EMAIL             0
CNT_FAM_MEMBERS        0
dtype: int64
```

```
data.drop(['OCCUPATION_TYPE'], axis=1, inplace=True)
```

```
data.isnull().sum()
```

```
ID                     0
CODE_GENDER            0
FLAG_OWN_CAR           0
FLAG_OWN_REALTY        0
CNT_CHILDREN           0
AMT_INCOME_TOTAL       0
NAME_INCOME_TYPE       0
NAME_EDUCATION_TYPE    0
NAME_FAMILY_STATUS     0
NAME_HOUSING_TYPE      0
DAYS_BIRTH             0
DAYS_EMPLOYED          0
FLAG_MOBIL             0
FLAG_WORK_PHONE        0
FLAG_PHONE             0
FLAG_EMAIL             0
CNT_FAM_MEMBERS        0
dtype: int64
```

```
join = pd.merge(df,data)
```

```
join
```

	ID	MONTHS_BALANCE	STATUS	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT
0	5008804	0	C	M	Y	Y	
1	5008804	-1	C	M	Y	Y	

```
join1 = pd.merge(df,data, on = 'ID', how='inner')
```

```
join1
```

	ID	MONTHS_BALANCE	STATUS	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT
0	5008804	0	C	M	Y	Y	
1	5008804	-1	C	M	Y	Y	
2	5008804	-2	C	M	Y	Y	
3	5008804	-3	C	M	Y	Y	
4	5008804	-4	C	M	Y	Y	
...	...	...	...	...	...	...	
777710	5150487	-25	C	M	Y	N	
777711	5150487	-26	C	M	Y	N	
777712	5150487	-27	C	M	Y	N	
777713	5150487	-28	C	M	Y	N	
777714	5150487	-29	C	M	Y	N	

777715 rows × 19 columns

```
join1.drop(['MONTHS_BALANCE','STATUS'], axis=1, inplace=True)
```

```
join1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 777715 entries, 0 to 777714
Data columns (total 17 columns):
 #   Column               Non-Null Count   Dtype  
---  ------               --------------   -----  
 0   ID                   777715 non-null  int64  
 1   CODE_GENDER          777715 non-null  object 
 2   FLAG_OWN_CAR         777715 non-null  object 
 3   FLAG_OWN_REALTY      777715 non-null  object 
 4   CNT_CHILDREN         777715 non-null  int64  
 5   AMT_INCOME_TOTAL     777715 non-null  float64
 6   NAME_INCOME_TYPE     777715 non-null  object 
 7   NAME_EDUCATION_TYPE  777715 non-null  object 
 8   NAME_FAMILY_STATUS   777715 non-null  object 
 9   NAME_HOUSING_TYPE    777715 non-null  object 
 10  DAYS_BIRTH           777715 non-null  int64  
 11  DAYS_EMPLOYED        777715 non-null  int64  
 12  FLAG_MOBIL           777715 non-null  int64  
 13  FLAG_WORK_PHONE      777715 non-null  int64  
 14  FLAG_PHONE           777715 non-null  int64  
 15  FLAG_EMAIL           777715 non-null  int64  
 16  CNT_FAM_MEMBERS      777715 non-null  float64
dtypes: float64(2), int64(8), object(7)
memory usage: 106.8+ MB
```

```
##AMT_INCOME_TOTAL, NAME_EDUCATION_TYPE ,NAME_INCOME_TYPE  ,NAME_FAMILY_STATUS, NAME_HOUSING_TYPE
 ##student are considered as bad
###commercial associate and state_servent
```

```
join1['NAME_INCOME_TYPE'].unique()
```

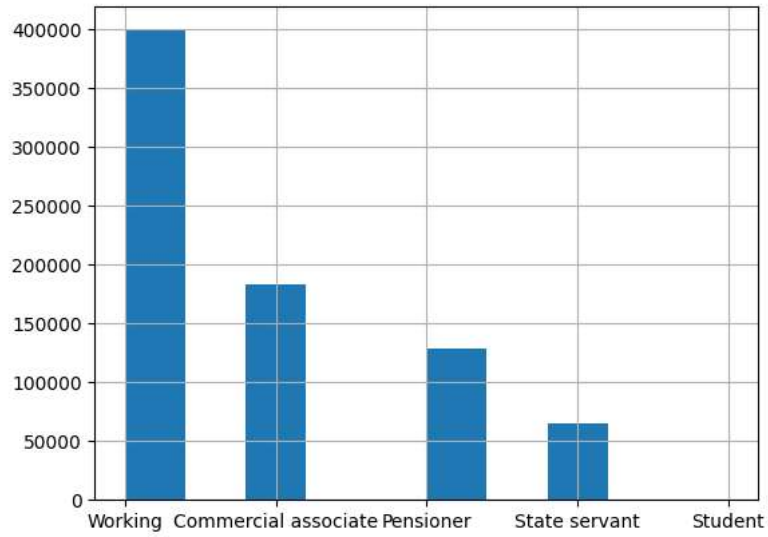
```
array(['Working', 'Commercial associate', 'Pensioner', 'State servant',
       'Student'], dtype=object)
```

```
join1.isnull().sum()
```

```
ID                     0
CODE_GENDER            0
FLAG_OWN_CAR           0
FLAG_OWN_REALTY        0
CNT_CHILDREN           0
AMT_INCOME_TOTAL       0
NAME_INCOME_TYPE       0
NAME_EDUCATION_TYPE    0
NAME_FAMILY_STATUS     0
NAME_HOUSING_TYPE      0
DAYS_BIRTH             0
DAYS_EMPLOYED          0
FLAG_MOBIL             0
FLAG_WORK_PHONE        0
FLAG_PHONE             0
FLAG_EMAIL             0
CNT_FAM_MEMBERS        0
dtype: int64
```

```
join1['NAME_INCOME_TYPE'].hist()
```

```
<Axes: >
```




```
from sklearn.preprocessing import LabelEncoder
features = ['AMT_INCOME_TOTAL','NAME_EDUCATION_TYPE' ,'NAME_INCOME_TYPE'  ,'NAME_FAMILY_STATUS', 'NAME_HOUSING_TYPE']

le = LabelEncoder()

for col in features:
    join1[col] = le.fit_transform(join1[col])

join1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 777715 entries, 0 to 777714
Data columns (total 17 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   ID                   777715 non-null  int64
 1   CODE_GENDER          777715 non-null  object
 2   FLAG_OWN_CAR         777715 non-null  object
 3   FLAG_OWN_REALTY      777715 non-null  object
 4   CNT_CHILDREN         777715 non-null  int64
 5   AMT_INCOME_TOTAL     777715 non-null  int64
 6   NAME_INCOME_TYPE     777715 non-null  int64
 7   NAME_EDUCATION_TYPE  777715 non-null  int64
 8   NAME_FAMILY_STATUS   777715 non-null  int64
 9   NAME_HOUSING_TYPE    777715 non-null  int64
 10  DAYS_BIRTH           777715 non-null  int64
 11  DAYS_EMPLOYED        777715 non-null  int64
 12  FLAG_MOBIL           777715 non-null  int64
```

```
 13  FLAG_WORK_PHONE      777715 non-null  int64
 14  FLAG_PHONE           777715 non-null  int64
 15  FLAG_EMAIL           777715 non-null  int64
 16  CNT_FAM_MEMBERS      777715 non-null  float64
dtypes: float64(1), int64(13), object(3)
memory usage: 106.8+ MB
```

```
join1['Applicant_type'] = ""
```

```
join1.head()
```

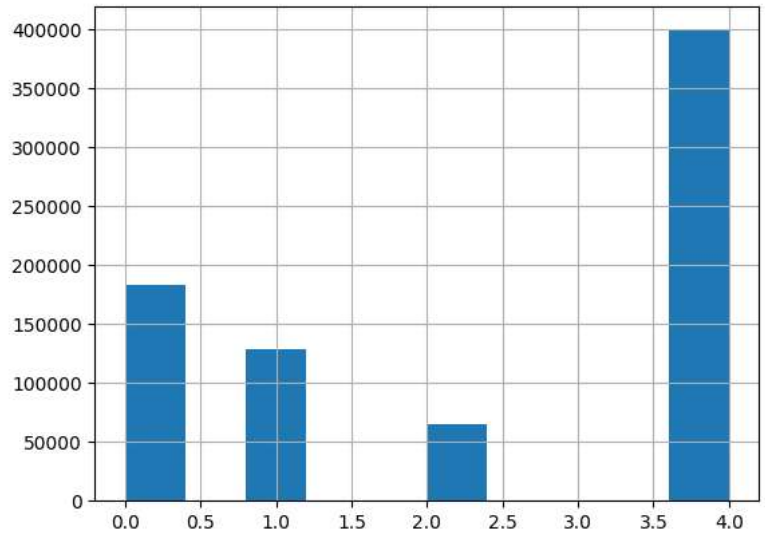
	ID	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL
0	5008804	M	Y	Y	0	226
1	5008804	M	Y	Y	0	226
2	5008804	M	Y	Y	0	226
3	5008804	M	Y	Y	0	226
4	5008804	M	Y	Y	0	226

```
join1['NAME_INCOME_TYPE'].unique()
```

```
array([4, 0, 1, 2, 3])
```

```
join1['NAME_INCOME_TYPE'].hist()
```

```
<Axes: >
```



class 0 is for commercial associate

class 1 for pensioner

class 2 for state_servant

class 3 for students

class 4 for working

```
join1['Applicant_type'] = join1['NAME_INCOME_TYPE'].apply(lambda x: 'Bad' if x == 3 else 'Good')
```

```
join1.head()
```

	ID	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL
0	5008804	M	Y	Y	0	226
1	5008804	M	Y	Y	0	226
2	5008804	M	Y	Y	0	226
3	5008804	M	Y	Y	0	226
4	5008804	M	Y	Y	0	226

```
join1['Applicant_type'] = le.fit_transform(join1['Applicant_type'])
```

```
join1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 777715 entries, 0 to 777714
Data columns (total 18 columns):
 #   Column               Non-Null Count   Dtype  
---  ------               --------------   -----  
 0   ID                   777715 non-null  int64  
 1   CODE_GENDER          777715 non-null  object 
 2   FLAG_OWN_CAR         777715 non-null  object 
 3   FLAG_OWN_REALTY      777715 non-null  object 
 4   CNT_CHILDREN         777715 non-null  int64  
 5   AMT_INCOME_TOTAL     777715 non-null  int64  
 6   NAME_INCOME_TYPE     777715 non-null  int64  
 7   NAME_EDUCATION_TYPE  777715 non-null  int64  
 8   NAME_FAMILY_STATUS   777715 non-null  int64  
 9   NAME_HOUSING_TYPE    777715 non-null  int64  
 10  DAYS_BIRTH           777715 non-null  int64  
 11  DAYS_EMPLOYED        777715 non-null  int64  
 12  FLAG_MOBIL           777715 non-null  int64  
 13  FLAG_WORK_PHONE      777715 non-null  int64  
 14  FLAG_PHONE           777715 non-null  int64  
 15  FLAG_EMAIL           777715 non-null  int64  
 16  CNT_FAM_MEMBERS      777715 non-null  float64
 17  Applicant_type       777715 non-null  int64  
dtypes: float64(1), int64(14), object(3)
memory usage: 112.7+ MB
```

```
temp = join1.drop(columns = ['DAYS_BIRTH', 'FLAG_EMAIL', 'FLAG_WORK_PHONE', 'FLAG_MOBIL', 'CNT_CHILDREN'])
```

```
temp.duplicated().sum()
```

```
741258
```

```
Y = ['Applicant_type']
X = join1.drop(['Applicant_type'], axis=1)
```

```
X['CODE_GENDER'] = le.fit_transform(X['CODE_GENDER'])
X['FLAG_OWN_CAR'] = le.fit_transform(X['FLAG_OWN_CAR'])
X['FLAG_OWN_REALTY'] = le.fit_transform(X['FLAG_OWN_REALTY'])
```

```
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 777715 entries, 0 to 777714
Data columns (total 18 columns):
 #   Column               Non-Null Count   Dtype  
---  ------               --------------   -----  
 0   ID                   777715 non-null  int64  
 1   CODE_GENDER          777715 non-null  int64  
 2   FLAG_OWN_CAR         777715 non-null  int64  
 3   FLAG_OWN_REALTY      777715 non-null  int64  
 4   CNT_CHILDREN         777715 non-null  int64  
 5   AMT_INCOME_TOTAL     777715 non-null  int64  
 6   NAME_INCOME_TYPE     777715 non-null  int64  
 7   NAME_EDUCATION_TYPE  777715 non-null  int64  
 8   NAME_FAMILY_STATUS   777715 non-null  int64  
 9   NAME_HOUSING_TYPE    777715 non-null  int64  
```

```
 10  DAYS_BIRTH           777715 non-null  int64
 11  DAYS_EMPLOYED        777715 non-null  int64
 12  FLAG_MOBIL           777715 non-null  int64
 13  FLAG_WORK_PHONE      777715 non-null  int64
 14  FLAG_PHONE           777715 non-null  int64
 15  FLAG_EMAIL           777715 non-null  int64
 16  CNT_FAM_MEMBERS      777715 non-null  float64
 17   FLAG_OWN_CAR        777715 non-null  int64
dtypes: float64(1), int64(17)
memory usage: 112.7 MB
```

```
X.value_counts()
```

```
ID       CODE_GENDER  FLAG_OWN_CAR  FLAG_OWN_REALTY  CNT_CHILDREN  AMT_INCOME_TOTAL  NAME_INCOME_TYPE  NAME_EDUCATION_TYPE
NAME_FAMILY_STATUS  NAME_HOUSING_TYPE  DAYS_BIRTH  DAYS_EMPLOYED  FLAG_MOBIL  FLAG_WORK_PHONE  FLAG_PHONE  FLAG_EMAIL  CNT_FAM_MEMBERS
FLAG_OWN_CAR
5148819  0            1             1                0             104               4                 1                   0
1                   -19841      -4428          1           1                1           0           2.0             1                61
5115964  1            1             1                2             104               4                 4                   1
1                   -14677      -3938          1           1                0           0           4.0             1                61
5061741  0            0             1                0             104               1                 4                   3
1                   -23929       365243        1           0                1           0           1.0             0                61
5078799  0            0             1                0             193               4                 1                   0
1                   -19808      -390           1           1                0           0           2.0             0                61
5061685  1            0             0                0             192               4                 4                   1
1                   -11822      -4246          1           0                0           0           2.0             0                61

..
5139553  0            1             1                2             120               0                 1                   1
1                   -13584      -6337          1           0                1           0           4.0             1                 1
5069020  0            0             1                0             158               4                 4                   1
1                   -20295      -3700          1           0                0           0           2.0             0                 1
5097025  1            1             0                0             157               0                 1                   1
1                   -13643      -2956          1           0                0           0           2.0             1                 1
5023604  0            0             1                0             104               4                 4                   1
1                   -20323      -1727          1           0                0           0           2.0             0                 1
5092141  1            0             0                0             89                4                 1                   3
1                   -11162      -1327          1           1                1           0           1.0             0                 1
Length: 36457, dtype: int64
```