Московский государственный технический университет им. Н.Э. Баумана
Факультет «Информатика и системы управления»
Кафедра «Системы обработки информации и управления»

Рубежный контроль №1
«Методы обработки»
по дисциплине «Методы машинного обучения»
Вариант 2

ИСПОЛНИТЕЛЬ:

Болотин Андрей Сергеевич
Группа ИУ5-23М

_____

"__"_____2021 г.

Москва 2021

Задача 2: Для набора данных проведите кодирование одного (произвольного) категориального признака с использованием метода "target (mean) encoding".

In [6]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from category_encoders.target_encoder import TargetEncoder as ce_TargetEncoder
%matplotlib inline
sns.set(style="ticks")
```

```
student = pd.read_csv('C:/archive/StudentsPerformance.csv')
student.head()
```

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

```
data_features = list(zip(
# признаки
[i for i in student.columns],
zip(
    # типы колонок
    [str(i) for i in student.dtypes],
    # проверим есть ли пропущенные значения
    [i for i in student.isnull().sum()]
)))
# Признаки с типом данных и количеством пропусков
data_features
```

```
[('gender', ('object', 0)),
 ('race/ethnicity', ('object', 0)),
 ('parental level of education', ('object', 0)),
 ('lunch', ('object', 0)),
 ('test preparation course', ('object', 0)),
 ('math score', ('int64', 0)),
 ('reading score', ('int64', 0)),
 ('writing score', ('int64', 0))]
```

```
#Добавим целевой признак gender_le
dct = {'female': 0, 'male': 1}
student['gender_le'] = student['gender'].map(dct)
student.head()
```

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score	gender_le
0	female	group B	bachelor's degree	standard	none	72	72	74	0
1	female	group C	some college	standard	completed	69	90	88	0
2	female	group B	master's degree	standard	none	90	95	93	0
3	male	group A	associate's degree	free/reduced	none	47	57	44	1
4	male	group C	some college	standard	none	76	78	75	1

```
ce_TargetEncoder1 = ce_TargetEncoder()
student_MEAN_ENC = ce_TargetEncoder1.fit_transform(student[student.columns.difference(['gender_le'])], student['gender_le'])
student_MEAN_ENC.head()
```

	gender	lunch	math score	parental level of education	race/ethnicity	reading score	test preparation course	writing score
0	0.0	0.489922	72	0.466102	0.452632	72	0.479751	74
1	0.0	0.489922	69	0.477876	0.435737	90	0.486034	88
2	0.0	0.489922	90	0.389831	0.452632	95	0.479751	93
3	1.0	0.467606	47	0.477477	0.595506	57	0.479751	44
4	1.0	0.489922	76	0.477876	0.435737	78	0.479751	75

```
student['race/ethnicity'].unique()
```

```
array(['group B', 'group C', 'group A', 'group D', 'group E'],
      dtype=object)
```

```
student_MEAN_ENC['race/ethnicity'].unique()
```

```
array([0.45263158, 0.43573668, 0.59550562, 0.50763359, 0.50714286])
```

Задача 22: Для набора данных проведите масштабирование данных для одного (произвольного) числового признака с использованием масштабирования по максимальному значению.

```
#Оставим только числовые признаки
X_AL = student.drop(student.columns[[0,1,2,3,4]], axis = 1)
X_ALL = X_AL.drop('gender_le', axis = 1)
X_ALL.head()
```

	math score	reading score	writing score
0	72	72	74
1	69	90	88
2	90	95	93
3	47	57	44
4	76	78	75

```
# Функция для восстановления датафрейма
# на основе масштабированных данных
def arr_to_df(arr_scaled):
    res = pd.DataFrame(arr_scaled, columns=X_ALL.columns)
    return res
```

```
from sklearn.preprocessing import MaxAbsScaler

cs51 = MaxAbsScaler()
data_cs51_scaled_temp = cs51.fit_transform(X_ALL)
# формируем DataFrame на основе массива
data_cs51_scaled = arr_to_df(data_cs51_scaled_temp)
data_cs51_scaled.describe()
```

	math score	reading score	writing score
count	1000.000000	1000.000000	1000.000000
mean	0.660890	0.691690	0.680540
std	0.151631	0.146002	0.151957
min	0.000000	0.170000	0.100000
25%	0.570000	0.590000	0.577500
50%	0.660000	0.700000	0.690000
75%	0.770000	0.790000	0.790000
max	1.000000	1.000000	1.000000

Дополнительное задание: Построить график «ящик с усами».

```
sns.boxplot( x=student["race/ethnicity"], y=student["math score"] )
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x5f8a530>
```