Цель лабораторной работы:

изучение способов предварительной обработки данных для дальнейшего формирования моделей.

Задание:

Выбрать набор данных (датасет), содержащий категориальные признаки и пропуски в данных. Для выполне ния следующих пунктов можно использовать несколько различных наборов данных (один для обработки пр опусков, другой для категориальных признаков и т.д.) Для выбранного датасета (датасетов) на основе материалов лекции решить следующие задачи: обработку пропусков в данных; кодирование категориальных признаков; масштабирование данных.

In [331]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

In [332]:

```
# Набор 1
data = pd.read_csv('quakes.csv', sep=";")
# Набор 2
data1 = pd.read_csv('esoph.csv', sep=";")
```

In [333]:

```
# Набор 1
data.shape
```

Out[333]:

```
(1000, 6)
```

In [334]:

```
# Набор 2
data1.shape
```

Out[334]:

```
(88, 6)
```

In [335]:

```
# Набор 1
data.dtypes
```

Out[335]:

```
Unnamed: 0      int64
lat           float64
long          float64
depth         float64
mag            object
stations        int64
dtype: object
```

In [336]:

```
# Набор 2
data1.dtypes
```

Out[336]:

```
Unnamed: 0      int64
agegp         float64
alcgp          object
tobgp          object
ncases          int64
ncontrols     float64
dtype: object
```

In [337]:

```
# пропуски в наборе 1
data.isnull().sum()
```

Out[337]:

```
Unnamed: 0     0
lat            0
long          12
depth          6
mag            0
stations       0
dtype: int64
```

In [338]:

```
# пропуски в наборе 2
data1.isnull().sum()
```

Out[338]:

```
Unnamed: 0    0
agegp         7
alcgp         0
tobgp         0
ncases        0
ncontrols     6
dtype: int64
```

In [339]:

```
# Набор 1
data.head()
```

Out[339]:

	Unnamed: 0	lat	long	depth	mag	stations
0	1	-20.42	181.62	562.0	04.Aug	41
1	2	-20.62	181.03	650.0	04.Feb	15
2	3	-26.00	184.10	42.0	05.Apr	43
3	4	-17.97	181.66	626.0	04.Jan	19
4	5	-20.42	181.96	649.0	4	11

In [340]:

```
# Набор 2
data1.head()
```

Out[340]:

	Unnamed: 0	agegp	alcgp	tobgp	ncases	ncontrols
0	1	2534.0	0-39g/day	0-9g/day	0	40.0
1	2	2534.0	0-39g/day	Oct.19	0	10.0
2	3	2534.0	0-39g/day	20-29	0	6.0
3	4	2534.0	0-39g/day	30+	0	5.0
4	5	2534.0	40-79	0-9g/day	0	27.0

In [341]:

```
total_count = data.shape[0]
print('Строки в наборе 1: {}'.format(total_count))
```

```
Строки в наборе 1: 1000
```

In [342]:

```
total_count1 = data1.shape[0]
print('Строки в наборе 2: {}'.format(total_count1))
```

```
Строки в наборе 2: 88
```

Обработка пропусков

In [343]:

```
# Удаление колонок, содержащих пустые значения в наборе 1
data_new_1 = data.dropna(axis=1, how='any')
(data.shape, data_new_1.shape)
```

Out[343]:

```
((1000, 6), (1000, 4))
```

In [344]:

```
# Удаление колонок, содержащих пустые значения в наборе 2
data_new_11 = data1.dropna(axis=1, how='any')
(data1.shape, data_new_11.shape)
```

Out[344]:

```
((88, 6), (88, 4))
```

In [345]:

```
# Удаление строк, содержащих пустые значения в наборе 1
data_new_2 = data.dropna(axis=0, how='any')
(data.shape, data_new_2.shape)
```

Out[345]:

```
((1000, 6), (982, 6))
```

In [346]:

```
data.head()
```

Out[346]:

	Unnamed: 0	lat	long	depth	mag	stations
0	1	-20.42	181.62	562.0	04.Aug	41
1	2	-20.62	181.03	650.0	04.Feb	15
2	3	-26.00	184.10	42.0	05.Apr	43
3	4	-17.97	181.66	626.0	04.Jan	19
4	5	-20.42	181.96	649.0	4	11

In [347]:

```
# Удаление строк, содержащих пустые значения в наборе 2
data_new_21 = data1.dropna(axis=0, how='any')
(data1.shape, data_new_21.shape)
```

Out[347]:

```
((88, 6), (76, 6))
```

In [348]:

```
data1.head()
```

Out[348]:

	Unnamed: 0	agegp	alcgp	tobgp	ncases	ncontrols
0	1	2534.0	0-39g/day	0-9g/day	0	40.0
1	2	2534.0	0-39g/day	Oct.19	0	10.0
2	3	2534.0	0-39g/day	20-29	0	6.0
3	4	2534.0	0-39g/day	30+	0	5.0
4	5	2534.0	40-79	0-9g/day	0	27.0

In [349]:

```
# Заполнение всех пропущенных значений нулями в наборе 1
data_new_3 = data.fillna(0)
data_new_3.head()
```

Out[349]:

	Unnamed: 0	lat	long	depth	mag	stations
0	1	-20.42	181.62	562.0	04.Aug	41
1	2	-20.62	181.03	650.0	04.Feb	15
2	3	-26.00	184.10	42.0	05.Apr	43
3	4	-17.97	181.66	626.0	04.Jan	19
4	5	-20.42	181.96	649.0	4	11

In [350]:

```
# Заполнение всех пропущенных значений нулями в наборе 1
data_new_31 = data1.fillna(0)
data_new_31.head()
```

Out[350]:

	Unnamed: 0	agegp	alcgp	tobgp	ncases	ncontrols
0	1	2534.0	0-39g/day	0-9g/day	0	40.0
1	2	2534.0	0-39g/day	Oct.19	0	10.0
2	3	2534.0	0-39g/day	20-29	0	6.0
3	4	2534.0	0-39g/day	30+	0	5.0
4	5	2534.0	40-79	0-9g/day	0	27.0

Импьютация

Числовые данные

In [351]:

```
# Выберем числовые колонки с пропущенными значениями
# Цикл по колонкам датасета набора 1
num_cols = []
for col in data.columns:
    # Количество пустых значений
    temp_null_count = data[data[col].isnull()].shape[0]
    dt = str(data[col].dtype)
    if temp_null_count>0 and (dt=='float64' or dt=='int64'):
        num_cols.append(col)
        temp_perc = round((temp_null_count / total_count) * 100.0,
2)
        print('Колонка {}. Тип данных {}. Количество пустых значени
й {}, {}%.'.format(col, dt, temp_null_count, temp_perc))
```

```
Колонка long. Тип данных float64. Количество пустых значений 12, 1
.2%.
Колонка depth. Тип данных float64. Количество пустых значений 6, 0
.6%.
```

In [352]:

```
# Выберем числовые колонки с пропущенными значениями
# Цикл по колонкам датасета набора 2
num_cols1 = []
for col in data1.columns:
    # Количество пустых значений
    temp_null_count1 = data1[data1[col].isnull()].shape[0]
    dt1 = str(data1[col].dtype)
    if temp_null_count1>0 and (dt1=='float64' or dt1=='int64'):
        num_cols1.append(col)
        temp_perc1 = round((temp_null_count1 / total_count1) * 100.
0, 2)
        print('Колонка {}. Тип данных {}. Количество пустых значени
й {}, {}%.'.format(col, dt1, temp_null_count1, temp_perc1))
```

```
Колонка agegp. Тип данных float64. Количество пустых значений 7, 7
.95%.
Колонка ncontrols. Тип данных float64. Количество пустых значений
6, 6.82%.
```

In [353]:

```
# Фильтр по колонкам с пропущенными значениями  набора 1
data_num = data[num_cols]
data_num
```

Out[353]:

	long	depth
0	181.62	562.0
1	181.03	650.0
2	184.10	42.0
3	181.66	626.0
4	181.96	649.0
...	...	...
995	179.54	470.0
996	167.06	248.0
997	184.20	244.0
998	187.80	40.0
999	170.56	165.0

1000 rows × 2 columns

In [354]:

```
# Фильтр по колонкам с пропущенными значениями  набора 2
data_num1 = data1[num_cols1]
data_num1
```

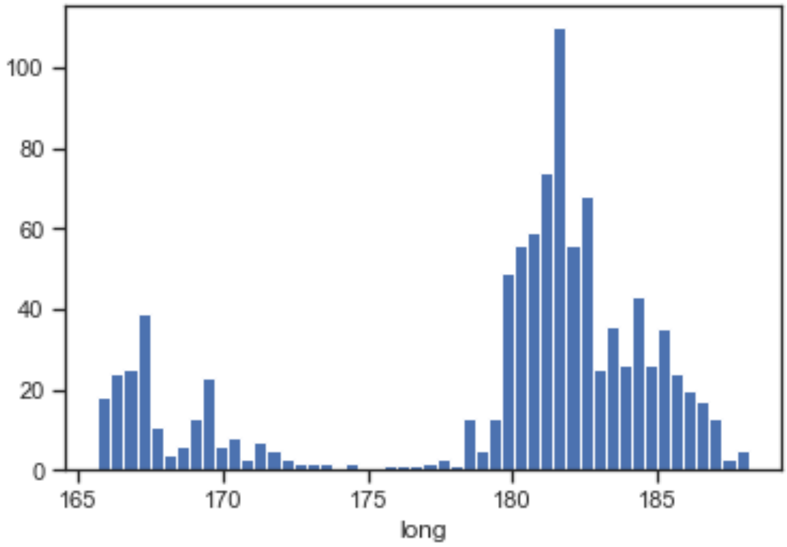
Out[354]:

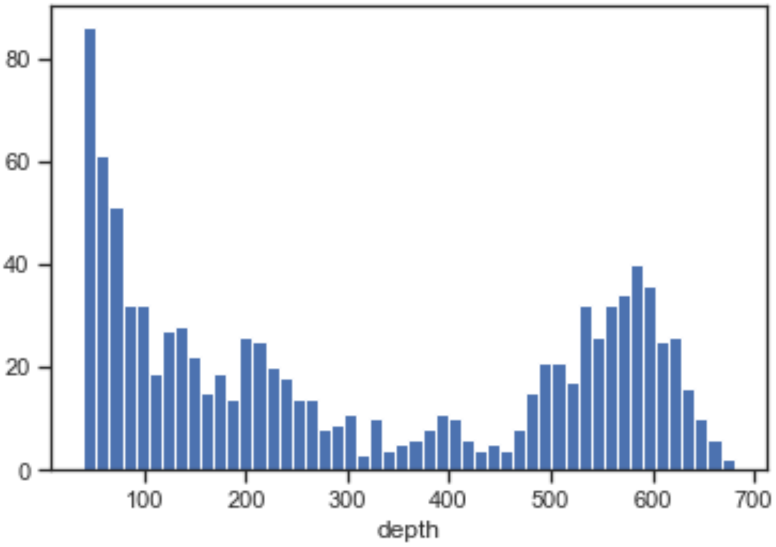
```
     agegp  ncontrols
 0  2534.0       40.0
 1  2534.0       10.0
 2  2534.0        6.0
 3  2534.0        5.0
 4  2534.0       27.0
...    ...        ...
83    75.0        1.0
84    75.0        1.0
85    75.0        1.0
86    75.0        NaN
87    75.0        1.0

88 rows × 2 columns
```

In [355]:

```
# Гистограмма по признакам  набора 1 - Rating
for col in data_num:
    plt.hist(data[col], 50)
    plt.xlabel(col)
    plt.show()
```

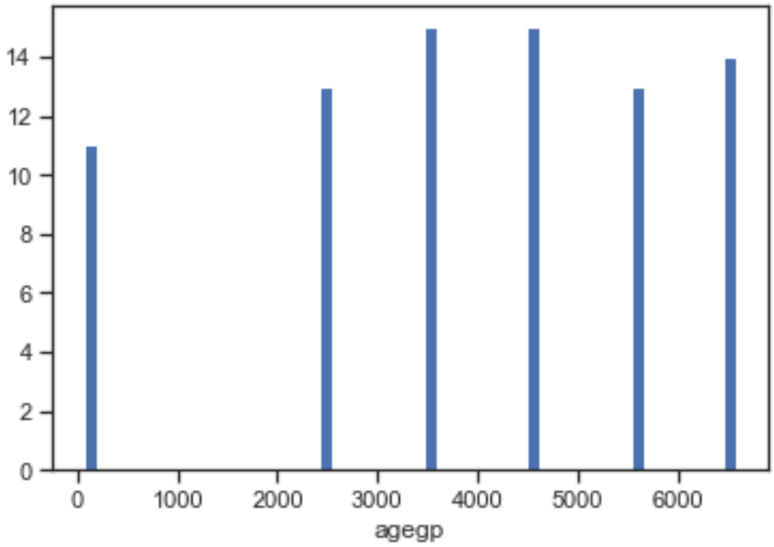
```
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:839
: RuntimeWarning: invalid value encountered in greater_equal
  keep = (tmp_a >= first_edge)
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:840
: RuntimeWarning: invalid value encountered in less_equal
  keep &= (tmp_a <= last_edge)
```

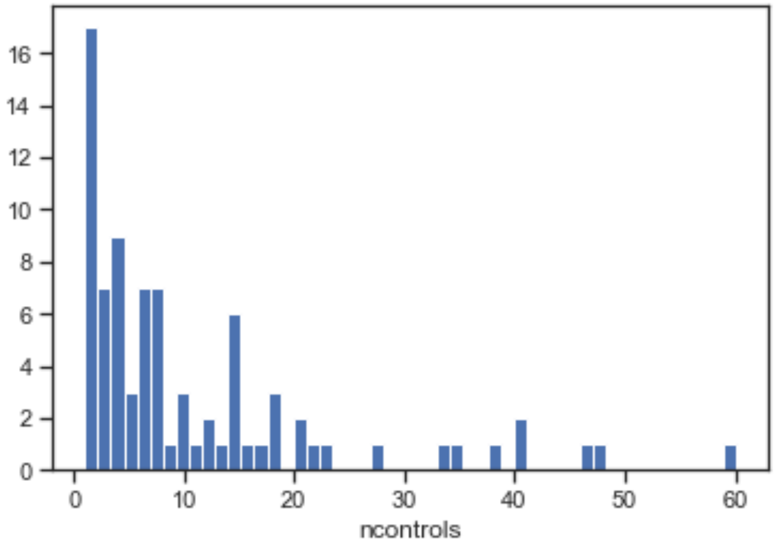




In [356]:

```
# Гистограмма по признакам  набора 1: Sentiment_Polarity, Sentiment
_subjectivity
for col in data_num1:
    plt.hist(data1[col], 50)
    plt.xlabel(col)
    plt.show()
```





In [357]:
```
# Фильтр по пустым значениям поля Rating
data[data['depth'].isnull()]
```

Out[357]:

	Unnamed: 0	lat	long	depth	mag	stations
81	82	-23.84	180.99	NaN	04.May	27
170	171	-17.82	181.83	NaN	04.Mar	24
188	189	-24.27	179.88	NaN	04.Jun	24
189	190	-15.85	185.13	NaN	04.Jun	29
345	346	-27.71	182.47	NaN	04.Mar	11
361	362	-16.90	185.72	NaN	4	22

In [358]:
```
# Фильтр по пустым значениям поля Sentiment_Polarity
data1[data1['ncontrols'].isnull()]
```

Out[358]:

	Unnamed: 0	agegp	alcgp	tobgp	ncases	ncontrols
11	12	2534.0	120+	0-9g/day	0	NaN
22	23	3544.0	40-79	30+	0	NaN
46	47	NaN	0-39g/day	0-9g/day	2	NaN
67	68	6574.0	40-79	Oct.19	3	NaN
81	82	75.0	40-79	Oct.19	1	NaN
86	87	75.0	120+	0-9g/day	2	NaN

In [359]:
```
# Запоминаем индексы строк с пустыми значениями  поля Rating
flt_index = data[data['depth'].isnull()].index
flt_index
```

Out[359]:
```
Int64Index([81, 170, 188, 189, 345, 361], dtype='int64')
```

In [360]:
```
# Запоминаем индексы строк с пустыми значениями поля Sentiment_Pola
rity
flt_index1 = data1[data1['ncontrols'].isnull()].index
flt_index1
```

Out[360]:
```
Int64Index([11, 22, 46, 67, 81, 86], dtype='int64')
```

In [361]:
```
# Запоминаем индексы строк с пустыми значениями поляSentiment_Subje
ctivity
flt_index11 = data1[data1['ncontrols'].isnull()].index
flt_index11
```

Out[361]:
```
Int64Index([11, 22, 46, 67, 81, 86], dtype='int64')
```

In [362]:

```
# Проверяем что выводятся нужные строки Rating
data[data.index.isin(flt_index)]
```

Out[362]:

	Unnamed: 0	lat	long	depth	mag	stations
81	82	-23.84	180.99	NaN	04.May	27
170	171	-17.82	181.83	NaN	04.Mar	24
188	189	-24.27	179.88	NaN	04.Jun	24
189	190	-15.85	185.13	NaN	04.Jun	29
345	346	-27.71	182.47	NaN	04.Mar	11
361	362	-16.90	185.72	NaN	4	22

In [363]:

```
# Проверяем что выводятся нужные строки Sentiment_Polarity
data1[data1.index.isin(flt_index1)]
```

Out[363]:

	Unnamed: 0	agegp	alcgp	tobgp	ncases	ncontrols
11	12	2534.0	120+	0-9g/day	0	NaN
22	23	3544.0	40-79	30+	0	NaN
46	47	NaN	0-39g/day	0-9g/day	2	NaN
67	68	6574.0	40-79	Oct.19	3	NaN
81	82	75.0	40-79	Oct.19	1	NaN
86	87	75.0	120+	0-9g/day	2	NaN

In [364]:

```
# Проверяем что выводятся нужные строки Sentiment_Subjectivity
data1[data1.index.isin(flt_index11)]
```

Out[364]:

	Unnamed: 0	agegp	alcgp	tobgp	ncases	ncontrols
11	12	2534.0	120+	0-9g/day	0	NaN
22	23	3544.0	40-79	30+	0	NaN
46	47	NaN	0-39g/day	0-9g/day	2	NaN
67	68	6574.0	40-79	Oct.19	3	NaN
81	82	75.0	40-79	Oct.19	1	NaN
86	87	75.0	120+	0-9g/day	2	NaN

In [365]:

```
# фильтр по колонке Rating
data_num[data_num.index.isin(flt_index)]['depth']
```

Out[365]:

```
81     NaN
170    NaN
188    NaN
189    NaN
345    NaN
361    NaN
Name: depth, dtype: float64
```

In [366]:

```
# фильтр по колонке Sentiment_Polarity
data_num1[data_num1.index.isin(flt_index1)]['ncontrols']
```

Out[366]:

```
11    NaN
22    NaN
46    NaN
67    NaN
81    NaN
86    NaN
Name: ncontrols, dtype: float64
```

In [367]:

```
data_num_Rating = data_num[['depth']]
data_num_Rating.head()
```

Out[367]:

	depth
0	562.0
1	650.0
2	42.0
3	626.0
4	649.0

In [368]:

```
data_num_SPol = data_num1[['ncontrols']]
data_num_SPol.head()
```

Out[368]:

	ncontrols
0	40.0
1	10.0
2	6.0
3	5.0
4	27.0

In [369]:

```
from sklearn.impute import SimpleImputer
from sklearn.impute import MissingIndicator
```

In [370]:

```
# Фильтр для проверки заполнения пустых значений
indicator = MissingIndicator()
mask_missing_values_only = indicator.fit_transform(data_num_Rating)
mask_missing_values_only
```

Out[370]:

```
array([[False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [ True],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [ True],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [ True],
       [ True],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [ True],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [ True],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False],
       [False]])
```

```
In [371]: # Фильтр для проверки заполнения пустых значений
          indicator = MissingIndicator()
          mask_missing_values_only1 = indicator.fit_transform(data_num_SPol)
          mask_missing_values_only1
```

```
Out[371]: array([[False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False]])
```

```
In [372]: # Фильтр для проверки заполнения пустых значений
          indicator = MissingIndicator()
          mask_missing_values_only11 = indicator.fit_transform(data_num_SPol)
          mask_missing_values_only11
```

```
Out[372]: array([[False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False],
                 [False],
                 [False],
                 [False],
                 [ True],
                 [False]])
```

In [373]:
```
strategies=['mean', 'median','most_frequent']
```

In [374]:
```
# Rating
def test_num_impute(strategy_param):
    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(data_num_Rating)
    return data_num_imp[mask_missing_values_only]
```

In [375]:
```
# Sentiment_Polarity
def test_num_impute1(strategy_param):
    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(data_num_SPol)
    return data_num_imp[mask_missing_values_only1]
```

In [376]:
```
# Sentiment_Subjectivity
def test_num_impute11(strategy_param):
    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(data_num_SPol)
    return data_num_imp[mask_missing_values_only11]
```

```
In [377]: strategies[0], test_num_impute(strategies[0])

Out[377]: ('mean', array([311.18008048, 311.18008048, 311.18008048, 311.1800
          8048,
                   311.18008048, 311.18008048]))
```

```
In [378]: strategies[0], test_num_impute1(strategies[0])

Out[378]: ('mean', array([11., 11., 11., 11., 11., 11.]))
```

```
In [379]: # Sentiment_Subjectivity
          strategies[0], test_num_impute11(strategies[0])

Out[379]: ('mean', array([11., 11., 11., 11., 11., 11.]))
```

```
In [380]: strategies[1], test_num_impute(strategies[1])

Out[380]: ('median', array([246., 246., 246., 246., 246., 246.]))
```

```
In [381]: strategies[1], test_num_impute1(strategies[1])

Out[381]: ('median', array([6., 6., 6., 6., 6., 6.]))
```

```
In [382]: # Sentiment_Subjectivity
          strategies[1], test_num_impute11(strategies[1])

Out[382]: ('median', array([6., 6., 6., 6., 6., 6.]))
```

```
In [383]: strategies[2], test_num_impute(strategies[2])

Out[383]: ('most_frequent', array([40., 40., 40., 40., 40., 40.]))
```

```
In [384]: strategies[2], test_num_impute1(strategies[2])

Out[384]: ('most_frequent', array([1., 1., 1., 1., 1., 1.]))
```

```
In [385]: # Sentiment_Subjectivity
          strategies[2], test_num_impute11(strategies[2])

Out[385]: ('most_frequent', array([1., 1., 1., 1., 1., 1.]))
```

In [386]:

```
# Более сложная функция, которая позволяет задавать колонку и вид и
мпьютации
def test_num_impute_col(dataset, column, strategy_param):
    temp_data = dataset[[column]]

    indicator = MissingIndicator()
    mask_missing_values_only = indicator.fit_transform(temp_data)

    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(temp_data)

    filled_data = data_num_imp[mask_missing_values_only]

    return column, strategy_param, filled_data.size, filled_data[0]
, filled_data[filled_data.size-1]
```

In [387]:

```
# Более сложная функция, которая позволяет задавать колонку и вид и
мпьютации
def test_num_impute_col1(dataset, column, strategy_param):
    temp_data = dataset[[column]]

    indicator = MissingIndicator()
    mask_missing_values_only = indicator.fit_transform(temp_data)

    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(temp_data)

    filled_data = data_num_imp[mask_missing_values_only1]

    return column, strategy_param, filled_data.size, filled_data[0]
, filled_data[filled_data.size-1]
```

In [388]:

```
# Sentiment_Subjectivity
# Более сложная функция, которая позволяет задавать колонку и вид и
мпьютации
def test_num_impute_col11(dataset, column, strategy_param):
    temp_data = dataset[[column]]

    indicator = MissingIndicator()
    mask_missing_values_only = indicator.fit_transform(temp_data)

    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(temp_data)

    filled_data = data_num_imp[mask_missing_values_only11]

    return column, strategy_param, filled_data.size, filled_data[0]
, filled_data[filled_data.size-1]
```

In [389]:
```
data[['depth']].describe()
```

Out[389]:

	depth
count	994.000000
mean	311.180080
std	215.647223
min	40.000000
25%	99.000000
50%	246.000000
75%	543.000000
max	680.000000

In [390]:
```
data1[['ncontrols']].describe()
```

Out[390]:

	ncontrols
count	82.000000
mean	11.000000
std	12.364825
min	1.000000
25%	3.000000
50%	6.000000
75%	14.000000
max	60.000000

In [391]:
```
test_num_impute_col(data, 'depth', strategies[0])
```

Out[391]:
```
('depth', 'mean', 6, 311.1800804828974, 311.1800804828974)
```

In [392]:
```
test_num_impute_col1(data1, 'ncontrols', strategies[0])
```

Out[392]:
```
('ncontrols', 'mean', 6, 11.0, 11.0)
```

In [393]:
```
test_num_impute_col(data, 'depth', strategies[1])
```

Out[393]:
```
('depth', 'median', 6, 246.0, 246.0)
```

In [394]:
```
test_num_impute_col1(data1, 'ncontrols', strategies[1])
```

Out[394]:
```
('ncontrols', 'median', 6, 6.0, 6.0)
```

In [395]:
```
test_num_impute_col(data, 'depth', strategies[2])
```

Out[395]:
```
('depth', 'most_frequent', 6, 40.0, 40.0)
```

In [396]:
```
test_num_impute_col1(data1, 'ncontrols', strategies[2])
```

Out[396]:
```
('ncontrols', 'most_frequent', 6, 1.0, 1.0)
```

Обработка пропусков в категориальных данных

In [397]:
```
# Выберем категориальные колонки с пропущенными значениями
# Цикл по колонкам датасета

cat_cols1 = []
for col in data1.columns:
    # Количество пустых значений
    temp_null_count1 = data1[data1[col].isnull()].shape[0]
    dt1 = str(data1[col].dtype)
    if temp_null_count1>0 and (dt1=='object'):
        cat_cols1.append(col)
        temp_perc1 = round((temp_null_count1 / total_count1) * 100.
0, 2)
        print('Колонка {}. Тип данных {}. Количество пустых значени
й {}, {}%.'.format(col, dt1, temp_null_count1, temp_perc1))
```

In [401]:
```
cat_temp_data = data[['depth']]
cat_temp_data.head()
```

Out[401]:

	depth
0	562.0
1	650.0
2	42.0
3	626.0
4	649.0

In [402]:

```
cat_temp_data1 = data1[['ncontrols']]
cat_temp_data1.head()
```

Out[402]:

	ncontrols
0	40.0
1	10.0
2	6.0
3	5.0
4	27.0

In [403]:

```
cat_temp_data['depth'].unique()
```

Out[403]:

```
array([562., 650.,  42., 626., 649., 195.,  82., 194., 211., 622.,
583.,
       249., 554., 600., 139., 306.,  50., 590., 570., 598., 576.,
512.,
       125., 431., 537., 155., 498., 582., 328., 553., 292., 349.,
48.,
       206., 574., 585., 230., 263.,  96., 511.,  94., 246.,  56.,
329.,
        70., 493., 129., 223.,  46., 593., 489., 445., 584., 535.,
530.,
       260., 613.,  84., 286., 587., 627.,  40., 152., 201., 506.,
546.,
       564., 197., 265., 323., 304.,  75.,  nan, 579., 284., 450.,
170.,
       117., 538., 123.,  69., 128., 236., 497., 271., 224., 375.,
365.,
       484., 108., 608.,  72., 636., 293., 100., 146., 280., 388.,
477.,
       617., 606., 609.,  64., 178., 248.,  81., 571.,  49., 517.,
307.,
       189., 527.,  63., 510., 624.,  53., 199., 149., 210., 658.,
220.,
       205., 614., 186.,  97., 462., 573., 127., 229., 112., 140.,
597.,
       452.,  93., 103., 504., 202.,  59., 244., 239., 434.,  99.,
399.,
       216., 544., 542., 339., 640.,  67., 161., 534.,  45., 309.,
234.,
       569., 605., 422., 637., 204., 175., 595., 360., 367., 190.,
629.,
       261., 603., 508., 350., 533., 411., 338., 226., 618., 242.,
342.,
        90., 130.,  65., 397., 505.,  71., 207., 154., 232., 106.,
664.,
        57., 525.,  74.,  44., 470., 298., 148., 107., 218., 619.,
150.,
       180., 179., 680., 254., 521., 526., 270., 548., 158., 300.,
482.,
       607., 105., 577., 529., 528., 492., 561., 413., 565., 138.,
383.,
       522., 671., 572., 641., 507., 601., 654., 126., 555., 500.,
515.,
       501.,  55., 644., 442., 464., 200., 479., 325., 575., 483.,
118.,
        83.,  61., 219.,  68.,  43.,  80.,  51.,  54., 403.,  60.,
406.,
       221., 502., 423., 536., 630., 153., 188., 124., 401., 102.,
556.,
       417., 591., 646.,  52.,  41., 109., 475.,  66., 481., 151.,
47.,
       119., 176., 602., 488., 343., 563., 259., 476., 499., 257.,
165.,
       136., 524., 467., 184., 237., 162., 604., 639., 628., 632.,
215.,
       135., 297., 568., 168., 269., 143.,  95., 142., 104., 169.,
474.,
       294., 594., 638., 520., 384.,  62., 203., 132., 543., 589.,
485.,
        58., 541., 144., 460., 137., 586., 213., 393., 296., 549.,
85.,
        98.,  89.,  76., 273., 264., 174., 420., 559., 405., 599.,
480.,
       566., 611., 409., 209., 134., 243., 615., 377., 278., 550.,
518.,
       116., 491., 376., 332., 182.,  79., 164., 651., 642., 390.,
539.,
       631., 299., 255., 616., 655., 356., 385., 208., 545., 487.,
183.,
       166., 133.,  86., 287., 348., 578., 361., 275.,  78., 302.,
440.,
       156., 391., 592.,  87., 490., 663., 625., 557., 402., 532.,
111.,
       364., 228., 225., 334., 326., 121., 432., 580., 581., 513.,
77.,
       315., 567., 560., 266., 231., 262., 331., 558., 268., 193.,
172.,
       217., 251., 291.])
```

```
In [404]: cat_temp_data1['ncontrols'].unique()
```

```
Out[404]: array([40., 10.,  6.,  5., 27.,  7.,  4.,  2.,  1., nan, 60., 14.,
8.,
       35., 23., 11.,  3., 46., 18., 38., 21., 15., 16., 22., 12.,
17.,
       48., 34.,  9., 13.])
```

In [405]:
```
cat_temp_data[cat_temp_data['depth'].isnull()].shape
```

Out[405]:
```
(6, 1)
```

In [406]:
```
# Импьютация наиболее частыми значениями
imp2 = SimpleImputer(missing_values=np.nan, strategy='most_frequent
')
data_imp2 = imp2.fit_transform(cat_temp_data)
data_imp2
```

Out[406]:
```
array([[562.],
       [650.],
       [ 42.],
       [626.],
       [649.],
       [195.],
       [ 82.],
       [194.],
       [211.],
       [622.],
       [583.],
       [249.],
       [554.],
       [600.],
       [139.],
       [306.],
       [ 50.],
       [590.],
       [570.],
       [598.],
       [576.],
       [211.],
       [512.],
       [125.],
       [431.],
       [537.],
       [155.],
       [498.],
       [582.],
       [328.],
       [553.],
       [ 50.],
       [292.],
       [349.],
       [ 48.],
       [600.],
       [206.],
       [574.],
       [585.],
       [230.],
       [263.],
       [ 96.],
       [511.],
                    [ 94.],
                    [246.],
                    [ 56.],
                    [329.],
                    [ 70.],
                    [493.],
                    [129.],
                    [554.],
                    [223.],
                    [ 46.],
                    [593.],
                    [489.],
                    [562.],
                    [445.],
                    [584.],
                    [535.],
                    [530.],
                    [582.],
                    [260.],
                    [613.],
                    [ 84.],
                    [593.],
                    [286.],
                    [587.],
                    [627.],
                    [530.],
                    [ 40.],
                    [152.],
                    [201.],
                    [ 96.],
                    [506.],
                    [546.],
                    [564.],
                    [197.],
                    [265.],
                    [323.],
                    [304.],
                    [ 75.],
                    [ 40.],
                    [579.],
                    [284.],
                    [450.],
                    [170.],
                    [117.],
                    [538.],
                    [123.],
                    [ 69.],
                    [128.],
                    [236.],
                    [497.],
                    [271.],
                    [224.],
                    [375.],
       [365.],
       [306.],
       [ 50.],
       [484.],
       [108.],
       [583.],
       [608.],
       [ 72.],
       [636.],
       [293.],
       [ 42.],
       [100.],
       [ 42.],
       [146.],
       [280.],
       [388.],
       [477.],
       [617.],
       [606.],
       [609.],
       [ 46.],
       [ 64.],
       [178.],
       [248.],
       [ 82.],
       [ 81.],
       [606.],
       [571.],
       [328.],
       [ 49.],
       [517.],
       [600.],
       [ 94.],
       [307.],
       [189.],
       [527.],
       [ 63.],
       [510.],
       [624.],
       [ 53.],
       [ 42.],
       [199.],
       [149.],
       [210.],
       [658.],
       [582.],
       [220.],
       [205.],
       [614.],
       [186.],
       [ 97.],
       [ 48.],
       [462.],
                   [573.],
                   [ 56.],
                   [127.],
                   [229.],
                   [112.],
                   [140.],
                   [597.],
                   [ 69.],
                   [452.],
                   [ 93.],
                   [103.],
                   [626.],
                   [504.],
                   [202.],
                   [ 42.],
                   [ 59.],
                   [ 40.],
                   [205.],
                   [244.],
                   [553.],
                   [239.],
                   [ 40.],
                   [434.],
                   [627.],
                   [ 99.],
                   [399.],
                   [216.],
                   [544.],
                   [206.],
                   [542.],
                   [339.],
                   [640.],
                   [ 67.],
                   [161.],
                   [375.],
                   [534.],
                   [ 45.],
                   [309.],
                   [576.],
                   [ 40.],
                   [ 40.],
                   [234.],
                   [223.],
                   [569.],
                   [605.],
                   [422.],
                   [637.],
                   [204.],
                   [175.],
                   [538.],
                   [595.],
                   [360.],
                   [445.],
                    [367.],
                    [190.],
                    [629.],
                    [261.],
                    [603.],
                    [508.],
                    [350.],
                    [533.],
                    [411.],
                    [338.],
                    [226.],
                    [ 93.],
                    [535.],
                    [573.],
                    [186.],
                    [618.],
                    [242.],
                    [342.],
                    [ 90.],
                    [130.],
                    [ 65.],
                    [397.],
                    [505.],
                    [ 71.],
                    [207.],
                    [154.],
                    [232.],
                    [106.],
                    [664.],
                    [397.],
                    [431.],
                    [ 57.],
                    [525.],
                    [574.],
                    [ 74.],
                    [617.],
                    [ 44.],
                    [470.],
                    [298.],
                    [148.],
                    [ 48.],
                    [107.],
                    [218.],
                    [597.],
                    [619.],
                    [ 46.],
                    [553.],
                    [150.],
                    [180.],
                    [ 97.],
                    [587.],
                    [106.],
                    [179.],
                [680.],
                [304.],
                [254.],
                [521.],
                [526.],
                [270.],
                [548.],
                [158.],
                [562.],
                [300.],
                [ 65.],
                [130.],
                [ 82.],
                [482.],
                [607.],
                [105.],
                [590.],
                [498.],
                [504.],
                [577.],
                [529.],
                [528.],
                [492.],
                [561.],
                [579.],
                [413.],
                [565.],
                [138.],
                [383.],
                [260.],
                [522.],
                [671.],
                [123.],
                [572.],
                [529.],
                [641.],
                [ 67.],
                [546.],
                [507.],
                [158.],
                [148.],
                [562.],
                [601.],
                [175.],
                [260.],
                [654.],
                [242.],
                [126.],
                [555.],
                [637.],
                [500.],
                [515.],
                [583.],
[501.],
[ 55.],
[644.],
[641.],
[442.],
[464.],
[200.],
[479.],
[497.],
[218.],
[492.],
[325.],
[123.],
[210.],
[575.],
[129.],
[ 74.],
[ 49.],
[483.],
[ 93.],
[118.],
[ 83.],
[ 61.],
[534.],
[ 42.],
[219.],
[544.],
[ 56.],
[ 68.],
[ 69.],
[ 45.],
[ 43.],
[ 65.],
[ 80.],
[ 51.],
[ 68.],
[ 69.],
[ 40.],
[ 61.],
[ 69.],
[ 51.],
[ 55.],
[ 54.],
[ 59.],
[573.],
[ 56.],
[ 65.],
[587.],
[150.],
[403.],
[ 60.],
[130.],
[590.],
       [ 40.],
       [583.],
       [406.],
       [221.],
       [ 40.],
       [502.],
       [103.],
       [423.],
       [158.],
       [527.],
       [219.],
       [536.],
       [630.],
       [249.],
       [ 48.],
       [553.],
       [112.],
       [153.],
       [130.],
       [188.],
       [226.],
       [124.],
       [204.],
       [605.],
       [221.],
       [573.],
       [401.],
       [195.],
       [ 56.],
       [102.],
       [ 44.],
       [556.],
       [417.],
       [591.],
       [646.],
       [565.],
       [ 52.],
       [535.],
       [641.],
       [ 41.],
       [109.],
       [ 40.],
       [548.],
       [118.],
       [593.],
       [492.],
       [123.],
       [475.],
       [153.],
       [112.],
       [ 99.],
       [ 66.],
       [481.],
                [139.],
                [211.],
                [151.],
                [211.],
                [246.],
                [ 47.],
                [484.],
                [119.],
                [ 70.],
                [579.],
                [ 57.],
                [176.],
                [602.],
                [488.],
                [149.],
                [546.],
                [343.],
                [530.],
                [563.],
                [537.],
                [325.],
                [ 80.],
                [259.],
                [476.],
                [499.],
                [257.],
                [165.],
                [136.],
                [146.],
                [524.],
                [ 82.],
                [ 90.],
                [138.],
                [499.],
                [538.],
                [467.],
                [184.],
                [538.],
                [ 45.],
                [237.],
                [162.],
                [136.],
                [604.],
                [107.],
                [538.],
                [639.],
                [ 45.],
                [636.],
                [628.],
                [632.],
                [215.],
                [556.],
                [135.],
       [297.],
       [568.],
       [168.],
       [269.],
       [143.],
       [ 95.],
       [142.],
       [104.],
       [169.],
       [ 65.],
       [ 54.],
       [474.],
       [125.],
       [617.],
       [294.],
       [180.],
       [ 47.],
       [593.],
       [ 94.],
       [201.],
       [537.],
       [594.],
       [638.],
       [ 80.],
       [211.],
       [520.],
       [384.],
       [223.],
       [ 54.],
       [ 57.],
       [ 49.],
       [508.],
       [242.],
       [ 62.],
       [ 63.],
       [203.],
       [132.],
       [543.],
       [589.],
       [ 51.],
       [ 45.],
       [ 63.],
       [485.],
       [ 66.],
       [ 58.],
       [ 70.],
       [541.],
       [598.],
       [ 50.],
       [102.],
       [144.],
       [ 58.],
       [460.],
       [ 69.],
       [570.],
       [ 70.],
       [137.],
       [ 41.],
       [586.],
       [140.],
       [213.],
       [393.],
       [ 51.],
       [ 64.],
       [ 45.],
       [296.],
       [ 50.],
       [ 44.],
       [ 68.],
       [549.],
       [150.],
       [ 47.],
       [543.],
       [ 40.],
       [100.],
       [ 85.],
       [ 98.],
       [ 58.],
       [125.],
       [ 89.],
       [590.],
       [ 49.],
       [543.],
       [506.],
       [ 40.],
       [ 42.],
       [ 76.],
       [ 63.],
       [104.],
       [ 93.],
       [ 64.],
       [587.],
       [ 83.],
       [579.],
       [ 40.],
       [ 62.],
       [273.],
       [393.],
       [264.],
       [174.],
       [574.],
       [309.],
       [ 75.],
       [195.],
       [ 44.],
       [420.],
       [ 63.],
       [609.],
       [ 40.],
       [575.],
       [559.],
       [405.],
       [ 70.],
       [ 41.],
       [605.],
       [130.],
       [577.],
       [599.],
       [ 82.],
       [ 50.],
       [480.],
       [559.],
       [566.],
       [611.],
       [409.],
       [209.],
       [ 70.],
       [ 74.],
       [134.],
       [150.],
       [406.],
       [243.],
       [ 89.],
       [ 53.],
       [ 68.],
       [605.],
       [615.],
       [546.],
       [176.],
       [ 52.],
       [ 66.],
       [377.],
       [186.],
       [ 51.],
       [ 67.],
       [234.],
       [597.],
       [ 64.],
       [511.],
       [ 47.],
       [ 49.],
       [546.],
       [ 75.],
       [ 60.],
       [278.],
       [550.],
       [518.],
       [116.],
       [518.],
       [ 75.],
       [491.],
       [ 56.],
       [376.],
       [584.],
       [ 48.],
       [ 53.],
       [617.],
       [294.],
       [417.],
       [280.],
       [332.],
       [ 85.],
       [525.],
       [613.],
       [182.],
       [ 57.],
       [ 79.],
       [595.],
       [164.],
       [148.],
       [ 82.],
       [153.],
       [556.],
       [649.],
       [651.],
       [ 93.],
       [533.],
       [642.],
       [ 47.],
       [548.],
       [154.],
       [627.],
       [622.],
       [286.],
       [390.],
       [624.],
       [539.],
       [624.],
       [631.],
       [631.],
       [299.],
       [498.],
       [255.],
       [539.],
       [195.],
       [594.],
       [573.],
       [128.],
       [137.],
       [143.],
       [511.],
       [564.],
                    [559.],
                    [248.],
                    [210.],
                    [390.],
                    [616.],
                    [ 98.],
                    [218.],
                    [655.],
                    [356.],
                    [564.],
                    [548.],
                    [655.],
                    [385.],
                    [518.],
                    [598.],
                    [476.],
                    [579.],
                    [603.],
                    [249.],
                    [208.],
                    [587.],
                    [221.],
                    [545.],
                    [586.],
                    [488.],
                    [246.],
                    [ 61.],
                    [524.],
                    [104.],
                    [271.],
                    [487.],
                    [ 83.],
                    [183.],
                    [ 55.],
                    [166.],
                    [586.],
                    [618.],
                    [524.],
                    [133.],
                    [201.],
                    [383.],
                    [ 86.],
                    [555.],
                    [605.],
                    [609.],
                    [204.],
                    [287.],
                    [390.],
                    [348.],
                    [550.],
                    [487.],
                    [578.],
                    [361.],
                [275.],
                [498.],
                [162.],
                [ 78.],
                [134.],
                [554.],
                [608.],
                [367.],
                [599.],
                [ 45.],
                [ 50.],
                [162.],
                [302.],
                [ 57.],
                [203.],
                [ 66.],
                [591.],
                [440.],
                [611.],
                [ 57.],
                [627.],
                [ 89.],
                [500.],
                [264.],
                [127.],
                [ 85.],
                [ 50.],
                [ 75.],
                [182.],
                [619.],
                [223.],
                [156.],
                [ 46.],
                [ 50.],
                [230.],
                [184.],
                [188.],
                [ 80.],
                [ 86.],
                [391.],
                [199.],
                [592.],
                [595.],
                [618.],
                [213.],
                [616.],
                [175.],
                [589.],
                [ 83.],
                [119.],
                [ 70.],
                [ 74.],
                [ 40.],
       [ 87.],
       [ 63.],
       [537.],
       [ 47.],
       [490.],
       [593.],
       [644.],
       [576.],
       [199.],
       [589.],
       [178.],
       [248.],
       [500.],
       [ 71.],
       [568.],
       [ 42.],
       [510.],
       [ 97.],
       [663.],
       [625.],
       [210.],
       [557.],
       [582.],
       [402.],
       [474.],
       [577.],
       [ 56.],
       [ 43.],
       [118.],
       [544.],
       [532.],
       [237.],
       [111.],
       [595.],
       [603.],
       [505.],
       [100.],
       [587.],
       [364.],
       [100.],
       [350.],
       [228.],
       [ 93.],
       [ 66.],
       [225.],
       [334.],
       [ 70.],
       [220.],
       [397.],
       [326.],
       [121.],
       [209.],
       [510.],
                    [ 54.],
                    [242.],
                    [574.],
                    [ 82.],
                    [102.],
                    [ 43.],
                    [432.],
                    [580.],
                    [464.],
                    [479.],
                    [581.],
                    [513.],
                    [ 77.],
                    [ 68.],
                    [ 71.],
                    [ 68.],
                    [586.],
                    [243.],
                    [658.],
                    [315.],
                    [210.],
                    [278.],
                    [638.],
                    [615.],
                    [567.],
                    [560.],
                    [ 99.],
                    [266.],
                    [397.],
                    [180.],
                    [ 40.],
                    [242.],
                    [530.],
                    [133.],
                    [ 62.],
                    [580.],
                    [530.],
                    [566.],
                    [501.],
                    [548.],
                    [587.],
                    [592.],
                    [221.],
                    [560.],
                    [ 94.],
                    [139.],
                    [162.],
                    [562.],
                    [204.],
                    [ 56.],
                    [ 49.],
                    [100.],
                    [231.],
                   [601.],
                   [ 42.],
                   [180.],
                   [559.],
                   [524.],
                   [ 69.],
                   [594.],
                   [262.],
                   [538.],
                   [331.],
                   [ 48.],
                   [ 47.],
                   [558.],
                   [524.],
                   [545.],
                   [477.],
                   [129.],
                   [268.],
                   [117.],
                   [541.],
                   [112.],
                   [162.],
                   [609.],
                   [ 76.],
                   [ 61.],
                   [202.],
                   [ 90.],
                   [133.],
                   [589.],
                   [190.],
                   [138.],
                   [598.],
                   [600.],
                   [162.],
                   [626.],
                   [137.],
                   [ 57.],
                   [201.],
                   [ 69.],
                   [219.],
                   [553.],
                   [524.],
                   [ 51.],
                   [107.],
                   [ 44.],
                   [574.],
                   [128.],
                   [568.],
                   [583.],
                   [622.],
                   [193.],
                   [544.],
                   [118.],
                   [ 51.],
                   [ 63.],
                   [442.],
                   [ 87.],
                   [ 61.],
                   [ 60.],
                   [561.],
                   [138.],
                   [174.],
                   [543.],
                   [530.],
                   [497.],
                   [ 63.],
                   [ 82.],
                   [605.],
                   [234.],
                   [ 41.],
                   [ 40.],
                   [137.],
                   [223.],
                   [109.],
                   [595.],
                   [512.],
                   [613.],
                   [ 60.],
                   [ 43.],
                   [172.],
                   [ 54.],
                   [ 68.],
                   [217.],
                   [102.],
                   [178.],
                   [251.],
                   [ 42.],
                   [575.],
                   [ 43.],
                   [577.],
                   [ 42.],
                   [ 75.],
                   [ 71.],
                   [ 60.],
                   [291.],
                   [125.],
                   [ 69.],
                   [614.],
                   [108.],
                   [575.],
                   [409.],
                   [243.],
                   [642.],
                   [ 45.],
                   [470.],
                   [248.],
              [244.],
              [ 40.],
              [165.]])
```

In [407]:

```
# Импьютация наиболее частыми значениями
imp2 = SimpleImputer(missing_values=np.nan, strategy='most_frequent
')
data_imp21 = imp2.fit_transform(cat_temp_data1)
data_imp21
```

Out[407]:

```
array([[40.],
       [10.],
       [ 6.],
       [ 5.],
       [27.],
       [ 7.],
       [ 4.],
       [ 7.],
       [ 2.],
       [ 1.],
       [ 2.],
       [ 1.],
       [ 1.],
       [ 1.],
       [ 2.],
       [60.],
       [14.],
       [ 7.],
       [ 8.],
       [35.],
       [23.],
       [14.],
       [ 1.],
       [11.],
       [ 6.],
       [ 2.],
       [ 1.],
       [ 3.],
       [ 3.],
       [ 4.],
       [46.],
       [18.],
       [10.],
       [ 4.],
       [38.],
       [21.],
       [15.],
       [ 7.],
       [16.],
       [14.],
       [ 5.],
       [ 4.],
       [ 4.],
        [ 4.],
        [ 3.],
        [ 4.],
        [ 1.],
        [22.],
        [12.],
        [ 6.],
        [40.],
        [21.],
        [17.],
        [ 6.],
        [18.],
        [15.],
        [ 6.],
        [ 4.],
        [10.],
        [ 7.],
        [ 3.],
        [ 6.],
        [48.],
        [14.],
        [ 7.],
        [ 2.],
        [34.],
        [ 1.],
        [ 9.],
        [13.],
        [12.],
        [ 3.],
        [ 1.],
        [ 4.],
        [ 2.],
        [ 1.],
        [ 1.],
        [18.],
        [ 6.],
        [ 3.],
        [ 5.],
        [ 1.],
        [ 3.],
        [ 1.],
        [ 1.],
        [ 1.],
        [ 1.],
        [ 1.]])
```

In [408]:

```
# Импьютация наиболее частыми значениями
imp2 = SimpleImputer(missing_values=np.nan, strategy='most_frequent
')
data_imp211 = imp2.fit_transform(cat_temp_data11)
data_imp211
```

Out[408]:

```
array([[2534.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [3544.],
                 [3544.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [2534.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [3544.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [3544.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [4554.],
                 [3544.],
                 [3544.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [3544.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [5564.],
                 [6574.],
                 [3544.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [6574.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.],
                 [  75.]])
```

In [409]:

```
# Пустые значения отсутствуют
np.unique(data_imp2)
```

Out[409]:

```
array([ 40.,  41.,  42.,  43.,  44.,  45.,  46.,  47.,  48.,  49.,
50.,
        51.,  52.,  53.,  54.,  55.,  56.,  57.,  58.,  59.,  60.,
61.,
        62.,  63.,  64.,  65.,  66.,  67.,  68.,  69.,  70.,  71.,
72.,
        74.,  75.,  76.,  77.,  78.,  79.,  80.,  81.,  82.,  83.,
84.,
        85.,  86.,  87.,  89.,  90.,  93.,  94.,  95.,  96.,  97.,
98.,
        99., 100., 102., 103., 104., 105., 106., 107., 108., 109.,
111.,
       112., 116., 117., 118., 119., 121., 123., 124., 125., 126.,
127.,
       128., 129., 130., 132., 133., 134., 135., 136., 137., 138.,
139.,
       140., 142., 143., 144., 146., 148., 149., 150., 151., 152.,
153.,
       154., 155., 156., 158., 161., 162., 164., 165., 166., 168.,
169.,
       170., 172., 174., 175., 176., 178., 179., 180., 182., 183.,
184.,
       186., 188., 189., 190., 193., 194., 195., 197., 199., 200.,
201.,
       202., 203., 204., 205., 206., 207., 208., 209., 210., 211.,
213.,
       215., 216., 217., 218., 219., 220., 221., 223., 224., 225.,
226.,
       228., 229., 230., 231., 232., 234., 236., 237., 239., 242.,
243.,
       244., 246., 248., 249., 251., 254., 255., 257., 259., 260.,
261.,
       262., 263., 264., 265., 266., 268., 269., 270., 271., 273.,
275.,
       278., 280., 284., 286., 287., 291., 292., 293., 294., 296.,
297.,
       298., 299., 300., 302., 304., 306., 307., 309., 315., 323.,
325.,
       326., 328., 329., 331., 332., 334., 338., 339., 342., 343.,
348.,
       349., 350., 356., 360., 361., 364., 365., 367., 375., 376.,
377.,
       383., 384., 385., 388., 390., 391., 393., 397., 399., 401.,
402.,
       403., 405., 406., 409., 411., 413., 417., 420., 422., 423.,
431.,
       432., 434., 440., 442., 445., 450., 452., 460., 462., 464.,
467.,
       470., 474., 475., 476., 477., 479., 480., 481., 482., 483.,
484.,
       485., 487., 488., 489., 490., 491., 492., 493., 497., 498.,
499.,
       500., 501., 502., 504., 505., 506., 507., 508., 510., 511.,
512.,
       513., 515., 517., 518., 520., 521., 522., 524., 525., 526.,
527.,
       528., 529., 530., 532., 533., 534., 535., 536., 537., 538.,
539.,
       541., 542., 543., 544., 545., 546., 548., 549., 550., 553.,
554.,
       555., 556., 557., 558., 559., 560., 561., 562., 563., 564.,
565.,
       566., 567., 568., 569., 570., 571., 572., 573., 574., 575.,
576.,
       577., 578., 579., 580., 581., 582., 583., 584., 585., 586.,
587.,
       589., 590., 591., 592., 593., 594., 595., 597., 598., 599.,
600.,
       601., 602., 603., 604., 605., 606., 607., 608., 609., 611.,
613.,
       614., 615., 616., 617., 618., 619., 622., 624., 625., 626.,
627.,
       628., 629., 630., 631., 632., 636., 637., 638., 639., 640.,
641.,
       642., 644., 646., 649., 650., 651., 654., 655., 658., 663.,
664.,
       671., 680.])
```

In [410]:
```
# Пустые значения отсутствуют
np.unique(data_imp21)
```

Out[410]:
```
array([ 1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9., 10., 11., 12.,
13.,
       14., 15., 16., 17., 18., 21., 22., 23., 27., 34., 35., 38.,
40.,
       46., 48., 60.])
```

In [411]:
```
# Пустые значения отсутствуют
np.unique(data_imp211)
```

Out[411]:
```
array([  75., 2534., 3544., 4554., 5564., 6574.])
```

In [412]:
```
# Импьютация константой
imp3 = SimpleImputer(missing_values=np.nan, strategy='constant', fi
ll_value=1)
data_imp3 = imp3.fit_transform(cat_temp_data)
data_imp3
```

Out[412]:
```
array([[562.],
       [650.],
       [ 42.],
       [626.],
       [649.],
       [195.],
       [ 82.],
       [194.],
       [211.],
       [622.],
       [583.],
       [249.],
       [554.],
       [600.],
       [139.],
       [306.],
       [ 50.],
       [590.],
       [570.],
       [598.],
       [576.],
       [211.],
       [512.],
                    [125.],
                    [431.],
                    [537.],
                    [155.],
                    [498.],
                    [582.],
                    [328.],
                    [553.],
                    [ 50.],
                    [292.],
                    [349.],
                    [ 48.],
                    [600.],
                    [206.],
                    [574.],
                    [585.],
                    [230.],
                    [263.],
                    [ 96.],
                    [511.],
                    [ 94.],
                    [246.],
                    [ 56.],
                    [329.],
                    [ 70.],
                    [493.],
                    [129.],
                    [554.],
                    [223.],
                    [ 46.],
                    [593.],
                    [489.],
                    [562.],
                    [445.],
                    [584.],
                    [535.],
                    [530.],
                    [582.],
                    [260.],
                    [613.],
                    [ 84.],
                    [593.],
                    [286.],
                    [587.],
                    [627.],
                    [530.],
                    [ 40.],
                    [152.],
                    [201.],
                    [ 96.],
                    [506.],
                    [546.],
                    [564.],
       [197.],
       [265.],
       [323.],
       [304.],
       [ 75.],
       [  1.],
       [579.],
       [284.],
       [450.],
       [170.],
       [117.],
       [538.],
       [123.],
       [ 69.],
       [128.],
       [236.],
       [497.],
       [271.],
       [224.],
       [375.],
       [365.],
       [306.],
       [ 50.],
       [484.],
       [108.],
       [583.],
       [608.],
       [ 72.],
       [636.],
       [293.],
       [ 42.],
       [100.],
       [ 42.],
       [146.],
       [280.],
       [388.],
       [477.],
       [617.],
       [606.],
       [609.],
       [ 46.],
       [ 64.],
       [178.],
       [248.],
       [ 82.],
       [ 81.],
       [606.],
       [571.],
       [328.],
       [ 49.],
       [517.],
       [600.],
       [ 94.],
                   [307.],
                   [189.],
                   [527.],
                   [ 63.],
                   [510.],
                   [624.],
                   [ 53.],
                   [ 42.],
                   [199.],
                   [149.],
                   [210.],
                   [658.],
                   [582.],
                   [220.],
                   [205.],
                   [614.],
                   [186.],
                   [ 97.],
                   [ 48.],
                   [462.],
                   [573.],
                   [ 56.],
                   [127.],
                   [229.],
                   [112.],
                   [140.],
                   [597.],
                   [ 69.],
                   [452.],
                   [ 93.],
                   [103.],
                   [626.],
                   [504.],
                   [202.],
                   [ 42.],
                   [ 59.],
                   [ 40.],
                   [205.],
                   [244.],
                   [553.],
                   [239.],
                   [  1.],
                   [434.],
                   [627.],
                   [ 99.],
                   [399.],
                   [216.],
                   [544.],
                   [206.],
                   [542.],
                   [339.],
                   [640.],
                   [ 67.],
       [161.],
       [375.],
       [534.],
       [ 45.],
       [309.],
       [576.],
       [  1.],
       [  1.],
       [234.],
       [223.],
       [569.],
       [605.],
       [422.],
       [637.],
       [204.],
       [175.],
       [538.],
       [595.],
       [360.],
       [445.],
       [367.],
       [190.],
       [629.],
       [261.],
       [603.],
       [508.],
       [350.],
       [533.],
       [411.],
       [338.],
       [226.],
       [ 93.],
       [535.],
       [573.],
       [186.],
       [618.],
       [242.],
       [342.],
       [ 90.],
       [130.],
       [ 65.],
       [397.],
       [505.],
       [ 71.],
       [207.],
       [154.],
       [232.],
       [106.],
       [664.],
       [397.],
       [431.],
       [ 57.],
       [525.],
       [574.],
       [ 74.],
       [617.],
       [ 44.],
       [470.],
       [298.],
       [148.],
       [ 48.],
       [107.],
       [218.],
       [597.],
       [619.],
       [ 46.],
       [553.],
       [150.],
       [180.],
       [ 97.],
       [587.],
       [106.],
       [179.],
       [680.],
       [304.],
       [254.],
       [521.],
       [526.],
       [270.],
       [548.],
       [158.],
       [562.],
       [300.],
       [ 65.],
       [130.],
       [ 82.],
       [482.],
       [607.],
       [105.],
       [590.],
       [498.],
       [504.],
       [577.],
       [529.],
       [528.],
       [492.],
       [561.],
       [579.],
       [413.],
       [565.],
       [138.],
       [383.],
       [260.],
       [522.],
       [671.],
       [123.],
                [572.],
                [529.],
                [641.],
                [ 67.],
                [546.],
                [507.],
                [158.],
                [148.],
                [562.],
                [601.],
                [175.],
                [260.],
                [654.],
                [242.],
                [126.],
                [555.],
                [637.],
                [500.],
                [515.],
                [583.],
                [501.],
                [ 55.],
                [644.],
                [641.],
                [442.],
                [464.],
                [200.],
                [479.],
                [497.],
                [218.],
                [492.],
                [325.],
                [123.],
                [210.],
                [575.],
                [129.],
                [ 74.],
                [ 49.],
                [483.],
                [ 93.],
                [118.],
                [ 83.],
                [ 61.],
                [534.],
                [ 42.],
                [219.],
                [544.],
                [ 56.],
                [ 68.],
                [ 69.],
                [ 45.],
                [ 43.],
                [ 65.],
       [ 80.],
       [ 51.],
       [ 68.],
       [ 69.],
       [  1.],
       [ 61.],
       [ 69.],
       [ 51.],
       [ 55.],
       [ 54.],
       [ 59.],
       [573.],
       [ 56.],
       [ 65.],
       [587.],
       [150.],
       [403.],
       [ 60.],
       [130.],
       [590.],
       [  1.],
       [583.],
       [406.],
       [221.],
       [ 40.],
       [502.],
       [103.],
       [423.],
       [158.],
       [527.],
       [219.],
       [536.],
       [630.],
       [249.],
       [ 48.],
       [553.],
       [112.],
       [153.],
       [130.],
       [188.],
       [226.],
       [124.],
       [204.],
       [605.],
       [221.],
       [573.],
       [401.],
       [195.],
       [ 56.],
       [102.],
       [ 44.],
       [556.],
       [417.],
                   [591.],
                   [646.],
                   [565.],
                   [ 52.],
                   [535.],
                   [641.],
                   [ 41.],
                   [109.],
                   [ 40.],
                   [548.],
                   [118.],
                   [593.],
                   [492.],
                   [123.],
                   [475.],
                   [153.],
                   [112.],
                   [ 99.],
                   [ 66.],
                   [481.],
                   [139.],
                   [211.],
                   [151.],
                   [211.],
                   [246.],
                   [ 47.],
                   [484.],
                   [119.],
                   [ 70.],
                   [579.],
                   [ 57.],
                   [176.],
                   [602.],
                   [488.],
                   [149.],
                   [546.],
                   [343.],
                   [530.],
                   [563.],
                   [537.],
                   [325.],
                   [ 80.],
                   [259.],
                   [476.],
                   [499.],
                   [257.],
                   [165.],
                   [136.],
                   [146.],
                   [524.],
                   [ 82.],
                   [ 90.],
                   [138.],
                   [499.],
                   [538.],
                   [467.],
                   [184.],
                   [538.],
                   [ 45.],
                   [237.],
                   [162.],
                   [136.],
                   [604.],
                   [107.],
                   [538.],
                   [639.],
                   [ 45.],
                   [636.],
                   [628.],
                   [632.],
                   [215.],
                   [556.],
                   [135.],
                   [297.],
                   [568.],
                   [168.],
                   [269.],
                   [143.],
                   [ 95.],
                   [142.],
                   [104.],
                   [169.],
                   [ 65.],
                   [ 54.],
                   [474.],
                   [125.],
                   [617.],
                   [294.],
                   [180.],
                   [ 47.],
                   [593.],
                   [ 94.],
                   [201.],
                   [537.],
                   [594.],
                   [638.],
                   [ 80.],
                   [211.],
                   [520.],
                   [384.],
                   [223.],
                   [ 54.],
                   [ 57.],
                   [ 49.],
                   [508.],
                   [242.],
                [ 62.],
                [ 63.],
                [203.],
                [132.],
                [543.],
                [589.],
                [ 51.],
                [ 45.],
                [ 63.],
                [485.],
                [ 66.],
                [ 58.],
                [ 70.],
                [541.],
                [598.],
                [ 50.],
                [102.],
                [144.],
                [ 58.],
                [460.],
                [ 69.],
                [570.],
                [ 70.],
                [137.],
                [ 41.],
                [586.],
                [140.],
                [213.],
                [393.],
                [ 51.],
                [ 64.],
                [ 45.],
                [296.],
                [ 50.],
                [ 44.],
                [ 68.],
                [549.],
                [150.],
                [ 47.],
                [543.],
                [ 40.],
                [100.],
                [ 85.],
                [ 98.],
                [ 58.],
                [125.],
                [ 89.],
                [590.],
                [ 49.],
                [543.],
                [506.],
                [ 40.],
                [ 42.],
       [ 76.],
       [ 63.],
       [104.],
       [ 93.],
       [ 64.],
       [587.],
       [ 83.],
       [579.],
       [ 40.],
       [ 62.],
       [273.],
       [393.],
       [264.],
       [174.],
       [574.],
       [309.],
       [ 75.],
       [195.],
       [ 44.],
       [420.],
       [ 63.],
       [609.],
       [ 40.],
       [575.],
       [559.],
       [405.],
       [ 70.],
       [ 41.],
       [605.],
       [130.],
       [577.],
       [599.],
       [ 82.],
       [ 50.],
       [480.],
       [559.],
       [566.],
       [611.],
       [409.],
       [209.],
       [ 70.],
       [ 74.],
       [134.],
       [150.],
       [406.],
       [243.],
       [ 89.],
       [ 53.],
       [ 68.],
       [605.],
       [615.],
       [546.],
       [176.],
       [ 52.],
       [ 66.],
       [377.],
       [186.],
       [ 51.],
       [ 67.],
       [234.],
       [597.],
       [ 64.],
       [511.],
       [ 47.],
       [ 49.],
       [546.],
       [ 75.],
       [ 60.],
       [278.],
       [550.],
       [518.],
       [116.],
       [518.],
       [ 75.],
       [491.],
       [ 56.],
       [376.],
       [584.],
       [ 48.],
       [ 53.],
       [617.],
       [294.],
       [417.],
       [280.],
       [332.],
       [ 85.],
       [525.],
       [613.],
       [182.],
       [ 57.],
       [ 79.],
       [595.],
       [164.],
       [148.],
       [ 82.],
       [153.],
       [556.],
       [649.],
       [651.],
       [ 93.],
       [533.],
       [642.],
       [ 47.],
       [548.],
       [154.],
       [627.],
       [622.],
       [286.],
       [390.],
       [624.],
       [539.],
       [624.],
       [631.],
       [631.],
       [299.],
       [498.],
       [255.],
       [539.],
       [195.],
       [594.],
       [573.],
       [128.],
       [137.],
       [143.],
       [511.],
       [564.],
       [559.],
       [248.],
       [210.],
       [390.],
       [616.],
       [ 98.],
       [218.],
       [655.],
       [356.],
       [564.],
       [548.],
       [655.],
       [385.],
       [518.],
       [598.],
       [476.],
       [579.],
       [603.],
       [249.],
       [208.],
       [587.],
       [221.],
       [545.],
       [586.],
       [488.],
       [246.],
       [ 61.],
       [524.],
       [104.],
       [271.],
       [487.],
       [ 83.],
       [183.],
       [ 55.],
       [166.],
       [586.],
       [618.],
       [524.],
       [133.],
       [201.],
       [383.],
       [ 86.],
       [555.],
       [605.],
       [609.],
       [204.],
       [287.],
       [390.],
       [348.],
       [550.],
       [487.],
       [578.],
       [361.],
       [275.],
       [498.],
       [162.],
       [ 78.],
       [134.],
       [554.],
       [608.],
       [367.],
       [599.],
       [ 45.],
       [ 50.],
       [162.],
       [302.],
       [ 57.],
       [203.],
       [ 66.],
       [591.],
       [440.],
       [611.],
       [ 57.],
       [627.],
       [ 89.],
       [500.],
       [264.],
       [127.],
       [ 85.],
       [ 50.],
       [ 75.],
       [182.],
       [619.],
       [223.],
       [156.],
       [ 46.],
                    [ 50.],
                    [230.],
                    [184.],
                    [188.],
                    [ 80.],
                    [ 86.],
                    [391.],
                    [199.],
                    [592.],
                    [595.],
                    [618.],
                    [213.],
                    [616.],
                    [175.],
                    [589.],
                    [ 83.],
                    [119.],
                    [ 70.],
                    [ 74.],
                    [ 40.],
                    [ 87.],
                    [ 63.],
                    [537.],
                    [ 47.],
                    [490.],
                    [593.],
                    [644.],
                    [576.],
                    [199.],
                    [589.],
                    [178.],
                    [248.],
                    [500.],
                    [ 71.],
                    [568.],
                    [ 42.],
                    [510.],
                    [ 97.],
                    [663.],
                    [625.],
                    [210.],
                    [557.],
                    [582.],
                    [402.],
                    [474.],
                    [577.],
                    [ 56.],
                    [ 43.],
                    [118.],
                    [544.],
                    [532.],
                    [237.],
                    [111.],
       [595.],
       [603.],
       [505.],
       [100.],
       [587.],
       [364.],
       [100.],
       [350.],
       [228.],
       [ 93.],
       [ 66.],
       [225.],
       [334.],
       [ 70.],
       [220.],
       [397.],
       [326.],
       [121.],
       [209.],
       [510.],
       [ 54.],
       [242.],
       [574.],
       [ 82.],
       [102.],
       [ 43.],
       [432.],
       [580.],
       [464.],
       [479.],
       [581.],
       [513.],
       [ 77.],
       [ 68.],
       [ 71.],
       [ 68.],
       [586.],
       [243.],
       [658.],
       [315.],
       [210.],
       [278.],
       [638.],
       [615.],
       [567.],
       [560.],
       [ 99.],
       [266.],
       [397.],
       [180.],
       [ 40.],
       [242.],
       [530.],
                    [133.],
                    [ 62.],
                    [580.],
                    [530.],
                    [566.],
                    [501.],
                    [548.],
                    [587.],
                    [592.],
                    [221.],
                    [560.],
                    [ 94.],
                    [139.],
                    [162.],
                    [562.],
                    [204.],
                    [ 56.],
                    [ 49.],
                    [100.],
                    [231.],
                    [601.],
                    [ 42.],
                    [180.],
                    [559.],
                    [524.],
                    [ 69.],
                    [594.],
                    [262.],
                    [538.],
                    [331.],
                    [ 48.],
                    [ 47.],
                    [558.],
                    [524.],
                    [545.],
                    [477.],
                    [129.],
                    [268.],
                    [117.],
                    [541.],
                    [112.],
                    [162.],
                    [609.],
                    [ 76.],
                    [ 61.],
                    [202.],
                    [ 90.],
                    [133.],
                    [589.],
                    [190.],
                    [138.],
                    [598.],
                    [600.],
       [162.],
       [626.],
       [137.],
       [ 57.],
       [201.],
       [ 69.],
       [219.],
       [553.],
       [524.],
       [ 51.],
       [107.],
       [ 44.],
       [574.],
       [128.],
       [568.],
       [583.],
       [622.],
       [193.],
       [544.],
       [118.],
       [ 51.],
       [ 63.],
       [442.],
       [ 87.],
       [ 61.],
       [ 60.],
       [561.],
       [138.],
       [174.],
       [543.],
       [530.],
       [497.],
       [ 63.],
       [ 82.],
       [605.],
       [234.],
       [ 41.],
       [ 40.],
       [137.],
       [223.],
       [109.],
       [595.],
       [512.],
       [613.],
       [ 60.],
       [ 43.],
       [172.],
       [ 54.],
       [ 68.],
       [217.],
       [102.],
       [178.],
       [251.],
              [ 42.],
              [575.],
              [ 43.],
              [577.],
              [ 42.],
              [ 75.],
              [ 71.],
              [ 60.],
              [291.],
              [125.],
              [ 69.],
              [614.],
              [108.],
              [575.],
              [409.],
              [243.],
              [642.],
              [ 45.],
              [470.],
              [248.],
              [244.],
              [ 40.],
              [165.]])
```

In [413]:

```
# Импьютация константой
imp3 = SimpleImputer(missing_values=np.nan, strategy='constant', fi
ll_value=11)
data_imp31 = imp3.fit_transform(cat_temp_data1)
data_imp31
```

Out[413]:

```
array([[40.],
       [10.],
       [ 6.],
       [ 5.],
       [27.],
       [ 7.],
       [ 4.],
       [ 7.],
       [ 2.],
       [ 1.],
       [ 2.],
       [11.],
       [ 1.],
       [ 1.],
       [ 2.],
       [60.],
       [14.],
       [ 7.],
       [ 8.],
       [35.],
       [23.],
       [14.],
       [11.],
       [11.],
       [ 6.],
       [ 2.],
       [ 1.],
       [ 3.],
       [ 3.],
       [ 4.],
       [46.],
       [18.],
       [10.],
       [ 4.],
       [38.],
       [21.],
       [15.],
       [ 7.],
       [16.],
       [14.],
       [ 5.],
       [ 4.],
       [ 4.],
       [ 4.],
       [ 3.],
       [ 4.],
       [11.],
       [22.],
       [12.],
       [ 6.],
       [40.],
       [21.],
       [17.],
       [ 6.],
       [18.],
       [15.],
       [ 6.],
       [ 4.],
       [10.],
       [ 7.],
       [ 3.],
       [ 6.],
       [48.],
       [14.],
       [ 7.],
       [ 2.],
       [34.],
       [11.],
       [ 9.],
       [13.],
       [12.],
       [ 3.],
       [ 1.],
       [ 4.],
       [ 2.],
       [ 1.],
       [ 1.],
       [18.],
       [ 6.],
       [ 3.],
       [ 5.],
       [11.],
       [ 3.],
       [ 1.],
       [ 1.],
       [ 1.],
       [11.],
       [ 1.]])
```

In [414]:

```
# Импьютация константой
imp3 = SimpleImputer(missing_values=np.nan, strategy='constant', fi
ll_value=3)
data_imp311 = imp3.fit_transform(cat_temp_data11)
data_imp311
```

Out[414]:

```
array([[2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [3.000e+00],
       [3.000e+00],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [2.534e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [3.544e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [3.000e+00],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [4.554e+03],
       [3.000e+00],
       [3.000e+00],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [3.000e+00],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [5.564e+03],
       [6.574e+03],
       [3.000e+00],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [6.574e+03],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
       [7.500e+01],
        [7.500e+01]])
```

In [415]:

```
np.unique(data_imp3)
```

Out[415]:

```
array([  1.,  40.,  41.,  42.,  43.,  44.,  45.,  46.,  47.,  48.,
49.,
         50.,  51.,  52.,  53.,  54.,  55.,  56.,  57.,  58.,  59.,
60.,
         61.,  62.,  63.,  64.,  65.,  66.,  67.,  68.,  69.,  70.,
71.,
         72.,  74.,  75.,  76.,  77.,  78.,  79.,  80.,  81.,  82.,
83.,
         84.,  85.,  86.,  87.,  89.,  90.,  93.,  94.,  95.,  96.,
97.,
         98.,  99., 100., 102., 103., 104., 105., 106., 107., 108.,
109.,
        111., 112., 116., 117., 118., 119., 121., 123., 124., 125.,
126.,
        127., 128., 129., 130., 132., 133., 134., 135., 136., 137.,
138.,
        139., 140., 142., 143., 144., 146., 148., 149., 150., 151.,
152.,
        153., 154., 155., 156., 158., 161., 162., 164., 165., 166.,
168.,
        169., 170., 172., 174., 175., 176., 178., 179., 180., 182.,
183.,
        184., 186., 188., 189., 190., 193., 194., 195., 197., 199.,
200.,
        201., 202., 203., 204., 205., 206., 207., 208., 209., 210.,
211.,
        213., 215., 216., 217., 218., 219., 220., 221., 223., 224.,
225.,
        226., 228., 229., 230., 231., 232., 234., 236., 237., 239.,
242.,
        243., 244., 246., 248., 249., 251., 254., 255., 257., 259.,
260.,
        261., 262., 263., 264., 265., 266., 268., 269., 270., 271.,
273.,
        275., 278., 280., 284., 286., 287., 291., 292., 293., 294.,
296.,
        297., 298., 299., 300., 302., 304., 306., 307., 309., 315.,
323.,
        325., 326., 328., 329., 331., 332., 334., 338., 339., 342.,
343.,
        348., 349., 350., 356., 360., 361., 364., 365., 367., 375.,
376.,
        377., 383., 384., 385., 388., 390., 391., 393., 397., 399.,
401.,
        402., 403., 405., 406., 409., 411., 413., 417., 420., 422.,
423.,
        431., 432., 434., 440., 442., 445., 450., 452., 460., 462.,
464.,
        467., 470., 474., 475., 476., 477., 479., 480., 481., 482.,
483.,
          484., 485., 487., 488., 489., 490., 491., 492., 493., 497.,
498.,
          499., 500., 501., 502., 504., 505., 506., 507., 508., 510.,
511.,
          512., 513., 515., 517., 518., 520., 521., 522., 524., 525.,
526.,
          527., 528., 529., 530., 532., 533., 534., 535., 536., 537.,
538.,
          539., 541., 542., 543., 544., 545., 546., 548., 549., 550.,
553.,
          554., 555., 556., 557., 558., 559., 560., 561., 562., 563.,
564.,
          565., 566., 567., 568., 569., 570., 571., 572., 573., 574.,
575.,
          576., 577., 578., 579., 580., 581., 582., 583., 584., 585.,
586.,
          587., 589., 590., 591., 592., 593., 594., 595., 597., 598.,
599.,
          600., 601., 602., 603., 604., 605., 606., 607., 608., 609.,
611.,
          613., 614., 615., 616., 617., 618., 619., 622., 624., 625.,
626.,
          627., 628., 629., 630., 631., 632., 636., 637., 638., 639.,
640.,
          641., 642., 644., 646., 649., 650., 651., 654., 655., 658.,
663.,
          664., 671., 680.])
```

In [416]:

```
np.unique(data_imp311)
```

Out[416]:

```
array([3.000e+00, 7.500e+01, 2.534e+03, 3.544e+03, 4.554e+03, 5.56
4e+03,
          6.574e+03])
```

In [417]:

```
data_imp3[data_imp3==1].size
```

Out[417]:

```
6
```

In [418]:

```
data_imp31[data_imp31==2].size
```

Out[418]:

```
6
```

In [419]:

```
data_imp311[data_imp311==3].size
```

Out[419]:

```
7
```

In [420]:

```
data.shape
```

Out[420]:

```
(1000, 6)
```

In [421]:

```
data1.shape
```

Out[421]:

```
(88, 6)
```

Преобразование категориальных признаков в числовые

In [422]:

```
cat_enc = pd.DataFrame({'c1':data_imp2.T[0]})
cat_enc
```

Out[422]:

	c1
0	562.0
1	650.0
2	42.0
3	626.0
4	649.0
...	...
995	470.0
996	248.0
997	244.0
998	40.0
999	165.0

1000 rows × 1 columns

In [423]:

```
cat_enc1 = pd.DataFrame({'c1':data_imp21.T[0]})
cat_enc1
```

Out[423]:

	c1
0	40.0
1	10.0
2	6.0
3	5.0
4	27.0
...	...
83	1.0
84	1.0
85	1.0
86	1.0
87	1.0

88 rows × 1 columns

In [424]:

```
cat_enc11 = pd.DataFrame({'c1':data_imp211.T[0]})
cat_enc11
```

Out[424]:

	c1
0	2534.0
1	2534.0
2	2534.0
3	2534.0
4	2534.0
...	...
83	75.0
84	75.0
85	75.0
86	75.0
87	75.0

88 rows × 1 columns

Кодирование категорий целочисленными значениями

```
In [425]: from sklearn.preprocessing import LabelEncoder, OneHotEncoder
```

```
In [426]: le = LabelEncoder()
          cat_enc_le = le.fit_transform(cat_enc['c1'])
```

```
In [427]: le1 = LabelEncoder()
          cat_enc_le1 = le1.fit_transform(cat_enc1['c1'])
```

```
In [428]: le11 = LabelEncoder()
          cat_enc_le11 = le11.fit_transform(cat_enc11['c1'])
```

```
In [429]: cat_enc['c1'].unique()
```

```
Out[429]: array([562., 650.,  42., 626., 649., 195.,  82., 194., 211., 622.,
          583.,
                 249., 554., 600., 139., 306.,  50., 590., 570., 598., 576.,
          512.,
                 125., 431., 537., 155., 498., 582., 328., 553., 292., 349.,
          48.,
                 206., 574., 585., 230., 263.,  96., 511.,  94., 246.,  56.,
          329.,
                  70., 493., 129., 223.,  46., 593., 489., 445., 584., 535.,
          530.,
                 260., 613.,  84., 286., 587., 627.,  40., 152., 201., 506.,
          546.,
                 564., 197., 265., 323., 304.,  75., 579., 284., 450., 170.,
          117.,
                 538., 123.,  69., 128., 236., 497., 271., 224., 375., 365.,
          484.,
                 108., 608.,  72., 636., 293., 100., 146., 280., 388., 477.,
          617.,
                 606., 609.,  64., 178., 248.,  81., 571.,  49., 517., 307.,
          189.,
                 527.,  63., 510., 624.,  53., 199., 149., 210., 658., 220.,
          205.,
                 614., 186.,  97., 462., 573., 127., 229., 112., 140., 597.,
          452.,
                  93., 103., 504., 202.,  59., 244., 239., 434.,  99., 399.,
          216.,
                 544., 542., 339., 640.,  67., 161., 534.,  45., 309., 234.,
          569.,
                 605., 422., 637., 204., 175., 595., 360., 367., 190., 629.,
          261.,
                 603., 508., 350., 533., 411., 338., 226., 618., 242., 342.,
          90.,
                 130.,  65., 397., 505.,  71., 207., 154., 232., 106., 664.,
          57.,
                 525.,  74.,  44., 470., 298., 148., 107., 218., 619., 150.,
          180.,
                 179., 680., 254., 521., 526., 270., 548., 158., 300., 482.,
          607.,
                 105., 577., 529., 528., 492., 561., 413., 565., 138., 383.,
          522.,
                 671., 572., 641., 507., 601., 654., 126., 555., 500., 515.,
          501.,
                  55., 644., 442., 464., 200., 479., 325., 575., 483., 118.,
          83.,
                  61., 219.,  68.,  43.,  80.,  51.,  54., 403.,  60., 406.,
          221.,
                 502., 423., 536., 630., 153., 188., 124., 401., 102., 556.,
          417.,
                 591., 646.,  52.,  41., 109., 475.,  66., 481., 151.,  47.,
          119.,
                 176., 602., 488., 343., 563., 259., 476., 499., 257., 165.,
          136.,
                 524., 467., 184., 237., 162., 604., 639., 628., 632., 215.,
          135.,
                 297., 568., 168., 269., 143.,  95., 142., 104., 169., 474.,
          294.,
                 594., 638., 520., 384.,  62., 203., 132., 543., 589., 485.,
          58.,
                 541., 144., 460., 137., 586., 213., 393., 296., 549.,  85.,
          98.,
                  89.,  76., 273., 264., 174., 420., 559., 405., 599., 480.,
          566.,
                 611., 409., 209., 134., 243., 615., 377., 278., 550., 518.,
          116.,
                 491., 376., 332., 182.,  79., 164., 651., 642., 390., 539.,
          631.,
                 299., 255., 616., 655., 356., 385., 208., 545., 487., 183.,
          166.,
                 133.,  86., 287., 348., 578., 361., 275.,  78., 302., 440.,
          156.,
                 391., 592.,  87., 490., 663., 625., 557., 402., 532., 111.,
          364.,
                 228., 225., 334., 326., 121., 432., 580., 581., 513.,  77.,
          315.,
                 567., 560., 266., 231., 262., 331., 558., 268., 193., 172.,
          217.,
                 251., 291.])
```

```
In [430]: cat_enc1['c1'].unique()
```

```
Out[430]: array([40., 10.,  6.,  5., 27.,  7.,  4.,  2.,  1., 60., 14.,  8.,
35.,
       23., 11.,  3., 46., 18., 38., 21., 15., 16., 22., 12., 17.,
48.,
       34.,  9., 13.])
```

In [431]:
```
cat_enc11['c1'].unique()
```

Out[431]:
```
array([2534., 3544., 4554., 5564., 6574.,   75.])
```

In [432]:
```
np.unique(cat_enc_le)
```

Out[432]:
```
array([  0,   1,   2,   3,   4,   5,   6,   7,   8,   9,  10,  11,
12,
        13,  14,  15,  16,  17,  18,  19,  20,  21,  22,  23,  24,
25,
        26,  27,  28,  29,  30,  31,  32,  33,  34,  35,  36,  37,
38,
        39,  40,  41,  42,  43,  44,  45,  46,  47,  48,  49,  50,
51,
        52,  53,  54,  55,  56,  57,  58,  59,  60,  61,  62,  63,
64,
        65,  66,  67,  68,  69,  70,  71,  72,  73,  74,  75,  76,
77,
        78,  79,  80,  81,  82,  83,  84,  85,  86,  87,  88,  89,
90,
        91,  92,  93,  94,  95,  96,  97,  98,  99, 100, 101, 102,
103,
       104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115,
116,
       117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128,
129,
       130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141,
142,
       143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154,
155,
       156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167,
168,
       169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180,
181,
       182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193,
194,
       195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206,
207,
       208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219,
220,
       221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232,
233,
       234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245,
246,
       247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258,
259,
       260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271,
272,
       273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284,
285,
       286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297,
298,
       299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310,
311,
              312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323,
324,
              325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336,
337,
              338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349,
350,
              351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362,
363,
              364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375,
376,
              377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388,
389,
              390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401,
402,
              403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414,
415,
              416, 417, 418, 419])
```

In [433]:
```
np.unique(cat_enc_le1)
```

Out[433]:
```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14,
15, 16,
              17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28])
```

In [434]:
```
np.unique(cat_enc_le11)
```

Out[434]:
```
array([0, 1, 2, 3, 4, 5])
```

In [435]:
```
le.inverse_transform([0, 1, 2])
```

Out[435]:
```
array([40., 41., 42.])
```

In [436]:
```
le1.inverse_transform([0, 1, 2])
```

Out[436]:
```
array([1., 2., 3.])
```

In [437]:
```
le11.inverse_transform([0, 1, 2])
```

Out[437]:
```
array([  75., 2534., 3544.])
```

Кодирование категорий наборами бинарных значений

In [438]:
```
ohe = OneHotEncoder()
cat_enc_ohe = ohe.fit_transform(cat_enc[['c1']])
```

In [439]:
```
ohe1 = OneHotEncoder()
cat_enc_ohe1 = ohe1.fit_transform(cat_enc1[['c1']])
```

```
In [440]: ohe11 = OneHotEncoder()
          cat_enc_ohe11 = ohe11.fit_transform(cat_enc11[['c1']])

In [441]: cat_enc.shape

Out[441]: (1000, 1)

In [442]: cat_enc1.shape

Out[442]: (88, 1)

In [443]: cat_enc11.shape

Out[443]: (88, 1)

In [444]: cat_enc_ohe.shape

Out[444]: (1000, 420)

In [445]: cat_enc_ohe1.shape

Out[445]: (88, 29)

In [446]: cat_enc_ohe11.shape

Out[446]: (88, 6)

In [447]: cat_enc_ohe

Out[447]: <1000x420 sparse matrix of type '<class 'numpy.float64'>'
                  with 1000 stored elements in Compressed Sparse Row format>

In [448]: cat_enc_ohe1

Out[448]: <88x29 sparse matrix of type '<class 'numpy.float64'>'
                  with 88 stored elements in Compressed Sparse Row format>

In [449]: cat_enc_ohe1

Out[449]: <88x29 sparse matrix of type '<class 'numpy.float64'>'
                  with 88 stored elements in Compressed Sparse Row format>
```

In [450]:
```
cat_enc_ohe.todense()[0:10]
```

Out[450]:
```
matrix([[0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 1., ..., 0., 0., 0.],
        ...,
        [0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 0., ..., 0., 0., 0.]])
```

In [451]:
```
cat_enc_ohe1[:45000].todense()[0:10]
```

Out[451]:
```
matrix([[0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0.],
        [0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
        [1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0
., 0.,
          0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]])
```

```
In [452]: cat_enc_ohe11[:45000].todense()[0:10]
```

```
Out[452]: matrix([[0., 1., 0., 0., 0., 0.],
                  [0., 1., 0., 0., 0., 0.],
                  [0., 1., 0., 0., 0., 0.],
                  [0., 1., 0., 0., 0., 0.],
                  [0., 1., 0., 0., 0., 0.],
                  [0., 1., 0., 0., 0., 0.],
                  [0., 0., 1., 0., 0., 0.],
                  [0., 0., 1., 0., 0., 0.],
                  [0., 1., 0., 0., 0., 0.],
                  [0., 1., 0., 0., 0., 0.]])
```

```
In [453]: cat_enc.head(10)
```

Out[453]:

	c1
0	562.0
1	650.0
2	42.0
3	626.0
4	649.0
5	195.0
6	82.0
7	194.0
8	211.0
9	622.0

In [454]:

```
cat_enc1.head(10)
```

Out[454]:

	c1
0	40.0
1	10.0
2	6.0
3	5.0
4	27.0
5	7.0
6	4.0
7	7.0
8	2.0
9	1.0

In [455]:

```
cat_enc11.head(10)
```

Out[455]:

	c1
0	2534.0
1	2534.0
2	2534.0
3	2534.0
4	2534.0
5	2534.0
6	3544.0
7	3544.0
8	2534.0
9	2534.0

Масштабирование данных

In [456]:

```
from sklearn.preprocessing import MinMaxScaler, StandardScaler, Nor
malizer
```

MinMax

In [457]:

```
# data = pd.read_csv('googleplaystore.csv', sep=",")
strategies[0], test_num_impute(strategies[0])
sc1 = MinMaxScaler()
sc1_data = sc1.fit_transform(data[['depth']])
```

In [458]:

```
strategies[0], test_num_impute1(strategies[0])
sc11 = MinMaxScaler()
sc1_data1 = sc11.fit_transform(data1[['agegp']])
```

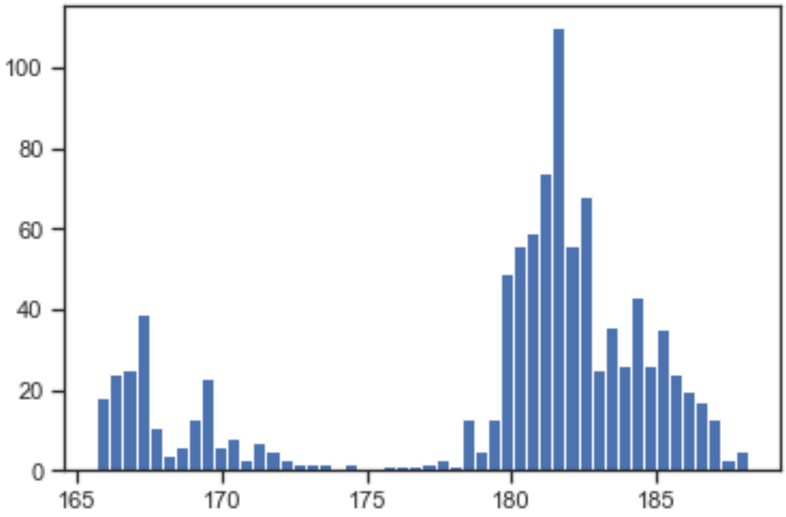
In [459]:

```
strategies[0], test_num_impute1(strategies[0])
sc111 = MinMaxScaler()
sc1_data11 = sc111.fit_transform(data1[['ncases']])
```

In [460]:

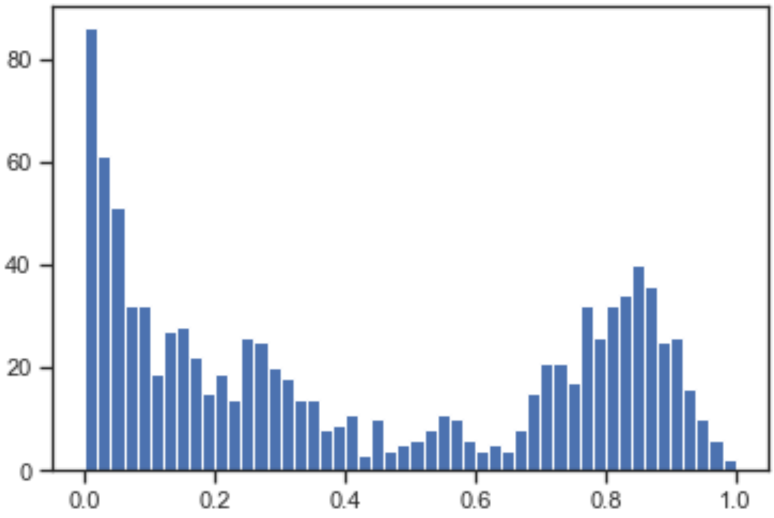
```
plt.hist(data['long'], 50)
plt.show()
```

```
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:839
: RuntimeWarning: invalid value encountered in greater_equal
  keep = (tmp_a >= first_edge)
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:840
: RuntimeWarning: invalid value encountered in less_equal
  keep &= (tmp_a <= last_edge)
```



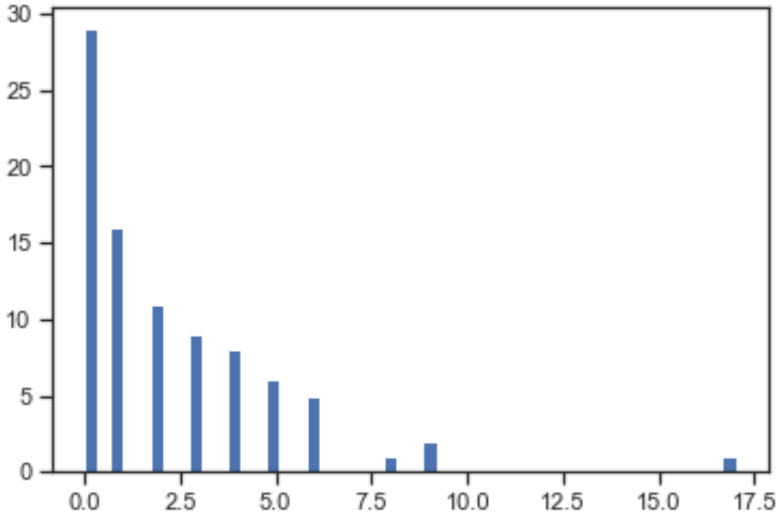
In [461]:

```
plt.hist(sc1_data, 50)
plt.show()
```

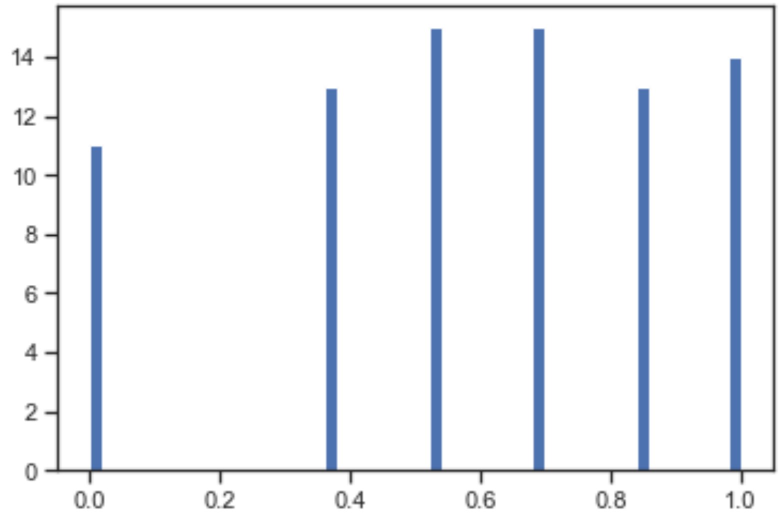


In [462]:

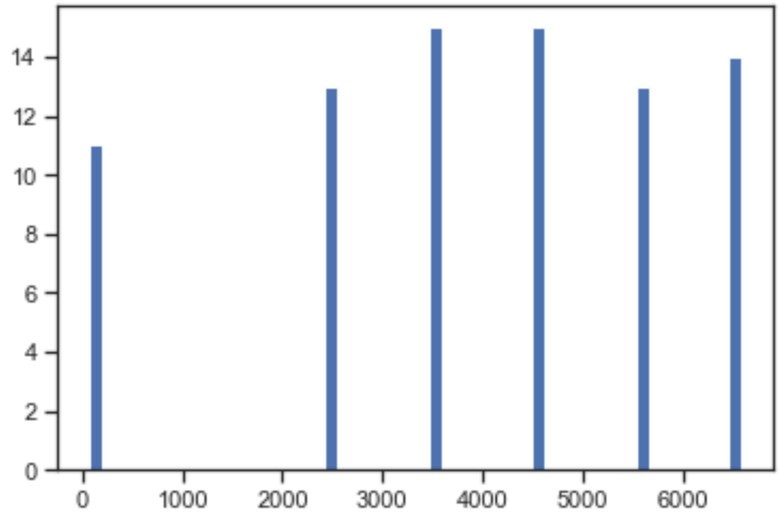
```
plt.hist(data1['ncases'], 50)
plt.show()
```



```
In [463]: plt.hist(sc1_data1, 50)
          plt.show()
```

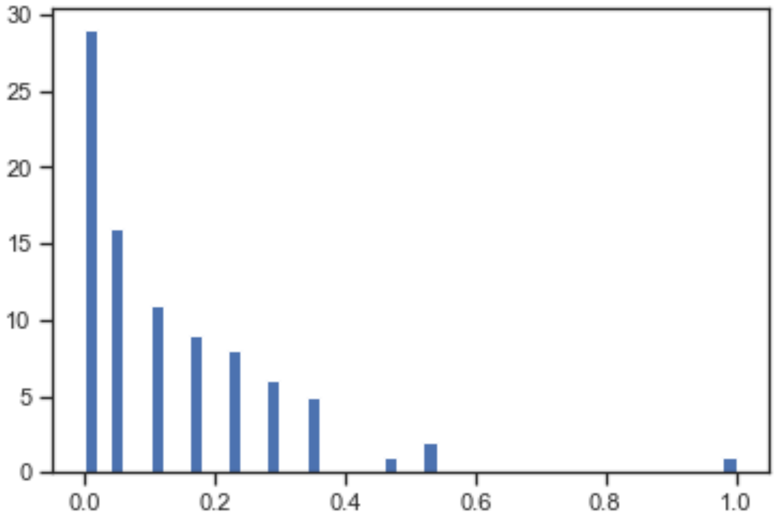


```
In [464]: plt.hist(data1['agegp'], 50)
          plt.show()
```



In [465]:

```
plt.hist(sc1_data11, 50)
plt.show()
```



Z-оценка

In [466]:

```
sc2 = StandardScaler()
sc2_data = sc2.fit_transform(data[['depth']])
```

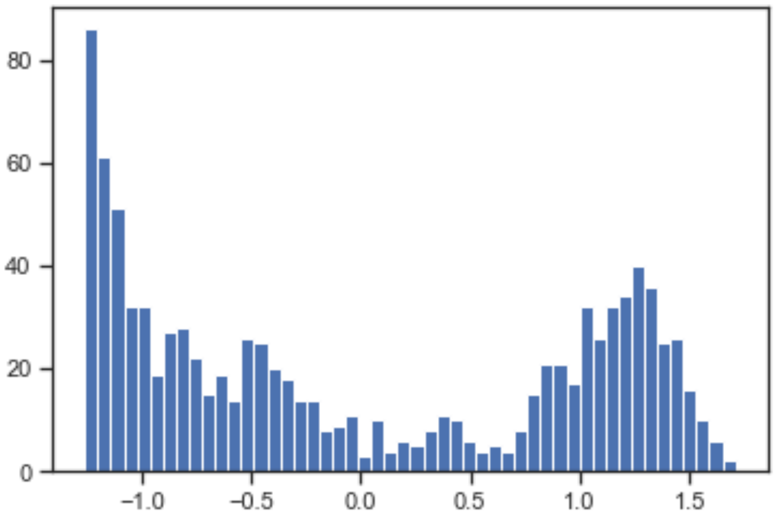
In [467]:

```
sc21 = StandardScaler()
sc2_data1 = sc21.fit_transform(data1[['ncontrols']])
```

In [468]:

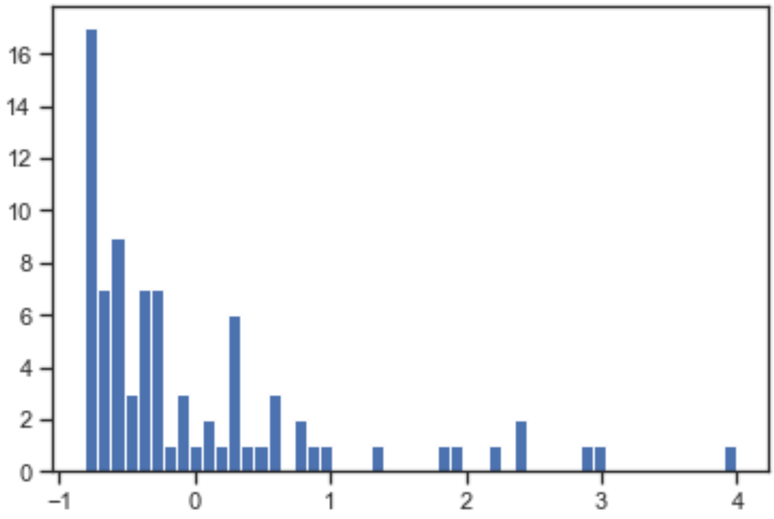
```
plt.hist(sc2_data, 50)
plt.show()
```

```
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:839
: RuntimeWarning: invalid value encountered in greater_equal
  keep = (tmp_a >= first_edge)
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:840
: RuntimeWarning: invalid value encountered in less_equal
  keep &= (tmp_a <= last_edge)
```

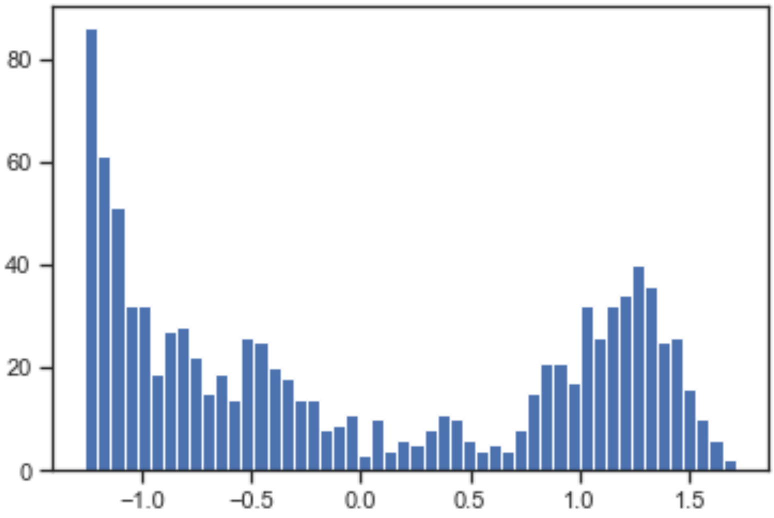


In [469]:

```
plt.hist(sc2_data1, 50)
plt.show()
```



In [470]:
```
plt.hist(sc2_data, 50)
plt.show()
```

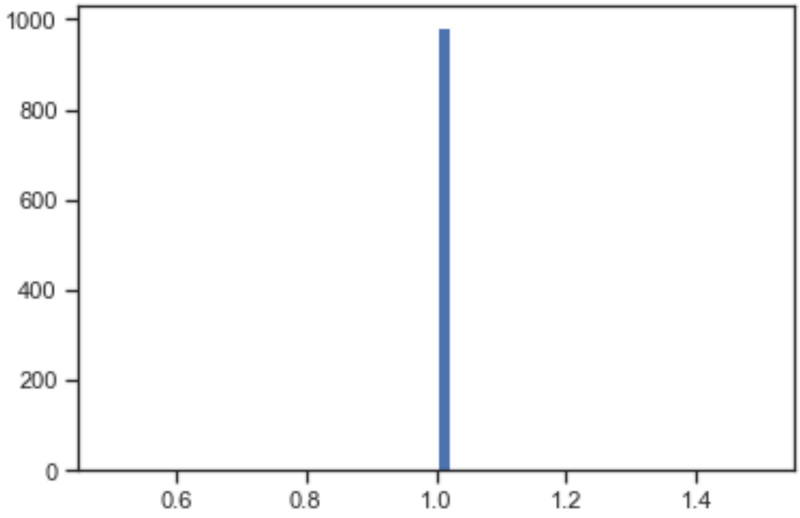


Нормализация

In [471]:
```
sc3 = Normalizer()
sc3_data = sc3.fit_transform(data_new_2[['depth']])
```

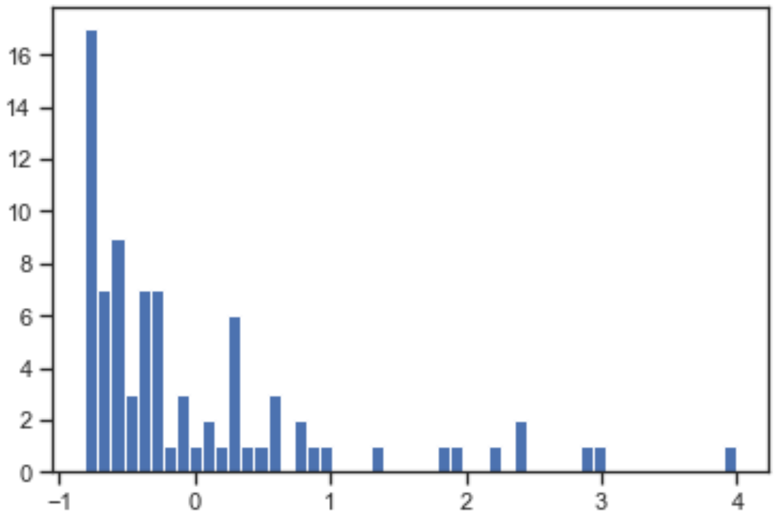
In [472]:
```
sc31 = StandardScaler()
sc3_data1 = sc31.fit_transform(data1[['ncontrols']])
```

In [473]:
```
plt.hist(sc3_data, 50)
plt.show()
```



```
In [474]: plt.hist(sc3_data1, 50)
          plt.show()
```

```
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:839
: RuntimeWarning: invalid value encountered in greater_equal
  keep = (tmp_a >= first_edge)
/usr/local/lib/python3.7/site-packages/numpy/lib/histograms.py:840
: RuntimeWarning: invalid value encountered in less_equal
  keep &= (tmp_a <= last_edge)
```



Вывод:

В процессе выполнения данной работы были изучены методы обработки пропусков в данных, кодирования категориальных признаков и масштабирования данных.

```
In [ ]:
```