*Group 2: Fabiana Caccavale, Matteo Gioia, Martina Manno, Vincenzo J. Striano, Marco Vita Antonio, Marco Pisano **

Ethics for AI project: Potential biases in AI-based financial services

Libraries with fixed versions and data loading

```
In [ ]: !conda install --yes python==3.7.10
        !conda install --yes xgboost==1.6.2
        !conda install --yes shap==0.41.0
        !conda install --yes pandas==1.3.5
        !conda install --yes plotly==5.10.0
        !conda install --yes scikit-learn==1.0.2
        !conda install --yes matplotlib
        !conda install --yes seaborn
        !conda install --yes dython
```

```
In [1]: import xgboost as xgb
        import pandas as pd
        import numpy as np
        import shap
        import plotly.express as px
        import matplotlib.pyplot as plt
        import seaborn as sns
        import dython
        from sklearn.model_selection import train_test_split
        from sklearn import metrics
        from sklearn.preprocessing import OneHotEncoder
        from dython.nominal import associations
        from plotly.offline import iplot, init_notebook_mode
        init_notebook_mode(connected = True)
        from IPython.display import Image
```

The project (https://www.kaggle.com/c/home-credit-default-risk) is inspired by a kaggle competition focused on predicting the future payment behaviour of clients from a loan application.

The dataset considered is the "application train" and from this, 26 columns (1 target, 1 ID and 24 features) were selected out of a total of 122 in order to have a more interpretable overview.

```
In [2]: ap_train = pd.read_csv('application_train.csv',
                               usecols =  ['TARGET', 'SK_ID_CURR', 'NAME_CONTRACT_TYPE', 'CODE_GENDER', 'FLAG_OWN_CAR',
                                           'FLAG_OWN_REALTY', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL', 'AMT_CREDIT', 'AMT_ANNUITY',
                                           'AMT_GOODS_PRICE', 'NAME_TYPE_SUITE', 'NAME_INCOME_TYPE', 'NAME_EDUCATION_TYPE', 'OCCUPATION_TYPE',
                                           'NAME_FAMILY_STATUS', 'NAME_HOUSING_TYPE', 'REGION_POPULATION_RELATIVE', 'DAYS_BIRTH',
                                           'DAYS_EMPLOYED', 'DAYS_REGISTRATION', 'DAYS_ID_PUBLISH', 'OWN_CAR_AGE', 'EXT_SOURCE_1',
                                           'EXT_SOURCE_2', 'EXT_SOURCE_3'])
        print(f"Application train dataset shape: {ap_train.shape}")

        Application train dataset shape: (307511, 26)
```

```
In [3]: # DAYS_BIRTH is transformed in AGE
        ap_train['AGE'] = -round(ap_train['DAYS_BIRTH']/365)
        ap_train.drop(columns='DAYS_BIRTH', inplace=True)
```

[#1] Interesting variables

The first step is to reflect on which variables might be prone to an ethical discussion.

In anti-discrimination philosophy, Cass Sunstein writes, *“without good reason, social and legal structures should not turn differences that are both highly visible and irrelevant from the moral point of view into systematic social disadvantages”* (Sunstein 1994).

Algorithmic-related bias refers to systematic and repeatable errors in a mathematical or computer system that leads to ‘unfair’ outputs, privileging one or more groups over others.

The variables that can lead to unjust judgment due to: historical bias representation bias measurement bias have been labelled as ethical.

1. **CODE_GENDER**: results in ethical relevance because an individual's gender should not be a useful variable in the choice of granting a mortgage (Men vs. Women, Men vs. Transgender)
2. **FLAG_OWN_CAR**: results in ethical relevance because the ownership of a car does not affect the choice of granting a mortgage. Owning or not owning a car does not turn out to be related to a customer's creditworthiness. It is more than plausible, that individuals living in a large city prefer to use public or sustainable transportation.
3. **FLAG_OWN_REALTY**: homeownership does not affect the choice of granting a mortgage. Owning or not owning a car does not appear to be related to a client's creditworthiness. Although home ownership is commonly associated with large economic capacity, statistics show that in countries with higher GDP, the percentage of individuals owning a home for residential use decreases. For example, Romania does not rank as a high-income economy within the European Union, has the highest homeownership rates in the EU, and has the most crowded housing. Romania has 96.4 percent of its population owning homes.
4. **CNT_CHILDREN**: this turns out to be of ethical relevance because the number of children does not affect the choice to grant a mortgage. There is a tendency to think that a large number of children results in large expenses that decrease the ability to repay the debt. First, relying only on the number of offspring does not indicate how many more of the children are dependent on the individual; they may already be financially independent. Moreover, it is incorrect not to take into consideration that a possible partner might contribute to the expenses of supporting the offspring. Therefore, we believe that other variables should also be taken into consideration.
5. **NAME_TYPE_SUITE**: who accompanies the borrower should not be considered in the granting of a mortgage. The one who should be solvent is the one who makes the commitment. The judgment cannot be influenced by the accompanying person under any circumstances. The judgment must be based on the borrower.
6. **NAME_INCOME_TYPE**: This variable is prone to ethical discourse in part. It is not fair to make a judgment about solvency between an individual state servant and a commercial associate or pensioner for example. In both cases, the value should not bias the judgment and thus is of ethical relevance. However, in this variable, we also find the unemployment factor, which in our view cannot be considered ethically relevant. The fact that an individual is unemployed is a strong factor that should condition the individual's judgment of insolvency. The fact that an individual does not have a fixed salary and therefore economic insecurity in theory should be a very relevant factor for the bank.
7. **NAME_EDUCATION_TYPE**: turns out to be ethically relevant because the level of education should not be taken into consideration in the granting of a mortgage (Although a high level of education may indicate possible higher earnings, it still does not ensure economic stability or a lower level of education does not undermine an individual's creditworthiness).
8. **OCCUPATION_TYPE**: The type of occupation is not an accurate index of income. Even in the same category of occupation, there can be a huge income gap. A chef in a starred restaurant will not have the same income as a chef in a chain restaurant.
9. **NAME_FAMILY_STATUS**: this turns out to be ethically relevant because family status should not be taken into consideration when granting a mortgage. The number of individuals in the family does not ensure greater economic availability. Two married individuals might still have less aggregate income than a single individual or a widow.
10. **NAME_HOUSING_TYPE**: The type of property should not be taken into consideration when granting a mortgage. Homeownership does not indicate greater economic availability and vice versa. Young people might be discriminated against because, especially in Italy, it is not common to own a home. Those who travel a lot and therefore have economic availability may not have home ownership.
11. **REGION_POPULATION_RELATIVE**: We consider it of ethical relevance as well as absolute irrelevance toward judgment since a highly populated region is not an indication of wealth or poverty (Rio de Janeiro vs. New York City). Judging also on the basis of the regional population may be unnecessary as well as incorrect.
12. **DAYS_BIRTH**: this variable discriminates against young people because they are considered less creditworthy or the elderly because they are subject to lower life expectancy.
13. **DAYS_EMPLOYED**: it is not necessarily the case that someone who is a new employee has a lower salary, and therefore potentially less creditworthy, or that someone who has been employed for a long time has more disposable income.
14. **DAYS_ID_PUBLISH**: this variable is more important for a discourse of verifying the identity of the individual and thus in the validity of the loan issuance rather than the solvency of the individual.
15. **OWN_CAR_AGE**: The age of a car cannot be an indication of the individual's solvency. A dated car might be worth more than a new car, and an individual might keep a car as a matter of affection, and this is not an indication of the individual's poverty or wealth.

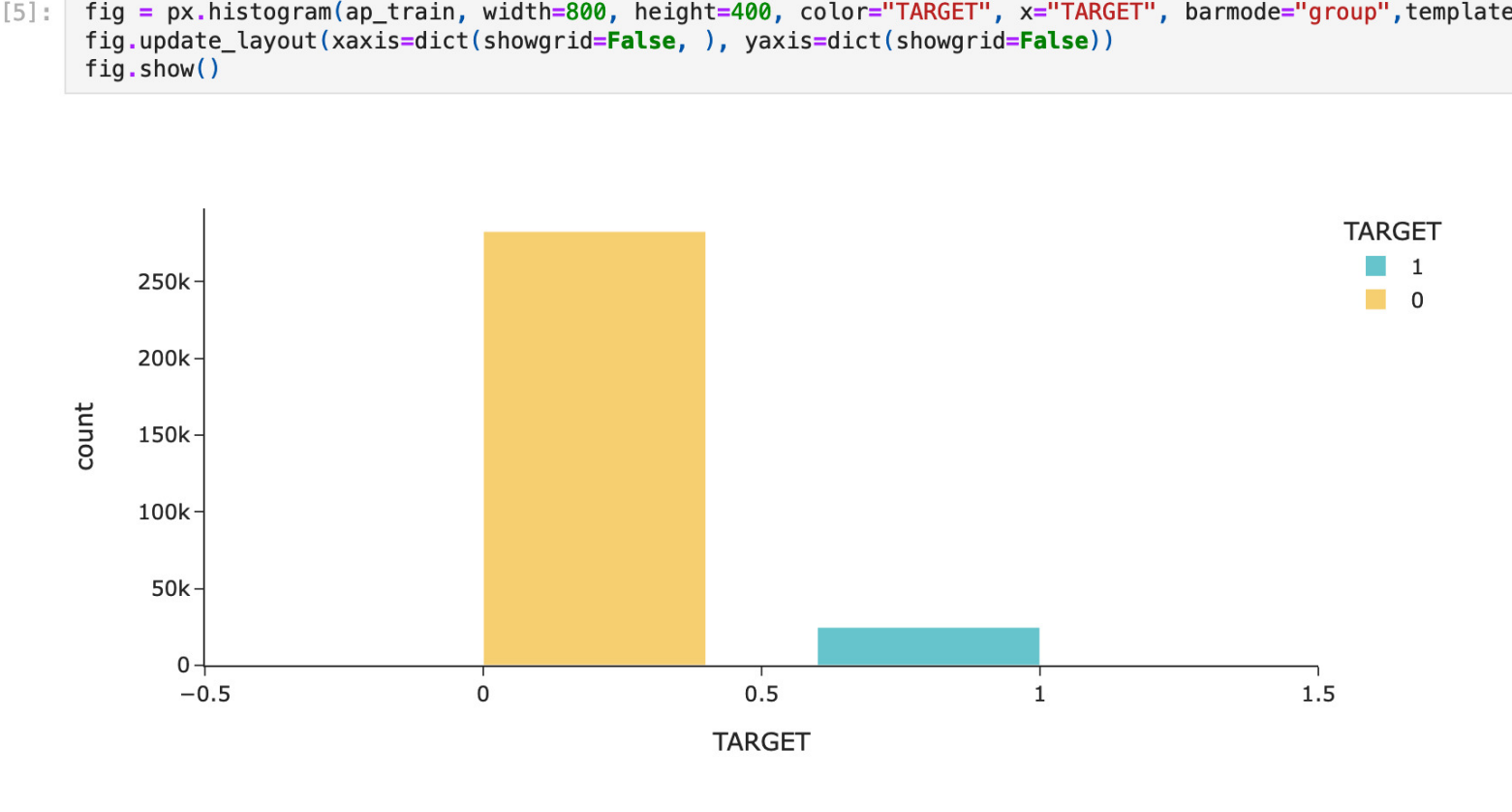
[#2] Interesting variables with respect to target

The second step concerns the understanding of the ethical variables highlited with respect to the target variable.

TARGET

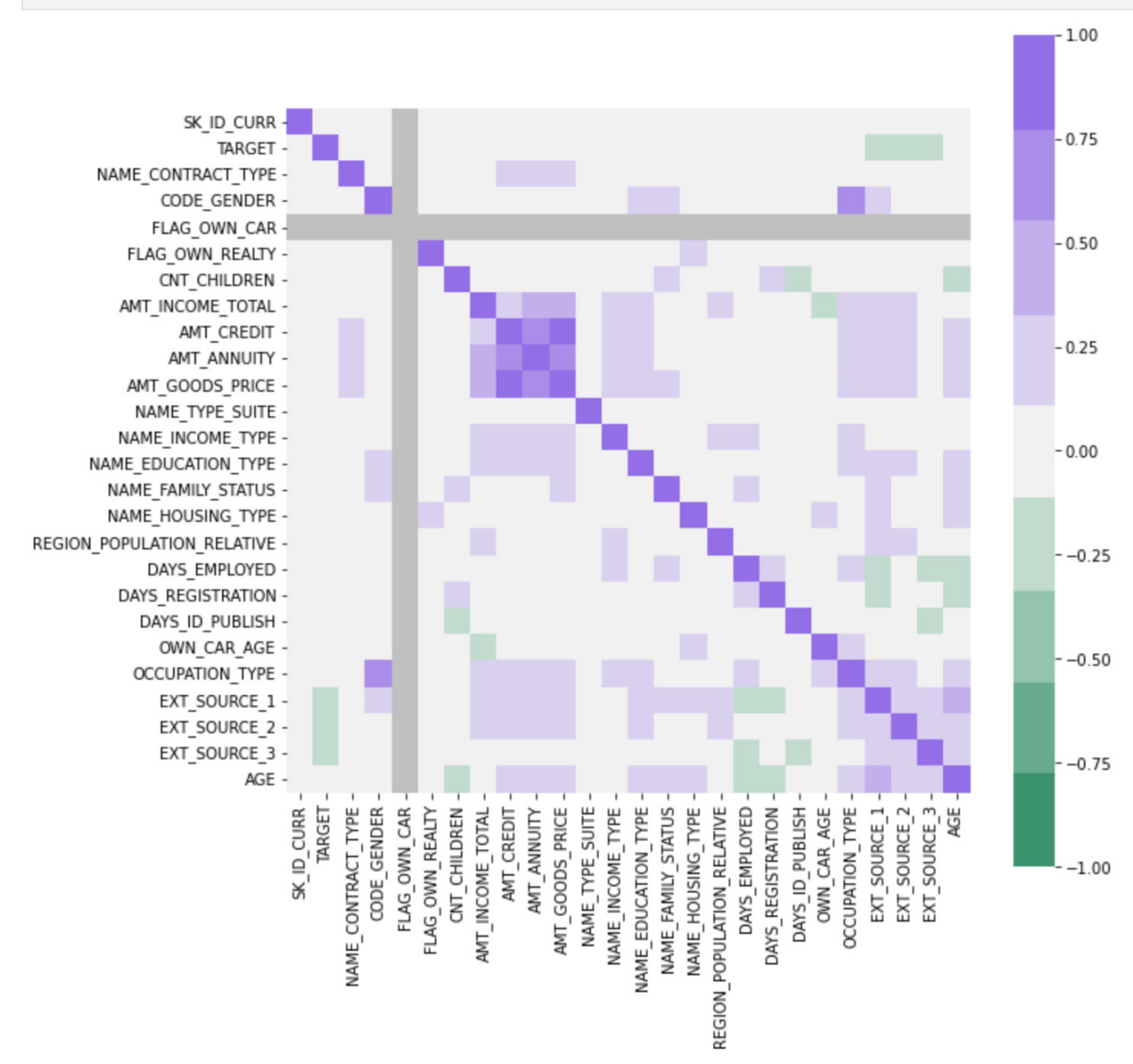
```
In [4]: ap_train.groupby('TARGET').apply(len)

Out[4]: TARGET
        0    282686
        1     24825
        dtype: int64
```

```
In [5]: fig = px.histogram(ap_train, width=800, height=400, color="TARGET", x="TARGET", barmode="group",template="simple_white", color_discrete_sequence
        fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
        fig.show()
```



The target is a binary variable with no missing values. It can assume the value 1 whether the client has payment difficulties (such as late payment of more than X days on at least one of the first Y instalments of the loan); or 0 in all other cases. The overall number of clients is about 307.511 and the graph above shows that only 24.825 clients (8% of the total) have been categorized among those who have payment difficulties.

```
In [6]: complete_correlation= associations(ap_train, figsize=(10,10), nan_strategy = 'drop_samples', vmin=-1, vmax=1,
                                           cmap=sns.diverging_palette(150, 275, s=80, l=55, n=9),  annot= False)
```

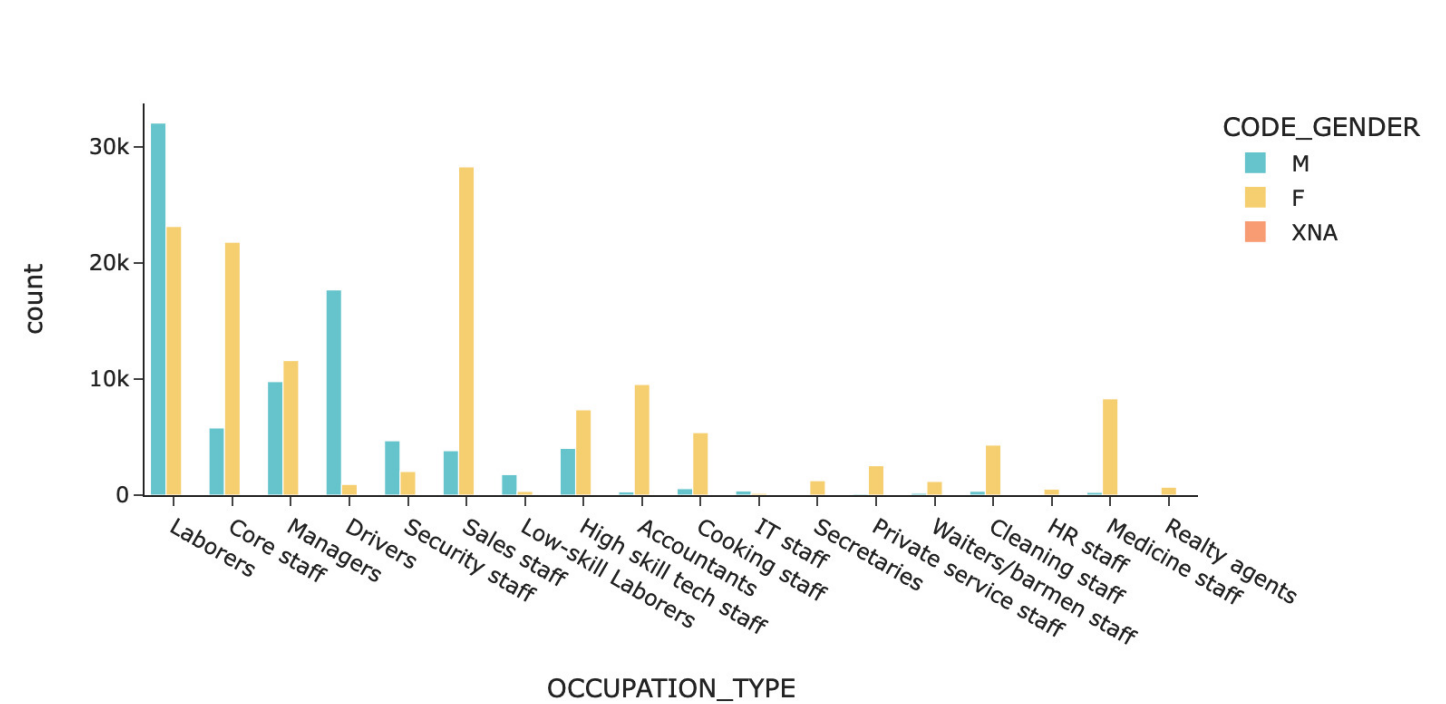


The target variable appears to be not correlated with any of the other dataset's variables. There is only a slight negative correlation with the variables EXT_SOURCE_1, EXT_SOURCE_2, and EXT_SOURCE_3 (at which the correlation is respectively -0.14, -0.14, and -0.17). A possible interpretation is that clients who have a low external source score would be inclined to be in the target number 1, then in the group of clients with some payment difficulties. However, since it is not known whether the normalized score from the external data source, to which these variables refer, has an ethical nature or not, it is not possible to determine whether these variables have ethical relevance.

Despite the fact that the financial institution has requested a lot of data, many of which are extremely sensitive, the architecture of the model allows it to determine for itself which of the variables are relevant or not. According to the model, the relevant ones are unethical and mostly come from third-party sources, over which we have no knowledge and the bank no control.

Another consideration can be done in terms of one of the variables that we have considered most prone to the ethical discussion: this is the code gender. This variable seems to be correlated with another ethical variable which is occupation type.

```
In [7]: fig = px.histogram(ap_train,width=800, height=400, color="CODE_GENDER", x="OCCUPATION_TYPE", barmode="group",template="simple_white", color_disc
        fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
        fig.show()
```

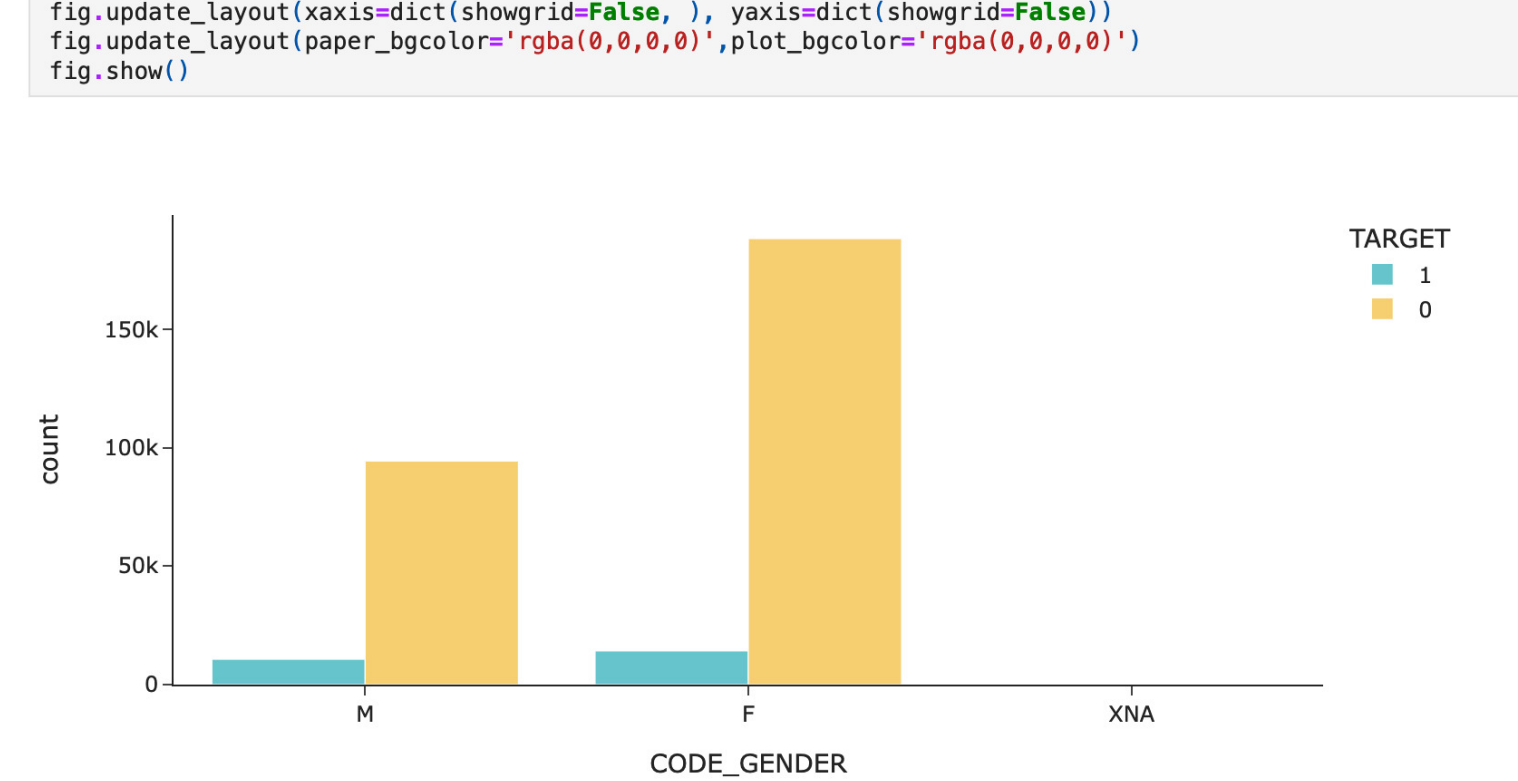


As shown in the graph, there are some occupation types, such as drivers, in which 95% is male, and on the other hand some occupations, such as the medicine staff, in which 97% is female.

CODE_GENDER

```
In [8]: ap_train.groupby('CODE_GENDER').apply(len)

Out[8]: CODE_GENDER
        F      202448
        M      105059
        XNA         4
        dtype: int64
```

```
In [9]: fig = px.histogram(ap_train,width=800, height=400, color="TARGET", x="CODE_GENDER", barmode="group",template="simple_white", color_discrete_sequ
        fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
        fig.update_layout(paper_bgcolor='rgba(0,0,0,0)',plot_bgcolor='rgba(0,0,0,0)')
        fig.show()
```



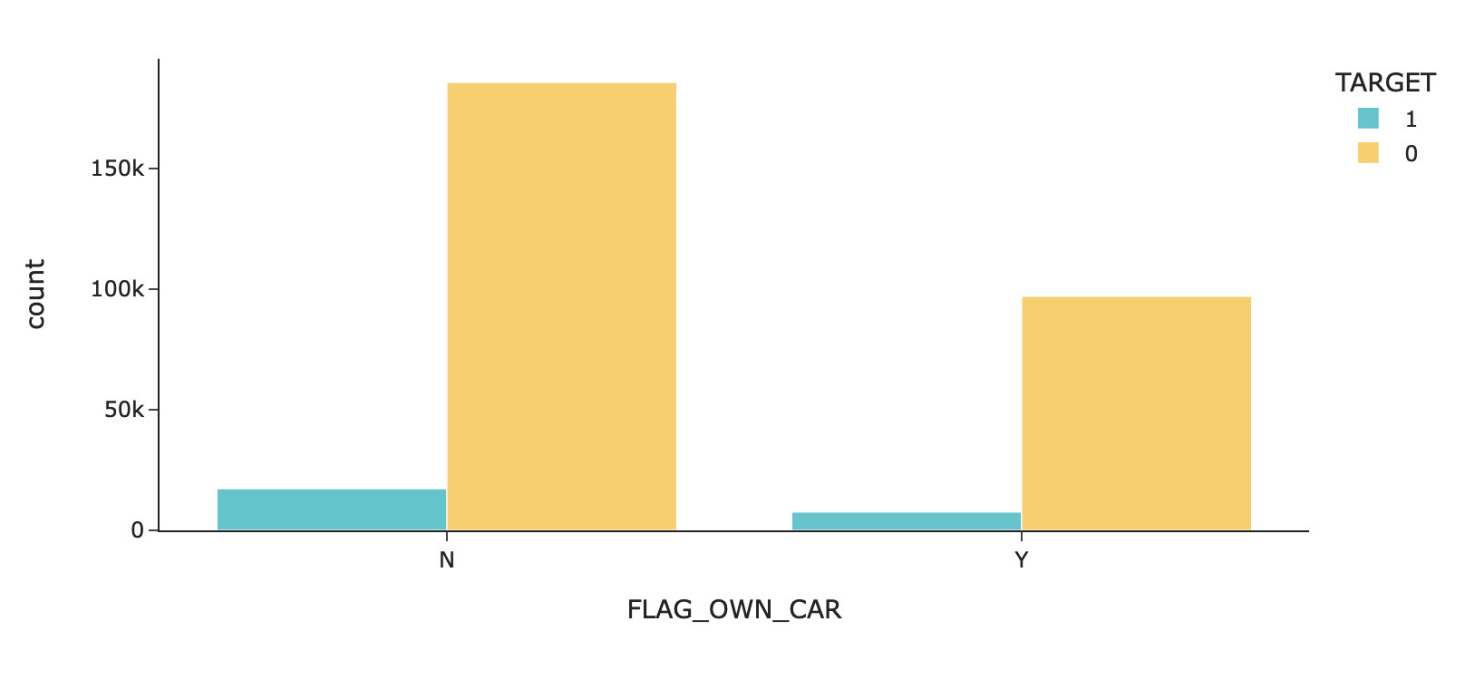
The variable indicates the gender of the client applying for the loan. The sample is composed of 65% women. Out of the total number of clients, there are 4 who preferred not to declare their gender, this could be due to a random reason or because it might be perceived by the client as sensible data. Moreover, this variable appears to be not relevant in the payment behaviours of clients because it cannot in any way be generalized that women or men are more or less solvent. The variable allows for several reflections:

- What would this same graph have looked like 50 years ago?
- Would the percentage of women in the total have been as high?
- Would the percentage of women with payment difficulties have been higher?
- If so, could these steps forward be due to measures to reduce the gender pay gap?

FLAG_OWN_CAR

```
In [10]: ap_train.groupby('FLAG_OWN_CAR').apply(len)

Out[10]: FLAG_OWN_CAR
         N    202924
         Y    104587
         dtype: int64
```

```
In [11]: fig = px.histogram(ap_train, width=800, height=400,color="TARGET", x="FLAG_OWN_CAR", barmode="group", template="simple_white", color_discrete_se
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```

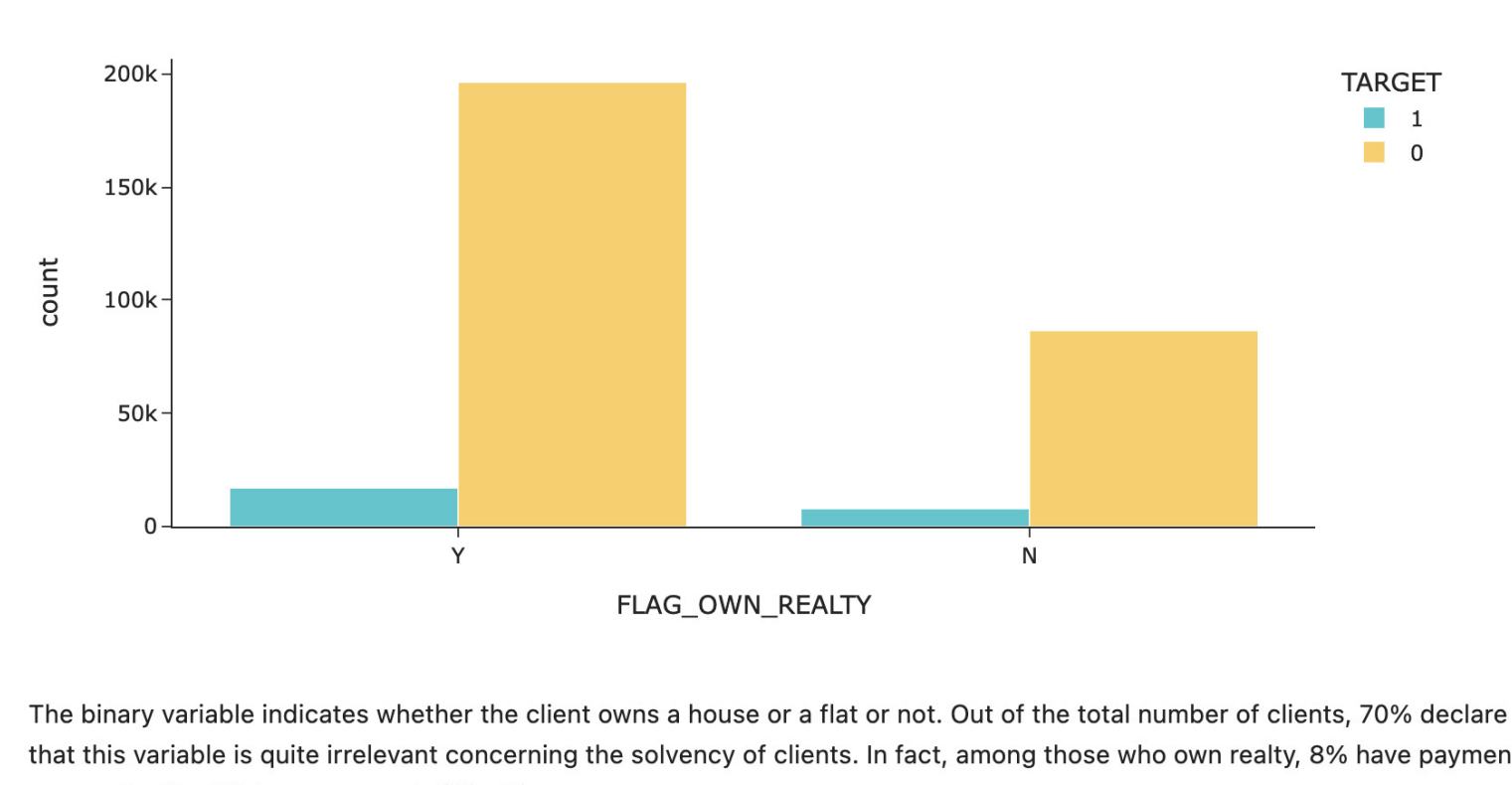


The binary variable indicates whether the client owns a car or not. Out of the total number of clients, 65% declare to not own a car. This variable is not relevant with regard to the solvency of clients because, as shown in the graph, among those who do not own a car, 8% have payment difficulties but also among those who own a car, the 7% have solvency problems.

FLAG_OWN_REALTY

```
In [12]: ap_train.groupby('FLAG_OWN_REALTY').apply(len)

Out[12]: FLAG_OWN_REALTY
         N     94199
         Y    213312
         dtype: int64
```

```
In [13]: fig = px.histogram(ap_train, width=800, height=400,color="TARGET", x="FLAG_OWN_REALTY", barmode="group", template="simple_white", color_discrete_
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```



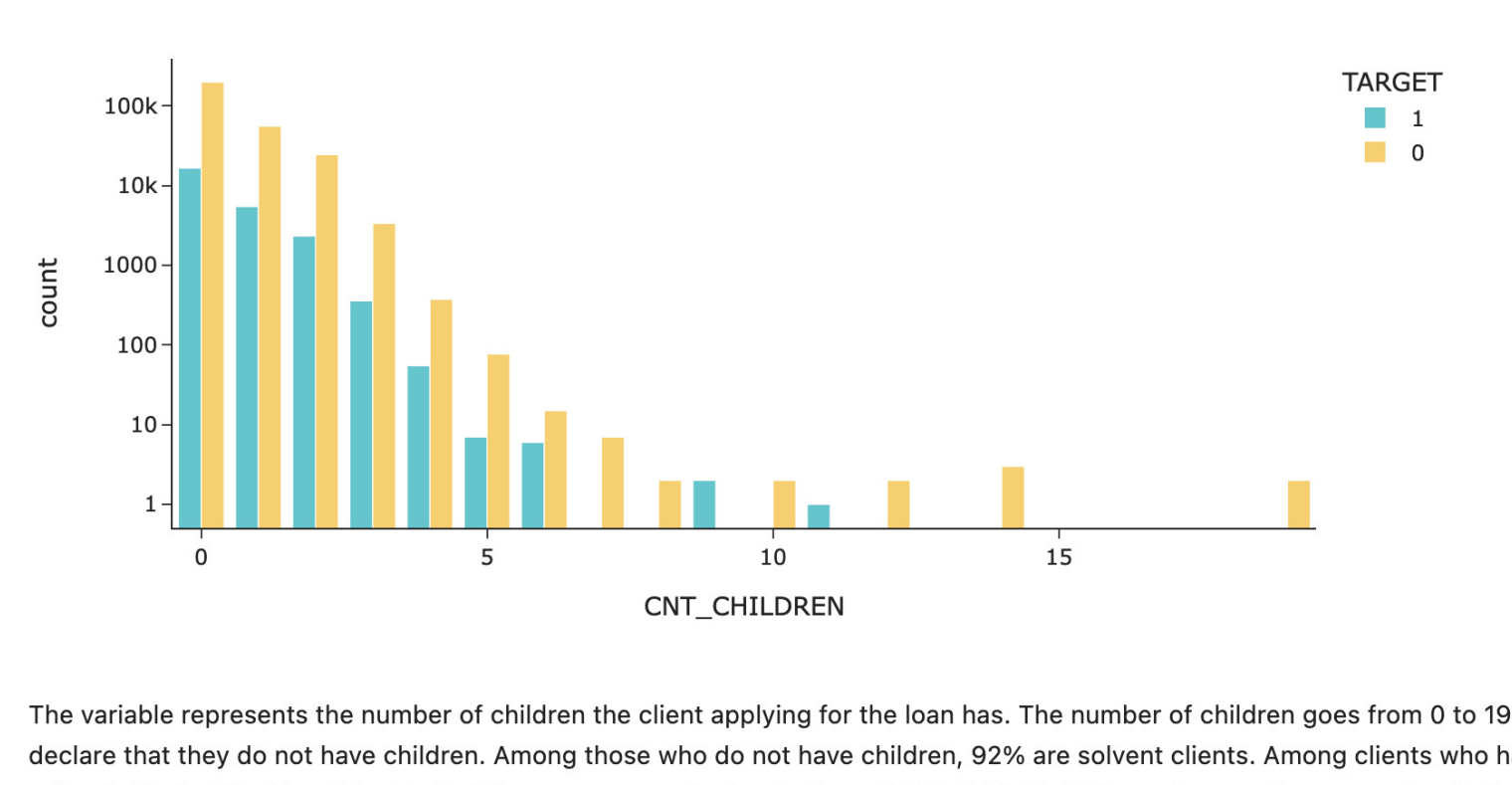
The binary variable indicates whether the client owns a house or a flat or not. Out of the total number of clients, 70% declare to own a house/flat. The graph above shows that this variable is quite irrelevant concerning the solvency of clients. In fact, among those who own realty, 8% have payment difficulties but also among those who do not own realty, the 8% have payment difficulties.

CNT_CHILDREN

```
In [14]: ap_train.groupby('CNT_CHILDREN').apply(len)

Out[14]: CNT_CHILDREN
         0     215371
         1      61119
         2      26749
         3       3717
         4        429
         5         84
         6         21
         7          7
         8          2
         9          2
         10         2
         11         1
         12         2
         14         3
         19         2
         dtype: int64
```

```
In [15]: ap_train['CNT_CHILDREN'].describe()

Out[15]: count    307511.000000
         mean          0.417052
         std           0.722121
         min           0.000000
         25%           0.000000
         50%           0.000000
         75%           1.000000
         max          19.000000
         Name: CNT_CHILDREN, dtype: float64
```

```
In [16]: fig = px.histogram(ap_train, width=800, height=400,color="TARGET", x="CNT_CHILDREN", barmode="group", log_y = True, template="simple_white", col
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```



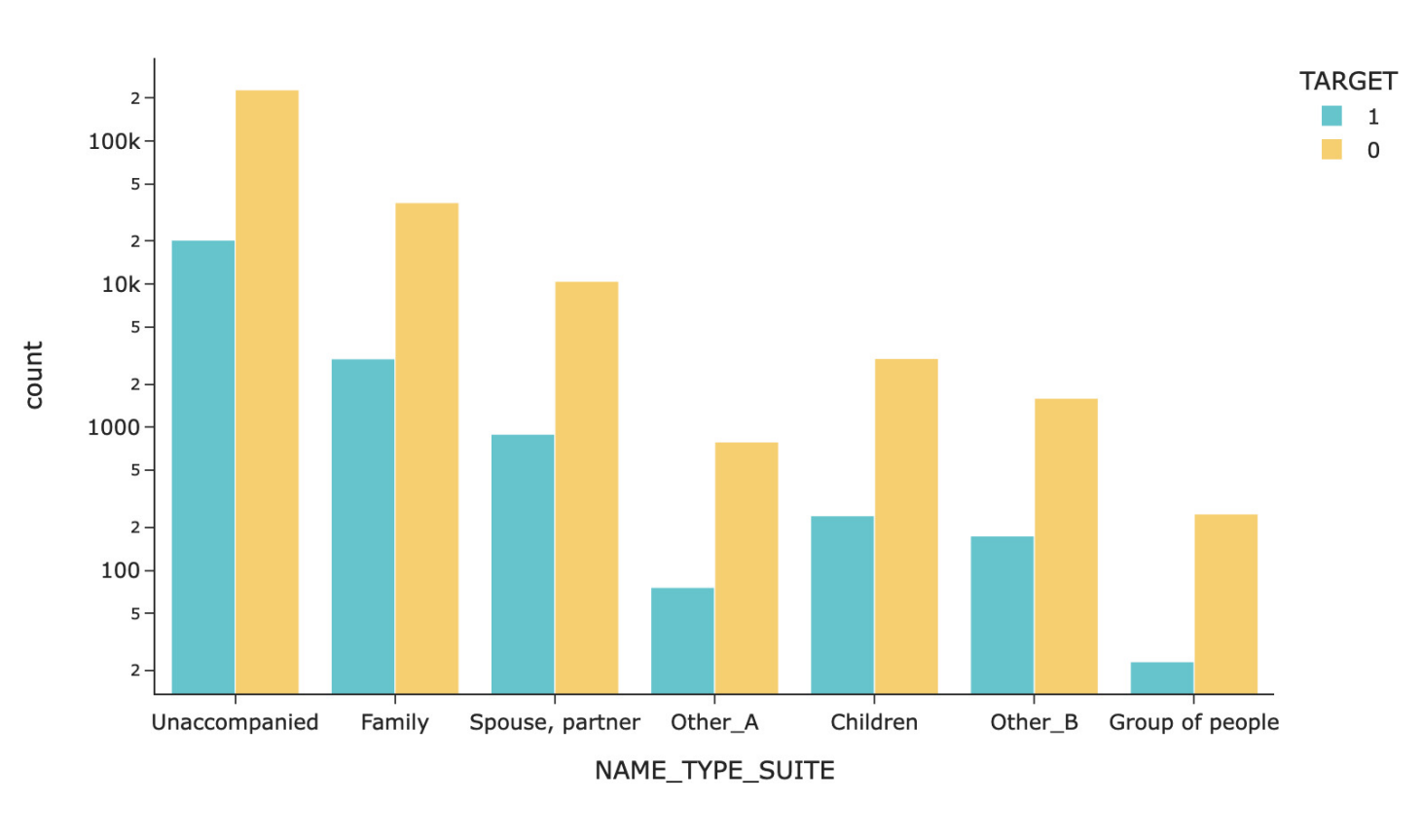
The variable represents the number of children the client applying for the loan has. The number of children goes from 0 to 19. There are 215.371 over 307.511 clients who declare that they do not have children. Among those who do not have children, 92% are solvent clients. Among clients who have from 1 to 6 children, at least, 70% are solvent clients. What is evident is that there are some clients who have 7,8,10,12,14,19 children who are all among solvent clients but take into consideration that they are only 18 over the total amount of clients. On the other hand, clients who have 9 and 11 children are all with payment difficulties but they are overall only 3 clients. It is possible to conclude that clients who have 0 or at least 1 child represent 75% of the overall amount of clients and respectively 92% and 91% of them are solvent.

NAME_TYPE_SUITE

```
In [17]: ap_train['NAME_TYPE_SUITE'].isnull().sum()

Out[17]: 1292
```

```
In [18]: ap_train.groupby('NAME_TYPE_SUITE').apply(len)

Out[18]: NAME_TYPE_SUITE
         Children             3267
         Family              40149
         Group of people       271
         Other_A               866
         Other_B              1770
         Spouse, partner     11370
         Unaccompanied      248526
         dtype: int64
```

```
In [19]: fig = px.histogram(ap_train,width=800, height=500, color="TARGET", x="NAME_TYPE_SUITE", barmode="group",log_y=True, template="simple_white", col
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```

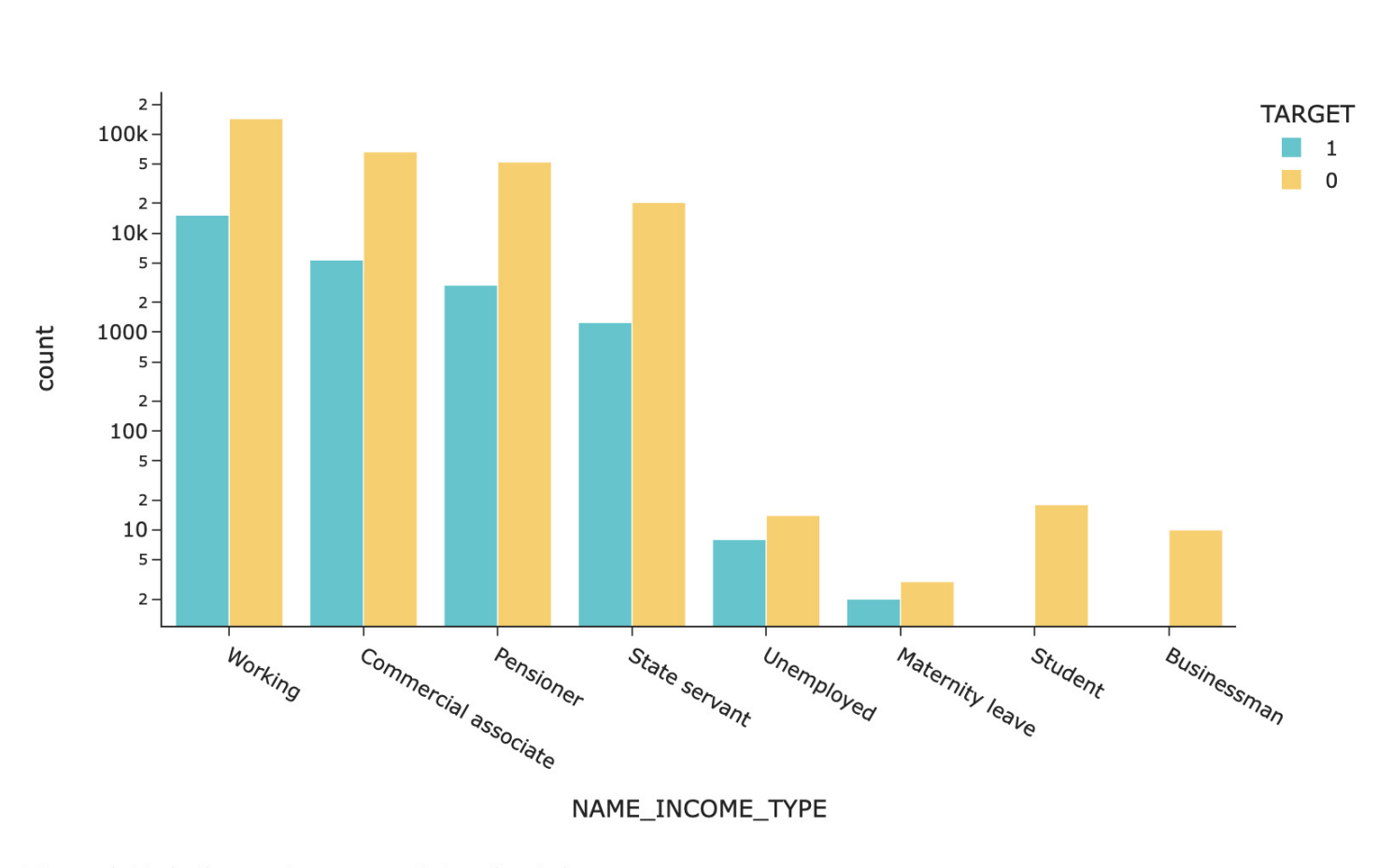


The variable indicates who was accompanying the client when applying for the loan. There are 1292 missing values which refers to clients who choose not to declare who they are accompanied by. The most frequent type of this variable is "unaccompanied" which covers more or less 80% of cases. As shown by the graph above, there is not a category that has an influence on the target variable, in fact, all of them have at least 90% of solvent clients.

NAME_INCOME_TYPE

```
In [20]: ap_train.groupby('NAME_INCOME_TYPE').apply(len)

Out[20]: NAME_INCOME_TYPE
         Businessman                 10
         Commercial associate     71617
         Maternity leave              5
         Pensioner                55362
         State servant            21703
         Student                     18
         Unemployed                  22
         Working                 158774
         dtype: int64
```

```
In [21]: fig = px.histogram(ap_train, width=800, height=500,color="TARGET", x="NAME_INCOME_TYPE", barmode="group",template="simple_white", log_y=True, co
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```



The variable indicates the source of the client's income:

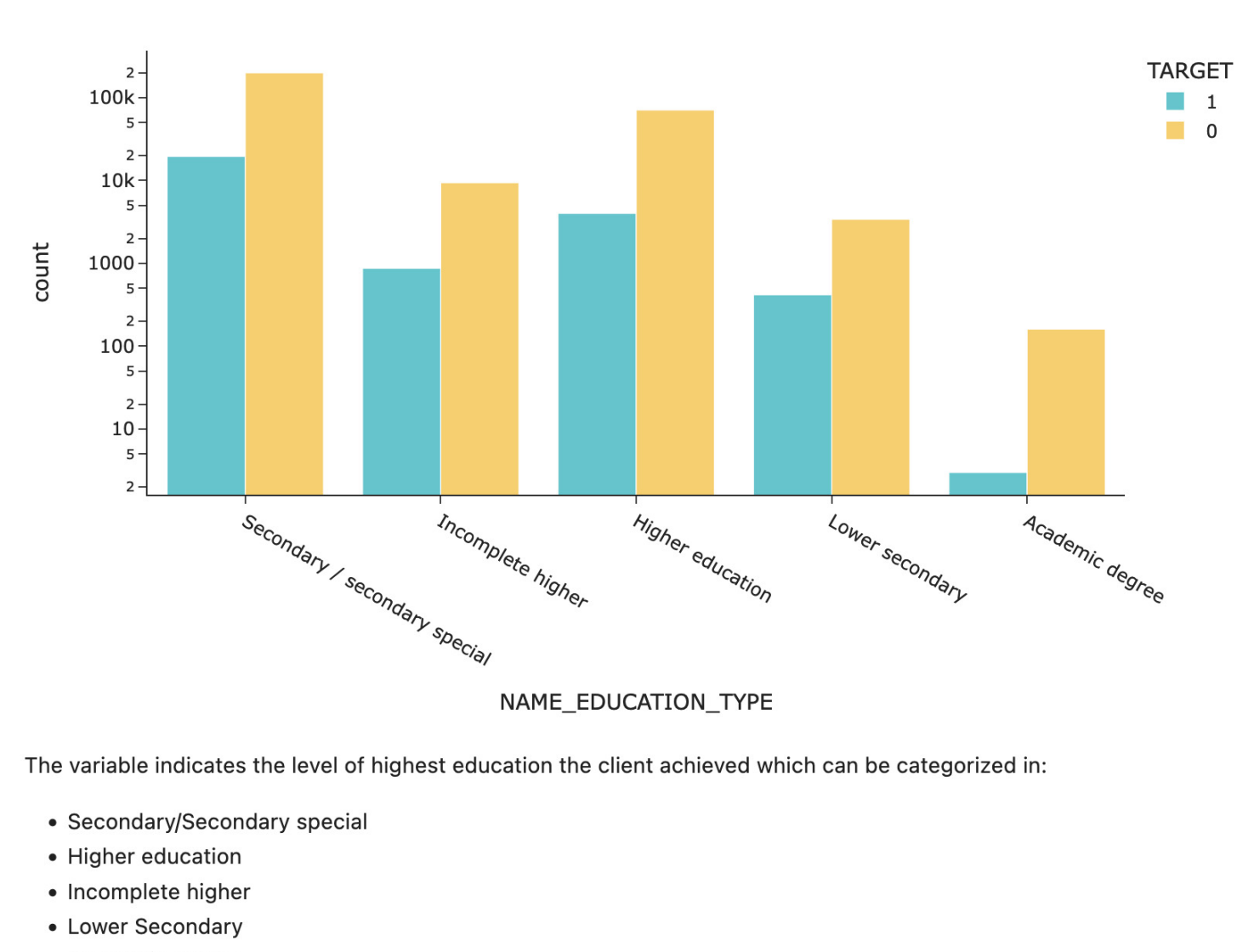
- Working (51% of clients)
- Commercial associate (23% of clients)
- Pensioner (18% of clients)
- State servant (7% of clients)
- Student
- Maternity leave
- Businessman
- Unemployed

The most frequent source of the client's income is general work, the other main categories of income are made up of commercial associates, state servants, and pensioners. The last 1% of income type derives from students, businessmen, people who perceived maternity leave, and people unemployed. In this case, what is expected is fulfilled because students, unemployed persons, businessmen, and women on maternity leave are unlikely to apply for a loan for diametrically opposite reasons. For example, a student or an unemployed person will not apply for a loan because they are unlikely to be granted one and on the other hand, a businessman should not need a loan. One interesting thing that emerges from the graph is that both students and businessmen have no problems paying their loans. Whereas, 30% of unemployed (8 clients) have payment difficulties.

NAME_EDUCATION_TYPE

```
In [22]: ap_train.groupby('NAME_EDUCATION_TYPE').apply(len)

Out[22]: NAME_EDUCATION_TYPE
         Academic degree                     164
         Higher education                  74863
         Incomplete higher                 10277
         Lower secondary                    3816
         Secondary / secondary special    218391
         dtype: int64
```

```
In [23]: fig = px.histogram(ap_train,width=800, height=500, color="TARGET", x="NAME_EDUCATION_TYPE", barmode="group", log_y=True,template="simple_white",
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```



The variable indicates the level of highest education the client achieved which can be categorized in:

- Secondary/Secondary special
- Higher education
- Incomplete higher
- Lower Secondary
- Academic degree.

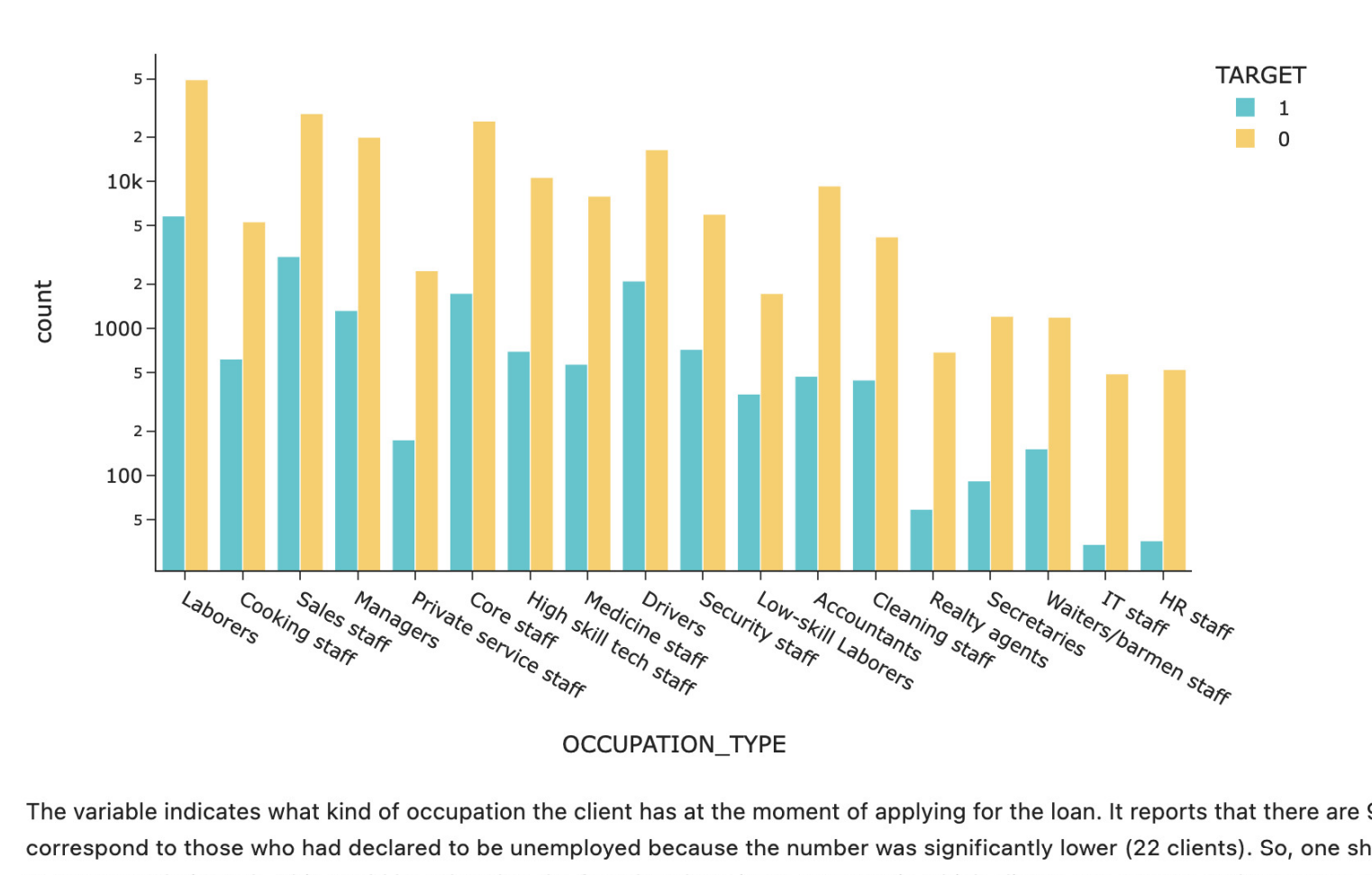
71% of clients have a secondary level of education and 91% of them are solvent. All these categories report, at least, 89% of solvent clients. Therefore, this variable is not relevant to highlighted client categories in terms of solvency.

OCCUPATION_TYPE

```
In [24]: ap_train['OCCUPATION_TYPE'].isnull().sum()

Out[24]: 96391
```

```
In [25]: ap_train.groupby('OCCUPATION_TYPE').apply(len)

Out[25]: OCCUPATION_TYPE
         Accountants               9813
         Cleaning staff            4653
         Cooking staff             5946
         Core staff               27570
         Drivers                  18603
         HR staff                   563
         High skill tech staff    11380
         IT staff                   526
         Laborers                 55186
         Low-skill Laborers        2093
         Managers                 21371
         Medicine staff            8537
         Private service staff     2652
         Realty agents              751
         Sales staff              32102
         Secretaries               1305
         Security staff            6721
         Waiters/barmen staff      1348
         dtype: int64
```

```
In [26]: fig = px.histogram(ap_train, width=800, height=500,x="OCCUPATION_TYPE", color="TARGET", barmode="group", log_y=True, template="simple_white", co
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```

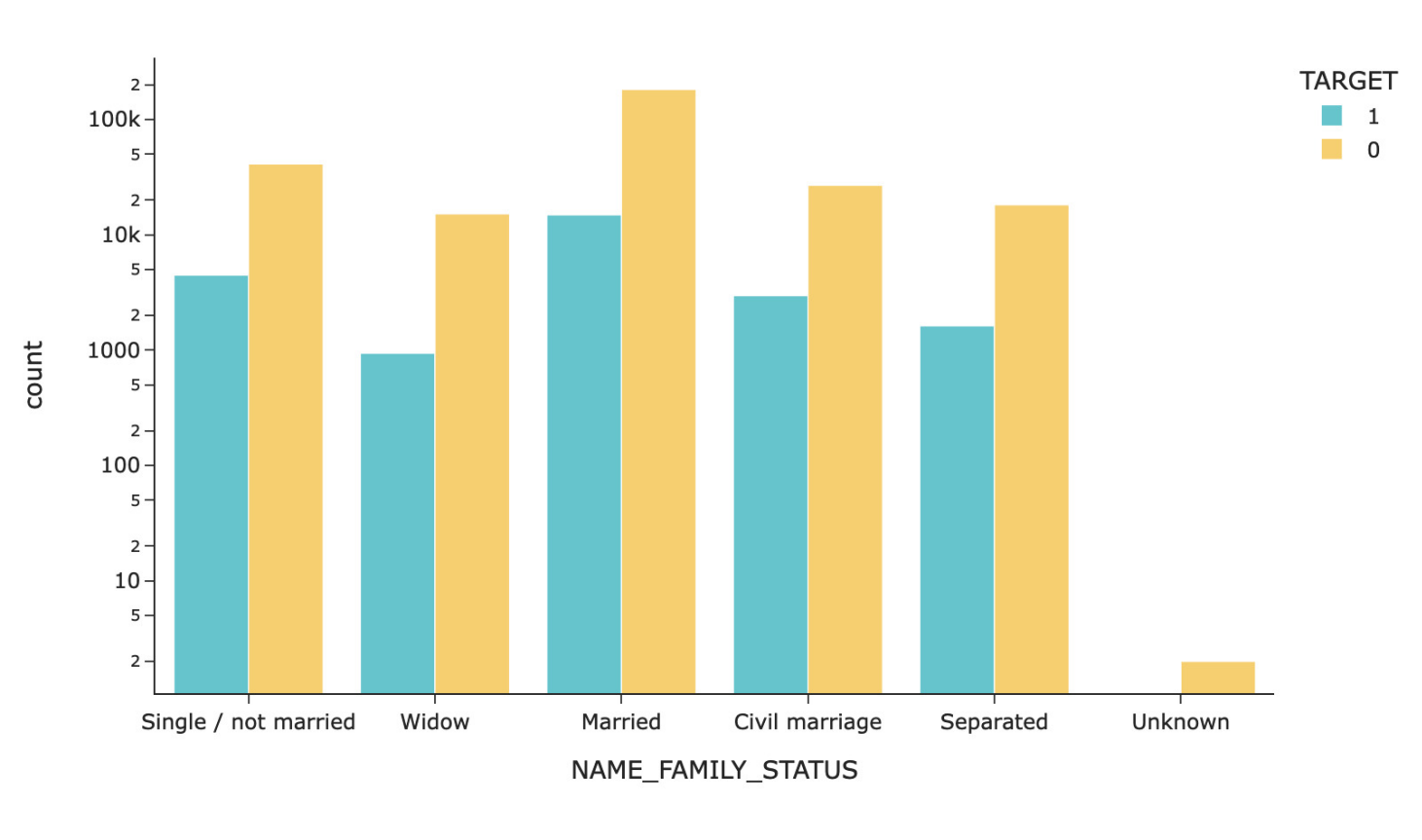


The variable indicates what kind of occupation the client has at the moment of applying for the loan. It reports that there are 96391 missing values which do not correspond to those who had declared to be unemployed because the number was significantly lower (22 clients). So, one should reflect on why 31% of clients preferred to not state their work. This could be related to the fact that there is no category in which clients can express "other" types of work or it could be due to the fact that clients do not feel comfortable expressing their job. Concerning the target variable, all the occupation categories report at least 82% of solvent clients, so there is no evidence that belonging to a certain job category affects the client's solvency.

NAME_FAMILY_TYPE

```
In [27]: ap_train.groupby('NAME_FAMILY_STATUS').apply(len)

Out[27]: NAME_FAMILY_STATUS
         Civil marriage           29775
         Married                 196432
         Separated                19770
         Single / not married     45444
         Unknown                      2
         Widow                    16088
         dtype: int64
```

```
In [28]: fig = px.histogram(ap_train,width=800, height=500, color="TARGET", x="NAME_FAMILY_STATUS", barmode="group", log_y=True, template="simple_white",
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```

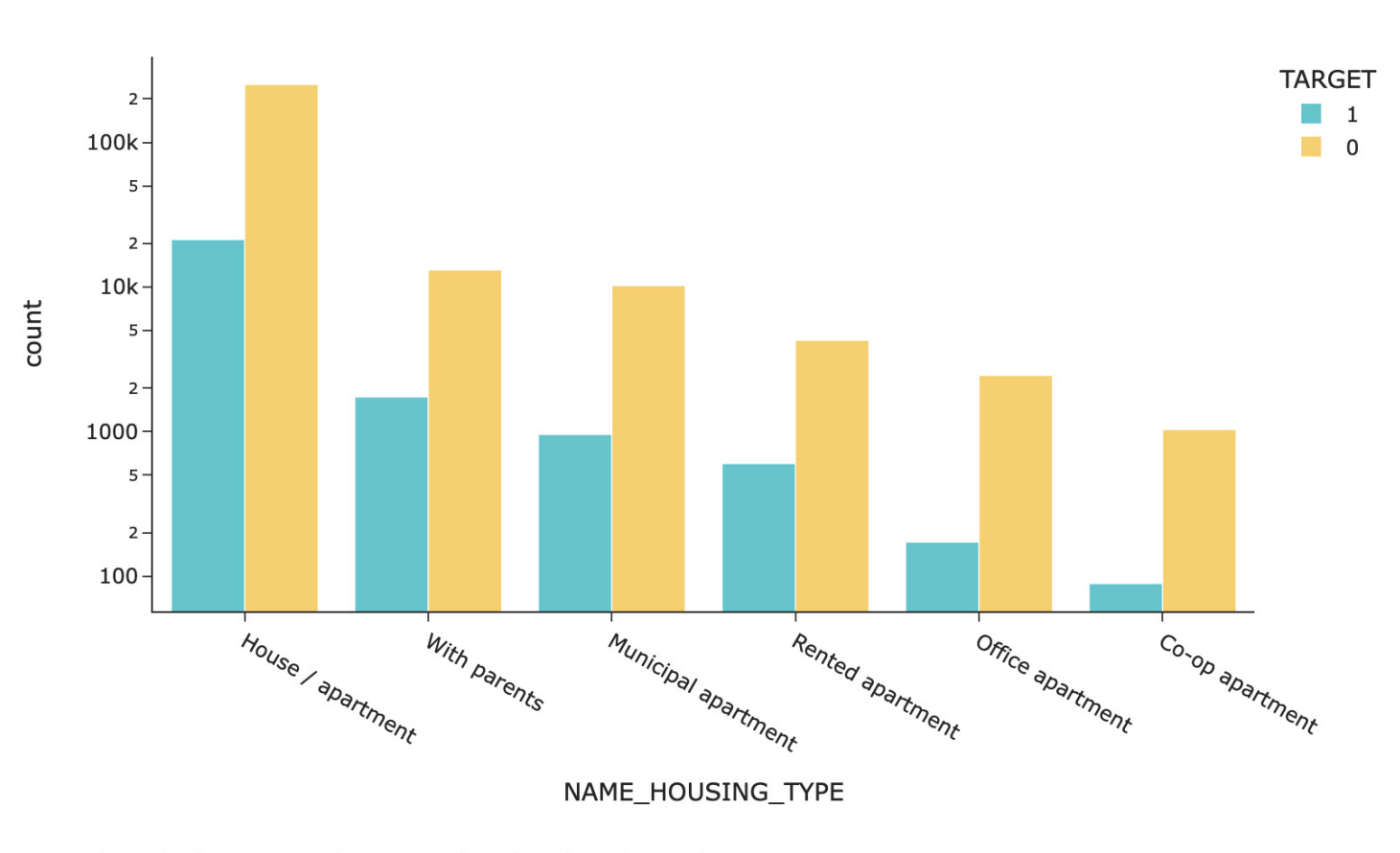


The variable represents the family status of the client. It must be considered that there are 2 different categories for people married: one is related to civil marriage, so the other will probably refer to a religious marriage. Belong to the latter category 63% of clients. Regarding the relationship with the target variable, at least 90% in each category has no payment difficulties, so it is not possible to say that this variable indicates the client's solvency.

NAME_HOUSING_TYPE

```
In [29]: ap_train.groupby('NAME_HOUSING_TYPE').apply(len)

Out[29]: NAME_HOUSING_TYPE
         Co-op apartment          1122
         House / apartment      272868
         Municipal apartment     11183
         Office apartment         2617
         Rented apartment         4881
         With parents            14840
         dtype: int64
```

```
In [30]: fig = px.histogram(ap_train,width=800, height=500, color="TARGET", x="NAME_HOUSING_TYPE", barmode="group", log_y=True, template="simple_white",
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```



The variable indicates what is the housing situation of the client:

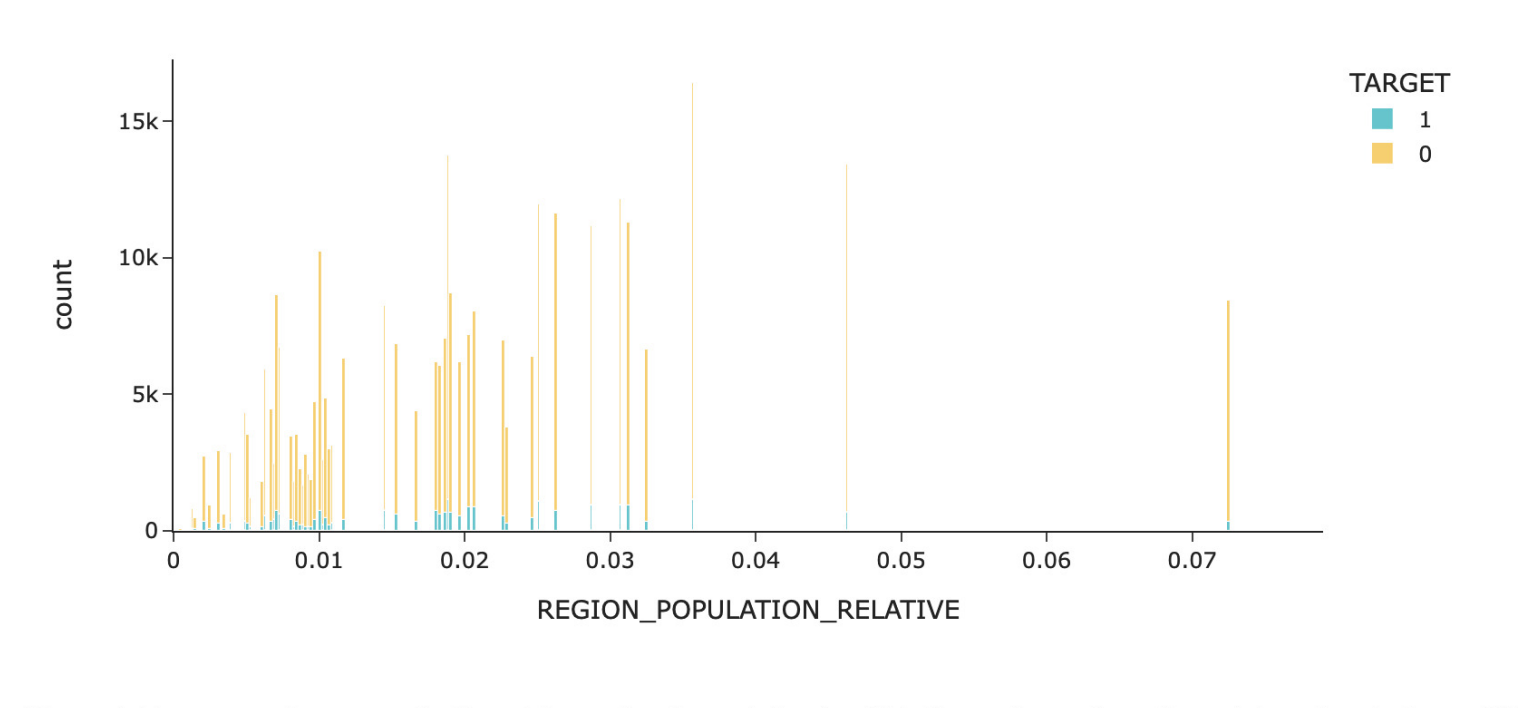
- House/apartment
- Rented apartment
- With parents
- Municipal apartment
- Office apartment
- Co-op apartment

With respect to the target variable, the graph shows that 88% of clients results to have a house or apartment, and among these only 8% have some payment difficulties. The 12% of clients who have rented apartments and the 11% who live with parents have some payment difficulties, but they are only a few cases over the overall number of clients.

REGION_POPULATION_RELATIVE

```
In [31]: ap_train['REGION_POPULATION_RELATIVE'].describe()

Out[31]: count    307511.000000
         mean          0.020868
         std           0.013831
         min           0.000290
         25%           0.010006
         50%           0.018850
         75%           0.028663
         max           0.072508
         Name: REGION_POPULATION_RELATIVE, dtype: float64
```

```
In [32]: fig = px.histogram(ap_train,width=800, height=400, x="REGION_POPULATION_RELATIVE", color="TARGET", range_x=[0,0.079],template="simple_white", co
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```



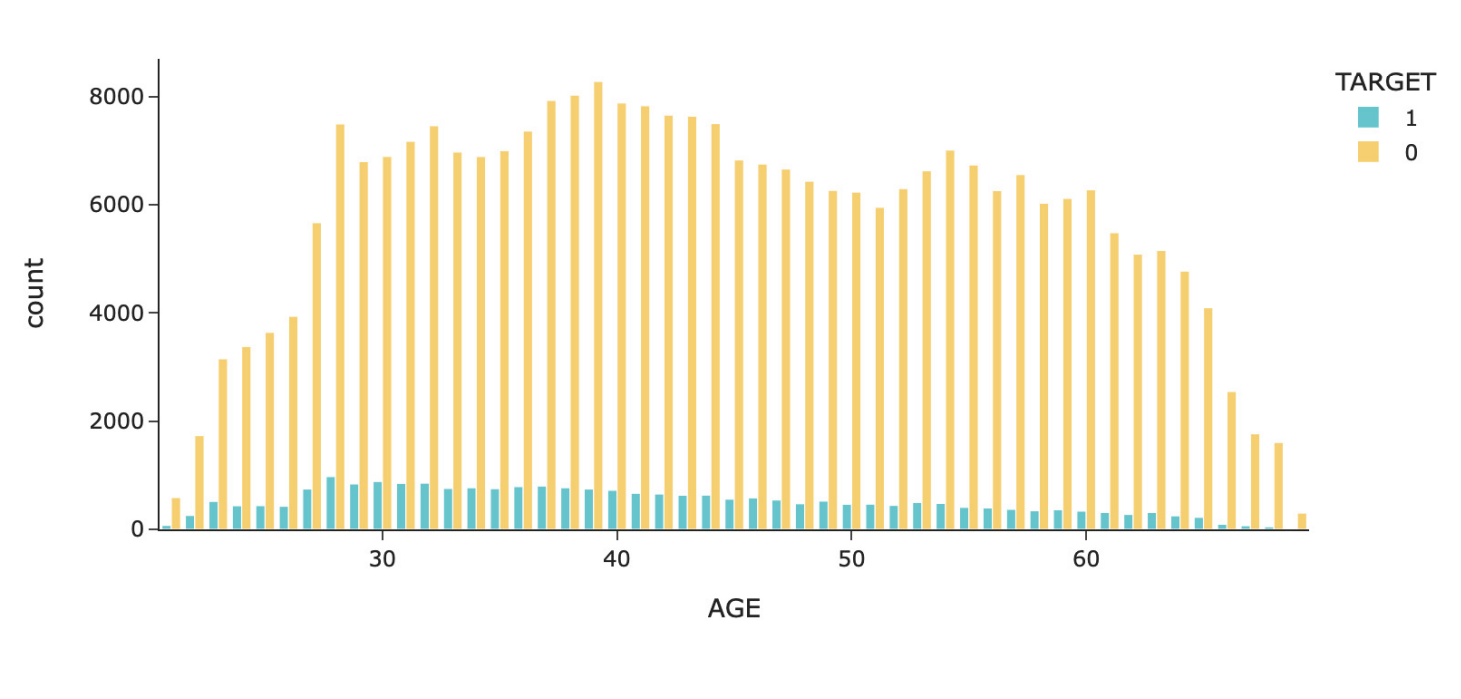
The variable represents a normalisation of the regional population in which the customer lives. It can take values between 0.000290 and 0.072508: a higher number means that the customer lives in a more populated region. The graph shows that most customers live in sparsely populated regions.

DAYS_BIRTH

```
In [33]: ap_train['AGE'].min()

Out[33]: 21.0
```

```
In [34]: ap_train['AGE'].max()

Out[34]: 69.0
```

```
In [35]: fig = px.histogram(ap_train, width=800, height=400, x="AGE", color="TARGET", barmode="group",template="simple_white", color_discrete_sequence=px
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```



The variable reports the age of clients which range goes from 21 to 69. Overall, the variable has no relevant implication on the target, but, as expected, the younger age group reflects a more difficult payment behaviour.

DAYS_EMPLOYED

```
In [36]: ap_train['DAYS_EMPLOYED']

Out[36]: 0           -637
         1          -1188
         2           -225
         3          -3039
         4          -3038
                    ...
         307506      -236
         307507    365243
         307508     -7921
         307509     -4786
         307510     -1262
         Name: DAYS_EMPLOYED, Length: 307511, dtype: int64
```

```
In [37]: year_employed = ap_train['DAYS_EMPLOYED']//365
         year_employed.describe()

Out[37]: count    307511.000000
         mean        174.303742
         std         386.994347
         min         -50.000000
         25%          -8.000000
         50%          -4.000000
         75%          -1.000000
         max        1000.000000
         Name: DAYS_EMPLOYED, dtype: float64
```

```
In [38]: fig = px.histogram(ap_train, width=800, height=400, x="DAYS_EMPLOYED", color="TARGET",template="simple_white", color_discrete_sequence=px.colors
         fig.update_layout(xaxis=dict(showgrid=False, ), yaxis=dict(showgrid=False))
         fig.show()
```