

PREVISÃO DE RISCO DE INADIMPLÊNCIA — HOME CREDIT

1. Objetivo

Desenvolver um modelo preditivo que estime a probabilidade de inadimplência de clientes (variável TARGET) usando dados demográficos, financeiros, histórico de crédito e comportamento passado. O objetivo prático é permitir que a instituição tome decisões de crédito mais seguras: ajustar limites, aprovar ou recusar solicitações e priorizar ações de cobrança preventivas, reduzindo perdas e melhorando rentabilidade.

Considerando a complexidade e o alto desbalanceamento do conjunto de dados do Home Credit Default Risk, o modelo foi ajustado para equilibrar precisão e capacidade de detecção (recall), priorizando a identificação de clientes com maior risco de inadimplência. O foco é apoiar decisões de crédito mais seguras e proativas, minimizando o risco de perdas e fortalecendo a gestão do portfólio financeiro.

2. Sumário Executivo

O projeto teve como objetivo o **desenvolvimento de um modelo preditivo de inadimplência** para apoiar decisões de crédito mais seguras e rentáveis, utilizando o conjunto de dados *Home Credit Default Risk*, caracterizado por **alta complexidade, mais de 800 variáveis e forte desbalanceamento de classes** (aproximadamente **8% de inadimplentes**).

Mesmo diante de variáveis com **baixa correlação individual** com o risco de inadimplência, a aplicação de técnicas estatísticas e de *machine learning* permitiu **combinar múltiplos sinais fracos em um modelo robusto e interpretável**, capaz de estimar com consistência a probabilidade de atraso.

1. Desafio de Negócio

A necessidade central era **reduzir perdas financeiras com inadimplência** sem comprometer a experiência e a conversão de bons clientes. Os principais desafios foram:

- Identificar clientes de **alto risco com antecedência**;
- Reduzir o impacto de **falsos negativos** (clientes inadimplentes não detectados);
- Controlar o volume de **falsos positivos** e seus custos operacionais;
- Garantir **escalabilidade** em um ambiente com centenas de variáveis.

2. Solução Desenvolvida

Foi construído um **modelo de classificação binária** treinado com um conjunto amplo de variáveis demográficas, socioeconômicas, comportamentais e de histórico de crédito, totalizando **mais de 800 atributos**.

O modelo foi calibrado com **threshold ótimo de 0,191 (baseado em F1-score ≈ 0,344)**, priorizando a **captura de inadimplentes** sem perda excessiva de clientes bons, tornando-o adequado para **etapas de triagem e prevenção de risco**.

Principais padrões identificados:

- Estabilidade empregatícia e histórico financeiro consistente reduzem o risco;
- Alto volume de consultas ao crédito e decisões financeiras recentes elevam o risco;

- Mudanças comportamentais recentes são bons indicadores de instabilidade;
 - A **combinação de múltiplas variáveis** mostrou-se essencial para a capacidade preditiva.
-

3. Desempenho e Resultados

Métrica	Resultado
Recall (Inadimplentes)	38%
Precision (Inadimplentes)	31%
AUC-ROC	0,80
MCC	0,28
Acurácia geral	88%

O modelo demonstrou **boa capacidade de discriminação**, especialmente considerando o forte desbalanceamento da base e a alta dimensionalidade dos dados.

4. Impacto Financeiro

Com base em simulações realistas (46.127 clientes, perda média de **R\$ 5.000** por inadimplente e custo médio de **R\$ 500** por ação preventiva), o impacto estimado foi:

Cenário	Lucro Líquido Estimado
Sem modelo	R\$ 23,78 milhões
Com modelo (th = 0,191)	R\$ 30,17 milhões

Ganho incremental estimado: aproximadamente **R\$ 6,39 milhões**.

Esse resultado mostra que o modelo gera **valor financeiro direto**, mesmo após considerar perdas com falsos positivos e custos operacionais.

5. Conclusão

O modelo cumpriu seu objetivo principal: **reduzir o risco de inadimplência e aumentar a rentabilidade da carteira** de forma mensurável.

O projeto demonstrou que, mesmo em um cenário de **alta complexidade, mais de 800 variáveis e forte desbalanceamento**, é possível construir soluções:

- Robustas,
- Interpretáveis,
- Escaláveis,
- Financeiramente vantajosas.

Em síntese, o trabalho reforça que a **ciência de dados aplicada à gestão de crédito** não apenas melhora métricas técnicas, mas **gera impacto real e sustentável no resultado do negócio**.

3. Sobre os Dados

Os dados utilizados nesta análise referem-se ao histórico financeiro, demográfico e de comportamento de clientes da base **Home Credit Default Risk**, que contém informações detalhadas sobre milhares de indivíduos com diferentes perfis socioeconômicos.

Atualmente, a tomada de decisão de crédito depende de múltiplos fatores, como histórico de

empréstimos, capacidade de pagamento, características do imóvel, situação profissional e padrões de comportamento social. Essa complexidade e variabilidade tornam as decisões manuais pouco consistentes, justificando a aplicação de técnicas de Ciência de Dados e Aprendizado de Máquina para apoiar a avaliação de risco de inadimplência e a gestão proativa do portfólio de clientes.

4. Iniciando

4.1 Importação de bibliotecas

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import chi2_contingency
from sklearn.preprocessing import RobustScaler, StandardScaler, MinMaxScaler, OneHotEncoder
from sklearn.model_selection import train_test_split
import xgboost as xgb
from sklearn.metrics import recall_score, precision_score, f1_score, roc_auc_score, average_precision_score
from skopt import BayesSearchCV
from skopt.space import Real, Integer, Categorical
import lightgbm as lgb
from imblearn.ensemble import BalancedRandomForestClassifier
from catboost import CatBoostClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.base import clone
from sklearn.ensemble import AdaBoostClassifier, HistGradientBoostingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression
import shap
from sklearn.preprocessing import PolynomialFeatures
```

4.2 Importação de Estilo

```
In [2]: plt.style.use('..../styles/personalestilo.mplstyle')

Bad key axes.color_cycle in file ..../styles/personalestilo.mplstyle, line 9 ('axes.color_cycle:
df691b, 5cb85c, 5bc0de, f0ad4e, d9534f, 4e5d6c')
You probably need to get an updated matplotlibrc file from
https://github.com/matplotlib/matplotlib/blob/v3.10.6/lib/matplotlib/mpl-data/matplotlibrc
or from the matplotlib source distribution
```

4.3 Funções Adicionais

```
In [2]: def snake_case(lst):
    def convert(s):
        s = s.replace(' ', '_')
        new_s = ""
        for i, c in enumerate(s):
            if c.isupper():
                if i > 0 and (s[i-1].islower() or (i+1 < len(s) and s[i+1].islower())):
                    new_s += "_"
                new_s += c.lower()
            else:
                new_s += c
        return new_s
    return [convert(s) for s in lst]

def agg_numeric(df, key, prefix):
    df_num = df.select_dtypes(include=[np.number])

    df_num[key] = df[key]
```

```

agg = df_num.groupby(key).agg(['mean', 'max', 'min', 'count'])

agg.columns = [prefix + "_" + "_".join(col) for col in agg.columns]

return agg

def plot_numeric_block(df_block, bins=30, color='orange', rows=3, figsize=(18, 12)):
    plt.figure(figsize=figsize)
    cols = df_block.columns
    n_cols = int(np.ceil(len(cols) / rows))
    for i, col in enumerate(cols, 1):
        plt.subplot(rows, n_cols, i)
        sns.histplot(df_block[col], kde=True, color=color, bins=bins)
        plt.title(col)
        plt.xlabel(col)
        plt.ylabel('Frequênciac')
    plt.tight_layout()
    plt.show()

def plot_target_by_categorical(df, cat_col, target_col='target', suptitle=None):
    categorias = df[cat_col].unique()
    categorias = [str(c) for c in categorias]
    n_categorias = len(categorias)
    n_linhas = (n_categorias // 3) + (1 if n_categorias % 3 != 0 else 0)
    n_linhas = max(n_linhas, 1)
    ncols = 3 if n_categorias > 1 else 1

    fig, axes = plt.subplots(n_linhas, ncols, figsize=(7 * ncols, 5 * n_linhas))
    axes = np.array(axes).reshape(-1)

    palette_target = {'0': "orange", '1': "red"}

    for ax in axes[n_categorias:]:
        ax.axis('off')

    for i, categoria in enumerate(categorias):
        data_plot = df[df[cat_col] == categoria].copy()
        data_plot['target_str'] = data_plot[target_col].astype(str)
        sns.countplot(
            data=data_plot,
            x='target_str',
            hue='target_str',
            palette=palette_target,
            order=['0', '1'],
            ax=axes[i],
            legend=False
        )
        axes[i].set_title(f'{cat_col}: {categoria}')
        axes[i].set_xlabel(target_col)
        axes[i].set_ylabel('Frequênciac')
        axes[i].set_xticks([0, 1])
        axes[i].set_xticklabels(['Não Inadimplente (0)', 'Inadimplente (1)'])

    if suptitle is None:
        suptitle = f'Frequênciac de {target_col} por categoria de {cat_col}'
    plt.suptitle(suptitle)
    plt.tight_layout(rect=[0, 0, 1, 0.95])
    plt.show()

def find_best_threshold(y_true, y_proba, metric="f1"):
    thresholds = np.linspace(0, 1, 200)

    best_t = 0.5
    best_score = -1

    for t in thresholds:
        y_pred = (y_proba >= t).astype(int)

```

```

    if metric == "f1":
        score = f1_score(y_true, y_pred)
    elif metric == "mcc":
        score = matthews_corrcoef(y_true, y_pred)
    elif metric == "recall":
        score = recall_score(y_true, y_pred)
    elif metric == "precision":
        score = precision_score(y_true, y_pred)
    elif metric == "f2":
        score = fbeta_score(y_true, y_pred, beta=2)
    elif metric == "f05":
        score = fbeta_score(y_true, y_pred, beta=0.5)
    else:
        raise ValueError("Métrica inválida.")

    if score > best_score:
        best_score = score
        best_t = t

    return best_t, best_score

def get_fold(X, idx):
    return X.iloc[idx] if hasattr(X, "iloc") else X[idx]

def get_model_clone(name, model):
    if "cat" in name:
        return CatBoostClassifier(**model.get_params())
    elif "xgb" in name:
        return xgb.XGBClassifier(**model.get_params())
    elif "lgbm" in name:
        return lgb.LGBMClassifier(**model.get_params())
    else:
        return clone(model)

def plot_binaria_target(
    df,
    var_binaria,
    target='target',
    label_0='Não Possui',
    label_1='Possui',
    suptitle=None
):
    flags = [0, 1]
    palette_target = {'0': "orange", '1': "red"}

    fig, axes = plt.subplots(1, 2, figsize=(14, 6), sharey=False)
    for i, flag in enumerate(flags):
        data_plot = df[df[var_binaria] == flag].copy()
        data_plot['target_str'] = data_plot[target].astype(str)
        sns.countplot(
            data=data_plot,
            x='target_str',
            hue='target_str',
            palette=palette_target,
            order=['0', '1'],
            ax=axes[i],
            legend=False
        )
        label = label_0 if flag == 0 else label_1
        axes[i].set_title(f"{label}")
        axes[i].set_xlabel('Resposta')
        axes[i].set_ylabel('Frequência')
        axes[i].set_xticks([0, 1])
        axes[i].set_xticklabels(['Não Inadimplente (0)', 'Inadimplente (1)'])

    if suptitle is None:

```

```

        suptitle = f'Frequênci a de {target} por {var_binaria}'
        plt.suptitle(suptitle)
        plt.tight_layout(rect=[0, 0, 1, 0.95])
        plt.show()

def plot_num_var_by_target(
    df,
    num_var,
    target_var='target',
    title_0=None,
    title_1=None,
    label_x=None,
    color_0='blue',
    color_1='orange',
    figsize=(10, 8),
    discrete=False
):
    fig, axes = plt.subplots(2, 1, figsize=figsize, sharey=False)

    sns.histplot(
        data=df[df[target_var] == 0],
        x=num_var,
        color=color_0,
        ax=axes[0],
        discrete=discrete
    )
    axes[0].set_title(title_0 or f'Distribuição de {num_var} (Sem Inadimplência)')
    axes[0].set_ylabel('Frequênci a')
    axes[0].set_xlabel(label_x or num_var)

    sns.histplot(
        data=df[df[target_var] == 1],
        x=num_var,
        color=color_1,
        ax=axes[1],
        discrete=discrete
    )
    axes[1].set_title(title_1 or f'Distribuição de {num_var} (Inadimplentes)')
    axes[1].set_ylabel('Frequênci a')
    axes[1].set_xlabel(label_x or num_var)

    plt.tight_layout()
    plt.show()

def calcular_cramers_v(df, col1, col2):
    contingency_table = pd.crosstab(df[col1], df[col2])
    chi2_stat, p, dof, expected = chi2_contingency(contingency_table)
    n = contingency_table.sum().sum()
    min_dim = min(contingency_table.shape) - 1
    cramers_v = np.sqrt(chi2_stat / (n * min_dim)) if min_dim > 0 else np.nan
    print(f"V de Cramer entre {col1} e {col2}: {cramers_v:.4f}")

def avaliar_metricas(modelo, X_val, y_val, X_test, y_test):
    def obter_prob(modelo, X):
        if hasattr(modelo, "predict_proba"):
            probas = modelo.predict_proba(X)
            return probas[:, 1]
        elif hasattr(modelo, "predict"):
            if isinstance(X, (pd.DataFrame, pd.Series, np.ndarray)):
                dmatrix = xgb.DMatrix(X)
            else:
                dmatrix = X
            probas = modelo.predict(dmatrix)
            return probas
        else:
            raise ValueError("O modelo precisa implementar predict_proba ou ser booster do xgbc")

```

```

y_val_prob = obter_prob(modelo, X_val)
y_test_prob = obter_prob(modelo, X_test)

y_val_pred = (y_val_prob >= 0.5).astype(int)
y_test_pred = (y_test_prob >= 0.5).astype(int)

metrics = {
    'recall': [
        recall_score(y_val, y_val_pred),
        recall_score(y_test, y_test_pred)
    ],
    'precision': [
        precision_score(y_val, y_val_pred),
        precision_score(y_test, y_test_pred)
    ],
    'f1_score': [
        f1_score(y_val, y_val_pred),
        f1_score(y_test, y_test_pred)
    ],
    'auc_roc': [
        roc_auc_score(y_val, y_val_prob),
        roc_auc_score(y_test, y_test_prob)
    ],
    'auc_pr': [
        average_precision_score(y_val, y_val_prob),
        average_precision_score(y_test, y_test_prob)
    ]
}
df_metrics = pd.DataFrame(metrics, index=['validacao', 'teste']).T
return df_metrics

def income_stability(row):
    low_income_types = ['Unemployed', 'Student', 'Maternity leave', 'Pensioner']
    high_income_types = ['State servant', 'Businessman', 'Civil servant']
    stable_occupations = [
        'Accountants', 'Core staff', 'HR staff', 'High skill tech staff', 'Managers', 'Medicine',
        'Private service staff', 'Secretaries', 'Officials', 'IT staff'
    ]
    unstable_occupations = [
        'Laborers', 'Low-skill Laborers', 'Cleaning staff', 'Waiters/barmen staff', 'Security s'
    ]
    inc_type = str(row['name_income_type'])
    occ_type = str(row['occupation_type'])
    days_employed = abs(row['days_employed']) if not pd.isnull(row['days_employed']) else 0

    if (inc_type in low_income_types) or (occ_type in unstable_occupations) or (days_employed <
        return 0
    elif (inc_type in high_income_types) or (occ_type in stable_occupations and days_employed >
        return 2
    else:
        return 1

```

4.4 Carregando os dados

In [4]:

```
app_train = pd.read_csv('../data/application_train.csv')
app_test = pd.read_csv('../data/application_test.csv')
```

In [5]:

```
bureau = pd.read_csv('../data/bureau.csv')
bureau.name = "bureau"

bureau_balance = pd.read_csv('../data/bureau_balance.csv')
bureau_balance.name = "bb"

prev = pd.read_csv('../data/previous_application.csv')
prev.name = "prev"
```

```

pos = pd.read_csv('../data/POS_CASH_balance.csv')
pos.name = "pos"

cc = pd.read_csv('../data/credit_card_balance.csv')
cc.name = "cc"

inst = pd.read_csv('../data/installments_payments.csv')
inst.name = "inst"

```

In [6]:

```

bb_agg = agg_numeric(bureau_balance, 'SK_ID_BUREAU', 'bb')
bureau = bureau.merge(bb_agg, on='SK_ID_BUREAU', how='left')

bureau_agg = agg_numeric(bureau, 'SK_ID_CURR', 'bur')

prev_agg = agg_numeric(prev, 'SK_ID_CURR', 'prev')

pos_agg = agg_numeric(pos, 'SK_ID_PREV', 'pos')

prev_pos = prev[['SK_ID_CURR', 'SK_ID_PREV']].merge(pos_agg, on='SK_ID_PREV', how='left')
pos_final = agg_numeric(prev_pos, 'SK_ID_CURR', 'posf')

cc_agg = agg_numeric(cc, 'SK_ID_PREV', 'cc')

prev_cc = prev[['SK_ID_CURR', 'SK_ID_PREV']].merge(cc_agg, on='SK_ID_PREV', how='left')
cc_final = agg_numeric(prev_cc, 'SK_ID_CURR', 'ccf')

inst_agg = agg_numeric(inst, 'SK_ID_PREV', 'inst')

prev_inst = prev[['SK_ID_CURR', 'SK_ID_PREV']].merge(inst_agg, on='SK_ID_PREV', how='left')
inst_final = agg_numeric(prev_inst, 'SK_ID_CURR', 'instf')

df = app_train.copy()
df_test = app_test.copy()

for auxiliary in [bureau_agg, prev_agg, pos_final, cc_final, inst_final]:
    df = df.merge(auxiliary, on='SK_ID_CURR', how='left')
    df_test = df_test.merge(auxiliary, on='SK_ID_CURR', how='left')

```

In [31]:

```

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

```

In [8]:

```

df.to_csv('../data/train.csv', index=False)
df_test.to_csv('../data/test.csv', index=False)

```

5. Decrição dos Dados

In []:

```

df1 = df.copy()

```

5.1 Renomeando as Colunas

In [4]:

```

cols_old = list(df1.columns)

cols_old = list(df1.columns)

cols_new = snake_case(cols_old)

df1.columns = cols_new

```

In [5]:

```

for idx, (col, dtype) in enumerate(df1.dtypes.items(), 1):
    print(f'{idx} - {col}: {dtype}')

```

```
1 - sk_id_curr: int64
2 - target: int64
3 - name_contract_type: object
4 - code_gender: object
5 - flag_own_car: object
6 - flag_own_realty: object
7 - cnt_children: int64
8 - amt_income_total: float64
9 - amt_credit: float64
10 - amt_annuity: float64
11 - amt_goods_price: float64
12 - name_type_suite: object
13 - name_income_type: object
14 - name_education_type: object
15 - name_family_status: object
16 - name_housing_type: object
17 - region_population_relative: float64
18 - days_birth: int64
19 - days_employed: int64
20 - days_registration: float64
21 - days_id_publish: int64
22 - own_car_age: float64
23 - flag_mobil: int64
24 - flag_emp_phone: int64
25 - flag_work_phone: int64
26 - flag_cont_mobile: int64
27 - flag_phone: int64
28 - flag_email: int64
29 - occupation_type: object
30 - cnt_fam_members: float64
31 - region_rating_client: int64
32 - region_rating_client_w_city: int64
33 - weekday_appr_process_start: object
34 - hour_appr_process_start: int64
35 - reg_region_not_live_region: int64
36 - reg_region_not_work_region: int64
37 - live_region_not_work_region: int64
38 - reg_city_not_live_city: int64
39 - reg_city_not_work_city: int64
40 - live_city_not_work_city: int64
41 - organization_type: object
42 - ext_source_1: float64
43 - ext_source_2: float64
44 - ext_source_3: float64
45 - apartments_avg: float64
46 - basementarea_avg: float64
47 - years_beginexpluatation_avg: float64
48 - years_build_avg: float64
49 - commonarea_avg: float64
50 - elevators_avg: float64
51 - entrances_avg: float64
52 - floorsmax_avg: float64
53 - floorsmin_avg: float64
54 - landarea_avg: float64
55 - livingapartments_avg: float64
56 - livingarea_avg: float64
57 - nonlivingapartments_avg: float64
58 - nonlivingarea_avg: float64
59 - apartments_mode: float64
60 - basementarea_mode: float64
61 - years_beginexpluatation_mode: float64
62 - years_build_mode: float64
63 - commonarea_mode: float64
64 - elevators_mode: float64
65 - entrances_mode: float64
66 - floorsmax_mode: float64
67 - floorsmin_mode: float64
68 - landarea_mode: float64
69 - livingapartments_mode: float64
```

```
70 - livingarea_mode: float64
71 - nonlivingapartments_mode: float64
72 - nonlivingarea_mode: float64
73 - apartments_medi: float64
74 - basementarea_medi: float64
75 - years_beginexpluatation_medi: float64
76 - years_build_medi: float64
77 - commonarea_medi: float64
78 - elevators_medi: float64
79 - entrances_medi: float64
80 - floorsmax_medi: float64
81 - floorsmin_medi: float64
82 - landarea_medi: float64
83 - livingapartments_medi: float64
84 - livingarea_medi: float64
85 - nonlivingapartments_medi: float64
86 - nonlivingarea_medi: float64
87 - fondkapremont_mode: object
88 - housetype_mode: object
89 - totalarea_mode: float64
90 - wallsmaterial_mode: object
91 - emergencystate_mode: object
92 - obs_30_cnt_social_circle: float64
93 - def_30_cnt_social_circle: float64
94 - obs_60_cnt_social_circle: float64
95 - def_60_cnt_social_circle: float64
96 - days_last_phone_change: float64
97 - flag_document_2: int64
98 - flag_document_3: int64
99 - flag_document_4: int64
100 - flag_document_5: int64
101 - flag_document_6: int64
102 - flag_document_7: int64
103 - flag_document_8: int64
104 - flag_document_9: int64
105 - flag_document_10: int64
106 - flag_document_11: int64
107 - flag_document_12: int64
108 - flag_document_13: int64
109 - flag_document_14: int64
110 - flag_document_15: int64
111 - flag_document_16: int64
112 - flag_document_17: int64
113 - flag_document_18: int64
114 - flag_document_19: int64
115 - flag_document_20: int64
116 - flag_document_21: int64
117 - amt_req_credit_bureau_hour: float64
118 - amt_req_credit_bureau_day: float64
119 - amt_req_credit_bureau_week: float64
120 - amt_req_credit_bureau_mon: float64
121 - amt_req_credit_bureau_qrt: float64
122 - amt_req_credit_bureau_year: float64
123 - bur_sk_id_bureau_mean: float64
124 - bur_sk_id_bureau_max: float64
125 - bur_sk_id_bureau_min: float64
126 - bur_sk_id_bureau_count: float64
127 - bur_days_credit_mean: float64
128 - bur_days_credit_max: float64
129 - bur_days_credit_min: float64
130 - bur_days_credit_count: float64
131 - bur_credit_day_overdue_mean: float64
132 - bur_credit_day_overdue_max: float64
133 - bur_credit_day_overdue_min: float64
134 - bur_credit_day_overdue_count: float64
135 - bur_days_credit_enddate_mean: float64
136 - bur_days_credit_enddate_max: float64
137 - bur_days_credit_enddate_min: float64
138 - bur_days_credit_enddate_count: float64
```

```
139 - bur_days_enddate_fact_mean: float64
140 - bur_days_enddate_fact_max: float64
141 - bur_days_enddate_fact_min: float64
142 - bur_days_enddate_fact_count: float64
143 - bur_amt_credit_max_overdue_mean: float64
144 - bur_amt_credit_max_overdue_max: float64
145 - bur_amt_credit_max_overdue_min: float64
146 - bur_amt_credit_max_overdue_count: float64
147 - bur_cnt_credit_prolong_mean: float64
148 - bur_cnt_credit_prolong_max: float64
149 - bur_cnt_credit_prolong_min: float64
150 - bur_cnt_credit_prolong_count: float64
151 - bur_amt_credit_sum_mean: float64
152 - bur_amt_credit_sum_max: float64
153 - bur_amt_credit_sum_min: float64
154 - bur_amt_credit_sum_count: float64
155 - bur_amt_credit_sum_debt_mean: float64
156 - bur_amt_credit_sum_debt_max: float64
157 - bur_amt_credit_sum_debt_min: float64
158 - bur_amt_credit_sum_debt_count: float64
159 - bur_amt_credit_sum_limit_mean: float64
160 - bur_amt_credit_sum_limit_max: float64
161 - bur_amt_credit_sum_limit_min: float64
162 - bur_amt_credit_sum_limit_count: float64
163 - bur_amt_credit_sum_overdue_mean: float64
164 - bur_amt_credit_sum_overdue_max: float64
165 - bur_amt_credit_sum_overdue_min: float64
166 - bur_amt_credit_sum_overdue_count: float64
167 - bur_days_credit_update_mean: float64
168 - bur_days_credit_update_max: float64
169 - bur_days_credit_update_min: float64
170 - bur_days_credit_update_count: float64
171 - bur_amt_annuity_mean: float64
172 - bur_amt_annuity_max: float64
173 - bur_amt_annuity_min: float64
174 - bur_amt_annuity_count: float64
175 - bur_bb_months_balance_mean_mean: float64
176 - bur_bb_months_balance_mean_max: float64
177 - bur_bb_months_balance_mean_min: float64
178 - bur_bb_months_balance_mean_count: float64
179 - bur_bb_months_balance_max_mean: float64
180 - bur_bb_months_balance_max_max: float64
181 - bur_bb_months_balance_max_min: float64
182 - bur_bb_months_balance_max_count: float64
183 - bur_bb_months_balance_min_mean: float64
184 - bur_bb_months_balance_min_max: float64
185 - bur_bb_months_balance_min_min: float64
186 - bur_bb_months_balance_min_count: float64
187 - bur_bb_months_balance_count_mean: float64
188 - bur_bb_months_balance_count_max: float64
189 - bur_bb_months_balance_count_min: float64
190 - bur_bb_months_balance_count_count: float64
191 - prev_sk_id_prev_mean: float64
192 - prev_sk_id_prev_max: float64
193 - prev_sk_id_prev_min: float64
194 - prev_sk_id_prev_count: float64
195 - prev_amt_annuity_mean: float64
196 - prev_amt_annuity_max: float64
197 - prev_amt_annuity_min: float64
198 - prev_amt_annuity_count: float64
199 - prev_amt_application_mean: float64
200 - prev_amt_application_max: float64
201 - prev_amt_application_min: float64
202 - prev_amt_application_count: float64
203 - prev_amt_credit_mean: float64
204 - prev_amt_credit_max: float64
205 - prev_amt_credit_min: float64
206 - prev_amt_credit_count: float64
207 - prev_amt_down_payment_mean: float64
```

```
208 - prev_amt_down_payment_max: float64
209 - prev_amt_down_payment_min: float64
210 - prev_amt_down_payment_count: float64
211 - prev_amt_goods_price_mean: float64
212 - prev_amt_goods_price_max: float64
213 - prev_amt_goods_price_min: float64
214 - prev_amt_goods_price_count: float64
215 - prev_hour_appr_process_start_mean: float64
216 - prev_hour_appr_process_start_max: float64
217 - prev_hour_appr_process_start_min: float64
218 - prev_hour_appr_process_start_count: float64
219 - prev_nflag_last_appl_in_day_mean: float64
220 - prev_nflag_last_appl_in_day_max: float64
221 - prev_nflag_last_appl_in_day_min: float64
222 - prev_nflag_last_appl_in_day_count: float64
223 - prev_rate_down_payment_mean: float64
224 - prev_rate_down_payment_max: float64
225 - prev_rate_down_payment_min: float64
226 - prev_rate_down_payment_count: float64
227 - prev_rate_interest_primary_mean: float64
228 - prev_rate_interest_primary_max: float64
229 - prev_rate_interest_primary_min: float64
230 - prev_rate_interest_primary_count: float64
231 - prev_rate_interest_privileged_mean: float64
232 - prev_rate_interest_privileged_max: float64
233 - prev_rate_interest_privileged_min: float64
234 - prev_rate_interest_privileged_count: float64
235 - prev_days_decision_mean: float64
236 - prev_days_decision_max: float64
237 - prev_days_decision_min: float64
238 - prev_days_decision_count: float64
239 - prev_sellerplace_area_mean: float64
240 - prev_sellerplace_area_max: float64
241 - prev_sellerplace_area_min: float64
242 - prev_sellerplace_area_count: float64
243 - prev_cnt_payment_mean: float64
244 - prev_cnt_payment_max: float64
245 - prev_cnt_payment_min: float64
246 - prev_cnt_payment_count: float64
247 - prev_days_first_drawing_mean: float64
248 - prev_days_first_drawing_max: float64
249 - prev_days_first_drawing_min: float64
250 - prev_days_first_drawing_count: float64
251 - prev_days_first_due_mean: float64
252 - prev_days_first_due_max: float64
253 - prev_days_first_due_min: float64
254 - prev_days_first_due_count: float64
255 - prev_days_last_due_1st_version_mean: float64
256 - prev_days_last_due_1st_version_max: float64
257 - prev_days_last_due_1st_version_min: float64
258 - prev_days_last_due_1st_version_count: float64
259 - prev_days_last_due_mean: float64
260 - prev_days_last_due_max: float64
261 - prev_days_last_due_min: float64
262 - prev_days_last_due_count: float64
263 - prev_days_termination_mean: float64
264 - prev_days_termination_max: float64
265 - prev_days_termination_min: float64
266 - prev_days_termination_count: float64
267 - prev_nflag_insured_on_approval_mean: float64
268 - prev_nflag_insured_on_approval_max: float64
269 - prev_nflag_insured_on_approval_min: float64
270 - prev_nflag_insured_on_approval_count: float64
271 - posf_sk_id_prev_mean: float64
272 - posf_sk_id_prev_max: float64
273 - posf_sk_id_prev_min: float64
274 - posf_sk_id_prev_count: float64
275 - posf_pos_sk_id_curr_mean_mean: float64
276 - posf_pos_sk_id_curr_mean_max: float64
```

277 - posf_pos_sk_id_curr_mean_min: float64
278 - posf_pos_sk_id_curr_mean_count: float64
279 - posf_pos_sk_id_curr_max_mean: float64
280 - posf_pos_sk_id_curr_max_max: float64
281 - posf_pos_sk_id_curr_max_min: float64
282 - posf_pos_sk_id_curr_max_count: float64
283 - posf_pos_sk_id_curr_min_mean: float64
284 - posf_pos_sk_id_curr_min_max: float64
285 - posf_pos_sk_id_curr_min_min: float64
286 - posf_pos_sk_id_curr_min_count: float64
287 - posf_pos_sk_id_curr_count_mean: float64
288 - posf_pos_sk_id_curr_count_max: float64
289 - posf_pos_sk_id_curr_count_min: float64
290 - posf_pos_sk_id_curr_count_count: float64
291 - posf_pos_months_balance_mean_mean: float64
292 - posf_pos_months_balance_mean_max: float64
293 - posf_pos_months_balance_mean_min: float64
294 - posf_pos_months_balance_mean_count: float64
295 - posf_pos_months_balance_max_mean: float64
296 - posf_pos_months_balance_max_max: float64
297 - posf_pos_months_balance_max_min: float64
298 - posf_pos_months_balance_max_count: float64
299 - posf_pos_months_balance_min_mean: float64
300 - posf_pos_months_balance_min_max: float64
301 - posf_pos_months_balance_min_min: float64
302 - posf_pos_months_balance_min_count: float64
303 - posf_pos_months_balance_count_mean: float64
304 - posf_pos_months_balance_count_max: float64
305 - posf_pos_months_balance_count_min: float64
306 - posf_pos_months_balance_count_count: float64
307 - posf_pos_cnt_instalment_mean_mean: float64
308 - posf_pos_cnt_instalment_mean_max: float64
309 - posf_pos_cnt_instalment_mean_min: float64
310 - posf_pos_cnt_instalment_mean_count: float64
311 - posf_pos_cnt_instalment_max_mean: float64
312 - posf_pos_cnt_instalment_max_max: float64
313 - posf_pos_cnt_instalment_max_min: float64
314 - posf_pos_cnt_instalment_max_count: float64
315 - posf_pos_cnt_instalment_min_mean: float64
316 - posf_pos_cnt_instalment_min_max: float64
317 - posf_pos_cnt_instalment_min_min: float64
318 - posf_pos_cnt_instalment_min_count: float64
319 - posf_pos_cnt_instalment_count_mean: float64
320 - posf_pos_cnt_instalment_count_max: float64
321 - posf_pos_cnt_instalment_count_min: float64
322 - posf_pos_cnt_instalment_count_count: float64
323 - posf_pos_cnt_instalment_future_mean_mean: float64
324 - posf_pos_cnt_instalment_future_mean_max: float64
325 - posf_pos_cnt_instalment_future_mean_min: float64
326 - posf_pos_cnt_instalment_future_mean_count: float64
327 - posf_pos_cnt_instalment_future_max_mean: float64
328 - posf_pos_cnt_instalment_future_max_max: float64
329 - posf_pos_cnt_instalment_future_max_min: float64
330 - posf_pos_cnt_instalment_future_max_count: float64
331 - posf_pos_cnt_instalment_future_min_mean: float64
332 - posf_pos_cnt_instalment_future_min_max: float64
333 - posf_pos_cnt_instalment_future_min_min: float64
334 - posf_pos_cnt_instalment_future_min_count: float64
335 - posf_pos_cnt_instalment_future_count_mean: float64
336 - posf_pos_cnt_instalment_future_count_max: float64
337 - posf_pos_cnt_instalment_future_count_min: float64
338 - posf_pos_cnt_instalment_future_count_count: float64
339 - posf_pos_sk_dpd_mean_mean: float64
340 - posf_pos_sk_dpd_mean_max: float64
341 - posf_pos_sk_dpd_mean_min: float64
342 - posf_pos_sk_dpd_mean_count: float64
343 - posf_pos_sk_dpd_max_mean: float64
344 - posf_pos_sk_dpd_max_max: float64
345 - posf_pos_sk_dpd_max_min: float64

346 - posf_pos_sk_dpd_max_count: float64
347 - posf_pos_sk_dpd_min_mean: float64
348 - posf_pos_sk_dpd_min_max: float64
349 - posf_pos_sk_dpd_min_min: float64
350 - posf_pos_sk_dpd_min_count: float64
351 - posf_pos_sk_dpd_count_mean: float64
352 - posf_pos_sk_dpd_count_max: float64
353 - posf_pos_sk_dpd_count_min: float64
354 - posf_pos_sk_dpd_count_count: float64
355 - posf_pos_sk_dpd_def_mean_mean: float64
356 - posf_pos_sk_dpd_def_mean_max: float64
357 - posf_pos_sk_dpd_def_mean_min: float64
358 - posf_pos_sk_dpd_def_mean_count: float64
359 - posf_pos_sk_dpd_def_max_mean: float64
360 - posf_pos_sk_dpd_def_max_max: float64
361 - posf_pos_sk_dpd_def_max_min: float64
362 - posf_pos_sk_dpd_def_max_count: float64
363 - posf_pos_sk_dpd_def_min_mean: float64
364 - posf_pos_sk_dpd_def_min_max: float64
365 - posf_pos_sk_dpd_def_min_min: float64
366 - posf_pos_sk_dpd_def_min_count: float64
367 - posf_pos_sk_dpd_def_count_mean: float64
368 - posf_pos_sk_dpd_def_count_max: float64
369 - posf_pos_sk_dpd_def_count_min: float64
370 - posf_pos_sk_dpd_def_count_count: float64
371 - ccf_sk_id_prev_mean: float64
372 - ccf_sk_id_prev_max: float64
373 - ccf_sk_id_prev_min: float64
374 - ccf_sk_id_prev_count: float64
375 - ccf_cc_sk_id_curr_mean_mean: float64
376 - ccf_cc_sk_id_curr_mean_max: float64
377 - ccf_cc_sk_id_curr_mean_min: float64
378 - ccf_cc_sk_id_curr_mean_count: float64
379 - ccf_cc_sk_id_curr_max_mean: float64
380 - ccf_cc_sk_id_curr_max_max: float64
381 - ccf_cc_sk_id_curr_max_min: float64
382 - ccf_cc_sk_id_curr_max_count: float64
383 - ccf_cc_sk_id_curr_min_mean: float64
384 - ccf_cc_sk_id_curr_min_max: float64
385 - ccf_cc_sk_id_curr_min_min: float64
386 - ccf_cc_sk_id_curr_min_count: float64
387 - ccf_cc_sk_id_curr_count_mean: float64
388 - ccf_cc_sk_id_curr_count_max: float64
389 - ccf_cc_sk_id_curr_count_min: float64
390 - ccf_cc_sk_id_curr_count_count: float64
391 - ccf_cc_months_balance_mean_mean: float64
392 - ccf_cc_months_balance_mean_max: float64
393 - ccf_cc_months_balance_mean_min: float64
394 - ccf_cc_months_balance_mean_count: float64
395 - ccf_cc_months_balance_max_mean: float64
396 - ccf_cc_months_balance_max_max: float64
397 - ccf_cc_months_balance_max_min: float64
398 - ccf_cc_months_balance_max_count: float64
399 - ccf_cc_months_balance_min_mean: float64
400 - ccf_cc_months_balance_min_max: float64
401 - ccf_cc_months_balance_min_min: float64
402 - ccf_cc_months_balance_min_count: float64
403 - ccf_cc_months_balance_count_mean: float64
404 - ccf_cc_months_balance_count_max: float64
405 - ccf_cc_months_balance_count_min: float64
406 - ccf_cc_months_balance_count_count: float64
407 - ccf_cc_amt_balance_mean_mean: float64
408 - ccf_cc_amt_balance_mean_max: float64
409 - ccf_cc_amt_balance_mean_min: float64
410 - ccf_cc_amt_balance_mean_count: float64
411 - ccf_cc_amt_balance_max_mean: float64
412 - ccf_cc_amt_balance_max_max: float64
413 - ccf_cc_amt_balance_max_min: float64
414 - ccf_cc_amt_balance_max_count: float64

```
415 - ccf_cc_amt_balance_min_mean: float64
416 - ccf_cc_amt_balance_min_max: float64
417 - ccf_cc_amt_balance_min_min: float64
418 - ccf_cc_amt_balance_min_count: float64
419 - ccf_cc_amt_balance_count_mean: float64
420 - ccf_cc_amt_balance_count_max: float64
421 - ccf_cc_amt_balance_count_min: float64
422 - ccf_cc_amt_balance_count_count: float64
423 - ccf_cc_amt_credit_limit_actual_mean_mean: float64
424 - ccf_cc_amt_credit_limit_actual_mean_max: float64
425 - ccf_cc_amt_credit_limit_actual_mean_min: float64
426 - ccf_cc_amt_credit_limit_actual_mean_count: float64
427 - ccf_cc_amt_credit_limit_actual_max_mean: float64
428 - ccf_cc_amt_credit_limit_actual_max_max: float64
429 - ccf_cc_amt_credit_limit_actual_max_min: float64
430 - ccf_cc_amt_credit_limit_actual_max_count: float64
431 - ccf_cc_amt_credit_limit_actual_min_mean: float64
432 - ccf_cc_amt_credit_limit_actual_min_max: float64
433 - ccf_cc_amt_credit_limit_actual_min_min: float64
434 - ccf_cc_amt_credit_limit_actual_min_count: float64
435 - ccf_cc_amt_credit_limit_actual_count_mean: float64
436 - ccf_cc_amt_credit_limit_actual_count_max: float64
437 - ccf_cc_amt_credit_limit_actual_count_min: float64
438 - ccf_cc_amt_credit_limit_actual_count_count: float64
439 - ccf_cc_amt_drawings_atm_current_mean_mean: float64
440 - ccf_cc_amt_drawings_atm_current_mean_max: float64
441 - ccf_cc_amt_drawings_atm_current_mean_min: float64
442 - ccf_cc_amt_drawings_atm_current_mean_count: float64
443 - ccf_cc_amt_drawings_atm_current_max_mean: float64
444 - ccf_cc_amt_drawings_atm_current_max_max: float64
445 - ccf_cc_amt_drawings_atm_current_max_min: float64
446 - ccf_cc_amt_drawings_atm_current_max_count: float64
447 - ccf_cc_amt_drawings_atm_current_min_mean: float64
448 - ccf_cc_amt_drawings_atm_current_min_max: float64
449 - ccf_cc_amt_drawings_atm_current_min_min: float64
450 - ccf_cc_amt_drawings_atm_current_min_count: float64
451 - ccf_cc_amt_drawings_atm_current_count_mean: float64
452 - ccf_cc_amt_drawings_atm_current_count_max: float64
453 - ccf_cc_amt_drawings_atm_current_count_min: float64
454 - ccf_cc_amt_drawings_atm_current_count_count: float64
455 - ccf_cc_amt_drawings_current_mean_mean: float64
456 - ccf_cc_amt_drawings_current_mean_max: float64
457 - ccf_cc_amt_drawings_current_mean_min: float64
458 - ccf_cc_amt_drawings_current_mean_count: float64
459 - ccf_cc_amt_drawings_current_max_mean: float64
460 - ccf_cc_amt_drawings_current_max_max: float64
461 - ccf_cc_amt_drawings_current_max_min: float64
462 - ccf_cc_amt_drawings_current_max_count: float64
463 - ccf_cc_amt_drawings_current_min_mean: float64
464 - ccf_cc_amt_drawings_current_min_max: float64
465 - ccf_cc_amt_drawings_current_min_min: float64
466 - ccf_cc_amt_drawings_current_min_count: float64
467 - ccf_cc_amt_drawings_current_count_mean: float64
468 - ccf_cc_amt_drawings_current_count_max: float64
469 - ccf_cc_amt_drawings_current_count_min: float64
470 - ccf_cc_amt_drawings_current_count_count: float64
471 - ccf_cc_amt_drawings_other_current_mean_mean: float64
472 - ccf_cc_amt_drawings_other_current_mean_max: float64
473 - ccf_cc_amt_drawings_other_current_mean_min: float64
474 - ccf_cc_amt_drawings_other_current_mean_count: float64
475 - ccf_cc_amt_drawings_other_current_max_mean: float64
476 - ccf_cc_amt_drawings_other_current_max_max: float64
477 - ccf_cc_amt_drawings_other_current_max_min: float64
478 - ccf_cc_amt_drawings_other_current_max_count: float64
479 - ccf_cc_amt_drawings_other_current_min_mean: float64
480 - ccf_cc_amt_drawings_other_current_min_max: float64
481 - ccf_cc_amt_drawings_other_current_min_min: float64
482 - ccf_cc_amt_drawings_other_current_min_count: float64
483 - ccf_cc_amt_drawings_other_current_count_mean: float64
```

484 - ccf_cc_amt_drawings_other_current_count_max: float64
485 - ccf_cc_amt_drawings_other_current_count_min: float64
486 - ccf_cc_amt_drawings_other_current_count_count: float64
487 - ccf_cc_amt_drawings_pos_current_mean_mean: float64
488 - ccf_cc_amt_drawings_pos_current_mean_max: float64
489 - ccf_cc_amt_drawings_pos_current_mean_min: float64
490 - ccf_cc_amt_drawings_pos_current_mean_count: float64
491 - ccf_cc_amt_drawings_pos_current_max_mean: float64
492 - ccf_cc_amt_drawings_pos_current_max_max: float64
493 - ccf_cc_amt_drawings_pos_current_max_min: float64
494 - ccf_cc_amt_drawings_pos_current_max_count: float64
495 - ccf_cc_amt_drawings_pos_current_min_mean: float64
496 - ccf_cc_amt_drawings_pos_current_min_max: float64
497 - ccf_cc_amt_drawings_pos_current_min_min: float64
498 - ccf_cc_amt_drawings_pos_current_min_count: float64
499 - ccf_cc_amt_drawings_pos_current_count_mean: float64
500 - ccf_cc_amt_drawings_pos_current_count_max: float64
501 - ccf_cc_amt_drawings_pos_current_count_min: float64
502 - ccf_cc_amt_drawings_pos_current_count_count: float64
503 - ccf_cc_inst_min_regularity_mean_mean: float64
504 - ccf_cc_inst_min_regularity_mean_max: float64
505 - ccf_cc_inst_min_regularity_mean_min: float64
506 - ccf_cc_inst_min_regularity_mean_count: float64
507 - ccf_cc_inst_min_regularity_max_mean: float64
508 - ccf_cc_inst_min_regularity_max_max: float64
509 - ccf_cc_inst_min_regularity_max_min: float64
510 - ccf_cc_inst_min_regularity_max_count: float64
511 - ccf_cc_inst_min_regularity_min_mean: float64
512 - ccf_cc_inst_min_regularity_min_max: float64
513 - ccf_cc_inst_min_regularity_min_min: float64
514 - ccf_cc_inst_min_regularity_min_count: float64
515 - ccf_cc_inst_min_regularity_count_mean: float64
516 - ccf_cc_inst_min_regularity_count_max: float64
517 - ccf_cc_inst_min_regularity_count_min: float64
518 - ccf_cc_inst_min_regularity_count_count: float64
519 - ccf_cc_payment_current_mean_mean: float64
520 - ccf_cc_payment_current_mean_max: float64
521 - ccf_cc_payment_current_mean_min: float64
522 - ccf_cc_payment_current_mean_count: float64
523 - ccf_cc_payment_current_max_mean: float64
524 - ccf_cc_payment_current_max_max: float64
525 - ccf_cc_payment_current_max_min: float64
526 - ccf_cc_payment_current_max_count: float64
527 - ccf_cc_payment_current_min_mean: float64
528 - ccf_cc_payment_current_min_max: float64
529 - ccf_cc_payment_current_min_min: float64
530 - ccf_cc_payment_current_min_count: float64
531 - ccf_cc_payment_current_count_mean: float64
532 - ccf_cc_payment_current_count_max: float64
533 - ccf_cc_payment_current_count_min: float64
534 - ccf_cc_payment_current_count_count: float64
535 - ccf_cc_payment_total_current_mean_mean: float64
536 - ccf_cc_payment_total_current_mean_max: float64
537 - ccf_cc_payment_total_current_mean_min: float64
538 - ccf_cc_payment_total_current_mean_count: float64
539 - ccf_cc_payment_total_current_max_mean: float64
540 - ccf_cc_payment_total_current_max_max: float64
541 - ccf_cc_payment_total_current_max_min: float64
542 - ccf_cc_payment_total_current_max_count: float64
543 - ccf_cc_payment_total_current_min_mean: float64
544 - ccf_cc_payment_total_current_min_max: float64
545 - ccf_cc_payment_total_current_min_min: float64
546 - ccf_cc_payment_total_current_min_count: float64
547 - ccf_cc_payment_total_current_count_mean: float64
548 - ccf_cc_payment_total_current_count_max: float64
549 - ccf_cc_payment_total_current_count_min: float64
550 - ccf_cc_payment_total_current_count_count: float64
551 - ccf_cc_receivable_principal_mean_mean: float64
552 - ccf_cc_receivable_principal_mean_max: float64

553 - ccf_cc_amt_receivable_principal_mean_min: float64
554 - ccf_cc_amt_receivable_principal_mean_count: float64
555 - ccf_cc_amt_receivable_principal_max_mean: float64
556 - ccf_cc_amt_receivable_principal_max_max: float64
557 - ccf_cc_amt_receivable_principal_max_min: float64
558 - ccf_cc_amt_receivable_principal_max_count: float64
559 - ccf_cc_amt_receivable_principal_min_mean: float64
560 - ccf_cc_amt_receivable_principal_min_max: float64
561 - ccf_cc_amt_receivable_principal_min_min: float64
562 - ccf_cc_amt_receivable_principal_min_count: float64
563 - ccf_cc_amt_receivable_principal_count_mean: float64
564 - ccf_cc_amt_receivable_principal_count_max: float64
565 - ccf_cc_amt_receivable_principal_count_min: float64
566 - ccf_cc_amt_receivable_principal_count_count: float64
567 - ccf_cc_amt_recivable_mean_mean: float64
568 - ccf_cc_amt_recivable_mean_max: float64
569 - ccf_cc_amt_recivable_mean_min: float64
570 - ccf_cc_amt_recivable_mean_count: float64
571 - ccf_cc_amt_recivable_max_mean: float64
572 - ccf_cc_amt_recivable_max_max: float64
573 - ccf_cc_amt_recivable_max_min: float64
574 - ccf_cc_amt_recivable_max_count: float64
575 - ccf_cc_amt_recivable_min_mean: float64
576 - ccf_cc_amt_recivable_min_max: float64
577 - ccf_cc_amt_recivable_min_min: float64
578 - ccf_cc_amt_recivable_min_count: float64
579 - ccf_cc_amt_recivable_count_mean: float64
580 - ccf_cc_amt_recivable_count_max: float64
581 - ccf_cc_amt_recivable_count_min: float64
582 - ccf_cc_amt_recivable_count_count: float64
583 - ccf_cc_amt_total_receivable_mean_mean: float64
584 - ccf_cc_amt_total_receivable_mean_max: float64
585 - ccf_cc_amt_total_receivable_mean_min: float64
586 - ccf_cc_amt_total_receivable_mean_count: float64
587 - ccf_cc_amt_total_receivable_max_mean: float64
588 - ccf_cc_amt_total_receivable_max_max: float64
589 - ccf_cc_amt_total_receivable_max_min: float64
590 - ccf_cc_amt_total_receivable_max_count: float64
591 - ccf_cc_amt_total_receivable_min_mean: float64
592 - ccf_cc_amt_total_receivable_min_max: float64
593 - ccf_cc_amt_total_receivable_min_min: float64
594 - ccf_cc_amt_total_receivable_min_count: float64
595 - ccf_cc_amt_total_receivable_count_mean: float64
596 - ccf_cc_amt_total_receivable_count_max: float64
597 - ccf_cc_amt_total_receivable_count_min: float64
598 - ccf_cc_amt_total_receivable_count_count: float64
599 - ccf_cc_cnt_drawings_atm_current_mean_mean: float64
600 - ccf_cc_cnt_drawings_atm_current_mean_max: float64
601 - ccf_cc_cnt_drawings_atm_current_mean_min: float64
602 - ccf_cc_cnt_drawings_atm_current_mean_count: float64
603 - ccf_cc_cnt_drawings_atm_current_max_mean: float64
604 - ccf_cc_cnt_drawings_atm_current_max_max: float64
605 - ccf_cc_cnt_drawings_atm_current_max_min: float64
606 - ccf_cc_cnt_drawings_atm_current_max_count: float64
607 - ccf_cc_cnt_drawings_atm_current_min_mean: float64
608 - ccf_cc_cnt_drawings_atm_current_min_max: float64
609 - ccf_cc_cnt_drawings_atm_current_min_min: float64
610 - ccf_cc_cnt_drawings_atm_current_min_count: float64
611 - ccf_cc_cnt_drawings_atm_current_count_mean: float64
612 - ccf_cc_cnt_drawings_atm_current_count_max: float64
613 - ccf_cc_cnt_drawings_atm_current_count_min: float64
614 - ccf_cc_cnt_drawings_atm_current_count_count: float64
615 - ccf_cc_cnt_drawings_current_mean_mean: float64
616 - ccf_cc_cnt_drawings_current_mean_max: float64
617 - ccf_cc_cnt_drawings_current_mean_min: float64
618 - ccf_cc_cnt_drawings_current_mean_count: float64
619 - ccf_cc_cnt_drawings_current_max_mean: float64
620 - ccf_cc_cnt_drawings_current_max_max: float64
621 - ccf_cc_cnt_drawings_current_max_min: float64

622 - ccf_cc_cnt_drawings_current_max_count: float64
623 - ccf_cc_cnt_drawings_current_min_mean: float64
624 - ccf_cc_cnt_drawings_current_min_max: float64
625 - ccf_cc_cnt_drawings_current_min_min: float64
626 - ccf_cc_cnt_drawings_current_min_count: float64
627 - ccf_cc_cnt_drawings_current_count_mean: float64
628 - ccf_cc_cnt_drawings_current_count_max: float64
629 - ccf_cc_cnt_drawings_current_count_min: float64
630 - ccf_cc_cnt_drawings_current_count_count: float64
631 - ccf_cc_cnt_drawings_other_current_mean_mean: float64
632 - ccf_cc_cnt_drawings_other_current_mean_max: float64
633 - ccf_cc_cnt_drawings_other_current_mean_min: float64
634 - ccf_cc_cnt_drawings_other_current_mean_count: float64
635 - ccf_cc_cnt_drawings_other_current_max_mean: float64
636 - ccf_cc_cnt_drawings_other_current_max_max: float64
637 - ccf_cc_cnt_drawings_other_current_max_min: float64
638 - ccf_cc_cnt_drawings_other_current_max_count: float64
639 - ccf_cc_cnt_drawings_other_current_min_mean: float64
640 - ccf_cc_cnt_drawings_other_current_min_max: float64
641 - ccf_cc_cnt_drawings_other_current_min_min: float64
642 - ccf_cc_cnt_drawings_other_current_min_count: float64
643 - ccf_cc_cnt_drawings_other_current_count_mean: float64
644 - ccf_cc_cnt_drawings_other_current_count_max: float64
645 - ccf_cc_cnt_drawings_other_current_count_min: float64
646 - ccf_cc_cnt_drawings_other_current_count_count: float64
647 - ccf_cc_cnt_drawings_pos_current_mean_mean: float64
648 - ccf_cc_cnt_drawings_pos_current_mean_max: float64
649 - ccf_cc_cnt_drawings_pos_current_mean_min: float64
650 - ccf_cc_cnt_drawings_pos_current_mean_count: float64
651 - ccf_cc_cnt_drawings_pos_current_max_mean: float64
652 - ccf_cc_cnt_drawings_pos_current_max_max: float64
653 - ccf_cc_cnt_drawings_pos_current_max_min: float64
654 - ccf_cc_cnt_drawings_pos_current_max_count: float64
655 - ccf_cc_cnt_drawings_pos_current_min_mean: float64
656 - ccf_cc_cnt_drawings_pos_current_min_max: float64
657 - ccf_cc_cnt_drawings_pos_current_min_min: float64
658 - ccf_cc_cnt_drawings_pos_current_min_count: float64
659 - ccf_cc_cnt_drawings_pos_current_count_mean: float64
660 - ccf_cc_cnt_drawings_pos_current_count_max: float64
661 - ccf_cc_cnt_drawings_pos_current_count_min: float64
662 - ccf_cc_cnt_drawings_pos_current_count_count: float64
663 - ccf_cc_cnt_instalment_mature_cum_mean_mean: float64
664 - ccf_cc_cnt_instalment_mature_cum_mean_max: float64
665 - ccf_cc_cnt_instalment_mature_cum_mean_min: float64
666 - ccf_cc_cnt_instalment_mature_cum_mean_count: float64
667 - ccf_cc_cnt_instalment_mature_cum_max_mean: float64
668 - ccf_cc_cnt_instalment_mature_cum_max_max: float64
669 - ccf_cc_cnt_instalment_mature_cum_max_min: float64
670 - ccf_cc_cnt_instalment_mature_cum_max_count: float64
671 - ccf_cc_cnt_instalment_mature_cum_min_mean: float64
672 - ccf_cc_cnt_instalment_mature_cum_min_max: float64
673 - ccf_cc_cnt_instalment_mature_cum_min_min: float64
674 - ccf_cc_cnt_instalment_mature_cum_min_count: float64
675 - ccf_cc_cnt_instalment_mature_cum_count_mean: float64
676 - ccf_cc_cnt_instalment_mature_cum_count_max: float64
677 - ccf_cc_cnt_instalment_mature_cum_count_min: float64
678 - ccf_cc_cnt_instalment_mature_cum_count_count: float64
679 - ccf_cc_sk_dpd_mean_mean: float64
680 - ccf_cc_sk_dpd_mean_max: float64
681 - ccf_cc_sk_dpd_mean_min: float64
682 - ccf_cc_sk_dpd_mean_count: float64
683 - ccf_cc_sk_dpd_max_mean: float64
684 - ccf_cc_sk_dpd_max_max: float64
685 - ccf_cc_sk_dpd_max_min: float64
686 - ccf_cc_sk_dpd_max_count: float64
687 - ccf_cc_sk_dpd_min_mean: float64
688 - ccf_cc_sk_dpd_min_max: float64
689 - ccf_cc_sk_dpd_min_min: float64
690 - ccf_cc_sk_dpd_min_count: float64

691 - ccf_cc_sk_dpd_count_mean: float64
692 - ccf_cc_sk_dpd_count_max: float64
693 - ccf_cc_sk_dpd_count_min: float64
694 - ccf_cc_sk_dpd_count_count: float64
695 - ccf_cc_sk_dpd_def_mean_mean: float64
696 - ccf_cc_sk_dpd_def_mean_max: float64
697 - ccf_cc_sk_dpd_def_mean_min: float64
698 - ccf_cc_sk_dpd_def_mean_count: float64
699 - ccf_cc_sk_dpd_def_max_mean: float64
700 - ccf_cc_sk_dpd_def_max_max: float64
701 - ccf_cc_sk_dpd_def_max_min: float64
702 - ccf_cc_sk_dpd_def_max_count: float64
703 - ccf_cc_sk_dpd_def_min_mean: float64
704 - ccf_cc_sk_dpd_def_min_max: float64
705 - ccf_cc_sk_dpd_def_min_min: float64
706 - ccf_cc_sk_dpd_def_min_count: float64
707 - ccf_cc_sk_dpd_def_count_mean: float64
708 - ccf_cc_sk_dpd_def_count_max: float64
709 - ccf_cc_sk_dpd_def_count_min: float64
710 - ccf_cc_sk_dpd_def_count_count: float64
711 - instf_sk_id_prev_mean: float64
712 - instf_sk_id_prev_max: float64
713 - instf_sk_id_prev_min: float64
714 - instf_sk_id_prev_count: float64
715 - instf_inst_sk_id_curr_mean_mean: float64
716 - instf_inst_sk_id_curr_mean_max: float64
717 - instf_inst_sk_id_curr_mean_min: float64
718 - instf_inst_sk_id_curr_mean_count: float64
719 - instf_inst_sk_id_curr_max_mean: float64
720 - instf_inst_sk_id_curr_max_max: float64
721 - instf_inst_sk_id_curr_max_min: float64
722 - instf_inst_sk_id_curr_max_count: float64
723 - instf_inst_sk_id_curr_min_mean: float64
724 - instf_inst_sk_id_curr_min_max: float64
725 - instf_inst_sk_id_curr_min_min: float64
726 - instf_inst_sk_id_curr_min_count: float64
727 - instf_inst_sk_id_curr_count_mean: float64
728 - instf_inst_sk_id_curr_count_max: float64
729 - instf_inst_sk_id_curr_count_min: float64
730 - instf_inst_sk_id_curr_count_count: float64
731 - instf_inst_num_instalment_version_mean_mean: float64
732 - instf_inst_num_instalment_version_mean_max: float64
733 - instf_inst_num_instalment_version_mean_min: float64
734 - instf_inst_num_instalment_version_mean_count: float64
735 - instf_inst_num_instalment_version_max_mean: float64
736 - instf_inst_num_instalment_version_max_max: float64
737 - instf_inst_num_instalment_version_max_min: float64
738 - instf_inst_num_instalment_version_max_count: float64
739 - instf_inst_num_instalment_version_min_mean: float64
740 - instf_inst_num_instalment_version_min_max: float64
741 - instf_inst_num_instalment_version_min_min: float64
742 - instf_inst_num_instalment_version_min_count: float64
743 - instf_inst_num_instalment_version_count_mean: float64
744 - instf_inst_num_instalment_version_count_max: float64
745 - instf_inst_num_instalment_version_count_min: float64
746 - instf_inst_num_instalment_version_count_count: float64
747 - instf_inst_num_instalment_number_mean_mean: float64
748 - instf_inst_num_instalment_number_mean_max: float64
749 - instf_inst_num_instalment_number_mean_min: float64
750 - instf_inst_num_instalment_number_mean_count: float64
751 - instf_inst_num_instalment_number_max_mean: float64
752 - instf_inst_num_instalment_number_max_max: float64
753 - instf_inst_num_instalment_number_max_min: float64
754 - instf_inst_num_instalment_number_max_count: float64
755 - instf_inst_num_instalment_number_min_mean: float64
756 - instf_inst_num_instalment_number_min_max: float64
757 - instf_inst_num_instalment_number_min_min: float64
758 - instf_inst_num_instalment_number_min_count: float64
759 - instf_inst_num_instalment_number_count_mean: float64

```
760 - instf_inst_num_instalment_number_count_max: float64
761 - instf_inst_num_instalment_number_count_min: float64
762 - instf_inst_num_instalment_number_count_count: float64
763 - instf_inst_days_instalment_mean_mean: float64
764 - instf_inst_days_instalment_mean_max: float64
765 - instf_inst_days_instalment_mean_min: float64
766 - instf_inst_days_instalment_mean_count: float64
767 - instf_inst_days_instalment_max_mean: float64
768 - instf_inst_days_instalment_max_max: float64
769 - instf_inst_days_instalment_max_min: float64
770 - instf_inst_days_instalment_max_count: float64
771 - instf_inst_days_instalment_min_mean: float64
772 - instf_inst_days_instalment_min_max: float64
773 - instf_inst_days_instalment_min_min: float64
774 - instf_inst_days_instalment_min_count: float64
775 - instf_inst_days_instalment_count_mean: float64
776 - instf_inst_days_instalment_count_max: float64
777 - instf_inst_days_instalment_count_min: float64
778 - instf_inst_days_instalment_count_count: float64
779 - instf_inst_days_entry_payment_mean_mean: float64
780 - instf_inst_days_entry_payment_mean_max: float64
781 - instf_inst_days_entry_payment_mean_min: float64
782 - instf_inst_days_entry_payment_mean_count: float64
783 - instf_inst_days_entry_payment_max_mean: float64
784 - instf_inst_days_entry_payment_max_max: float64
785 - instf_inst_days_entry_payment_max_min: float64
786 - instf_inst_days_entry_payment_max_count: float64
787 - instf_inst_days_entry_payment_min_mean: float64
788 - instf_inst_days_entry_payment_min_max: float64
789 - instf_inst_days_entry_payment_min_min: float64
790 - instf_inst_days_entry_payment_min_count: float64
791 - instf_inst_days_entry_payment_count_mean: float64
792 - instf_inst_days_entry_payment_count_max: float64
793 - instf_inst_days_entry_payment_count_min: float64
794 - instf_inst_days_entry_payment_count_count: float64
795 - instf_inst_amt_instalment_mean_mean: float64
796 - instf_inst_amt_instalment_mean_max: float64
797 - instf_inst_amt_instalment_mean_min: float64
798 - instf_inst_amt_instalment_mean_count: float64
799 - instf_inst_amt_instalment_max_mean: float64
800 - instf_inst_amt_instalment_max_max: float64
801 - instf_inst_amt_instalment_max_min: float64
802 - instf_inst_amt_instalment_max_count: float64
803 - instf_inst_amt_instalment_min_mean: float64
804 - instf_inst_amt_instalment_min_max: float64
805 - instf_inst_amt_instalment_min_min: float64
806 - instf_inst_amt_instalment_min_count: float64
807 - instf_inst_amt_instalment_count_mean: float64
808 - instf_inst_amt_instalment_count_max: float64
809 - instf_inst_amt_instalment_count_min: float64
810 - instf_inst_amt_instalment_count_count: float64
811 - instf_inst_amt_payment_mean_mean: float64
812 - instf_inst_amt_payment_mean_max: float64
813 - instf_inst_amt_payment_mean_min: float64
814 - instf_inst_amt_payment_mean_count: float64
815 - instf_inst_amt_payment_max_mean: float64
816 - instf_inst_amt_payment_max_max: float64
817 - instf_inst_amt_payment_max_min: float64
818 - instf_inst_amt_payment_max_count: float64
819 - instf_inst_amt_payment_min_mean: float64
820 - instf_inst_amt_payment_min_max: float64
821 - instf_inst_amt_payment_min_min: float64
822 - instf_inst_amt_payment_min_count: float64
823 - instf_inst_amt_payment_count_mean: float64
824 - instf_inst_amt_payment_count_max: float64
825 - instf_inst_amt_payment_count_min: float64
826 - instf_inst_amt_payment_count_count: float64
```

```
In [6]: df1 = df1.drop(columns=['sk_id_curr'])
```

5.2 Dimensão dos Dados

```
In [7]: print('Número de linhas:', df1.shape[0])
print('Número de colunas:', df1.shape[1])
```

Número de linhas: 307511
Número de colunas: 825

5.3 Tipos dos dados

```
In [8]: for idx, (col, dtype) in enumerate(df1.dtypes.items(), 1):
    print(f'{idx} - {col}: {dtype}')
```

```
1 - target: int64
2 - name_contract_type: object
3 - code_gender: object
4 - flag_own_car: object
5 - flag_own_realty: object
6 - cnt_children: int64
7 - amt_income_total: float64
8 - amt_credit: float64
9 - amt_annuity: float64
10 - amt_goods_price: float64
11 - name_type_suite: object
12 - name_income_type: object
13 - name_education_type: object
14 - name_family_status: object
15 - name_housing_type: object
16 - region_population_relative: float64
17 - days_birth: int64
18 - days_employed: int64
19 - days_registration: float64
20 - days_id_publish: int64
21 - own_car_age: float64
22 - flag_mobil: int64
23 - flag_emp_phone: int64
24 - flag_work_phone: int64
25 - flag_cont_mobile: int64
26 - flag_phone: int64
27 - flag_email: int64
28 - occupation_type: object
29 - cnt_fam_members: float64
30 - region_rating_client: int64
31 - region_rating_client_w_city: int64
32 - weekday_appr_process_start: object
33 - hour_appr_process_start: int64
34 - reg_region_not_live_region: int64
35 - reg_region_not_work_region: int64
36 - live_region_not_work_region: int64
37 - reg_city_not_live_city: int64
38 - reg_city_not_work_city: int64
39 - live_city_not_work_city: int64
40 - organization_type: object
41 - ext_source_1: float64
42 - ext_source_2: float64
43 - ext_source_3: float64
44 - apartments_avg: float64
45 - basementarea_avg: float64
46 - years_beginexpluatation_avg: float64
47 - years_build_avg: float64
48 - commonarea_avg: float64
49 - elevators_avg: float64
50 - entrances_avg: float64
51 - floorsmax_avg: float64
52 - floorsmin_avg: float64
53 - landarea_avg: float64
54 - livingapartments_avg: float64
55 - livingarea_avg: float64
56 - nonlivingapartments_avg: float64
57 - nonlivingarea_avg: float64
58 - apartments_mode: float64
59 - basementarea_mode: float64
60 - years_beginexpluatation_mode: float64
61 - years_build_mode: float64
62 - commonarea_mode: float64
63 - elevators_mode: float64
64 - entrances_mode: float64
65 - floorsmax_mode: float64
66 - floorsmin_mode: float64
67 - landarea_mode: float64
68 - livingapartments_mode: float64
69 - livingarea_mode: float64
```

70 - nonlivingapartments_mode: float64
71 - nonlivingarea_mode: float64
72 - apartments_medi: float64
73 - basementarea_medi: float64
74 - years_beginexpluatation_medi: float64
75 - years_build_medi: float64
76 - commonarea_medi: float64
77 - elevators_medi: float64
78 - entrances_medi: float64
79 - floorsmax_medi: float64
80 - floorsmin_medi: float64
81 - landarea_medi: float64
82 - livingapartments_medi: float64
83 - livingarea_medi: float64
84 - nonlivingapartments_medi: float64
85 - nonlivingarea_medi: float64
86 - fondkapremont_mode: object
87 - housetype_mode: object
88 - totalarea_mode: float64
89 - wallsmaterial_mode: object
90 - emergencystate_mode: object
91 - obs_30_cnt_social_circle: float64
92 - def_30_cnt_social_circle: float64
93 - obs_60_cnt_social_circle: float64
94 - def_60_cnt_social_circle: float64
95 - days_last_phone_change: float64
96 - flag_document_2: int64
97 - flag_document_3: int64
98 - flag_document_4: int64
99 - flag_document_5: int64
100 - flag_document_6: int64
101 - flag_document_7: int64
102 - flag_document_8: int64
103 - flag_document_9: int64
104 - flag_document_10: int64
105 - flag_document_11: int64
106 - flag_document_12: int64
107 - flag_document_13: int64
108 - flag_document_14: int64
109 - flag_document_15: int64
110 - flag_document_16: int64
111 - flag_document_17: int64
112 - flag_document_18: int64
113 - flag_document_19: int64
114 - flag_document_20: int64
115 - flag_document_21: int64
116 - amt_req_credit_bureau_hour: float64
117 - amt_req_credit_bureau_day: float64
118 - amt_req_credit_bureau_week: float64
119 - amt_req_credit_bureau_mon: float64
120 - amt_req_credit_bureau_qrt: float64
121 - amt_req_credit_bureau_year: float64
122 - bur_sk_id_bureau_mean: float64
123 - bur_sk_id_bureau_max: float64
124 - bur_sk_id_bureau_min: float64
125 - bur_sk_id_bureau_count: float64
126 - bur_days_credit_mean: float64
127 - bur_days_credit_max: float64
128 - bur_days_credit_min: float64
129 - bur_days_credit_count: float64
130 - bur_credit_day_overdue_mean: float64
131 - bur_credit_day_overdue_max: float64
132 - bur_credit_day_overdue_min: float64
133 - bur_credit_day_overdue_count: float64
134 - bur_days_credit_enddate_mean: float64
135 - bur_days_credit_enddate_max: float64
136 - bur_days_credit_enddate_min: float64
137 - bur_days_credit_enddate_count: float64
138 - bur_days_enddate_fact_mean: float64

139 - bur_days_enddate_fact_max: float64
140 - bur_days_enddate_fact_min: float64
141 - bur_days_enddate_fact_count: float64
142 - bur_amt_credit_max_overdue_mean: float64
143 - bur_amt_credit_max_overdue_max: float64
144 - bur_amt_credit_max_overdue_min: float64
145 - bur_amt_credit_max_overdue_count: float64
146 - bur_cnt_credit_prolong_mean: float64
147 - bur_cnt_credit_prolong_max: float64
148 - bur_cnt_credit_prolong_min: float64
149 - bur_cnt_credit_prolong_count: float64
150 - bur_amt_credit_sum_mean: float64
151 - bur_amt_credit_sum_max: float64
152 - bur_amt_credit_sum_min: float64
153 - bur_amt_credit_sum_count: float64
154 - bur_amt_credit_sum_debt_mean: float64
155 - bur_amt_credit_sum_debt_max: float64
156 - bur_amt_credit_sum_debt_min: float64
157 - bur_amt_credit_sum_debt_count: float64
158 - bur_amt_credit_sum_limit_mean: float64
159 - bur_amt_credit_sum_limit_max: float64
160 - bur_amt_credit_sum_limit_min: float64
161 - bur_amt_credit_sum_limit_count: float64
162 - bur_amt_credit_sum_overdue_mean: float64
163 - bur_amt_credit_sum_overdue_max: float64
164 - bur_amt_credit_sum_overdue_min: float64
165 - bur_amt_credit_sum_overdue_count: float64
166 - bur_days_credit_update_mean: float64
167 - bur_days_credit_update_max: float64
168 - bur_days_credit_update_min: float64
169 - bur_days_credit_update_count: float64
170 - bur_amt_annuity_mean: float64
171 - bur_amt_annuity_max: float64
172 - bur_amt_annuity_min: float64
173 - bur_amt_annuity_count: float64
174 - bur_bb_months_balance_mean_mean: float64
175 - bur_bb_months_balance_mean_max: float64
176 - bur_bb_months_balance_mean_min: float64
177 - bur_bb_months_balance_mean_count: float64
178 - bur_bb_months_balance_max_mean: float64
179 - bur_bb_months_balance_max_max: float64
180 - bur_bb_months_balance_max_min: float64
181 - bur_bb_months_balance_max_count: float64
182 - bur_bb_months_balance_min_mean: float64
183 - bur_bb_months_balance_min_max: float64
184 - bur_bb_months_balance_min_min: float64
185 - bur_bb_months_balance_min_count: float64
186 - bur_bb_months_balance_count_mean: float64
187 - bur_bb_months_balance_count_max: float64
188 - bur_bb_months_balance_count_min: float64
189 - bur_bb_months_balance_count_count: float64
190 - prev_sk_id_prev_mean: float64
191 - prev_sk_id_prev_max: float64
192 - prev_sk_id_prev_min: float64
193 - prev_sk_id_prev_count: float64
194 - prev_amt_annuity_mean: float64
195 - prev_amt_annuity_max: float64
196 - prev_amt_annuity_min: float64
197 - prev_amt_annuity_count: float64
198 - prev_amt_application_mean: float64
199 - prev_amt_application_max: float64
200 - prev_amt_application_min: float64
201 - prev_amt_application_count: float64
202 - prev_amt_credit_mean: float64
203 - prev_amt_credit_max: float64
204 - prev_amt_credit_min: float64
205 - prev_amt_credit_count: float64
206 - prev_amt_down_payment_mean: float64
207 - prev_amt_down_payment_max: float64

```
208 - prev_amt_down_payment_min: float64
209 - prev_amt_down_payment_count: float64
210 - prev_amt_goods_price_mean: float64
211 - prev_amt_goods_price_max: float64
212 - prev_amt_goods_price_min: float64
213 - prev_amt_goods_price_count: float64
214 - prev_hour_appr_process_start_mean: float64
215 - prev_hour_appr_process_start_max: float64
216 - prev_hour_appr_process_start_min: float64
217 - prev_hour_appr_process_start_count: float64
218 - prev_nflag_last_appl_in_day_mean: float64
219 - prev_nflag_last_appl_in_day_max: float64
220 - prev_nflag_last_appl_in_day_min: float64
221 - prev_nflag_last_appl_in_day_count: float64
222 - prev_rate_down_payment_mean: float64
223 - prev_rate_down_payment_max: float64
224 - prev_rate_down_payment_min: float64
225 - prev_rate_down_payment_count: float64
226 - prev_rate_interest_primary_mean: float64
227 - prev_rate_interest_primary_max: float64
228 - prev_rate_interest_primary_min: float64
229 - prev_rate_interest_primary_count: float64
230 - prev_rate_interest_privileged_mean: float64
231 - prev_rate_interest_privileged_max: float64
232 - prev_rate_interest_privileged_min: float64
233 - prev_rate_interest_privileged_count: float64
234 - prev_days_decision_mean: float64
235 - prev_days_decision_max: float64
236 - prev_days_decision_min: float64
237 - prev_days_decision_count: float64
238 - prev_sellerplace_area_mean: float64
239 - prev_sellerplace_area_max: float64
240 - prev_sellerplace_area_min: float64
241 - prev_sellerplace_area_count: float64
242 - prev_cnt_payment_mean: float64
243 - prev_cnt_payment_max: float64
244 - prev_cnt_payment_min: float64
245 - prev_cnt_payment_count: float64
246 - prev_days_first_drawing_mean: float64
247 - prev_days_first_drawing_max: float64
248 - prev_days_first_drawing_min: float64
249 - prev_days_first_drawing_count: float64
250 - prev_days_first_due_mean: float64
251 - prev_days_first_due_max: float64
252 - prev_days_first_due_min: float64
253 - prev_days_first_due_count: float64
254 - prev_days_last_due_1st_version_mean: float64
255 - prev_days_last_due_1st_version_max: float64
256 - prev_days_last_due_1st_version_min: float64
257 - prev_days_last_due_1st_version_count: float64
258 - prev_days_last_due_mean: float64
259 - prev_days_last_due_max: float64
260 - prev_days_last_due_min: float64
261 - prev_days_last_due_count: float64
262 - prev_days_termination_mean: float64
263 - prev_days_termination_max: float64
264 - prev_days_termination_min: float64
265 - prev_days_termination_count: float64
266 - prev_nflag_insured_on_approval_mean: float64
267 - prev_nflag_insured_on_approval_max: float64
268 - prev_nflag_insured_on_approval_min: float64
269 - prev_nflag_insured_on_approval_count: float64
270 - posf_sk_id_prev_mean: float64
271 - posf_sk_id_prev_max: float64
272 - posf_sk_id_prev_min: float64
273 - posf_sk_id_prev_count: float64
274 - posf_pos_sk_id_curr_mean_mean: float64
275 - posf_pos_sk_id_curr_mean_max: float64
276 - posf_pos_sk_id_curr_mean_min: float64
```

```
277 - posf_pos_sk_id_curr_mean_count: float64
278 - posf_pos_sk_id_curr_max_mean: float64
279 - posf_pos_sk_id_curr_max_max: float64
280 - posf_pos_sk_id_curr_max_min: float64
281 - posf_pos_sk_id_curr_max_count: float64
282 - posf_pos_sk_id_curr_min_mean: float64
283 - posf_pos_sk_id_curr_min_max: float64
284 - posf_pos_sk_id_curr_min_min: float64
285 - posf_pos_sk_id_curr_min_count: float64
286 - posf_pos_sk_id_curr_count_mean: float64
287 - posf_pos_sk_id_curr_count_max: float64
288 - posf_pos_sk_id_curr_count_min: float64
289 - posf_pos_sk_id_curr_count_count: float64
290 - posf_pos_months_balance_mean_mean: float64
291 - posf_pos_months_balance_mean_max: float64
292 - posf_pos_months_balance_mean_min: float64
293 - posf_pos_months_balance_mean_count: float64
294 - posf_pos_months_balance_max_mean: float64
295 - posf_pos_months_balance_max_max: float64
296 - posf_pos_months_balance_max_min: float64
297 - posf_pos_months_balance_max_count: float64
298 - posf_pos_months_balance_min_mean: float64
299 - posf_pos_months_balance_min_max: float64
300 - posf_pos_months_balance_min_min: float64
301 - posf_pos_months_balance_min_count: float64
302 - posf_pos_months_balance_count_mean: float64
303 - posf_pos_months_balance_count_max: float64
304 - posf_pos_months_balance_count_min: float64
305 - posf_pos_months_balance_count_count: float64
306 - posf_pos_cnt_instalment_mean_mean: float64
307 - posf_pos_cnt_instalment_mean_max: float64
308 - posf_pos_cnt_instalment_mean_min: float64
309 - posf_pos_cnt_instalment_mean_count: float64
310 - posf_pos_cnt_instalment_max_mean: float64
311 - posf_pos_cnt_instalment_max_max: float64
312 - posf_pos_cnt_instalment_max_min: float64
313 - posf_pos_cnt_instalment_max_count: float64
314 - posf_pos_cnt_instalment_min_mean: float64
315 - posf_pos_cnt_instalment_min_max: float64
316 - posf_pos_cnt_instalment_min_min: float64
317 - posf_pos_cnt_instalment_min_count: float64
318 - posf_pos_cnt_instalment_count_mean: float64
319 - posf_pos_cnt_instalment_count_max: float64
320 - posf_pos_cnt_instalment_count_min: float64
321 - posf_pos_cnt_instalment_count_count: float64
322 - posf_pos_cnt_instalment_future_mean_mean: float64
323 - posf_pos_cnt_instalment_future_mean_max: float64
324 - posf_pos_cnt_instalment_future_mean_min: float64
325 - posf_pos_cnt_instalment_future_mean_count: float64
326 - posf_pos_cnt_instalment_future_max_mean: float64
327 - posf_pos_cnt_instalment_future_max_max: float64
328 - posf_pos_cnt_instalment_future_max_min: float64
329 - posf_pos_cnt_instalment_future_max_count: float64
330 - posf_pos_cnt_instalment_future_min_mean: float64
331 - posf_pos_cnt_instalment_future_min_max: float64
332 - posf_pos_cnt_instalment_future_min_min: float64
333 - posf_pos_cnt_instalment_future_min_count: float64
334 - posf_pos_cnt_instalment_future_count_mean: float64
335 - posf_pos_cnt_instalment_future_count_max: float64
336 - posf_pos_cnt_instalment_future_count_min: float64
337 - posf_pos_cnt_instalment_future_count_count: float64
338 - posf_pos_sk_dpd_mean_mean: float64
339 - posf_pos_sk_dpd_mean_max: float64
340 - posf_pos_sk_dpd_mean_min: float64
341 - posf_pos_sk_dpd_mean_count: float64
342 - posf_pos_sk_dpd_max_mean: float64
343 - posf_pos_sk_dpd_max_max: float64
344 - posf_pos_sk_dpd_max_min: float64
345 - posf_pos_sk_dpd_max_count: float64
```

346 - posf_pos_sk_dpd_min_mean: float64
347 - posf_pos_sk_dpd_min_max: float64
348 - posf_pos_sk_dpd_min_min: float64
349 - posf_pos_sk_dpd_min_count: float64
350 - posf_pos_sk_dpd_count_mean: float64
351 - posf_pos_sk_dpd_count_max: float64
352 - posf_pos_sk_dpd_count_min: float64
353 - posf_pos_sk_dpd_count_count: float64
354 - posf_pos_sk_dpd_def_mean_mean: float64
355 - posf_pos_sk_dpd_def_mean_max: float64
356 - posf_pos_sk_dpd_def_mean_min: float64
357 - posf_pos_sk_dpd_def_mean_count: float64
358 - posf_pos_sk_dpd_def_max_mean: float64
359 - posf_pos_sk_dpd_def_max_max: float64
360 - posf_pos_sk_dpd_def_max_min: float64
361 - posf_pos_sk_dpd_def_max_count: float64
362 - posf_pos_sk_dpd_def_min_mean: float64
363 - posf_pos_sk_dpd_def_min_max: float64
364 - posf_pos_sk_dpd_def_min_min: float64
365 - posf_pos_sk_dpd_def_min_count: float64
366 - posf_pos_sk_dpd_def_count_mean: float64
367 - posf_pos_sk_dpd_def_count_max: float64
368 - posf_pos_sk_dpd_def_count_min: float64
369 - posf_pos_sk_dpd_def_count_count: float64
370 - ccf_sk_id_prev_mean: float64
371 - ccf_sk_id_prev_max: float64
372 - ccf_sk_id_prev_min: float64
373 - ccf_sk_id_prev_count: float64
374 - ccf_cc_sk_id_curr_mean_mean: float64
375 - ccf_cc_sk_id_curr_mean_max: float64
376 - ccf_cc_sk_id_curr_mean_min: float64
377 - ccf_cc_sk_id_curr_mean_count: float64
378 - ccf_cc_sk_id_curr_max_mean: float64
379 - ccf_cc_sk_id_curr_max_max: float64
380 - ccf_cc_sk_id_curr_max_min: float64
381 - ccf_cc_sk_id_curr_max_count: float64
382 - ccf_cc_sk_id_curr_min_mean: float64
383 - ccf_cc_sk_id_curr_min_max: float64
384 - ccf_cc_sk_id_curr_min_min: float64
385 - ccf_cc_sk_id_curr_min_count: float64
386 - ccf_cc_sk_id_curr_count_mean: float64
387 - ccf_cc_sk_id_curr_count_max: float64
388 - ccf_cc_sk_id_curr_count_min: float64
389 - ccf_cc_sk_id_curr_count_count: float64
390 - ccf_cc_months_balance_mean_mean: float64
391 - ccf_cc_months_balance_mean_max: float64
392 - ccf_cc_months_balance_mean_min: float64
393 - ccf_cc_months_balance_mean_count: float64
394 - ccf_cc_months_balance_max_mean: float64
395 - ccf_cc_months_balance_max_max: float64
396 - ccf_cc_months_balance_max_min: float64
397 - ccf_cc_months_balance_max_count: float64
398 - ccf_cc_months_balance_min_mean: float64
399 - ccf_cc_months_balance_min_max: float64
400 - ccf_cc_months_balance_min_min: float64
401 - ccf_cc_months_balance_min_count: float64
402 - ccf_cc_months_balance_count_mean: float64
403 - ccf_cc_months_balance_count_max: float64
404 - ccf_cc_months_balance_count_min: float64
405 - ccf_cc_months_balance_count_count: float64
406 - ccf_cc_amt_balance_mean_mean: float64
407 - ccf_cc_amt_balance_mean_max: float64
408 - ccf_cc_amt_balance_mean_min: float64
409 - ccf_cc_amt_balance_mean_count: float64
410 - ccf_cc_amt_balance_max_mean: float64
411 - ccf_cc_amt_balance_max_max: float64
412 - ccf_cc_amt_balance_max_min: float64
413 - ccf_cc_amt_balance_max_count: float64
414 - ccf_cc_amt_balance_min_mean: float64

```
415 - ccf_cc_amt_balance_min_max: float64
416 - ccf_cc_amt_balance_min_min: float64
417 - ccf_cc_amt_balance_min_count: float64
418 - ccf_cc_amt_balance_count_mean: float64
419 - ccf_cc_amt_balance_count_max: float64
420 - ccf_cc_amt_balance_count_min: float64
421 - ccf_cc_amt_balance_count_count: float64
422 - ccf_cc_amt_credit_limit_actual_mean_mean: float64
423 - ccf_cc_amt_credit_limit_actual_mean_max: float64
424 - ccf_cc_amt_credit_limit_actual_mean_min: float64
425 - ccf_cc_amt_credit_limit_actual_mean_count: float64
426 - ccf_cc_amt_credit_limit_actual_max_mean: float64
427 - ccf_cc_amt_credit_limit_actual_max_max: float64
428 - ccf_cc_amt_credit_limit_actual_max_min: float64
429 - ccf_cc_amt_credit_limit_actual_max_count: float64
430 - ccf_cc_amt_credit_limit_actual_min_mean: float64
431 - ccf_cc_amt_credit_limit_actual_min_max: float64
432 - ccf_cc_amt_credit_limit_actual_min_min: float64
433 - ccf_cc_amt_credit_limit_actual_min_count: float64
434 - ccf_cc_amt_credit_limit_actual_count_mean: float64
435 - ccf_cc_amt_credit_limit_actual_count_max: float64
436 - ccf_cc_amt_credit_limit_actual_count_min: float64
437 - ccf_cc_amt_credit_limit_actual_count_count: float64
438 - ccf_cc_amt_drawings_atm_current_mean_mean: float64
439 - ccf_cc_amt_drawings_atm_current_mean_max: float64
440 - ccf_cc_amt_drawings_atm_current_mean_min: float64
441 - ccf_cc_amt_drawings_atm_current_mean_count: float64
442 - ccf_cc_amt_drawings_atm_current_max_mean: float64
443 - ccf_cc_amt_drawings_atm_current_max_max: float64
444 - ccf_cc_amt_drawings_atm_current_max_min: float64
445 - ccf_cc_amt_drawings_atm_current_max_count: float64
446 - ccf_cc_amt_drawings_atm_current_min_mean: float64
447 - ccf_cc_amt_drawings_atm_current_min_max: float64
448 - ccf_cc_amt_drawings_atm_current_min_min: float64
449 - ccf_cc_amt_drawings_atm_current_min_count: float64
450 - ccf_cc_amt_drawings_atm_current_count_mean: float64
451 - ccf_cc_amt_drawings_atm_current_count_max: float64
452 - ccf_cc_amt_drawings_atm_current_count_min: float64
453 - ccf_cc_amt_drawings_atm_current_count_count: float64
454 - ccf_cc_amt_drawings_current_mean_mean: float64
455 - ccf_cc_amt_drawings_current_mean_max: float64
456 - ccf_cc_amt_drawings_current_mean_min: float64
457 - ccf_cc_amt_drawings_current_mean_count: float64
458 - ccf_cc_amt_drawings_current_max_mean: float64
459 - ccf_cc_amt_drawings_current_max_max: float64
460 - ccf_cc_amt_drawings_current_max_min: float64
461 - ccf_cc_amt_drawings_current_max_count: float64
462 - ccf_cc_amt_drawings_current_min_mean: float64
463 - ccf_cc_amt_drawings_current_min_max: float64
464 - ccf_cc_amt_drawings_current_min_min: float64
465 - ccf_cc_amt_drawings_current_min_count: float64
466 - ccf_cc_amt_drawings_current_count_mean: float64
467 - ccf_cc_amt_drawings_current_count_max: float64
468 - ccf_cc_amt_drawings_current_count_min: float64
469 - ccf_cc_amt_drawings_current_count_count: float64
470 - ccf_cc_amt_drawings_other_current_mean_mean: float64
471 - ccf_cc_amt_drawings_other_current_mean_max: float64
472 - ccf_cc_amt_drawings_other_current_mean_min: float64
473 - ccf_cc_amt_drawings_other_current_mean_count: float64
474 - ccf_cc_amt_drawings_other_current_max_mean: float64
475 - ccf_cc_amt_drawings_other_current_max_max: float64
476 - ccf_cc_amt_drawings_other_current_max_min: float64
477 - ccf_cc_amt_drawings_other_current_max_count: float64
478 - ccf_cc_amt_drawings_other_current_min_mean: float64
479 - ccf_cc_amt_drawings_other_current_min_max: float64
480 - ccf_cc_amt_drawings_other_current_min_min: float64
481 - ccf_cc_amt_drawings_other_current_min_count: float64
482 - ccf_cc_amt_drawings_other_current_count_mean: float64
483 - ccf_cc_amt_drawings_other_current_count_max: float64
```

484 - ccf_cc_amt_drawings_other_current_count_min: float64
485 - ccf_cc_amt_drawings_other_current_count_count: float64
486 - ccf_cc_amt_drawings_pos_current_mean_mean: float64
487 - ccf_cc_amt_drawings_pos_current_mean_max: float64
488 - ccf_cc_amt_drawings_pos_current_mean_min: float64
489 - ccf_cc_amt_drawings_pos_current_mean_count: float64
490 - ccf_cc_amt_drawings_pos_current_max_mean: float64
491 - ccf_cc_amt_drawings_pos_current_max_max: float64
492 - ccf_cc_amt_drawings_pos_current_max_min: float64
493 - ccf_cc_amt_drawings_pos_current_max_count: float64
494 - ccf_cc_amt_drawings_pos_current_min_mean: float64
495 - ccf_cc_amt_drawings_pos_current_min_max: float64
496 - ccf_cc_amt_drawings_pos_current_min_min: float64
497 - ccf_cc_amt_drawings_pos_current_min_count: float64
498 - ccf_cc_amt_drawings_pos_current_count_mean: float64
499 - ccf_cc_amt_drawings_pos_current_count_max: float64
500 - ccf_cc_amt_drawings_pos_current_count_min: float64
501 - ccf_cc_amt_drawings_pos_current_count_count: float64
502 - ccf_cc_inst_min_regularity_mean_mean: float64
503 - ccf_cc_inst_min_regularity_mean_max: float64
504 - ccf_cc_inst_min_regularity_mean_min: float64
505 - ccf_cc_inst_min_regularity_mean_count: float64
506 - ccf_cc_inst_min_regularity_max_mean: float64
507 - ccf_cc_inst_min_regularity_max_max: float64
508 - ccf_cc_inst_min_regularity_max_min: float64
509 - ccf_cc_inst_min_regularity_max_count: float64
510 - ccf_cc_inst_min_regularity_min_mean: float64
511 - ccf_cc_inst_min_regularity_min_max: float64
512 - ccf_cc_inst_min_regularity_min_min: float64
513 - ccf_cc_inst_min_regularity_min_count: float64
514 - ccf_cc_inst_min_regularity_count_mean: float64
515 - ccf_cc_inst_min_regularity_count_max: float64
516 - ccf_cc_inst_min_regularity_count_min: float64
517 - ccf_cc_inst_min_regularity_count_count: float64
518 - ccf_cc_amt_payment_current_mean_mean: float64
519 - ccf_cc_amt_payment_current_mean_max: float64
520 - ccf_cc_amt_payment_current_mean_min: float64
521 - ccf_cc_amt_payment_current_mean_count: float64
522 - ccf_cc_amt_payment_current_max_mean: float64
523 - ccf_cc_amt_payment_current_max_max: float64
524 - ccf_cc_amt_payment_current_max_min: float64
525 - ccf_cc_amt_payment_current_max_count: float64
526 - ccf_cc_amt_payment_current_min_mean: float64
527 - ccf_cc_amt_payment_current_min_max: float64
528 - ccf_cc_amt_payment_current_min_min: float64
529 - ccf_cc_amt_payment_current_min_count: float64
530 - ccf_cc_amt_payment_current_count_mean: float64
531 - ccf_cc_amt_payment_current_count_max: float64
532 - ccf_cc_amt_payment_current_count_min: float64
533 - ccf_cc_amt_payment_current_count_count: float64
534 - ccf_cc_amt_payment_total_current_mean_mean: float64
535 - ccf_cc_amt_payment_total_current_mean_max: float64
536 - ccf_cc_amt_payment_total_current_mean_min: float64
537 - ccf_cc_amt_payment_total_current_mean_count: float64
538 - ccf_cc_amt_payment_total_current_max_mean: float64
539 - ccf_cc_amt_payment_total_current_max_max: float64
540 - ccf_cc_amt_payment_total_current_max_min: float64
541 - ccf_cc_amt_payment_total_current_max_count: float64
542 - ccf_cc_amt_payment_total_current_min_mean: float64
543 - ccf_cc_amt_payment_total_current_min_max: float64
544 - ccf_cc_amt_payment_total_current_min_min: float64
545 - ccf_cc_amt_payment_total_current_min_count: float64
546 - ccf_cc_amt_payment_total_current_count_mean: float64
547 - ccf_cc_amt_payment_total_current_count_max: float64
548 - ccf_cc_amt_payment_total_current_count_min: float64
549 - ccf_cc_amt_payment_total_current_count_count: float64
550 - ccf_cc_amt_receivable_principal_mean_mean: float64
551 - ccf_cc_amt_receivable_principal_mean_max: float64
552 - ccf_cc_amt_receivable_principal_mean_min: float64

553 - ccf_cc_amt_receivable_principal_mean_count: float64
554 - ccf_cc_amt_receivable_principal_max_mean: float64
555 - ccf_cc_amt_receivable_principal_max_max: float64
556 - ccf_cc_amt_receivable_principal_max_min: float64
557 - ccf_cc_amt_receivable_principal_max_count: float64
558 - ccf_cc_amt_receivable_principal_min_mean: float64
559 - ccf_cc_amt_receivable_principal_min_max: float64
560 - ccf_cc_amt_receivable_principal_min_min: float64
561 - ccf_cc_amt_receivable_principal_min_count: float64
562 - ccf_cc_amt_receivable_principal_count_mean: float64
563 - ccf_cc_amt_receivable_principal_count_max: float64
564 - ccf_cc_amt_receivable_principal_count_min: float64
565 - ccf_cc_amt_receivable_principal_count_count: float64
566 - ccf_cc_amt_recivable_mean_mean: float64
567 - ccf_cc_amt_recivable_mean_max: float64
568 - ccf_cc_amt_recivable_mean_min: float64
569 - ccf_cc_amt_recivable_mean_count: float64
570 - ccf_cc_amt_recivable_max_mean: float64
571 - ccf_cc_amt_recivable_max_max: float64
572 - ccf_cc_amt_recivable_max_min: float64
573 - ccf_cc_amt_recivable_max_count: float64
574 - ccf_cc_amt_recivable_min_mean: float64
575 - ccf_cc_amt_recivable_min_max: float64
576 - ccf_cc_amt_recivable_min_min: float64
577 - ccf_cc_amt_recivable_min_count: float64
578 - ccf_cc_amt_recivable_count_mean: float64
579 - ccf_cc_amt_recivable_count_max: float64
580 - ccf_cc_amt_recivable_count_min: float64
581 - ccf_cc_amt_recivable_count_count: float64
582 - ccf_cc_amt_total_receivable_mean_mean: float64
583 - ccf_cc_amt_total_receivable_mean_max: float64
584 - ccf_cc_amt_total_receivable_mean_min: float64
585 - ccf_cc_amt_total_receivable_mean_count: float64
586 - ccf_cc_amt_total_receivable_max_mean: float64
587 - ccf_cc_amt_total_receivable_max_max: float64
588 - ccf_cc_amt_total_receivable_max_min: float64
589 - ccf_cc_amt_total_receivable_max_count: float64
590 - ccf_cc_amt_total_receivable_min_mean: float64
591 - ccf_cc_amt_total_receivable_min_max: float64
592 - ccf_cc_amt_total_receivable_min_min: float64
593 - ccf_cc_amt_total_receivable_min_count: float64
594 - ccf_cc_amt_total_receivable_count_mean: float64
595 - ccf_cc_amt_total_receivable_count_max: float64
596 - ccf_cc_amt_total_receivable_count_min: float64
597 - ccf_cc_amt_total_receivable_count_count: float64
598 - ccf_cc_cnt_drawings_atm_current_mean_mean: float64
599 - ccf_cc_cnt_drawings_atm_current_mean_max: float64
600 - ccf_cc_cnt_drawings_atm_current_mean_min: float64
601 - ccf_cc_cnt_drawings_atm_current_mean_count: float64
602 - ccf_cc_cnt_drawings_atm_current_max_mean: float64
603 - ccf_cc_cnt_drawings_atm_current_max_max: float64
604 - ccf_cc_cnt_drawings_atm_current_max_min: float64
605 - ccf_cc_cnt_drawings_atm_current_max_count: float64
606 - ccf_cc_cnt_drawings_atm_current_min_mean: float64
607 - ccf_cc_cnt_drawings_atm_current_min_max: float64
608 - ccf_cc_cnt_drawings_atm_current_min_min: float64
609 - ccf_cc_cnt_drawings_atm_current_min_count: float64
610 - ccf_cc_cnt_drawings_atm_current_count_mean: float64
611 - ccf_cc_cnt_drawings_atm_current_count_max: float64
612 - ccf_cc_cnt_drawings_atm_current_count_min: float64
613 - ccf_cc_cnt_drawings_atm_current_count_count: float64
614 - ccf_cc_cnt_drawings_current_mean_mean: float64
615 - ccf_cc_cnt_drawings_current_mean_max: float64
616 - ccf_cc_cnt_drawings_current_mean_min: float64
617 - ccf_cc_cnt_drawings_current_mean_count: float64
618 - ccf_cc_cnt_drawings_current_max_mean: float64
619 - ccf_cc_cnt_drawings_current_max_max: float64
620 - ccf_cc_cnt_drawings_current_max_min: float64
621 - ccf_cc_cnt_drawings_current_max_count: float64

622 - ccf_cc_cnt_drawings_current_min_mean: float64
623 - ccf_cc_cnt_drawings_current_min_max: float64
624 - ccf_cc_cnt_drawings_current_min_min: float64
625 - ccf_cc_cnt_drawings_current_min_count: float64
626 - ccf_cc_cnt_drawings_current_count_mean: float64
627 - ccf_cc_cnt_drawings_current_count_max: float64
628 - ccf_cc_cnt_drawings_current_count_min: float64
629 - ccf_cc_cnt_drawings_current_count_count: float64
630 - ccf_cc_cnt_drawings_other_current_mean_mean: float64
631 - ccf_cc_cnt_drawings_other_current_mean_max: float64
632 - ccf_cc_cnt_drawings_other_current_mean_min: float64
633 - ccf_cc_cnt_drawings_other_current_mean_count: float64
634 - ccf_cc_cnt_drawings_other_current_max_mean: float64
635 - ccf_cc_cnt_drawings_other_current_max_max: float64
636 - ccf_cc_cnt_drawings_other_current_max_min: float64
637 - ccf_cc_cnt_drawings_other_current_max_count: float64
638 - ccf_cc_cnt_drawings_other_current_min_mean: float64
639 - ccf_cc_cnt_drawings_other_current_min_max: float64
640 - ccf_cc_cnt_drawings_other_current_min_min: float64
641 - ccf_cc_cnt_drawings_other_current_min_count: float64
642 - ccf_cc_cnt_drawings_other_current_count_mean: float64
643 - ccf_cc_cnt_drawings_other_current_count_max: float64
644 - ccf_cc_cnt_drawings_other_current_count_min: float64
645 - ccf_cc_cnt_drawings_other_current_count_count: float64
646 - ccf_cc_cnt_drawings_pos_current_mean_mean: float64
647 - ccf_cc_cnt_drawings_pos_current_mean_max: float64
648 - ccf_cc_cnt_drawings_pos_current_mean_min: float64
649 - ccf_cc_cnt_drawings_pos_current_mean_count: float64
650 - ccf_cc_cnt_drawings_pos_current_max_mean: float64
651 - ccf_cc_cnt_drawings_pos_current_max_max: float64
652 - ccf_cc_cnt_drawings_pos_current_max_min: float64
653 - ccf_cc_cnt_drawings_pos_current_max_count: float64
654 - ccf_cc_cnt_drawings_pos_current_min_mean: float64
655 - ccf_cc_cnt_drawings_pos_current_min_max: float64
656 - ccf_cc_cnt_drawings_pos_current_min_min: float64
657 - ccf_cc_cnt_drawings_pos_current_min_count: float64
658 - ccf_cc_cnt_drawings_pos_current_count_mean: float64
659 - ccf_cc_cnt_drawings_pos_current_count_max: float64
660 - ccf_cc_cnt_drawings_pos_current_count_min: float64
661 - ccf_cc_cnt_drawings_pos_current_count_count: float64
662 - ccf_cc_cnt_instalment_mature_cum_mean_mean: float64
663 - ccf_cc_cnt_instalment_mature_cum_mean_max: float64
664 - ccf_cc_cnt_instalment_mature_cum_mean_min: float64
665 - ccf_cc_cnt_instalment_mature_cum_mean_count: float64
666 - ccf_cc_cnt_instalment_mature_cum_max_mean: float64
667 - ccf_cc_cnt_instalment_mature_cum_max_max: float64
668 - ccf_cc_cnt_instalment_mature_cum_max_min: float64
669 - ccf_cc_cnt_instalment_mature_cum_max_count: float64
670 - ccf_cc_cnt_instalment_mature_cum_min_mean: float64
671 - ccf_cc_cnt_instalment_mature_cum_min_max: float64
672 - ccf_cc_cnt_instalment_mature_cum_min_min: float64
673 - ccf_cc_cnt_instalment_mature_cum_min_count: float64
674 - ccf_cc_cnt_instalment_mature_cum_count_mean: float64
675 - ccf_cc_cnt_instalment_mature_cum_count_max: float64
676 - ccf_cc_cnt_instalment_mature_cum_count_min: float64
677 - ccf_cc_cnt_instalment_mature_cum_count_count: float64
678 - ccf_cc_sk_dpd_mean_mean: float64
679 - ccf_cc_sk_dpd_mean_max: float64
680 - ccf_cc_sk_dpd_mean_min: float64
681 - ccf_cc_sk_dpd_mean_count: float64
682 - ccf_cc_sk_dpd_max_mean: float64
683 - ccf_cc_sk_dpd_max_max: float64
684 - ccf_cc_sk_dpd_max_min: float64
685 - ccf_cc_sk_dpd_max_count: float64
686 - ccf_cc_sk_dpd_min_mean: float64
687 - ccf_cc_sk_dpd_min_max: float64
688 - ccf_cc_sk_dpd_min_min: float64
689 - ccf_cc_sk_dpd_min_count: float64
690 - ccf_cc_sk_dpd_count_mean: float64

691 - ccf_cc_sk_dpd_count_max: float64
692 - ccf_cc_sk_dpd_count_min: float64
693 - ccf_cc_sk_dpd_count_count: float64
694 - ccf_cc_sk_dpd_def_mean_mean: float64
695 - ccf_cc_sk_dpd_def_mean_max: float64
696 - ccf_cc_sk_dpd_def_mean_min: float64
697 - ccf_cc_sk_dpd_def_mean_count: float64
698 - ccf_cc_sk_dpd_def_max_mean: float64
699 - ccf_cc_sk_dpd_def_max_max: float64
700 - ccf_cc_sk_dpd_def_max_min: float64
701 - ccf_cc_sk_dpd_def_max_count: float64
702 - ccf_cc_sk_dpd_def_min_mean: float64
703 - ccf_cc_sk_dpd_def_min_max: float64
704 - ccf_cc_sk_dpd_def_min_min: float64
705 - ccf_cc_sk_dpd_def_min_count: float64
706 - ccf_cc_sk_dpd_def_count_mean: float64
707 - ccf_cc_sk_dpd_def_count_max: float64
708 - ccf_cc_sk_dpd_def_count_min: float64
709 - ccf_cc_sk_dpd_def_count_count: float64
710 - instf_sk_id_prev_mean: float64
711 - instf_sk_id_prev_max: float64
712 - instf_sk_id_prev_min: float64
713 - instf_sk_id_prev_count: float64
714 - instf_inst_sk_id_curr_mean_mean: float64
715 - instf_inst_sk_id_curr_mean_max: float64
716 - instf_inst_sk_id_curr_mean_min: float64
717 - instf_inst_sk_id_curr_mean_count: float64
718 - instf_inst_sk_id_curr_max_mean: float64
719 - instf_inst_sk_id_curr_max_max: float64
720 - instf_inst_sk_id_curr_max_min: float64
721 - instf_inst_sk_id_curr_max_count: float64
722 - instf_inst_sk_id_curr_min_mean: float64
723 - instf_inst_sk_id_curr_min_max: float64
724 - instf_inst_sk_id_curr_min_min: float64
725 - instf_inst_sk_id_curr_min_count: float64
726 - instf_inst_sk_id_curr_count_mean: float64
727 - instf_inst_sk_id_curr_count_max: float64
728 - instf_inst_sk_id_curr_count_min: float64
729 - instf_inst_sk_id_curr_count_count: float64
730 - instf_inst_num_instalment_version_mean_mean: float64
731 - instf_inst_num_instalment_version_mean_max: float64
732 - instf_inst_num_instalment_version_mean_min: float64
733 - instf_inst_num_instalment_version_mean_count: float64
734 - instf_inst_num_instalment_version_max_mean: float64
735 - instf_inst_num_instalment_version_max_max: float64
736 - instf_inst_num_instalment_version_max_min: float64
737 - instf_inst_num_instalment_version_max_count: float64
738 - instf_inst_num_instalment_version_min_mean: float64
739 - instf_inst_num_instalment_version_min_max: float64
740 - instf_inst_num_instalment_version_min_min: float64
741 - instf_inst_num_instalment_version_min_count: float64
742 - instf_inst_num_instalment_version_count_mean: float64
743 - instf_inst_num_instalment_version_count_max: float64
744 - instf_inst_num_instalment_version_count_min: float64
745 - instf_inst_num_instalment_version_count_count: float64
746 - instf_inst_num_instalment_number_mean_mean: float64
747 - instf_inst_num_instalment_number_mean_max: float64
748 - instf_inst_num_instalment_number_mean_min: float64
749 - instf_inst_num_instalment_number_mean_count: float64
750 - instf_inst_num_instalment_number_max_mean: float64
751 - instf_inst_num_instalment_number_max_max: float64
752 - instf_inst_num_instalment_number_max_min: float64
753 - instf_inst_num_instalment_number_max_count: float64
754 - instf_inst_num_instalment_number_min_mean: float64
755 - instf_inst_num_instalment_number_min_max: float64
756 - instf_inst_num_instalment_number_min_min: float64
757 - instf_inst_num_instalment_number_min_count: float64
758 - instf_inst_num_instalment_number_count_mean: float64
759 - instf_inst_num_instalment_number_count_max: float64

```
760 - instf_inst_num_instalment_number_count_min: float64
761 - instf_inst_num_instalment_number_count_count: float64
762 - instf_inst_days_instalment_mean_mean: float64
763 - instf_inst_days_instalment_mean_max: float64
764 - instf_inst_days_instalment_mean_min: float64
765 - instf_inst_days_instalment_mean_count: float64
766 - instf_inst_days_instalment_max_mean: float64
767 - instf_inst_days_instalment_max_max: float64
768 - instf_inst_days_instalment_max_min: float64
769 - instf_inst_days_instalment_max_count: float64
770 - instf_inst_days_instalment_min_mean: float64
771 - instf_inst_days_instalment_min_max: float64
772 - instf_inst_days_instalment_min_min: float64
773 - instf_inst_days_instalment_min_count: float64
774 - instf_inst_days_instalment_count_mean: float64
775 - instf_inst_days_instalment_count_max: float64
776 - instf_inst_days_instalment_count_min: float64
777 - instf_inst_days_instalment_count_count: float64
778 - instf_inst_days_entry_payment_mean_mean: float64
779 - instf_inst_days_entry_payment_mean_max: float64
780 - instf_inst_days_entry_payment_mean_min: float64
781 - instf_inst_days_entry_payment_mean_count: float64
782 - instf_inst_days_entry_payment_max_mean: float64
783 - instf_inst_days_entry_payment_max_max: float64
784 - instf_inst_days_entry_payment_max_min: float64
785 - instf_inst_days_entry_payment_max_count: float64
786 - instf_inst_days_entry_payment_min_mean: float64
787 - instf_inst_days_entry_payment_min_max: float64
788 - instf_inst_days_entry_payment_min_min: float64
789 - instf_inst_days_entry_payment_min_count: float64
790 - instf_inst_days_entry_payment_count_mean: float64
791 - instf_inst_days_entry_payment_count_max: float64
792 - instf_inst_days_entry_payment_count_min: float64
793 - instf_inst_days_entry_payment_count_count: float64
794 - instf_inst_amt_instalment_mean_mean: float64
795 - instf_inst_amt_instalment_mean_max: float64
796 - instf_inst_amt_instalment_mean_min: float64
797 - instf_inst_amt_instalment_mean_count: float64
798 - instf_inst_amt_instalment_max_mean: float64
799 - instf_inst_amt_instalment_max_max: float64
800 - instf_inst_amt_instalment_max_min: float64
801 - instf_inst_amt_instalment_max_count: float64
802 - instf_inst_amt_instalment_min_mean: float64
803 - instf_inst_amt_instalment_min_max: float64
804 - instf_inst_amt_instalment_min_min: float64
805 - instf_inst_amt_instalment_min_count: float64
806 - instf_inst_amt_instalment_count_mean: float64
807 - instf_inst_amt_instalment_count_max: float64
808 - instf_inst_amt_instalment_count_min: float64
809 - instf_inst_amt_instalment_count_count: float64
810 - instf_inst_amt_payment_mean_mean: float64
811 - instf_inst_amt_payment_mean_max: float64
812 - instf_inst_amt_payment_mean_min: float64
813 - instf_inst_amt_payment_mean_count: float64
814 - instf_inst_amt_payment_max_mean: float64
815 - instf_inst_amt_payment_max_max: float64
816 - instf_inst_amt_payment_max_min: float64
817 - instf_inst_amt_payment_max_count: float64
818 - instf_inst_amt_payment_min_mean: float64
819 - instf_inst_amt_payment_min_max: float64
820 - instf_inst_amt_payment_min_min: float64
821 - instf_inst_amt_payment_min_count: float64
822 - instf_inst_amt_payment_count_mean: float64
823 - instf_inst_amt_payment_count_max: float64
824 - instf_inst_amt_payment_count_min: float64
825 - instf_inst_amt_payment_count_count: float64
```

5.3.1 Modificando Tipos

```
In [9]: df1['flag_own_car'] = df1['flag_own_car'].map({'N': 0, 'Y': 1})
df1['flag_own_realty'] = df1['flag_own_realty'].map({'N': 0, 'Y': 1})
df1['emergencystate_mode'] = df1['emergencystate_mode'].map({'No': 0, 'Yes': 1})

weekday_map = {
    'MONDAY': 1,
    'TUESDAY': 2,
    'WEDNESDAY': 3,
    'THURSDAY': 4,
    'FRIDAY': 5,
    'SATURDAY': 6,
    'SUNDAY': 7
}

df1['weekday_appr_process_start'] = df1['weekday_appr_process_start'].map(weekday_map)
```

5.4 Check e Fillout NAN

```
In [10]: df1 = df1.replace('XNA', pd.NA)

print(df1.isna().sum().to_string())
```

target	0
name_contract_type	0
code_gender	4
flag_own_car	0
flag_own_realty	0
cnt_children	0
amt_income_total	0
amt_credit	0
amt_annuity	12
amt_goods_price	278
name_type_suite	1292
name_income_type	0
name_education_type	0
name_family_status	0
name_housing_type	0
region_population_relative	0
days_birth	0
days_employed	0
days_registration	0
days_id_publish	0
own_car_age	202929
flag_mobil	0
flag_emp_phone	0
flag_work_phone	0
flag_cont_mobile	0
flag_phone	0
flag_email	0
occupation_type	96391
cnt_fam_members	2
region_rating_client	0
region_rating_client_w_city	0
weekday_appr_process_start	0
hour_appr_process_start	0
reg_region_not_live_region	0
reg_region_not_work_region	0
live_region_not_work_region	0
reg_city_not_live_city	0
reg_city_not_work_city	0
live_city_not_work_city	0
organization_type	55374
ext_source_1	173378
ext_source_2	660
ext_source_3	60965
apartments_avg	156061
basementarea_avg	179943
years_beginexpluatation_avg	150007
years_build_avg	204488
commonarea_avg	214865
elevators_avg	163891
entrances_avg	154828
floorsmax_avg	153020
floorsmin_avg	208642
landarea_avg	182590
livingapartments_avg	210199
livingarea_avg	154350
nonlivingapartments_avg	213514
nonlivingarea_avg	169682
apartments_mode	156061
basementarea_mode	179943
years_beginexpluatation_mode	150007
years_build_mode	204488
commonarea_mode	214865
elevators_mode	163891
entrances_mode	154828
floorsmax_mode	153020
floorsmin_mode	208642
landarea_mode	182590
livingapartments_mode	210199
livingarea_mode	154350

nonlivingapartments_mode	213514
nonlivingarea_mode	169682
apartments_medi	156061
basementarea_medi	179943
years_beginexploitation_medi	150007
years_build_medi	204488
commonarea_medi	214865
elevators_medi	163891
entrances_medi	154828
floorsmax_medi	153020
floorsmin_medi	208642
landarea_medi	182590
livingapartments_medi	210199
livingarea_medi	154350
nonlivingapartments_medi	213514
nonlivingarea_medi	169682
fondkapremont_mode	210295
housetype_mode	154297
totalarea_mode	148431
wallsmaterial_mode	156341
emergencystate_mode	145755
obs_30_cnt_social_circle	1021
def_30_cnt_social_circle	1021
obs_60_cnt_social_circle	1021
def_60_cnt_social_circle	1021
days_last_phone_change	1
flag_document_2	0
flag_document_3	0
flag_document_4	0
flag_document_5	0
flag_document_6	0
flag_document_7	0
flag_document_8	0
flag_document_9	0
flag_document_10	0
flag_document_11	0
flag_document_12	0
flag_document_13	0
flag_document_14	0
flag_document_15	0
flag_document_16	0
flag_document_17	0
flag_document_18	0
flag_document_19	0
flag_document_20	0
flag_document_21	0
amt_req_credit_bureau_hour	41519
amt_req_credit_bureau_day	41519
amt_req_credit_bureau_week	41519
amt_req_credit_bureau_mon	41519
amt_req_credit_bureau_qrt	41519
amt_req_credit_bureau_year	41519
bur_sk_id_bureau_mean	44020
bur_sk_id_bureau_max	44020
bur_sk_id_bureau_min	44020
bur_sk_id_bureau_count	44020
bur_days_credit_mean	44020
bur_days_credit_max	44020
bur_days_credit_min	44020
bur_days_credit_count	44020
bur_credit_day_overdue_mean	44020
bur_credit_day_overdue_max	44020
bur_credit_day_overdue_min	44020
bur_credit_day_overdue_count	44020
bur_days_credit_enddate_mean	46269
bur_days_credit_enddate_max	46269
bur_days_credit_enddate_min	46269
bur_days_credit_enddate_count	44020
bur_days_enddate_fact_mean	77156

bur_days_enddate_fact_max	77156
bur_days_enddate_fact_min	77156
bur_days_enddate_fact_count	44020
bur_amt_credit_max_overdue_mean	123625
bur_amt_credit_max_overdue_max	123625
bur_amt_credit_max_overdue_min	123625
bur_amt_credit_max_overdue_count	44020
bur_cnt_credit_prolong_mean	44020
bur_cnt_credit_prolong_max	44020
bur_cnt_credit_prolong_min	44020
bur_cnt_credit_prolong_count	44020
bur_amt_credit_sum_mean	44021
bur_amt_credit_sum_max	44021
bur_amt_credit_sum_min	44021
bur_amt_credit_sum_count	44020
bur_amt_credit_sum_debt_mean	51380
bur_amt_credit_sum_debt_max	51380
bur_amt_credit_sum_debt_min	51380
bur_amt_credit_sum_debt_count	44020
bur_amt_credit_sum_limit_mean	65069
bur_amt_credit_sum_limit_max	65069
bur_amt_credit_sum_limit_min	65069
bur_amt_credit_sum_limit_count	44020
bur_amt_credit_sum_overdue_mean	44020
bur_amt_credit_sum_overdue_max	44020
bur_amt_credit_sum_overdue_min	44020
bur_amt_credit_sum_overdue_count	44020
bur_days_credit_update_mean	44020
bur_days_credit_update_max	44020
bur_days_credit_update_min	44020
bur_days_credit_update_count	44020
bur_amt_annuity_mean	227502
bur_amt_annuity_max	227502
bur_amt_annuity_min	227502
bur_amt_annuity_count	44020
bur_bb_months_balance_mean_mean	215280
bur_bb_months_balance_mean_max	215280
bur_bb_months_balance_mean_min	215280
bur_bb_months_balance_mean_count	44020
bur_bb_months_balance_max_mean	215280
bur_bb_months_balance_max_max	215280
bur_bb_months_balance_max_min	215280
bur_bb_months_balance_max_count	44020
bur_bb_months_balance_min_mean	215280
bur_bb_months_balance_min_max	215280
bur_bb_months_balance_min_min	215280
bur_bb_months_balance_min_count	44020
bur_bb_months_balance_count_mean	215280
bur_bb_months_balance_count_max	215280
bur_bb_months_balance_count_min	215280
bur_bb_months_balance_count_count	44020
prev_sk_id_prev_mean	16454
prev_sk_id_prev_max	16454
prev_sk_id_prev_min	16454
prev_sk_id_prev_count	16454
prev_amt_annuity_mean	16871
prev_amt_annuity_max	16871
prev_amt_annuity_min	16871
prev_amt_annuity_count	16454
prev_amt_application_mean	16454
prev_amt_application_max	16454
prev_amt_application_min	16454
prev_amt_application_count	16454
prev_amt_credit_mean	16454
prev_amt_credit_max	16454
prev_amt_credit_min	16454
prev_amt_credit_count	16454
prev_amt_down_payment_mean	33906
prev_amt_down_payment_max	33906

prev_amt_down_payment_min	33906
prev_amt_down_payment_count	16454
prev_amt_goods_price_mean	17429
prev_amt_goods_price_max	17429
prev_amt_goods_price_min	17429
prev_amt_goods_price_count	16454
prev_hour_appr_process_start_mean	16454
prev_hour_appr_process_start_max	16454
prev_hour_appr_process_start_min	16454
prev_hour_appr_process_start_count	16454
prev_nflag_last_appl_in_day_mean	16454
prev_nflag_last_appl_in_day_max	16454
prev_nflag_last_appl_in_day_min	16454
prev_nflag_last_appl_in_day_count	16454
prev_rate_down_payment_mean	33906
prev_rate_down_payment_max	33906
prev_rate_down_payment_min	33906
prev_rate_down_payment_count	16454
prev_rate_interest_primary_mean	302902
prev_rate_interest_primary_max	302902
prev_rate_interest_primary_min	302902
prev_rate_interest_primary_count	16454
prev_rate_interest_privileged_mean	302902
prev_rate_interest_privileged_max	302902
prev_rate_interest_privileged_min	302902
prev_rate_interest_privileged_count	16454
prev_days_decision_mean	16454
prev_days_decision_max	16454
prev_days_decision_min	16454
prev_days_decision_count	16454
prev_sellerplace_area_mean	16454
prev_sellerplace_area_max	16454
prev_sellerplace_area_min	16454
prev_sellerplace_area_count	16454
prev_cnt_payment_mean	16869
prev_cnt_payment_max	16869
prev_cnt_payment_min	16869
prev_cnt_payment_count	16454
prev_days_first_drawing_mean	17751
prev_days_first_drawing_max	17751
prev_days_first_drawing_min	17751
prev_days_first_drawing_count	16454
prev_days_first_due_mean	17751
prev_days_first_due_max	17751
prev_days_first_due_min	17751
prev_days_first_due_count	16454
prev_days_last_due_1st_version_mean	17751
prev_days_last_due_1st_version_max	17751
prev_days_last_due_1st_version_min	17751
prev_days_last_due_1st_version_count	16454
prev_days_last_due_mean	17751
prev_days_last_due_max	17751
prev_days_last_due_min	17751
prev_days_last_due_count	16454
prev_days_termination_mean	17751
prev_days_termination_max	17751
prev_days_termination_min	17751
prev_days_termination_count	16454
prev_nflag_insured_on_approval_mean	17751
prev_nflag_insured_on_approval_max	17751
prev_nflag_insured_on_approval_min	17751
prev_nflag_insured_on_approval_count	16454
posf_sk_id_prev_mean	16454
posf_sk_id_prev_max	16454
posf_sk_id_prev_min	16454
posf_sk_id_prev_count	16454
posf_pos_sk_id_curr_mean_mean	20544
posf_pos_sk_id_curr_mean_max	20544
posf_pos_sk_id_curr_mean_min	20544

posf_pos_sk_id_curr_mean_count	16454
posf_pos_sk_id_curr_max_mean	20544
posf_pos_sk_id_curr_max_max	20544
posf_pos_sk_id_curr_max_min	20544
posf_pos_sk_id_curr_max_count	16454
posf_pos_sk_id_curr_min_mean	20544
posf_pos_sk_id_curr_min_max	20544
posf_pos_sk_id_curr_min_min	20544
posf_pos_sk_id_curr_min_count	16454
posf_pos_sk_id_curr_count_mean	20544
posf_pos_sk_id_curr_count_max	20544
posf_pos_sk_id_curr_count_min	20544
posf_pos_sk_id_curr_count_count	16454
posf_pos_months_balance_mean_mean	20544
posf_pos_months_balance_mean_max	20544
posf_pos_months_balance_mean_min	20544
posf_pos_months_balance_mean_count	16454
posf_pos_months_balance_max_mean	20544
posf_pos_months_balance_max_max	20544
posf_pos_months_balance_max_min	20544
posf_pos_months_balance_max_count	16454
posf_pos_months_balance_min_mean	20544
posf_pos_months_balance_min_max	20544
posf_pos_months_balance_min_min	20544
posf_pos_months_balance_min_count	16454
posf_pos_months_balance_count_mean	20544
posf_pos_months_balance_count_max	20544
posf_pos_months_balance_count_min	20544
posf_pos_months_balance_count_count	16454
posf_pos_cnt_instalment_mean_mean	20576
posf_pos_cnt_instalment_mean_max	20576
posf_pos_cnt_instalment_mean_min	20576
posf_pos_cnt_instalment_mean_count	16454
posf_pos_cnt_instalment_max_mean	20576
posf_pos_cnt_instalment_max_max	20576
posf_pos_cnt_instalment_max_min	20576
posf_pos_cnt_instalment_max_count	16454
posf_pos_cnt_instalment_min_mean	20576
posf_pos_cnt_instalment_min_max	20576
posf_pos_cnt_instalment_min_min	20576
posf_pos_cnt_instalment_min_count	16454
posf_pos_cnt_instalment_count_mean	20544
posf_pos_cnt_instalment_count_max	20544
posf_pos_cnt_instalment_count_min	20544
posf_pos_cnt_instalment_count_count	16454
posf_pos_cnt_instalment_future_mean_mean	20576
posf_pos_cnt_instalment_future_mean_max	20576
posf_pos_cnt_instalment_future_mean_min	20576
posf_pos_cnt_instalment_future_mean_count	16454
posf_pos_cnt_instalment_future_max_mean	20576
posf_pos_cnt_instalment_future_max_max	20576
posf_pos_cnt_instalment_future_max_min	20576
posf_pos_cnt_instalment_future_max_count	16454
posf_pos_cnt_instalment_future_min_mean	20576
posf_pos_cnt_instalment_future_min_max	20576
posf_pos_cnt_instalment_future_min_min	20576
posf_pos_cnt_instalment_future_min_count	16454
posf_pos_cnt_instalment_future_count_mean	20544
posf_pos_cnt_instalment_future_count_max	20544
posf_pos_cnt_instalment_future_count_min	20544
posf_pos_cnt_instalment_future_count_count	16454
posf_pos_sk_dpd_mean_mean	20544
posf_pos_sk_dpd_mean_max	20544
posf_pos_sk_dpd_mean_min	20544
posf_pos_sk_dpd_mean_count	16454
posf_pos_sk_dpd_max_mean	20544
posf_pos_sk_dpd_max_max	20544
posf_pos_sk_dpd_max_min	20544
posf_pos_sk_dpd_max_count	16454

posf_pos_sk_dpd_min_mean	20544
posf_pos_sk_dpd_min_max	20544
posf_pos_sk_dpd_min_min	20544
posf_pos_sk_dpd_min_count	16454
posf_pos_sk_dpd_count_mean	20544
posf_pos_sk_dpd_count_max	20544
posf_pos_sk_dpd_count_min	20544
posf_pos_sk_dpd_count_count	16454
posf_pos_sk_dpd_def_mean_mean	20544
posf_pos_sk_dpd_def_mean_max	20544
posf_pos_sk_dpd_def_mean_min	20544
posf_pos_sk_dpd_def_mean_count	16454
posf_pos_sk_dpd_def_max_mean	20544
posf_pos_sk_dpd_def_max_max	20544
posf_pos_sk_dpd_def_max_min	20544
posf_pos_sk_dpd_def_max_count	16454
posf_pos_sk_dpd_def_min_mean	20544
posf_pos_sk_dpd_def_min_max	20544
posf_pos_sk_dpd_def_min_min	20544
posf_pos_sk_dpd_def_min_count	16454
posf_pos_sk_dpd_def_count_mean	20544
posf_pos_sk_dpd_def_count_max	20544
posf_pos_sk_dpd_def_count_min	20544
posf_pos_sk_dpd_def_count_count	16454
ccf_sk_id_prev_mean	16454
ccf_sk_id_prev_max	16454
ccf_sk_id_prev_min	16454
ccf_sk_id_prev_count	16454
ccf_cc_sk_id_curr_mean_mean	229577
ccf_cc_sk_id_curr_mean_max	229577
ccf_cc_sk_id_curr_mean_min	229577
ccf_cc_sk_id_curr_mean_count	16454
ccf_cc_sk_id_curr_max_mean	229577
ccf_cc_sk_id_curr_max_max	229577
ccf_cc_sk_id_curr_max_min	229577
ccf_cc_sk_id_curr_max_count	16454
ccf_cc_sk_id_curr_min_mean	229577
ccf_cc_sk_id_curr_min_max	229577
ccf_cc_sk_id_curr_min_min	229577
ccf_cc_sk_id_curr_min_count	16454
ccf_cc_sk_id_curr_count_mean	229577
ccf_cc_sk_id_curr_count_max	229577
ccf_cc_sk_id_curr_count_min	229577
ccf_cc_sk_id_curr_count_count	16454
ccf_cc_months_balance_mean_mean	229577
ccf_cc_months_balance_mean_max	229577
ccf_cc_months_balance_mean_min	229577
ccf_cc_months_balance_mean_count	16454
ccf_cc_months_balance_max_mean	229577
ccf_cc_months_balance_max_max	229577
ccf_cc_months_balance_max_min	229577
ccf_cc_months_balance_max_count	16454
ccf_cc_months_balance_min_mean	229577
ccf_cc_months_balance_min_max	229577
ccf_cc_months_balance_min_min	229577
ccf_cc_months_balance_min_count	16454
ccf_cc_months_balance_count_mean	229577
ccf_cc_months_balance_count_max	229577
ccf_cc_months_balance_count_min	229577
ccf_cc_months_balance_count_count	16454
ccf_cc_amt_balance_mean_mean	229577
ccf_cc_amt_balance_mean_max	229577
ccf_cc_amt_balance_mean_min	229577
ccf_cc_amt_balance_mean_count	16454
ccf_cc_amt_balance_max_mean	229577
ccf_cc_amt_balance_max_max	229577
ccf_cc_amt_balance_max_min	229577
ccf_cc_amt_balance_max_count	16454
ccf_cc_amt_balance_min_mean	229577

ccf_cc_amt_balance_min_max	229577
ccf_cc_amt_balance_min_min	229577
ccf_cc_amt_balance_min_count	16454
ccf_cc_amt_balance_count_mean	229577
ccf_cc_amt_balance_count_max	229577
ccf_cc_amt_balance_count_min	229577
ccf_cc_amt_balance_count_count	16454
ccf_cc_amt_credit_limit_actual_mean_mean	229577
ccf_cc_amt_credit_limit_actual_mean_max	229577
ccf_cc_amt_credit_limit_actual_mean_min	229577
ccf_cc_amt_credit_limit_actual_mean_count	16454
ccf_cc_amt_credit_limit_actual_max_mean	229577
ccf_cc_amt_credit_limit_actual_max_max	229577
ccf_cc_amt_credit_limit_actual_max_min	229577
ccf_cc_amt_credit_limit_actual_max_count	16454
ccf_cc_amt_credit_limit_actual_min_mean	229577
ccf_cc_amt_credit_limit_actual_min_max	229577
ccf_cc_amt_credit_limit_actual_min_min	229577
ccf_cc_amt_credit_limit_actual_min_count	16454
ccf_cc_amt_credit_limit_actual_count_mean	229577
ccf_cc_amt_credit_limit_actual_count_max	229577
ccf_cc_amt_credit_limit_actual_count_min	229577
ccf_cc_amt_credit_limit_actual_count_count	16454
ccf_cc_amt_drawings_atm_current_mean_mean	254581
ccf_cc_amt_drawings_atm_current_mean_max	254581
ccf_cc_amt_drawings_atm_current_mean_min	254581
ccf_cc_amt_drawings_atm_current_mean_count	16454
ccf_cc_amt_drawings_atm_current_max_mean	254581
ccf_cc_amt_drawings_atm_current_max_max	254581
ccf_cc_amt_drawings_atm_current_max_min	254581
ccf_cc_amt_drawings_atm_current_max_count	16454
ccf_cc_amt_drawings_atm_current_min_mean	254581
ccf_cc_amt_drawings_atm_current_min_max	254581
ccf_cc_amt_drawings_atm_current_min_min	254581
ccf_cc_amt_drawings_atm_current_min_count	16454
ccf_cc_amt_drawings_atm_current_count_mean	229577
ccf_cc_amt_drawings_atm_current_count_max	229577
ccf_cc_amt_drawings_atm_current_count_min	229577
ccf_cc_amt_drawings_atm_current_count_count	16454
ccf_cc_amt_drawings_current_mean_mean	229577
ccf_cc_amt_drawings_current_mean_max	229577
ccf_cc_amt_drawings_current_mean_min	229577
ccf_cc_amt_drawings_current_mean_count	16454
ccf_cc_amt_drawings_current_max_mean	229577
ccf_cc_amt_drawings_current_max_max	229577
ccf_cc_amt_drawings_current_max_min	229577
ccf_cc_amt_drawings_current_max_count	16454
ccf_cc_amt_drawings_current_min_mean	229577
ccf_cc_amt_drawings_current_min_max	229577
ccf_cc_amt_drawings_current_min_min	229577
ccf_cc_amt_drawings_current_min_count	16454
ccf_cc_amt_drawings_current_count_mean	229577
ccf_cc_amt_drawings_current_count_max	229577
ccf_cc_amt_drawings_current_count_min	229577
ccf_cc_amt_drawings_current_count_count	16454
ccf_cc_amt_drawings_other_current_mean_mean	254581
ccf_cc_amt_drawings_other_current_mean_max	254581
ccf_cc_amt_drawings_other_current_mean_min	254581
ccf_cc_amt_drawings_other_current_mean_count	16454
ccf_cc_amt_drawings_other_current_max_mean	254581
ccf_cc_amt_drawings_other_current_max_max	254581
ccf_cc_amt_drawings_other_current_max_min	254581
ccf_cc_amt_drawings_other_current_max_count	16454
ccf_cc_amt_drawings_other_current_min_mean	254581
ccf_cc_amt_drawings_other_current_min_max	254581
ccf_cc_amt_drawings_other_current_min_min	254581
ccf_cc_amt_drawings_other_current_min_count	16454
ccf_cc_amt_drawings_other_current_mean_mean	229577
ccf_cc_amt_drawings_other_current_mean_max	229577

ccf_cc_amt_drawings_other_current_count_min	229577
ccf_cc_amt_drawings_other_current_count_count	16454
ccf_cc_amt_drawings_pos_current_mean_mean	254581
ccf_cc_amt_drawings_pos_current_mean_max	254581
ccf_cc_amt_drawings_pos_current_mean_min	254581
ccf_cc_amt_drawings_pos_current_mean_count	16454
ccf_cc_amt_drawings_pos_current_max_mean	254581
ccf_cc_amt_drawings_pos_current_max_max	254581
ccf_cc_amt_drawings_pos_current_max_min	254581
ccf_cc_amt_drawings_pos_current_max_count	16454
ccf_cc_amt_drawings_pos_current_min_mean	254581
ccf_cc_amt_drawings_pos_current_min_max	254581
ccf_cc_amt_drawings_pos_current_min_min	254581
ccf_cc_amt_drawings_pos_current_min_count	16454
ccf_cc_amt_drawings_pos_current_count_mean	229577
ccf_cc_amt_drawings_pos_current_count_max	229577
ccf_cc_amt_drawings_pos_current_count_min	229577
ccf_cc_amt_drawings_pos_current_count_count	16454
ccf_cc_amt_inst_min_regularity_mean_mean	229577
ccf_cc_amt_inst_min_regularity_mean_max	229577
ccf_cc_amt_inst_min_regularity_mean_min	229577
ccf_cc_amt_inst_min_regularity_mean_count	16454
ccf_cc_amt_inst_min_regularity_max_mean	229577
ccf_cc_amt_inst_min_regularity_max_max	229577
ccf_cc_amt_inst_min_regularity_max_min	229577
ccf_cc_amt_inst_min_regularity_max_count	16454
ccf_cc_amt_inst_min_regularity_min_mean	229577
ccf_cc_amt_inst_min_regularity_min_max	229577
ccf_cc_amt_inst_min_regularity_min_min	229577
ccf_cc_amt_inst_min_regularity_min_count	16454
ccf_cc_amt_inst_min_regularity_count_mean	229577
ccf_cc_amt_inst_min_regularity_count_max	229577
ccf_cc_amt_inst_min_regularity_count_min	229577
ccf_cc_amt_inst_min_regularity_count_count	16454
ccf_cc_amt_payment_current_mean_mean	254669
ccf_cc_amt_payment_current_mean_max	254669
ccf_cc_amt_payment_current_mean_min	254669
ccf_cc_amt_payment_current_mean_count	16454
ccf_cc_amt_payment_current_max_mean	254669
ccf_cc_amt_payment_current_max_max	254669
ccf_cc_amt_payment_current_max_min	254669
ccf_cc_amt_payment_current_max_count	16454
ccf_cc_amt_payment_current_min_mean	254669
ccf_cc_amt_payment_current_min_max	254669
ccf_cc_amt_payment_current_min_min	254669
ccf_cc_amt_payment_current_min_count	16454
ccf_cc_amt_payment_current_count_mean	229577
ccf_cc_amt_payment_current_count_max	229577
ccf_cc_amt_payment_current_count_min	229577
ccf_cc_amt_payment_current_count_count	16454
ccf_cc_amt_payment_total_current_mean_mean	229577
ccf_cc_amt_payment_total_current_mean_max	229577
ccf_cc_amt_payment_total_current_mean_min	229577
ccf_cc_amt_payment_total_current_mean_count	16454
ccf_cc_amt_payment_total_current_max_mean	229577
ccf_cc_amt_payment_total_current_max_max	229577
ccf_cc_amt_payment_total_current_max_min	229577
ccf_cc_amt_payment_total_current_max_count	16454
ccf_cc_amt_payment_total_current_min_mean	229577
ccf_cc_amt_payment_total_current_min_max	229577
ccf_cc_amt_payment_total_current_min_min	229577
ccf_cc_amt_payment_total_current_min_count	16454
ccf_cc_amt_payment_total_current_mean_mean	229577
ccf_cc_amt_payment_total_current_mean_max	229577
ccf_cc_amt_payment_total_current_mean_min	229577
ccf_cc_amt_payment_total_current_count_mean	16454
ccf_cc_amt_payment_total_current_count_max	229577
ccf_cc_amt_payment_total_current_count_min	229577
ccf_cc_amt_payment_total_current_count_count	16454
ccf_cc_amt_receivable_principal_mean_mean	229577
ccf_cc_amt_receivable_principal_mean_max	229577
ccf_cc_amt_receivable_principal_mean_min	229577

ccf_cc_amt_receivable_principal_mean_count	16454
ccf_cc_amt_receivable_principal_max_mean	229577
ccf_cc_amt_receivable_principal_max_max	229577
ccf_cc_amt_receivable_principal_max_min	229577
ccf_cc_amt_receivable_principal_max_count	16454
ccf_cc_amt_receivable_principal_min_mean	229577
ccf_cc_amt_receivable_principal_min_max	229577
ccf_cc_amt_receivable_principal_min_min	229577
ccf_cc_amt_receivable_principal_min_count	16454
ccf_cc_amt_receivable_principal_count_mean	229577
ccf_cc_amt_receivable_principal_count_max	229577
ccf_cc_amt_receivable_principal_count_min	229577
ccf_cc_amt_receivable_principal_count_count	16454
ccf_cc_amt_recivable_mean_mean	229577
ccf_cc_amt_recivable_mean_max	229577
ccf_cc_amt_recivable_mean_min	229577
ccf_cc_amt_recivable_mean_count	16454
ccf_cc_amt_recivable_max_mean	229577
ccf_cc_amt_recivable_max_max	229577
ccf_cc_amt_recivable_max_min	229577
ccf_cc_amt_recivable_max_count	16454
ccf_cc_amt_recivable_min_mean	229577
ccf_cc_amt_recivable_min_max	229577
ccf_cc_amt_recivable_min_min	229577
ccf_cc_amt_recivable_min_count	16454
ccf_cc_amt_recivable_count_mean	229577
ccf_cc_amt_recivable_count_max	229577
ccf_cc_amt_recivable_count_min	229577
ccf_cc_amt_recivable_count_count	16454
ccf_cc_amt_total_receivable_mean_mean	229577
ccf_cc_amt_total_receivable_mean_max	229577
ccf_cc_amt_total_receivable_mean_min	229577
ccf_cc_amt_total_receivable_mean_count	16454
ccf_cc_amt_total_receivable_max_mean	229577
ccf_cc_amt_total_receivable_max_max	229577
ccf_cc_amt_total_receivable_max_min	229577
ccf_cc_amt_total_receivable_max_count	16454
ccf_cc_amt_total_receivable_min_mean	229577
ccf_cc_amt_total_receivable_min_max	229577
ccf_cc_amt_total_receivable_min_min	229577
ccf_cc_amt_total_receivable_min_count	16454
ccf_cc_amt_total_receivable_count_mean	229577
ccf_cc_amt_total_receivable_count_max	229577
ccf_cc_amt_total_receivable_count_min	229577
ccf_cc_amt_total_receivable_count_count	16454
ccf_cc_cnt_drawings_atm_current_mean_mean	254581
ccf_cc_cnt_drawings_atm_current_mean_max	254581
ccf_cc_cnt_drawings_atm_current_mean_min	254581
ccf_cc_cnt_drawings_atm_current_mean_count	16454
ccf_cc_cnt_drawings_atm_current_max_mean	254581
ccf_cc_cnt_drawings_atm_current_max_max	254581
ccf_cc_cnt_drawings_atm_current_max_min	254581
ccf_cc_cnt_drawings_atm_current_max_count	16454
ccf_cc_cnt_drawings_atm_current_min_mean	254581
ccf_cc_cnt_drawings_atm_current_min_max	254581
ccf_cc_cnt_drawings_atm_current_min_min	254581
ccf_cc_cnt_drawings_atm_current_min_count	16454
ccf_cc_cnt_drawings_atm_current_count_mean	229577
ccf_cc_cnt_drawings_atm_current_count_max	229577
ccf_cc_cnt_drawings_atm_current_count_min	229577
ccf_cc_cnt_drawings_atm_current_count_count	16454
ccf_cc_cnt_drawings_current_mean_mean	229577
ccf_cc_cnt_drawings_current_mean_max	229577
ccf_cc_cnt_drawings_current_mean_min	229577
ccf_cc_cnt_drawings_current_mean_count	16454
ccf_cc_cnt_drawings_current_max_mean	229577
ccf_cc_cnt_drawings_current_max_max	229577
ccf_cc_cnt_drawings_current_max_min	229577
ccf_cc_cnt_drawings_current_max_count	16454

ccf_cc_cnt_drawings_current_min_mean	229577
ccf_cc_cnt_drawings_current_min_max	229577
ccf_cc_cnt_drawings_current_min_min	229577
ccf_cc_cnt_drawings_current_min_count	16454
ccf_cc_cnt_drawings_current_count_mean	229577
ccf_cc_cnt_drawings_current_count_max	229577
ccf_cc_cnt_drawings_current_count_min	229577
ccf_cc_cnt_drawings_current_count_count	16454
ccf_cc_cnt_drawings_other_current_mean_mean	254581
ccf_cc_cnt_drawings_other_current_mean_max	254581
ccf_cc_cnt_drawings_other_current_mean_min	254581
ccf_cc_cnt_drawings_other_current_mean_count	16454
ccf_cc_cnt_drawings_other_current_max_mean	254581
ccf_cc_cnt_drawings_other_current_max_max	254581
ccf_cc_cnt_drawings_other_current_max_min	254581
ccf_cc_cnt_drawings_other_current_max_count	16454
ccf_cc_cnt_drawings_other_current_min_mean	254581
ccf_cc_cnt_drawings_other_current_min_max	254581
ccf_cc_cnt_drawings_other_current_min_min	254581
ccf_cc_cnt_drawings_other_current_min_count	16454
ccf_cc_cnt_drawings_other_current_count_mean	229577
ccf_cc_cnt_drawings_other_current_count_max	229577
ccf_cc_cnt_drawings_other_current_count_min	229577
ccf_cc_cnt_drawings_other_current_count_count	16454
ccf_cc_cnt_drawings_pos_current_mean_mean	254581
ccf_cc_cnt_drawings_pos_current_mean_max	254581
ccf_cc_cnt_drawings_pos_current_mean_min	254581
ccf_cc_cnt_drawings_pos_current_mean_count	16454
ccf_cc_cnt_drawings_pos_current_max_mean	254581
ccf_cc_cnt_drawings_pos_current_max_max	254581
ccf_cc_cnt_drawings_pos_current_max_min	254581
ccf_cc_cnt_drawings_pos_current_max_count	16454
ccf_cc_cnt_drawings_pos_current_min_mean	254581
ccf_cc_cnt_drawings_pos_current_min_max	254581
ccf_cc_cnt_drawings_pos_current_min_min	254581
ccf_cc_cnt_drawings_pos_current_min_count	16454
ccf_cc_cnt_drawings_pos_current_count_mean	229577
ccf_cc_cnt_drawings_pos_current_count_max	229577
ccf_cc_cnt_drawings_pos_current_count_min	229577
ccf_cc_cnt_drawings_pos_current_count_count	16454
ccf_cc_instalment_mature_cum_mean_mean	229577
ccf_cc_instalment_mature_cum_mean_max	229577
ccf_cc_instalment_mature_cum_mean_min	229577
ccf_cc_instalment_mature_cum_mean_count	16454
ccf_cc_instalment_mature_cum_max_mean	229577
ccf_cc_instalment_mature_cum_max_max	229577
ccf_cc_instalment_mature_cum_max_min	229577
ccf_cc_instalment_mature_cum_max_count	16454
ccf_cc_instalment_mature_cum_min_mean	229577
ccf_cc_instalment_mature_cum_min_max	229577
ccf_cc_instalment_mature_cum_min_min	229577
ccf_cc_instalment_mature_cum_min_count	16454
ccf_cc_instalment_mature_cum_count_mean	229577
ccf_cc_instalment_mature_cum_count_max	229577
ccf_cc_instalment_mature_cum_count_min	229577
ccf_cc_instalment_mature_cum_count_count	16454
ccf_cc_sk_dpd_mean_mean	229577
ccf_cc_sk_dpd_mean_max	229577
ccf_cc_sk_dpd_mean_min	229577
ccf_cc_sk_dpd_mean_count	16454
ccf_cc_sk_dpd_max_mean	229577
ccf_cc_sk_dpd_max_max	229577
ccf_cc_sk_dpd_max_min	229577
ccf_cc_sk_dpd_max_count	16454
ccf_cc_sk_dpd_min_mean	229577
ccf_cc_sk_dpd_min_max	229577
ccf_cc_sk_dpd_min_min	229577
ccf_cc_sk_dpd_min_count	16454
ccf_cc_sk_dpd_count_mean	229577

ccf_cc_sk_dpd_count_max	229577
ccf_cc_sk_dpd_count_min	229577
ccf_cc_sk_dpd_count_count	16454
ccf_cc_sk_dpd_def_mean_mean	229577
ccf_cc_sk_dpd_def_mean_max	229577
ccf_cc_sk_dpd_def_mean_min	229577
ccf_cc_sk_dpd_def_mean_count	16454
ccf_cc_sk_dpd_def_max_mean	229577
ccf_cc_sk_dpd_def_max_max	229577
ccf_cc_sk_dpd_def_max_min	229577
ccf_cc_sk_dpd_def_max_count	16454
ccf_cc_sk_dpd_def_min_mean	229577
ccf_cc_sk_dpd_def_min_max	229577
ccf_cc_sk_dpd_def_min_min	229577
ccf_cc_sk_dpd_def_min_count	16454
ccf_cc_sk_dpd_def_count_mean	229577
ccf_cc_sk_dpd_def_count_max	229577
ccf_cc_sk_dpd_def_count_min	229577
ccf_cc_sk_dpd_def_count_count	16454
instf_sk_id_prev_mean	16454
instf_sk_id_prev_max	16454
instf_sk_id_prev_min	16454
instf_sk_id_prev_count	16454
instf_inst_sk_id_curr_mean_mean	18105
instf_inst_sk_id_curr_mean_max	18105
instf_inst_sk_id_curr_mean_min	18105
instf_inst_sk_id_curr_mean_count	16454
instf_inst_sk_id_curr_max_mean	18105
instf_inst_sk_id_curr_max_max	18105
instf_inst_sk_id_curr_max_min	18105
instf_inst_sk_id_curr_max_count	16454
instf_inst_sk_id_curr_min_mean	18105
instf_inst_sk_id_curr_min_max	18105
instf_inst_sk_id_curr_min_min	18105
instf_inst_sk_id_curr_min_count	16454
instf_inst_sk_id_curr_count_mean	18105
instf_inst_sk_id_curr_count_max	18105
instf_inst_sk_id_curr_count_min	18105
instf_inst_sk_id_curr_count_count	16454
instf_inst_num_instalment_version_mean_mean	18105
instf_inst_num_instalment_version_mean_max	18105
instf_inst_num_instalment_version_mean_min	18105
instf_inst_num_instalment_version_mean_count	16454
instf_inst_num_instalment_version_max_mean	18105
instf_inst_num_instalment_version_max_max	18105
instf_inst_num_instalment_version_max_min	18105
instf_inst_num_instalment_version_max_count	16454
instf_inst_num_instalment_version_min_mean	18105
instf_inst_num_instalment_version_min_max	18105
instf_inst_num_instalment_version_min_min	18105
instf_inst_num_instalment_version_min_count	16454
instf_inst_num_instalment_version_count_mean	18105
instf_inst_num_instalment_version_count_max	18105
instf_inst_num_instalment_version_count_min	18105
instf_inst_num_instalment_version_count_count	16454
instf_inst_num_instalment_number_mean_mean	18105
instf_inst_num_instalment_number_mean_max	18105
instf_inst_num_instalment_number_mean_min	18105
instf_inst_num_instalment_number_mean_count	16454
instf_inst_num_instalment_number_max_mean	18105
instf_inst_num_instalment_number_max_max	18105
instf_inst_num_instalment_number_max_min	18105
instf_inst_num_instalment_number_max_count	16454
instf_inst_num_instalment_number_min_mean	18105
instf_inst_num_instalment_number_min_max	18105
instf_inst_num_instalment_number_min_min	18105
instf_inst_num_instalment_number_min_count	16454
instf_inst_num_instalment_number_count_mean	18105
instf_inst_num_instalment_number_count_max	18105

instf_inst_num_instalment_number_count_min	18105
instf_inst_num_instalment_number_count_count	16454
instf_inst_days_instalment_mean_mean	18105
instf_inst_days_instalment_mean_max	18105
instf_inst_days_instalment_mean_min	18105
instf_inst_days_instalment_mean_count	16454
instf_inst_days_instalment_max_mean	18105
instf_inst_days_instalment_max_max	18105
instf_inst_days_instalment_max_min	18105
instf_inst_days_instalment_max_count	16454
instf_inst_days_instalment_min_mean	18105
instf_inst_days_instalment_min_max	18105
instf_inst_days_instalment_min_min	18105
instf_inst_days_instalment_min_count	16454
instf_inst_days_instalment_count_mean	18105
instf_inst_days_instalment_count_max	18105
instf_inst_days_instalment_count_min	18105
instf_inst_days_instalment_count_count	16454
instf_inst_days_entry_payment_mean_mean	18113
instf_inst_days_entry_payment_mean_max	18113
instf_inst_days_entry_payment_mean_min	18113
instf_inst_days_entry_payment_mean_count	16454
instf_inst_days_entry_payment_max_mean	18113
instf_inst_days_entry_payment_max_max	18113
instf_inst_days_entry_payment_max_min	18113
instf_inst_days_entry_payment_max_count	16454
instf_inst_days_entry_payment_min_mean	18113
instf_inst_days_entry_payment_min_max	18113
instf_inst_days_entry_payment_min_min	18113
instf_inst_days_entry_payment_min_count	16454
instf_inst_days_entry_payment_count_mean	18105
instf_inst_days_entry_payment_count_max	18105
instf_inst_days_entry_payment_count_min	18105
instf_inst_days_entry_payment_count_count	16454
instf_inst_amt_instalment_mean_mean	18105
instf_inst_amt_instalment_mean_max	18105
instf_inst_amt_instalment_mean_min	18105
instf_inst_amt_instalment_mean_count	16454
instf_inst_amt_instalment_max_mean	18105
instf_inst_amt_instalment_max_max	18105
instf_inst_amt_instalment_max_min	18105
instf_inst_amt_instalment_max_count	16454
instf_inst_amt_instalment_min_mean	18105
instf_inst_amt_instalment_min_max	18105
instf_inst_amt_instalment_min_min	18105
instf_inst_amt_instalment_min_count	16454
instf_inst_amt_instalment_count_mean	18105
instf_inst_amt_instalment_count_max	18105
instf_inst_amt_instalment_count_min	18105
instf_inst_amt_instalment_count_count	16454
instf_inst_amt_payment_mean_mean	18113
instf_inst_amt_payment_mean_max	18113
instf_inst_amt_payment_mean_min	18113
instf_inst_amt_payment_mean_count	16454
instf_inst_amt_payment_max_mean	18113
instf_inst_amt_payment_max_max	18113
instf_inst_amt_payment_max_min	18113
instf_inst_amt_payment_max_count	16454
instf_inst_amt_payment_min_mean	18113
instf_inst_amt_payment_min_max	18113
instf_inst_amt_payment_min_min	18113
instf_inst_amt_payment_min_count	16454
instf_inst_amt_payment_count_mean	18105
instf_inst_amt_payment_count_max	18105
instf_inst_amt_payment_count_min	18105
instf_inst_amt_payment_count_count	16454

5.4.1 Preenchendo Valores Ausentes

```
In [11]: aux_missing_cols = {}

aux_missing_cols['is_building'] = df1.filter(like='commonarea_').notna().any(axis=1).z
aux_missing_cols['is_building_missing'] = (~df1.filter(like='commonarea_').notna().any(axis=1))
aux_missing_cols['has_basement'] = df1['basementarea_avg'].notna().astype(int)
aux_missing_cols['basementarea_missing'] = df1['basementarea_avg'].isna().astype(int)
aux_missing_cols['has_noliving'] = df1['nonlivingarea_avg'].notna().astype(int)
aux_missing_cols['nonlivingarea_missing'] = df1['nonlivingarea_avg'].isna().astype(int)
aux_missing_cols['is_employed'] = (~df1['days_employed'].isna()).astype(int)

aux_missing_cols['livingarea_missing'] = df1['livingarea_avg'].isna().astype(int)
aux_missing_cols['apartments_missing'] = df1['apartments_avg'].isna().astype(int)
aux_missing_cols['floorsmax_missing'] = df1['floorsmax_avg'].isna().astype(int)
aux_missing_cols['years_build_missing'] = df1['years_build_avg'].isna().astype(int)
aux_missing_cols['commonarea_missing'] = df1['commonarea_avg'].isna().astype(int)

aux_missing_cols['emergencystate_missing'] = df1['emergencystate_mode'].isna().astype(int)
df1['emergencystate_mode'] = df1['emergencystate_mode'].fillna(0)
for col in ['elevators_avg', 'elevators_mode', 'elevators_medi']:
    aux_missing_cols[col + "_missing"] = df1[col].isna().astype(int)
    df1[col] = df1[col].fillna(0)

cols_moda = [
    'housetype_mode', 'wallsmaterial_mode', 'occupation_type', 'organization_type'
]
for col in cols_moda:
    aux_missing_cols[col + '_missing'] = df1[col].isna().astype(int)

df1['housetype_mode'] = df1['housetype_mode'].fillna('Unknown')
df1['wallsmaterial_mode'] = df1['wallsmaterial_mode'].fillna('Unknown')
df1['occupation_type'] = df1['occupation_type'].fillna('Other')
df1['organization_type'] = df1['organization_type'].fillna('Unknown')

drop_columns = {
    'fondkapremont_mode', 'landarea_mode', 'landarea_medi',
    'livingarea_mode', 'livingarea_medi',
    'entrances_avg', 'entrances_mode', 'entrances_medi',
    'floorsmax_mode', 'floorsmax_medi',
    'apartments_mode', 'apartments_medi', 'years_build_mode', 'years_build_medi'
}

do_not_drop = {
    'livingarea_avg', 'apartments_avg', 'floorsmax_avg', 'years_build_avg',
    'basementarea_avg', 'nonlivingarea_avg', 'livingapartments_missing',
    'own_car_age', 'years_beginexpluatation_avg', 'commonarea_avg', 'nonlivingapartments_avg',
}

prefixes = (
    'nonlivingapartments_', 'livingapartments_', 'floorsmin_', 'commonarea_', 'basementarea_',
    'nonlivingarea_', 'years_beginexpluatation_'
)

cols_to_drop = [
    col for col in df1.columns
    if ((col.startswith(prefixes) and col not in do_not_drop) or col in drop_columns)
]

df1 = df1.drop(columns=cols_to_drop)
```

```
In [12]: aux_missing_cols['name_type_suite_missing'] = df1['name_type_suite'].isna().astype(int)
mode_suite = df1["name_type_suite"].mode()[0]
df1["name_type_suite"] = df1["name_type_suite"].fillna(mode_suite)

aux_missing_cols['code_gender_missing'] = df1['code_gender'].isna().astype(int)
mode_gender = df1["code_gender"].mode()[0]
df1["code_gender"] = df1["code_gender"].fillna(mode_gender)
```

```
cols_nan_remanescentes = df1.columns[df1.isnull().any()]
df_mediana = pd.DataFrame(columns=['coluna', 'mediana'])
medianas = []
for col_nome in cols_nan_remanescentes:
    aux_missing_cols[col_nome + '_missing'] = df1[col_nome].isna().astype(int)
    mediana = df1[col_nome].median()
    medianas.append({'coluna': col_nome, 'mediana': mediana})
    df1[col_nome] = df1[col_nome].fillna(mediana)
df_mediana = pd.DataFrame(medianas)
```

```
In [13]: df1 = pd.concat([df1, pd.DataFrame(aux_missing_cols, index=df1.index)], axis=1)
df1 = df1.copy()
```

5.4.2 Checando Valores Ausentes Após Modificações

```
In [14]: print(df1.isna().sum().to_string())
```

target	0
name_contract_type	0
code_gender	0
flag_own_car	0
flag_own_realty	0
cnt_children	0
amt_income_total	0
amt_credit	0
amt_annuity	0
amt_goods_price	0
name_type_suite	0
name_income_type	0
name_education_type	0
name_family_status	0
name_housing_type	0
region_population_relative	0
days_birth	0
days_employed	0
days_registration	0
days_id_publish	0
own_car_age	0
flag_mobil	0
flag_emp_phone	0
flag_work_phone	0
flag_cont_mobile	0
flag_phone	0
flag_email	0
occupation_type	0
cnt_fam_members	0
region_rating_client	0
region_rating_client_w_city	0
weekday_appr_process_start	0
hour_appr_process_start	0
reg_region_not_live_region	0
reg_region_not_work_region	0
live_region_not_work_region	0
reg_city_not_live_city	0
reg_city_not_work_city	0
live_city_not_work_city	0
organization_type	0
ext_source_1	0
ext_source_2	0
ext_source_3	0
apartments_avg	0
basementarea_avg	0
years_beginexpluatation_avg	0
years_build_avg	0
commonarea_avg	0
elevators_avg	0
floorsmax_avg	0
landarea_avg	0
livingapartments_avg	0
livingarea_avg	0
nonlivingapartments_avg	0
nonlivingarea_avg	0
elevators_mode	0
elevators_medi	0
housetype_mode	0
totalarea_mode	0
wallsmaterial_mode	0
emergencystate_mode	0
obs_30_cnt_social_circle	0
def_30_cnt_social_circle	0
obs_60_cnt_social_circle	0
def_60_cnt_social_circle	0
days_last_phone_change	0
flag_document_2	0
flag_document_3	0
flag_document_4	0

flag_document_5	0
flag_document_6	0
flag_document_7	0
flag_document_8	0
flag_document_9	0
flag_document_10	0
flag_document_11	0
flag_document_12	0
flag_document_13	0
flag_document_14	0
flag_document_15	0
flag_document_16	0
flag_document_17	0
flag_document_18	0
flag_document_19	0
flag_document_20	0
flag_document_21	0
amt_req_credit_bureau_hour	0
amt_req_credit_bureau_day	0
amt_req_credit_bureau_week	0
amt_req_credit_bureau_mon	0
amt_req_credit_bureau_qrt	0
amt_req_credit_bureau_year	0
bur_sk_id_bureau_mean	0
bur_sk_id_bureau_max	0
bur_sk_id_bureau_min	0
bur_sk_id_bureau_count	0
bur_days_credit_mean	0
bur_days_credit_max	0
bur_days_credit_min	0
bur_days_credit_count	0
bur_credit_day_overdue_mean	0
bur_credit_day_overdue_max	0
bur_credit_day_overdue_min	0
bur_credit_day_overdue_count	0
bur_days_credit_enddate_mean	0
bur_days_credit_enddate_max	0
bur_days_credit_enddate_min	0
bur_days_credit_enddate_count	0
bur_days_enddate_fact_mean	0
bur_days_enddate_fact_max	0
bur_days_enddate_fact_min	0
bur_days_enddate_fact_count	0
bur_amt_credit_max_overdue_mean	0
bur_amt_credit_max_overdue_max	0
bur_amt_credit_max_overdue_min	0
bur_amt_credit_max_overdue_count	0
bur_cnt_credit_prolong_mean	0
bur_cnt_credit_prolong_max	0
bur_cnt_credit_prolong_min	0
bur_cnt_credit_prolong_count	0
bur_amt_credit_sum_mean	0
bur_amt_credit_sum_max	0
bur_amt_credit_sum_min	0
bur_amt_credit_sum_count	0
bur_amt_credit_sum_debt_mean	0
bur_amt_credit_sum_debt_max	0
bur_amt_credit_sum_debt_min	0
bur_amt_credit_sum_debt_count	0
bur_amt_credit_sum_limit_mean	0
bur_amt_credit_sum_limit_max	0
bur_amt_credit_sum_limit_min	0
bur_amt_credit_sum_limit_count	0
bur_amt_credit_sum_overdue_mean	0
bur_amt_credit_sum_overdue_max	0
bur_amt_credit_sum_overdue_min	0
bur_amt_credit_sum_overdue_count	0
bur_days_credit_update_mean	0
bur_days_credit_update_max	0

bur_days_credit_update_min	0
bur_days_credit_update_count	0
bur_amt_annuity_mean	0
bur_amt_annuity_max	0
bur_amt_annuity_min	0
bur_amt_annuity_count	0
bur_bb_months_balance_mean_mean	0
bur_bb_months_balance_mean_max	0
bur_bb_months_balance_mean_min	0
bur_bb_months_balance_mean_count	0
bur_bb_months_balance_max_mean	0
bur_bb_months_balance_max_max	0
bur_bb_months_balance_max_min	0
bur_bb_months_balance_max_count	0
bur_bb_months_balance_min_mean	0
bur_bb_months_balance_min_max	0
bur_bb_months_balance_min_min	0
bur_bb_months_balance_min_count	0
bur_bb_months_balance_count_mean	0
bur_bb_months_balance_count_max	0
bur_bb_months_balance_count_min	0
bur_bb_months_balance_count_count	0
prev_sk_id_prev_mean	0
prev_sk_id_prev_max	0
prev_sk_id_prev_min	0
prev_sk_id_prev_count	0
prev_amt_annuity_mean	0
prev_amt_annuity_max	0
prev_amt_annuity_min	0
prev_amt_annuity_count	0
prev_amt_application_mean	0
prev_amt_application_max	0
prev_amt_application_min	0
prev_amt_application_count	0
prev_amt_credit_mean	0
prev_amt_credit_max	0
prev_amt_credit_min	0
prev_amt_credit_count	0
prev_amt_down_payment_mean	0
prev_amt_down_payment_max	0
prev_amt_down_payment_min	0
prev_amt_down_payment_count	0
prev_amt_goods_price_mean	0
prev_amt_goods_price_max	0
prev_amt_goods_price_min	0
prev_amt_goods_price_count	0
prev_hour_appr_process_start_mean	0
prev_hour_appr_process_start_max	0
prev_hour_appr_process_start_min	0
prev_hour_appr_process_start_count	0
prev_nflag_last_appl_in_day_mean	0
prev_nflag_last_appl_in_day_max	0
prev_nflag_last_appl_in_day_min	0
prev_nflag_last_appl_in_day_count	0
prev_rate_down_payment_mean	0
prev_rate_down_payment_max	0
prev_rate_down_payment_min	0
prev_rate_down_payment_count	0
prev_rate_interest_primary_mean	0
prev_rate_interest_primary_max	0
prev_rate_interest_primary_min	0
prev_rate_interest_primary_count	0
prev_rate_interest_privileged_mean	0
prev_rate_interest_privileged_max	0
prev_rate_interest_privileged_min	0
prev_rate_interest_privileged_count	0
prev_days_decision_mean	0
prev_days_decision_max	0
prev_days_decision_min	0

prev_days_decision_count	0
prev_sellerplace_area_mean	0
prev_sellerplace_area_max	0
prev_sellerplace_area_min	0
prev_sellerplace_area_count	0
prev_cnt_payment_mean	0
prev_cnt_payment_max	0
prev_cnt_payment_min	0
prev_cnt_payment_count	0
prev_days_first_drawing_mean	0
prev_days_first_drawing_max	0
prev_days_first_drawing_min	0
prev_days_first_drawing_count	0
prev_days_first_due_mean	0
prev_days_first_due_max	0
prev_days_first_due_min	0
prev_days_first_due_count	0
prev_days_last_due_1st_version_mean	0
prev_days_last_due_1st_version_max	0
prev_days_last_due_1st_version_min	0
prev_days_last_due_1st_version_count	0
prev_days_last_due_mean	0
prev_days_last_due_max	0
prev_days_last_due_min	0
prev_days_last_due_count	0
prev_days_termination_mean	0
prev_days_termination_max	0
prev_days_termination_min	0
prev_days_termination_count	0
prev_nflag_insured_on_approval_mean	0
prev_nflag_insured_on_approval_max	0
prev_nflag_insured_on_approval_min	0
prev_nflag_insured_on_approval_count	0
posf_sk_id_prev_mean	0
posf_sk_id_prev_max	0
posf_sk_id_prev_min	0
posf_sk_id_prev_count	0
posf_pos_sk_id_curr_mean_mean	0
posf_pos_sk_id_curr_mean_max	0
posf_pos_sk_id_curr_mean_min	0
posf_pos_sk_id_curr_mean_count	0
posf_pos_sk_id_curr_max_mean	0
posf_pos_sk_id_curr_max_max	0
posf_pos_sk_id_curr_max_min	0
posf_pos_sk_id_curr_max_count	0
posf_pos_sk_id_curr_min_mean	0
posf_pos_sk_id_curr_min_max	0
posf_pos_sk_id_curr_min_min	0
posf_pos_sk_id_curr_min_count	0
posf_pos_sk_id_curr_count_mean	0
posf_pos_sk_id_curr_count_max	0
posf_pos_sk_id_curr_count_min	0
posf_pos_sk_id_curr_count_count	0
posf_pos_months_balance_mean_mean	0
posf_pos_months_balance_mean_max	0
posf_pos_months_balance_mean_min	0
posf_pos_months_balance_mean_count	0
posf_pos_months_balance_max_mean	0
posf_pos_months_balance_max_max	0
posf_pos_months_balance_max_min	0
posf_pos_months_balance_max_count	0
posf_pos_months_balance_min_mean	0
posf_pos_months_balance_min_max	0
posf_pos_months_balance_min_min	0
posf_pos_months_balance_min_count	0
posf_pos_months_balance_count_mean	0
posf_pos_months_balance_count_max	0
posf_pos_months_balance_count_min	0
posf_pos_months_balance_count_count	0

posf_pos_cnt_instalment_mean_mean	0
posf_pos_cnt_instalment_mean_max	0
posf_pos_cnt_instalment_mean_min	0
posf_pos_cnt_instalment_mean_count	0
posf_pos_cnt_instalment_max_mean	0
posf_pos_cnt_instalment_max_max	0
posf_pos_cnt_instalment_max_min	0
posf_pos_cnt_instalment_max_count	0
posf_pos_cnt_instalment_min_mean	0
posf_pos_cnt_instalment_min_max	0
posf_pos_cnt_instalment_min_min	0
posf_pos_cnt_instalment_min_count	0
posf_pos_cnt_instalment_count_mean	0
posf_pos_cnt_instalment_count_max	0
posf_pos_cnt_instalment_count_min	0
posf_pos_cnt_instalment_count_count	0
posf_pos_cnt_instalment_future_mean_mean	0
posf_pos_cnt_instalment_future_mean_max	0
posf_pos_cnt_instalment_future_mean_min	0
posf_pos_cnt_instalment_future_mean_count	0
posf_pos_cnt_instalment_future_max_mean	0
posf_pos_cnt_instalment_future_max_max	0
posf_pos_cnt_instalment_future_max_min	0
posf_pos_cnt_instalment_future_max_count	0
posf_pos_cnt_instalment_future_min_mean	0
posf_pos_cnt_instalment_future_min_max	0
posf_pos_cnt_instalment_future_min_min	0
posf_pos_cnt_instalment_future_min_count	0
posf_pos_cnt_instalment_future_count_mean	0
posf_pos_cnt_instalment_future_count_max	0
posf_pos_cnt_instalment_future_count_min	0
posf_pos_cnt_instalment_future_count_count	0
posf_pos_sk_dpd_mean_mean	0
posf_pos_sk_dpd_mean_max	0
posf_pos_sk_dpd_mean_min	0
posf_pos_sk_dpd_mean_count	0
posf_pos_sk_dpd_max_mean	0
posf_pos_sk_dpd_max_max	0
posf_pos_sk_dpd_max_min	0
posf_pos_sk_dpd_max_count	0
posf_pos_sk_dpd_min_mean	0
posf_pos_sk_dpd_min_max	0
posf_pos_sk_dpd_min_min	0
posf_pos_sk_dpd_min_count	0
posf_pos_sk_dpd_count_mean	0
posf_pos_sk_dpd_count_max	0
posf_pos_sk_dpd_count_min	0
posf_pos_sk_dpd_count_count	0
posf_pos_sk_dpd_def_mean_mean	0
posf_pos_sk_dpd_def_mean_max	0
posf_pos_sk_dpd_def_mean_min	0
posf_pos_sk_dpd_def_mean_count	0
posf_pos_sk_dpd_def_max_mean	0
posf_pos_sk_dpd_def_max_max	0
posf_pos_sk_dpd_def_max_min	0
posf_pos_sk_dpd_def_max_count	0
posf_pos_sk_dpd_def_min_mean	0
posf_pos_sk_dpd_def_min_max	0
posf_pos_sk_dpd_def_min_min	0
posf_pos_sk_dpd_def_min_count	0
posf_pos_sk_dpd_def_count_mean	0
posf_pos_sk_dpd_def_count_max	0
posf_pos_sk_dpd_def_count_min	0
posf_pos_sk_dpd_def_count_count	0
ccf_sk_id_prev_mean	0
ccf_sk_id_prev_max	0
ccf_sk_id_prev_min	0
ccf_sk_id_prev_count	0
ccf_cc_sk_id_curr_mean_mean	0

ccf_cc_sk_id_curr_mean_max	0
ccf_cc_sk_id_curr_mean_min	0
ccf_cc_sk_id_curr_mean_count	0
ccf_cc_sk_id_curr_max_mean	0
ccf_cc_sk_id_curr_max_max	0
ccf_cc_sk_id_curr_max_min	0
ccf_cc_sk_id_curr_max_count	0
ccf_cc_sk_id_curr_min_mean	0
ccf_cc_sk_id_curr_min_max	0
ccf_cc_sk_id_curr_min_min	0
ccf_cc_sk_id_curr_min_count	0
ccf_cc_sk_id_curr_count_mean	0
ccf_cc_sk_id_curr_count_max	0
ccf_cc_sk_id_curr_count_min	0
ccf_cc_sk_id_curr_count_count	0
ccf_cc_months_balance_mean_mean	0
ccf_cc_months_balance_mean_max	0
ccf_cc_months_balance_mean_min	0
ccf_cc_months_balance_mean_count	0
ccf_cc_months_balance_max_mean	0
ccf_cc_months_balance_max_max	0
ccf_cc_months_balance_max_min	0
ccf_cc_months_balance_max_count	0
ccf_cc_months_balance_min_mean	0
ccf_cc_months_balance_min_max	0
ccf_cc_months_balance_min_min	0
ccf_cc_months_balance_min_count	0
ccf_cc_months_balance_count_mean	0
ccf_cc_months_balance_count_max	0
ccf_cc_months_balance_count_min	0
ccf_cc_months_balance_count_count	0
ccf_cc_amt_balance_mean_mean	0
ccf_cc_amt_balance_mean_max	0
ccf_cc_amt_balance_mean_min	0
ccf_cc_amt_balance_mean_count	0
ccf_cc_amt_balance_max_mean	0
ccf_cc_amt_balance_max_max	0
ccf_cc_amt_balance_max_min	0
ccf_cc_amt_balance_max_count	0
ccf_cc_amt_balance_min_mean	0
ccf_cc_amt_balance_min_max	0
ccf_cc_amt_balance_min_min	0
ccf_cc_amt_balance_min_count	0
ccf_cc_amt_balance_count_mean	0
ccf_cc_amt_balance_count_max	0
ccf_cc_amt_balance_count_min	0
ccf_cc_amt_balance_count_count	0
ccf_cc_amt_credit_limit_actual_mean_mean	0
ccf_cc_amt_credit_limit_actual_mean_max	0
ccf_cc_amt_credit_limit_actual_mean_min	0
ccf_cc_amt_credit_limit_actual_mean_count	0
ccf_cc_amt_credit_limit_actual_max_mean	0
ccf_cc_amt_credit_limit_actual_max_max	0
ccf_cc_amt_credit_limit_actual_max_min	0
ccf_cc_amt_credit_limit_actual_max_count	0
ccf_cc_amt_credit_limit_actual_min_mean	0
ccf_cc_amt_credit_limit_actual_min_max	0
ccf_cc_amt_credit_limit_actual_min_min	0
ccf_cc_amt_credit_limit_actual_min_count	0
ccf_cc_amt_credit_limit_actual_count_mean	0
ccf_cc_amt_credit_limit_actual_count_max	0
ccf_cc_amt_credit_limit_actual_count_min	0
ccf_cc_amt_credit_limit_actual_count_count	0
ccf_cc_amt_drawings_atm_current_mean_mean	0
ccf_cc_amt_drawings_atm_current_mean_max	0
ccf_cc_amt_drawings_atm_current_mean_min	0
ccf_cc_amt_drawings_atm_current_mean_count	0
ccf_cc_amt_drawings_atm_current_max_mean	0
ccf_cc_amt_drawings_atm_current_max_max	0

ccf_cc_amt_drawings_atm_current_max_min	0
ccf_cc_amt_drawings_atm_current_max_count	0
ccf_cc_amt_drawings_atm_current_min_mean	0
ccf_cc_amt_drawings_atm_current_min_max	0
ccf_cc_amt_drawings_atm_current_min_min	0
ccf_cc_amt_drawings_atm_current_min_count	0
ccf_cc_amt_drawings_atm_current_count_mean	0
ccf_cc_amt_drawings_atm_current_count_max	0
ccf_cc_amt_drawings_atm_current_count_min	0
ccf_cc_amt_drawings_atm_current_count_count	0
ccf_cc_amt_drawings_current_mean_mean	0
ccf_cc_amt_drawings_current_mean_max	0
ccf_cc_amt_drawings_current_mean_min	0
ccf_cc_amt_drawings_current_mean_count	0
ccf_cc_amt_drawings_current_max_mean	0
ccf_cc_amt_drawings_current_max_max	0
ccf_cc_amt_drawings_current_max_min	0
ccf_cc_amt_drawings_current_max_count	0
ccf_cc_amt_drawings_current_min_mean	0
ccf_cc_amt_drawings_current_min_max	0
ccf_cc_amt_drawings_current_min_min	0
ccf_cc_amt_drawings_current_min_count	0
ccf_cc_amt_drawings_current_count_mean	0
ccf_cc_amt_drawings_current_count_max	0
ccf_cc_amt_drawings_current_count_min	0
ccf_cc_amt_drawings_current_count_count	0
ccf_cc_amt_drawings_other_current_mean_mean	0
ccf_cc_amt_drawings_other_current_mean_max	0
ccf_cc_amt_drawings_other_current_mean_min	0
ccf_cc_amt_drawings_other_current_mean_count	0
ccf_cc_amt_drawings_other_current_max_mean	0
ccf_cc_amt_drawings_other_current_max_max	0
ccf_cc_amt_drawings_other_current_max_min	0
ccf_cc_amt_drawings_other_current_max_count	0
ccf_cc_amt_drawings_other_current_min_mean	0
ccf_cc_amt_drawings_other_current_min_max	0
ccf_cc_amt_drawings_other_current_min_min	0
ccf_cc_amt_drawings_other_current_min_count	0
ccf_cc_amt_drawings_other_current_count_mean	0
ccf_cc_amt_drawings_other_current_count_max	0
ccf_cc_amt_drawings_other_current_count_min	0
ccf_cc_amt_drawings_other_current_count_count	0
ccf_cc_amt_drawings_pos_current_mean_mean	0
ccf_cc_amt_drawings_pos_current_mean_max	0
ccf_cc_amt_drawings_pos_current_mean_min	0
ccf_cc_amt_drawings_pos_current_mean_count	0
ccf_cc_amt_drawings_pos_current_max_mean	0
ccf_cc_amt_drawings_pos_current_max_max	0
ccf_cc_amt_drawings_pos_current_max_min	0
ccf_cc_amt_drawings_pos_current_max_count	0
ccf_cc_amt_drawings_pos_current_min_mean	0
ccf_cc_amt_drawings_pos_current_min_max	0
ccf_cc_amt_drawings_pos_current_min_min	0
ccf_cc_amt_drawings_pos_current_min_count	0
ccf_cc_amt_drawings_pos_current_count_mean	0
ccf_cc_amt_drawings_pos_current_count_max	0
ccf_cc_amt_drawings_pos_current_count_min	0
ccf_cc_amt_drawings_pos_current_count_count	0
ccf_cc_amt_inst_min_regularity_mean_mean	0
ccf_cc_amt_inst_min_regularity_mean_max	0
ccf_cc_amt_inst_min_regularity_mean_min	0
ccf_cc_amt_inst_min_regularity_mean_count	0
ccf_cc_amt_inst_min_regularity_max_mean	0
ccf_cc_amt_inst_min_regularity_max_max	0
ccf_cc_amt_inst_min_regularity_max_min	0
ccf_cc_amt_inst_min_regularity_max_count	0
ccf_cc_amt_inst_min_regularity_min_mean	0
ccf_cc_amt_inst_min_regularity_min_max	0
ccf_cc_amt_inst_min_regularity_min_min	0

ccf_cc_amt_inst_min_regularity_min_count	0
ccf_cc_amt_inst_min_regularity_count_mean	0
ccf_cc_amt_inst_min_regularity_count_max	0
ccf_cc_amt_inst_min_regularity_count_min	0
ccf_cc_amt_inst_min_regularity_count_count	0
ccf_cc_amt_payment_current_mean_mean	0
ccf_cc_amt_payment_current_mean_max	0
ccf_cc_amt_payment_current_mean_min	0
ccf_cc_amt_payment_current_mean_count	0
ccf_cc_amt_payment_current_max_mean	0
ccf_cc_amt_payment_current_max_max	0
ccf_cc_amt_payment_current_max_min	0
ccf_cc_amt_payment_current_max_count	0
ccf_cc_amt_payment_current_min_mean	0
ccf_cc_amt_payment_current_min_max	0
ccf_cc_amt_payment_current_min_min	0
ccf_cc_amt_payment_current_min_count	0
ccf_cc_amt_payment_current_count_mean	0
ccf_cc_amt_payment_current_count_max	0
ccf_cc_amt_payment_current_count_min	0
ccf_cc_amt_payment_current_count_count	0
ccf_cc_amt_payment_total_current_mean_mean	0
ccf_cc_amt_payment_total_current_mean_max	0
ccf_cc_amt_payment_total_current_mean_min	0
ccf_cc_amt_payment_total_current_mean_count	0
ccf_cc_amt_payment_total_current_max_mean	0
ccf_cc_amt_payment_total_current_max_max	0
ccf_cc_amt_payment_total_current_max_min	0
ccf_cc_amt_payment_total_current_max_count	0
ccf_cc_amt_payment_total_current_min_mean	0
ccf_cc_amt_payment_total_current_min_max	0
ccf_cc_amt_payment_total_current_min_min	0
ccf_cc_amt_payment_total_current_min_count	0
ccf_cc_amt_payment_total_current_count_mean	0
ccf_cc_amt_payment_total_current_count_max	0
ccf_cc_amt_payment_total_current_count_min	0
ccf_cc_amt_payment_total_current_count_count	0
ccf_cc_amt_receivable_principal_mean_mean	0
ccf_cc_amt_receivable_principal_mean_max	0
ccf_cc_amt_receivable_principal_mean_min	0
ccf_cc_amt_receivable_principal_mean_count	0
ccf_cc_amt_receivable_principal_max_mean	0
ccf_cc_amt_receivable_principal_max_max	0
ccf_cc_amt_receivable_principal_max_min	0
ccf_cc_amt_receivable_principal_max_count	0
ccf_cc_amt_receivable_principal_min_mean	0
ccf_cc_amt_receivable_principal_min_max	0
ccf_cc_amt_receivable_principal_min_min	0
ccf_cc_amt_receivable_principal_min_count	0
ccf_cc_amt_receivable_principal_count_mean	0
ccf_cc_amt_receivable_principal_count_max	0
ccf_cc_amt_receivable_principal_count_min	0
ccf_cc_amt_receivable_principal_count_count	0
ccf_cc_amt_recivable_mean_mean	0
ccf_cc_amt_recivable_mean_max	0
ccf_cc_amt_recivable_mean_min	0
ccf_cc_amt_recivable_mean_count	0
ccf_cc_amt_recivable_max_mean	0
ccf_cc_amt_recivable_max_max	0
ccf_cc_amt_recivable_max_min	0
ccf_cc_amt_recivable_max_count	0
ccf_cc_amt_recivable_min_mean	0
ccf_cc_amt_recivable_min_max	0
ccf_cc_amt_recivable_min_min	0
ccf_cc_amt_recivable_min_count	0
ccf_cc_amt_recivable_count_mean	0
ccf_cc_amt_recivable_count_max	0
ccf_cc_amt_recivable_count_min	0
ccf_cc_amt_recivable_count_count	0

ccf_cc_amt_total_receivable_mean_mean	0
ccf_cc_amt_total_receivable_mean_max	0
ccf_cc_amt_total_receivable_mean_min	0
ccf_cc_amt_total_receivable_mean_count	0
ccf_cc_amt_total_receivable_max_mean	0
ccf_cc_amt_total_receivable_max_max	0
ccf_cc_amt_total_receivable_max_min	0
ccf_cc_amt_total_receivable_max_count	0
ccf_cc_amt_total_receivable_min_mean	0
ccf_cc_amt_total_receivable_min_max	0
ccf_cc_amt_total_receivable_min_min	0
ccf_cc_amt_total_receivable_min_count	0
ccf_cc_amt_total_receivable_count_mean	0
ccf_cc_amt_total_receivable_count_max	0
ccf_cc_amt_total_receivable_count_min	0
ccf_cc_amt_total_receivable_count_count	0
ccf_cc_cnt_drawings_atm_current_mean_mean	0
ccf_cc_cnt_drawings_atm_current_mean_max	0
ccf_cc_cnt_drawings_atm_current_mean_min	0
ccf_cc_cnt_drawings_atm_current_mean_count	0
ccf_cc_cnt_drawings_atm_current_max_mean	0
ccf_cc_cnt_drawings_atm_current_max_max	0
ccf_cc_cnt_drawings_atm_current_max_min	0
ccf_cc_cnt_drawings_atm_current_max_count	0
ccf_cc_cnt_drawings_atm_current_min_mean	0
ccf_cc_cnt_drawings_atm_current_min_max	0
ccf_cc_cnt_drawings_atm_current_min_min	0
ccf_cc_cnt_drawings_atm_current_min_count	0
ccf_cc_cnt_drawings_atm_current_count_mean	0
ccf_cc_cnt_drawings_atm_current_count_max	0
ccf_cc_cnt_drawings_atm_current_count_min	0
ccf_cc_cnt_drawings_atm_current_count_count	0
ccf_cc_cnt_drawings_current_mean_mean	0
ccf_cc_cnt_drawings_current_mean_max	0
ccf_cc_cnt_drawings_current_mean_min	0
ccf_cc_cnt_drawings_current_mean_count	0
ccf_cc_cnt_drawings_current_max_mean	0
ccf_cc_cnt_drawings_current_max_max	0
ccf_cc_cnt_drawings_current_max_min	0
ccf_cc_cnt_drawings_current_max_count	0
ccf_cc_cnt_drawings_current_min_mean	0
ccf_cc_cnt_drawings_current_min_max	0
ccf_cc_cnt_drawings_current_min_min	0
ccf_cc_cnt_drawings_current_min_count	0
ccf_cc_cnt_drawings_current_count_mean	0
ccf_cc_cnt_drawings_current_count_max	0
ccf_cc_cnt_drawings_current_count_min	0
ccf_cc_cnt_drawings_current_count_count	0
ccf_cc_cnt_drawings_other_current_mean_mean	0
ccf_cc_cnt_drawings_other_current_mean_max	0
ccf_cc_cnt_drawings_other_current_mean_min	0
ccf_cc_cnt_drawings_other_current_mean_count	0
ccf_cc_cnt_drawings_other_current_max_mean	0
ccf_cc_cnt_drawings_other_current_max_max	0
ccf_cc_cnt_drawings_other_current_max_min	0
ccf_cc_cnt_drawings_other_current_max_count	0
ccf_cc_cnt_drawings_other_current_min_mean	0
ccf_cc_cnt_drawings_other_current_min_max	0
ccf_cc_cnt_drawings_other_current_min_min	0
ccf_cc_cnt_drawings_other_current_min_count	0
ccf_cc_cnt_drawings_other_current_count_mean	0
ccf_cc_cnt_drawings_other_current_count_max	0
ccf_cc_cnt_drawings_other_current_count_min	0
ccf_cc_cnt_drawings_other_current_count_count	0
ccf_cc_cnt_drawings_pos_current_mean_mean	0
ccf_cc_cnt_drawings_pos_current_mean_max	0
ccf_cc_cnt_drawings_pos_current_mean_min	0
ccf_cc_cnt_drawings_pos_current_mean_count	0
ccf_cc_cnt_drawings_pos_current_max_mean	0

ccf_cc_cnt_drawings_pos_current_max_max	0
ccf_cc_cnt_drawings_pos_current_max_min	0
ccf_cc_cnt_drawings_pos_current_max_count	0
ccf_cc_cnt_drawings_pos_current_min_mean	0
ccf_cc_cnt_drawings_pos_current_min_max	0
ccf_cc_cnt_drawings_pos_current_min_min	0
ccf_cc_cnt_drawings_pos_current_min_count	0
ccf_cc_cnt_drawings_pos_current_count_mean	0
ccf_cc_cnt_drawings_pos_current_count_max	0
ccf_cc_cnt_drawings_pos_current_count_min	0
ccf_cc_cnt_drawings_pos_current_count_count	0
ccf_cc_cnt_instalment_mature_cum_mean_mean	0
ccf_cc_cnt_instalment_mature_cum_mean_max	0
ccf_cc_cnt_instalment_mature_cum_mean_min	0
ccf_cc_cnt_instalment_mature_cum_mean_count	0
ccf_cc_cnt_instalment_mature_cum_max_mean	0
ccf_cc_cnt_instalment_mature_cum_max_max	0
ccf_cc_cnt_instalment_mature_cum_max_min	0
ccf_cc_cnt_instalment_mature_cum_max_count	0
ccf_cc_cnt_instalment_mature_cum_min_mean	0
ccf_cc_cnt_instalment_mature_cum_min_max	0
ccf_cc_cnt_instalment_mature_cum_min_min	0
ccf_cc_cnt_instalment_mature_cum_min_count	0
ccf_cc_cnt_instalment_mature_cum_count_mean	0
ccf_cc_cnt_instalment_mature_cum_count_max	0
ccf_cc_cnt_instalment_mature_cum_count_min	0
ccf_cc_cnt_instalment_mature_cum_count_count	0
ccf_cc_sk_dpd_mean_mean	0
ccf_cc_sk_dpd_mean_max	0
ccf_cc_sk_dpd_mean_min	0
ccf_cc_sk_dpd_mean_count	0
ccf_cc_sk_dpd_max_mean	0
ccf_cc_sk_dpd_max_max	0
ccf_cc_sk_dpd_max_min	0
ccf_cc_sk_dpd_max_count	0
ccf_cc_sk_dpd_min_mean	0
ccf_cc_sk_dpd_min_max	0
ccf_cc_sk_dpd_min_min	0
ccf_cc_sk_dpd_min_count	0
ccf_cc_sk_dpd_count_mean	0
ccf_cc_sk_dpd_count_max	0
ccf_cc_sk_dpd_count_min	0
ccf_cc_sk_dpd_count_count	0
ccf_cc_sk_dpd_def_mean_mean	0
ccf_cc_sk_dpd_def_mean_max	0
ccf_cc_sk_dpd_def_mean_min	0
ccf_cc_sk_dpd_def_mean_count	0
ccf_cc_sk_dpd_def_max_mean	0
ccf_cc_sk_dpd_def_max_max	0
ccf_cc_sk_dpd_def_max_min	0
ccf_cc_sk_dpd_def_max_count	0
ccf_cc_sk_dpd_def_min_mean	0
ccf_cc_sk_dpd_def_min_max	0
ccf_cc_sk_dpd_def_min_min	0
ccf_cc_sk_dpd_def_min_count	0
ccf_cc_sk_dpd_def_count_mean	0
ccf_cc_sk_dpd_def_count_max	0
ccf_cc_sk_dpd_def_count_min	0
ccf_cc_sk_dpd_def_count_count	0
instf_sk_id_prev_mean	0
instf_sk_id_prev_max	0
instf_sk_id_prev_min	0
instf_sk_id_prev_count	0
instf_inst_sk_id_curr_mean_mean	0
instf_inst_sk_id_curr_mean_max	0
instf_inst_sk_id_curr_mean_min	0
instf_inst_sk_id_curr_mean_count	0
instf_inst_sk_id_curr_max_mean	0
instf_inst_sk_id_curr_max_max	0

instf_inst_sk_id_curr_max_min	0
instf_inst_sk_id_curr_max_count	0
instf_inst_sk_id_curr_min_mean	0
instf_inst_sk_id_curr_min_max	0
instf_inst_sk_id_curr_min_min	0
instf_inst_sk_id_curr_min_count	0
instf_inst_sk_id_curr_count_mean	0
instf_inst_sk_id_curr_count_max	0
instf_inst_sk_id_curr_count_min	0
instf_inst_sk_id_curr_count_count	0
instf_inst_num_instalment_version_mean_mean	0
instf_inst_num_instalment_version_mean_max	0
instf_inst_num_instalment_version_mean_min	0
instf_inst_num_instalment_version_mean_count	0
instf_inst_num_instalment_version_max_mean	0
instf_inst_num_instalment_version_max_max	0
instf_inst_num_instalment_version_max_min	0
instf_inst_num_instalment_version_max_count	0
instf_inst_num_instalment_version_min_mean	0
instf_inst_num_instalment_version_min_max	0
instf_inst_num_instalment_version_min_min	0
instf_inst_num_instalment_version_min_count	0
instf_inst_num_instalment_version_count_mean	0
instf_inst_num_instalment_version_count_max	0
instf_inst_num_instalment_version_count_min	0
instf_inst_num_instalment_version_count_count	0
instf_inst_num_instalment_number_mean_mean	0
instf_inst_num_instalment_number_mean_max	0
instf_inst_num_instalment_number_mean_min	0
instf_inst_num_instalment_number_mean_count	0
instf_inst_num_instalment_number_max_mean	0
instf_inst_num_instalment_number_max_max	0
instf_inst_num_instalment_number_max_min	0
instf_inst_num_instalment_number_max_count	0
instf_inst_num_instalment_number_min_mean	0
instf_inst_num_instalment_number_min_max	0
instf_inst_num_instalment_number_min_min	0
instf_inst_num_instalment_number_min_count	0
instf_inst_num_instalment_number_count_mean	0
instf_inst_num_instalment_number_count_max	0
instf_inst_num_instalment_number_count_min	0
instf_inst_num_instalment_number_count_count	0
instf_inst_days_instalment_mean_mean	0
instf_inst_days_instalment_mean_max	0
instf_inst_days_instalment_mean_min	0
instf_inst_days_instalment_mean_count	0
instf_inst_days_instalment_max_mean	0
instf_inst_days_instalment_max_max	0
instf_inst_days_instalment_max_min	0
instf_inst_days_instalment_max_count	0
instf_inst_days_instalment_min_mean	0
instf_inst_days_instalment_min_max	0
instf_inst_days_instalment_min_min	0
instf_inst_days_instalment_min_count	0
instf_inst_days_instalment_count_mean	0
instf_inst_days_instalment_count_max	0
instf_inst_days_instalment_count_min	0
instf_inst_days_instalment_count_count	0
instf_inst_days_entry_payment_mean_mean	0
instf_inst_days_entry_payment_mean_max	0
instf_inst_days_entry_payment_mean_min	0
instf_inst_days_entry_payment_mean_count	0
instf_inst_days_entry_payment_max_mean	0
instf_inst_days_entry_payment_max_max	0
instf_inst_days_entry_payment_max_min	0
instf_inst_days_entry_payment_max_count	0
instf_inst_days_entry_payment_min_mean	0
instf_inst_days_entry_payment_min_max	0
instf_inst_days_entry_payment_min_min	0

instf_inst_days_entry_payment_min_count	0
instf_inst_days_entry_payment_count_mean	0
instf_inst_days_entry_payment_count_max	0
instf_inst_days_entry_payment_count_min	0
instf_inst_days_entry_payment_count_count	0
instf_inst_amt_instalment_mean_mean	0
instf_inst_amt_instalment_mean_max	0
instf_inst_amt_instalment_mean_min	0
instf_inst_amt_instalment_mean_count	0
instf_inst_amt_instalment_max_mean	0
instf_inst_amt_instalment_max_max	0
instf_inst_amt_instalment_max_min	0
instf_inst_amt_instalment_max_count	0
instf_inst_amt_instalment_min_mean	0
instf_inst_amt_instalment_min_max	0
instf_inst_amt_instalment_min_min	0
instf_inst_amt_instalment_min_count	0
instf_inst_amt_instalment_count_mean	0
instf_inst_amt_instalment_count_max	0
instf_inst_amt_instalment_count_min	0
instf_inst_amt_instalment_count_count	0
instf_inst_amt_payment_mean_mean	0
instf_inst_amt_payment_mean_max	0
instf_inst_amt_payment_mean_min	0
instf_inst_amt_payment_mean_count	0
instf_inst_amt_payment_max_mean	0
instf_inst_amt_payment_max_max	0
instf_inst_amt_payment_max_min	0
instf_inst_amt_payment_max_count	0
instf_inst_amt_payment_min_mean	0
instf_inst_amt_payment_min_max	0
instf_inst_amt_payment_min_min	0
instf_inst_amt_payment_min_count	0
instf_inst_amt_payment_count_mean	0
instf_inst_amt_payment_count_max	0
instf_inst_amt_payment_count_min	0
instf_inst_amt_payment_count_count	0
is_building	0
is_building_missing	0
has_basement	0
basementarea_missing	0
has_noliving	0
nonlivingarea_missing	0
is_employed	0
livingarea_missing	0
apartments_missing	0
floorsmax_missing	0
years_build_missing	0
commonarea_missing	0
emergencystate_missing	0
elevators_avg_missing	0
elevators_mode_missing	0
elevators_medi_missing	0
housetype_mode_missing	0
wallsmaterial_mode_missing	0
occupation_type_missing	0
organization_type_missing	0
name_type_suite_missing	0
code_gender_missing	0
amt_annuity_missing	0
amt_goods_price_missing	0
own_car_age_missing	0
cnt_fam_members_missing	0
ext_source_1_missing	0
ext_source_2_missing	0
ext_source_3_missing	0
apartments_avg_missing	0
basementarea_avg_missing	0
years_beginexpluatation_avg_missing	0

years_build_avg_missing	0
commonarea_avg_missing	0
floorsmax_avg_missing	0
landarea_avg_missing	0
livingapartments_avg_missing	0
livingarea_avg_missing	0
nonlivingapartments_avg_missing	0
nonlivingarea_avg_missing	0
totalarea_mode_missing	0
obs_30_cnt_social_circle_missing	0
def_30_cnt_social_circle_missing	0
obs_60_cnt_social_circle_missing	0
def_60_cnt_social_circle_missing	0
days_last_phone_change_missing	0
amt_req_credit_bureau_hour_missing	0
amt_req_credit_bureau_day_missing	0
amt_req_credit_bureau_week_missing	0
amt_req_credit_bureau_mon_missing	0
amt_req_credit_bureau_qrt_missing	0
amt_req_credit_bureau_year_missing	0
bur_sk_id_bureau_mean_missing	0
bur_sk_id_bureau_max_missing	0
bur_sk_id_bureau_min_missing	0
bur_sk_id_bureau_count_missing	0
bur_days_credit_mean_missing	0
bur_days_credit_max_missing	0
bur_days_credit_min_missing	0
bur_days_credit_count_missing	0
bur_credit_day_overdue_mean_missing	0
bur_credit_day_overdue_max_missing	0
bur_credit_day_overdue_min_missing	0
bur_credit_day_overdue_count_missing	0
bur_days_credit_enddate_mean_missing	0
bur_days_credit_enddate_max_missing	0
bur_days_credit_enddate_min_missing	0
bur_days_credit_enddate_count_missing	0
bur_days_enddate_fact_mean_missing	0
bur_days_enddate_fact_max_missing	0
bur_days_enddate_fact_min_missing	0
bur_days_enddate_fact_count_missing	0
bur_amt_credit_max_overdue_mean_missing	0
bur_amt_credit_max_overdue_max_missing	0
bur_amt_credit_max_overdue_min_missing	0
bur_amt_credit_max_overdue_count_missing	0
bur_cnt_credit_prolong_mean_missing	0
bur_cnt_credit_prolong_max_missing	0
bur_cnt_credit_prolong_min_missing	0
bur_cnt_credit_prolong_count_missing	0
bur_amt_credit_sum_mean_missing	0
bur_amt_credit_sum_max_missing	0
bur_amt_credit_sum_min_missing	0
bur_amt_credit_sum_count_missing	0
bur_amt_credit_sum_debt_mean_missing	0
bur_amt_credit_sum_debt_max_missing	0
bur_amt_credit_sum_debt_min_missing	0
bur_amt_credit_sum_debt_count_missing	0
bur_amt_credit_sum_limit_mean_missing	0
bur_amt_credit_sum_limit_max_missing	0
bur_amt_credit_sum_limit_min_missing	0
bur_amt_credit_sum_limit_count_missing	0
bur_amt_credit_sum_overdue_mean_missing	0
bur_amt_credit_sum_overdue_max_missing	0
bur_amt_credit_sum_overdue_min_missing	0
bur_amt_credit_sum_overdue_count_missing	0
bur_days_credit_update_mean_missing	0
bur_days_credit_update_max_missing	0
bur_days_credit_update_min_missing	0
bur_days_credit_update_count_missing	0
bur_amt_annuity_mean_missing	0

bur_amt_annuity_max_missing	0
bur_amt_annuity_min_missing	0
bur_amt_annuity_count_missing	0
bur_bb_months_balance_mean_mean_missing	0
bur_bb_months_balance_mean_max_missing	0
bur_bb_months_balance_mean_min_missing	0
bur_bb_months_balance_mean_count_missing	0
bur_bb_months_balance_max_mean_missing	0
bur_bb_months_balance_max_max_missing	0
bur_bb_months_balance_max_min_missing	0
bur_bb_months_balance_max_count_missing	0
bur_bb_months_balance_min_mean_missing	0
bur_bb_months_balance_min_max_missing	0
bur_bb_months_balance_min_min_missing	0
bur_bb_months_balance_min_count_missing	0
bur_bb_months_balance_count_mean_missing	0
bur_bb_months_balance_count_max_missing	0
bur_bb_months_balance_count_min_missing	0
bur_bb_months_balance_count_count_missing	0
prev_sk_id_prev_mean_missing	0
prev_sk_id_prev_max_missing	0
prev_sk_id_prev_min_missing	0
prev_sk_id_prev_count_missing	0
prev_amt_annuity_mean_missing	0
prev_amt_annuity_max_missing	0
prev_amt_annuity_min_missing	0
prev_amt_annuity_count_missing	0
prev_amt_application_mean_missing	0
prev_amt_application_max_missing	0
prev_amt_application_min_missing	0
prev_amt_application_count_missing	0
prev_amt_credit_mean_missing	0
prev_amt_credit_max_missing	0
prev_amt_credit_min_missing	0
prev_amt_credit_count_missing	0
prev_amt_down_payment_mean_missing	0
prev_amt_down_payment_max_missing	0
prev_amt_down_payment_min_missing	0
prev_amt_down_payment_count_missing	0
prev_amt_goods_price_mean_missing	0
prev_amt_goods_price_max_missing	0
prev_amt_goods_price_min_missing	0
prev_amt_goods_price_count_missing	0
prev_hour_appr_process_start_mean_missing	0
prev_hour_appr_process_start_max_missing	0
prev_hour_appr_process_start_min_missing	0
prev_hour_appr_process_start_count_missing	0
prev_nflag_last_appl_in_day_mean_missing	0
prev_nflag_last_appl_in_day_max_missing	0
prev_nflag_last_appl_in_day_min_missing	0
prev_nflag_last_appl_in_day_count_missing	0
prev_rate_down_payment_mean_missing	0
prev_rate_down_payment_max_missing	0
prev_rate_down_payment_min_missing	0
prev_rate_down_payment_count_missing	0
prev_rate_interest_primary_mean_missing	0
prev_rate_interest_primary_max_missing	0
prev_rate_interest_primary_min_missing	0
prev_rate_interest_primary_count_missing	0
prev_rate_interest_privileged_mean_missing	0
prev_rate_interest_privileged_max_missing	0
prev_rate_interest_privileged_min_missing	0
prev_rate_interest_privileged_count_missing	0
prev_days_decision_mean_missing	0
prev_days_decision_max_missing	0
prev_days_decision_min_missing	0
prev_days_decision_count_missing	0
prev_sellerplace_area_mean_missing	0
prev_sellerplace_area_max_missing	0

prev_sellerplace_area_min_missing	0
prev_sellerplace_area_count_missing	0
prev_cnt_payment_mean_missing	0
prev_cnt_payment_max_missing	0
prev_cnt_payment_min_missing	0
prev_cnt_payment_count_missing	0
prev_days_first_drawing_mean_missing	0
prev_days_first_drawing_max_missing	0
prev_days_first_drawing_min_missing	0
prev_days_first_drawing_count_missing	0
prev_days_first_due_mean_missing	0
prev_days_first_due_max_missing	0
prev_days_first_due_min_missing	0
prev_days_first_due_count_missing	0
prev_days_last_due_1st_version_mean_missing	0
prev_days_last_due_1st_version_max_missing	0
prev_days_last_due_1st_version_min_missing	0
prev_days_last_due_1st_version_count_missing	0
prev_days_last_due_mean_missing	0
prev_days_last_due_max_missing	0
prev_days_last_due_min_missing	0
prev_days_last_due_count_missing	0
prev_days_termination_mean_missing	0
prev_days_termination_max_missing	0
prev_days_termination_min_missing	0
prev_days_termination_count_missing	0
prev_nflag_insured_on_approval_mean_missing	0
prev_nflag_insured_on_approval_max_missing	0
prev_nflag_insured_on_approval_min_missing	0
prev_nflag_insured_on_approval_count_missing	0
posf_sk_id_prev_mean_missing	0
posf_sk_id_prev_max_missing	0
posf_sk_id_prev_min_missing	0
posf_sk_id_prev_count_missing	0
posf_pos_sk_id_curr_mean_mean_missing	0
posf_pos_sk_id_curr_mean_max_missing	0
posf_pos_sk_id_curr_mean_min_missing	0
posf_pos_sk_id_curr_mean_count_missing	0
posf_pos_sk_id_curr_max_mean_missing	0
posf_pos_sk_id_curr_max_max_missing	0
posf_pos_sk_id_curr_max_min_missing	0
posf_pos_sk_id_curr_max_count_missing	0
posf_pos_sk_id_curr_min_mean_missing	0
posf_pos_sk_id_curr_min_max_missing	0
posf_pos_sk_id_curr_min_min_missing	0
posf_pos_sk_id_curr_min_count_missing	0
posf_pos_sk_id_curr_count_mean_missing	0
posf_pos_sk_id_curr_count_max_missing	0
posf_pos_sk_id_curr_count_min_missing	0
posf_pos_sk_id_curr_count_count_missing	0
posf_pos_months_balance_mean_mean_missing	0
posf_pos_months_balance_mean_max_missing	0
posf_pos_months_balance_mean_min_missing	0
posf_pos_months_balance_mean_count_missing	0
posf_pos_months_balance_max_mean_missing	0
posf_pos_months_balance_max_max_missing	0
posf_pos_months_balance_max_min_missing	0
posf_pos_months_balance_max_count_missing	0
posf_pos_months_balance_min_mean_missing	0
posf_pos_months_balance_min_max_missing	0
posf_pos_months_balance_min_min_missing	0
posf_pos_months_balance_min_count_missing	0
posf_pos_months_balance_count_mean_missing	0
posf_pos_months_balance_count_max_missing	0
posf_pos_months_balance_count_min_missing	0
posf_pos_months_balance_count_count_missing	0
posf_pos_cnt_instalment_mean_mean_missing	0
posf_pos_cnt_instalment_mean_max_missing	0
posf_pos_cnt_instalment_mean_min_missing	0

posf_pos_cnt_instalment_mean_count_missing	0
posf_pos_cnt_instalment_max_mean_missing	0
posf_pos_cnt_instalment_max_max_missing	0
posf_pos_cnt_instalment_max_min_missing	0
posf_pos_cnt_instalment_max_count_missing	0
posf_pos_cnt_instalment_min_mean_missing	0
posf_pos_cnt_instalment_min_max_missing	0
posf_pos_cnt_instalment_min_min_missing	0
posf_pos_cnt_instalment_min_count_missing	0
posf_pos_cnt_instalment_count_mean_missing	0
posf_pos_cnt_instalment_count_max_missing	0
posf_pos_cnt_instalment_count_min_missing	0
posf_pos_cnt_instalment_count_count_missing	0
posf_pos_cnt_instalment_future_mean_mean_missing	0
posf_pos_cnt_instalment_future_mean_max_missing	0
posf_pos_cnt_instalment_future_mean_min_missing	0
posf_pos_cnt_instalment_future_mean_count_missing	0
posf_pos_cnt_instalment_future_max_mean_missing	0
posf_pos_cnt_instalment_future_max_max_missing	0
posf_pos_cnt_instalment_future_max_min_missing	0
posf_pos_cnt_instalment_future_max_count_missing	0
posf_pos_cnt_instalment_future_min_mean_missing	0
posf_pos_cnt_instalment_future_min_max_missing	0
posf_pos_cnt_instalment_future_min_min_missing	0
posf_pos_cnt_instalment_future_min_count_missing	0
posf_pos_cnt_instalment_future_count_mean_missing	0
posf_pos_cnt_instalment_future_count_max_missing	0
posf_pos_cnt_instalment_future_count_min_missing	0
posf_pos_cnt_instalment_future_count_count_missing	0
posf_pos_sk_dpd_mean_mean_missing	0
posf_pos_sk_dpd_mean_max_missing	0
posf_pos_sk_dpd_mean_min_missing	0
posf_pos_sk_dpd_mean_count_missing	0
posf_pos_sk_dpd_max_mean_missing	0
posf_pos_sk_dpd_max_max_missing	0
posf_pos_sk_dpd_max_min_missing	0
posf_pos_sk_dpd_max_count_missing	0
posf_pos_sk_dpd_min_mean_missing	0
posf_pos_sk_dpd_min_max_missing	0
posf_pos_sk_dpd_min_min_missing	0
posf_pos_sk_dpd_min_count_missing	0
posf_pos_sk_dpd_count_mean_missing	0
posf_pos_sk_dpd_count_max_missing	0
posf_pos_sk_dpd_count_min_missing	0
posf_pos_sk_dpd_count_count_missing	0
posf_pos_sk_dpd_def_mean_mean_missing	0
posf_pos_sk_dpd_def_mean_max_missing	0
posf_pos_sk_dpd_def_mean_min_missing	0
posf_pos_sk_dpd_def_mean_count_missing	0
posf_pos_sk_dpd_def_max_mean_missing	0
posf_pos_sk_dpd_def_max_max_missing	0
posf_pos_sk_dpd_def_max_min_missing	0
posf_pos_sk_dpd_def_max_count_missing	0
posf_pos_sk_dpd_def_min_mean_missing	0
posf_pos_sk_dpd_def_min_max_missing	0
posf_pos_sk_dpd_def_min_min_missing	0
posf_pos_sk_dpd_def_min_count_missing	0
posf_pos_sk_dpd_def_count_mean_missing	0
posf_pos_sk_dpd_def_count_max_missing	0
posf_pos_sk_dpd_def_count_min_missing	0
posf_pos_sk_dpd_def_count_count_missing	0
ccf_sk_id_prev_mean_missing	0
ccf_sk_id_prev_max_missing	0
ccf_sk_id_prev_min_missing	0
ccf_sk_id_prev_count_missing	0
ccf_cc_sk_id_curr_mean_mean_missing	0
ccf_cc_sk_id_curr_mean_max_missing	0
ccf_cc_sk_id_curr_mean_min_missing	0
ccf_cc_sk_id_curr_mean_count_missing	0

ccf_cc_sk_id_curr_max_mean_missing	0
ccf_cc_sk_id_curr_max_max_missing	0
ccf_cc_sk_id_curr_max_min_missing	0
ccf_cc_sk_id_curr_max_count_missing	0
ccf_cc_sk_id_curr_min_mean_missing	0
ccf_cc_sk_id_curr_min_max_missing	0
ccf_cc_sk_id_curr_min_min_missing	0
ccf_cc_sk_id_curr_min_count_missing	0
ccf_cc_sk_id_curr_count_mean_missing	0
ccf_cc_sk_id_curr_count_max_missing	0
ccf_cc_sk_id_curr_count_min_missing	0
ccf_cc_sk_id_curr_count_count_missing	0
ccf_cc_months_balance_mean_mean_missing	0
ccf_cc_months_balance_mean_max_missing	0
ccf_cc_months_balance_mean_min_missing	0
ccf_cc_months_balance_mean_count_missing	0
ccf_cc_months_balance_max_mean_missing	0
ccf_cc_months_balance_max_max_missing	0
ccf_cc_months_balance_max_min_missing	0
ccf_cc_months_balance_max_count_missing	0
ccf_cc_months_balance_min_mean_missing	0
ccf_cc_months_balance_min_max_missing	0
ccf_cc_months_balance_min_min_missing	0
ccf_cc_months_balance_min_count_missing	0
ccf_cc_months_balance_count_mean_missing	0
ccf_cc_months_balance_count_max_missing	0
ccf_cc_months_balance_count_min_missing	0
ccf_cc_months_balance_count_count_missing	0
ccf_cc_amt_balance_mean_mean_missing	0
ccf_cc_amt_balance_mean_max_missing	0
ccf_cc_amt_balance_mean_min_missing	0
ccf_cc_amt_balance_mean_count_missing	0
ccf_cc_amt_balance_max_mean_missing	0
ccf_cc_amt_balance_max_max_missing	0
ccf_cc_amt_balance_max_min_missing	0
ccf_cc_amt_balance_max_count_missing	0
ccf_cc_amt_balance_min_mean_missing	0
ccf_cc_amt_balance_min_max_missing	0
ccf_cc_amt_balance_min_min_missing	0
ccf_cc_amt_balance_min_count_missing	0
ccf_cc_amt_balance_count_mean_missing	0
ccf_cc_amt_balance_count_max_missing	0
ccf_cc_amt_balance_count_min_missing	0
ccf_cc_amt_balance_count_count_missing	0
ccf_cc_amt_credit_limit_actual_mean_mean_missing	0
ccf_cc_amt_credit_limit_actual_mean_max_missing	0
ccf_cc_amt_credit_limit_actual_mean_min_missing	0
ccf_cc_amt_credit_limit_actual_mean_count_missing	0
ccf_cc_amt_credit_limit_actual_max_mean_missing	0
ccf_cc_amt_credit_limit_actual_max_max_missing	0
ccf_cc_amt_credit_limit_actual_max_min_missing	0
ccf_cc_amt_credit_limit_actual_max_count_missing	0
ccf_cc_amt_credit_limit_actual_min_mean_missing	0
ccf_cc_amt_credit_limit_actual_min_max_missing	0
ccf_cc_amt_credit_limit_actual_min_min_missing	0
ccf_cc_amt_credit_limit_actual_min_count_missing	0
ccf_cc_amt_credit_limit_actual_count_mean_missing	0
ccf_cc_amt_credit_limit_actual_count_max_missing	0
ccf_cc_amt_credit_limit_actual_count_min_missing	0
ccf_cc_amt_credit_limit_actual_count_count_missing	0
ccf_cc_amt_drawings_atm_current_mean_mean_missing	0
ccf_cc_amt_drawings_atm_current_mean_max_missing	0
ccf_cc_amt_drawings_atm_current_mean_min_missing	0
ccf_cc_amt_drawings_atm_current_mean_count_missing	0
ccf_cc_amt_drawings_atm_current_max_mean_missing	0
ccf_cc_amt_drawings_atm_current_max_max_missing	0
ccf_cc_amt_drawings_atm_current_max_min_missing	0
ccf_cc_amt_drawings_atm_current_max_count_missing	0
ccf_cc_amt_drawings_atm_current_min_mean_missing	0

ccf_cc_amt_drawings_atm_current_min_max_missing	0
ccf_cc_amt_drawings_atm_current_min_min_missing	0
ccf_cc_amt_drawings_atm_current_min_count_missing	0
ccf_cc_amt_drawings_atm_current_count_mean_missing	0
ccf_cc_amt_drawings_atm_current_count_max_missing	0
ccf_cc_amt_drawings_atm_current_count_min_missing	0
ccf_cc_amt_drawings_atm_current_count_count_missing	0
ccf_cc_amt_drawings_current_mean_mean_missing	0
ccf_cc_amt_drawings_current_mean_max_missing	0
ccf_cc_amt_drawings_current_mean_min_missing	0
ccf_cc_amt_drawings_current_mean_count_missing	0
ccf_cc_amt_drawings_current_max_mean_missing	0
ccf_cc_amt_drawings_current_max_max_missing	0
ccf_cc_amt_drawings_current_max_min_missing	0
ccf_cc_amt_drawings_current_max_count_missing	0
ccf_cc_amt_drawings_current_min_mean_missing	0
ccf_cc_amt_drawings_current_min_max_missing	0
ccf_cc_amt_drawings_current_min_min_missing	0
ccf_cc_amt_drawings_current_min_count_missing	0
ccf_cc_amt_drawings_current_count_mean_missing	0
ccf_cc_amt_drawings_current_count_max_missing	0
ccf_cc_amt_drawings_current_count_min_missing	0
ccf_cc_amt_drawings_current_count_count_missing	0
ccf_cc_amt_drawings_other_current_mean_mean_missing	0
ccf_cc_amt_drawings_other_current_mean_max_missing	0
ccf_cc_amt_drawings_other_current_mean_min_missing	0
ccf_cc_amt_drawings_other_current_mean_count_missing	0
ccf_cc_amt_drawings_other_current_max_mean_missing	0
ccf_cc_amt_drawings_other_current_max_max_missing	0
ccf_cc_amt_drawings_other_current_max_min_missing	0
ccf_cc_amt_drawings_other_current_max_count_missing	0
ccf_cc_amt_drawings_other_current_min_mean_missing	0
ccf_cc_amt_drawings_other_current_min_max_missing	0
ccf_cc_amt_drawings_other_current_min_min_missing	0
ccf_cc_amt_drawings_other_current_min_count_missing	0
ccf_cc_amt_drawings_other_current_count_mean_missing	0
ccf_cc_amt_drawings_other_current_count_max_missing	0
ccf_cc_amt_drawings_other_current_count_min_missing	0
ccf_cc_amt_drawings_other_current_count_count_missing	0
ccf_cc_amt_drawings_pos_current_mean_mean_missing	0
ccf_cc_amt_drawings_pos_current_mean_max_missing	0
ccf_cc_amt_drawings_pos_current_mean_min_missing	0
ccf_cc_amt_drawings_pos_current_mean_count_missing	0
ccf_cc_amt_drawings_pos_current_max_mean_missing	0
ccf_cc_amt_drawings_pos_current_max_max_missing	0
ccf_cc_amt_drawings_pos_current_max_min_missing	0
ccf_cc_amt_drawings_pos_current_max_count_missing	0
ccf_cc_amt_drawings_pos_current_min_mean_missing	0
ccf_cc_amt_drawings_pos_current_min_max_missing	0
ccf_cc_amt_drawings_pos_current_min_min_missing	0
ccf_cc_amt_drawings_pos_current_min_count_missing	0
ccf_cc_amt_drawings_pos_current_count_mean_missing	0
ccf_cc_amt_drawings_pos_current_count_max_missing	0
ccf_cc_amt_drawings_pos_current_count_min_missing	0
ccf_cc_amt_drawings_pos_current_count_count_missing	0
ccf_cc_amt_inst_min_regularity_mean_mean_missing	0
ccf_cc_amt_inst_min_regularity_mean_max_missing	0
ccf_cc_amt_inst_min_regularity_mean_min_missing	0
ccf_cc_amt_inst_min_regularity_mean_count_missing	0
ccf_cc_amt_inst_min_regularity_max_mean_missing	0
ccf_cc_amt_inst_min_regularity_max_max_missing	0
ccf_cc_amt_inst_min_regularity_max_min_missing	0
ccf_cc_amt_inst_min_regularity_max_count_missing	0
ccf_cc_amt_inst_min_regularity_min_mean_missing	0
ccf_cc_amt_inst_min_regularity_min_max_missing	0
ccf_cc_amt_inst_min_regularity_min_min_missing	0
ccf_cc_amt_inst_min_regularity_min_count_missing	0
ccf_cc_amt_inst_min_regularity_count_mean_missing	0
ccf_cc_amt_inst_min_regularity_count_max_missing	0

ccf_cc_amt_inst_min_regularity_count_min_missing	0
ccf_cc_amt_inst_min_regularity_count_count_missing	0
ccf_cc_amt_payment_current_mean_mean_missing	0
ccf_cc_amt_payment_current_mean_max_missing	0
ccf_cc_amt_payment_current_mean_min_missing	0
ccf_cc_amt_payment_current_mean_count_missing	0
ccf_cc_amt_payment_current_max_mean_missing	0
ccf_cc_amt_payment_current_max_max_missing	0
ccf_cc_amt_payment_current_max_min_missing	0
ccf_cc_amt_payment_current_max_count_missing	0
ccf_cc_amt_payment_current_min_mean_missing	0
ccf_cc_amt_payment_current_min_max_missing	0
ccf_cc_amt_payment_current_min_min_missing	0
ccf_cc_amt_payment_current_min_count_missing	0
ccf_cc_amt_payment_current_count_mean_missing	0
ccf_cc_amt_payment_current_count_max_missing	0
ccf_cc_amt_payment_current_count_min_missing	0
ccf_cc_amt_payment_current_count_count_missing	0
ccf_cc_amt_payment_total_current_mean_mean_missing	0
ccf_cc_amt_payment_total_current_mean_max_missing	0
ccf_cc_amt_payment_total_current_mean_min_missing	0
ccf_cc_amt_payment_total_current_mean_count_missing	0
ccf_cc_amt_payment_total_current_max_mean_missing	0
ccf_cc_amt_payment_total_current_max_max_missing	0
ccf_cc_amt_payment_total_current_max_min_missing	0
ccf_cc_amt_payment_total_current_max_count_missing	0
ccf_cc_amt_payment_total_current_min_mean_missing	0
ccf_cc_amt_payment_total_current_min_max_missing	0
ccf_cc_amt_payment_total_current_min_min_missing	0
ccf_cc_amt_payment_total_current_min_count_missing	0
ccf_cc_amt_payment_total_current_count_mean_missing	0
ccf_cc_amt_payment_total_current_count_max_missing	0
ccf_cc_amt_payment_total_current_count_min_missing	0
ccf_cc_amt_payment_total_current_count_count_missing	0
ccf_cc_amt_receivable_principal_mean_mean_missing	0
ccf_cc_amt_receivable_principal_mean_max_missing	0
ccf_cc_amt_receivable_principal_mean_min_missing	0
ccf_cc_amt_receivable_principal_mean_count_missing	0
ccf_cc_amt_receivable_principal_max_mean_missing	0
ccf_cc_amt_receivable_principal_max_max_missing	0
ccf_cc_amt_receivable_principal_max_min_missing	0
ccf_cc_amt_receivable_principal_max_count_missing	0
ccf_cc_amt_receivable_principal_min_mean_missing	0
ccf_cc_amt_receivable_principal_min_max_missing	0
ccf_cc_amt_receivable_principal_min_min_missing	0
ccf_cc_amt_receivable_principal_min_count_missing	0
ccf_cc_amt_receivable_principal_count_mean_missing	0
ccf_cc_amt_receivable_principal_count_max_missing	0
ccf_cc_amt_receivable_principal_count_min_missing	0
ccf_cc_amt_receivable_principal_count_count_missing	0
ccf_cc_amt_recivable_mean_mean_missing	0
ccf_cc_amt_recivable_mean_max_missing	0
ccf_cc_amt_recivable_mean_min_missing	0
ccf_cc_amt_recivable_mean_count_missing	0
ccf_cc_amt_recivable_max_mean_missing	0
ccf_cc_amt_recivable_max_max_missing	0
ccf_cc_amt_recivable_max_min_missing	0
ccf_cc_amt_recivable_max_count_missing	0
ccf_cc_amt_recivable_min_mean_missing	0
ccf_cc_amt_recivable_min_max_missing	0
ccf_cc_amt_recivable_min_min_missing	0
ccf_cc_amt_recivable_min_count_missing	0
ccf_cc_amt_recivable_count_mean_missing	0
ccf_cc_amt_recivable_count_max_missing	0
ccf_cc_amt_recivable_count_min_missing	0
ccf_cc_amt_recivable_count_count_missing	0
ccf_cc_amt_total_receivable_mean_mean_missing	0
ccf_cc_amt_total_receivable_mean_max_missing	0
ccf_cc_amt_total_receivable_mean_min_missing	0

ccf_cc_amt_total_receivable_mean_count_missing	0
ccf_cc_amt_total_receivable_max_mean_missing	0
ccf_cc_amt_total_receivable_max_max_missing	0
ccf_cc_amt_total_receivable_max_min_missing	0
ccf_cc_amt_total_receivable_max_count_missing	0
ccf_cc_amt_total_receivable_min_mean_missing	0
ccf_cc_amt_total_receivable_min_max_missing	0
ccf_cc_amt_total_receivable_min_min_missing	0
ccf_cc_amt_total_receivable_min_count_missing	0
ccf_cc_amt_total_receivable_count_mean_missing	0
ccf_cc_amt_total_receivable_count_max_missing	0
ccf_cc_amt_total_receivable_count_min_missing	0
ccf_cc_amt_total_receivable_count_count_missing	0
ccf_cc_cnt_drawings_atm_current_mean_mean_missing	0
ccf_cc_cnt_drawings_atm_current_mean_max_missing	0
ccf_cc_cnt_drawings_atm_current_mean_min_missing	0
ccf_cc_cnt_drawings_atm_current_mean_count_missing	0
ccf_cc_cnt_drawings_atm_current_max_mean_missing	0
ccf_cc_cnt_drawings_atm_current_max_max_missing	0
ccf_cc_cnt_drawings_atm_current_max_min_missing	0
ccf_cc_cnt_drawings_atm_current_max_count_missing	0
ccf_cc_cnt_drawings_atm_current_min_mean_missing	0
ccf_cc_cnt_drawings_atm_current_min_max_missing	0
ccf_cc_cnt_drawings_atm_current_min_min_missing	0
ccf_cc_cnt_drawings_atm_current_min_count_missing	0
ccf_cc_cnt_drawings_atm_current_count_mean_missing	0
ccf_cc_cnt_drawings_atm_current_count_max_missing	0
ccf_cc_cnt_drawings_atm_current_count_min_missing	0
ccf_cc_cnt_drawings_atm_current_count_count_missing	0
ccf_cc_cnt_drawings_current_mean_mean_missing	0
ccf_cc_cnt_drawings_current_mean_max_missing	0
ccf_cc_cnt_drawings_current_mean_min_missing	0
ccf_cc_cnt_drawings_current_mean_count_missing	0
ccf_cc_cnt_drawings_current_max_mean_missing	0
ccf_cc_cnt_drawings_current_max_max_missing	0
ccf_cc_cnt_drawings_current_max_min_missing	0
ccf_cc_cnt_drawings_current_min_mean_missing	0
ccf_cc_cnt_drawings_current_min_max_missing	0
ccf_cc_cnt_drawings_current_min_min_missing	0
ccf_cc_cnt_drawings_current_min_count_missing	0
ccf_cc_cnt_drawings_current_count_mean_missing	0
ccf_cc_cnt_drawings_current_count_max_missing	0
ccf_cc_cnt_drawings_current_count_min_missing	0
ccf_cc_cnt_drawings_current_count_count_missing	0
ccf_cc_cnt_drawings_other_current_mean_mean_missing	0
ccf_cc_cnt_drawings_other_current_mean_max_missing	0
ccf_cc_cnt_drawings_other_current_mean_min_missing	0
ccf_cc_cnt_drawings_other_current_mean_count_missing	0
ccf_cc_cnt_drawings_other_current_max_mean_missing	0
ccf_cc_cnt_drawings_other_current_max_max_missing	0
ccf_cc_cnt_drawings_other_current_max_min_missing	0
ccf_cc_cnt_drawings_other_current_max_count_missing	0
ccf_cc_cnt_drawings_other_current_min_mean_missing	0
ccf_cc_cnt_drawings_other_current_min_max_missing	0
ccf_cc_cnt_drawings_other_current_min_min_missing	0
ccf_cc_cnt_drawings_other_current_min_count_missing	0
ccf_cc_cnt_drawings_other_current_count_mean_missing	0
ccf_cc_cnt_drawings_other_current_count_max_missing	0
ccf_cc_cnt_drawings_other_current_count_min_missing	0
ccf_cc_cnt_drawings_other_current_count_count_missing	0
ccf_cc_cnt_drawings_pos_current_mean_mean_missing	0
ccf_cc_cnt_drawings_pos_current_mean_max_missing	0
ccf_cc_cnt_drawings_pos_current_mean_min_missing	0
ccf_cc_cnt_drawings_pos_current_mean_count_missing	0
ccf_cc_cnt_drawings_pos_current_max_mean_missing	0
ccf_cc_cnt_drawings_pos_current_max_max_missing	0
ccf_cc_cnt_drawings_pos_current_max_min_missing	0
ccf_cc_cnt_drawings_pos_current_max_count_missing	0

ccf_cc_cnt_drawings_pos_current_min_mean_missing	0
ccf_cc_cnt_drawings_pos_current_min_max_missing	0
ccf_cc_cnt_drawings_pos_current_min_min_missing	0
ccf_cc_cnt_drawings_pos_current_min_count_missing	0
ccf_cc_cnt_drawings_pos_current_count_mean_missing	0
ccf_cc_cnt_drawings_pos_current_count_max_missing	0
ccf_cc_cnt_drawings_pos_current_count_min_missing	0
ccf_cc_cnt_drawings_pos_current_count_count_missing	0
ccf_cc_cnt_instalment_mature_cum_mean_mean_missing	0
ccf_cc_cnt_instalment_mature_cum_mean_max_missing	0
ccf_cc_cnt_instalment_mature_cum_mean_min_missing	0
ccf_cc_cnt_instalment_mature_cum_mean_count_missing	0
ccf_cc_cnt_instalment_mature_cum_max_mean_missing	0
ccf_cc_cnt_instalment_mature_cum_max_max_missing	0
ccf_cc_cnt_instalment_mature_cum_max_min_missing	0
ccf_cc_cnt_instalment_mature_cum_max_count_missing	0
ccf_cc_cnt_instalment_mature_cum_min_mean_missing	0
ccf_cc_cnt_instalment_mature_cum_min_max_missing	0
ccf_cc_cnt_instalment_mature_cum_min_min_missing	0
ccf_cc_cnt_instalment_mature_cum_min_count_missing	0
ccf_cc_sk_dpd_mean_mean_missing	0
ccf_cc_sk_dpd_mean_max_missing	0
ccf_cc_sk_dpd_mean_min_missing	0
ccf_cc_sk_dpd_mean_count_missing	0
ccf_cc_sk_dpd_max_mean_missing	0
ccf_cc_sk_dpd_max_max_missing	0
ccf_cc_sk_dpd_max_min_missing	0
ccf_cc_sk_dpd_max_count_missing	0
ccf_cc_sk_dpd_min_mean_missing	0
ccf_cc_sk_dpd_min_max_missing	0
ccf_cc_sk_dpd_min_min_missing	0
ccf_cc_sk_dpd_min_count_missing	0
ccf_cc_sk_dpd_count_mean_missing	0
ccf_cc_sk_dpd_count_max_missing	0
ccf_cc_sk_dpd_count_min_missing	0
ccf_cc_sk_dpd_count_count_missing	0
ccf_cc_sk_dpd_def_mean_mean_missing	0
ccf_cc_sk_dpd_def_mean_max_missing	0
ccf_cc_sk_dpd_def_mean_min_missing	0
ccf_cc_sk_dpd_def_mean_count_missing	0
ccf_cc_sk_dpd_def_max_mean_missing	0
ccf_cc_sk_dpd_def_max_max_missing	0
ccf_cc_sk_dpd_def_max_min_missing	0
ccf_cc_sk_dpd_def_max_count_missing	0
ccf_cc_sk_dpd_def_min_mean_missing	0
ccf_cc_sk_dpd_def_min_max_missing	0
ccf_cc_sk_dpd_def_min_min_missing	0
ccf_cc_sk_dpd_def_min_count_missing	0
ccf_cc_sk_dpd_def_count_mean_missing	0
ccf_cc_sk_dpd_def_count_max_missing	0
ccf_cc_sk_dpd_def_count_min_missing	0
ccf_cc_sk_dpd_def_count_count_missing	0
instf_sk_id_prev_mean_missing	0
instf_sk_id_prev_max_missing	0
instf_sk_id_prev_min_missing	0
instf_sk_id_prev_count_missing	0
instf_inst_sk_id_curr_mean_mean_missing	0
instf_inst_sk_id_curr_mean_max_missing	0
instf_inst_sk_id_curr_mean_min_missing	0
instf_inst_sk_id_curr_mean_count_missing	0
instf_inst_sk_id_curr_max_mean_missing	0
instf_inst_sk_id_curr_max_max_missing	0
instf_inst_sk_id_curr_max_min_missing	0
instf_inst_sk_id_curr_max_count_missing	0
instf_inst_sk_id_curr_min_mean_missing	0

instf_inst_sk_id_curr_min_max_missing	0
instf_inst_sk_id_curr_min_min_missing	0
instf_inst_sk_id_curr_min_count_missing	0
instf_inst_sk_id_curr_count_mean_missing	0
instf_inst_sk_id_curr_count_max_missing	0
instf_inst_sk_id_curr_count_min_missing	0
instf_inst_sk_id_curr_count_count_missing	0
instf_inst_num_instalment_version_mean_mean_missing	0
instf_inst_num_instalment_version_mean_max_missing	0
instf_inst_num_instalment_version_mean_min_missing	0
instf_inst_num_instalment_version_mean_count_missing	0
instf_inst_num_instalment_version_max_mean_missing	0
instf_inst_num_instalment_version_max_max_missing	0
instf_inst_num_instalment_version_max_min_missing	0
instf_inst_num_instalment_version_max_count_missing	0
instf_inst_num_instalment_version_min_mean_missing	0
instf_inst_num_instalment_version_min_max_missing	0
instf_inst_num_instalment_version_min_min_missing	0
instf_inst_num_instalment_version_min_count_missing	0
instf_inst_num_instalment_version_mean_count_missing	0
instf_inst_num_instalment_version_count_max_missing	0
instf_inst_num_instalment_version_count_min_missing	0
instf_inst_num_instalment_version_count_count_missing	0
instf_inst_num_instalment_number_mean_mean_missing	0
instf_inst_num_instalment_number_mean_max_missing	0
instf_inst_num_instalment_number_mean_min_missing	0
instf_inst_num_instalment_number_mean_count_missing	0
instf_inst_num_instalment_number_max_mean_missing	0
instf_inst_num_instalment_number_max_max_missing	0
instf_inst_num_instalment_number_max_min_missing	0
instf_inst_num_instalment_number_max_count_missing	0
instf_inst_num_instalment_number_min_mean_missing	0
instf_inst_num_instalment_number_min_max_missing	0
instf_inst_num_instalment_number_min_min_missing	0
instf_inst_num_instalment_number_min_count_missing	0
instf_inst_num_instalment_number_count_mean_missing	0
instf_inst_num_instalment_number_count_max_missing	0
instf_inst_num_instalment_number_count_min_missing	0
instf_inst_num_instalment_number_count_count_missing	0
instf_inst_days_instalment_mean_mean_missing	0
instf_inst_days_instalment_mean_max_missing	0
instf_inst_days_instalment_mean_min_missing	0
instf_inst_days_instalment_mean_count_missing	0
instf_inst_days_instalment_max_mean_missing	0
instf_inst_days_instalment_max_max_missing	0
instf_inst_days_instalment_max_min_missing	0
instf_inst_days_instalment_max_count_missing	0
instf_inst_days_instalment_min_mean_missing	0
instf_inst_days_instalment_min_max_missing	0
instf_inst_days_instalment_min_min_missing	0
instf_inst_days_instalment_min_count_missing	0
instf_inst_days_instalment_min_mean_mean_missing	0
instf_inst_days_instalment_min_max_mean_missing	0
instf_inst_days_instalment_min_min_mean_missing	0
instf_inst_days_instalment_min_mean_max_missing	0
instf_inst_days_instalment_min_min_max_missing	0
instf_inst_days_entry_payment_mean_mean_missing	0
instf_inst_days_entry_payment_mean_max_missing	0
instf_inst_days_entry_payment_mean_min_missing	0
instf_inst_days_entry_payment_mean_count_missing	0
instf_inst_days_entry_payment_max_mean_mean_missing	0
instf_inst_days_entry_payment_max_max_missing	0
instf_inst_days_entry_payment_max_min_missing	0
instf_inst_days_entry_payment_max_count_missing	0
instf_inst_days_entry_payment_min_mean_mean_missing	0
instf_inst_days_entry_payment_min_max_mean_missing	0
instf_inst_days_entry_payment_min_min_mean_missing	0
instf_inst_days_entry_payment_min_mean_max_missing	0
instf_inst_days_entry_payment_min_min_max_missing	0
instf_inst_days_entry_payment_count_mean_mean_missing	0
instf_inst_days_entry_payment_count_max_mean_missing	0

instf_inst_days_entry_payment_count_min_missing	0
instf_inst_days_entry_payment_count_count_missing	0
instf_inst_amt_instalment_mean_mean_missing	0
instf_inst_amt_instalment_mean_max_missing	0
instf_inst_amt_instalment_mean_min_missing	0
instf_inst_amt_instalment_mean_count_missing	0
instf_inst_amt_instalment_max_mean_missing	0
instf_inst_amt_instalment_max_max_missing	0
instf_inst_amt_instalment_max_min_missing	0
instf_inst_amt_instalment_max_count_missing	0
instf_inst_amt_instalment_min_mean_missing	0
instf_inst_amt_instalment_min_max_missing	0
instf_inst_amt_instalment_min_min_missing	0
instf_inst_amt_instalment_min_count_missing	0
instf_inst_amt_instalment_count_mean_missing	0
instf_inst_amt_instalment_count_max_missing	0
instf_inst_amt_instalment_count_min_missing	0
instf_inst_amt_instalment_count_count_missing	0
instf_inst_amt_payment_mean_mean_missing	0
instf_inst_amt_payment_mean_max_missing	0
instf_inst_amt_payment_mean_min_missing	0
instf_inst_amt_payment_mean_count_missing	0
instf_inst_amt_payment_max_mean_missing	0
instf_inst_amt_payment_max_max_missing	0
instf_inst_amt_payment_max_min_missing	0
instf_inst_amt_payment_max_count_missing	0
instf_inst_amt_payment_min_mean_missing	0
instf_inst_amt_payment_min_max_missing	0
instf_inst_amt_payment_min_min_missing	0
instf_inst_amt_payment_min_count_missing	0
instf_inst_amt_payment_count_mean_missing	0
instf_inst_amt_payment_count_max_missing	0
instf_inst_amt_payment_count_min_missing	0
instf_inst_amt_payment_count_count_missing	0

6. Feature Engineering

```
In [ ]: df2 = df1.copy()
```

6.1 Lista de Hipóteses

Bloco 1 - (1 a 20)

H1 — O tipo de contrato influencia a probabilidade de inadimplência.

Espera-se que clientes com Revolving loans apresentem maior risco, por serem créditos de curto prazo e com juros mais altos.

H2 — O gênero está associado ao risco de inadimplência.

Diferenças de comportamento financeiro entre homens e mulheres podem refletir na taxa de pagamento.

H3 — Clientes que possuem carro têm menor probabilidade de inadimplência.

A posse de um veículo pode indicar maior estabilidade financeira.

H4 — Ter imóvel próprio reduz a chance de inadimplência.

A posse de um bem fixo pode refletir segurança financeira e histórico de crédito mais sólido.

H5 — O número de filhos influencia o risco de inadimplência.

Famílias maiores tendem a ter mais despesas fixas, o que pode aumentar a probabilidade de atraso.

H6 — Renda total mais alta está associada a menor inadimplência.

Clientes com maior capacidade financeira têm mais margem para honrar dívidas.

H7 — Valores de crédito mais altos estão relacionados a maior risco de inadimplência.

Empréstimos mais elevados podem sobrecarregar o orçamento, especialmente em clientes de menor renda.

H8 — Parcelas mensais maiores aumentam a chance de inadimplência.

Pagamentos altos podem comprometer o fluxo de caixa do cliente.

H9 — O valor dos bens adquiridos influencia o risco de inadimplência.

Compras de alto valor podem indicar endividamento mais elevado e risco financeiro maior.

H10 — O tipo de companhia ao solicitar crédito está associado ao risco de inadimplência.

Clientes que aplicam sozinhos (Unaccompanied) podem ter comportamento financeiro diferente de casais ou famílias.

H11 — O tipo de fonte de renda impacta a inadimplência.

Trabalhadores formais tendem a apresentar menor risco em comparação a autônomos ou desempregados.

H12 — Níveis educacionais mais altos estão associados a menor risco de inadimplência.

Maior escolaridade costuma correlacionar-se com renda e estabilidade financeira superiores.

H13 — O estado civil influencia o risco de inadimplência.

Casados podem apresentar maior estabilidade financeira do que solteiros ou divorciados.

H14 — O tipo de moradia está associado à inadimplência.

Clientes que vivem em imóvel próprio (House/apartment) tendem a ter menor risco que os que vivem com parentes ou em moradias alugadas.

H15 — Regiões mais populosas apresentam maior inadimplência.

Ambientes urbanos tendem a ter maior competitividade econômica e endividamento.

H16 — A idade do cliente influencia o risco de inadimplência.

Clientes mais jovens podem ter maior propensão ao risco e menor estabilidade financeira.

H17 — O tempo de emprego está inversamente relacionado à inadimplência.

Maior tempo de vínculo empregatício indica estabilidade e menor risco.

H18 — Clientes com registro mais antigo (mais tempo na base) apresentam menor inadimplência.

Maior relacionamento com a instituição tende a refletir em melhor histórico de crédito.

H19 — A data de emissão do documento influencia o risco de inadimplência.

Documentos emitidos recentemente podem indicar instabilidade de identidade ou reemissão por problemas anteriores.

H20 — Possuir celular registrado está associado a menor inadimplência.

A ausência de telefone móvel pode refletir menor acesso à comunicação e a serviços bancários digitais.

Bloco 2 - (21 a 40)

H21 — Ter telefone de trabalho cadastrado reduz o risco de inadimplência.

O vínculo de emprego validado pode indicar estabilidade profissional e maior confiabilidade de renda.

H22 — Clientes com telefone profissional informado são menos propensos à inadimplência.

Esse dado tende a ser fornecido por trabalhadores formais, refletindo melhor estabilidade econômica.

H23 — Possuir celular de contato registrado está associado a menor risco de inadimplência.

Indica maior facilidade de comunicação e acompanhamento pelo credor.

H24 — Ter telefone fixo registrado reduz o risco de inadimplência.

Pode estar associado a residências fixas e maior estabilidade social.

H25 — Clientes com e-mail cadastrado tendem a apresentar menor inadimplência.

Indica maior formalização, acesso digital e facilidade de comunicação para cobranças ou notificações.

H26 — O tipo de ocupação influencia o risco de inadimplência.

Trabalhadores autônomos e temporários tendem a ter rendas mais instáveis, aumentando o risco em relação a empregados formais ou servidores públicos.

H27 — O número de membros na família está positivamente relacionado à inadimplência.

Famílias maiores implicam maiores despesas fixas e menor margem financeira para pagamentos.

H28 — Clientes de regiões com menor rating apresentam maior probabilidade de inadimplência.

O rating regional pode refletir condições socioeconômicas e acesso a crédito na área.

H29 — O rating da cidade do cliente influencia o risco de inadimplência.

Regiões urbanas com piores indicadores econômicos podem concentrar clientes com maior risco financeiro.

H30 — O dia da semana em que o pedido de crédito é feito influencia a probabilidade de inadimplência.

Solicitações feitas em dias úteis (especialmente início de semana) podem vir de perfis mais organizados do que pedidos feitos no fim de semana.

H31 — O horário da solicitação de crédito está associado à inadimplência.

Pedidos feitos em horários atípicos (tarde da noite, madrugada) podem estar ligados a comportamentos impulsivos ou maior risco.

H32 — Morar em região diferente da registrada aumenta o risco de inadimplência.

Indica mobilidade recente ou inconsistência cadastral, fatores ligados a menor confiabilidade.

H33 — Trabalhar em região diferente da registrada está associado a maior risco.

Pode representar deslocamento elevado ou instabilidade de vínculo profissional.

H34 — Residir e trabalhar em regiões diferentes pode aumentar a probabilidade de inadimplência.

O deslocamento pode indicar custos adicionais e menor tempo disponível, afetando estabilidade

financeira.

H35 — Morar em cidade diferente da registrada aumenta a chance de inadimplência.

Diferenças entre endereço oficial e real podem sinalizar inconsistências cadastrais ou instabilidade residencial.

H36 — Trabalhar em cidade diferente da registrada aumenta o risco de inadimplência.

Indica possível deslocamento elevado ou emprego informal.

H37 — Residir em cidade diferente do local de trabalho influencia o risco de inadimplência.

Longos deslocamentos podem representar custos maiores e menor estabilidade profissional.

H38 — O tipo de organização empregadora está associado ao risco de inadimplência.

Funcionários públicos e de grandes empresas tendem a apresentar menor risco comparado a autônomos ou empregados de pequenas empresas.

H39 — O score externo 2 está inversamente relacionado à inadimplência.

Quanto maior o valor de ext_source_2, menor o risco de inadimplência, pois indica melhor avaliação externa de crédito.

H40 — O score externo 3 está inversamente relacionado à inadimplência.

Valores altos refletem maior confiança e menor risco de crédito segundo fontes externas.

Bloco 3 - (41 a 72)

H41 — A quantidade média de elevadores no imóvel está inversamente relacionada à probabilidade de inadimplência.

Imóveis com mais elevadores geralmente pertencem a edifícios de maior padrão, o que pode indicar melhor condição socioeconômica e menor risco de crédito.

H42 — O tipo de habitação influencia o risco de inadimplência.

Clientes que vivem em casas próprias ou apartamentos tendem a ter menor inadimplência do que aqueles em residências alugadas ou de terceiros, refletindo maior estabilidade patrimonial.

H43 — O material predominante das paredes do imóvel está associado ao risco de inadimplência.

Materiais mais duráveis ou de melhor qualidade (como pedra ou tijolo) tendem a indicar imóveis de padrão mais elevado e clientes de melhor condição financeira.

H44 — Um maior número de observações ou de inadimplentes no círculo social de 30 dias aumenta a probabilidade de inadimplência do cliente.

Indivíduos inseridos em grupos sociais com maior incidência de inadimplência podem compartilhar perfis de risco semelhantes ou vulnerabilidade econômica.

H45 — A inadimplência no círculo social de 60 dias está positivamente associada ao risco de inadimplência individual.

O comportamento financeiro de contatos próximos em períodos mais longos tende a refletir influência social e condições econômicas semelhantes.

H46 — O tempo desde a última troca de telefone está relacionado à probabilidade de inadimplência.

Cientes que trocam de número com frequência podem apresentar maior instabilidade ou dificuldade de contato, fatores que aumentam o risco de crédito.

H47 — A quantidade de documentos entregues está inversamente associada à inadimplência.

O envio de mais documentos indica maior transparência e conformidade cadastral, enquanto poucos documentos podem sinalizar risco de fraude ou baixa confiabilidade.

Bloco 4 - (73 a 90)

H48 — Histórico de consultas e valores de crédito influenciam a probabilidade de inadimplência.

Essas variáveis representam tanto a frequência de solicitações de crédito quanto os valores médios e totais de crédito contratados. Clientes que realizam muitas consultas ao bureau de crédito, ou possuem montantes elevados de crédito ativo, tendem a demonstrar maior exposição financeira e, consequentemente, maior risco de inadimplência.

H49 — Características do imóvel e situação de emprego influenciam a capacidade de pagamento.

As variáveis relacionadas à estrutura do imóvel e à situação de emprego, ajudam a estimar a estabilidade econômica e social do cliente.

6.2 Derivação de Variáveis

```
In [6]: df2['days_employed_anom'] = df2["days_employed"] == 365243
df2['days_birth'] = df2['days_birth'].abs()
df2['days_employed'] = df2['days_employed'].abs()
df2['days_id_publish'] = df2['days_id_publish'].abs()
df2['days_registration'] = df2['days_registration'].abs()
df2['age_years'] = df2['days_birth'] / 365
df2['employment_years'] = df2['days_employed'] / 365
df2['id_publish_years'] = df2['days_id_publish'] / 365

df2['annuity_burden_ratio'] = df2['amt_annuity'] / (df2['bur_amt_credit_sum_mean'] + 1e-6)
df2['phone_change_rate'] = df2['days_last_phone_change'] / (df2['days_birth'] + 1e-6)
df2['id_change_rate'] = df2['days_id_publish'] / (df2['days_birth'] + 1e-6)
df2['ext_source_weighted_mean'] = (
    0.5 * df2['ext_source_2'] + 0.3 * df2['ext_source_3'] + 0.2 * df2['ext_source_1']
)
df2['ext_source_max'] = df2[['ext_source_1', 'ext_source_2', 'ext_source_3']].max(axis=1)
df2['amt_application_credit_diff'] = df2['prev_amt_application_mean'] - df2['bur_amt_credit_sum']
df2['amt_application_goods_diff'] = df2['prev_amt_application_mean'] - df2['amt_goods_price']

flag_document_cols = [col for col in df2.columns if col.startswith('flag_document_')]
df2['num_flag_document'] = df2[flag_document_cols].sum(axis=1)
df2['credit_to_income_ratio'] = df2['bur_amt_credit_sum_mean'] / (df2['amt_income_total'] + 1e-6)
df2['annuity_to_income_ratio'] = df2['amt_annuity'] / (df2['amt_income_total'] + 1e-5)
df2['payment_rate'] = df2['amt_annuity'] / (df2['bur_amt_credit_sum_mean'] + 1e-5)
df2['goods_to_credit_ratio'] = df2['bur_amt_credit_sum_mean'] / (df2['amt_goods_price'] + 1e-5)
df2['credit_efficiency'] = df2['bur_amt_credit_sum_mean'] / (df2['prev_amt_application_mean'] + 1e-5)
df2['employment_life_ratio'] = df2['days_employed'] / (df2['days_birth'])
df2['bureau_request_intensity'] = (
    df2['amt_req_credit_bureau_day'] +
    df2['amt_req_credit_bureau_week'] +
    df2['amt_req_credit_bureau_mon'] +
    df2['amt_req_credit_bureau_qrt']
)

ext_sources = ['ext_source_1', 'ext_source_2', 'ext_source_3']
df2['ext_source_mean'] = df2[ext_sources].mean(axis=1)
df2['ext_source_range'] = df2[ext_sources].max(axis=1) - df2[ext_sources].min(axis=1)
df2['ext_source_std'] = df2[ext_sources].std(axis=1)
df2['ext_source_agreement'] = 1 / (df2['ext_source_std'] + 1e-5)
```

```

df2['decision_credit_diff'] = df2['prev_days_decision_mean'] / (df2['bur_amt_credit_sum_mean'])
df2['bureau_to_credit_ratio'] = df2['bureau_request_intensity'] / (df2['bur_amt_credit_sum_mean'])

df2['credit_per_employment_year'] = df2['bur_amt_credit_sum_mean'] / (df2['employment_years'] + 1)
df2['annuity_per_employment_year'] = df2['amt_annuity'] / (df2['employment_years'] + 1e-6)

df2['income_per_employed'] = df2['amt_income_total'] / ((df2['days_employed']+ 1e-5))
df2['income_per_birth'] = df2['amt_income_total'] / (df2['days_birth'])
df2['own_car_age_birth_ratio'] = df2['own_car_age'] / (df2['days_birth'])
df2['own_car_age_employed_ratio'] = df2['own_car_age'] / ((df2['days_employed']+ 1e-5))
df2['days_since_last_employment_until_application'] = df2['days_employed'] - df2['days_birth']

poly_vars = [
    'days_birth',
    'payment_rate',
    'ext_source_mean',
]
X_poly = df2[poly_vars].fillna(0).copy()
poly = PolynomialFeatures(degree=3, include_bias=False)
X_poly_trans = poly.fit_transform(X_poly)
poly_feature_names = poly.get_feature_names_out(poly_vars)
df_poly = pd.DataFrame(X_poly_trans, columns=poly_feature_names, index=df2.index)
df_poly = df_poly.drop(columns=poly_vars, errors='ignore')
df2 = pd.concat([df2, df_poly], axis=1)

df2['employment_stability'] = df2['employment_years'] / (df2['age_years'] + 1e-6)
df2['id_document_age_ratio'] = df2['id_publish_years'] / (df2['age_years'] + 1e-6)
df2['phone_change_frequency'] = 1 / (df2['days_last_phone_change'] + 1e-6)

df2['recent_instability'] = df2['phone_change_rate'] + df2['id_change_rate']
df2['bureau_overdue_ratio'] = df2['bur_amt_credit_max_overdue_max'] / (df2['bur_amt_credit_sum'])
df2['credit_risk_signal'] = df2['credit_to_income_ratio'] * ((df2['ext_source_1'] + df2['ext_source_2']) / 2)
df2['annuity_per_age'] = df2['amt_annuity'] / (df2['own_car_age'] + 1e-6)
df2["overdue_flag"] = (df2["bur_amt_credit_max_overdue_max"] > 0).astype(int)
df2["entry_vs_due_ratio"] = df2["instf_inst_days_entry_payment_mean_mean"] / (df2["prev_days_fi"])
df2["ext_source_interaction"] = df2["ext_source_1"] * df2["ext_source_3"]
df2["pos_balance_range"] = df2["posf_pos_months_balance_max_max"] - df2["bur_bb_months_balance_max"]
df2["area_quality"] = df2["totalarea_mode"] / (df2["region_population_relative"] + 1e-6)

```

7. Análise Exploratória dos Dados

In []: df3 = df2.copy()

7.1 Analise Univariada

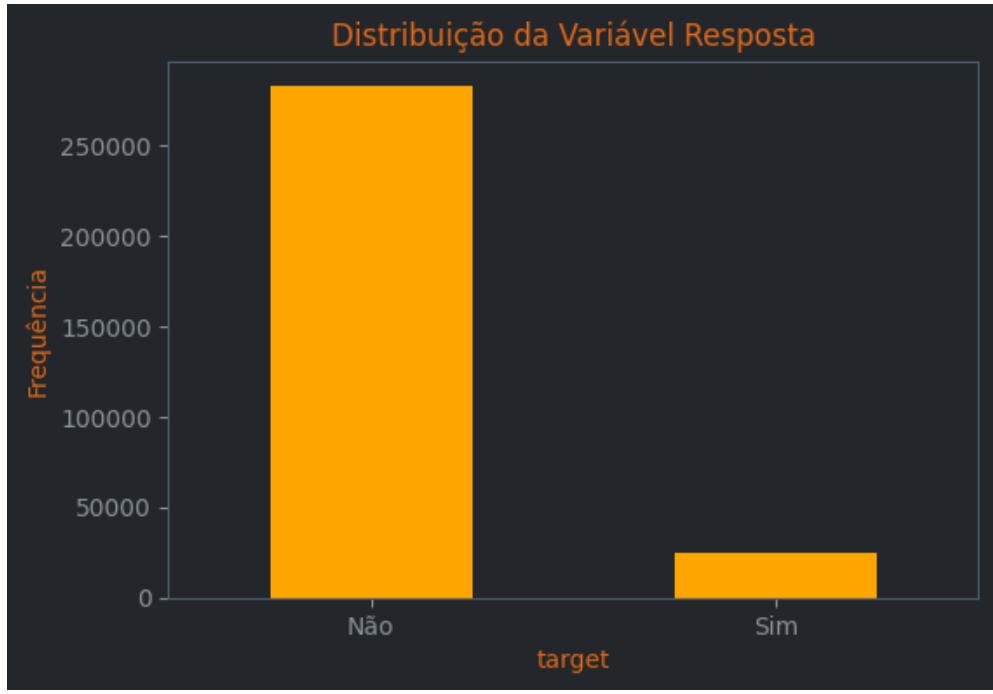
7.1.1 Variável Target

In [57]:

```

plt.figure(figsize=(6,4))
df3['target'].value_counts().plot(kind='bar', color='orange')
plt.title('Distribuição da Variável Resposta')
plt.xlabel('target')
plt.ylabel('Frequência')
plt.xticks([0,1], ['Não', 'Sim'], rotation=0)
plt.show()

```



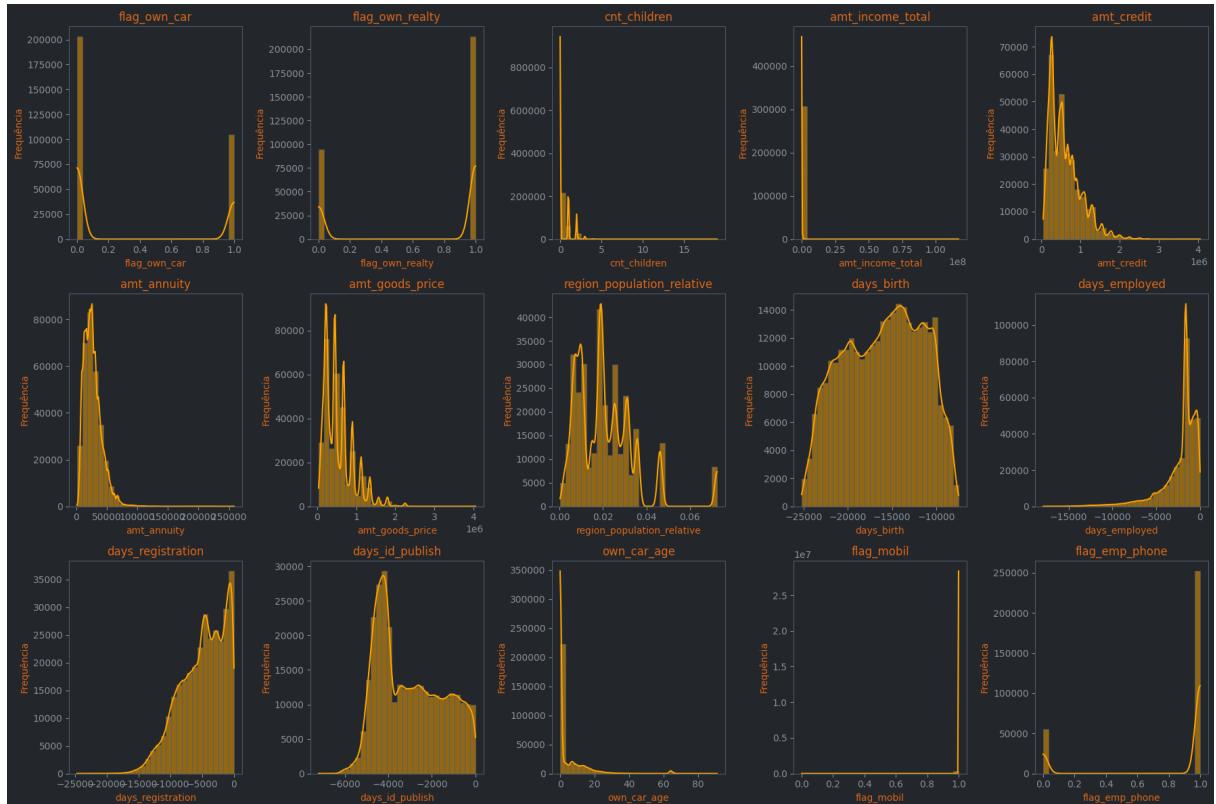
7.1.2 Variáveis Númericas

```
In [ ]: df_numeric = df3.select_dtypes(include=['number']).drop(columns=['target'])

df_numeric_1 = df_numeric.iloc[:, 0:15]      # colunas 0 a 14 (15 colunas)
df_numeric_2 = df_numeric.iloc[:, 15:30]     # colunas 15 a 29 (15 colunas)
df_numeric_3 = df_numeric.iloc[:, 30:45]     # colunas 30 a 44 (15 colunas)
df_numeric_4 = df_numeric.iloc[:, 45:60]     # colunas 45 a 59 (15 colunas)
df_numeric_5 = df_numeric.iloc[:, 60:75]     # colunas 60 a 74 (15 colunas)
```

Bloco 1 - (1 a 15)

```
In [59]: plot_numeric_block(df_numeric_1)
```



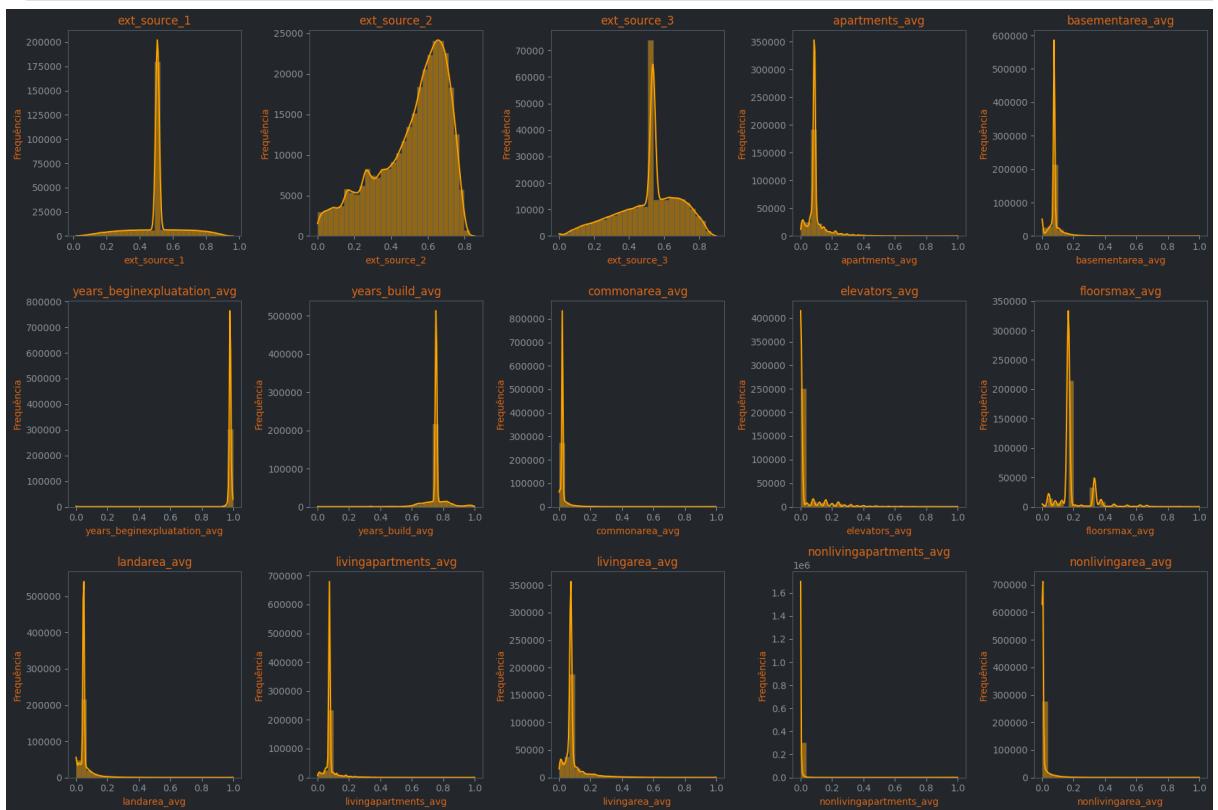
Bloco 2 - (16 a 30)

```
In [60]: plot_numeric_block(df_numeric_2)
```



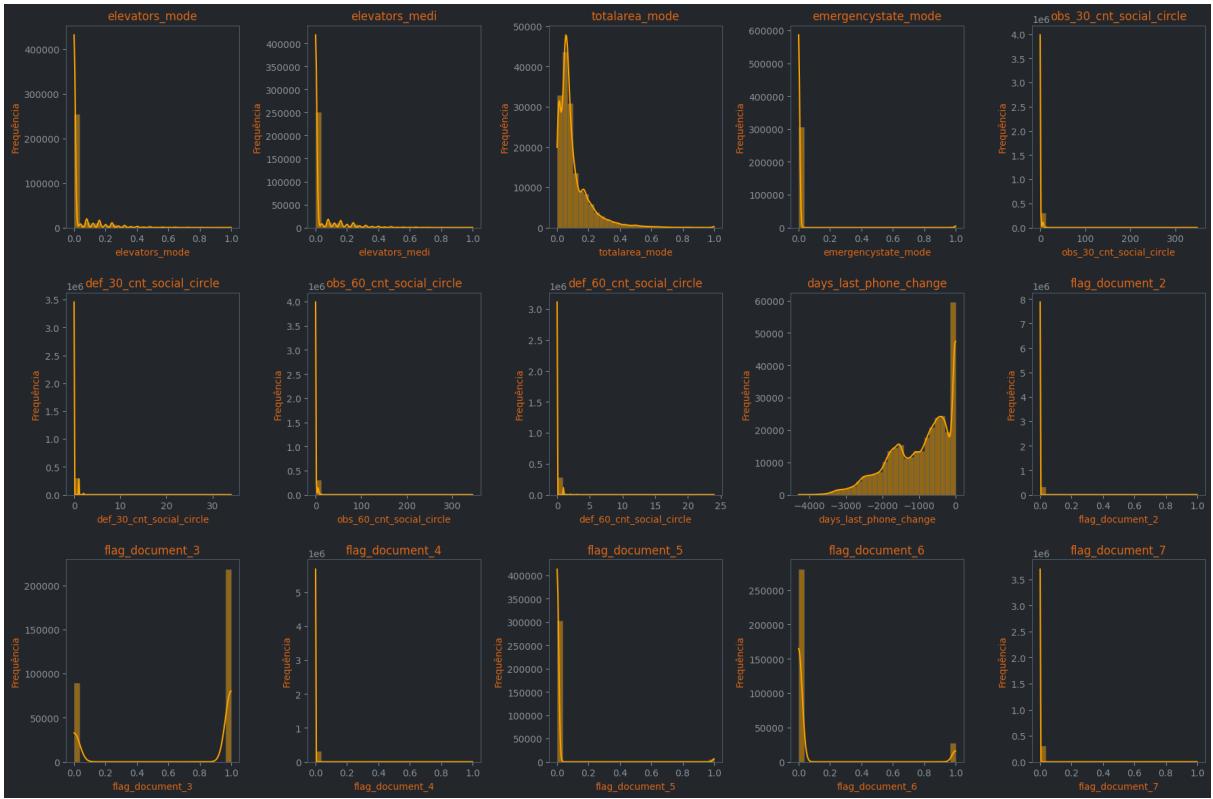
Bloco 3 - (31 a 45)

```
In [61]: plot_numeric_block(df_numeric_3)
```



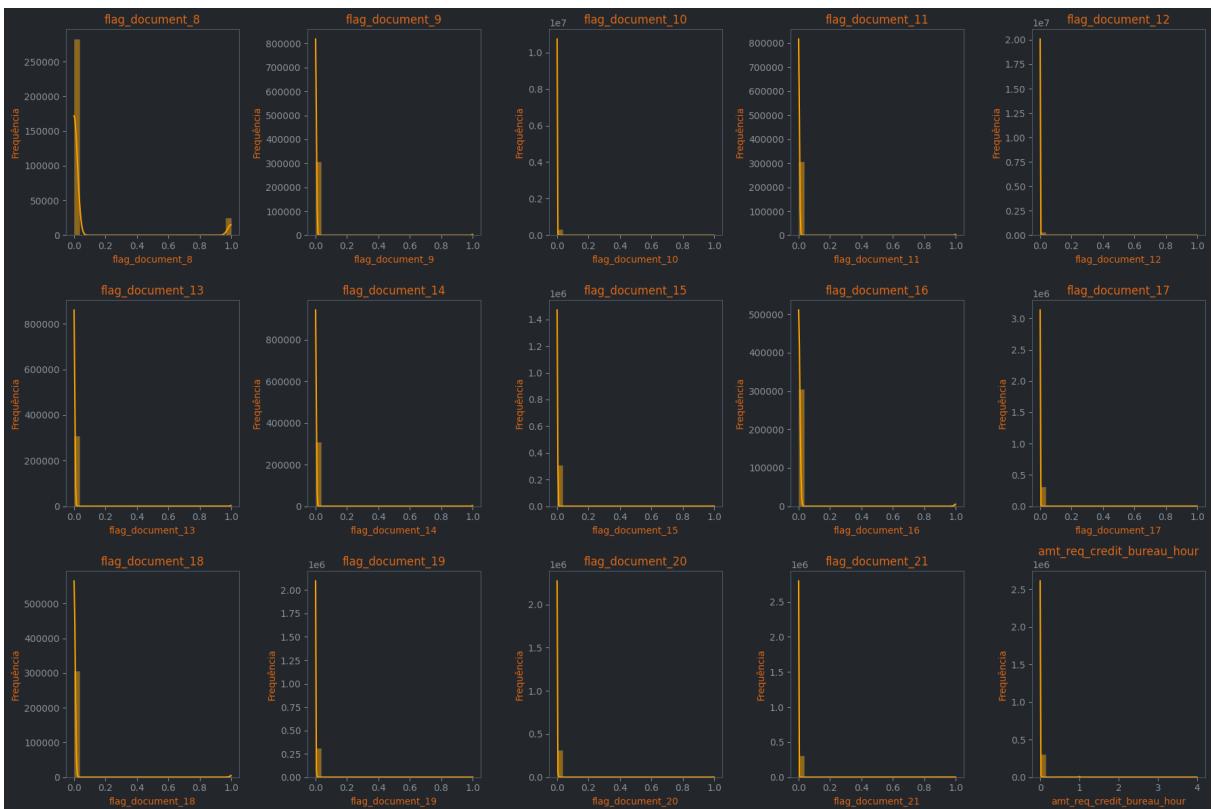
Bloco 4 - (46 a 60)

```
In [62]: plot_numeric_block(df_numeric_4)
```



Bloco 5 - (61 a 75)

In [63]: `plot_numeric_block(df_numeric_5)`



7.1.3 Variáveis Categóricas

```
In [66]: df_categorical = df3.select_dtypes(include=['object', 'category'])

cat_cols = df_categorical.columns.tolist()

df_categorical_1 = df_categorical[cat_cols[:4]]
df_categorical_2 = df_categorical[cat_cols[4:8]]
```

```
df_categorical_3 = df_categorical[cat_cols[8:13]]
```

Bloco 1 - (1 a 4)

```
In [67]: fig, axs = plt.subplots(2, 2, figsize=(16, 12))

categorical_columns = df_categorical_1.columns.tolist()
colors_list = [
    ['royalblue', 'orange', 'seagreen', 'crimson'],
    ['red', 'mediumseagreen', 'steelblue', 'gold'],
    ['firebrick', 'rebeccapurple', 'teal', 'magenta'],
    ['slateblue', 'orange', 'forestgreen', 'orchid']
]

for idx, col in enumerate(categorical_columns):
    row = idx // 2
    col_pos = idx % 2

    count_vals = df_categorical_1[col].value_counts()
    ordered_categories = count_vals.index.tolist()

    sns.countplot(
        x=col,
        data=df_categorical_1,
        ax=axs[row, col_pos],
        order=ordered_categories,
        color=None
    )
    colors = colors_list[idx] if idx < len(colors_list) else None
    for i, bar in enumerate(axs[row, col_pos].patches):
        if colors and i < len(colors):
            bar.set_color(colors[i])

    axs[row, col_pos].set_title(f'Countplot de {col}')
    axs[row, col_pos].set_xlabel(col)
    axs[row, col_pos].set_ylabel('Frequência')

plt.tight_layout()
plt.show()
```



Bloco 2 - (5 a 8)

```
In [68]: categorical_columns = df_categorical_2.columns.tolist()

fig, axs = plt.subplots(len(categorical_columns), 1, figsize=(12, 24))
colors_list = [
    ['royalblue', 'orange', 'seagreen', 'crimson'],
    ['red', 'mediumseagreen', 'steelblue', 'gold'],
    ['firebrick', 'rebeccapurple', 'teal', 'magenta'],
    ['slateblue', 'orange', 'forestgreen', 'orchid'],
]

if len(categorical_columns) == 1:
    axs = [axs]

for idx, col in enumerate(categorical_columns):
    ax = axs[idx]

    freq_order = df_categorical_2[col].value_counts().index.tolist()

    sns.countplot(
        x=col,
        data=df_categorical_2,
        ax=ax,
        order=freq_order,
        color=None
    )
    colors = colors_list[idx] if idx < len(colors_list) else None
    for i, bar in enumerate(ax.patches):
        if colors and i < len(colors):
            bar.set_color(colors[i])

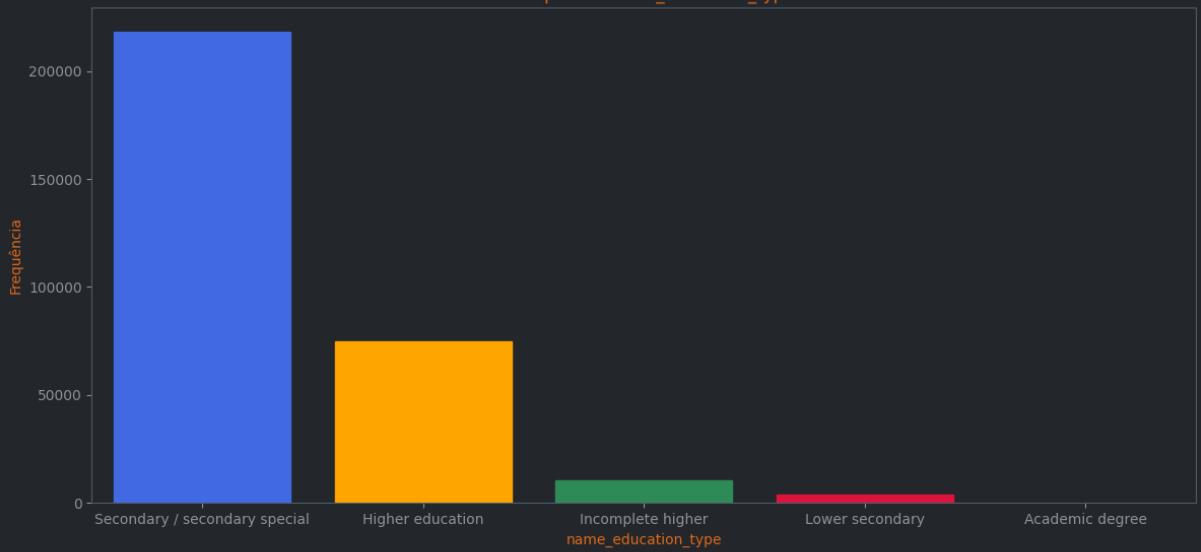
    ax.set_title(f'Countplot de {col}')
    ax.set_xlabel(col)
    ax.set_ylabel('Frequência')
```

```
if col.upper() == "OCCUPATION_TYPE":  
    ax.set_xticklabels(ax.get_xticklabels(), rotation=45, ha='right', fontsize=11)  
    ax.margins(y=0.18)  
  
plt.tight_layout()  
plt.show()
```

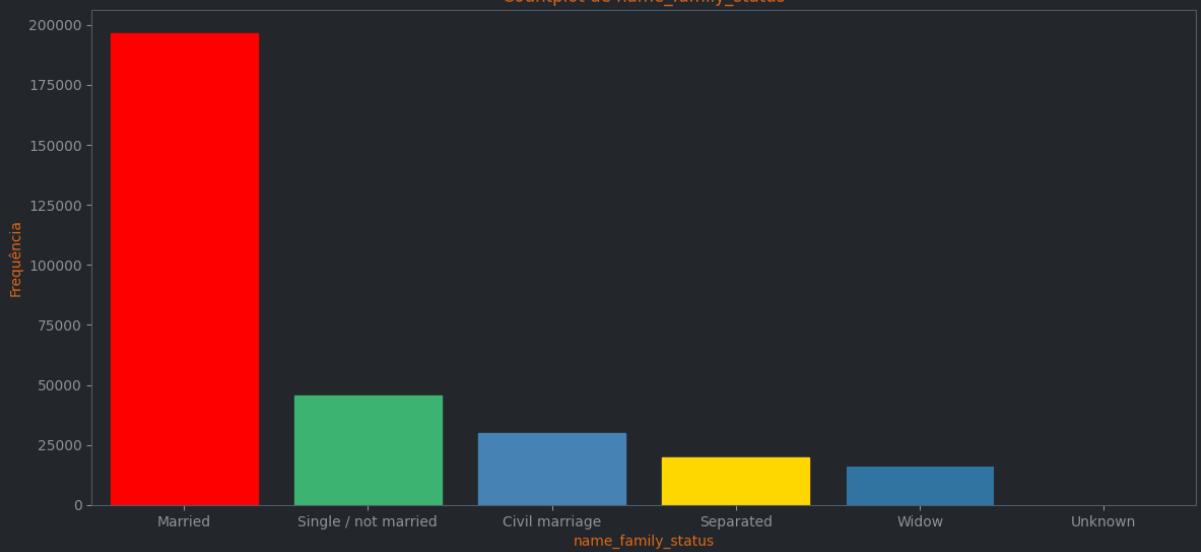
C:\Users\Patryck\AppData\Local\Temp\ipykernel_12620\382229923.py:37: UserWarning: set_ticklabel
s() should only be used with a fixed number of ticks, i.e. after set_ticks() or using a FixedLo
cator.

```
    ax.set_xticklabels(ax.get_xticklabels(), rotation=45, ha='right', fontsize=11)
```

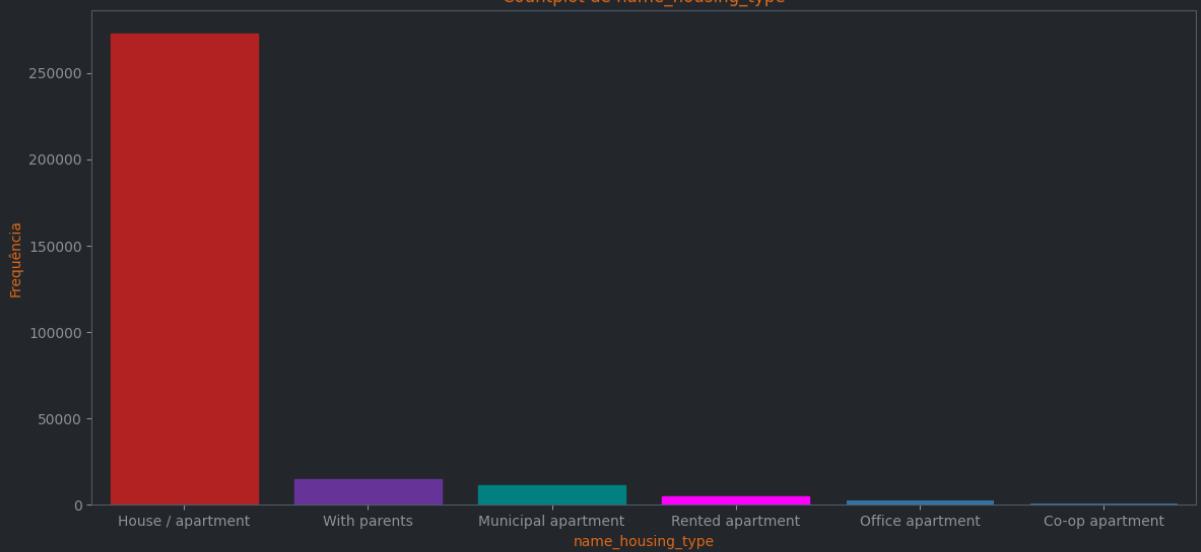
Countplot de name_education_type



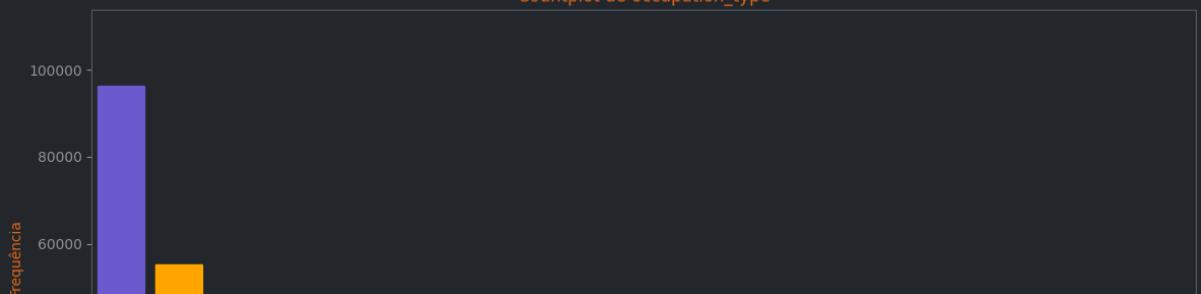
Countplot de name_family_status

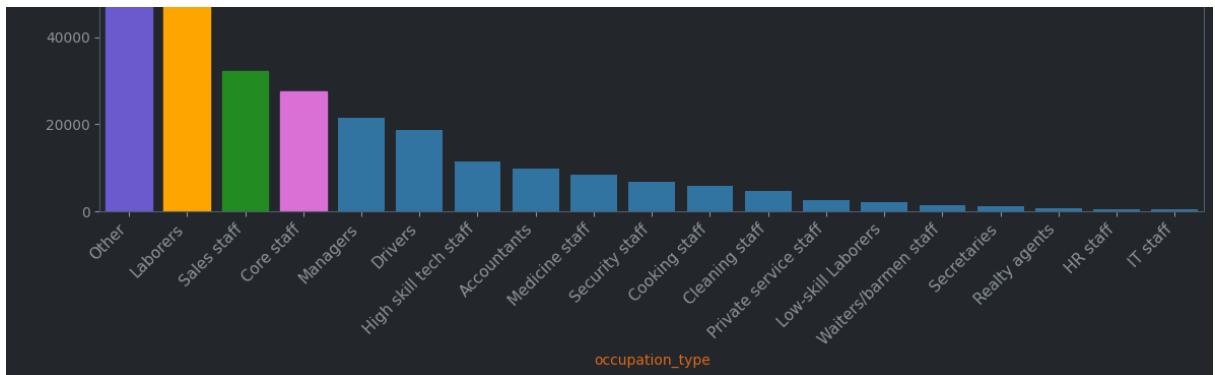


Countplot de name_housing_type



Countplot de occupation_type





Bloco 3 - (8 a 13)

```
In [69]: categorical_columns = df_categorical_3.columns.tolist()

fig, axs = plt.subplots(5, 1, figsize=(12, 28))

colors_list = [
    ['royalblue', 'orange', 'seagreen', 'crimson'],
    ['red', 'mediumseagreen', 'steelblue', 'gold'],
    ['firebrick', 'rebeccapurple', 'teal', 'magenta'],
    ['slateblue', 'orange', 'forestgreen', 'orchid'],
    ['goldenrod', 'orchid', 'dodgerblue', 'firebrick'],
]

for idx, col in enumerate(categorical_columns):
    ax = axs[idx]

    order = df_categorical_3[col].value_counts().index.tolist()

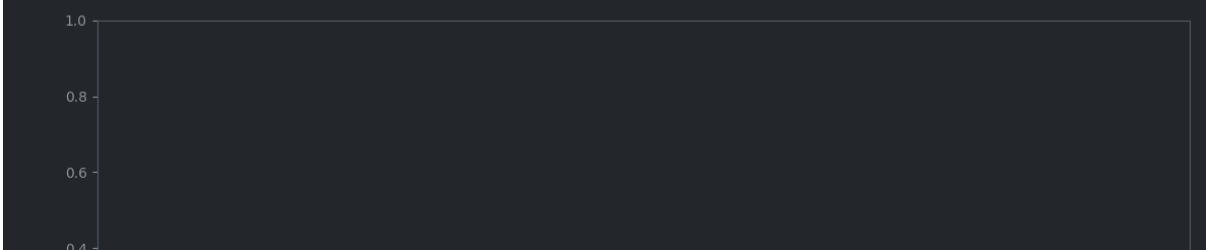
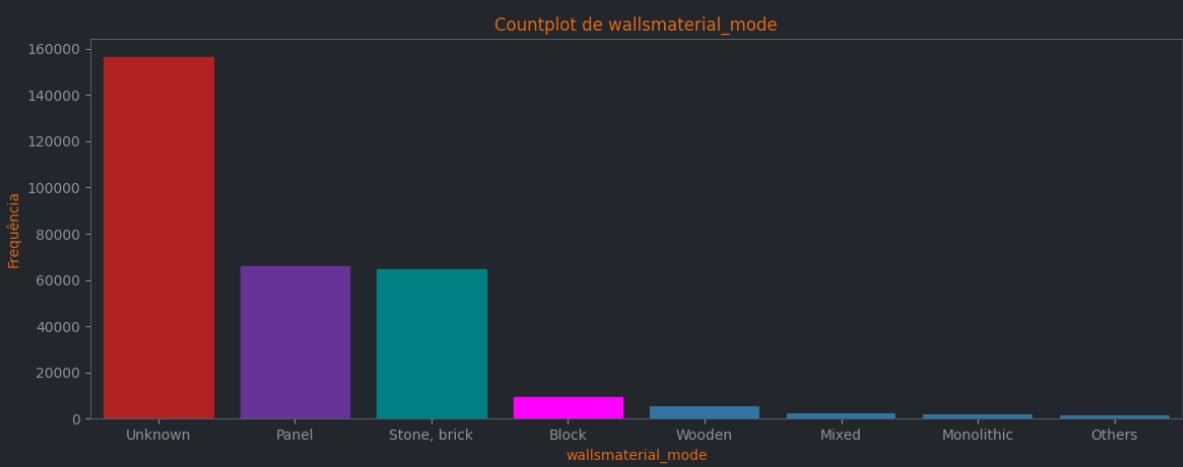
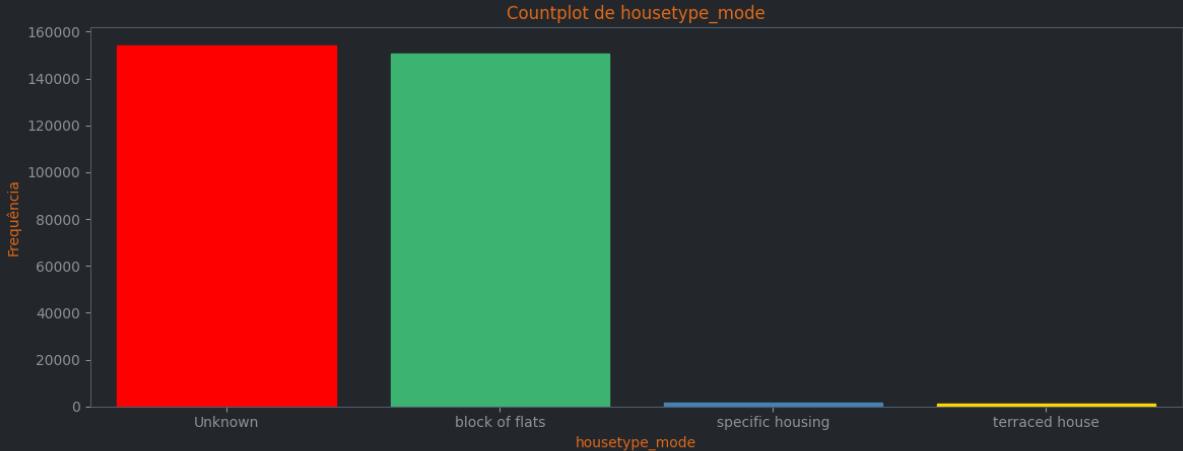
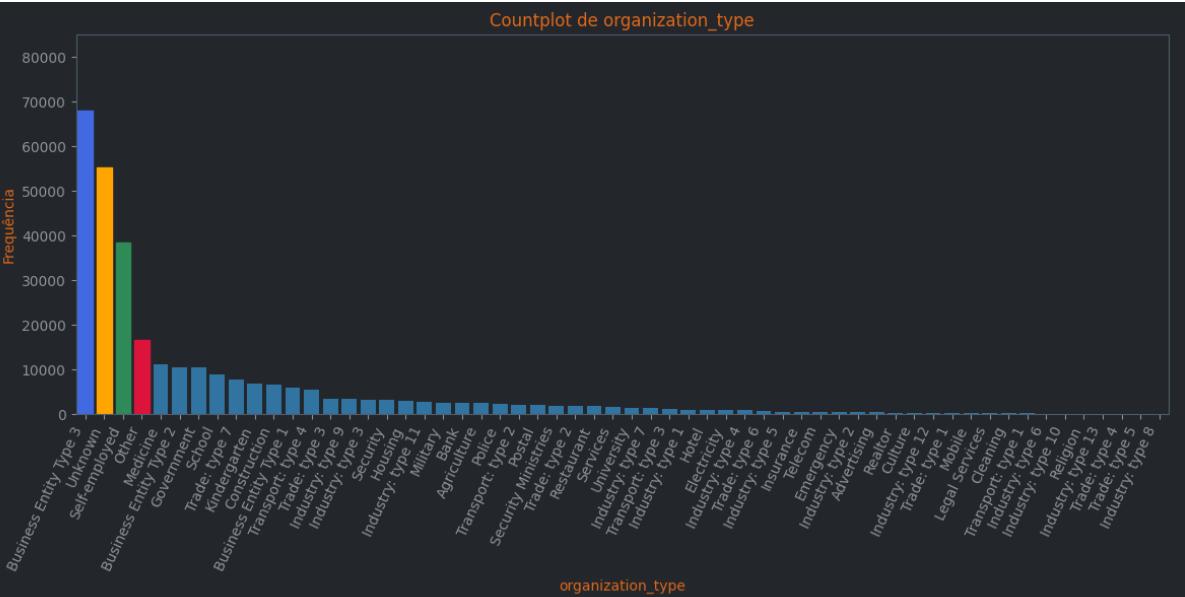
    sns.countplot(
        x=col,
        data=df_categorical_3,
        ax=ax,
        order=order,
        color=None
    )
    colors = colors_list[idx] if idx < len(colors_list) else None
    for i, bar in enumerate(ax.patches):
        if colors and i < len(colors):
            bar.set_color(colors[i])

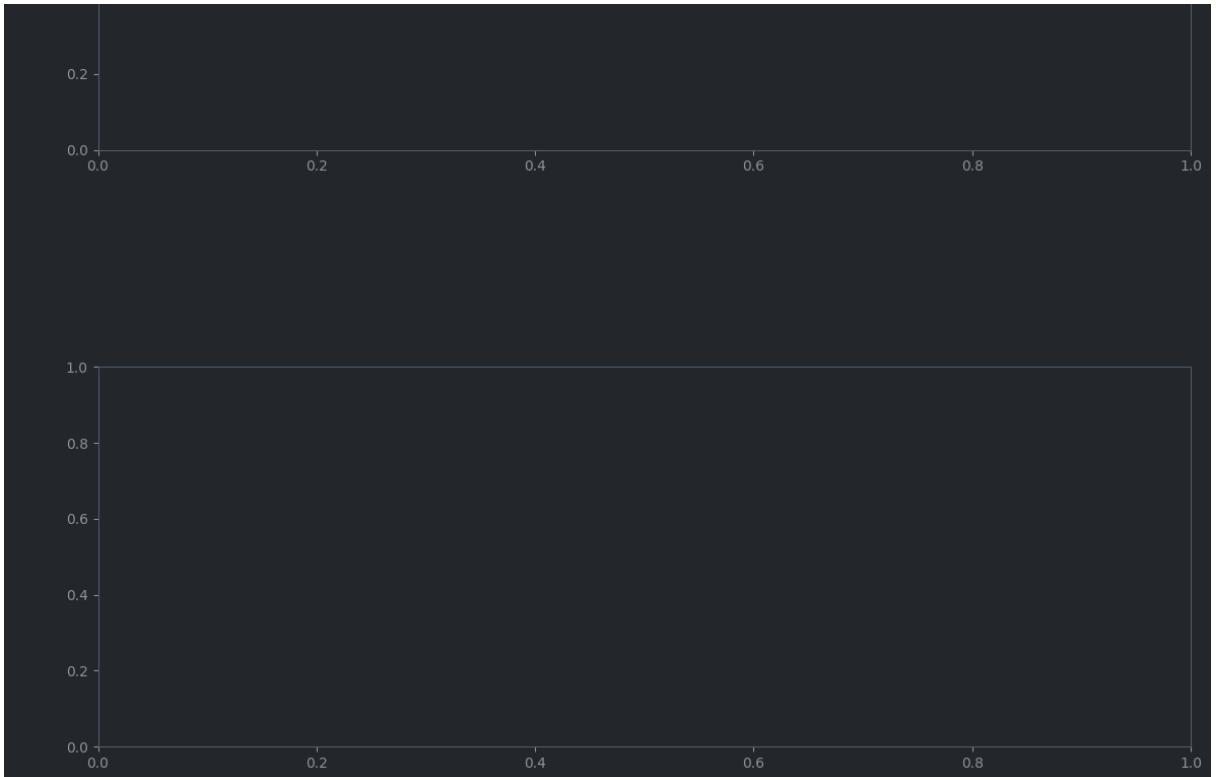
    ax.set_title(f'Countplot de {col}')
    ax.set_xlabel(col)
    ax.set_ylabel('Frequência')

    if col.upper() == "ORGANIZATION_TYPE":
        ax.set_xticklabels(
            ax.get_xticklabels(),
            rotation=65,
            ha='right',
            fontsize=10
        )
    ax.margins(y=0.25)
    plt.subplots_adjust(bottom=0.38, top=0.96)

plt.tight_layout()
plt.show()
```

C:\Users\Patryck\AppData\Local\Temp\ipykernel_12620\372829898.py:35: UserWarning: set_ticklabel
s() should only be used with a fixed number of ticks, i.e. after set_ticks() or using a FixedLo
cator.
ax.set_xticklabels(



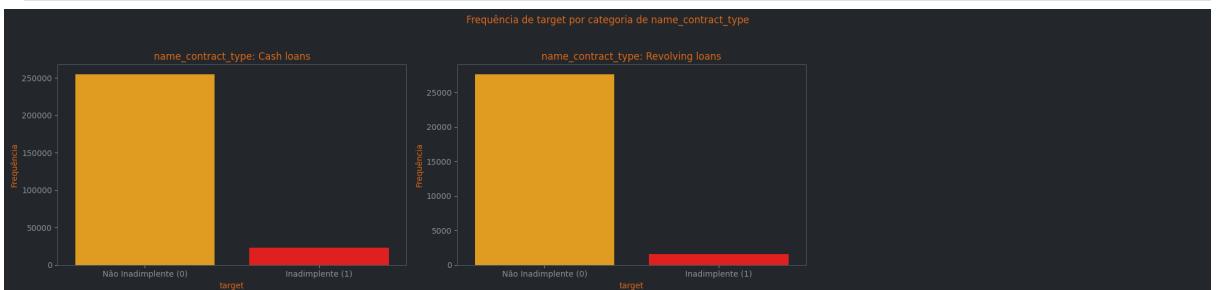


7.2 Análise Bivariada

Bloco 1 de Hipóteses

H1 — O tipo de contrato influencia a probabilidade de inadimplência.

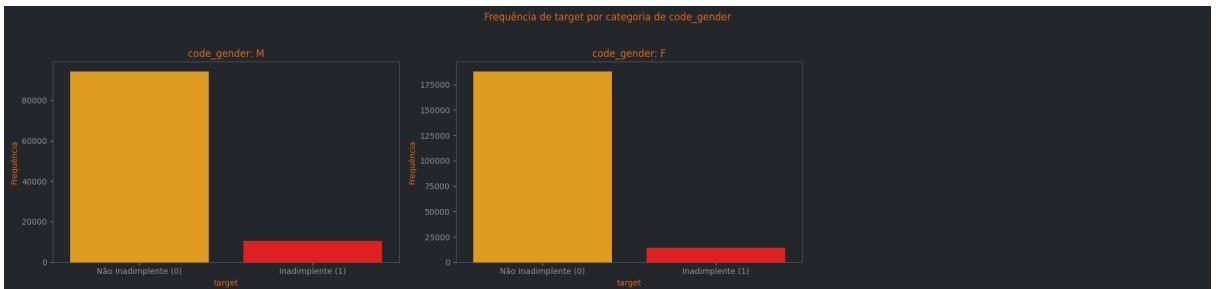
```
In [70]: plot_target_by_categorical(df3, cat_col='name_contract_type', target_col='target')
calcular_cramers_v(df3, 'name_contract_type', 'target')
```



V de Cramer entre name_contract_type e target: 0.0309

H2 — O gênero está associado ao risco de inadimplência.

```
In [71]: plot_target_by_categorical(df3, cat_col='code_gender', target_col='target')
calcular_cramers_v(df3, 'code_gender', 'target')
```

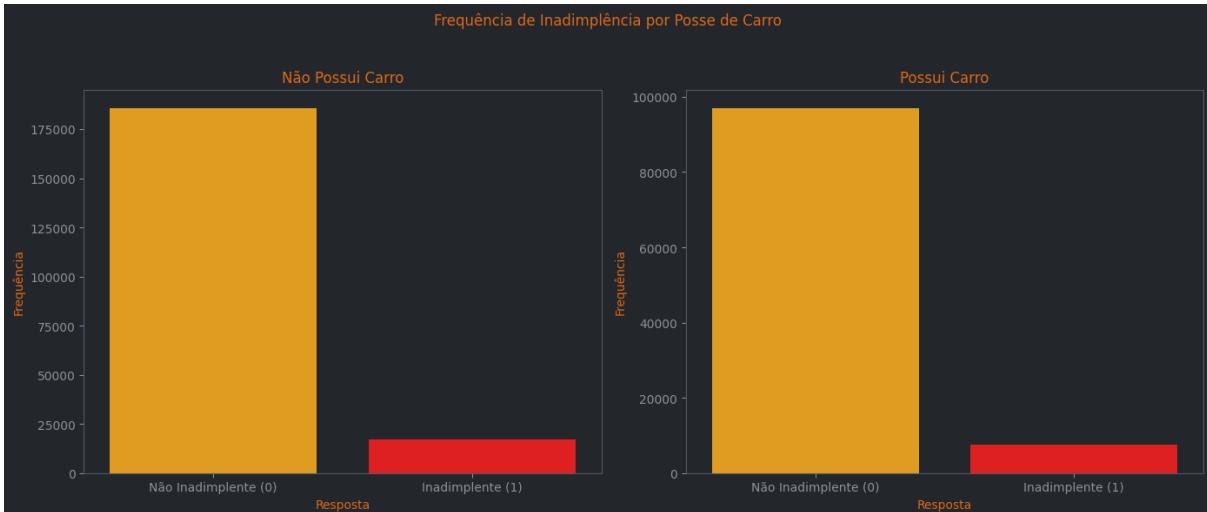


V de Cramer entre code_gender e target: 0.0547

H3 — Clientes que possuem carro têm menor probabilidade de inadimplência.

```
In [72]: plot_binaria_target(
    df3,
    var_binaria='flag_own_car',
    label_0='Não Possui Carro',
    label_1='Possui Carro',
    suptitle='Frequência de Inadimplência por Posse de Carro'
)

calcular_cramers_v(df3, 'flag_own_car', 'target')
```

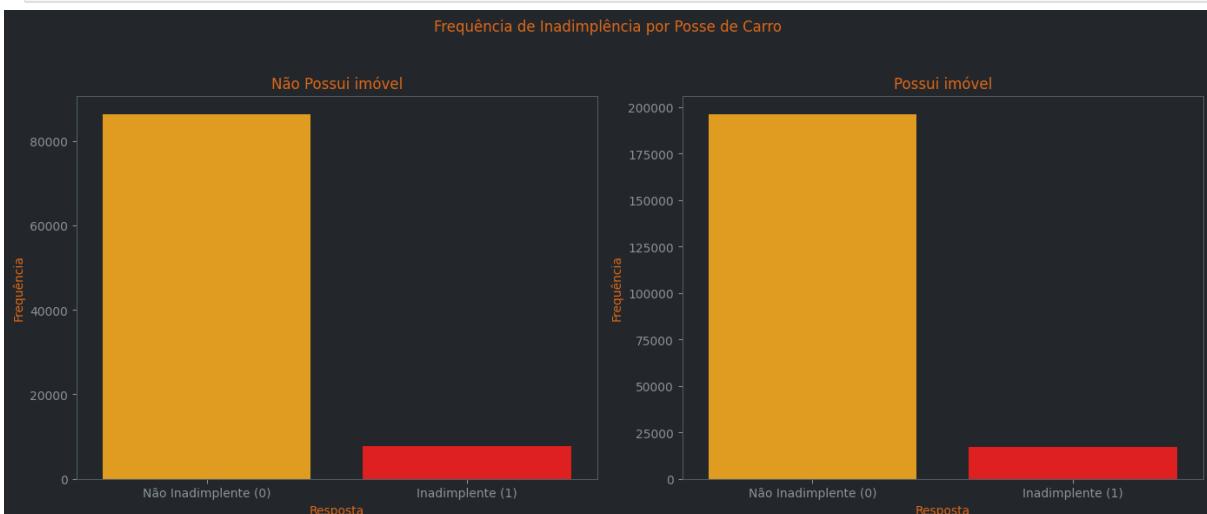


V de Cramer entre flag_own_car e target: 0.0218

H4 — Ter imóvel próprio reduz a chance de inadimplência.

```
In [73]: plot_binaria_target(
    df3,
    var_binaria='flag_own_realty',
    label_0='Não Possui imóvel',
    label_1='Possui imóvel',
    suptitle='Frequência de Inadimplência por Posse de Carro'
)

calcular_cramers_v(df3, 'flag_own_realty', 'target')
```



V de Cramer entre flag_own_realty e target: 0.0061

H5 — O número de filhos influencia o risco de inadimplência.

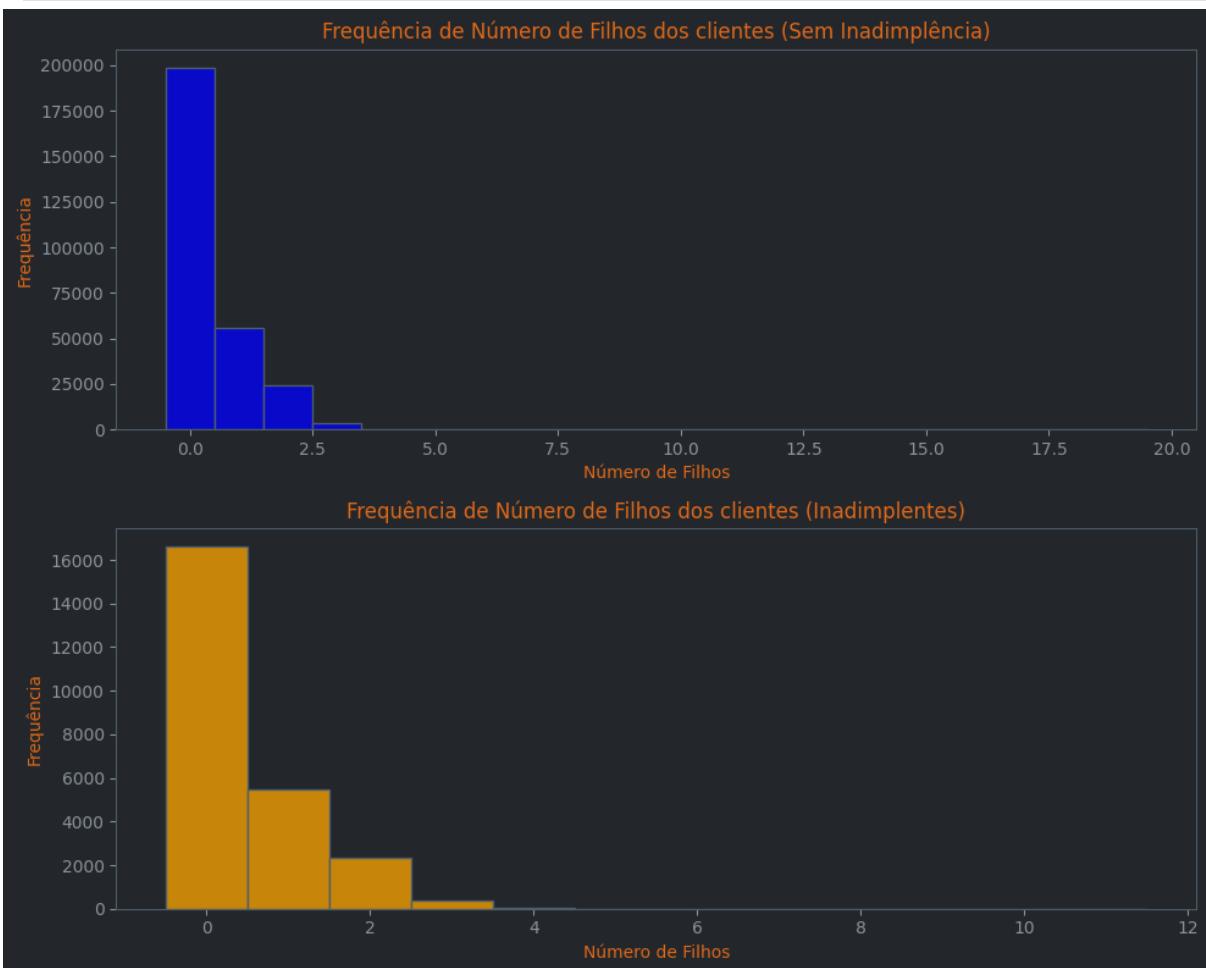
```
In [74]: plot_num_var_by_target(
    df3,
    num_var='cnt_children',
    title_0='Frequência de Número de Filhos dos clientes (Sem Inadimplência)',
    title_1='Frequência de Número de Filhos dos clientes (Inadimplentes)',
    label_x='Número de Filhos',
```

```

        discrete=True
    )

pearson_corr = df3['cnt_children'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre cnt_children e target: {pearson_corr:.4f}")

```



Correlação de Pearson entre cnt_children e target: 0.0192

H6 — Renda total mais alta está associada a menor inadimplência.

```

In [ ]: plot_num_var_by_target(
    df3,
    num_var='amt_income_total',
    title_0='Distribuição da Renda Total (amt_income_total) dos clientes (Sem Inadimplência)',
    title_1='Distribuição da Renda Total (amt_income_total) dos clientes (Inadimplentes)',
    label_x='Renda Total',
    discrete=True
)

pearson_corr = df3['amt_income_total'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_income_total e target: {pearson_corr:.4f}")

```

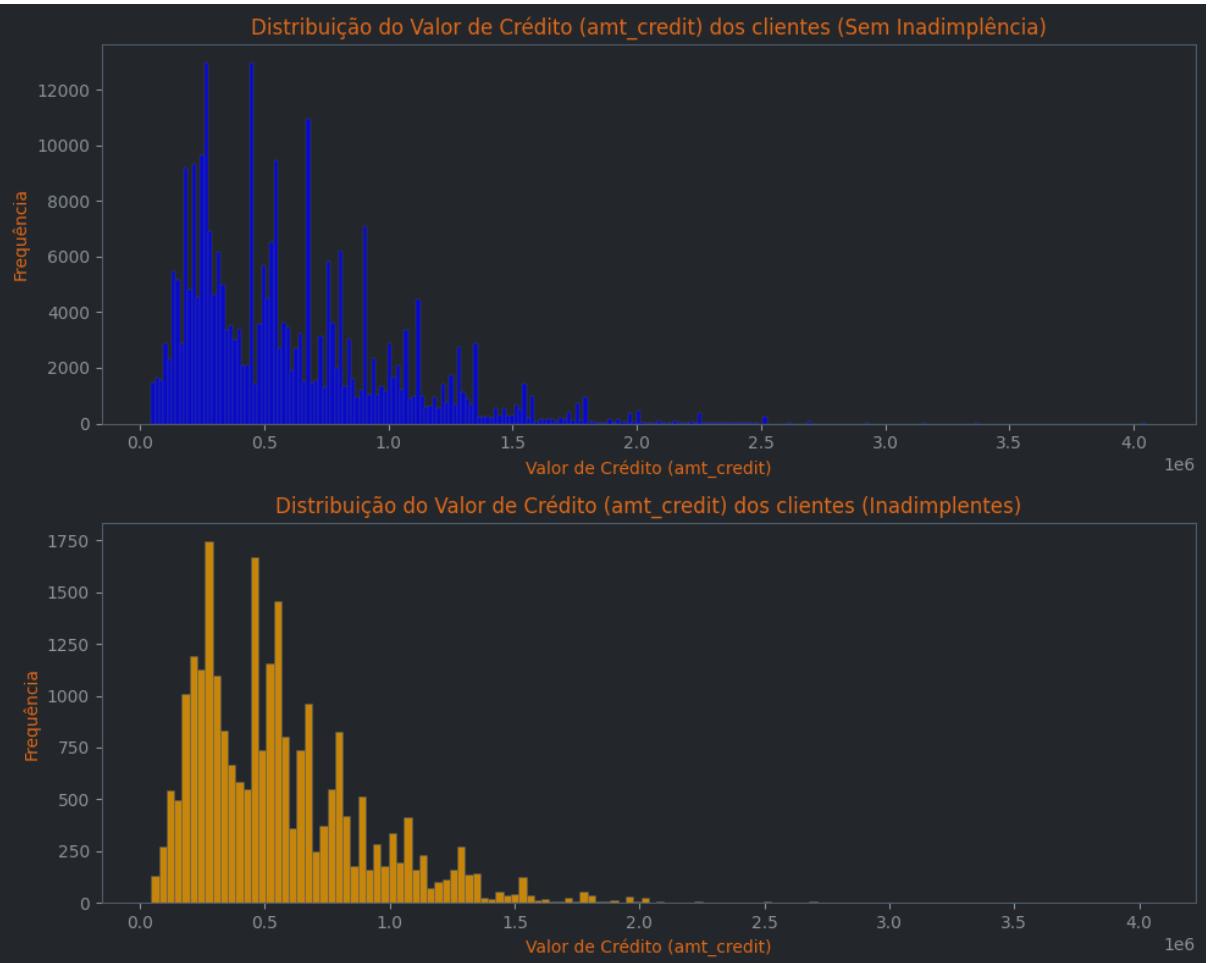
H7 — Valores de crédito mais altos estão relacionados a maior risco de inadimplência.

```

In [ ]: plot_num_var_by_target(
    df3,
    num_var='amt_credit',
    title_0='Distribuição do Valor de Crédito (amt_credit) dos clientes (Sem Inadimplência)',
    title_1='Distribuição do Valor de Crédito (amt_credit) dos clientes (Inadimplentes)',
    label_x='Valor de Crédito',
    discrete=True
)

pearson_corr = df3['amt_credit'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_credit e target: {pearson_corr:.4f}")

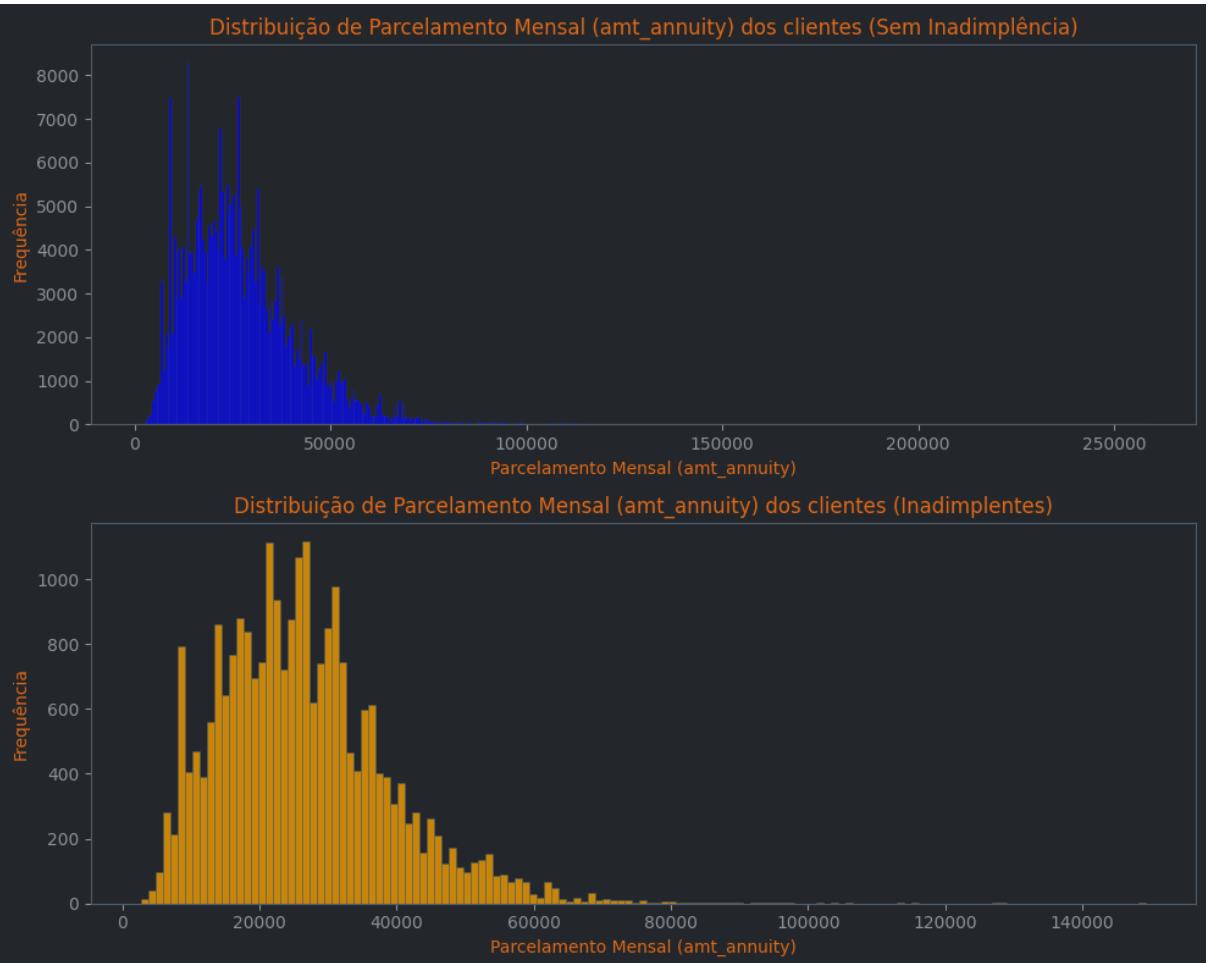
```



Correlação de Pearson entre amt_credit e target: -0.0304

H8 — Parcelas mensais maiores aumentam a chance de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='amt_annuity',
    title_0='Distribuição de Parcelamento Mensal (amt_annuity) dos clientes (Sem Inadimplência)',
    title_1='Distribuição de Parcelamento Mensal (amt_annuity) dos clientes (Inadimplentes)',
    label_x='Parcelamento Mensal',
    discrete=True
)
pearson_corr = df3['amt_annuity'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_annuity e target: {pearson_corr:.4f}")
```

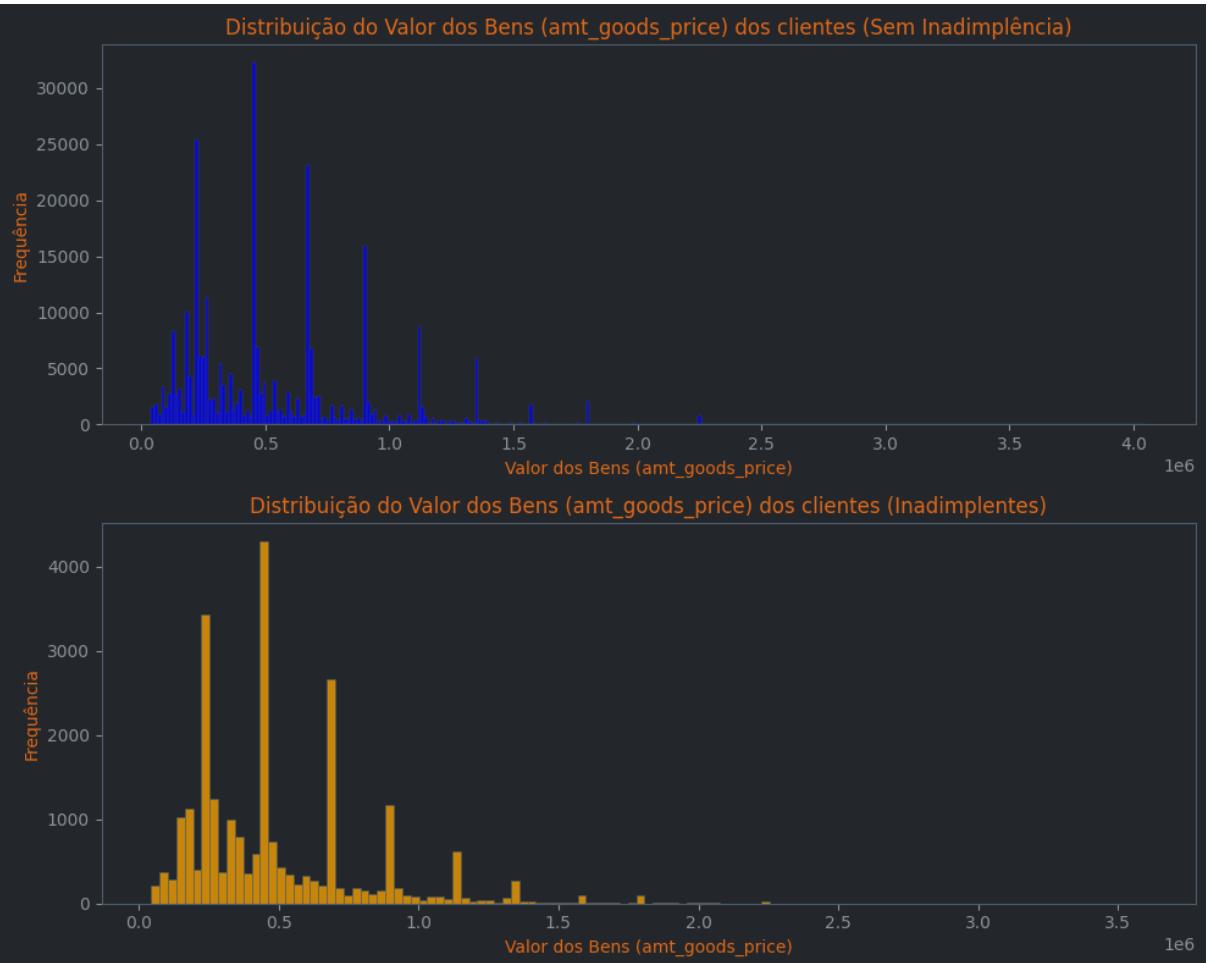


Correlação de Pearson entre amt_annuity e target: -0.0128

H9 — O valor dos bens adquiridos influencia o risco de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='amt_goods_price',
    title_0='Distribuição do Valor dos Bens (amt_goods_price) dos clientes (Sem Inadimplência)',
    title_1='Distribuição do Valor dos Bens (amt_goods_price) dos clientes (Inadimplentes)',
    label_x='Valor dos Bens',
    discrete=True
)

pearson_corr = df3['amt_goods_price'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_goods_price e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_goods_price e target: -0.0396

H10 — O tipo de companhia ao solicitar crédito está associado ao risco de inadimplência.

```
In [ ]: # Exemplo de uso:  
plot_target_by_categorical(df3, cat_col='name_type_suite', target_col='target')  
  
calcular_cramers_v(df3, 'name_type_suite', 'target')
```



V de Cramer entre name_type_suite e target: 0.0102

H11 — O tipo de fonte de renda impacta a inadimplência.

```
In [ ]: plot_target_by_categorical(df3, cat_col='name_income_type', target_col='target')

calcular_cramers_v(df3, 'name_income_type', 'target')
```



V de Cramer entre name_income_type e target: 0.0638

H12 — Níveis educacionais mais altos estão associados a menor risco de inadimplência.

```
In [ ]: plot_target_by_categorical(df3, cat_col='name_education_type', target_col='target')

calcular_cramers_v(df3, 'name_education_type', 'target')
```



V de Cramer entre name_education_type e target: 0.0576

H13 — O estado civil influencia o risco de inadimplência.

```
In [ ]: plot_target_by_categorical(df3, cat_col='name_family_status', target_col='target')

calcular_cramers_v(df3, 'name_family_status', 'target')
```



V de Cramer entre name_family_status e target: 0.0405

H14 — O tipo de moradia está associado à inadimplência.

```
In [ ]: plot_target_by_categorical(df3, cat_col='name_housing_type', target_col='target')

calcular_cramers_v(df3, 'name_housing_type', 'target')
```

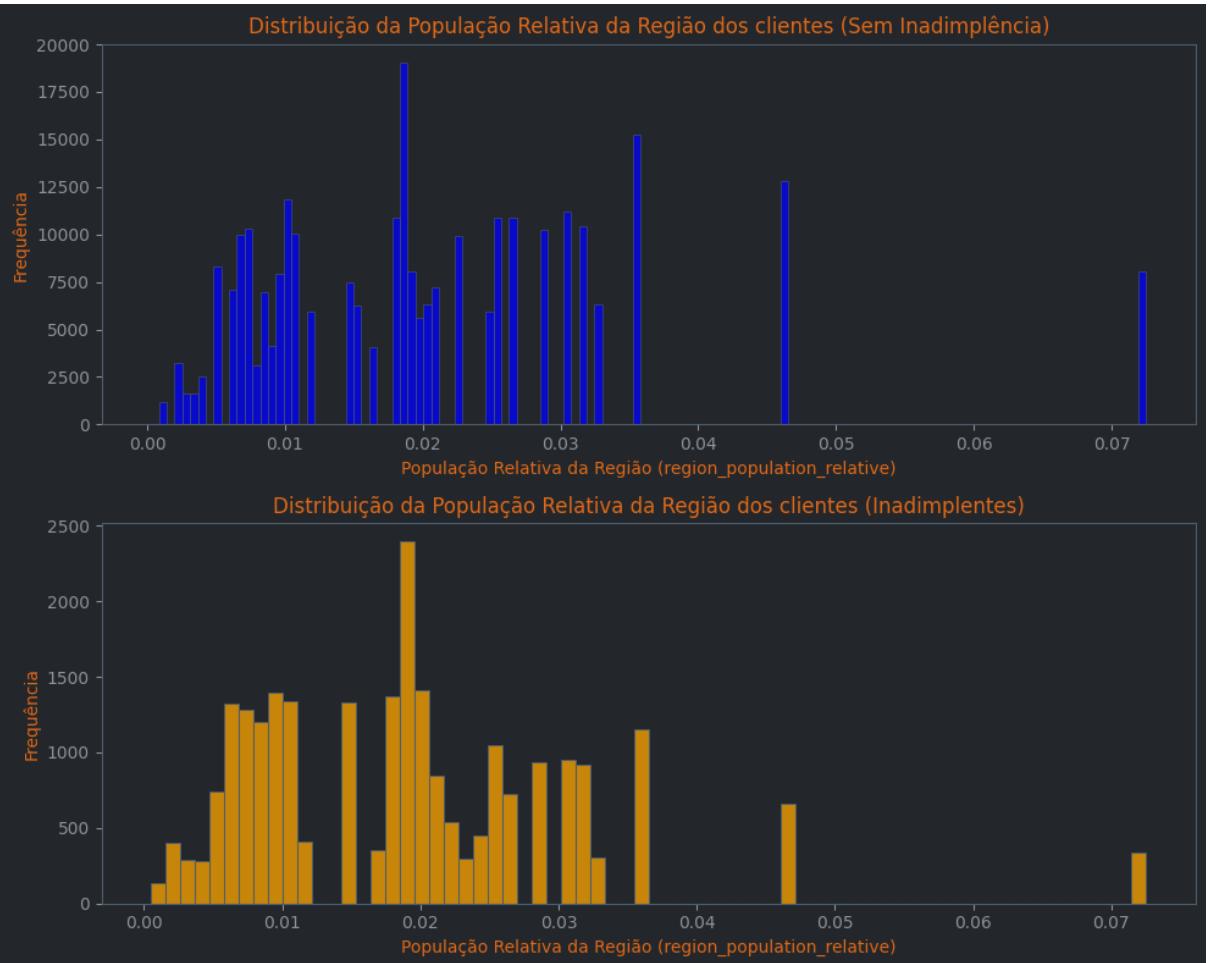


V de Cramer entre name_housing_type e target: 0.0370

H15 — Regiões mais populosas apresentam maior inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='region_population_relative',
    title_0='Distribuição da População Relativa da Região dos clientes (Sem Inadimplência)',
    title_1='Distribuição da População Relativa da Região dos clientes (Inadimplentes)',
    label_x='População Relativa da Região',
    discrete=True
)

pearson_corr = df3['region_population_relative'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre region_population_relative e target: {pearson_corr:.4f}")
```

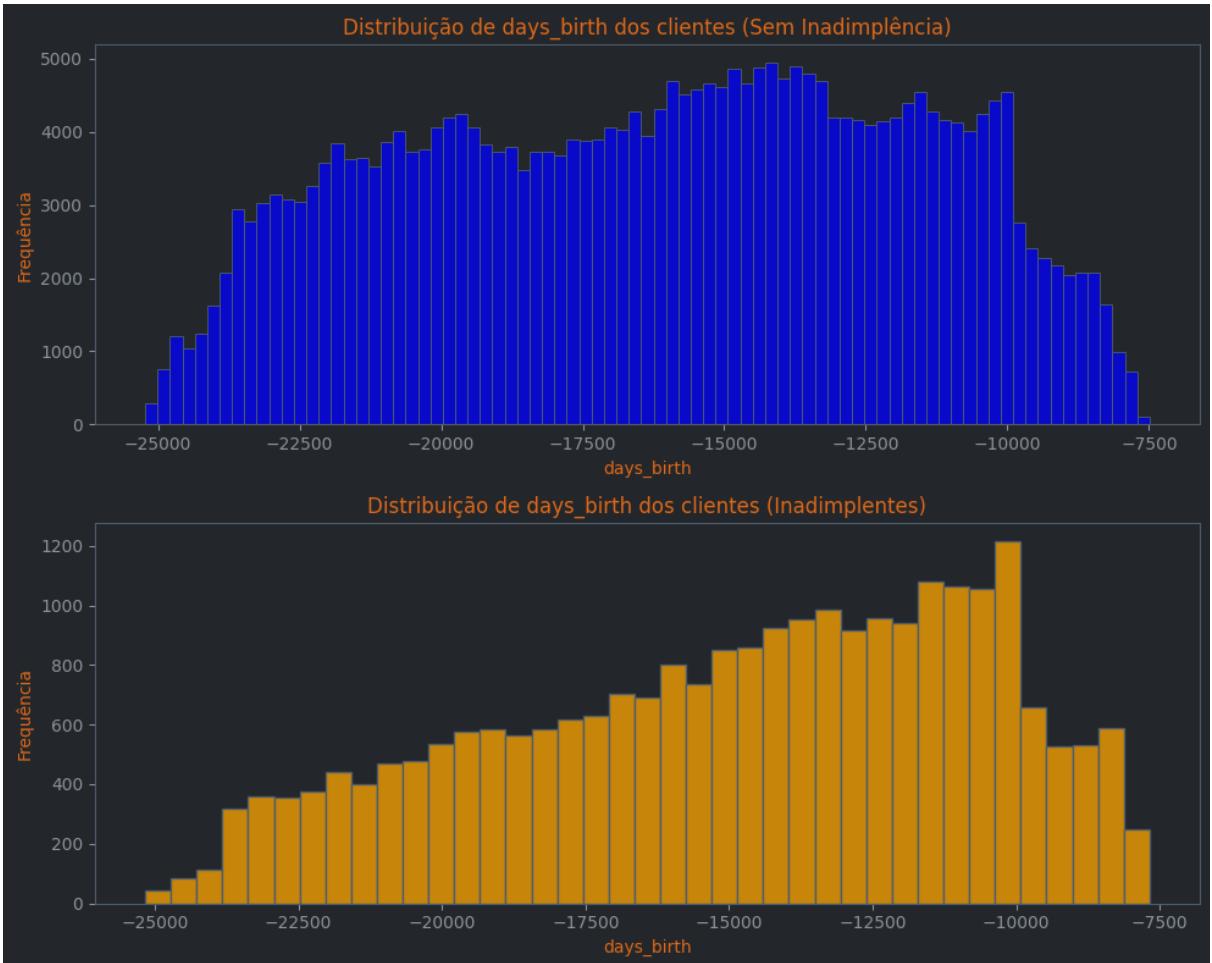


Correlação de Pearson entre region_population_relative e target: -0.0372

H16 — A idade do cliente influencia o risco de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='days_birth',
    title_0='Distribuição de days_birth dos clientes (Sem Inadimplência)',
    title_1='Distribuição de days_birth dos clientes (Inadimplentes)',
    label_x='days_birth',
    discrete=True
)

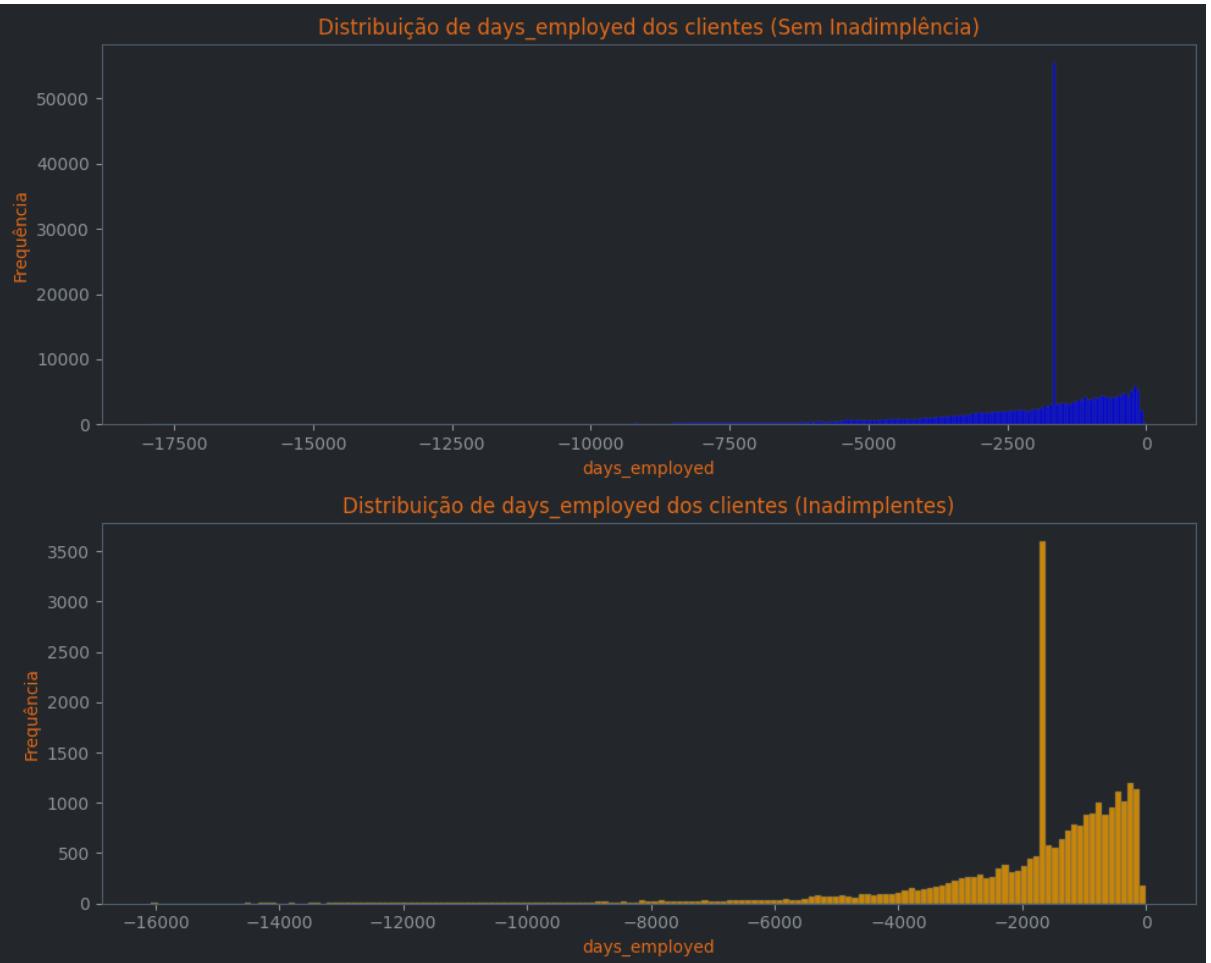
pearson_corr = df3['days_birth'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre days_birth e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre days_birth e target: 0.0782

H17 — O tempo de emprego está inversamente relacionado à inadimplência.

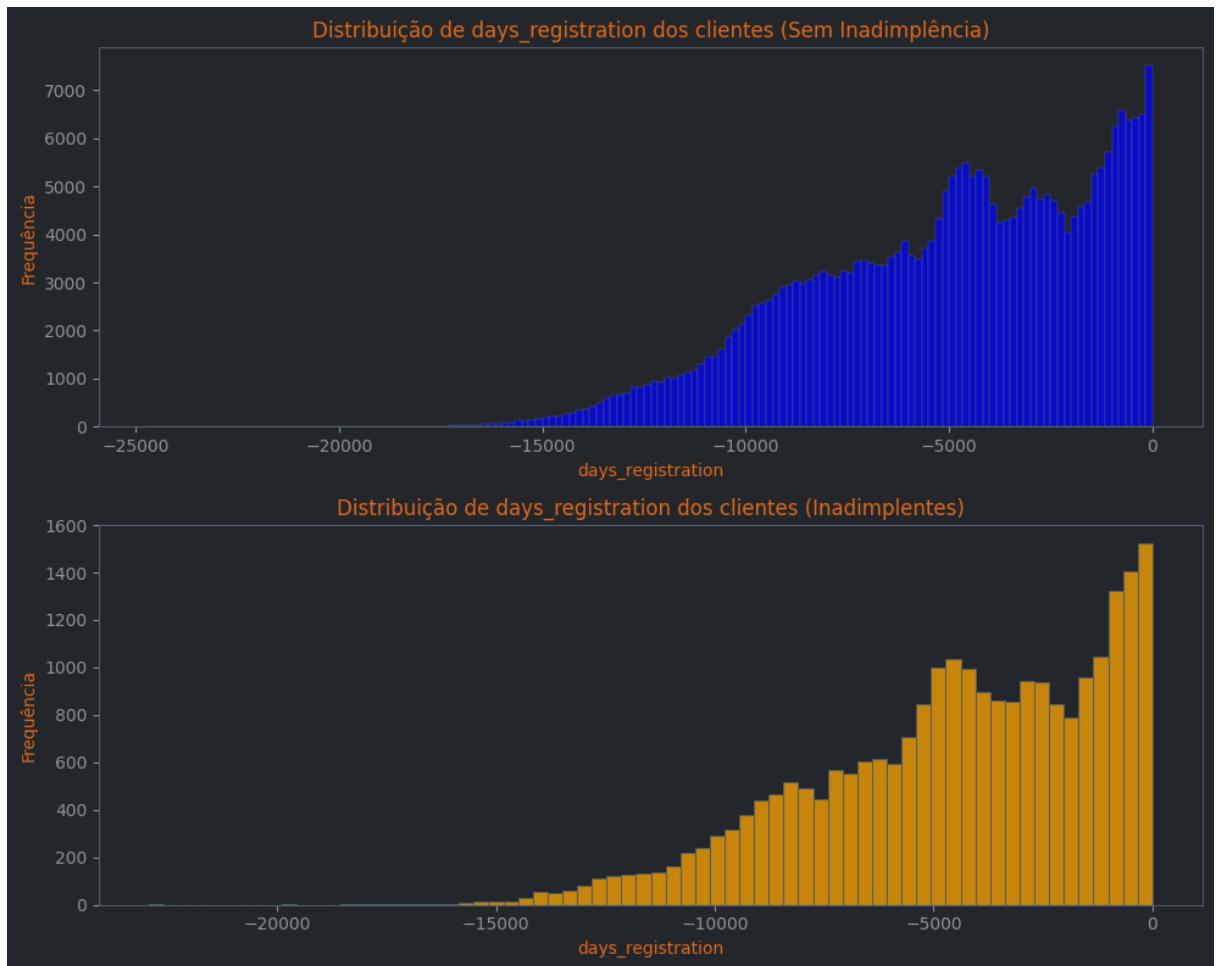
```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='days_employed',
    title_0='Distribuição de days_employed dos clientes (Sem Inadimplência)',
    title_1='Distribuição de days_employed dos clientes (Inadimplentes)',
    label_x='days_employed',
    discrete=True
)
pearson_corr = df3['days_employed'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre days_employed e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre days_employed e target: 0.0634

H18 — Clientes com registro mais antigo (mais tempo na base) apresentam menor inadimplência.

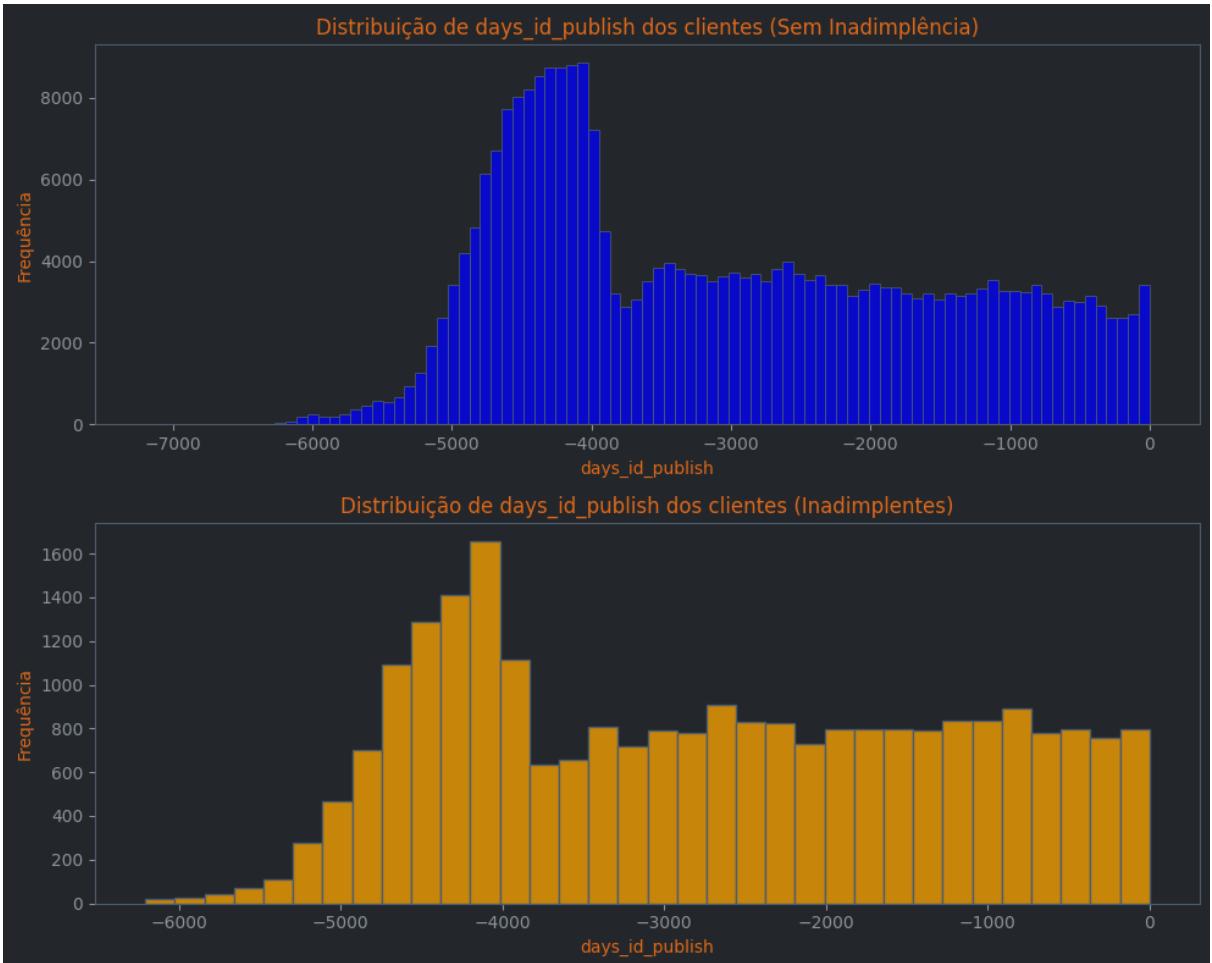
```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='days_registration',
    title_0='Distribuição de days_registration dos clientes (Sem Inadimplência)',
    title_1='Distribuição de days_registration dos clientes (Inadimplentes)',
    label_x='days_registration',
    discrete=True
)
pearson_corr = df3['days_registration'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre days_registration e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre days_registration e target: 0.0420

H19 — A data de emissão do documento influencia o risco de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='region_population_relative',
    title_0='Distribuição de days_id_publish dos clientes (Sem Inadimplência)',
    title_1='Distribuição de days_id_publish dos clientes (Inadimplentes)',
    label_x='days_id_publish',
    discrete=True
)
pearson_corr = df3['days_id_publish'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre days_id_publish e target: {pearson_corr:.4f}")
```

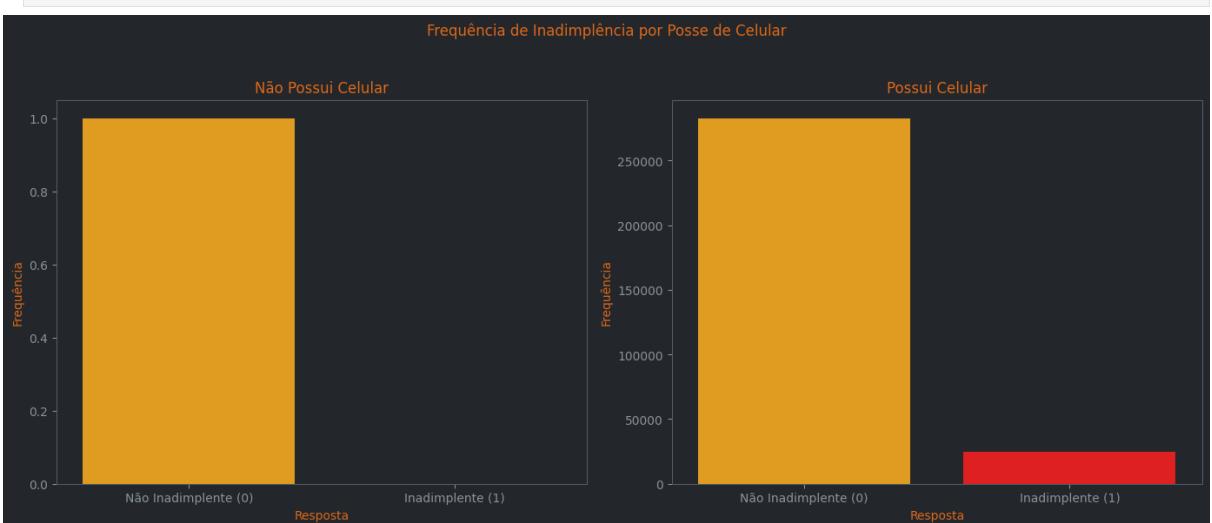


Correlação de Pearson entre days_id_publish e target: 0.0515

H20 — Possuir celular registrado está associado a menor inadimplência.

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_own_car',
    label_0='Não Possui Celular',
    label_1='Possui Celular',
    suptitle='Frequência de Inadimplência por Posse de Celular'
)

calcular_cramers_v(df3, 'flag_mobil', 'target')
```

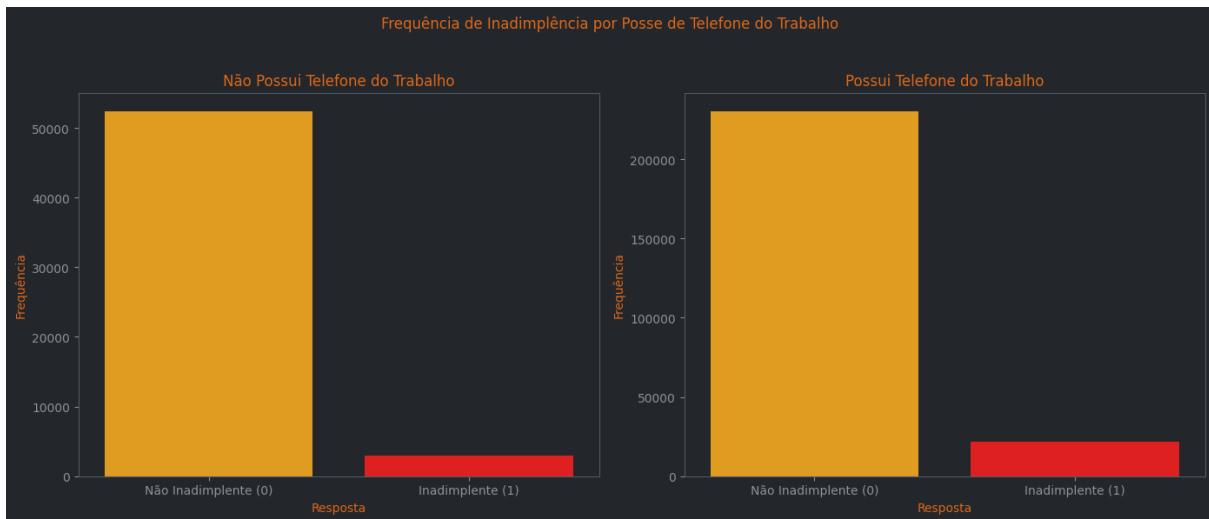


V de Cramer entre flag_mobil e target: 0.0000

Bloco 2 de Hipóteses

H21 — Ter telefone de trabalho cadastrado reduz o risco de inadimplência.

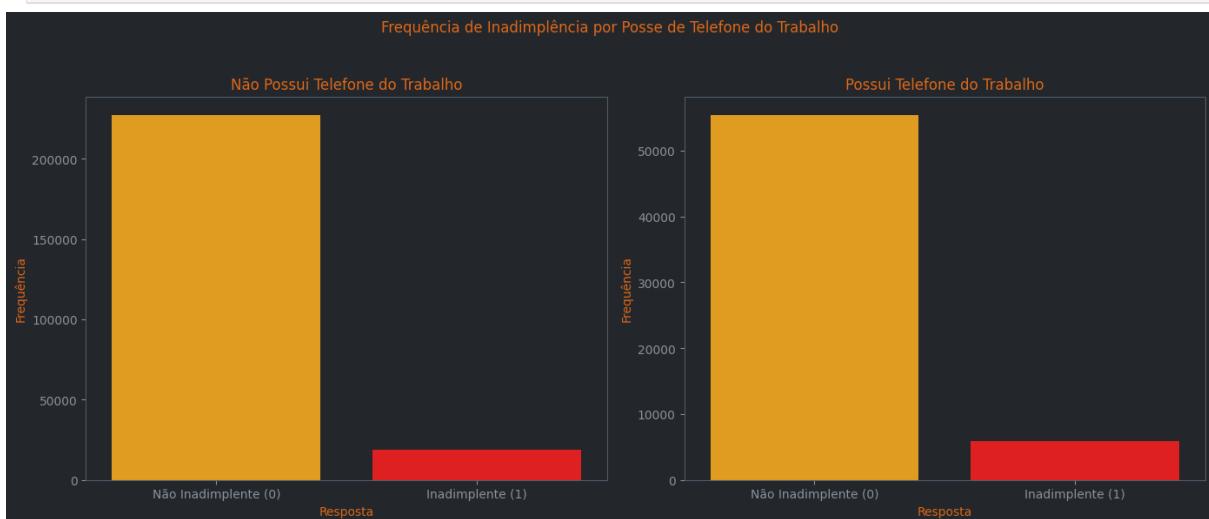
```
In [ ]: plot_binaria_target(  
    df3,  
    var_binaria='flag_emp_phone',  
    label_0='Não Possui Telefone do Trabalho',  
    label_1='Possui Telefone do Trabalho',  
    suptitle='Frequência de Inadimplência por Posse de Telefone do Trabalho'  
)  
  
calcular_cramers_v(df3, 'flag_emp_phone', 'target')
```



V de Cramer entre flag_emp_phone e target: 0.0460

H22 — Clientes com telefone profissional informado são menos propensos à inadimplência.

```
In [ ]: plot_binaria_target(  
    df3,  
    var_binaria='flag_work_phone',  
    label_0='Não Possui Telefone do Trabalho',  
    label_1='Possui Telefone do Trabalho',  
    suptitle='Frequência de Inadimplência por Posse de Telefone do Trabalho'  
)  
  
calcular_cramers_v(df3, 'flag_work_phone', 'target')
```



V de Cramer entre flag_work_phone e target: 0.0285

H23 — Possuir celular de contato registrado está associado a menor risco de inadimplência.

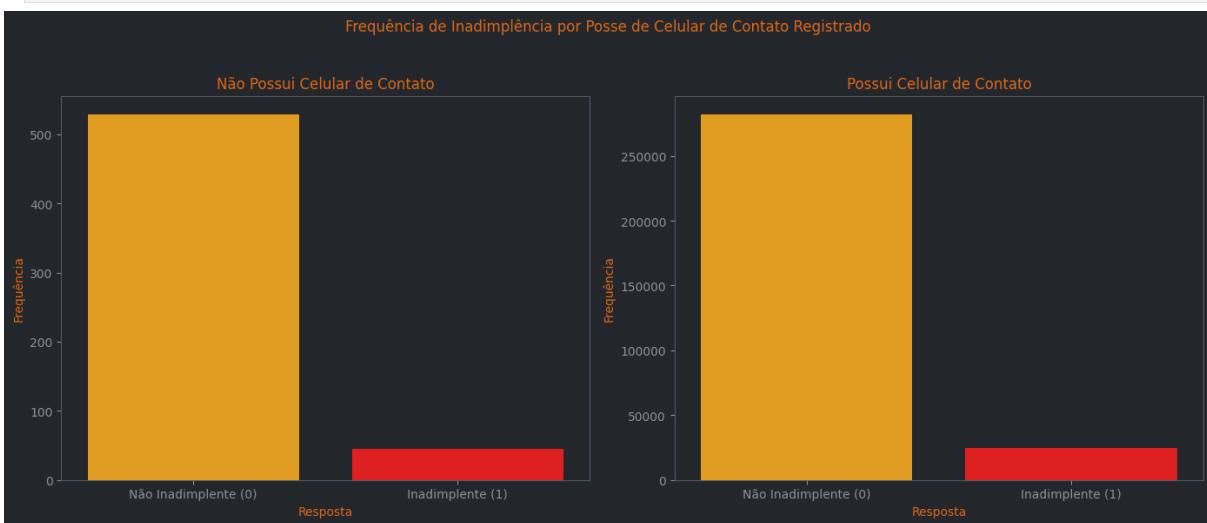
```
In [ ]: plot_binaria_target(  
    df3,  
    var_binaria='flag_cont_mobile',  
    label_0='Não Possui Celular de Contato',  
    label_1='Possui Celular de Contato',  
    suptitle='Frequência de Inadimplência por Posse de Celular de Contato')
```

```

        label_1='Possui Celular de Contato',
        suptitle='Frequência de Inadimplência por Posse de Celular de Contato Registrado'
    )

calcular_cramers_v(df3, 'flag_cont_mobile', 'target')

```



V de Cramer entre flag_cont_mobile e target: 0.0002

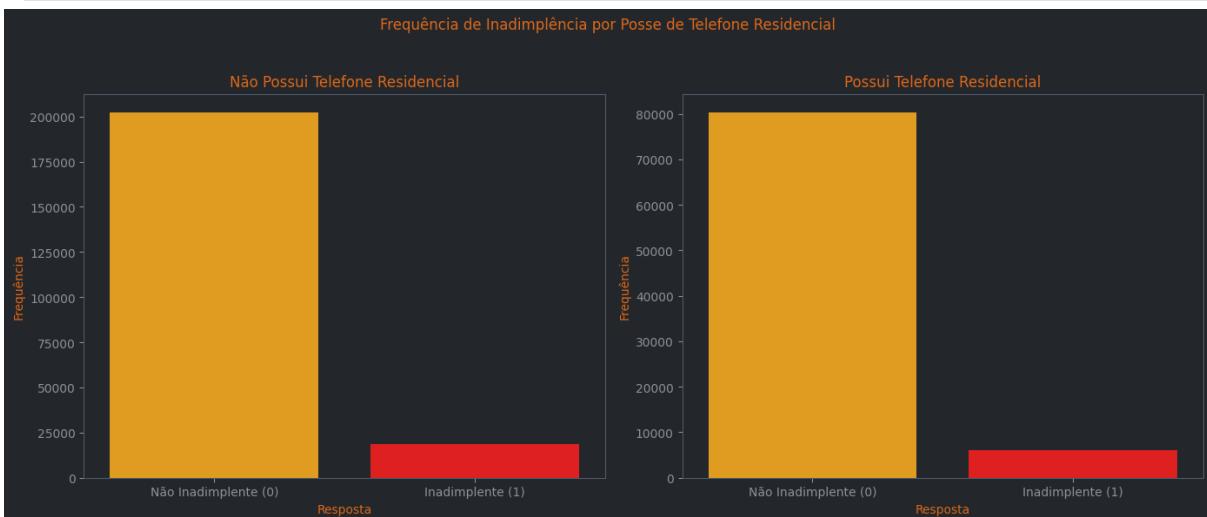
H24 — Ter telefone fixo registrado reduz o risco de inadimplência.

```

In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_phone',
    label_0='Não Possui Telefone Residencial',
    label_1='Possui Telefone Residencial',
    suptitle='Frequência de Inadimplência por Posse de Telefone Residencial'
)

calcular_cramers_v(df3, 'flag_phone', 'target')

```



V de Cramer entre flag_phone e target: 0.0238

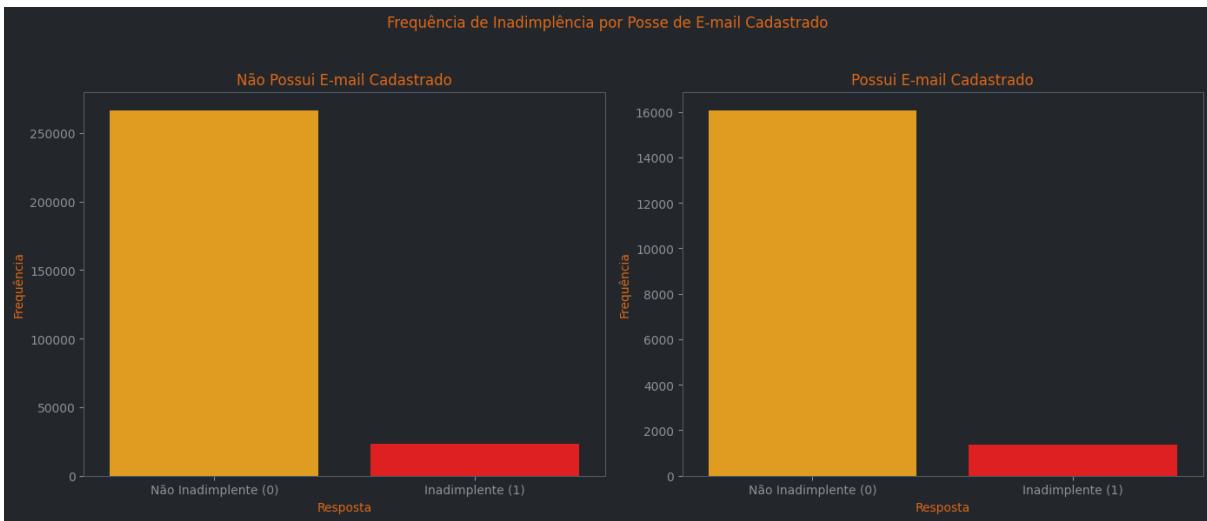
H25 — Clientes com e-mail cadastrado tendem a apresentar menor inadimplência.

```

In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_email',
    label_0='Não Possui E-mail Cadastrado',
    label_1='Possui E-mail Cadastrado',
    suptitle='Frequência de Inadimplência por Posse de E-mail Cadastrado'
)

```

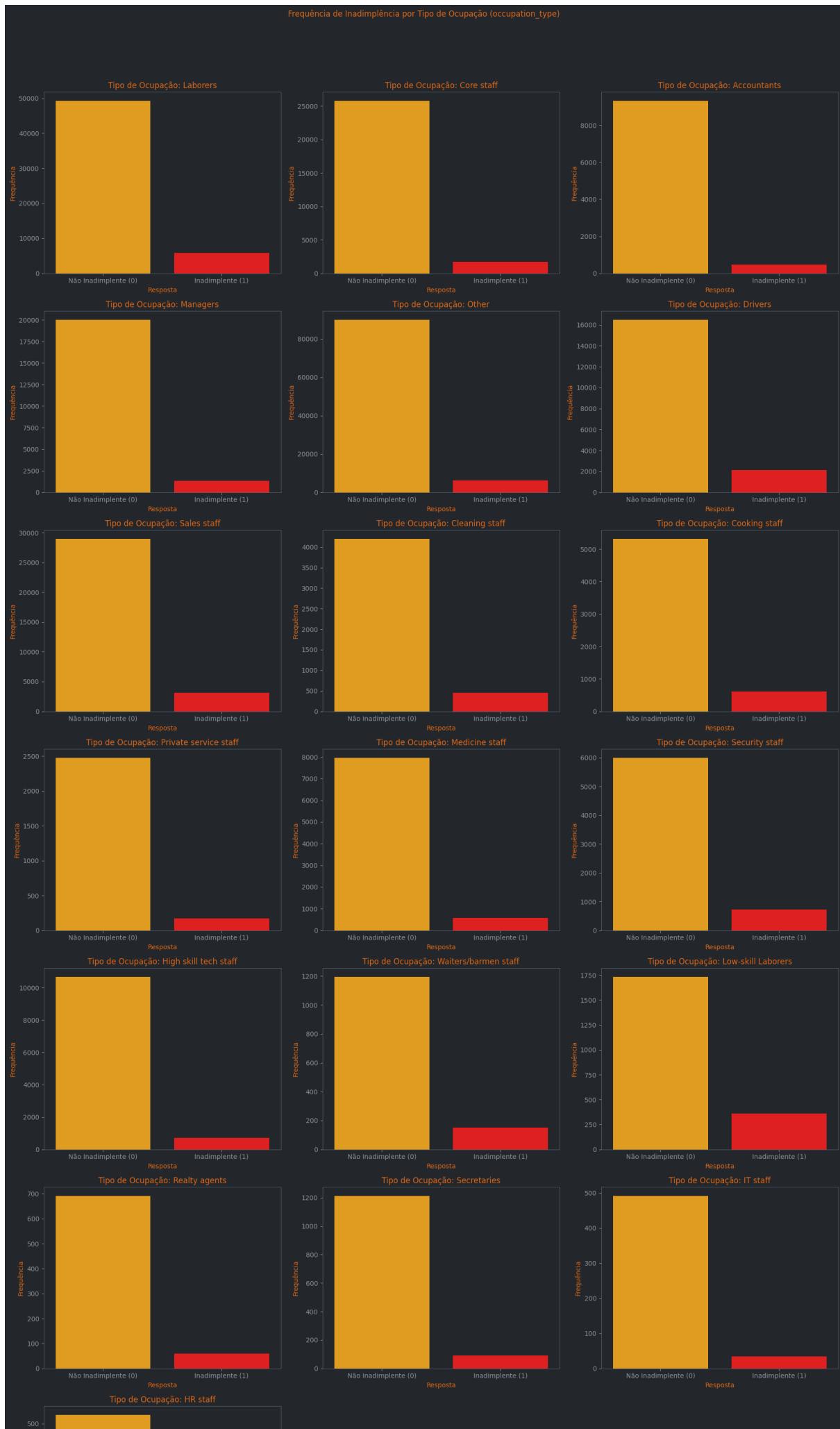
```
calcular_cramers_v(df3, 'flag_email', 'target')
```

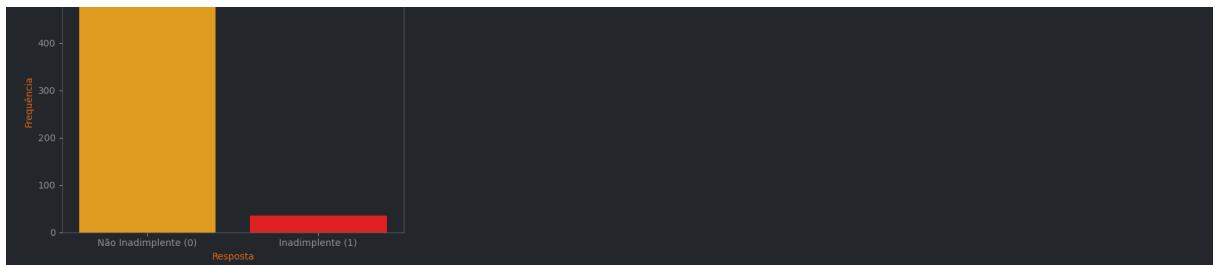


V de Cramer entre flag_email e target: 0.0017

H26 — O tipo de ocupação influencia o risco de inadimplência.

```
In [ ]: plot_target_by_categorical(df3, cat_col='occupation_type', target_col='target')  
calcular_cramers_v(df3, 'occupation_type', 'target')
```

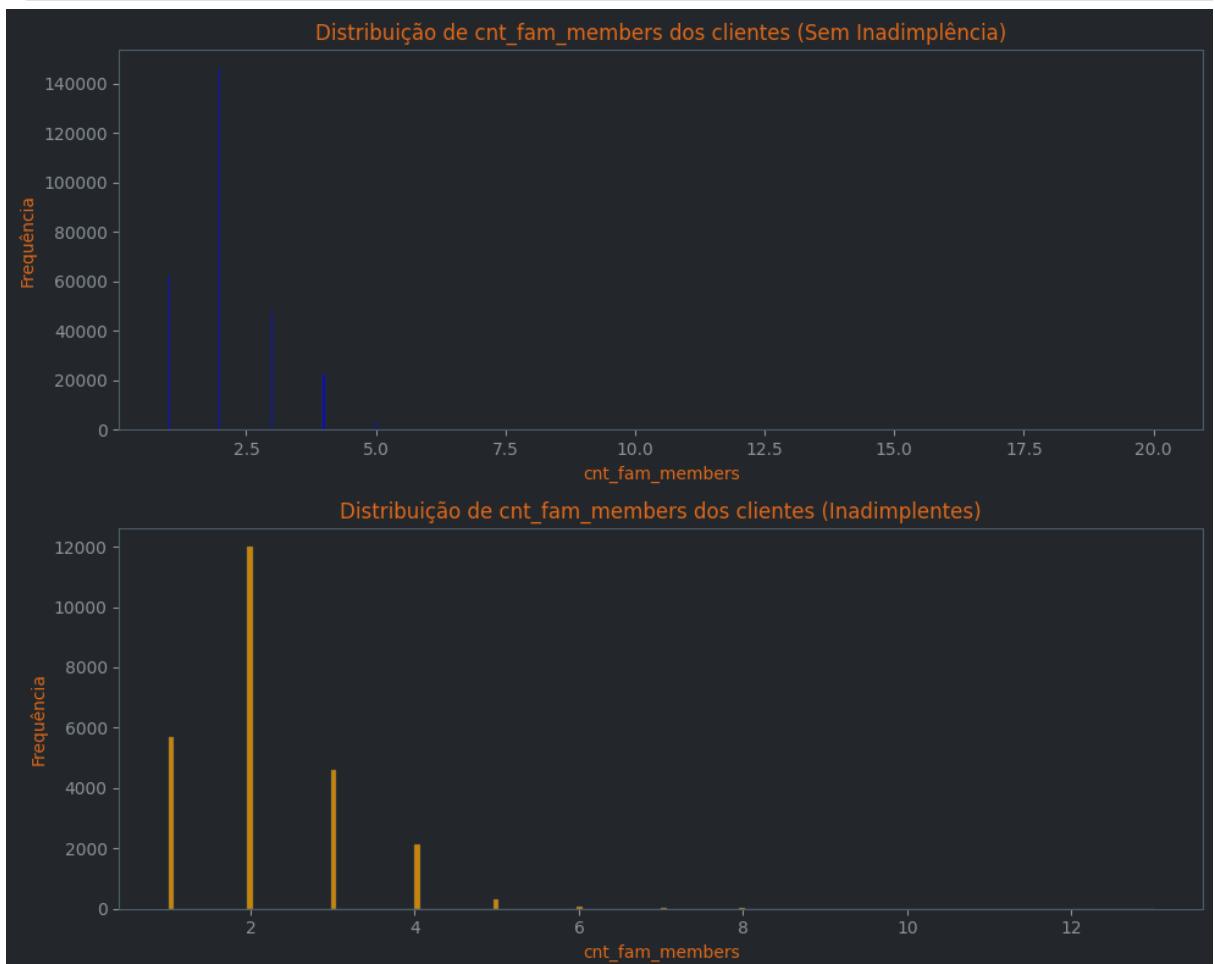




V de Cramer entre occupation_type e target: 0.0801

H27 — O número de membros na família está positivamente relacionado à inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='cnt_fam_members',
    title_0='Distribuição de cnt_fam_members dos clientes (Sem Inadimplência)',
    title_1='Distribuição de cnt_fam_members dos clientes (Inadimplentes)',
    label_x='cnt_fam_members',
    discrete=True
)
pearson_corr = df3['cnt_fam_members'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre cnt_fam_members e target: {pearson_corr:.4f}")
```

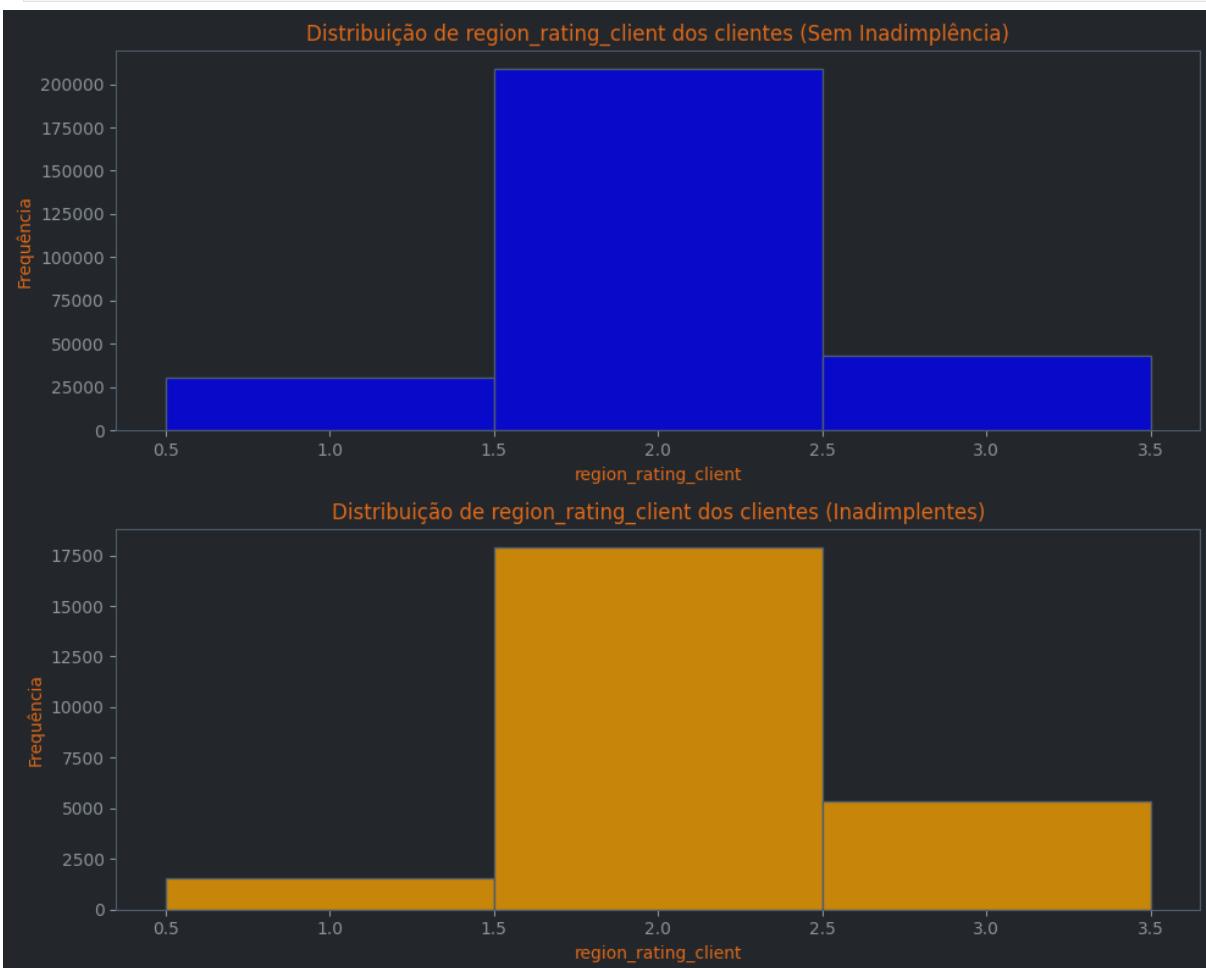


Correlação de Pearson entre cnt_fam_members e target: 0.0093

H28 — Clientes de regiões com menor rating apresentam maior probabilidade de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='region_rating_client',
    title_0='Distribuição de region_rating_client dos clientes (Sem Inadimplência)',
    title_1='Distribuição de region_rating_client dos clientes (Inadimplentes)',
    label_x='region_rating_client',
    discrete=True
)
```

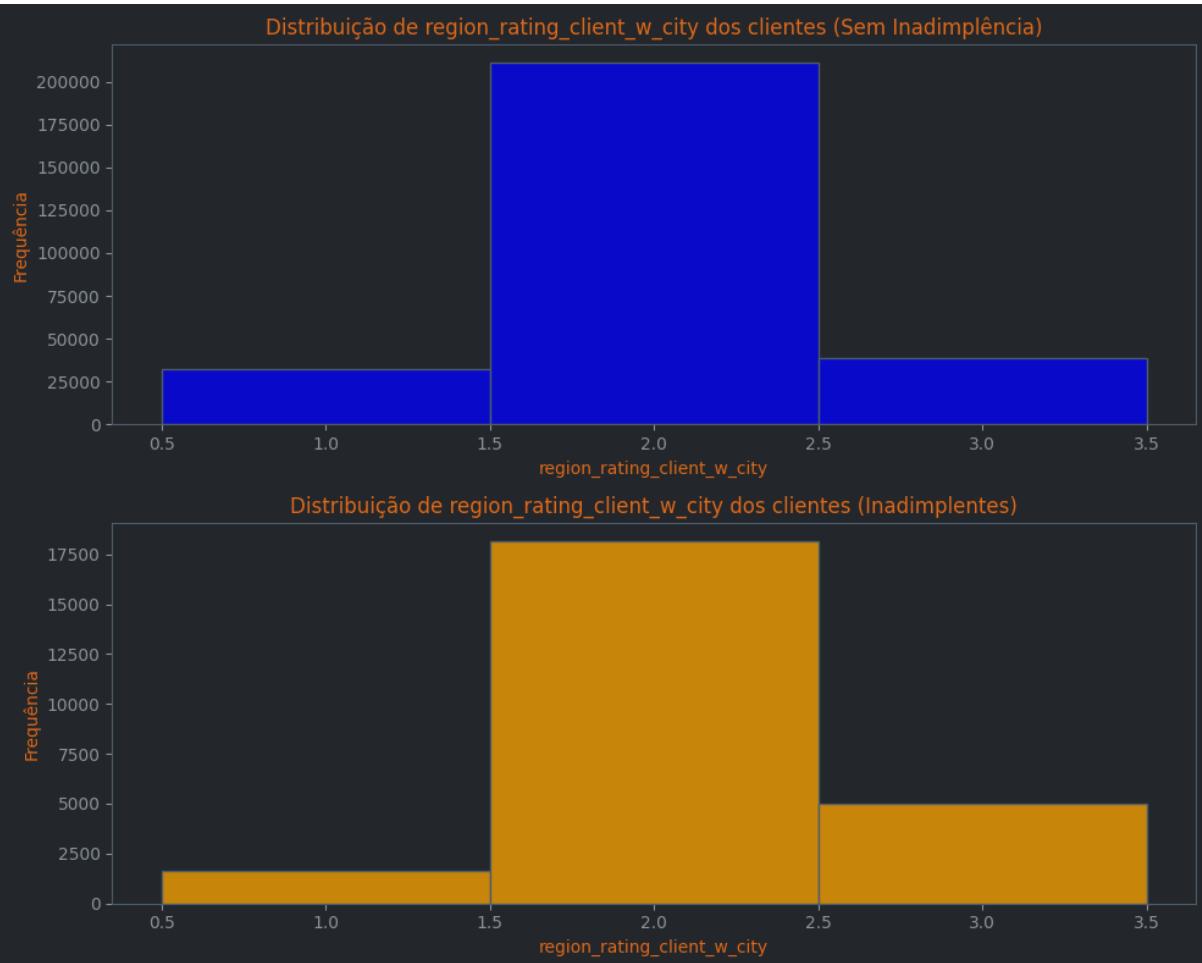
```
# Correlação de region_rating_client com a target
pearson_corr = df3['region_rating_client'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre region_rating_client e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre region_rating_client e target: 0.0589

H29 — O rating da cidade do cliente influencia o risco de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='region_rating_client_w_city',
    title_0='Distribuição de region_rating_client_w_city dos clientes (Sem Inadimplência)',
    title_1='Distribuição de region_rating_client_w_city dos clientes (Inadimplentes)',
    label_x='region_rating_client_w_city',
    discrete=True
)
# Correlação de region_rating_client_w_city com a target
pearson_corr = df3['region_rating_client_w_city'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre region_rating_client_w_city e target: {pearson_corr:.4f}")
```

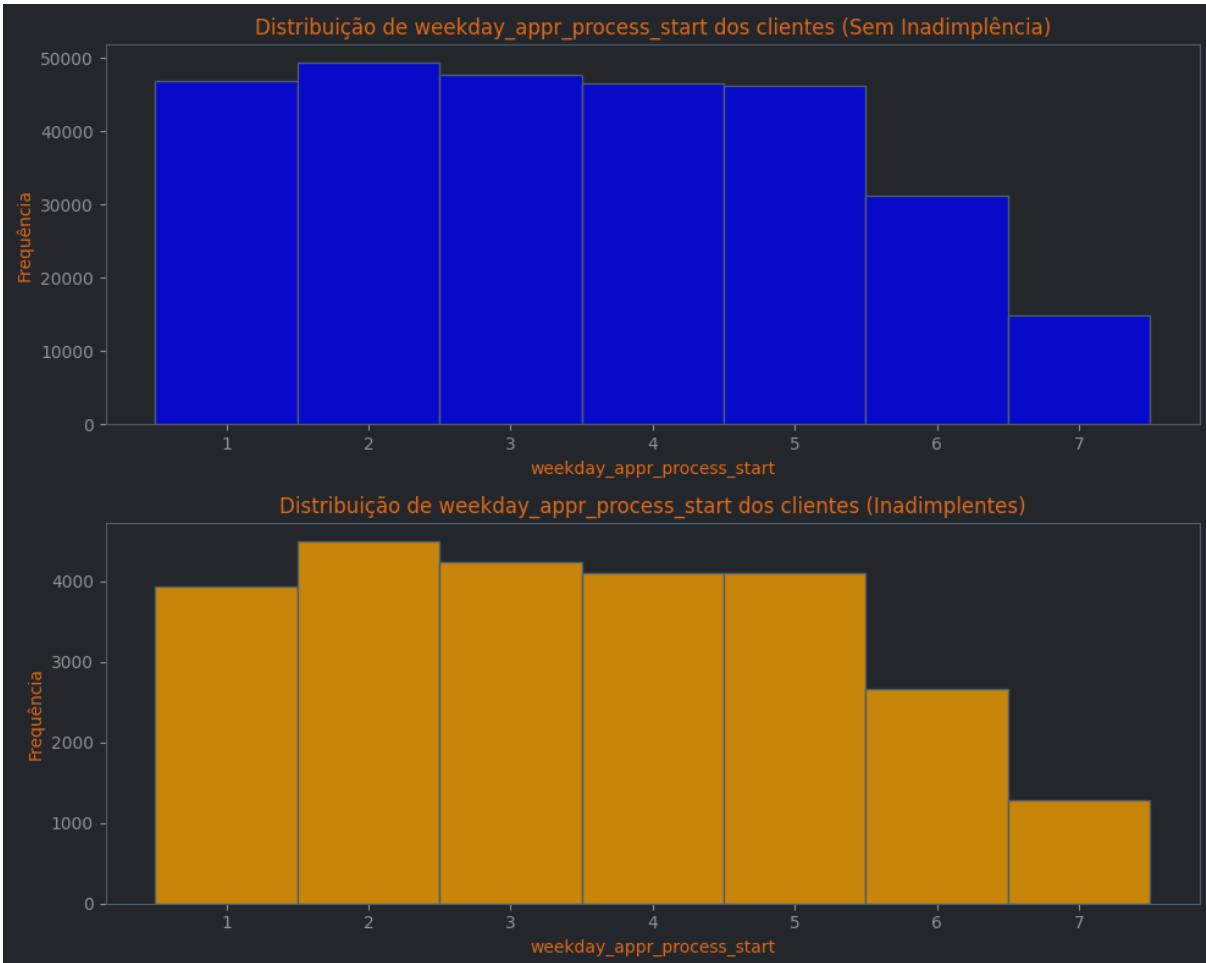


Correlação de Pearson entre region_rating_client_w_city e target: 0.0609

H30 — O dia da semana em que o pedido de crédito é feito influencia a probabilidade de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='weekday_appr_process_start',
    title_0='Distribuição de weekday_appr_process_start dos clientes (Sem Inadimplência)',
    title_1='Distribuição de weekday_appr_process_start dos clientes (Inadimplentes)',
    label_x='weekday_appr_process_start',
    discrete=True
)

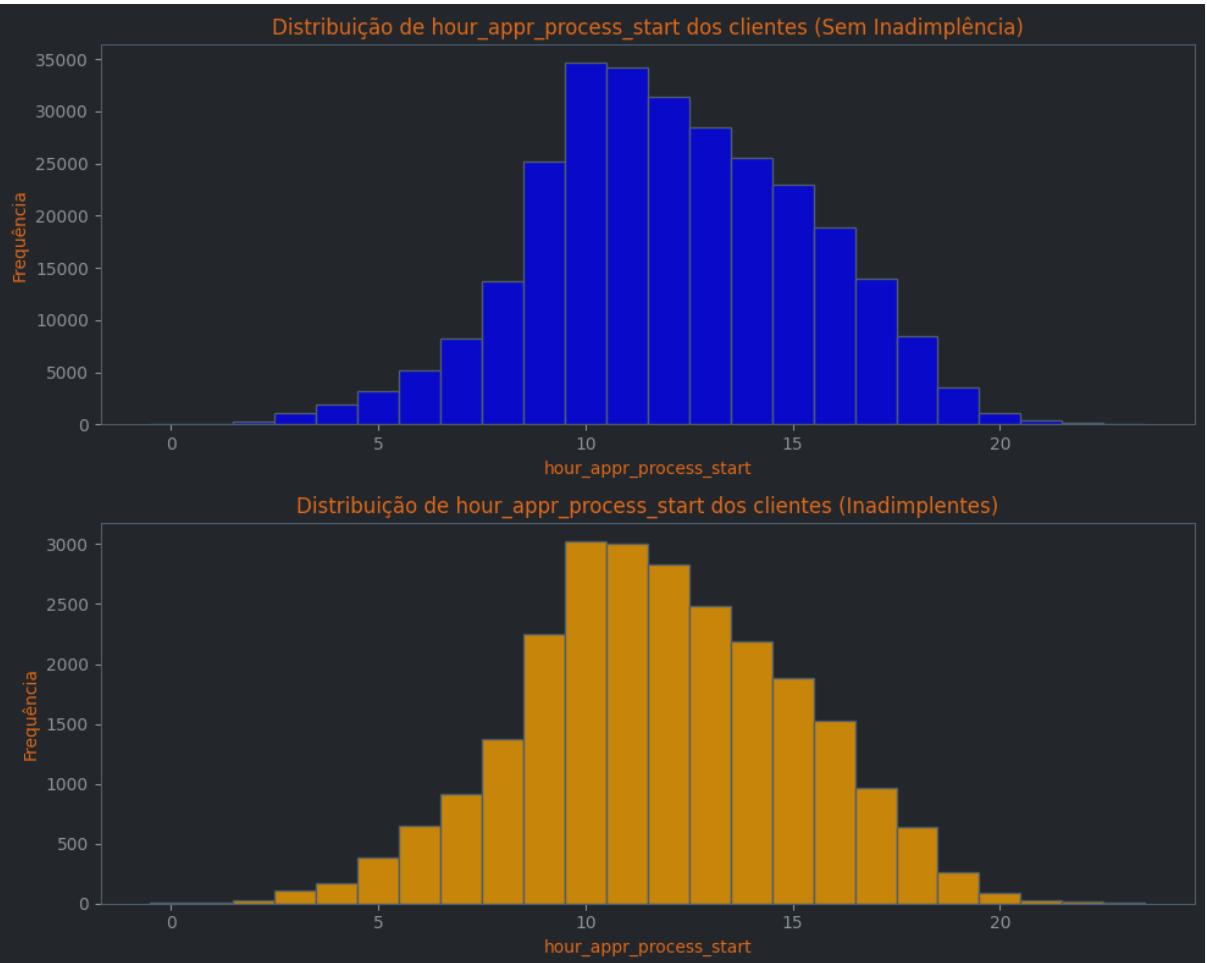
pearson_corr = df3['weekday_appr_process_start'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre weekday_appr_process_start e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre weekday_appr_process_start e target: -0.0002

H31 — O horário da solicitação de crédito está associado à inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='hour_appr_process_start',
    title_0='Distribuição de hour_appr_process_start dos clientes (Sem Inadimplência)',
    title_1='Distribuição de hour_appr_process_start dos clientes (Inadimplentes)',
    label_x='hour_appr_process_start',
    discrete=True
)
pearson_corr = df3['hour_appr_process_start'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre hour_appr_process_start e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre hour_appr_process_start e target: -0.0242

H32 — Morar em região diferente da registrada aumenta o risco de inadimplência.

```
In [ ]: reg_region_flags = [0, 1]

palette_target = {'0': "orange", '1': "red"}

fig, axes = plt.subplots(1, 2, figsize=(14, 6), sharey=False)

for i, flag in enumerate(reg_region_flags):
    data_plot = df3[df3['reg_region_not_live_region'] == flag].copy()
    data_plot['target_str'] = data_plot['target'].astype(str)
    sns.countplot(
        data=data_plot,
        x='target_str',
        hue='target_str',
        palette=palette_target,
        order=['0', '1'],
        ax=axes[i],
        legend=False
    )
    reg_region_label = "Mesma Região" if flag == 0 else "Regiões Diferentes"
    axes[i].set_title(f"Residência e Registro: {reg_region_label}")
    axes[i].set_xlabel('Resposta')
    axes[i].set_ylabel('Frequência')
    axes[i].set_xticks([0, 1])
    axes[i].set_xticklabels(['Não Inadimplente (0)', 'Inadimplente (1)'])

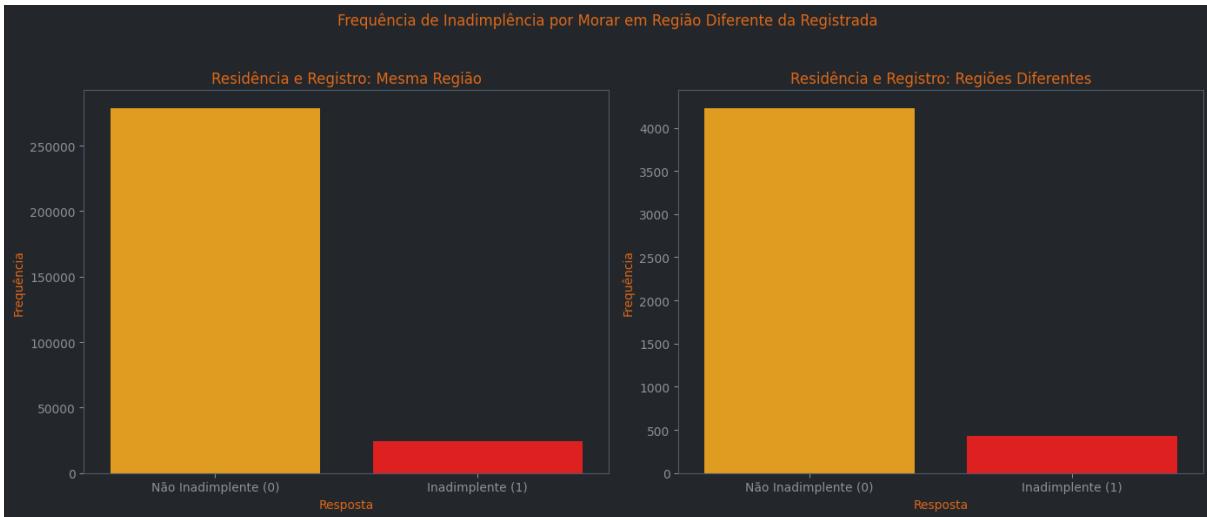
plt.suptitle('Frequência de Inadimplência por Morar em Região Diferente da Registrada')
plt.tight_layout(rect=[0, 0, 1, 0.95])
plt.show()

plot_num_var_by_target(
    df3,
    num_var='reg_region_not_live_region',
```

```

        title_0='Distribuição do Valor de Crédito (amt_credit) dos clientes (Sem Inadimplência)',
        title_1='Distribuição do Valor de Crédito (amt_credit) dos clientes (Inadimplentes)',
        label_x='reg_region_not_live_region',
        discrete=True
    )
calcular_cramers_v(df3, 'reg_region_not_live_region', 'target')

```



V de Cramer entre reg_region_not_live_region e target: 0.0055

H33 — Trabalhar em região diferente da registrada está associado a maior risco.

```

In [ ]: plot_binaria_target(
    df3,
    var_binaria='reg_region_not_work_region',
    label_0='Mesma Região',
    label_1='Regiões Diferentes',
    suptitle='Frequência de Inadimplência por Morar em Região Diferente da Região de Trabalho'
)

calcular_cramers_v(df3, 'reg_region_not_work_region', 'target')

```

H34 — Residir e trabalhar em regiões diferentes pode aumentar a probabilidade de inadimplência.

```

In [ ]: live_region_not_work_flags = [0, 1]

palette_target = {'0': "orange", '1': "red"}

fig, axes = plt.subplots(1, 2, figsize=(14, 6), sharey=False)

for i, flag in enumerate(live_region_not_work_flags):
    data_plot = df3[df3['live_region_not_work_region'] == flag].copy()

    data_plot['target_str'] = data_plot['target'].astype(str)
    sns.countplot(
        data=data_plot,
        x='target_str',
        hue='target_str',
        palette=palette_target,
        order=['0', '1'],
        ax=axes[i],
        legend=False
    )

    live_region_work_str = "Mesma Cidade" if flag == 0 else "Cidades Diferentes"
    axes[i].set_title(f"Cidade de Residência e Trabalho: {live_region_work_str}")
    axes[i].set_xlabel('Resposta')
    axes[i].set_ylabel('Frequência')
    axes[i].set_xticks([0, 1])
    axes[i].set_xticklabels(['Não Inadimplente (0)', 'Inadimplente (1)'])

plt.suptitle('Frequência de Inadimplência por Morar em Cidade Diferente da Cidade de Trabalho')

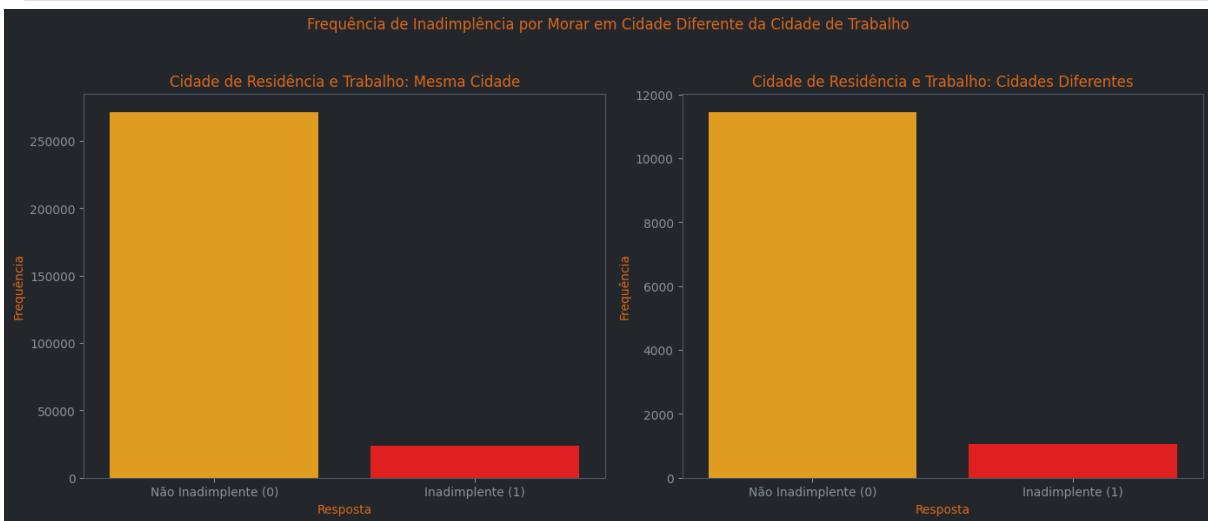
```

```

plt.tight_layout(rect=[0, 0, 1, 0.95])
plt.show()

plot_binaria_target(
    df3,
    var_binaria='live_region_not_work_region',
    label_0='Mesma Cidade',
    label_1='Cidades Diferentes',
    suptitle='Frequência de Inadimplência por Morar em Cidade Diferente da Cidade de Trabalho'
)
calcular_cramers_v(df3, 'live_region_not_work_region', 'target')

```



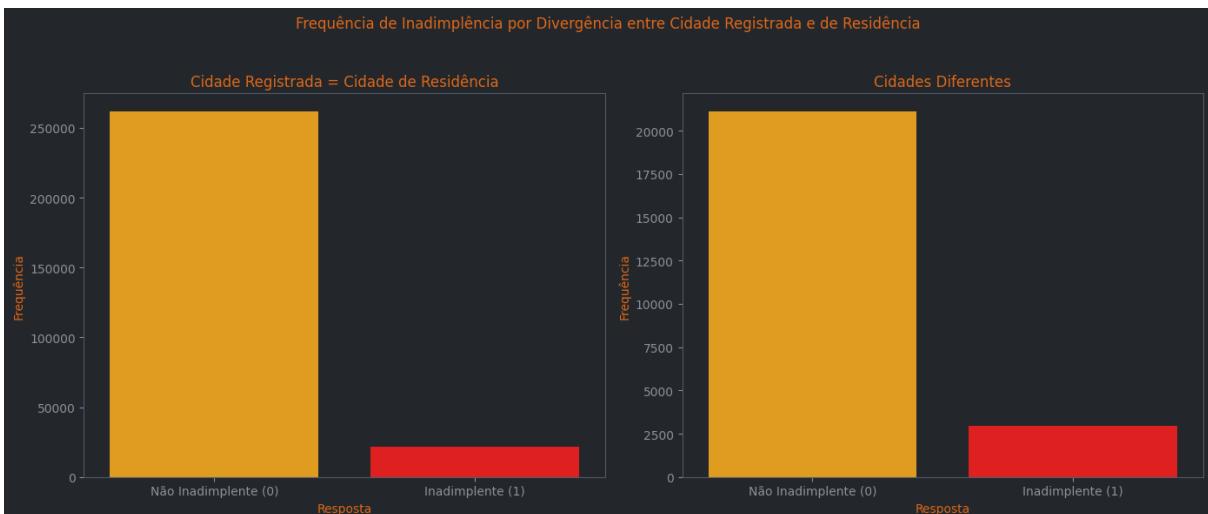
V de Cramer entre live_region_not_work_region e target: 0.0028

H35 — Morar em cidade diferente da registrada aumenta a chance de inadimplência.

```

In [ ]: plot_binaria_target(
    df3,
    var_binaria='reg_city_not_live_city',
    label_0='Cidade Registrada = Cidade de Residência',
    label_1='Cidades Diferentes',
    suptitle='Frequência de Inadimplência por Divergência entre Cidade Registrada e de Residência'
)
calcular_cramers_v(df3, 'reg_city_not_live_city', 'target')

```



V de Cramer entre reg_city_not_live_city e target: 0.0444

H36 — Trabalhar em cidade diferente da registrada aumenta o risco de inadimplência.

```

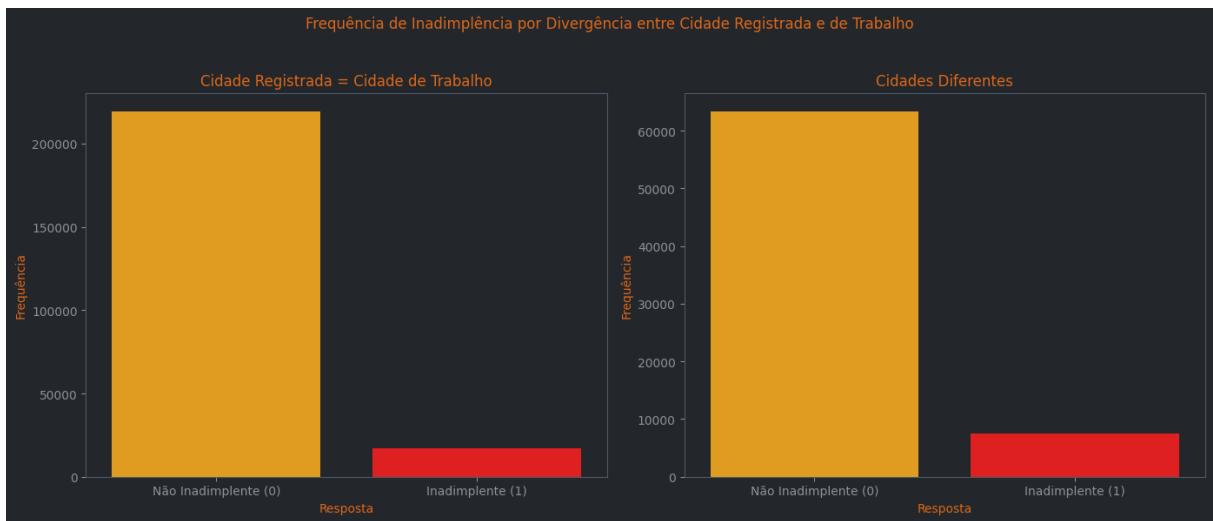
In [ ]: plot_binaria_target(
    df3,
    var_binaria='reg_city_not_work_city',
    label_0='Cidade Registrada = Cidade de Trabalho',
    label_1='Cidades Diferentes',

```

```

    suptitle='Frequência de Inadimplência por Divergência entre Cidade Registrada e de Trabalho
)
calcular_cramers_v(df3, 'reg_city_not_work_city', 'target')

```



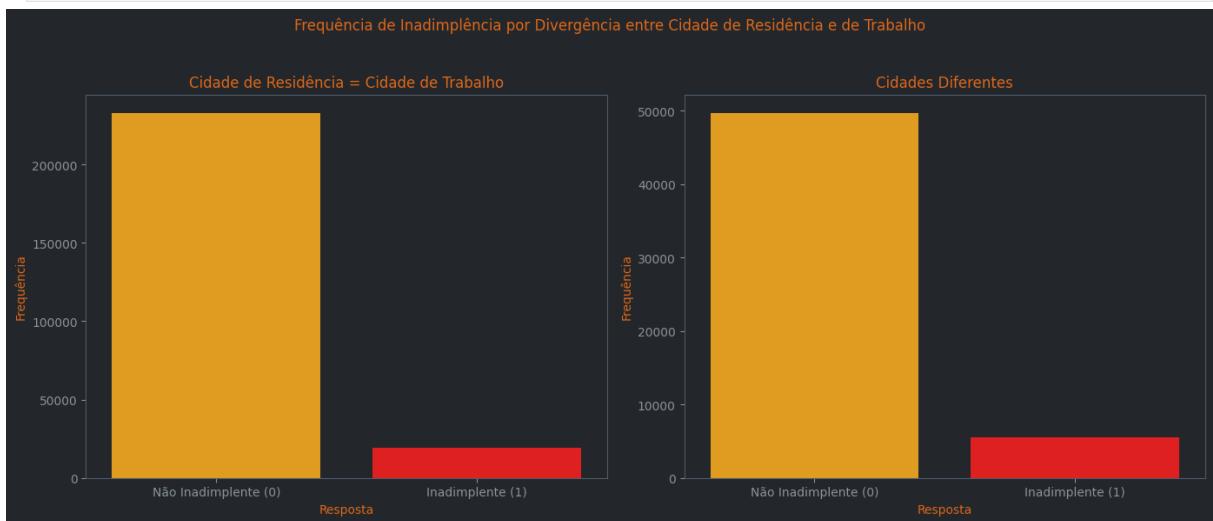
V de Cramer entre reg_city_not_work_city e target: 0.0510

H37 — Residir em cidade diferente do local de trabalho influencia o risco de inadimplência.

```

In [ ]: plot_binaria_target(
    df3,
    var_binaria='live_city_not_work_city',
    label_0='Cidade de Residência = Cidade de Trabalho',
    label_1='Cidades Diferentes',
    suptitle='Frequência de Inadimplência por Divergência entre Cidade de Residência e de Trabalho
)
calcular_cramers_v(df3, 'live_city_not_work_city', 'target')

```



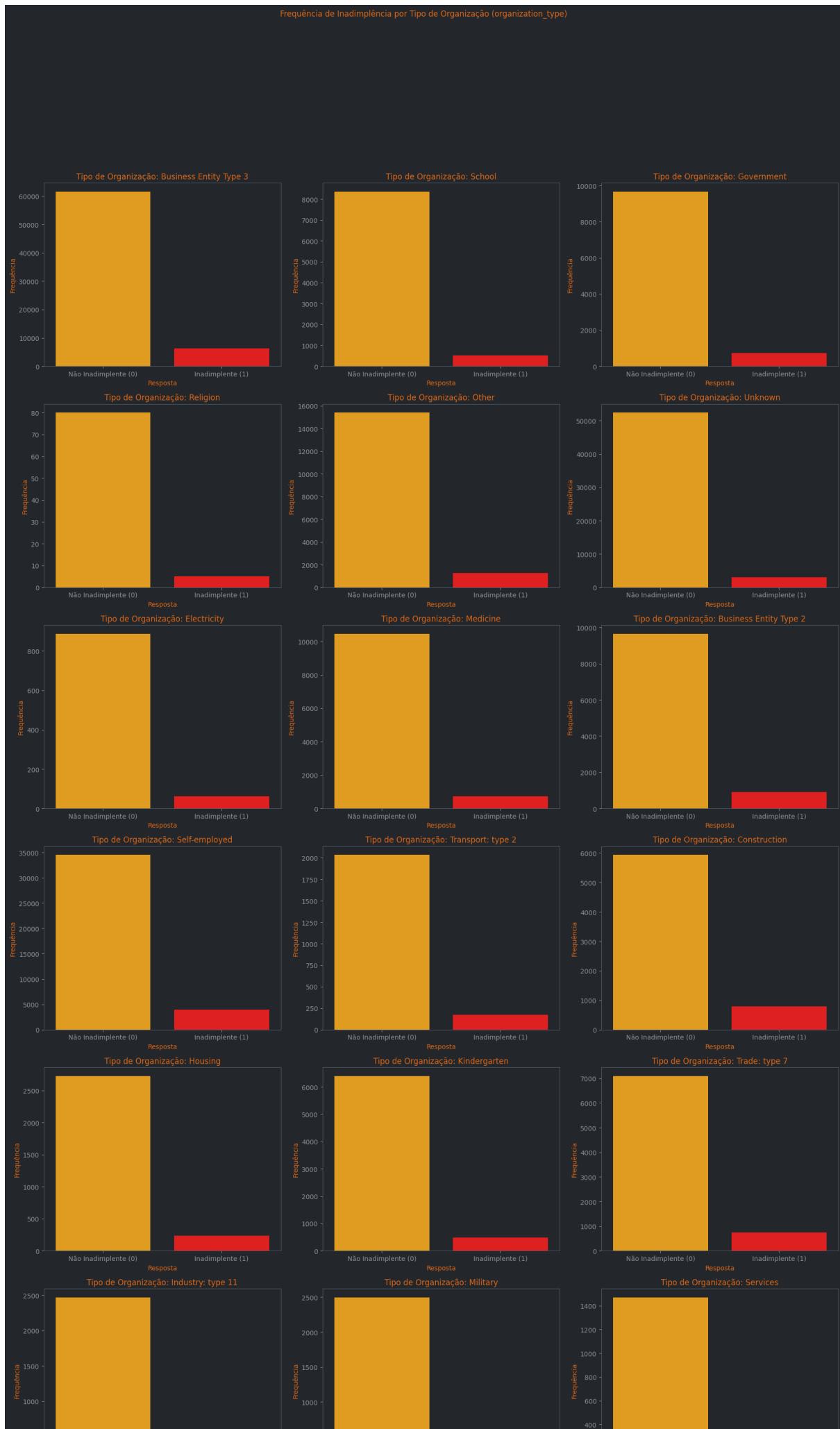
V de Cramer entre live_city_not_work_city e target: 0.0325

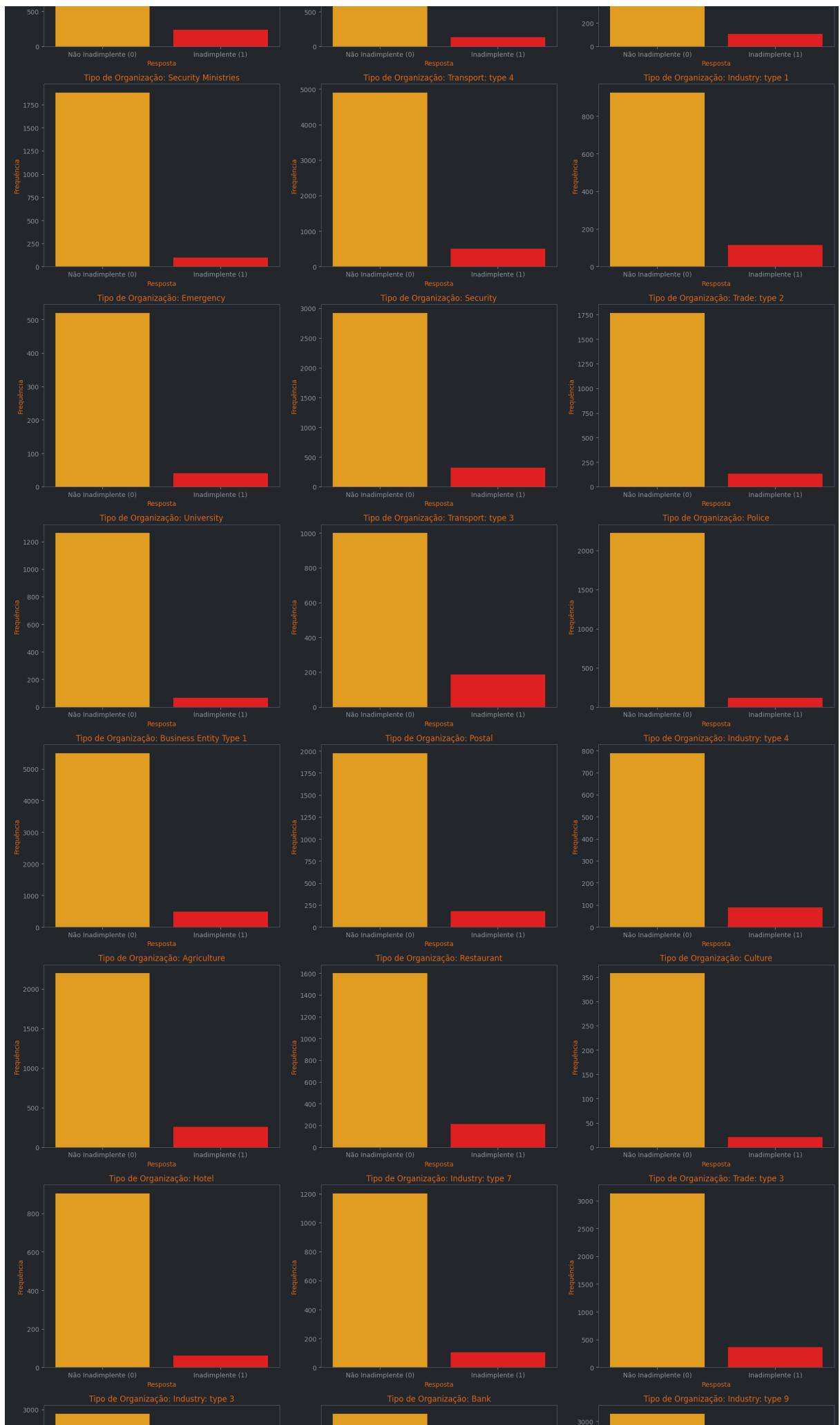
H38 — O tipo de organização empregadora está associado ao risco de inadimplência.

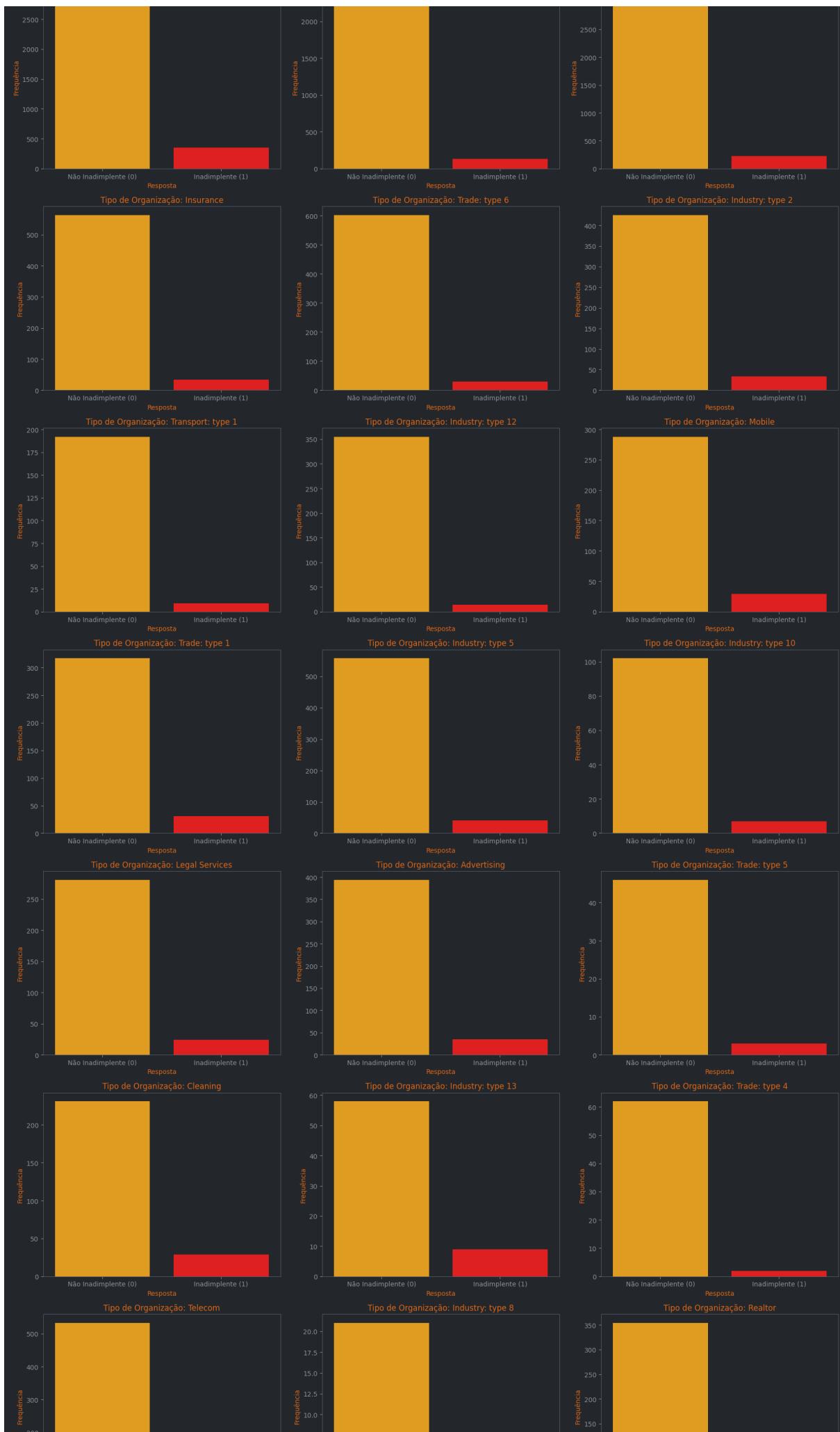
```

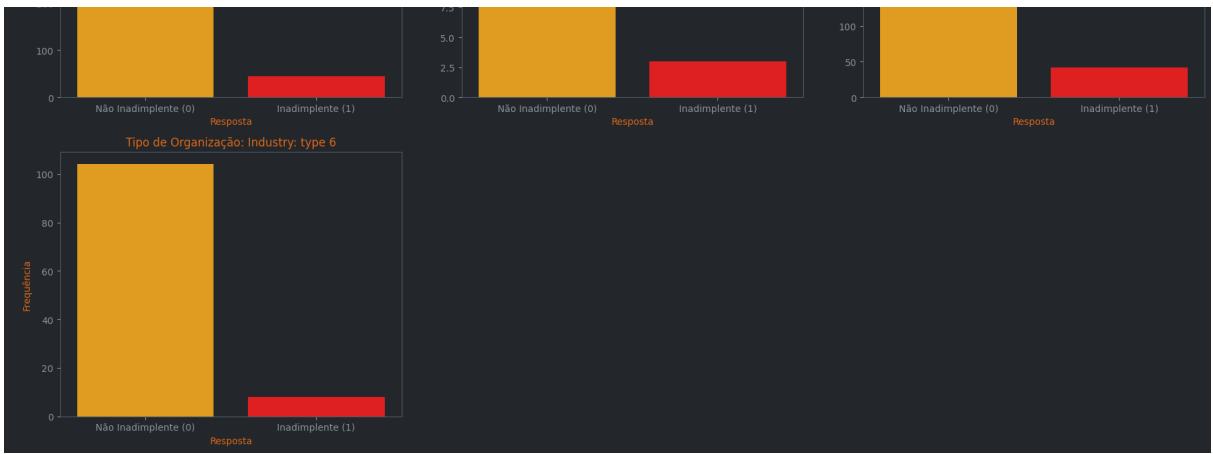
In [ ]: plot_target_by_categorical(df3, cat_col='organization_type', target_col='target')
calcular_cramers_v(df3, 'organization_type', 'target')

```





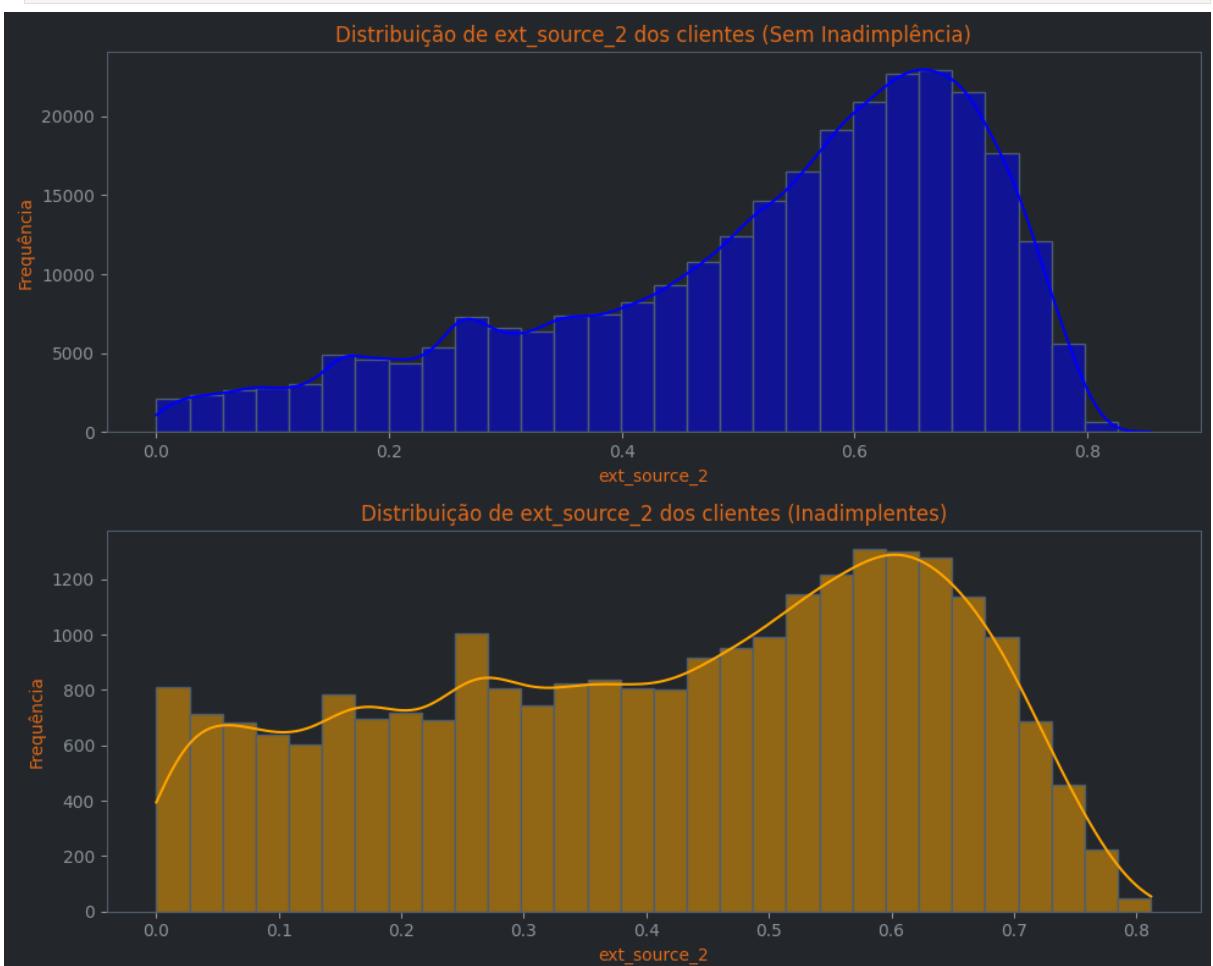




V de Cramer entre organization_type e target: 0.0723

H39 — O score externo 2 está inversamente relacionado à inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='ext_source_2',
    title_0='Distribuição de ext_source_2 dos clientes (Sem Inadimplência)',
    title_1='Distribuição de ext_source_2 dos clientes (Inadimplentes)',
    label_x='Distribuição de ext_source_2 dos clientes (Inadimplentes)',
    discrete=True
)
# Correlação de ext_source_2 com a target
pearson_corr = df3['ext_source_2'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre ext_source_2 e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre ext_source_2 e target: -0.1603

H40 — O score externo 3 está inversamente relacionado à inadimplência.

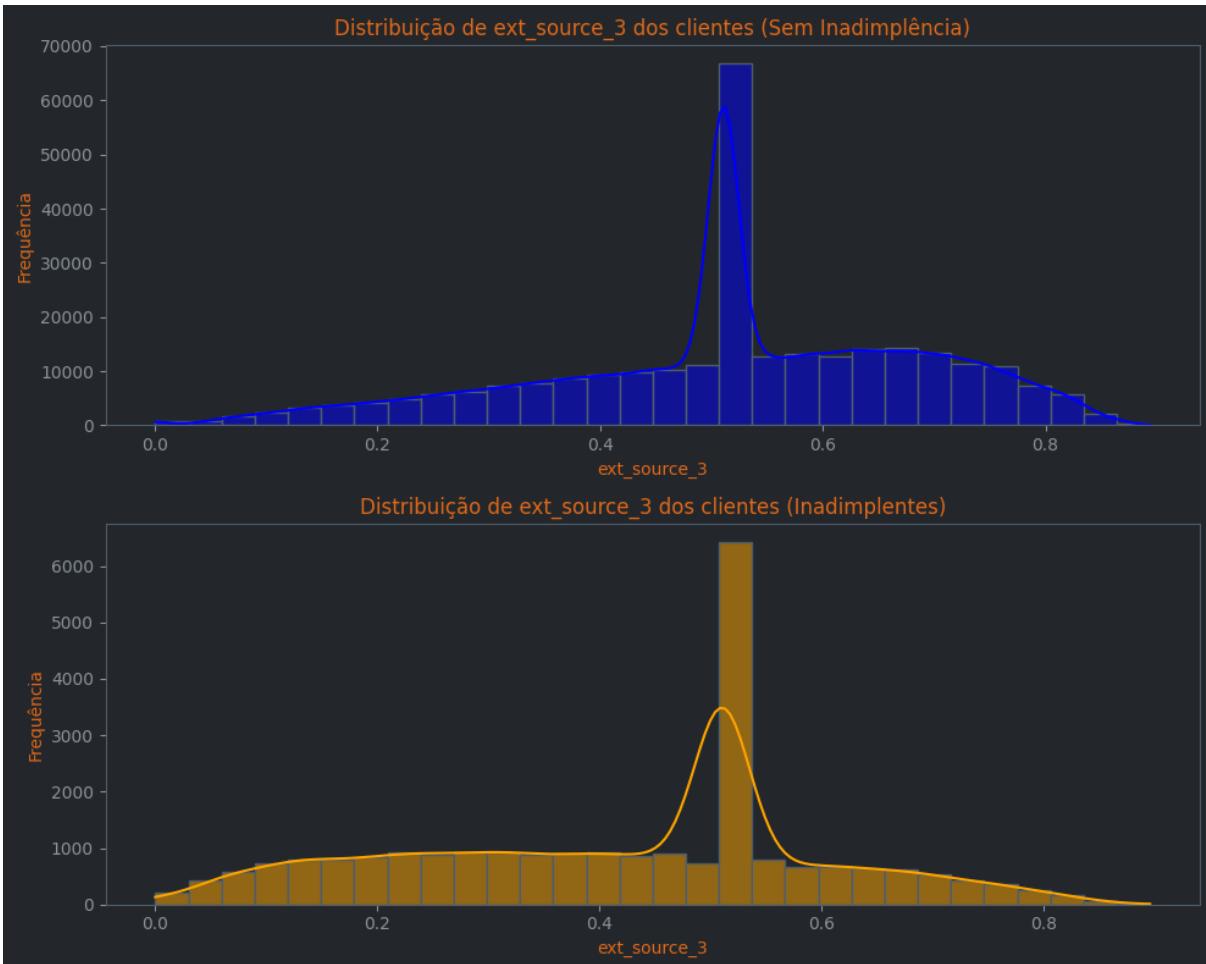
```
In [ ]: plot_num_var_by_target(
```

```

df3,
num_var='ext_source_2',
title_0='Distribuição de ext_source_3 dos clientes (Sem Inadimplência)',
title_1='Distribuição de ext_source_3 dos clientes (Inadimplentes)',
label_x='Distribuição de ext_source_3 dos clientes (Inadimplentes)',
discrete=True
)

# Correlação de ext_source_3 com a target
pearson_corr = df3['ext_source_3'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre ext_source_3 e target: {pearson_corr:.4f}")

```



Correlação de Pearson entre ext_source_3 e target: -0.1574

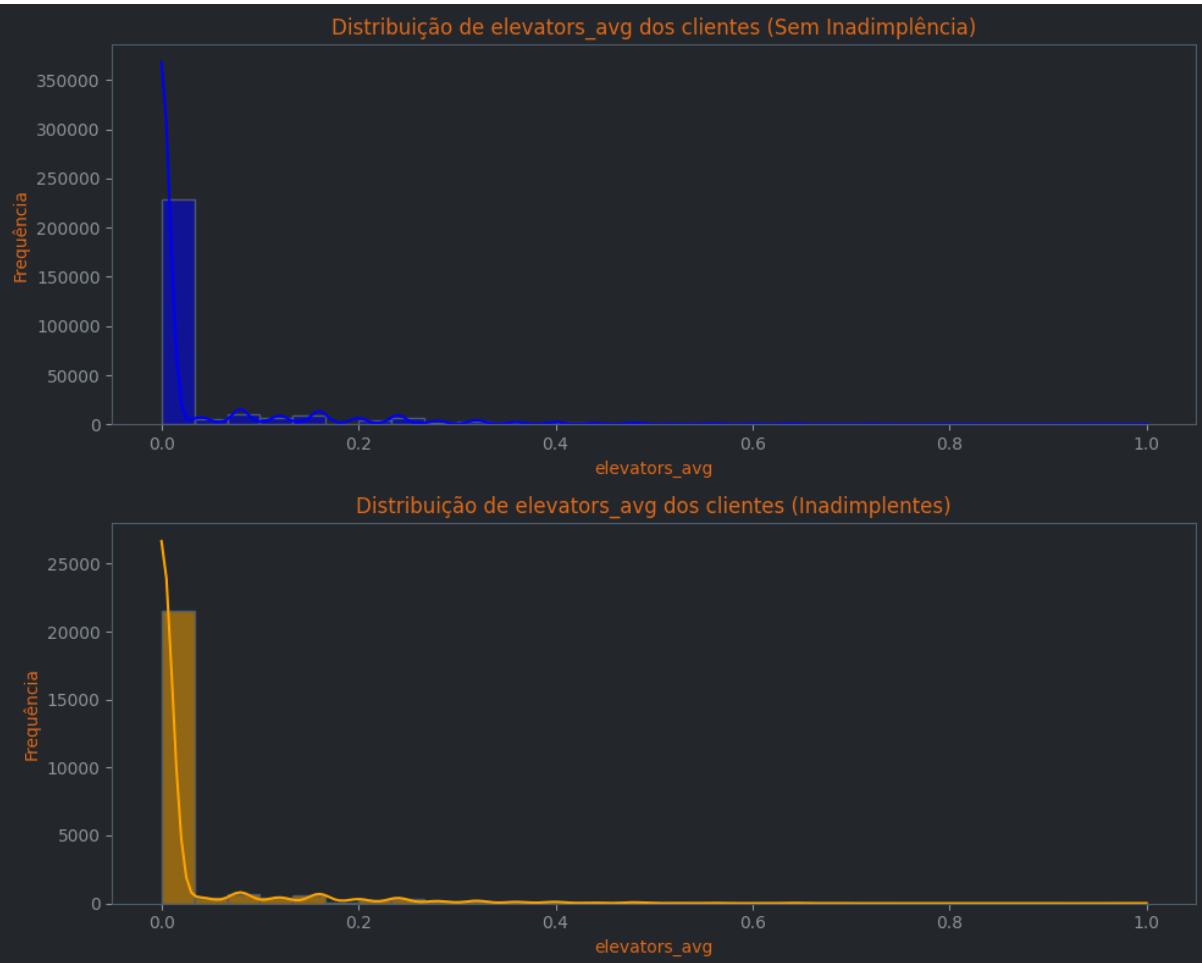
Bloco 3 de Hipóteses

H41 — A quantidade média de elevadores no imóvel está inversamente relacionada à probabilidade de inadimplência.

```

In [ ]: plot_num_var_by_target(
    df3,
    num_var='elevators_avg',
    title_0='Distribuição de elevators_avg dos clientes (Sem Inadimplência)',
    title_1='Distribuição de elevators_avg dos clientes (Inadimplentes)',
    label_x='elevators_avg',
    discrete=True
)
# Correlação de elevators_avg com a target
pearson_corr = df3['elevators_avg'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre elevators_avg e target: {pearson_corr:.4f}")

```



Correlação de Pearson entre elevators_avg e target: -0.0359

H42 — O tipo de habitação influencia o risco de inadimplência.

```
In [ ]: plot_target_by_categorical(df3, cat_col='housetype_mode', target_col='target')

calcular_cramers_v(df3, 'housetype_mode', 'target')
```



V de Cramer entre housetype_mode e target: 0.0407

H43 — O material predominante das paredes do imóvel está associado ao risco de inadimplência.

```
In [ ]: plot_target_by_categorical(df3, cat_col='wallsmaterial_mode', target_col='target')
```

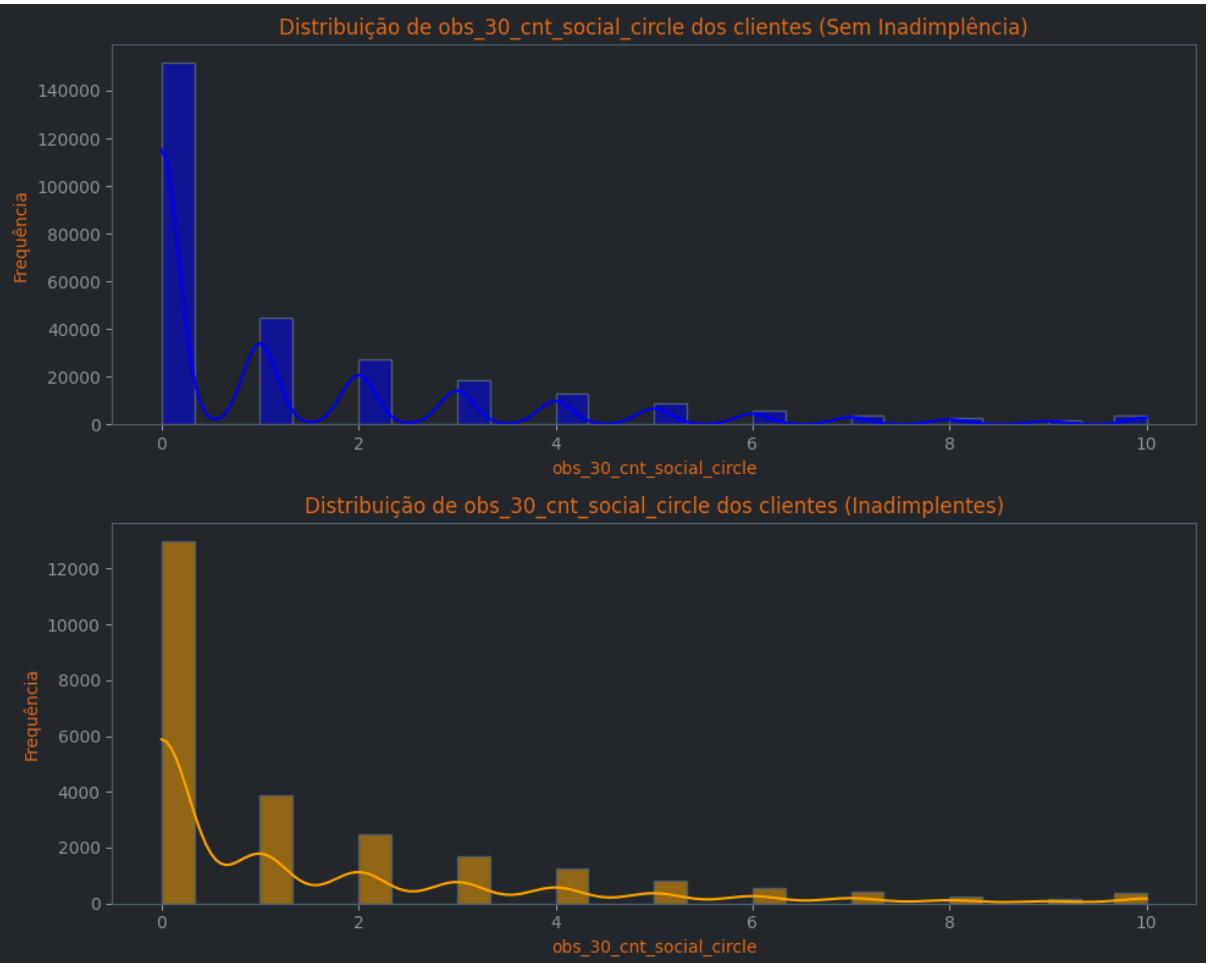
```
calcular_cramers_v(df3, 'wallsmaterial_mode', 'target')
```



V de Cramer entre wallsmaterial_mode e target: 0.0441

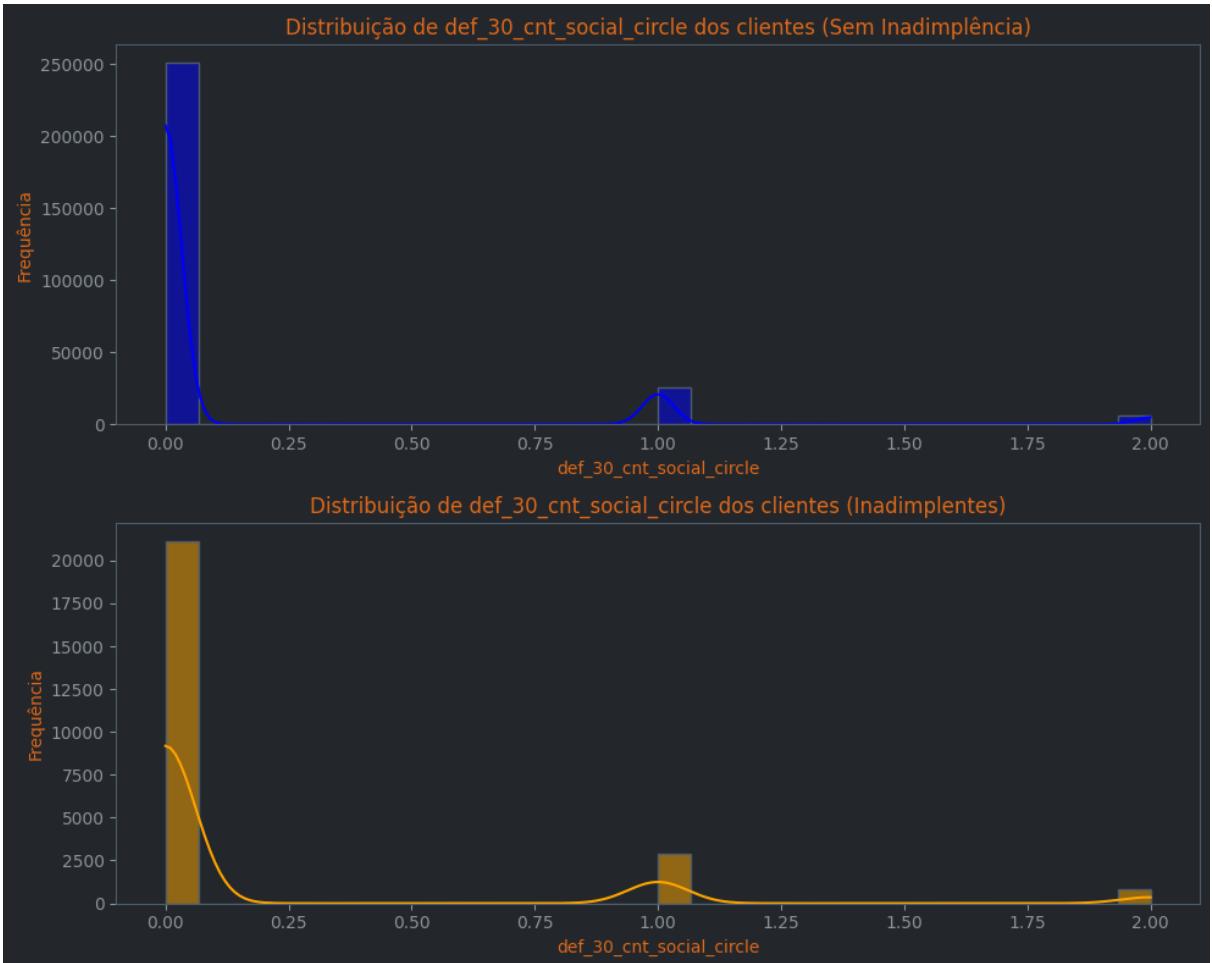
H44 — Um maior número de observações ou de inadimplentes no círculo social de 30 dias aumenta a probabilidade de inadimplência do cliente.

```
In [ ]: plot_num_var_by_target(  
    df3,  
    num_var='obs_30_cnt_social_circle',  
    title_0='Distribuição de obs_30_cnt_social_circle dos clientes (Sem Inadimplência)',  
    title_1='Distribuição de obs_30_cnt_social_circle dos clientes (Inadimplentes)',  
    label_x='obs_30_cnt_social_circle',  
    discrete=True  
)  
# Correlação de obs_30_cnt_social_circle com a target  
pearson_corr = df3['obs_30_cnt_social_circle'].corr(df3['target'], method='pearson')  
print(f"Correlação de Pearson entre obs_30_cnt_social_circle e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre `obs_30_cnt_social_circle` e `target`: 0.0104

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='def_30_cnt_social_circle',
    title_0='Distribuição de def_30_cnt_social_circle dos clientes (Sem Inadimplência)',
    title_1='Distribuição de def_30_cnt_social_circle dos clientes (Inadimplentes)',
    label_x='def_30_cnt_social_circle',
    discrete=True
)
# Correlação de def_30_cnt_social_circle com a target
pearson_corr = df3['def_30_cnt_social_circle'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre def_30_cnt_social_circle e target: {pearson_corr:.4f}")
```

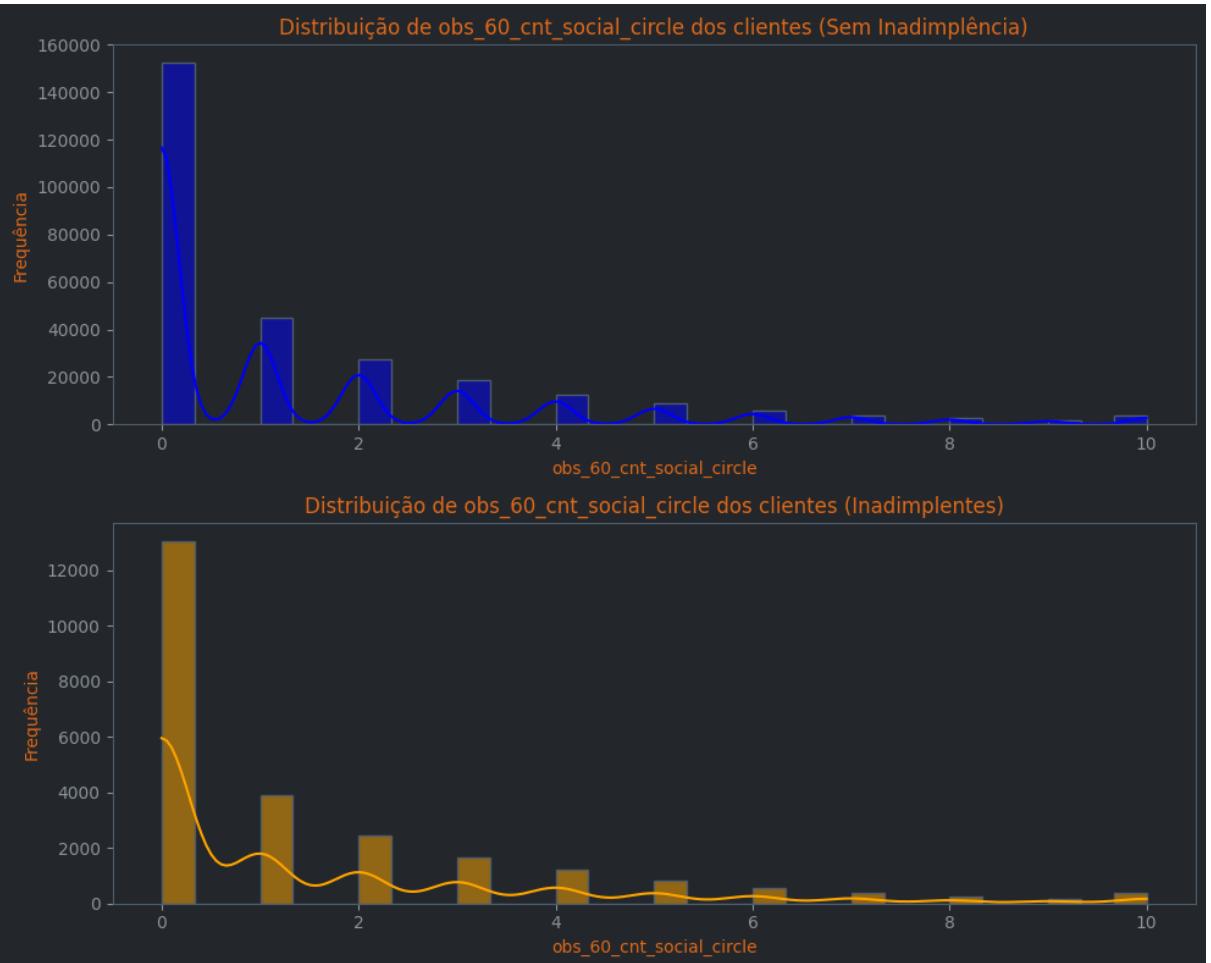


Correlação de Pearson entre def_30_cnt_social_circle e target: 0.0331

H45 — A inadimplência no círculo social de 60 dias está positivamente associada ao risco de inadimplência individual.

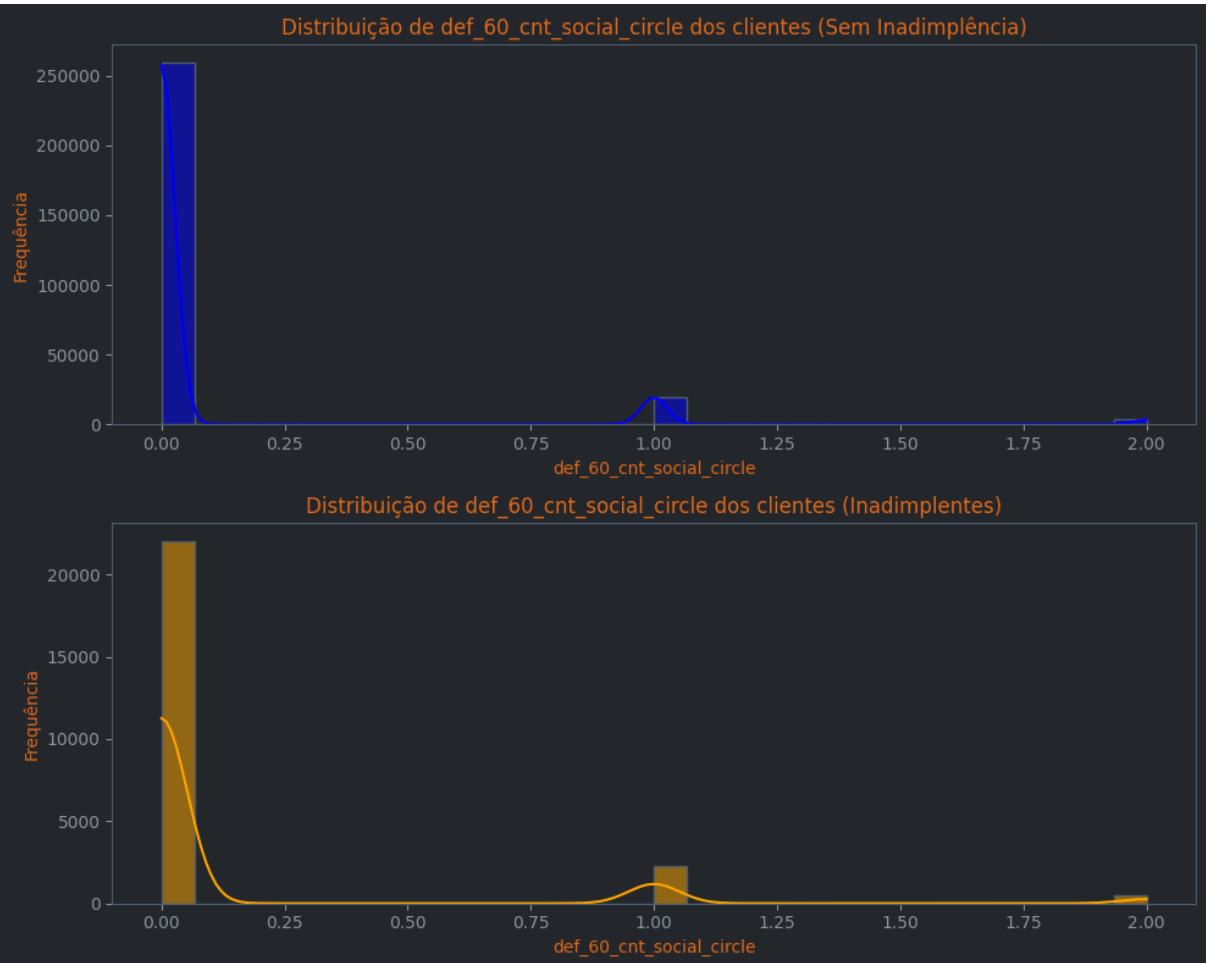
```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='obs_60_cnt_social_circle',
    title_0='Distribuição de obs_60_cnt_social_circle dos clientes (Sem Inadimplência)',
    title_1='Distribuição de obs_60_cnt_social_circle dos clientes (Inadimplentes)',
    label_x='obs_60_cnt_social_circle',
    discrete=True
)

# Correlação de obs_60_cnt_social_circle com a target
pearson_corr = df3['obs_60_cnt_social_circle'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre obs_60_cnt_social_circle e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre obs_60_cnt_social_circle e target: 0.0103

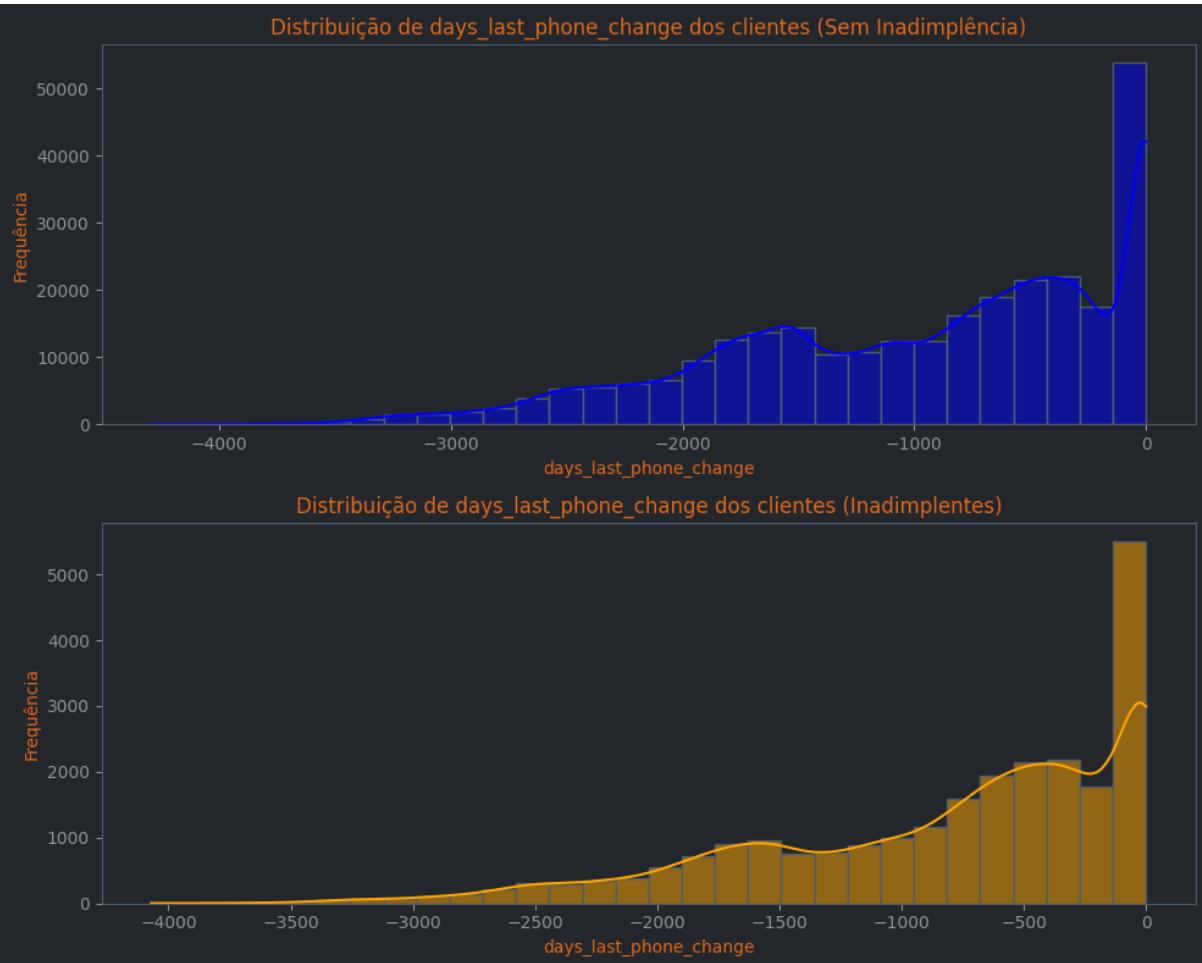
```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='def_60_cnt_social_circle',
    title_0='Distribuição de def_60_cnt_social_circle dos clientes (Sem Inadimplência)',
    title_1='Distribuição de def_60_cnt_social_circle dos clientes (Inadimplentes)',
    label_x='def_60_cnt_social_circle',
    discrete=True
)
# Correlação de def_60_cnt_social_circle com a target
pearson_corr = df3['def_60_cnt_social_circle'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre def_60_cnt_social_circle e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre `def_60_cnt_social_circle` e `target`: 0.0318

H46 — O tempo desde a última troca de telefone está relacionado à probabilidade de inadimplência.

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='days_last_phone_change',
    title_0='Distribuição de days_last_phone_change dos clientes (Sem Inadimplência)',
    title_1='Distribuição de days_last_phone_change dos clientes (Inadimplentes)',
    label_x='days_last_phone_change',
    discrete=True
)
# Correlação de days_last_phone_change com a target
pearson_corr = df3['days_last_phone_change'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre days_last_phone_change e target: {pearson_corr:.4f}")
```

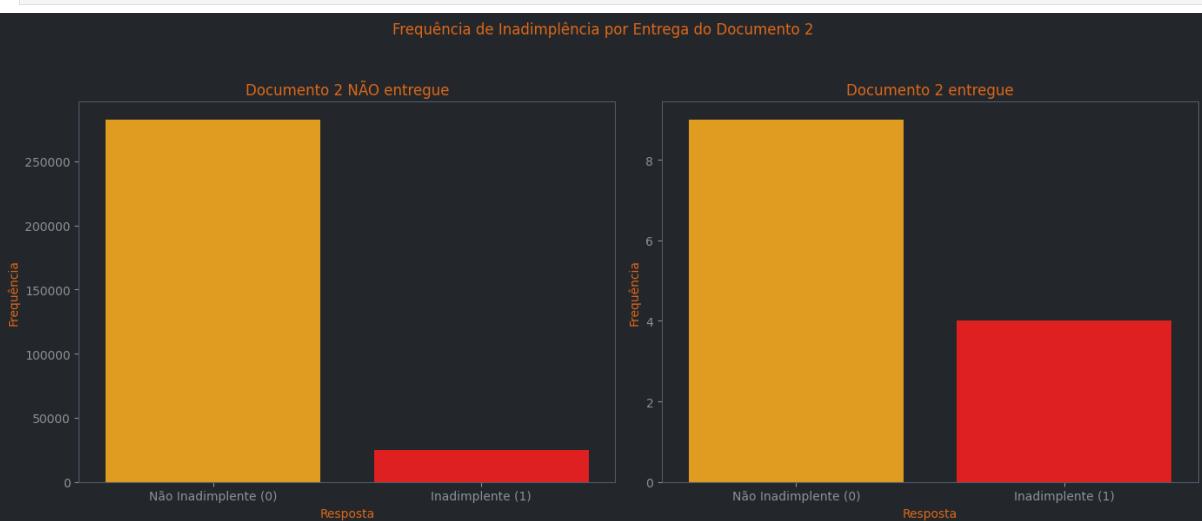


Correlação de Pearson entre days_last_phone_change e target: 0.0552

H47 — Algum documento não entregue está relacionado a inadimplência

flag_document_2

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_document_2',
    label_0='Documento 2 NÃO entregue',
    label_1='Documento 2 entregue',
    suptitle='Frequência de Inadimplência por Entrega do Documento 2'
)
calcular_cramers_v(df3, 'flag_document_2', 'target')
```

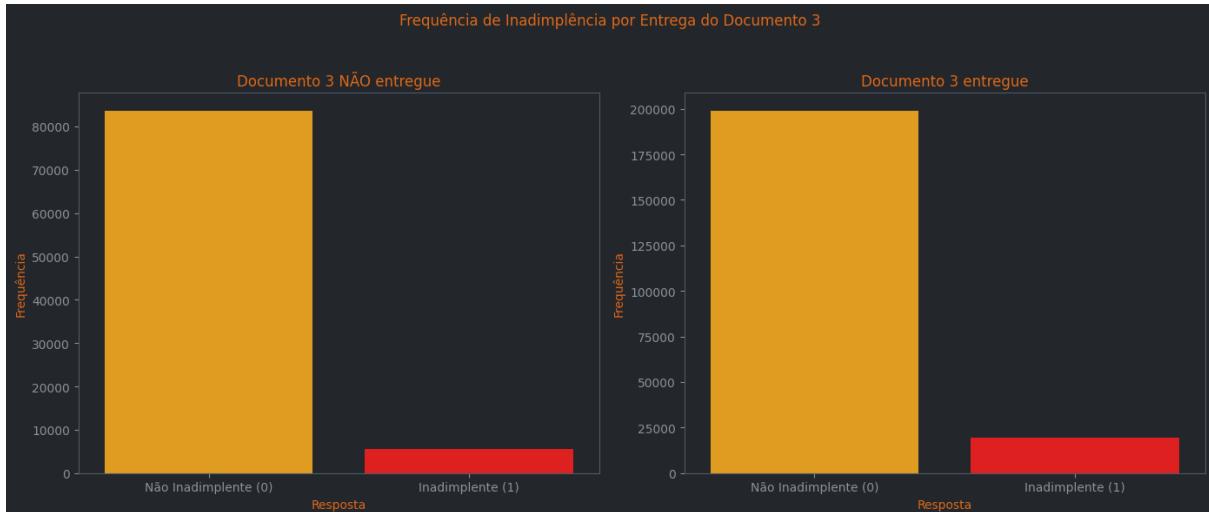


V de Cramer entre flag_document_2 e target: 0.0045

flag_document_3

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_document_3',
    label_0='Documento 3 NÃO entregue',
    label_1='Documento 3 entregue',
    suptitle='Frequência de Inadimplência por Entrega do Documento 3'
)

calcular_cramers_v(df3, 'flag_document_3', 'target')
```

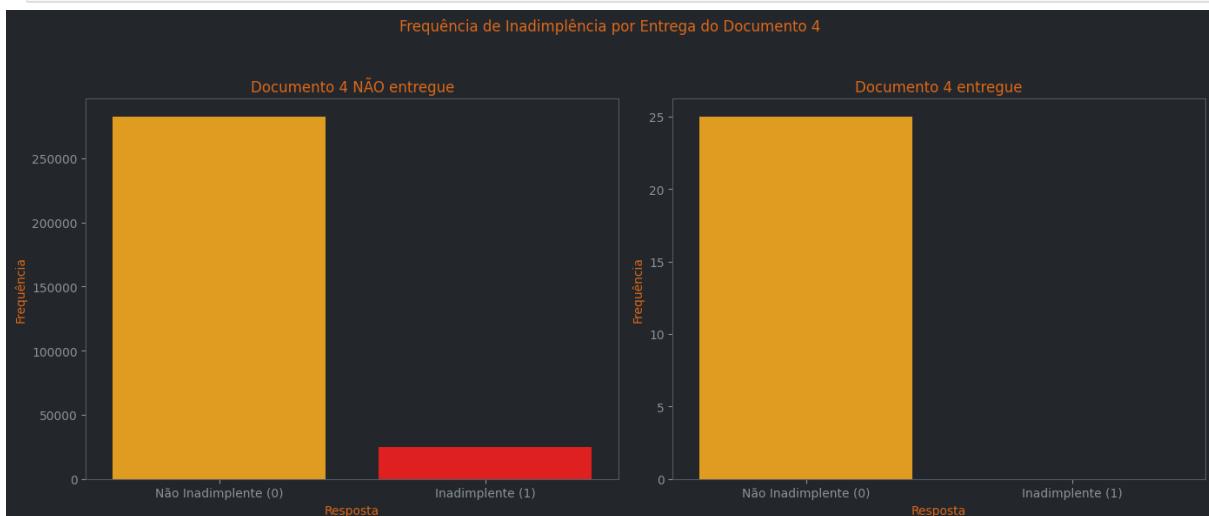


V de Cramer entre flag_document_3 e target: 0.0443

flag_document_4

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_document_4',
    label_0='Documento 4 NÃO entregue',
    label_1='Documento 4 entregue',
    suptitle='Frequência de Inadimplência por Entrega do Documento 4'
)

calcular_cramers_v(df3, 'flag_document_4', 'target')
```

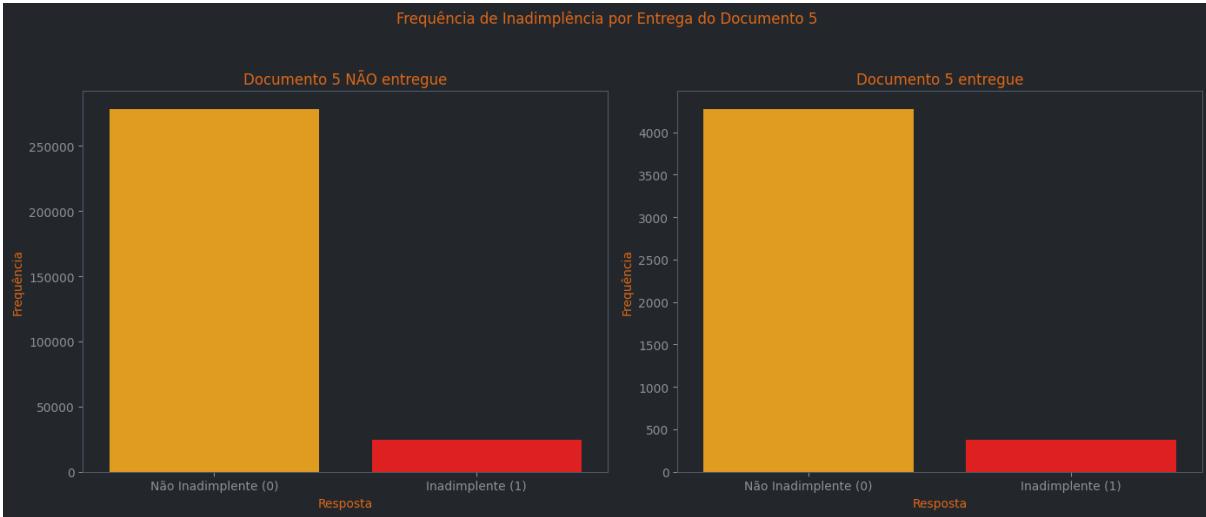


V de Cramer entre flag_document_4 e target: 0.0020

flag_document_5

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_document_5',
    label_0='Documento 5 NÃO entregue',
    label_1='Documento 5 entregue',
    suptitle='Frequência de Inadimplência por Entrega do Documento 5'
)
```

```
calcular_cramers_v(df3, 'flag_document_5', 'target')
```

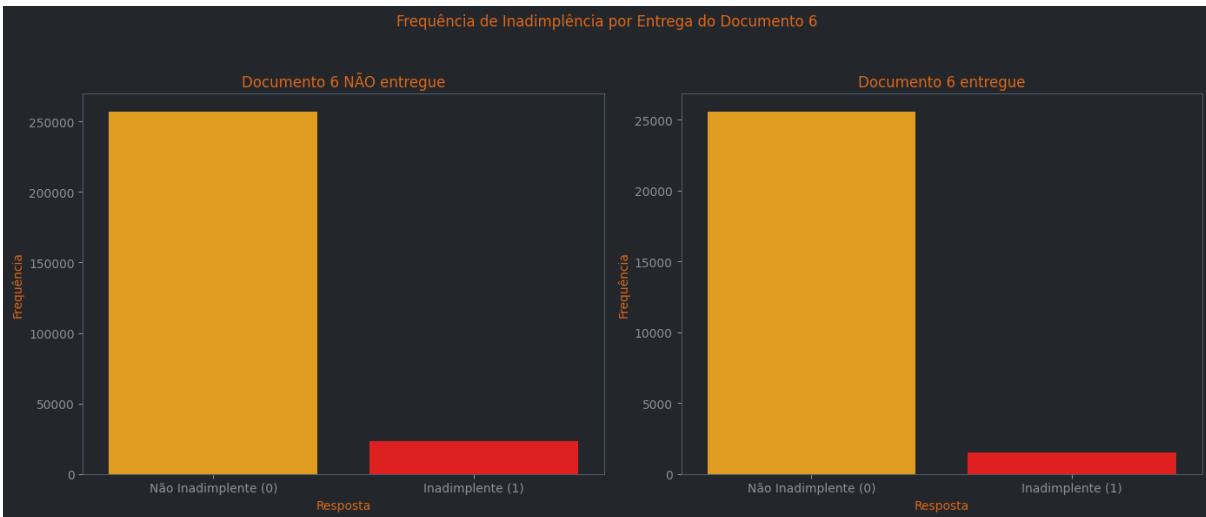


V de Cramer entre flag_document_5 e target: 0.0003

flag_document_6

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_6',  
        label_0='Documento 6 NÃO entregue',  
        label_1='Documento 6 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 6'  
)
```

```
calcular_cramers_v(df3, 'flag_document_6', 'target')
```



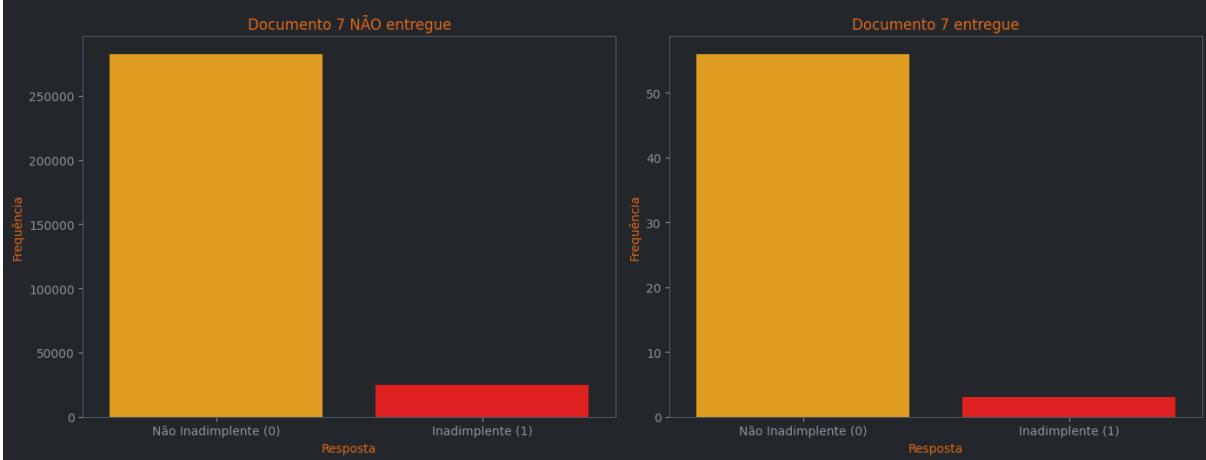
V de Cramer entre flag_document_6 e target: 0.0286

flag_document_7

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_7',  
        label_0='Documento 7 NÃO entregue',  
        label_1='Documento 7 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 7'  
)
```

```
calcular_cramers_v(df3, 'flag_document_7', 'target')
```

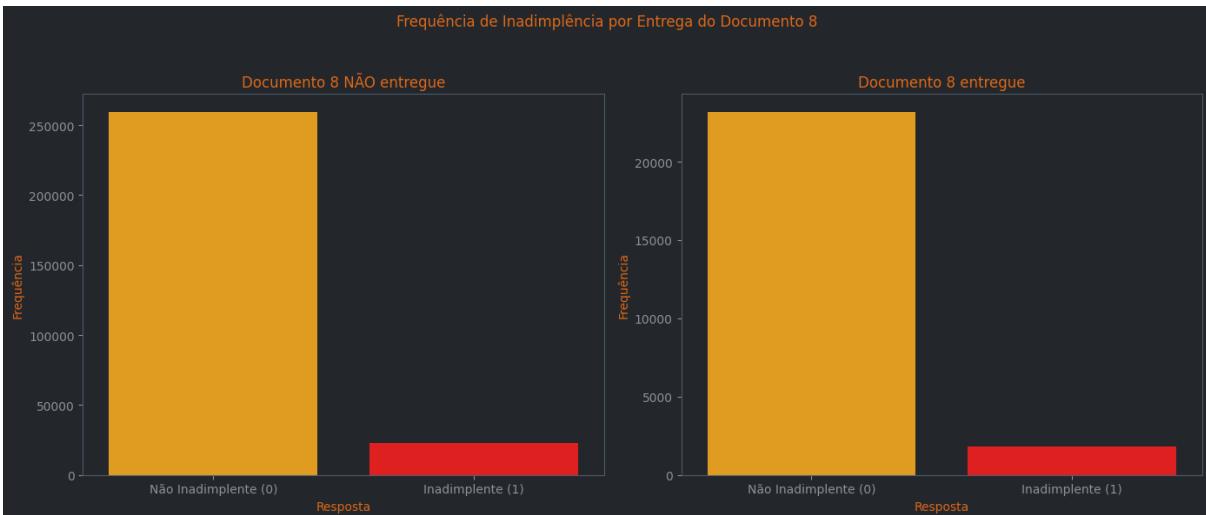
Frequência de Inadimplência por Entrega do Documento 7



V de Cramer entre flag_document_7 e target: 0.0011

flag_document_8

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_8',  
        label_0='Documento 8 NÃO entregue',  
        label_1='Documento 8 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 8'  
)  
  
calcular_cramers_v(df3, 'flag_document_8', 'target')
```

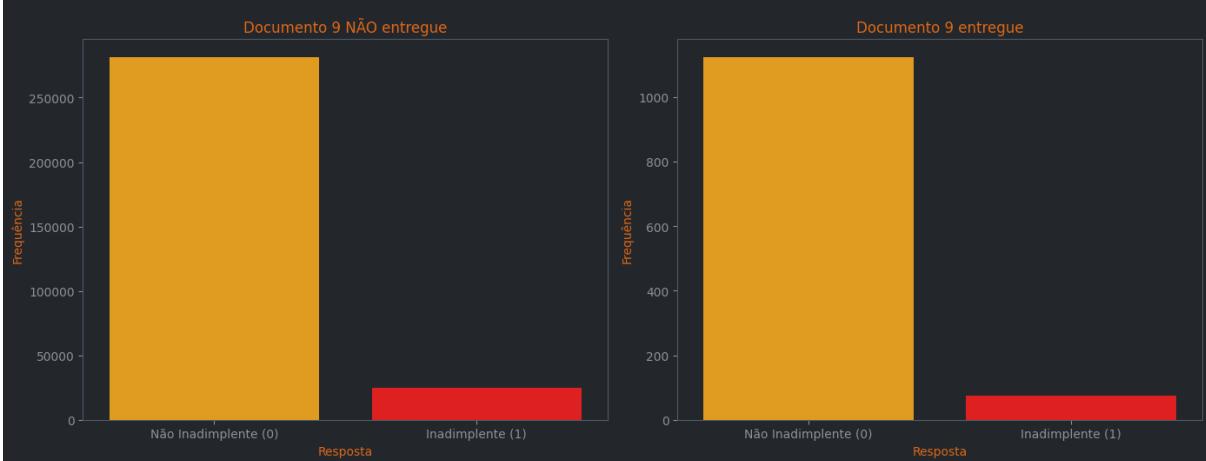


V de Cramer entre flag_document_8 e target: 0.0080

flag_document_9

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_9',  
        label_0='Documento 9 NÃO entregue',  
        label_1='Documento 9 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 9'  
)  
  
calcular_cramers_v(df3, 'flag_document_9', 'target')
```

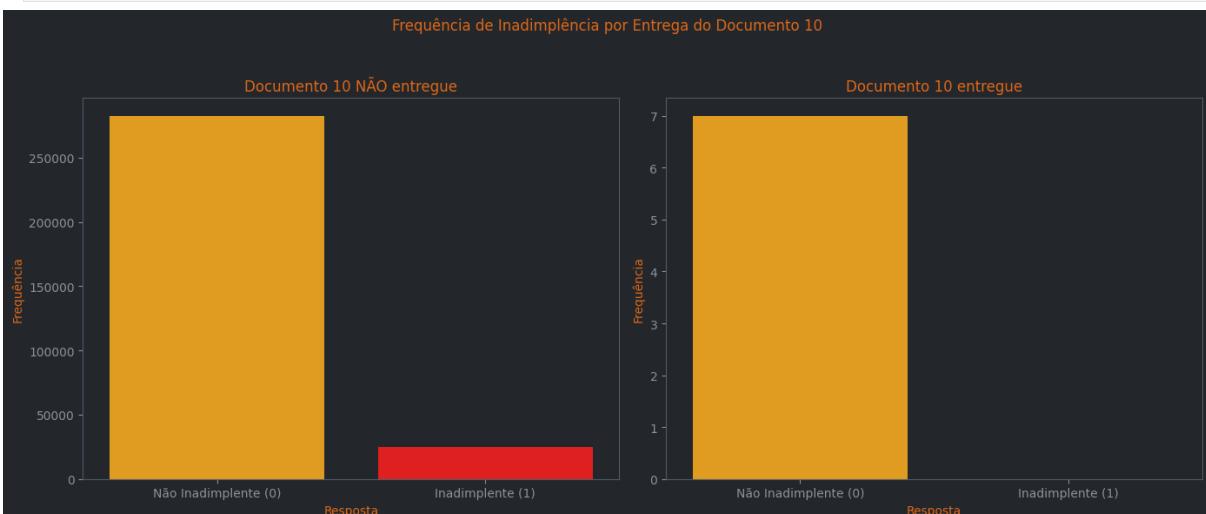
Frequência de Inadimplência por Entrega do Documento 9



V de Cramer entre flag_document_9 e target: 0.0043

flag_document_10

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_10',  
        label_0='Documento 10 NÃO entregue',  
        label_1='Documento 10 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 10'  
)  
  
calcular_cramers_v(df3, 'flag_document_10', 'target')
```

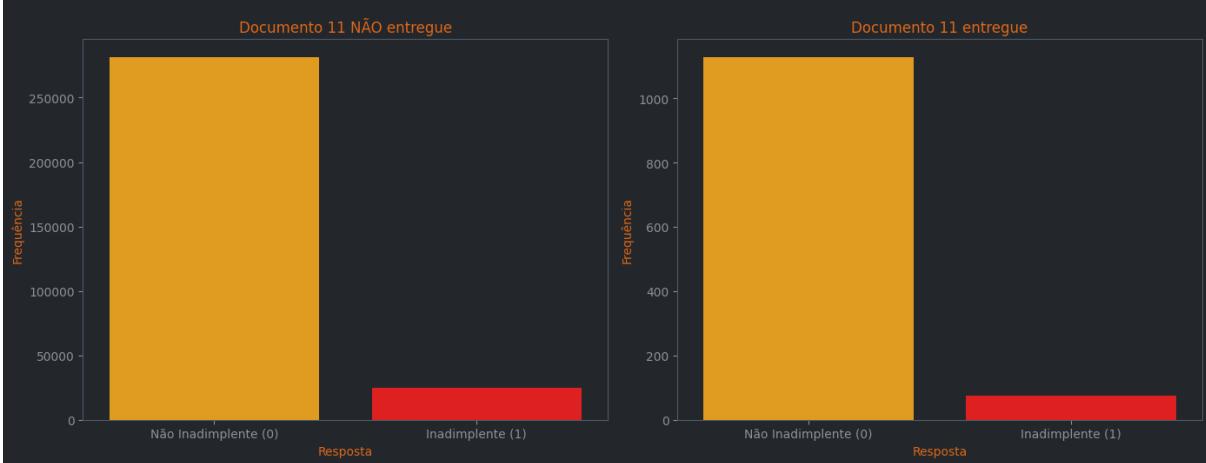


V de Cramer entre flag_document_10 e target: 0.0002

flag_document_11

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_11',  
        label_0='Documento 11 NÃO entregue',  
        label_1='Documento 11 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 11'  
)  
  
calcular_cramers_v(df3, 'flag_document_11', 'target')
```

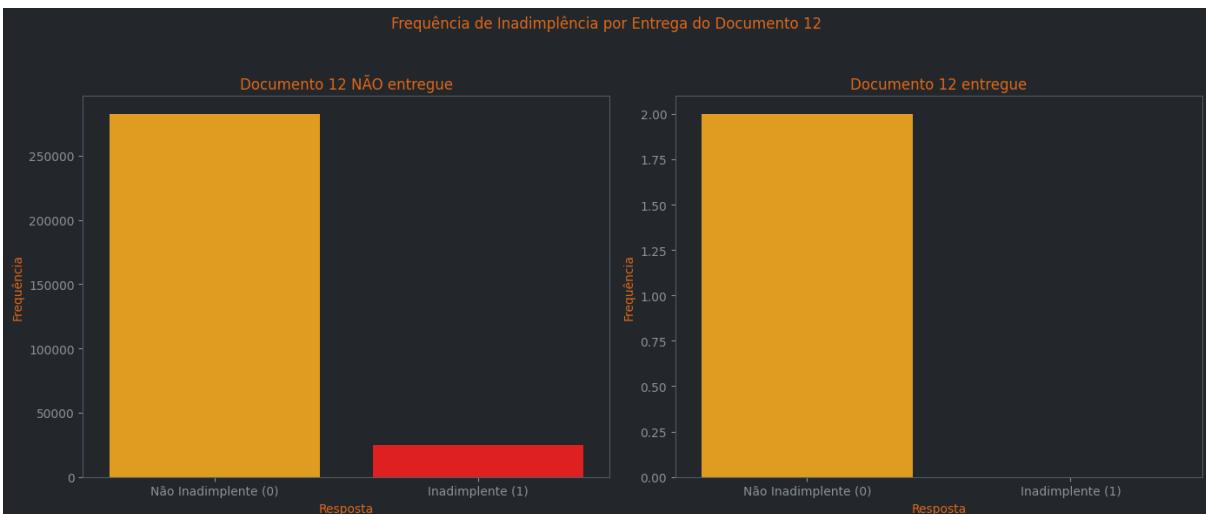
Frequência de Inadimplência por Entrega do Documento 11



V de Cramer entre flag_document_11 e target: 0.0041

flag_document_12

```
In [ ]: plot_binaria_target(  
           df3,  
           var_binaria='flag_document_12',  
           label_0='Documento 12 NÃO entregue',  
           label_1='Documento 12 entregue',  
           suptitle='Frequência de Inadimplência por Entrega do Documento 12'  
)  
  
calcular_cramers_v(df3, 'flag_document_12', 'target')
```

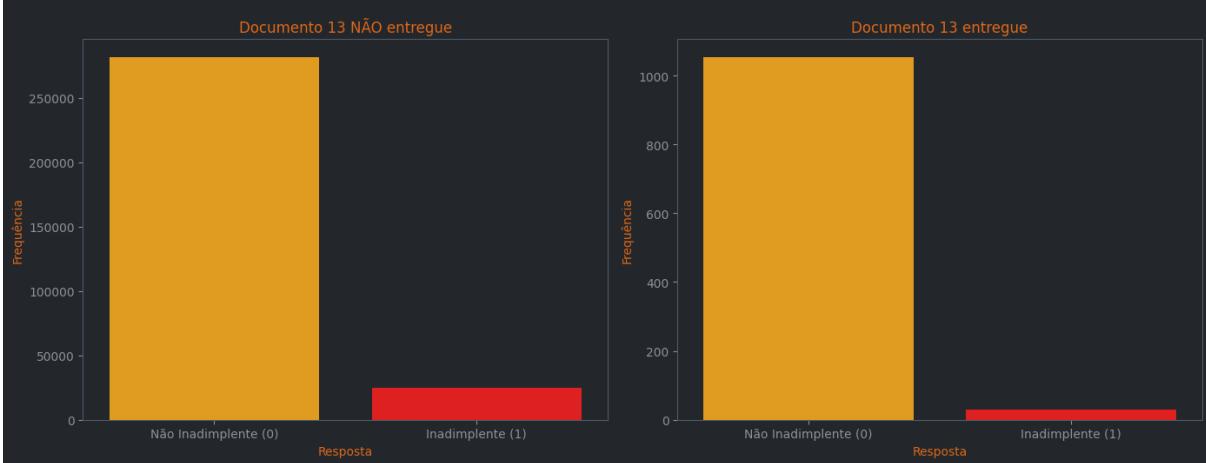


V de Cramer entre flag_document_12 e target: 0.0000

flag_document_13

```
In [ ]: plot_binaria_target(  
           df3,  
           var_binaria='flag_document_13',  
           label_0='Documento 13 NÃO entregue',  
           label_1='Documento 13 entregue',  
           suptitle='Frequência de Inadimplência por Entrega do Documento 13'  
)  
  
calcular_cramers_v(df3, 'flag_document_13', 'target')
```

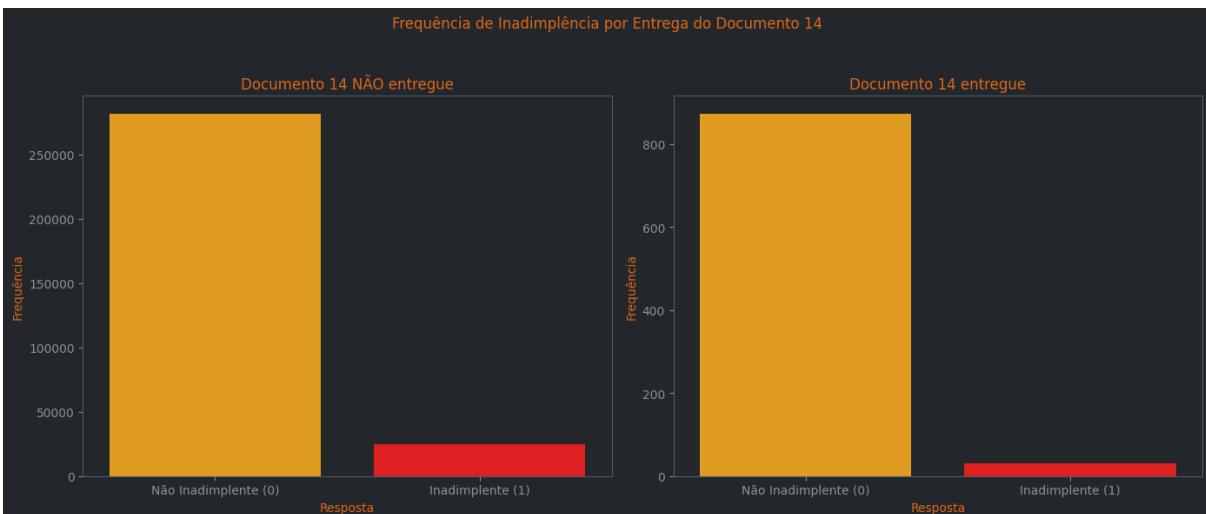
Frequência de Inadimplência por Entrega do Documento 13



V de Cramer entre flag_document_13 e target: 0.0115

flag_document_14

```
In [ ]: plot_binaria_target(  
         df3,  
         var_binaria='flag_document_14',  
         label_0='Documento 14 NÃO entregue',  
         label_1='Documento 14 entregue',  
         suptitle='Frequência de Inadimplência por Entrega do Documento 14'  
)  
calcular_cramers_v(df3, 'flag_document_14', 'target')
```

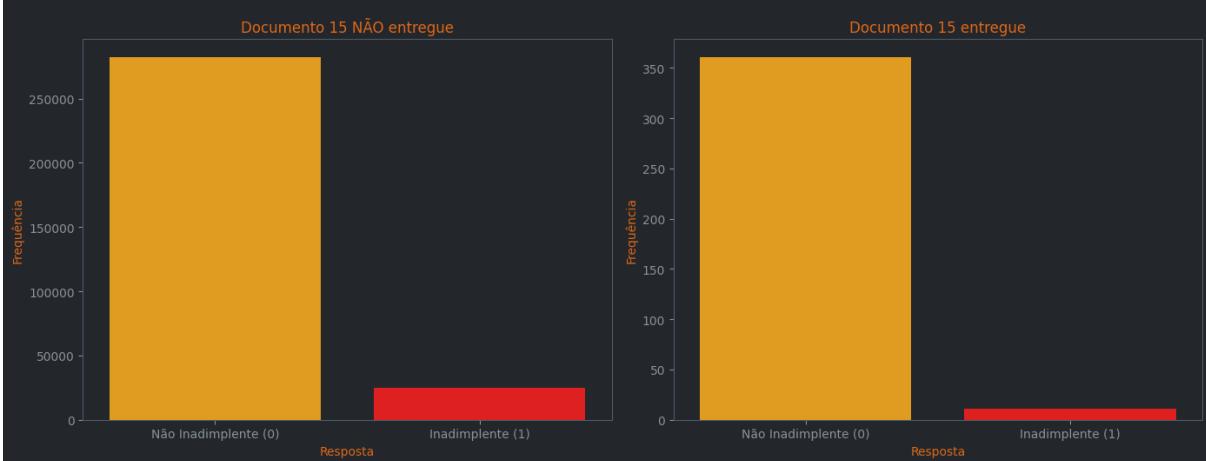


V de Cramer entre flag_document_14 e target: 0.0094

flag_document_15

```
In [ ]: plot_binaria_target(  
         df3,  
         var_binaria='flag_document_15',  
         label_0='Documento 15 NÃO entregue',  
         label_1='Documento 15 entregue',  
         suptitle='Frequência de Inadimplência por Entrega do Documento 15'  
)  
calcular_cramers_v(df3, 'flag_document_15', 'target')
```

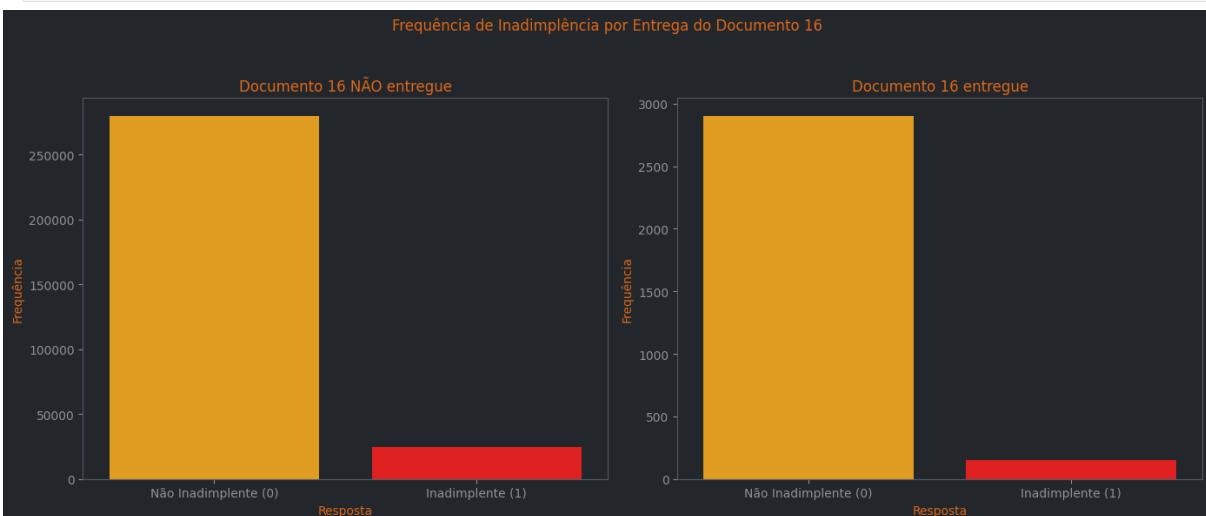
Frequência de Inadimplência por Entrega do Documento 15



V de Cramer entre flag_document_15 e target: 0.0064

flag_document_16

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_16',  
        label_0='Documento 16 NÃO entregue',  
        label_1='Documento 16 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 16'  
)  
  
calcular_cramers_v(df3, 'flag_document_16', 'target')
```

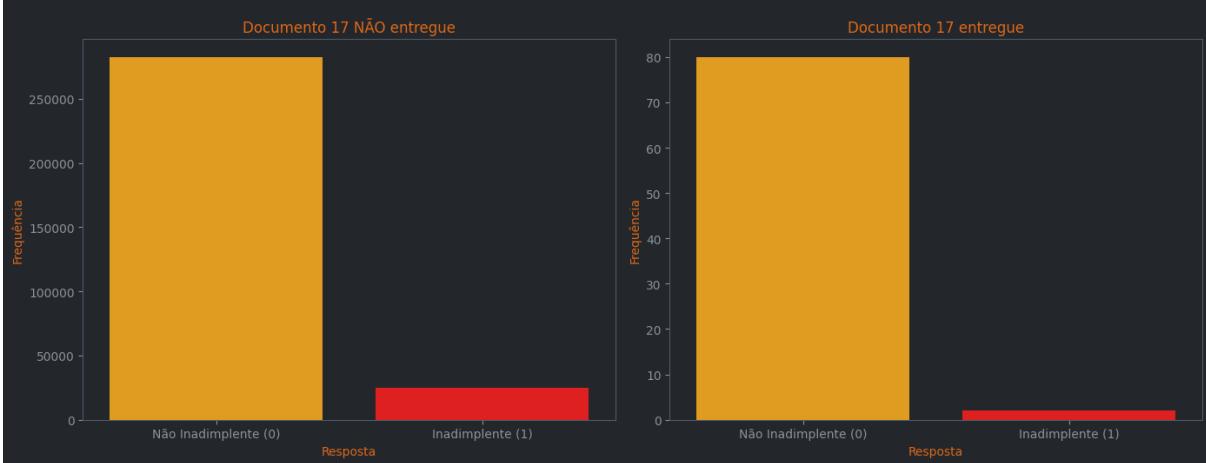


V de Cramer entre flag_document_16 e target: 0.0116

flag_document_17

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_17',  
        label_0='Documento 17 NÃO entregue',  
        label_1='Documento 17 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 17'  
)  
  
calcular_cramers_v(df3, 'flag_document_17', 'target')
```

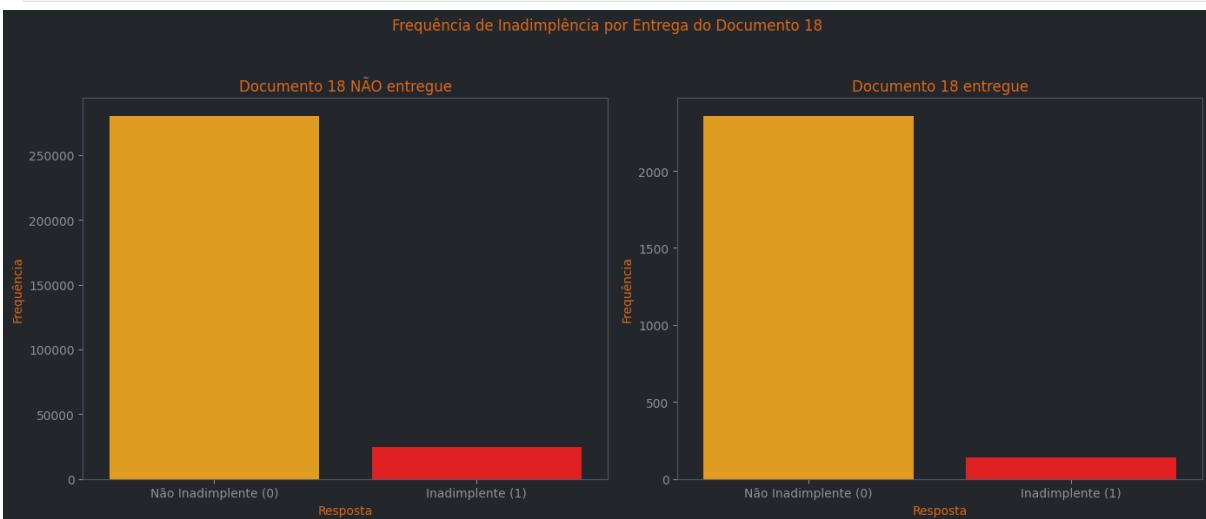
Frequência de Inadimplência por Entrega do Documento 17



V de Cramer entre flag_document_17 e target: 0.0030

flag_document_18

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_18',  
        label_0='Documento 18 NÃO entregue',  
        label_1='Documento 18 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 18'  
)  
  
calcular_cramers_v(df3, 'flag_document_18', 'target')
```

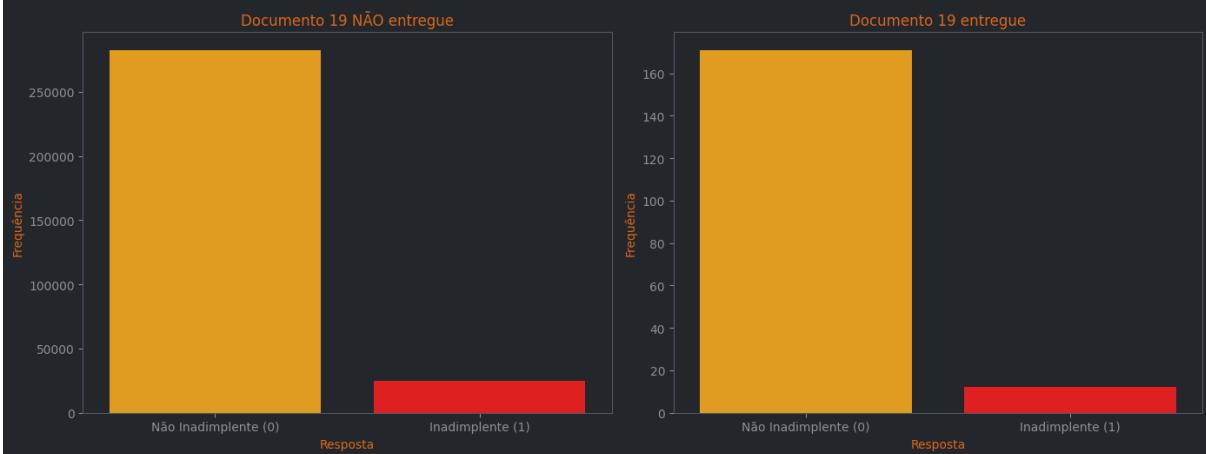


V de Cramer entre flag_document_18 e target: 0.0079

flag_document_19

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='flag_document_19',  
        label_0='Documento 19 NÃO entregue',  
        label_1='Documento 19 entregue',  
        suptitle='Frequência de Inadimplência por Entrega do Documento 19'  
)  
  
calcular_cramers_v(df3, 'flag_document_19', 'target')
```

Frequência de Inadimplência por Entrega do Documento 19



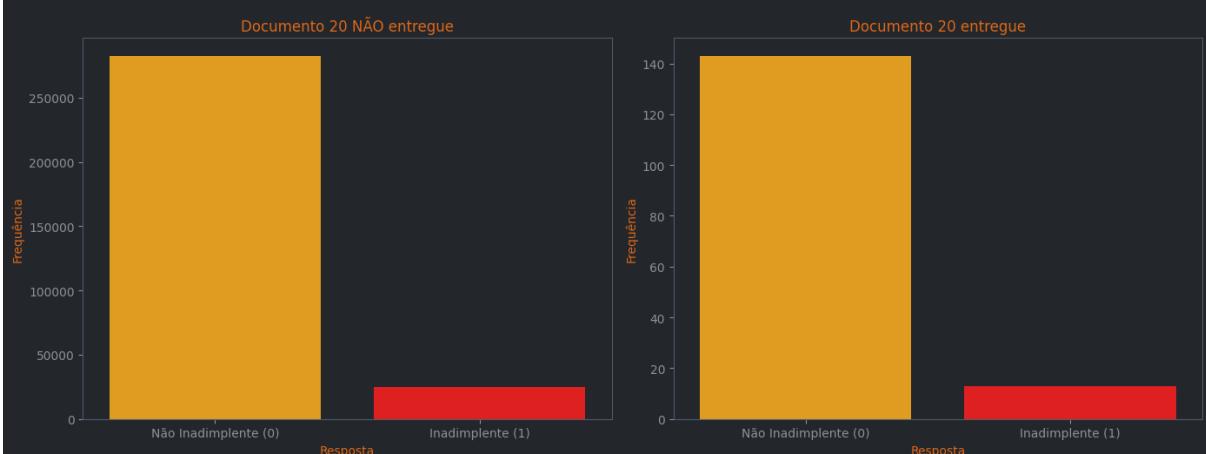
V de Cramer entre flag_document_19 e target: 0.0011

flag_document_20

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_document_20',
    label_0='Documento 20 NÃO entregue',
    label_1='Documento 20 entregue',
    suptitle='Frequência de Inadimplência por Entrega do Documento 20'
)

calcular_cramers_v(df3, 'flag_document_20', 'target')
```

Frequência de Inadimplência por Entrega do Documento 20

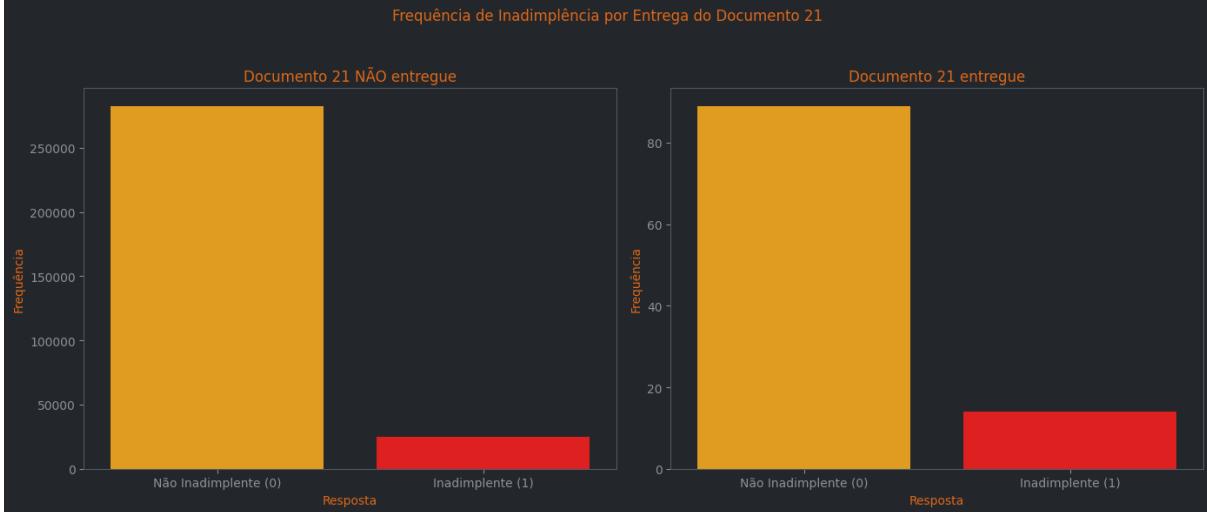


V de Cramer entre flag_document_20 e target: 0.0000

flag_document_21

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='flag_document_21',
    label_0='Documento 21 NÃO entregue',
    label_1='Documento 21 entregue',
    suptitle='Frequência de Inadimplência por Entrega do Documento 21'
)

calcular_cramers_v(df3, 'flag_document_21', 'target')
```



V de Cramer entre flag_document_21 e target: 0.0034

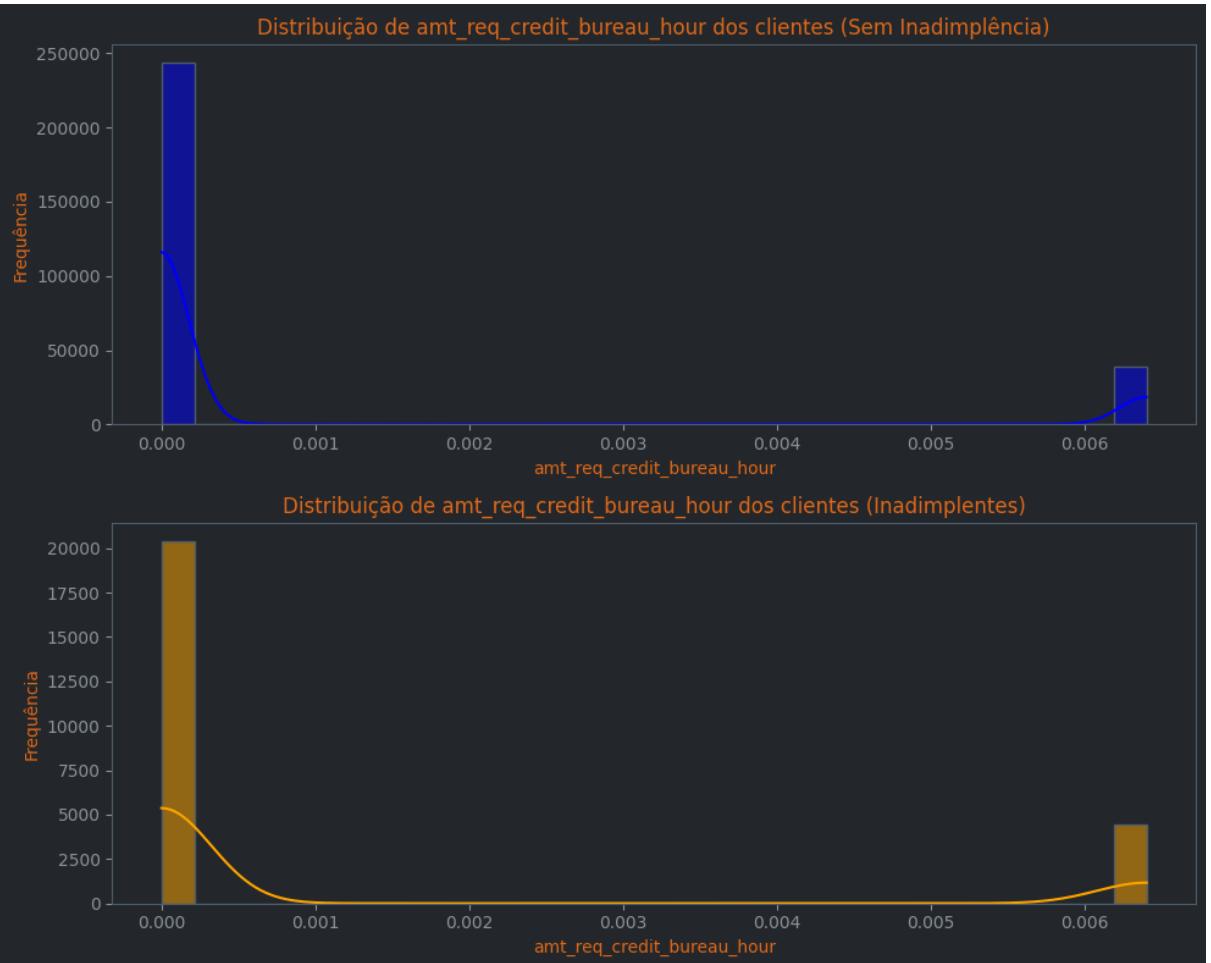
Bloco 4 de Hipóteses

H48 — Histórico de consultas e valores de crédito influenciam a probabilidade de inadimplência.

amt_req_credit_bureau_hour

```
In [ ]: plot_num_var_by_target(
    df3,
    num_var='amt_req_credit_bureau_hour',
    title_0='Distribuição de amt_req_credit_bureau_hour dos clientes (Sem Inadimplência)',
    title_1='Distribuição de amt_req_credit_bureau_hour dos clientes (Inadimplentes)',
    label_x='amt_req_credit_bureau_hour',
    discrete=True
)

# Correlação de amt_req_credit_bureau_hour com a target
pearson_corr = df3['amt_req_credit_bureau_hour'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_req_credit_bureau_hour e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_req_credit_bureau_hour e target: 0.0323

amt_req_credit_bureau_day

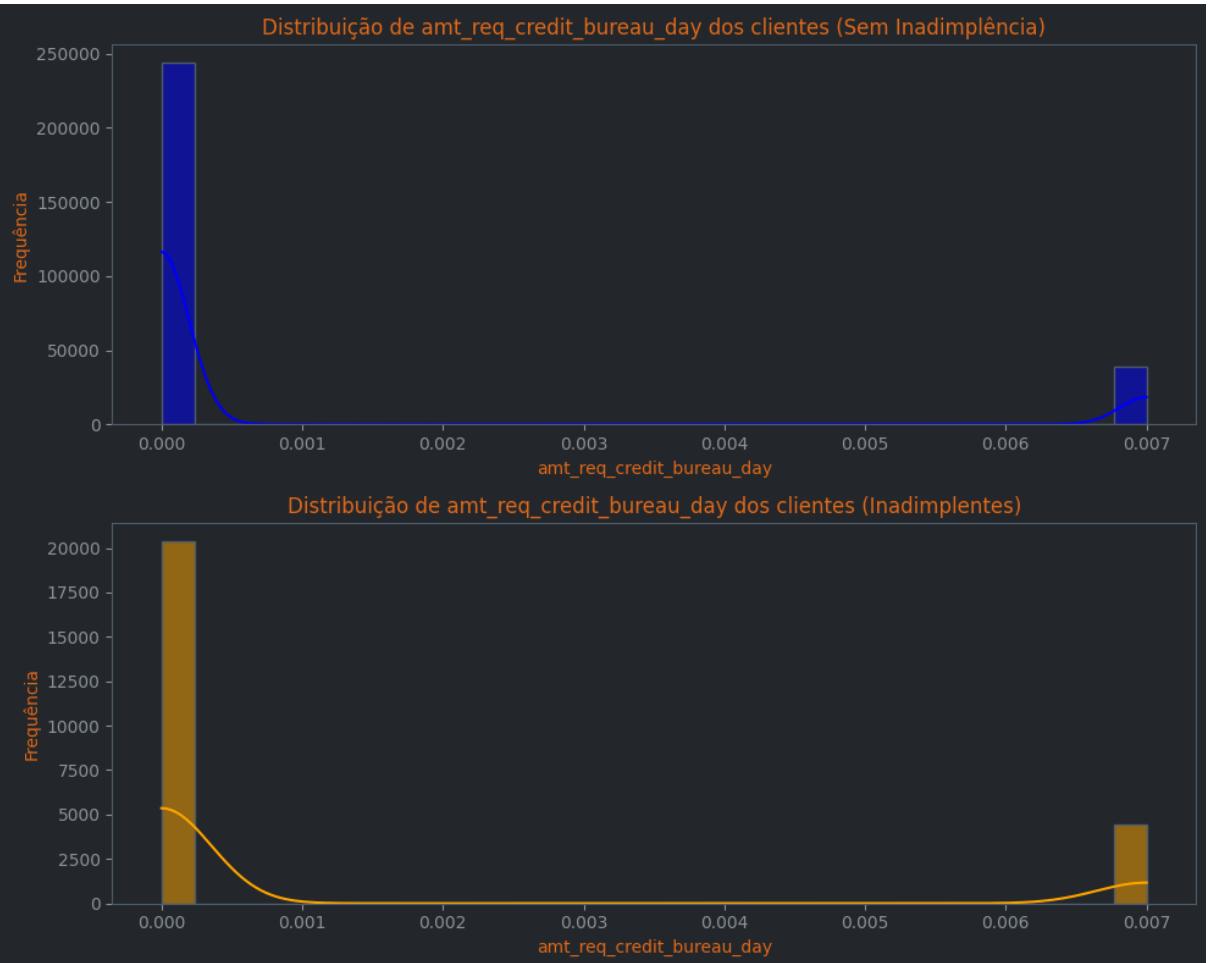
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_req_credit_bureau_day',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_req_credit_bureau_day dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_req_credit_bureau_day')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_req_credit_bureau_day',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_req_credit_bureau_day dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_req_credit_bureau_day')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_req_credit_bureau_day'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_req_credit_bureau_day e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_req_credit_bureau_day e target: 0.0331

amt_req_credit_bureau_week

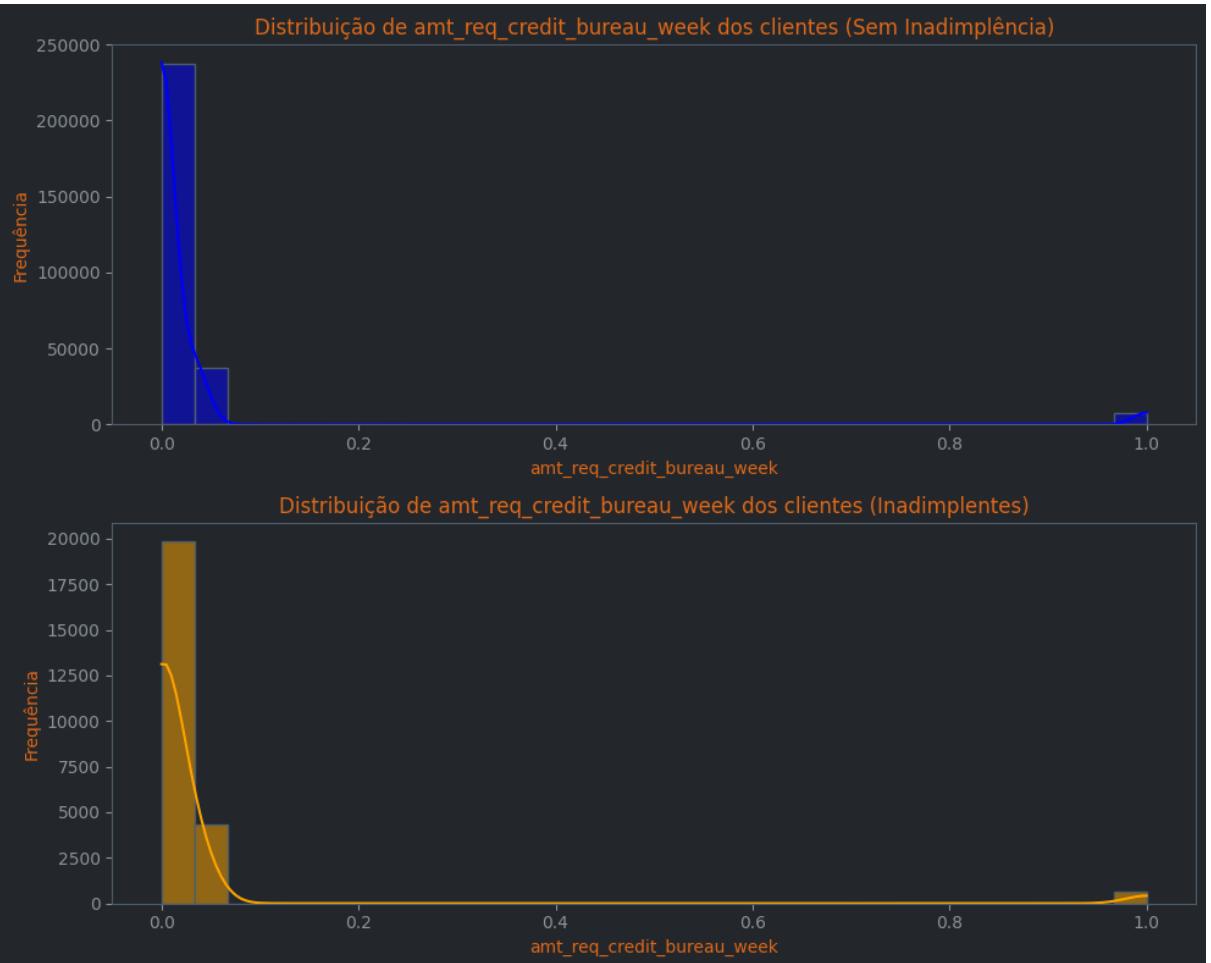
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_req_credit_bureau_week',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_req_credit_bureau_week dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_req_credit_bureau_week')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_req_credit_bureau_week',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_req_credit_bureau_week dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_req_credit_bureau_week')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_req_credit_bureau_week'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_req_credit_bureau_week e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_req_credit_bureau_week e target: 0.0008

amt_req_credit_bureau_mon

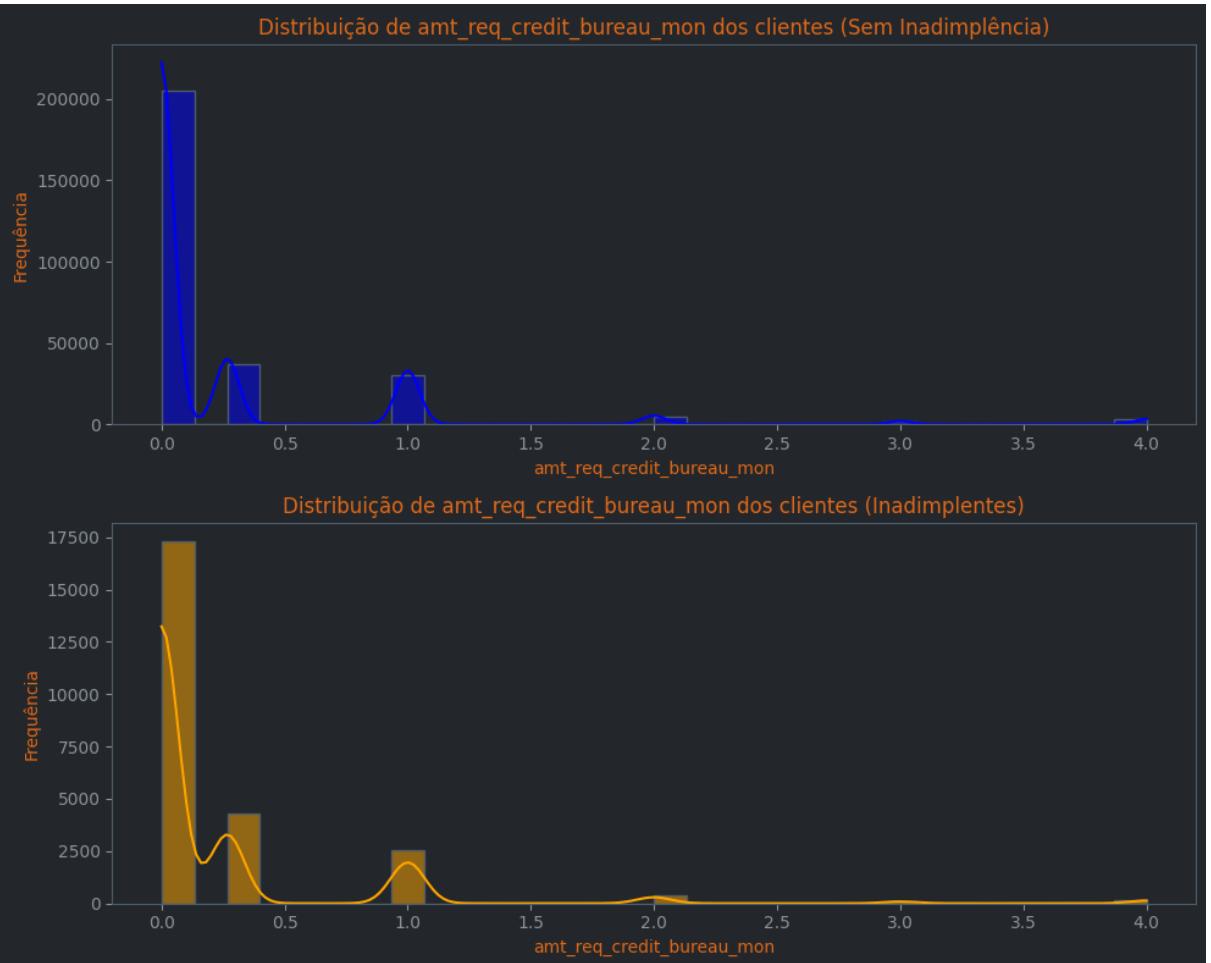
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_req_credit_bureau_mon',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_req_credit_bureau_mon dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_req_credit_bureau_mon')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_req_credit_bureau_mon',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_req_credit_bureau_mon dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_req_credit_bureau_mon')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_req_credit_bureau_mon'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_req_credit_bureau_mon e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_req_credit_bureau_mon e target: -0.0099

amt_req_credit_bureau_qrt

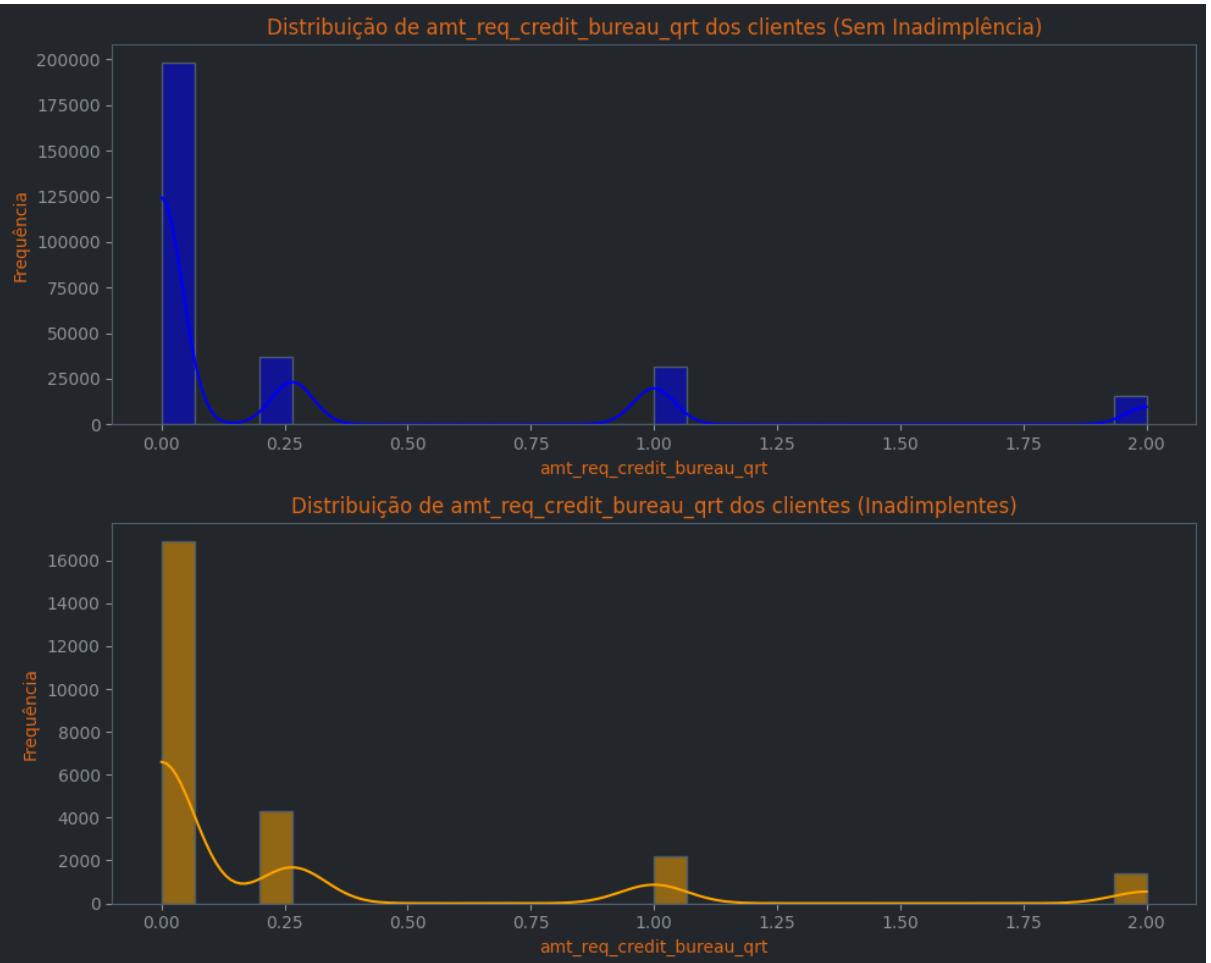
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_req_credit_bureau_qrt',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_req_credit_bureau_qrt dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_req_credit_bureau_qrt')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_req_credit_bureau_qrt',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_req_credit_bureau_qrt dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_req_credit_bureau_qrt')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_req_credit_bureau_qrt'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_req_credit_bureau_qrt e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_req_credit_bureau_qrt e target: -0.0035

amt_req_credit_bureau_year

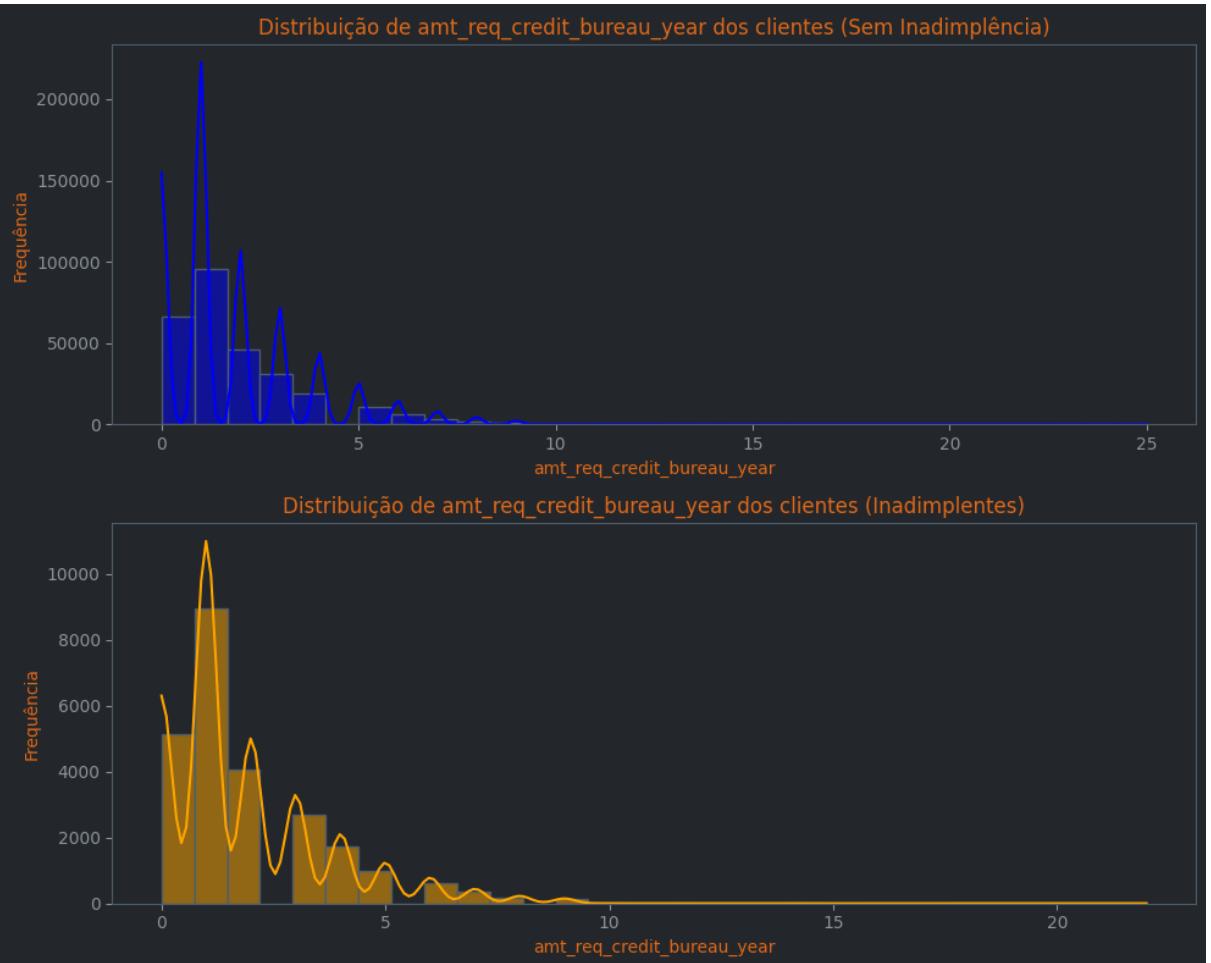
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_req_credit_bureau_year',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_req_credit_bureau_year dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_req_credit_bureau_year')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_req_credit_bureau_year',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_req_credit_bureau_year dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_req_credit_bureau_year')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_req_credit_bureau_year'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_req_credit_bureau_year e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_req_credit_bureau_year e target: 0.0122

amt_credit_sum_mean

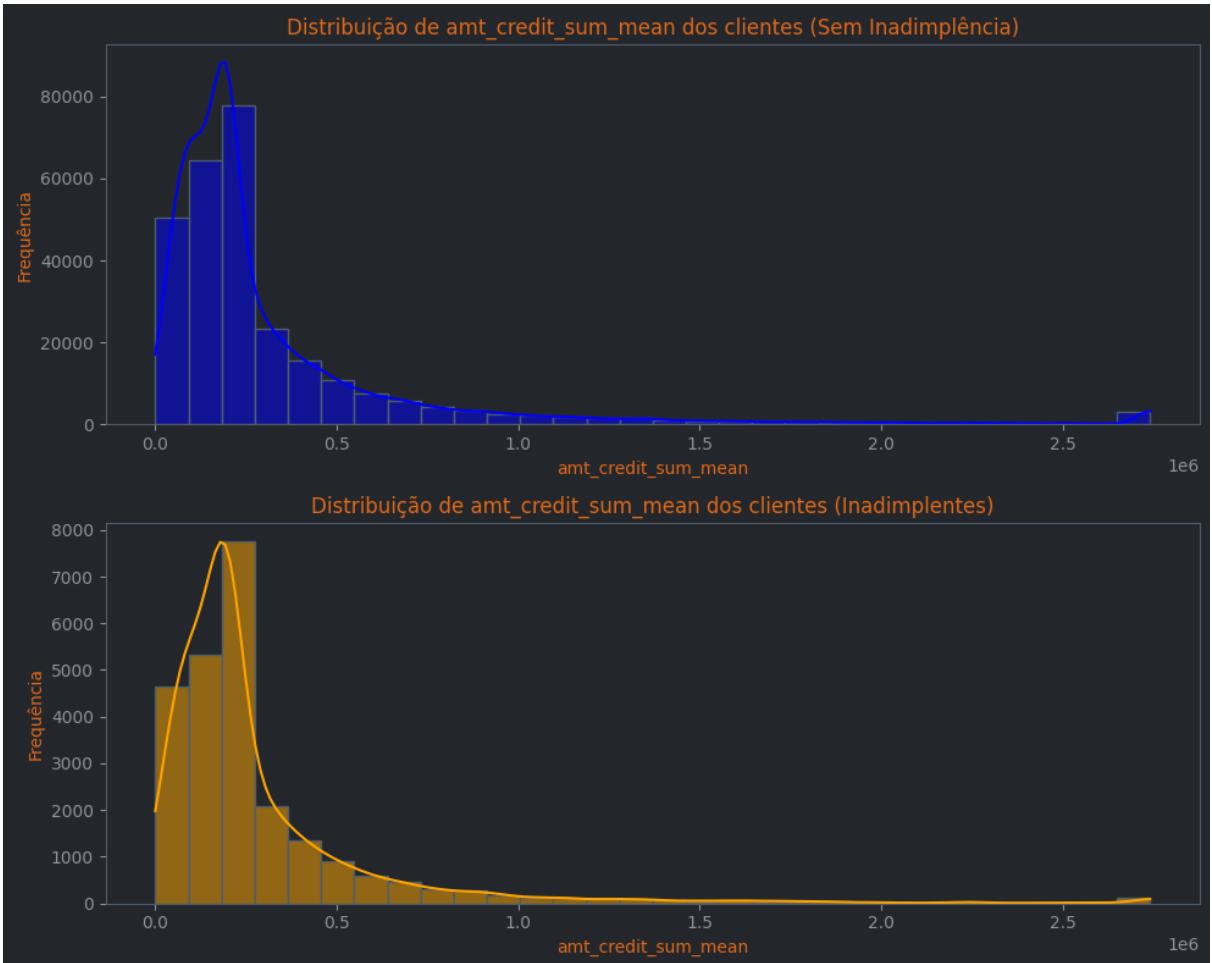
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_credit_sum_mean',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_credit_sum_mean dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_credit_sum_mean')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_credit_sum_mean',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_credit_sum_mean dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_credit_sum_mean')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_credit_sum_mean'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_credit_sum_mean e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_credit_sum_mean e target: -0.0289

amt_credit_sum_mean

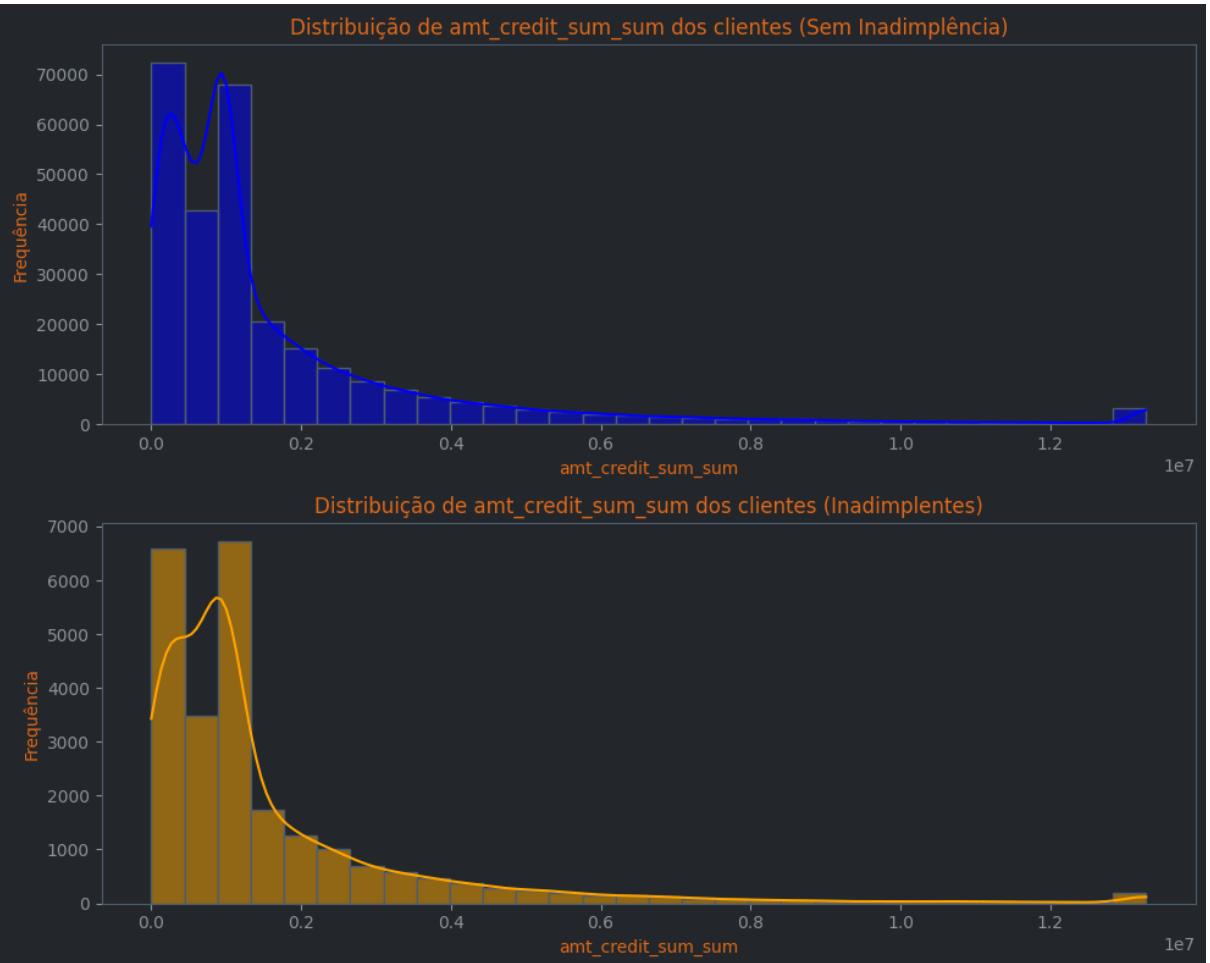
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_credit_sum_mean',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_credit_sum_mean dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_credit_sum_mean')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_credit_sum_mean',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_credit_sum_mean dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_credit_sum_mean')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_credit_sum_mean'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_credit_sum_mean e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_credit_sum_sum e target: -0.0198

credit_active_nunique

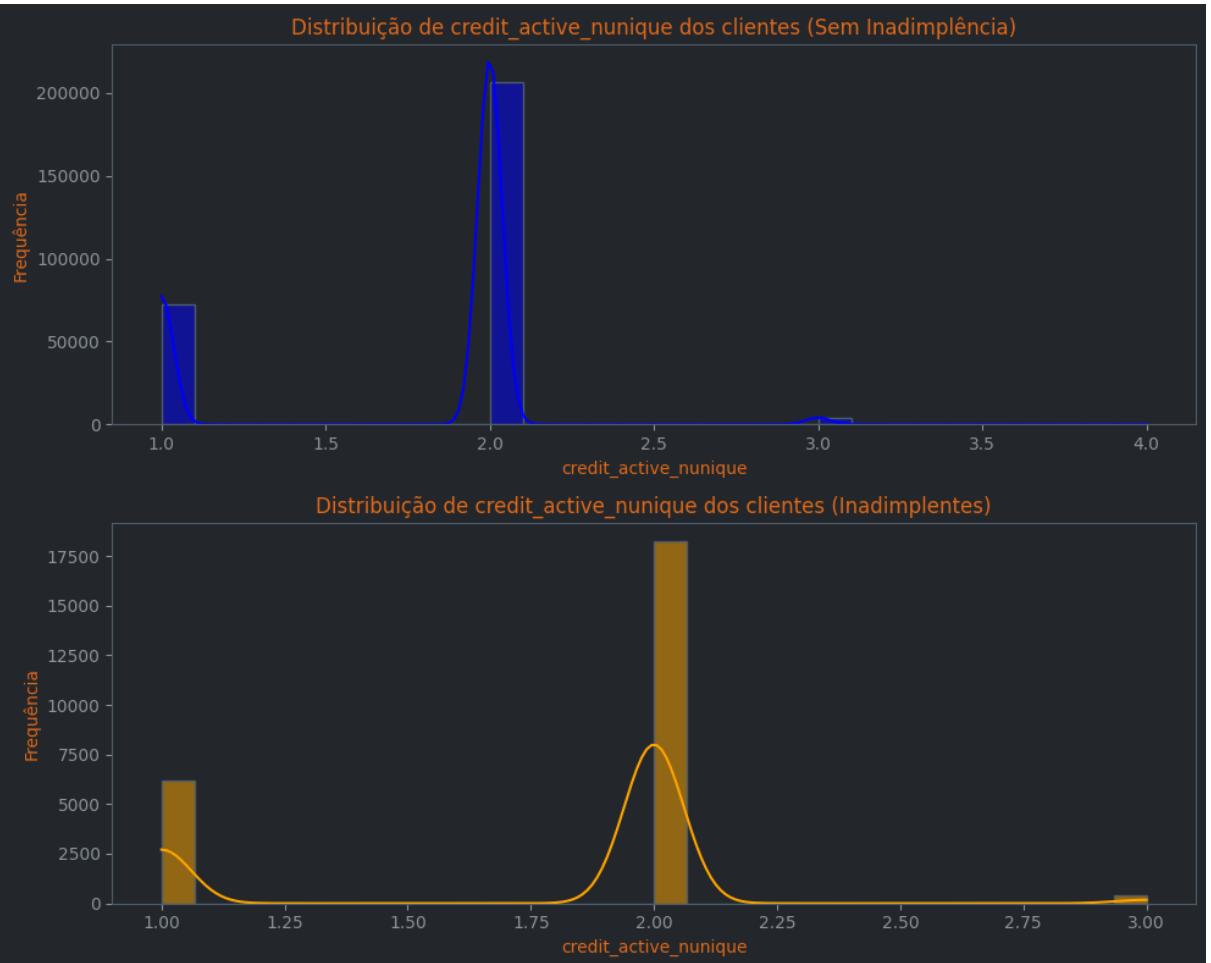
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='credit_active_nunique',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de credit_active_nunique dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('credit_active_nunique')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='credit_active_nunique',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de credit_active_nunique dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('credit_active_nunique')

plt.tight_layout()
plt.show()

pearson_corr = df3['credit_active_nunique'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre credit_active_nunique e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre credit_active_nunique e target: 0.0060

amt_credit_mean

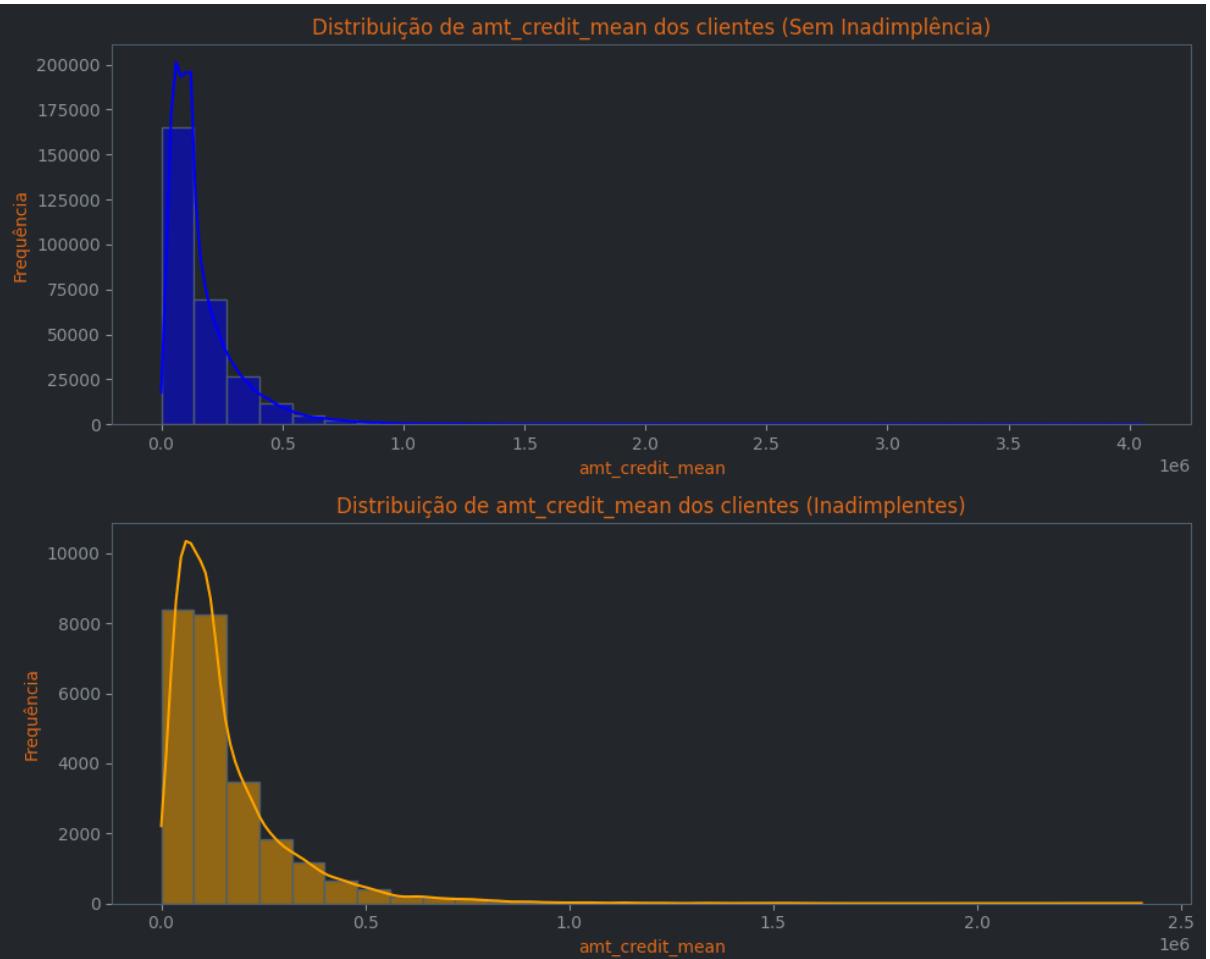
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_credit_mean',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_credit_mean dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_credit_mean')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_credit_mean',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_credit_mean dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_credit_mean')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_credit_mean'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_credit_mean e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_credit_mean e target: -0.0144

amt_credit_sum

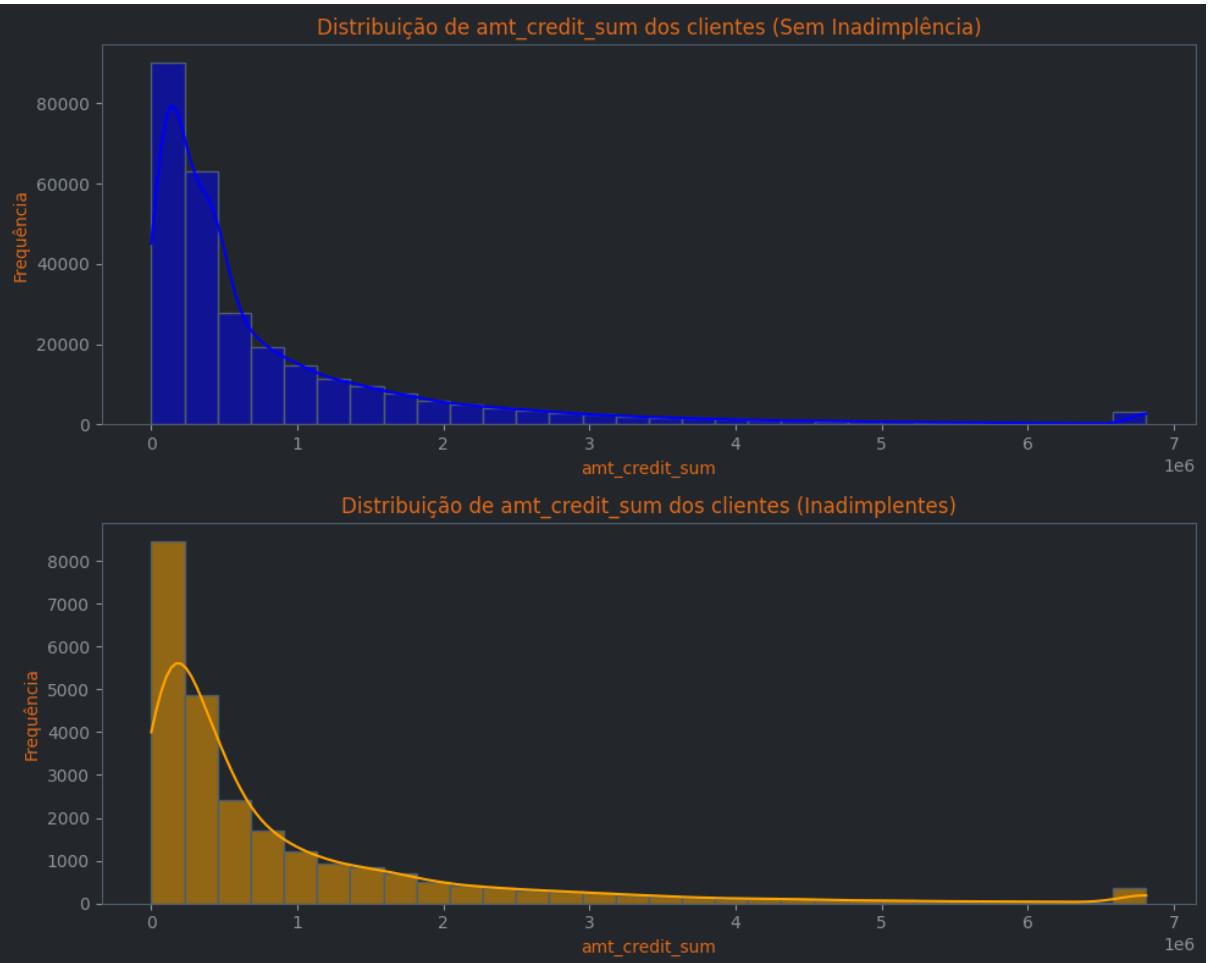
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_credit_sum',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_credit_sum dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_credit_sum')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_credit_sum',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_credit_sum dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_credit_sum')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_credit_sum'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_credit_sum e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_credit_sum e target: 0.0076

amt_credit_max

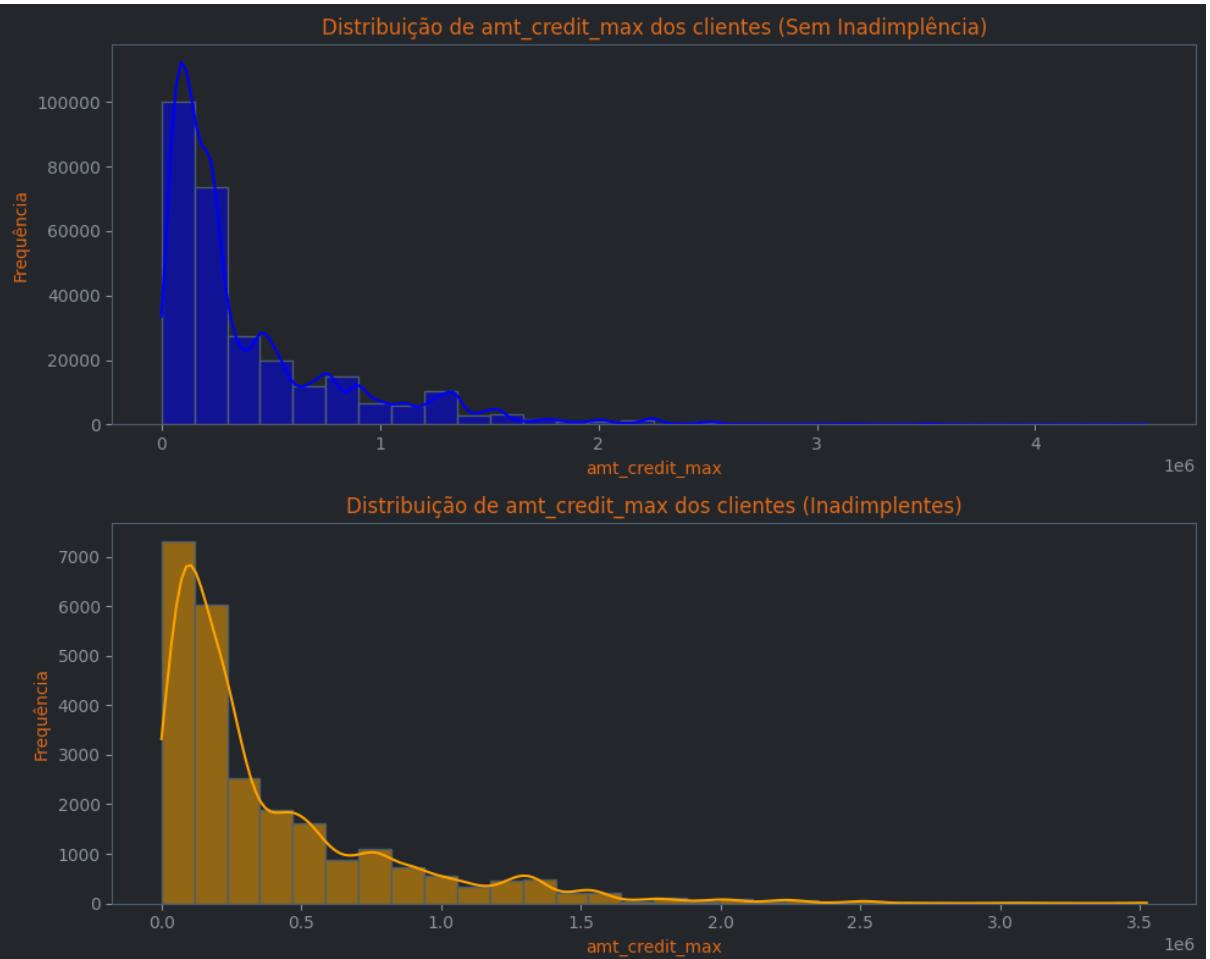
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_credit_max',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_credit_max dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_credit_max')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_credit_max',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_credit_max dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_credit_max')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_credit_max'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_credit_max e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_credit_max e target: -0.0065

amt_application_mean

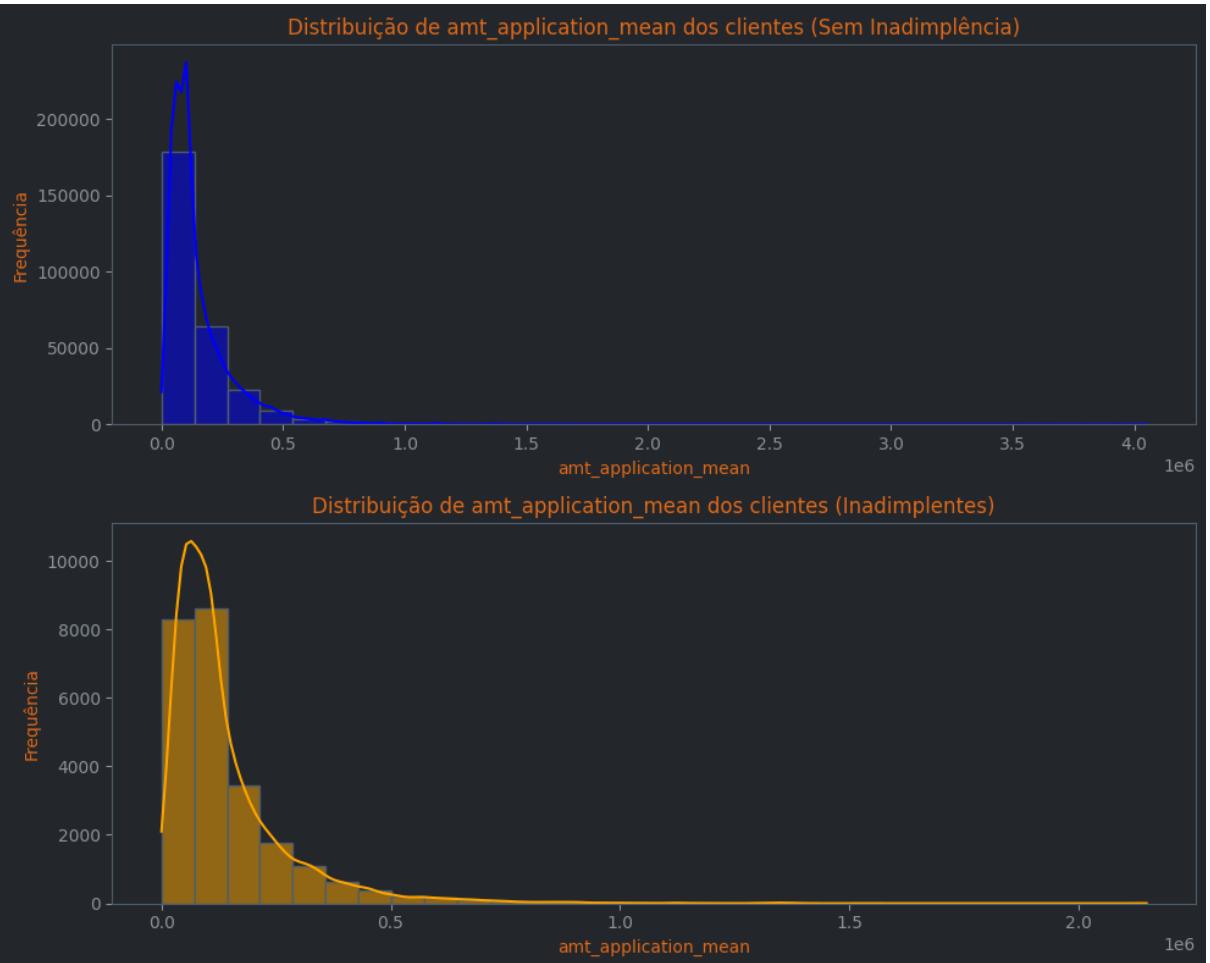
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='amt_application_mean',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de amt_application_mean dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('amt_application_mean')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='amt_application_mean',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de amt_application_mean dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('amt_application_mean')

plt.tight_layout()
plt.show()

pearson_corr = df3['amt_application_mean'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre amt_application_mean e target: {pearson_corr:.4f}")
```



Correlação de Pearson entre amt_application_mean e target: -0.0200

days_decision_mean

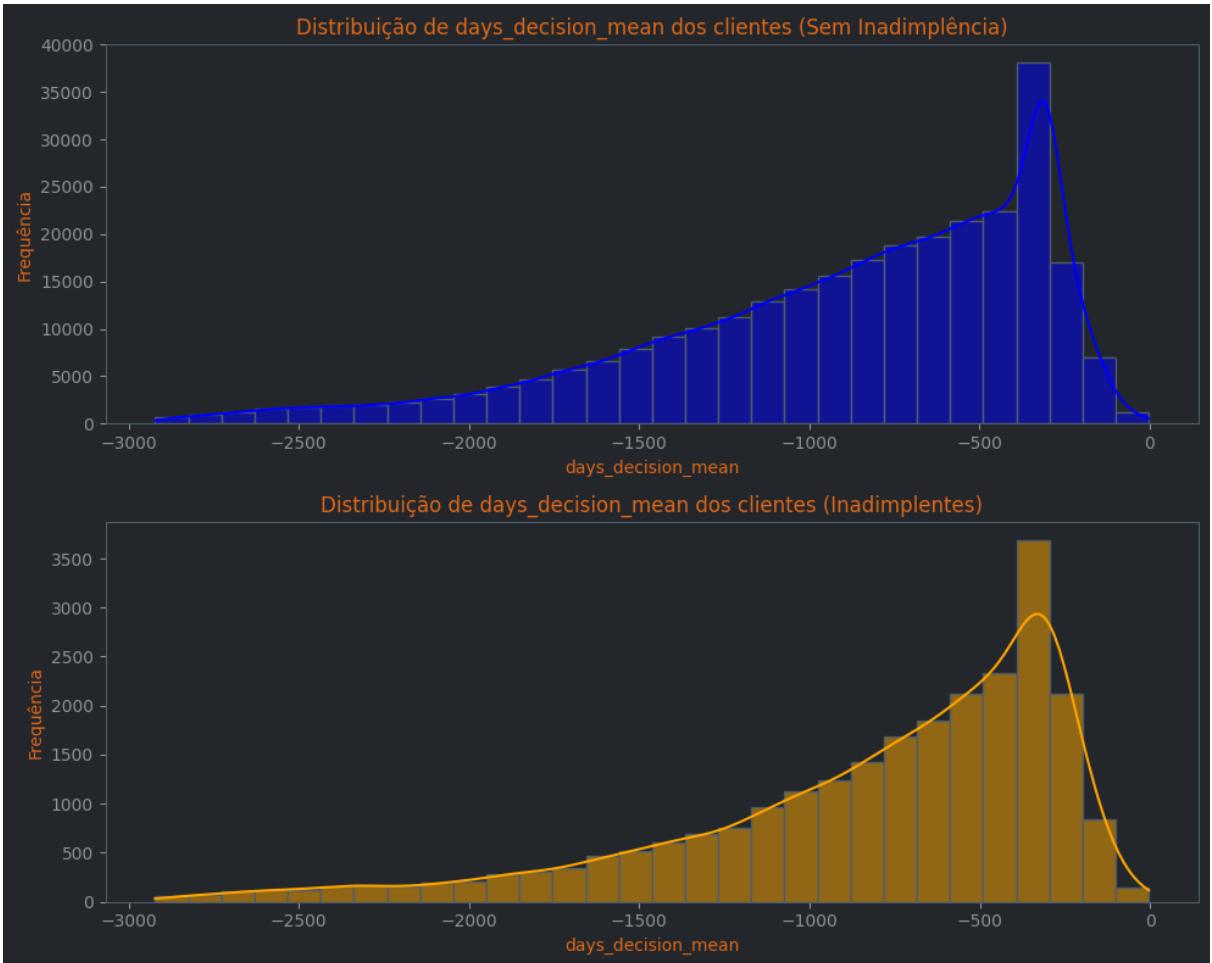
```
In [ ]: fig, axes = plt.subplots(2, 1, figsize=(10, 8), sharey=False)

sns.histplot(
    data=df3[df3['target'] == 0],
    x='days_decision_mean',
    color='blue',
    ax=axes[0],
    kde=True,
    bins=30
)
axes[0].set_title('Distribuição de days_decision_mean dos clientes (Sem Inadimplência)')
axes[0].set_ylabel('Frequência')
axes[0].set_xlabel('days_decision_mean')

sns.histplot(
    data=df3[df3['target'] == 1],
    x='days_decision_mean',
    color='orange',
    ax=axes[1],
    kde=True,
    bins=30
)
axes[1].set_title('Distribuição de days_decision_mean dos clientes (Inadimplentes)')
axes[1].set_ylabel('Frequência')
axes[1].set_xlabel('days_decision_mean')

plt.tight_layout()
plt.show()

pearson_corr = df3['days_decision_mean'].corr(df3['target'], method='pearson')
print(f"Correlação de Pearson entre days_decision_mean e target: {pearson_corr:.4f}")
```



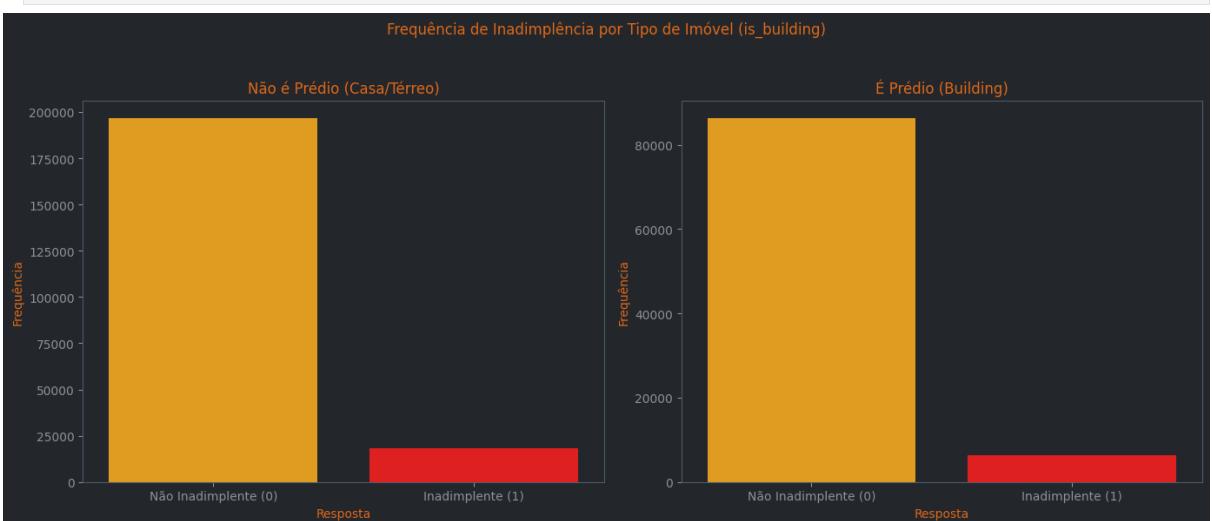
Correlação de Pearson entre days_decision_mean e target: 0.0402

H49 — Características do imóvel e situação de emprego influenciam a capacidade de pagamento.

is_building

```
In [ ]: plot_binaria_target(
    df3,
    var_binaria='is_building',
    label_0='Não é Prédio',
    label_1='É Prédio',
    suptitle='Frequência de Inadimplência por tipo do Imóvel'
)

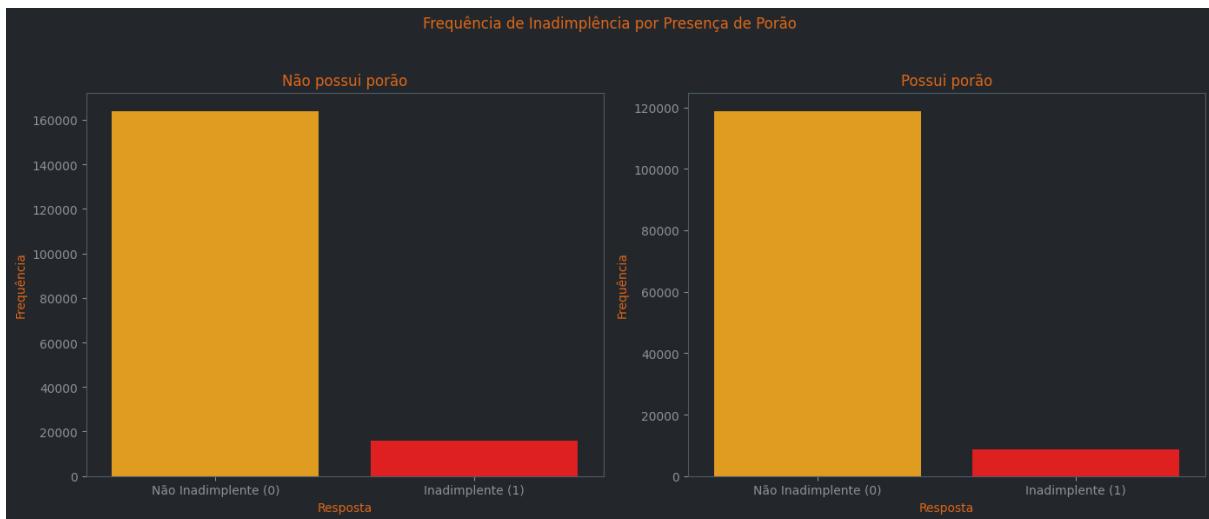
calcular_cramers_v(df3, 'is_building', 'target')
```



V de Cramer entre is_building e target: 0.0280

has_basement

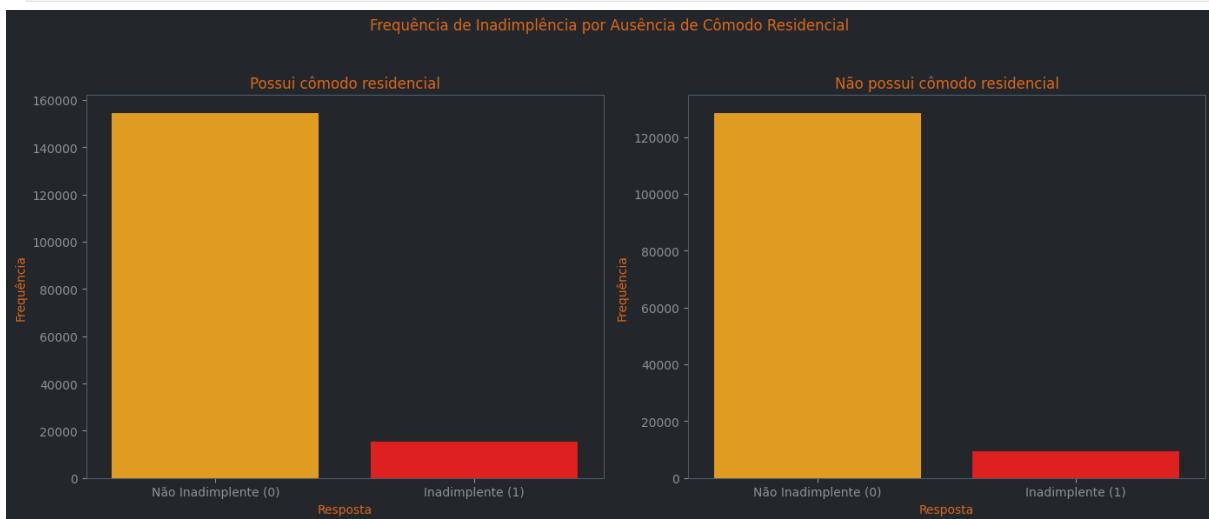
```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='has_basement',  
        label_0='Não possui porão',  
        label_1='Possui porão',  
        suptitle='Frequência de Inadimplência por Presença de Porão'  
)  
  
calcular_cramers_v(df3, 'has_basement', 'target')
```



V de Cramer entre has_basement e target: 0.0366

has_noliving

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='has_noliving',  
        label_0='Possui cômodo residencial',  
        label_1='Não possui cômodo residencial',  
        suptitle='Frequência de Inadimplência por Ausência de Cômodo Residencial'  
)  
  
calcular_cramers_v(df3, 'has_noliving', 'target')
```



V de Cramer entre has_noliving e target: 0.0392

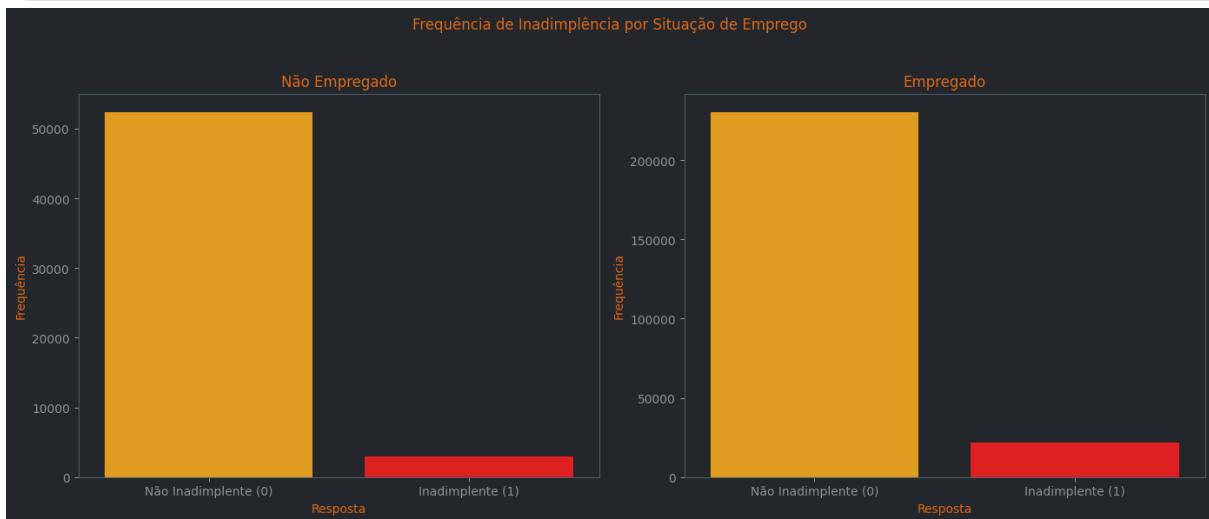
is_employed

```
In [ ]: plot_binaria_target(  
        df3,  
        var_binaria='is_building',  
        label_0='Não Empregado',
```

```

        label_1='Empregado',
        suptitle='Frequência de Inadimplência por Situação de Emprego'
    )
calcular_cramers_v(df3, 'is_employed', 'target')

```



V de Cramer entre is_employed e target: 0.0460

8. Preparação dos Dados

```
In [ ]: df4 = df3.copy()
```

8.1 Escala

8.1.1 Log Transform

```

In [8]: log_transform_cols = [
    'decision_credit_diff',
    'annuity_burden_ratio',
    'credit_efficiency',
]

for col in log_transform_cols:
    df4[col + '_log'] = np.log1p(df4[col])
    df4.drop(columns=col, inplace=True)

```

```
c:\Users\Patryck\AppData\Local\Programs\Python\Python311\Lib\site-packages\pandas\core\arraylike.py:399: RuntimeWarning: invalid value encountered in log1p
    result = getattr(ufunc, method)(*inputs, **kwargs)
```

8.1.2 Robust Scaler

```

In [9]: cols_to_scale = [
    'phone_change_frequency',
    'amt_income_total',
    'amt_credit',
    'num_flag_document',
    'amt_annuity',
    'amt_goods_price',
    'apartments_avg',
    'floorsmax_avg',
    'livingarea_avg',
    'years_build_avg',
    'phone_change_rate',
    'elevators_avg',
    'elevators_mode',
    'elevators_medi',
    'own_car_age',
    'obs_30_cnt_social_circle',
]

```

```

'def_30_cnt_social_circle',
'obs_60_cnt_social_circle',
'def_60_cnt_social_circle',
'amt_req_credit_bureau_hour',
'amt_req_credit_bureau_day',
'amt_req_credit_bureau_week',
'amt_req_credit_bureau_mon',
'amt_req_credit_bureau_qrt',
'amt_req_credit_bureau_year',
'goods_to_credit_ratio',
'basementarea_avg',
'livingapartments_avg',
'landarea_avg',
'nonlivingapartments_avg',
'commonarea_avg',
'years_beginexpluatation_avg',
'nonlivingarea_avg',
'days_employed',
'payment_rate',
'payment_rate^2',
'payment_rate ext_source_mean',
'days_birth^2 payment_rate',
'days_birth payment_rate^2',
'payment_rate^3',
'payment_rate^2 ext_source_mean',
'payment_rate ext_source_mean^2',
]
]

robust_scaler = RobustScaler()
df4[cols_to_scale] = robust_scaler.fit_transform(df4[cols_to_scale])

```

8.1.3 Standard Scaler

```

In [10]: cols_to_standard_scale = [
    'days_registration',
    'days_since_last_employment_until_application',
    'income_per_employed',
    'days_id_publish',
    'days_last_phone_change',
    'recent_instability',
    'employment_stability',
    'employment_life_ratio',
    'cnt_fam_members',
    'goods_to_credit_ratio',
    'days_birth',
    'days_birth^2',
    'days_birth payment_rate',
    'days_birth ext_source_mean',
    'ext_source_mean^2',
    'days_birth^3',
    'days_birth^2 ext_source_mean',
    'days_birth payment_rate ext_source_mean',
    'days_birth ext_source_mean^2',
    'ext_source_mean^3'
]

standard_scaler = StandardScaler()
df4[cols_to_standard_scale] = standard_scaler.fit_transform(df4[cols_to_standard_scale])

```

8.1.4 MinMaxScaler

```

In [11]: cols_to_minmax_scale = [
    'cnt_children',
    'region_population_relative',
    'bureau_to_credit_ratio',
    'bureau_request_intensity',

```

```

        'annuity_to_income_ratio',
        'id_document_age_ratio',
    ]
minmax_scaler = MinMaxScaler()

df4[cols_to_minmax_scale] = minmax_scaler.fit_transform(df4[cols_to_minmax_scale])

```

8.1.5 Scaling Final

```
In [12]: binary_cols = [col for col in df4.columns if sorted(df4[col].dropna().unique()) in [[0, 1], [1, 0]]]

lists_to_exclude = set(
    cols_to_standard_scale +
    cols_to_scale +
    cols_to_minmax_scale +
    binary_cols +
    [col for col in df4.columns if df4[col].dtype == 'object'] +
    ['target']
)

other_columns = [col for col in df4.columns if (col not in lists_to_exclude) and (col not in binary_cols)]

scaler = StandardScaler()
df4[other_columns] = scaler.fit_transform(df4[other_columns])

```

8.2 Encoding

8.2.1 BinaryEncoding

```
In [4]: df4['name_contract_type'] = (df4['name_contract_type'] != 'Cash loans').astype(int)
df4['code_gender'] = df4['code_gender'].map({'F': 0, 'M': 1})
```

8.2.2 OneHotEncoding

```
In [5]: encoder_suite = OneHotEncoder(drop='first', sparse_output=False)
encoded_suite = encoder_suite.fit_transform(df4[['name_type_suite']])
col_names_suite = [f"nts_{cat}" for cat in encoder_suite.categories_[0][1:]]
encoded_df_suite = pd.DataFrame(encoded_suite, columns=col_names_suite, index=df4.index)
df4 = pd.concat([df4.drop(columns=['name_type_suite']), encoded_df_suite], axis=1)

encoder_fam = OneHotEncoder(drop='first', sparse_output=False)
encoded_fam = encoder_fam.fit_transform(df4[['name_family_status']])
col_names_fam = [f"nfs_{cat}" for cat in encoder_fam.categories_[0][1:]]
encoded_df_fam = pd.DataFrame(encoded_fam, columns=col_names_fam, index=df4.index)
df4 = pd.concat([df4.drop(columns=['name_family_status']), encoded_df_fam], axis=1)

encoder_housing = OneHotEncoder(drop='first', sparse_output=False)
encoded_housing = encoder_housing.fit_transform(df4[['name_housing_type']])
col_names_housing = [f"nht_{cat}" for cat in encoder_housing.categories_[0][1:]]
encoded_df_housing = pd.DataFrame(encoded_housing, columns=col_names_housing, index=df4.index)
df4 = pd.concat([df4.drop(columns=['name_housing_type']), encoded_df_housing], axis=1)

encoder_housetype_mode = OneHotEncoder(drop='first', sparse_output=False)
encoded_housetype_mode = encoder_housetype_mode.fit_transform(df4[['housetype_mode']])
col_names_housetype_mode = [f"htm_{cat}" for cat in encoder_housetype_mode.categories_[0][1:]]
encoded_df_housetype_mode = pd.DataFrame(encoded_housetype_mode, columns=col_names_housetype_mode, index=df4.index)
df4 = pd.concat([df4.drop(columns=['housetype_mode']), encoded_df_housetype_mode], axis=1)

encoder_walls = OneHotEncoder(drop='first', sparse_output=False)
encoded_walls = encoder_walls.fit_transform(df4[['wallsmaterial_mode']])
col_names_walls = [f"wm_{cat}" for cat in encoder_walls.categories_[0][1:]]
encoded_df_walls = pd.DataFrame(encoded_walls, columns=col_names_walls, index=df4.index)
df4 = pd.concat([df4.drop(columns=['wallsmaterial_mode']), encoded_df_walls], axis=1)
```

8.2.3 TargetEncoding

```
In [6]: target_mean_income = df4.groupby('name_income_type')['target'].mean()
df4['name_income_type_te'] = df4['name_income_type'].map(target_mean_income)
df4.drop('name_income_type', axis=1, inplace=True)

target_mean_occupation = df4.groupby('occupation_type')['target'].mean()
df4['occupation_type_te'] = df4['occupation_type'].map(target_mean_occupation)
df4.drop('occupation_type', axis=1, inplace=True)

target_mean_organization = df4.groupby('organization_type')['target'].mean()
df4['organization_type_te'] = df4['organization_type'].map(target_mean_organization)
df4.drop('organization_type', axis=1, inplace=True)

target_mean_region_rating_client = df4.groupby('region_rating_client')['target'].mean()
df4['region_rating_client_te'] = df4['region_rating_client'].map(target_mean_region_rating_client)
df4.drop('region_rating_client', axis=1, inplace=True)

target_mean_region_rating_client_w_city = df4.groupby('region_rating_client_w_city')['target'].mean()
df4['region_rating_client_w_city_te'] = df4['region_rating_client_w_city'].map(target_mean_region_rating_client_w_city)
df4.drop('region_rating_client_w_city', axis=1, inplace=True)
```

8.2.4 OrdinalEncoding

```
In [7]: education_order = [
    'Lower secondary',
    'Secondary / secondary special',
    'Incomplete higher',
    'Higher education',
    'Academic degree'
]
df4['name_education_type'] = pd.Categorical(
    df4['name_education_type'],
    categories=education_order,
    ordered=True
).codes
```

9. Feature Selection

```
In [ ]: df5 = df3.copy()

In [4]: df5.columns = [col.replace('/', '_').replace(',', '_').replace(' ', '') for col in df5.columns]
```

9.1 Importância das Variáveis

```
In [5]: X = df5.drop(['target'], axis=1)
y = df5['target']

target = df5['target']
num_neg = (target == 0).sum()
num_pos = (target == 1).sum()
scale_pos_weight = num_neg / num_pos

xgbm_model = xgb.XGBClassifier(
    n_estimators=300,
    random_state=42,
    n_jobs=-1,
    scale_pos_weight=scale_pos_weight,
    eval_metric='logloss'
)
xgbm_model.fit(X, y)

lgbm_model = lgb.LGBMClassifier(
```

```

        n_estimators=300,
        random_state=42,
        n_jobs=1,
        scale_pos_weight=scale_pos_weight
    )
lgbm_model.fit(X, y)

cat_model = CatBoostClassifier(
    iterations=300,
    random_state=42,
    verbose=0,
    scale_pos_weight=scale_pos_weight
)
cat_model.fit(X, y)
shap_sample = X.sample(n=min(1000, len(X)), random_state=42)

explainer_xgb = shap.TreeExplainer(xgbm_model)
shap_values_xgb = explainer_xgb.shap_values(shap_sample)
mean_abs_shap_xgb = np.abs(shap_values_xgb).mean(axis=0)

feat_importance_xgb = pd.DataFrame({
    'Feature': X.columns,
    'SHAP_XGB': mean_abs_shap_xgb
})

explainer_lgb = shap.TreeExplainer(lgbm_model)
shap_values_lgb = explainer_lgb.shap_values(shap_sample)
if isinstance(shap_values_lgb, list):
    shap_values_lgb = shap_values_lgb[1]

mean_abs_shap_lgb = np.abs(shap_values_lgb).mean(axis=0)
feat_importance_lgb = pd.DataFrame({
    'Feature': X.columns,
    'SHAP_LGBM': mean_abs_shap_lgb
})

explainer_cat = shap.TreeExplainer(cat_model)
shap_values_cat = explainer_cat.shap_values(shap_sample)
if isinstance(shap_values_cat, (list, tuple)):
    shap_values_cat = shap_values_cat[1]

mean_abs_shap_cat = np.abs(shap_values_cat).mean(axis=0)
feat_importance_cat = pd.DataFrame({
    'Feature': X.columns,
    'SHAP_CAT': mean_abs_shap_cat
})

feat_importance_df = (
    feat_importance_xgb
        .merge(feat_importance_lgb, on='Feature')
        .merge(feat_importance_cat, on='Feature')
)

feat_importance_df['SHAP_XGB_NORM'] = feat_importance_df['SHAP_XGB'] / feat_importance_df['SHAP_XGB'].sum()
feat_importance_df['SHAP_LGBM_NORM'] = feat_importance_df['SHAP_LGBM'] / feat_importance_df['SHAP_LGBM'].sum()
feat_importance_df['SHAP_CAT_NORM'] = feat_importance_df['SHAP_CAT'] / feat_importance_df['SHAP_CAT'].sum()

feat_importance_df['SHAP_MEAN_NORM'] = feat_importance_df[
    ['SHAP_XGB_NORM', 'SHAP_LGBM_NORM', 'SHAP_CAT_NORM']
].mean(axis=1)

feat_importance_df = feat_importance_df.sort_values(by='SHAP_MEAN_NORM', ascending=False).reset_index()

```

```
[LightGBM] [Info] Number of positive: 24825, number of negative: 282686
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of testing was 5.431486
seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 119911
[LightGBM] [Info] Number of data points in the train set: 307511, number of used features: 1603
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.080729 -> initscore=-2.432486
[LightGBM] [Info] Start training from score -2.432486
c:\Users\Patryck\AppData\Local\Programs\Python\Python311\Lib\site-packages\shap\explainers\_tree.py:587: UserWarning: LightGBM binary classifier with TreeExplainer shap values output has changed to a list of ndarray
warnings.warn(

```

```
In [6]: feat_importance_df.head(50)
```

Out[6]:

	Feature	SHAP_XGB	SHAP_LGBM	SHAP_CAT	SHAP_XGB_NORM	SI
0	ext_source_mean	0.479819	0.136462	0.098005	0.053773	
1	ext_source_mean^3	0.000000	0.234489	0.196949	0.000000	
2	amt_goods_price	0.180001	0.119373	0.168450	0.020172	
3	amt_credit	0.188181	0.099917	0.149596	0.021089	
4	ext_source_mean^2	0.000000	0.128990	0.256225	0.000000	
5	code_gender	0.131131	0.120886	0.131836	0.014696	
6	name_education_type	0.093606	0.084137	0.100661	0.010490	
7	instf_inst_amt_payment_min_mean	0.112430	0.082894	0.070336	0.012600	
8	amt_annuity	0.100947	0.068007	0.084914	0.011313	
9	nfs_Married	0.080827	0.077176	0.077103	0.009058	
10	organization_type_te	0.082617	0.071622	0.078300	0.009259	
11	bur_amt_credit_sum_debt_mean	0.078518	0.064338	0.091906	0.008799	
12	ext_source_1_missing	0.085076	0.064582	0.085060	0.009534	
13	ext_source_weighted_mean	0.069700	0.057586	0.080541	0.007811	
14	instf_inst_amt_payment_min_min	0.071058	0.061853	0.058185	0.007963	
15	occupation_type_te	0.058921	0.054293	0.069931	0.006603	
16	bur_days_credit_enddate_max	0.072368	0.052928	0.056696	0.008110	
17	posf_pos_cnt_instalment_future_mean_mean	0.061260	0.042279	0.080664	0.006865	
18	employment_life_ratio	0.032763	0.033041	0.115123	0.003672	
19	prev_cnt_payment_max	0.038009	0.047791	0.069048	0.004260	
20	bur_days_enddate_fact_max	0.061550	0.043459	0.054840	0.006898	
21	flag_own_car	0.072230	0.059843	0.018257	0.008095	
22	posf_pos_months_balance_max_max	0.061593	0.043903	0.047087	0.006903	
23	num_flag_document	0.056485	0.046360	0.039938	0.006330	
24	bur_days_credit_max	0.056292	0.039222	0.049257	0.006309	
25	annuity_per_employment_year	0.041860	0.040980	0.057281	0.004691	
26	annuity_to_income_ratio	0.055552	0.032860	0.047513	0.006226	
27	instf_inst_days_entry_payment_max_max	0.064776	0.025673	0.046376	0.007259	
28	ext_source_3	0.041073	0.029021	0.058268	0.004603	
29	instf_inst_amt_payment_min_max	0.032871	0.028211	0.066105	0.003684	
30	own_car_age_employed_ratio	0.048990	0.038457	0.030191	0.005490	
31	days_birthext_source_mean^2	0.068693	0.034609	0.017099	0.007698	
32	prev_cnt_payment_mean	0.064248	0.032823	0.017195	0.007200	
33	region_population_relative	0.045858	0.024654	0.043919	0.005139	
34	bur_amt_credit_sum_min	0.046560	0.024092	0.044253	0.005218	
35	amt_application_goods_diff	0.048215	0.025340	0.039803	0.005403	
36	days_id_publish	0.047277	0.039052	0.014773	0.005298	

	Feature	SHAP_XGB	SHAP_LGBM	SHAP_CAT	SHAP_XGB_NORM	SI
37	prev_amt_down_payment_max	0.042961	0.034000	0.024261	0.004815	
38	ext_source_2	0.046125	0.021802	0.040196	0.005169	
39	days_birth	0.049212	0.021245	0.037912	0.005515	
40	instf_inst_days_instalment_max_max	0.050373	0.019795	0.039331	0.005645	
41	flag_document_3	0.035472	0.021595	0.047636	0.003975	
42	posf_pos_cnt_instalment_future_min_mean	0.039702	0.033119	0.024424	0.004449	
43	bur_amt_credit_max_overdue_max	0.053756	0.027124	0.020737	0.006024	
44	phone_change_rate	0.076217	0.014552	0.022423	0.008542	
45	income_per_employed	0.041240	0.009397	0.057745	0.004622	
46	posf_pos_cnt_instalment_future_min_max	0.024989	0.025092	0.044489	0.002800	
47	region_rating_client_w_city_te	0.033190	0.025745	0.035299	0.003720	
48	days_birth^2ext_source_mean	0.038953	0.016967	0.044518	0.004365	
49	days_registration	0.043024	0.019410	0.033100	0.004822	

```
In [7]: print(f"Número de colunas iniciais em X: {df5.shape[1]}")

feat_importance_df_ordenado = feat_importance_df.sort_values(by='SHAP_LGBM_NORM', ascending=False)

cumsum = feat_importance_df_ordenado['SHAP_LGBM_NORM'].cumsum()
n_feat_95 = (cumsum <= 0.95).sum() + 1

selected_features_95 = feat_importance_df_ordenado.loc[:n_feat_95-1, 'Feature'].tolist()

print(f"Quantidade de features para explicar 95% do fenômeno: {len(selected_features_95)}")
print(selected_features_95)

X = df5[selected_features_95]
```

Número de colunas iniciais em X: 1628

Quantidade de features para explicar 95% do fenômeno: 256

```
['ext_source_mean^3', 'ext_source_mean', 'ext_source_mean^2', 'code_gender', 'amt_goods_price', 'amt_credit', 'name_education_type', 'instf_inst_amt_payment_min_mean', 'nfs_Married', 'organization_type_te', 'amt_annuity', 'ext_source_1_missing', 'bur_amt_credit_sum_debt_mean', 'instf_inst_amt_payment_min_min', 'flag_own_car', 'ext_source_weighted_mean', 'occupation_type_te', 'bur_days_credit_enddate_max', 'prev_cnt_payment_max', 'num_flag_document', 'posf_pos_months_balance_max_max', 'bur_days_enddate_fact_max', 'posf_pos_cnt_instalment_future_mean_mean', 'annuity_per_employment_year', 'bur_days_credit_max', 'days_id_publish', 'own_car_age_employed_ratio', 'days_birthext_source_mean^2', 'prev_amt_down_payment_max', 'posf_pos_cnt_instalment_future_min_mean', 'employment_life_ratio', 'annuity_to_income_ratio', 'prev_cnt_payment_mean', 'ext_source_3', 'instf_inst_amt_payment_min_max', 'posf_pos_cnt_instalment_future_mean_max', 'bur_amt_credit_max_overdue_max', 'region_rating_client_w_city_te', 'instf_inst_days_entry_payment_max_max', 'amt_application_goods_diff', 'posf_pos_cnt_instalment_future_min_max', 'region_population_relative', 'bur_amt_credit_sum_min', 'own_car_age', 'ccf_cc_cnt_drawings_atm_current_mean_max', 'ext_source_2', 'flag_document_3', 'def_30_cnt_social_circle', 'amt_req_credit_bureau_qrt', 'days_birth', 'posf_pos_sk_id_curr_mean_count', 'weekday_appr_process_start', 'bur_days_enddate_fact_count', 'instf_inst_days_instalment_max_max', 'bur_days_credit_mean', 'days_registration', 'posf_pos_sk_dpd_def_max_mean', 'prev_rate_down_payment_max', 'ext_source_interaction', 'prev_sellerplace_area_mean', 'bur_amt_credit_sum_limit_count', 'days_last_phone_change', 'ext_source_1', 'days_birth^2ext_source_mean', 'prev_cnt_payment_min', 'bur_amt_credit_sum_max', 'instf_inst_days_entry_payment_count_mean', 'instf_inst_num_instalment_version_mean_mean', 'bur_amt_credit_sun_limit_mean', 'posf_pos_months_balance_mean_max', 'own_car_age_birth_ratio', 'flag_work_phone', 'days_since_last_employment_until_application', 'days_birth^2', 'phone_change_rate', 'days_birthpayment_rate', 'bureau_overdue_ratio', 'posf_pos_cnt_instalment_count_max', 'instf_inst_num_instalment_version_mean_max', 'days_birthext_source_mean', 'ext_source_max', 'reg_city_not_live_city', 'bur_amt_credit_max_overdue_mean', 'posf_pos_sk_dpd_def_mean_max', 'bur_sk_id_bureau_mean_missing', 'def_60_cnt_social_circle', 'instf_inst_num_instalment_number_max_max', 'bur_days_credit_update_max', 'instf_inst_amt_payment_mean_max', 'prev_days_first_drawing_count', 'prev_amt_annuity_mean', 'bur_days_credit_update_mean', 'prev_nflag_insured_on_approval_mean', 'prev_amt_annuity_min', 'name_contract_type', 'decision_credit_diff_log', 'pos_balance_range', 'bur_amt_credit_sum_limit_mean_missing', 'years_beginexpluatation_avg', 'area_quality', 'id_document_ag_ratio', 'goods_to_credit_ratio', 'prev_sellerplace_area_max', 'instf_inst_amt_instalment_min_min', 'ccf_cc_cnt_drawings_atm_current_mean_mean', 'wm_Panel', 'prev_amt_goods_price_mean', 'phone_change_frequency', 'prev_days_last_due_1st_version_max', 'prev_sk_id_prev_max', 'ccf_cc_amt_credit_limit_actual_mean_mean', 'name_income_type_te', 'hour_appr_process_start', 'posf_pos_cnt_instalment_mean_count', 'income_per_employed', 'prev_days_decision_mean', 'income_per_birth', 'posf_pos_cnt_instalment_count_mean', 'bur_amt_credit_sum_debt_max', 'ccf_cc_amt_balance_max_max', 'bur_days_credit_min', 'prev_amt_down_payment_mean', 'id_change_rate', 'instf_inst_num_instalment_version_max_max', 'credit_per_employment_year', 'posf_pos_sk_id_curr_count_max', 'instf_inst_amt_payment_mean_min', 'prev_days_last_due_mean', 'prev_sk_id_prev_count', 'prev_days_decision_max', 'prev_days_decision_min', 'entry_vs_due_ratio', 'bur_days_credit_enddate_mean', 'days_employed', 'ccf_cc_amt_payment_current_mean_mean', 'bur_sk_id_bureau_mean', 'prev_hour_appr_process_start_mean', 'prev_rate_down_payment_mean', 'prev_rate_down_payment_min', 'prev_days_first_due_min', 'flag_phone', 'credit_risk_signal', 'ccf_cc_cnt_drawings_atm_current_mean_min', 'posf_pos_sk_id_curr_count_mean', 'instf_inst_amt_instalment_max_max', 'annuity_per_age', 'bur_days_credit_update_min', 'instf_inst_amt_payment_max_mean', 'credit_efficiency_log', 'posf_pos_sk_id_curr_mean_mean', 'ccf_cc_cnt_drawings_current_max_max', 'prev_sk_id_prev_mean', 'apartments_avg', 'ccf_cc_cnt_drawings_current_mean_max', 'employment_years', 'bur_sk_id_bureau_min', 'prev_hour_appr_process_start_min', 'bur_bb_months_balance_mean_count', 'bureau_to_credit_ratio', 'days_birth^3', 'payment_rate', 'posf_pos_cnt_instalment_count_min', 'instf_inst_amt_instalment_min_max', 'landarea_avg', 'bur_days_credit_enddate_min', 'posf_pos_cnt_instalment_future_max_mean', 'posf_pos_sk_id_curr_count_min', 'bur_amt_credit_sum_limit_max', 'prev_hour_appr_process_start_max', 'bur_days_enddate_fact_min', 'days_birth^2payment_rate', 'posf_pos_cnt_instalment_max_mean', 'instf_inst_days_instalment_max_min', 'bur_amt_credit_sum_mean', 'posf_pos_cnt_instalment_future_count_mean', 'livingarea_avg', 'bur_amt_credit_max_overdue_mean_missing', 'instf_inst_amt_payment_max_max', 'recent_instability', 'floorsmax_avg', 'amt_income_total', 'instf_inst_num_instalment_number_max_mean', 'instf_inst_days_entry_payment_min_mean', 'credit_to_income_ratio', 'bur_sk_id_bureau_max', 'posf_pos_cnt_instalment_min_mean', 'instf_inst_days_instalment_min_min', 'ccf_cc_months_balance_mean_min', 'instf_inst_sk_id_curr_mean_mean', 'prev_amt_goods_price_min', 'posf_pos_months_balance_min_min', 'posf_pos_cnt_instalment_mean_mean', 'posf_pos_cnt_instalment_mean_min', 'years_build_avg', 'posf_pos_months_balance_min_mean', 'instf_inst_sk_id_curr_mean_count', 'ccf_cc_amt_balance_mean_mean', 'instf_inst_days_instalment_min_max', 'prev_days_last_due_1st_version_mean', 'days_birthpayment_rateext_source_mean', 'ccf_cc_amt_receivable_principal_mean_min', 'instf_inst_num_instalment_version_max_mean', 'ext_source_agreement', 'prev_sellerplace_area_min', 'bur_days_enddate_fact_mean', 'payment_rateext_source_mean^2', 'instf_inst_sk_id_curr_count_mean', 'prev_amt_application_mean', 'instf_inst_amt_instalment_min_mean', 'bur_bb_months_balance_count_mean', 'ccf_cc_amt_receivable_principal_mean_mean', 'prev_amt_application_max', 'instf_inst_days_entry_payment_mean_max', 'prev_amt_credit_min', 'prev_amt_a
```

```

nnuity_max', 'prev_amt_annuity_count', 'ext_source_range', 'commonarea_avg', 'bur_bb_months_balance_count_max',
'prev_amt_goods_price_max', 'prev_amt_credit_max', 'amt_application_credit_diff',
'posf_pos_cnt_instalment_min_min', 'annuity_burden_ratio_log', 'ccf_cc_amt_payment_total_current_mean_max',
'ext_source_std', 'bur_bb_months_balance_max_mean', 'prev_days_last_due_1st_version_min',
'ext_source_3_missing', 'ccf_cc_amt_receivable_principal_mean_max', 'bur_amt_credit_sum_debt_min',
'instf_inst_num_instalment_number_mean_min', 'instf_inst_amt_payment_mean_mean',
'prev_amt_application_min', 'ccf_cc_amt_balance_min_mean', 'ccf_cc_amt_balance_mean_max',
'instf_inst_days_instalment_mean_max', 'instf_inst_amt_instalment_max_min', 'days_birthpayment_rate^2',
'totalarea_mode', 'nonlivingarea_avg', 'ccf_cc_amt_credit_limit_actual_min_min',
'prev_sk_id_prev_min', 'bur_amt_credit_sum_debt_count', 'ccf_cc_amt_balance_mean_min',
'prev_days_termination_min', 'nht_Municipalapartment', 'posf_pos_sk_dpd_def_mean_mean',
'bur_sk_id_bureau_count', 'ccf_cc_cnt_drawings_current_mean_min', 'bureau_request_intensity',
'payment_rate^2ext_source_mean', 'ccf_cc_cnt_drawings_current_max_mean', 'prev_days_termination_mean',
'instf_inst_amt_instalment_max_mean', 'livingapartments_avg']

```

10. Machine Learning

```

In [9]: X_train, X_temp, y_train, y_temp = train_test_split(
    X, y, test_size=0.3, random_state=0, stratify=y
)

X_val, X_test, y_val, y_test = train_test_split(
    X_temp, y_temp, test_size=0.5, random_state=0, stratify=y_temp
)

```

10.1 Logistic Regression

```

In [9]: neg, pos = np.bincount(y_train)
scale_pos_weight = neg / pos

logreg = LogisticRegression(
    max_iter=2000,
    class_weight='balanced'
)

In [10]: logreg.fit(X_train, y_train)

df_metricas = avaliar_metricas(logreg, X_val, y_val, X_test, y_test)
display(df_metricas)

```

	validacao	teste
recall	0.997315	0.998120
precision	0.080792	0.080845
f1_score	0.149475	0.149574
auc_roc	0.500389	0.500746
auc_pr	0.080792	0.080845

10.2 Random Forest

```

In [ ]: rf_clf = BalancedRandomForestClassifier(
    n_estimators=1000,
    random_state=42,
    n_jobs=-1
)

In [ ]: rf_clf.fit(X_train, y_train)

df_metricas = avaliar_metricas(rf_clf, X_val, y_val, X_test, y_test)
display(df_metricas)

```

	validacao	teste
recall	0.504834	0.516917
precision	0.215893	0.219950
f1_score	0.302445	0.308592
auc_roc	0.757899	0.768832
auc_pr	0.230295	0.240063

10.3 XGBoost

```
In [10]: num_neg = (y_train == 0).sum()
num_pos = (y_train == 1).sum()
scale_pos_weight = num_neg / num_pos

xgb_clf = xgb.XGBClassifier(
    n_estimators=1000,
    random_state=42,
    scale_pos_weight=scale_pos_weight,
)

In [11]: xgb_clf.fit(X_train, y_train)

df_metricas = avaliar_metricas(xgb_clf, X_val, y_val, X_test, y_test)
display(df_metricas)
```

	validacao	teste
recall	0.209184	0.204082
precision	0.304654	0.292871
f1_score	0.248050	0.240544
auc_roc	0.729740	0.723740
auc_pr	0.217711	0.214192

10.4 LightGBM

```
In [12]: num_neg = (y_train == 0).sum()
num_pos = (y_train == 1).sum()
scale_pos_weight = num_neg / num_pos

lgbm_clf = lgb.LGBMClassifier(
    n_estimators=1000,
    random_state=42,
    scale_pos_weight=scale_pos_weight,
    verbose=-1
)

In [13]: lgbm_clf.fit(X_train, y_train)

df_metricas = avaliar_metricas(lgbm_clf, X_val, y_val, X_test, y_test)
display(df_metricas)
```

	validacao	teste
recall	0.488722	0.485768
precision	0.232024	0.230387
f1_score	0.314661	0.312543
auc_roc	0.764907	0.763319
auc_pr	0.252291	0.250929

10.5 CatBoost

```
In [14]: num_neg = (y_train == 0).sum()
num_pos = (y_train == 1).sum()
scale_pos_weight = num_neg / num_pos

cat_clf = CatBoostClassifier(
    iterations=1000,
    random_state=42,
    verbose=0,
    scale_pos_weight=scale_pos_weight,
)
```

```
In [15]: cat_clf.fit(X_train, y_train)

df_metricas = avaliar_metricas(cat_clf, X_val, y_val, X_test, y_test)
display(df_metricas)
```

	validacao	teste
recall	0.601504	0.594522
precision	0.208391	0.207265
f1_score	0.309542	0.307372
auc_roc	0.774728	0.776648
auc_pr	0.271922	0.267967

10.6 ADABoosting

```
In [ ]: adaboost_clf = AdaBoostClassifier(
    estimator=DecisionTreeClassifier(
        max_depth=1,
        class_weight={0: 1.0, 1: scale_pos_weight}
    ),
    n_estimators=1000,
    random_state=42
)
```

```
In [ ]: adaboost_clf.fit(X_train, y_train)

df_metricas = avaliar_metricas(adaboost_clf, X_val, y_val, X_test, y_test)
display(df_metricas)
```

	validacao	teste
recall	0.656284	0.675081
precision	0.143900	0.147026
f1_score	0.236044	0.241464
auc_roc	0.656692	0.665560
auc_pr	0.122189	0.125486

10.7 HistBoosting

```
In [ ]: histboost_clf = HistGradientBoostingClassifier(
    max_iter=200,
    learning_rate=0.1,
    max_depth=None,
    random_state=42,
    class_weight='balanced'
)

In [ ]: histboost_clf.fit(X_train, y_train)

df_metricas = avaliar_metricas(histboost_clf, X_val, y_val, X_test, y_test)
display(df_metricas)
```

	validacao	teste
recall	0.665951	0.693609
precision	0.182299	0.188595
f1_score	0.286242	0.296556
auc_roc	0.774241	0.783426
auc_pr	0.254090	0.272599

10.8 Nayves Bayes

```
In [ ]: nb_clf = GaussianNB()
nb_clf.fit(X_train, y_train)

df_metricas = avaliar_metricas(nb_clf, X_val, y_val, X_test, y_test)
display(df_metricas)
```

c:\Users\Patryck\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\metrics_classification.py:1731: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 due to no predicted samples. Use `zero_division` parameter to control this behavior.
 _warn_prf(average, modifier, f"{{metric.capitalize()}} is", result.shape[0])
c:\Users\Patryck\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\metrics_classification.py:1731: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 due to no predicted samples. Use `zero_division` parameter to control this behavior.
 _warn_prf(average, modifier, f"{{metric.capitalize()}} is", result.shape[0])

	validacao	teste
recall	0.000000	0.000000
precision	0.000000	0.000000
f1_score	0.000000	0.000000
auc_roc	0.500378	0.500664
auc_pr	0.080792	0.080839

11. HyperTuning

11.1 BayesSearch

```
In [ ]: X_full = pd.concat([X_train, X_val]).reset_index(drop=True)
y_full = pd.concat([y_train, y_val]).reset_index(drop=True)

search_spaces = {
    'n_estimators': Integer(2000, 6000),
    'learning_rate': Real(0.0005, 0.1, prior='log-uniform'),
    'max_depth': Integer(3, 12),
    'min_child_weight': Integer(1, 30),
    'gamma': Real(0, 5.0),
    'reg_alpha': Real(1e-8, 50, prior='log-uniform'),
    'reg_lambda': Real(1e-6, 300, prior='log-uniform'),
    'subsample': Real(0.5, 1.0),
    'colsample_bytree': Real(0.4, 1.0),
    'colsample_bylevel': Real(0.3, 1.0),
    'colsample_bynode': Real(0.3, 1.0),
    'booster': Categorical(['gbtree']),
    'tree_method': Categorical(['hist', 'exact']),
    'grow_policy': Categorical(['depthwise', 'lossguide']),
    'scale_pos_weight': Integer(1, 40),
    'max_delta_step': Integer(0, 10),
}

model = xgb.XGBClassifier(
    random_state=42,
    eval_metric='auc',
    tree_method='hist'
)

bayes = BayesSearchCV(
    estimator=model,
    search_spaces=search_spaces,
    scoring='roc_auc',
    n_iter=120,
    cv=3,
    n_jobs=-1,
```

```

        verbose=3,
        random_state=42,
        refit=True,
        optimizer_kwargs={'acq_func': 'EI'}
    )

bayes.fit(
    X_full,
    y_full,
    callback=None,
    **{
        "eval_set": [(X_val, y_val)],
        "early_stopping_rounds": 50,
        "verbose": False
    }
)

print("\n👉 Melhor score (roc_auc):", bayes.best_score_)
print("🔧 Melhores parâmetros encontrados:\n")
for p, v in bayes.best_params_.items():
    print(f"  {p}: {v}")

```

12. Treinando o modelo final

```

In [10]: xgb_params = {
    'booster': 'gbtree',
    'n_estimators': 5116,
    'learning_rate': 0.006466,
    'max_depth': 6,
    'subsample': 0.83743,
    'colsample_bytree': 0.921139,
    'colsample_bylevel': 0.674959,
    'gamma': 2.165815,
    'min_child_weight': 12,
    'reg_alpha': 0.000102,
    'reg_lambda': 17.3739,
    'scale_pos_weight': 1,
    'random_state': 42,
}

X_trainval = np.concatenate([X_train, X_val])
y_trainval = np.concatenate([y_train, y_val])

xgb_final = xgb.XGBClassifier(**xgb_params)
xgb_final.fit(X_trainval, y_trainval)

y_pred_proba = xgb_final.predict_proba(X_test)[:, 1]
y_pred = xgb_final.predict(X_test)

print("Classification Report (TESTE):")
print(classification_report(y_test, y_pred))

print("Matriz de Confusão (TESTE):")
print(confusion_matrix(y_test, y_pred))

print("\nAUC-ROC:", roc_auc_score(y_test, y_pred_proba))
print("AUC-PR:", average_precision_score(y_test, y_pred_proba))
print("Recall:", recall_score(y_test, y_pred))
print("Precision:", precision_score(y_test, y_pred))
print("F1 Score:", f1_score(y_test, y_pred))

```

```

Classification Report (TESTE):
      precision    recall  f1-score   support

          0       0.92     1.00      0.96     42403
          1       0.57     0.04      0.08      3724

   accuracy                           0.92     46127
macro avg       0.75     0.52      0.52     46127
weighted avg    0.89     0.92      0.89     46127

```

Matriz de Confusão (TESTE):

```

[[42290  113]
 [ 3574  150]]

```

AUC-ROC: 0.795399200495334

AUC-PR: 0.2904329200889078

Recall: 0.040279269602577876

Precision: 0.5703422053231939

F1 Score: 0.07524454477050414

```

In [11]: best_threshold, best_f1 = find_best_threshold(y_test, y_pred_proba, metric="f1")
y_pred_best = (y_pred_proba >= best_threshold).astype(int)

print(f"Melhor threshold para F1: {best_threshold:.4f} (F1: {best_f1:.4f})\n")

print("Classification Report (TESTE):")
print(classification_report(y_test, y_pred_best))

print("Matriz de Confusão (TESTE):")
print(confusion_matrix(y_test, y_pred_best))

print("\nAUC-ROC:", roc_auc_score(y_test, y_pred_proba))
print("AUC-PR:", average_precision_score(y_test, y_pred_proba))
print("Recall:", recall_score(y_test, y_pred_best))
print("Precision:", precision_score(y_test, y_pred_best))
print("F1 Score:", f1_score(y_test, y_pred_best))

```

Melhor threshold para F1: 0.1910 (F1: 0.3437)

```

Classification Report (TESTE):
      precision    recall  f1-score   support

          0       0.94     0.93      0.94     42403
          1       0.31     0.38      0.34      3724

   accuracy                           0.88     46127
macro avg       0.63     0.65      0.64     46127
weighted avg    0.89     0.88      0.89     46127

```

Matriz de Confusão (TESTE):

```

[[39255  3148]
 [ 2298  1426]]

```

AUC-ROC: 0.795399200495334

AUC-PR: 0.2904329200889078

Recall: 0.382921589688507

Precision: 0.31176213379973766

F1 Score: 0.34369727645215714

12.2 Treinando com Todos os Dados

```

In [13]: xgb_params = {
    'booster': 'gbtree',
    'n_estimators': 5116,
    'learning_rate': 0.006466,
    'max_depth': 6,
    'subsample': 0.83743,
    'colsample_bytree': 0.921139,
}

```

```
'colsample_bytree': 0.674959,
'gamma': 2.165815,
'min_child_weight': 12,
'reg_alpha': 0.000102,
'reg_lambda': 17.3739,
'scale_pos_weight': 1,
'random_state': 42,
}

xgb_final = xgb.XGBClassifier(**xgb_params)

xgb_final.fit(X, y)
```

Out[13]:

XGBClassifier		
► Parameters		
objective	'binary:logistic'	
base_score	None	
booster	'gbtree'	
callbacks	None	
colsample_bylevel	0.674959	
colsample_bynode	None	
colsample_bytree	0.921139	
device	None	
early_stopping_rounds	None	
enable_categorical	False	
eval_metric	None	
feature_types	None	
feature_weights	None	
gamma	2.165815	
grow_policy	None	
importance_type	None	
interaction_constraints	None	
learning_rate	0.006466	
max_bin	None	
max_cat_threshold	None	
max_cat_to_onehot	None	
max_delta_step	None	
max_depth	6	
max_leaves	None	
min_child_weight	12	
missing	nan	
monotone_constraints	None	
multi_strategy	None	
n_estimators	5116	
n_jobs	None	
num_parallel_tree	None	
random_state	42	
reg_alpha	0.000102	
reg_lambda	17.3739	
sampling_method	None	
scale_pos_weight	1	

	subsample	0.83743
	tree_method	None
	validate_parameters	None
	verbosity	None

13. Tradução e Interpretação do Erro

```
In [12]: best_threshold, best_f1 = find_best_threshold(y_test, y_pred_proba, metric="f1")
y_pred_best = (y_pred_proba >= best_threshold).astype(int)

print(f"Melhor threshold para F1: {best_threshold:.4f} (F1: {best_f1:.4f})")

print("\nClassification Report:")
print(classification_report(y_test, y_pred_best))

print("\nMatriz de Confusão:")
print(confusion_matrix(y_test, y_pred_best))

print("\nAUC-ROC:", roc_auc_score(y_test, y_pred_proba))
print("AUC-PR :", average_precision_score(y_test, y_pred_proba))
print("MCC   :", matthews_corrcoef(y_test, y_pred_best))
```

Melhor threshold para F1: 0.1910 (F1: 0.3437)

Classification Report:				
	precision	recall	f1-score	support
0	0.94	0.93	0.94	42403
1	0.31	0.38	0.34	3724
accuracy			0.88	46127
macro avg	0.63	0.65	0.64	46127
weighted avg	0.89	0.88	0.89	46127

Matriz de Confusão:

```
[[39255 3148]
 [ 2298 1426]]
```

AUC-ROC: 0.795399200495334

AUC-PR : 0.2904329200889078

MCC : 0.28136141644784807

Resumo dos Resultados do Modelo

O melhor ponto de corte (threshold) foi **0,1910**, com **F1-score de 0,3437** para a classe de inadimplentes.

O modelo apresenta:

- **Acurácia geral de 88%**, indicando bom desempenho global.
- **Recall de 38% para inadimplentes**, o que significa que o modelo consegue identificar parte relevante dos clientes de alto risco.
- **Precisão de 31% na classe inadimplente**, ainda impactada pelo desbalanceamento dos dados.
- **AUC-ROC de 0,795**, mostrando boa capacidade de separação entre bons e maus pagadores.
- **MCC de 0,28**, indicando correlação razoável entre previsões e valores reais.

Impacto no Negócio

Com o threshold em **0,19**, o modelo:

- Consegue **antecipar uma parcela importante dos inadimplentes**.

- Reduz o risco de liberar crédito para maus pagadores.
- Gera alguns **falsos positivos** (bons clientes marcados como risco), mas mantém o risco financeiro mais controlado.

Recomendação Prática

Usar o threshold próximo de **0,19** como **filtro inicial de risco**, priorizando a prevenção de perdas. Para decisões finais de crédito, recomenda-se um critério mais conservador para equilibrar aprovação e rentabilidade.

13.1 Interpretação do Erro em Termos Financeiros

Premissas

- Total de clientes: **46.127**
- Adimplentes: **42.403**
- Inadimplentes: **3.724**

Premissas financeiras:

- Lucro por adimplente: **R\$ 1.000**
- Perda por inadimplente: **R\$ 5.000**
- Custo por falso positivo: **R\$ 500**
- Falso positivo **não gera lucro**

Threshold do modelo

- **Threshold ótimo (F1): 0,1910**
- **F1-score: 0,3437**
- **AUC-ROC: 0,795**
- **MCC: 0,281**

Matriz de Confusão (0,1910)

	Previsto 0	Previsto 1
Real 0 (Adimplente)	39.255	3.148
Real 1 (Inadimplente)	2.298	1.426

Resumo:

- **VP:** 1.426
- **FN:** 2.298
- **FP:** 3.148
- **VN:** 39.255

Impacto Financeiro

1) **VP (inadimplentes evitados)** Ganho: $1.426 \times 5.000 = R\$ 7.130.000$

2) **FN (inadimplentes não detectados)** Perda: $2.298 \times 5.000 = -R\$ 11.490.000$

3) **FP (bons clientes bloqueados)** Perda: $3.148 \times (1.000 + 500) = -R\$ 4.722.000$

4) **VN (bons clientes mantidos)** Ganho: $39.255 \times 1.000 = R\$ 39.255.000$

Lucro Líquido com o Modelo

```
Lucro = 7.130.000
      - 11.490.000
      - 4.722.000
      + 39.255.000
```

Lucro líquido = R\$ 30.173.000

Cenário sem modelo

- Perda com inadimplentes: $3.724 \times 5.000 = -R\$ 18.620.000$
- Lucro com adimplentes: $42.403 \times 1.000 = R\$ 42.403.000$

Lucro líquido sem modelo = R\$ 23.783.000

Ganho Incremental do Modelo

$30.173.000 - 23.783.000 = R\$ 6.390.000$

Resultado final -

Com **threshold = 0,1910**, o modelo gera aproximadamente:

R\$ 6,39 milhões a mais em lucro em relação ao cenário sem modelo.

14. Testando em um Ambiente Real de Produção

```
In [16]: df_test = pd.read_csv('../data/test.csv')
```

14.1 Valores Ausentes

```
In [17]: cols_old = list(df_test.columns)

cols_old = list(df_test.columns)

cols_new = snake_case(cols_old)

df_test.columns = cols_new

df_test_final = df_test[['sk_id_curr']].copy()

df_test = df_test.drop(columns=["sk_id_curr"])

df_test['flag_own_car'] = df_test['flag_own_car'].map({'N': 0, 'Y': 1})
df_test['flag_own_realty'] = df_test['flag_own_realty'].map({'N': 0, 'Y': 1})
df_test['emergencystate_mode'] = df_test['emergencystate_mode'].map({'No': 0, 'Yes': 1})

weekday_map = {
    'MONDAY': 1,
    'TUESDAY': 2,
    'WEDNESDAY': 3,
    'THURSDAY': 4,
    'FRIDAY': 5,
    'SATURDAY': 6,
    'SUNDAY': 7
}

df_test['weekday_appr_process_start'] = df_test['weekday_appr_process_start'].map(weekday_map)

df_test = df_test.replace('XNA', pd.NA)
```

```

aux_missing_cols = {}

aux_missing_cols['is_building'] = df_test.filter(like='commonarea_').notna().any(axis=1)
aux_missing_cols['is_building_missing'] = (~df_test.filter(like='commonarea_').notna().any(axis=1)).astype(int)
aux_missing_cols['has_basement'] = df_test['basementarea_avg'].notna().astype(int)
aux_missing_cols['basementarea_missing'] = df_test['basementarea_avg'].isna().astype(int)
aux_missing_cols['has_noliving'] = df_test['nonlivingarea_avg'].notna().astype(int)
aux_missing_cols['nonlivingarea_missing'] = df_test['nonlivingarea_avg'].isna().astype(int)
aux_missing_cols['is_employed'] = (~df_test['days_employed'].isna()).astype(int)

aux_missing_cols['livingarea_missing'] = df_test['livingarea_avg'].isna().astype(int)
aux_missing_cols['apartments_missing'] = df_test['apartments_avg'].isna().astype(int)
aux_missing_cols['floorsmax_missing'] = df_test['floorsmax_avg'].isna().astype(int)
aux_missing_cols['years_build_missing'] = df_test['years_build_avg'].isna().astype(int)
aux_missing_cols['commonarea_missing'] = df_test['commonarea_avg'].isna().astype(int)

aux_missing_cols['emergencystate_missing'] = df_test['emergencystate_mode'].isna().astype(int)
df_test['emergencystate_mode'] = df_test['emergencystate_mode'].fillna(0)
for col in ['elevators_avg', 'elevators_mode', 'elevators_medi']:
    aux_missing_cols[col + "_missing"] = df_test[col].isna().astype(int)
    df_test[col] = df_test[col].fillna(0)

cols_moda = [
    'housetype_mode', 'wallsmaterial_mode', 'occupation_type', 'organization_type'
]
for col in cols_moda:
    aux_missing_cols[col + '_missing'] = df_test[col].isna().astype(int)

df_test['housetype_mode'] = df_test['housetype_mode'].fillna('Unknown')
df_test['wallsmaterial_mode'] = df_test['wallsmaterial_mode'].fillna('Unknown')
df_test['occupation_type'] = df_test['occupation_type'].fillna('Other')
df_test['organization_type'] = df_test['organization_type'].fillna('Unknown')

drop_columns = {
    'fondkapremont_mode', 'landarea_mode', 'landarea_medi',
    'livingarea_mode', 'livingarea_medi',
    'entrances_avg', 'entrances_mode', 'entrances_medi',
    'floorsmax_mode', 'floorsmax_medi',
    'apartments_mode', 'apartments_medi', 'years_build_mode', 'years_build_medi'
}

do_not_drop = {
    'livingarea_avg', 'apartments_avg', 'floorsmax_avg', 'years_build_avg',
    'basementarea_avg', 'nonlivingarea_avg', 'livingapartments_missing',
    'own_car_age', 'years_beginexpluatation_avg', 'commonarea_avg', 'nonlivingapartments_avg',
}
prefixes = (
    'nonlivingapartments_', 'livingapartments_', 'floorsmin_', 'commonarea_', 'basementarea_',
    'nonlivingarea_', 'years_beginexpluatation_'
)

cols_to_drop = [
    col for col in df_test.columns
    if ((col.startswith(prefixes) and col not in do_not_drop) or col in drop_columns)
]

df_test = df_test.drop(columns=cols_to_drop)

```

```

In [18]: aux_missing_cols['name_type_suite_missing'] = df_test['name_type_suite'].isna().astype(int)
df_test["name_type_suite"] = df_test["name_type_suite"].fillna(mode_suite)

aux_missing_cols['code_gender_missing'] = df_test['code_gender'].isna().astype(int)
df_test["code_gender"] = df_test["code_gender"].fillna(mode_gender)

```

```

cols_nan_remanescentes = df_test.columns[df_test.isnull().any()]
for col_nome in cols_nan_remanescentes:
    aux_missing_cols[col_nome + '_missing'] = df_test[col_nome].isna().astype(int)
    mediana = df_mediana.loc[df_mediana['coluna'] == col_nome, 'mediana'].values
    if len(mediana) > 0:
        mediana_value = mediana[0]
        df_test[col_nome] = df_test[col_nome].fillna(mediana_value)
    else:
        df_test[col_nome] = df_test[col_nome].fillna(0)

```

```
In [19]: df_test = pd.concat([df_test, pd.DataFrame(aux_missing_cols, index=df_test.index)], axis=1)
df_test = df_test.copy()
```

14.2 Feature Engineering

```
In [ ]: df_test2 = df_test.copy()
```

```

In [9]: df_test2['days_employed_anom'] = df_test2['days_employed'] == 365243
df_test2['days_birth'] = df_test2['days_birth'].abs()
df_test2['days_employed'] = df_test2['days_employed'].abs()
df_test2['days_id_publish'] = df_test2['days_id_publish'].abs()
df_test2['days_registration'] = df_test2['days_registration'].abs()
df_test2['age_years'] = df_test2['days_birth'] / 365
df_test2['employment_years'] = df_test2['days_employed'] / 365
df_test2['id_publish_years'] = df_test2['days_id_publish'] / 365

df_test2['annuity_burden_ratio'] = df_test2['amt_annuity'] / (df_test2['bur_amt_credit_sum_mean'])
df_test2['phone_change_rate'] = df_test2['days_last_phone_change'] / (df_test2['days_birth'] +
df_test2['id_change_rate'] = df_test2['days_id_publish'] / (df_test2['days_birth'] + 1e-6)
df_test2['ext_source_weighted_mean'] = (
    0.5 * df_test2['ext_source_2'] + 0.3 * df_test2['ext_source_3'] + 0.2 * df_test2['ext_source_1']
)
df_test2['ext_source_max'] = df_test2[['ext_source_1', 'ext_source_2', 'ext_source_3']].max(axis=1)
df_test2['amt_application_credit_diff'] = df_test2['prev_amt_application_mean'] - df_test2['bur_amt_credit_mean']
df_test2['amt_application_goods_diff'] = df_test2['prev_amt_application_mean'] - df_test2['amt_goods_mean']

flag_document_cols = [col for col in df_test2.columns if col.startswith('flag_document_')]
df_test2['num_flag_document'] = df_test2[flag_document_cols].sum(axis=1)
df_test2['credit_to_income_ratio'] = df_test2['bur_amt_credit_sum_mean'] / (df_test2['amt_income_total'] +
df_test2['annuity_to_income_ratio'] = df_test2['amt_annuity'] / (df_test2['amt_income_total'] +
df_test2['payment_rate'] = df_test2['amt_annuity'] / (df_test2['bur_amt_credit_sum_mean'] + 1e-6)
df_test2['goods_to_credit_ratio'] = df_test2['bur_amt_credit_sum_mean'] / (df_test2['amt_goods_mean'] +
df_test2['credit_efficiency'] = df_test2['bur_amt_credit_sum_mean'] / (df_test2['prev_amt_application_mean'] +
df_test2['employment_life_ratio'] = df_test2['days_employed'] / (df_test2['days_birth'])
df_test2['bureau_request_intensity'] = (
    df_test2['amt_req_credit_bureau_day'] +
    df_test2['amt_req_credit_bureau_week'] +
    df_test2['amt_req_credit_bureau_mon'] +
    df_test2['amt_req_credit_bureau_qrt']
)

ext_sources = ['ext_source_1', 'ext_source_2', 'ext_source_3']
df_test2['ext_source_mean'] = df_test2[ext_sources].mean(axis=1)
df_test2['ext_source_range'] = df_test2[ext_sources].max(axis=1) - df_test2[ext_sources].min(axis=1)
df_test2['ext_source_std'] = df_test2[ext_sources].std(axis=1)
df_test2['ext_source_agreement'] = 1 / (df_test2['ext_source_std'] + 1e-5)

df_test2['decision_credit_diff'] = df_test2['prev_days_decision_mean'] / (df_test2['bur_amt_credit_mean'] +
df_test2['bureau_to_credit_ratio'] = df_test2['bureau_request_intensity'] / (df_test2['bur_amt_credit_mean'] +
df_test2['credit_per_employment_year'] = df_test2['bur_amt_credit_sum_mean'] / (df_test2['employment_years'] +
df_test2['annuity_per_employment_year'] = df_test2['amt_annuity'] / (df_test2['employment_years'] +
df_test2['income_per_employed'] = df_test2['amt_income_total'] / ((df_test2['days_employed']+1) / (df_test2['days_employed'] + 1)) * (df_test2['amt_income_total'] / df_test2['amt_income_mean']) + 1

```

```

df_test2['income_per_birth'] = df_test2['amt_income_total'] / (df_test2['days_birth'])
df_test2['own_car_age_birth_ratio'] = df_test2['own_car_age'] / (df_test2['days_birth'])
df_test2['own_car_age_employed_ratio'] = df_test2['own_car_age'] / ((df_test2['days_employed']+df_test2['days_since_last_employment_until_application']) / df_test2['days_employed'])
df_test2['days_since_last_employment_until_application'] = df_test2['days_employed'] - df_test2['days_birth']

poly_vars = [
    'days_birth',
    'payment_rate',
    'ext_source_mean',
]
X_poly = df_test2[poly_vars].fillna(0).copy()

X_poly_trans = poly.transform(X_poly)
poly_feature_names = poly.get_feature_names_out(poly_vars)
df_poly = pd.DataFrame(X_poly_trans, columns=poly_feature_names, index=df_test2.index)
df_poly = df_poly.drop(columns=poly_vars, errors='ignore')
df_test2 = pd.concat([df_test2, df_poly], axis=1)

df_test2['employment_stability'] = df_test2['employment_years'] / (df_test2['age_years'] + 1e-6)
df_test2['id_document_age_ratio'] = df_test2['id_publish_years'] / (df_test2['age_years'] + 1e-6)
df_test2['phone_change_frequency'] = 1 / (df_test2['days_last_phone_change'] + 1e-6)

df_test2['recent_instability'] = df_test2['phone_change_rate'] + df_test2['id_change_rate']
df_test2['bureau_overdue_ratio'] = df_test2['bur_amt_credit_max_overdue_max'] / (df_test2['bureau_overdue_max'] + 1e-6)
df_test2['credit_risk_signal'] = df_test2['credit_to_income_ratio'] * ((df_test2['ext_source_1'] + df_test2['ext_source_2']) / (df_test2['ext_source_1'] + df_test2['ext_source_2']))
df_test2['annuity_per_age'] = df_test2['amt_annuity'] / (df_test2['own_car_age'] + 1e-6)
df_test2["overdue_flag"] = (df_test2["bur_amt_credit_max_overdue_max"] > 0).astype(int)
df_test2["entry_vs_due_ratio"] = df_test2["instf_inst_days_entry_payment_mean_mean"] / (df_test2["instf_inst_days_entry_payment_mean_mean"] + df_test2["instf_inst_days_entry_payment_mean_mean"])
df_test2["ext_source_interaction"] = df_test2["ext_source_1"] * df_test2["ext_source_3"]
df_test2["pos_balance_range"] = df_test2["posf_pos_months_balance_max_max"] - df_test2["bur_bb_max"]
df_test2["area_quality"] = df_test2["totalarea_mode"] / (df_test2["region_population_relative"] + df_test2["region_population_relative"])

```

14.3 Scaling e Encoding

```
In [ ]: df_test3 = df_test2.copy()
```

```
In [14]: log_transform_cols = [
    'decision_credit_diff',
    'annuity_burden_ratio',
    'credit_efficiency',
]

for col in log_transform_cols:
    df_test3[col + '_log'] = np.log1p(df_test3[col])
    df_test3.drop(columns=col, inplace=True)
```

c:\Users\Patryck\AppData\Local\Programs\Python\Python311\Lib\site-packages\pandas\core\arraylike.py:399: RuntimeWarning: invalid value encountered in log1p
 result = getattr(ufunc, method)(*inputs, **kwargs)

```
In [15]: cols_to_scale = [
    'phone_change_frequency',
    'amt_income_total',
    'amt_credit',
    'num_flag_document',
    'amt_annuity',
    'amt_goods_price',
    'apartments_avg',
    'floorsmax_avg',
    'livingarea_avg',
    'years_build_avg',
    'phone_change_rate',
    'elevators_avg',
```

```
'elevators_mode',
'elevators_medi',
'own_car_age',
'obs_30_cnt_social_circle',
'def_30_cnt_social_circle',
'obs_60_cnt_social_circle',
'def_60_cnt_social_circle',
'amt_req_credit_bureau_hour',
'amt_req_credit_bureau_day',
'amt_req_credit_bureau_week',
'amt_req_credit_bureau_mon',
'amt_req_credit_bureau_qrt',
'amt_req_credit_bureau_year',
'goods_to_credit_ratio',
'basementarea_avg',
'livingapartments_avg',
'landarea_avg',
'nonlivingapartments_avg',
'commonarea_avg',
'years_beginexploitation_avg',
'nonlivingarea_avg',
'days_employed',
'payment_rate',
'payment_rate^2',
'payment_rate ext_source_mean',
'days_birth^2 payment_rate',
'days_birth payment_rate^2',
'payment_rate^3',
'payment_rate^2 ext_source_mean',
'payment_rate ext_source_mean^2',
]
]
```

```
df_test3[cols_to_scale] = robust_scaler.transform(df_test3[cols_to_scale])
```

```
In [16]: cols_to_standard_scale = [
    'days_registration',
    'days_since_last_employment_until_application',
    'income_per_employed',
    'days_id_publish',
    'days_last_phone_change',
    'recent_instability',
    'employment_stability',
    'employment_life_ratio',
    'cnt_fam_members',
    'goods_to_credit_ratio',
    'days_birth',
    'days_birth^2',
    'days_birth payment_rate',
    'days_birth ext_source_mean',
    'ext_source_mean^2',
    'days_birth^3',
    'days_birth^2 ext_source_mean',
    'days_birth payment_rate ext_source_mean',
    'days_birth ext_source_mean^2',
    'ext_source_mean^3'
]
```

```
df_test3[cols_to_standard_scale] = standard_scaler.transform(df_test3[cols_to_standard_scale])
```

```
In [17]: cols_to_minmax_scale = [
    'cnt_children',
    'region_population_relative',
    'bureau_to_credit_ratio',
    'bureau_request_intensity',
    'annuity_to_income_ratio',
    'id_document_age_ratio',
]
```

```
df_test3[cols_to_minmax_scale] = minmax_scaler.transform(df_test3[cols_to_minmax_scale])
```

```
In [20]: binary_cols = [col for col in df_test3.columns if sorted(df_test3[col].dropna().unique()) in [1, 2]

lists_to_exclude = set(
    cols_to_standard_scale +
    cols_to_scale +
    cols_to_minmax_scale +
    binary_cols +
    [col for col in df_test3.columns if df_test3[col].dtype == 'object'] +
    ['target']
)

other_columns = [col for col in df_test3.columns if (col not in lists_to_exclude) and (col not in binary_cols)]
df_test3[other_columns] = scaler.transform(df_test3[other_columns])
```

```
In [10]: df_test3['name_contract_type'] = (df_test3['name_contract_type'] != 'Cash loans').astype(int)
df_test3['code_gender'] = df_test3['code_gender'].map({'F': 0, 'M': 1})

encoded_suite_test = encoder_suite.transform(df_test3[['name_type_suite']])
col_names_suite = [f"nts_{cat}" for cat in encoder_suite.categories_[0][1:]]
encoded_df_suite_test = pd.DataFrame(encoded_suite_test, columns=col_names_suite, index=df_test3.index)
df_test3 = pd.concat([df_test3.drop(columns=['name_type_suite']), encoded_df_suite_test], axis=1)

encoded_fam_test = encoder_fam.transform(df_test3[['name_family_status']])
col_names_fam = [f"nfs_{cat}" for cat in encoder_fam.categories_[0][1:]]
encoded_df_fam_test = pd.DataFrame(encoded_fam_test, columns=col_names_fam, index=df_test3.index)
df_test3 = pd.concat([df_test3.drop(columns=['name_family_status']), encoded_df_fam_test], axis=1)

encoded_housing_test = encoder_housing.transform(df_test3[['name_housing_type']])
col_names_housing = [f"nht_{cat}" for cat in encoder_housing.categories_[0][1:]]
encoded_df_housing_test = pd.DataFrame(encoded_housing_test, columns=col_names_housing, index=df_test3.index)
df_test3 = pd.concat([df_test3.drop(columns=['name_housing_type']), encoded_df_housing_test], axis=1)

encoded_housetype_mode_test = encoder_housetype_mode.transform(df_test3[['housetype_mode']])
col_names_housetype_mode = [f"htm_{cat}" for cat in encoder_housetype_mode.categories_[0][1:]]
encoded_df_housetype_mode_test = pd.DataFrame(encoded_housetype_mode_test, columns=col_names_housetype_mode, index=df_test3.index)
df_test3 = pd.concat([df_test3.drop(columns=['housetype_mode']), encoded_df_housetype_mode_test], axis=1)

encoded_walls_test = encoder_walls.transform(df_test3[['wallsmaterial_mode']])
col_names_walls = [f"wm_{cat}" for cat in encoder_walls.categories_[0][1:]]
encoded_df_walls_test = pd.DataFrame(encoded_walls_test, columns=col_names_walls, index=df_test3.index)
df_test3 = pd.concat([df_test3.drop(columns=['wallsmaterial_mode']), encoded_df_walls_test], axis=1)
```

```
In [11]: df_test3['name_income_type_te'] = df_test3['name_income_type'].map(target_mean_income)
df_test3.drop('name_income_type', axis=1, inplace=True)

df_test3['occupation_type_te'] = df_test3['occupation_type'].map(target_mean_occupation)
df_test3.drop('occupation_type', axis=1, inplace=True)

df_test3['organization_type_te'] = df_test3['organization_type'].map(target_mean_organization)
df_test3.drop('organization_type', axis=1, inplace=True)

df_test3['region_rating_client_te'] = df_test3['region_rating_client'].map(target_mean_region_rating)
df_test3.drop('region_rating_client', axis=1, inplace=True)

df_test3['region_rating_client_w_city_te'] = df_test3['region_rating_client_w_city'].map(target_mean_region_rating_w_city)
df_test3.drop('region_rating_client_w_city', axis=1, inplace=True)
```

```
In [12]: education_order = [
    'Lower secondary',
    'Secondary / secondary special',
    'Incomplete higher',
    'Higher education',
    'Academic degree'
]
```

```
df_test3['name_education_type'] = pd.Categorical(  
    df_test3['name_education_type'],  
    categories=education_order,  
    ordered=True  
) .codes
```

14.4 Selecionando Features

```
In [ ]: df_test4 = df_test3.copy()  
  
In [15]: df_test4.columns = [col.replace('/', '_').replace(',', '_').replace(' ', '') for col in df_test4.columns]  
  
In [16]: X_test_final = df_test4[X.columns].copy()  
  
In [18]: df_test_final = pd.read_csv("../data/test.csv")[['SK_ID_CURR']].copy()  
  
In [19]: df_test_final['TARGET'] = xgb_final.predict_proba(X_test_final)[:, 1]  
  
In [ ]: df_test_final.to_csv('../data/test_final.csv', index=False)
```

Auc final : 0.78421

15. Conclusão

Conclusão do Projeto

O desenvolvimento do modelo preditivo para estimar a **probabilidade de inadimplência de clientes** demonstrou, de forma prática, o nível de **complexidade real dos problemas de risco de crédito em larga escala**. Trabalhando com um conjunto de dados com **mais de 800 variáveis**, forte **desbalanceamento de classes** e sinais fracos em variáveis individuais, o projeto exigiu uma abordagem profundamente analítica, cuidadosa e orientada a valor de negócio.

O elevado número de variáveis trouxe desafios adicionais, como **multicolinearidade, ruído informacional e risco de overfitting**, exigindo forte disciplina em **seleção de atributos, regularização e validação cruzada**. Ainda assim, a combinação de engenharia de features, técnicas de ensemble e calibração probabilística permitiu extrair padrões relevantes de uma base altamente complexa e heterogênea.

Valor Estratégico do Modelo

O modelo demonstrou capacidade real de:

- Diferenciar, de forma consistente, **perfis de maior e menor risco**;
- Antecipar comportamentos de inadimplência com **impacto financeiro mensurável**;
- Sustentar decisões com base em **probabilidades calibradas**, e não apenas regras heurísticas.

O threshold escolhido mostrou-se adequado ao objetivo estratégico da instituição: **reduzir perdas sem comprometer a eficiência comercial**.

A simulação financeira confirmou que o uso do modelo gera **ganho incremental de lucro**, mesmo após considerados os custos operacionais de ações preventivas e o impacto de falsos positivos. Isso demonstra que o projeto não apenas melhora métricas estatísticas, mas **entrega resultado financeiro real**.

Maturidade Analítica e Técnica

Além do resultado prático, o projeto consolidou um nível relevante de **maturidade técnica**, incluindo:

- Tratamento de uma base com **mais de 800 variáveis** e diferentes tipos de dados;
- Balanceamento entre **interpretabilidade e performance** do modelo;
- Integração entre métricas técnicas (AUC, F1, MCC) e **métricas de negócio** (lucro, perdas evitadas e custo operacional).

O projeto reforça que modelos de risco de crédito não são construídos apenas para “prever”, mas para **sustentar decisões estratégicas com impacto direto em rentabilidade e sustentabilidade da carteira.**

Encerramento

O resultado final é um modelo **robusto, escalável e alinhado à realidade do negócio**, capaz de operar em cenários com grande volume de variáveis, forte desbalanceamento e sinais fracos.

Mais do que um exercício técnico, o projeto evidencia o papel da **ciência de dados como ferramenta central para gestão de risco, eficiência operacional e vantagem competitiva no setor financeiro.**

Em síntese, o trabalho demonstra que, mesmo em ambientes complexos e ruidosos, a combinação de **engenharia de dados, modelagem estatística e estratégia de negócio gera valor concreto e mensurável.**