Рубежный контроль

Вариант: *5*

Номер задачи №1: *5* Номер задачи №2: *25*

B [12]:

```
#!pip install category_encoders
```

B [2]:

```
import pandas as pd
import numpy as np
import seaborn as sns
```

B [3]:

```
df = pd.read_csv('sunshine hours by city.csv')
```

B [4]:

```
df.head()
```

Out[4]:

	Country	City	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
0	Afghanistan	Kabul	177.2	178.6	204.5	232.5	310.3	353.4	356.8	339.7	303.9	282.6
1	Albania	Tirana	124.0	125.0	165.0	191.0	263.0	298.0	354.0	327.0	264.0	218.0
2	Algeria	Algiers	149.0	165.0	202.0	258.0	319.0	318.0	350.0	319.0	237.0	229.0
3	Algeria	Tamanrasset	297.6	275.5	322.4	327.0	328.6	306.0	356.5	331.7	288.0	310.0
4	Angola	Luanda	219.0	208.0	213.0	199.0	233.0	223.0	175.0	150.0	145.0	164.0

Задача 1

Для набора данных проведите кодирование одного (произвольного) категориального признака с использованием метода "one-hot encoding".

B [5]:

```
pd.get_dummies(df[['Country']]).head()
```

Out[5]:

	Country_Afghanistan	Country_Albania	Country_Algeria	Country_Angola	Country_Argentina
0	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	1	0	0
4	0	0	0	1	0

5 rows × 140 columns

B [6]:

```
from category_encoders.one_hot import OneHotEncoder as ce_OneHotEncoder
ce_OneHotEncoder1 = ce_OneHotEncoder()
data_OHE = ce_OneHotEncoder1.fit_transform(df[df.columns.difference(['City'])])
data_OHE.head
data_OHE['City']=df['City']
data_OHE
```

Out[6]:

	Apr	Aug	Country_1	Country_2	Country_3	Country_4	Country_5	Country_6	Country_7	Country
0	232.5	339.7	1	0	0	0	0	0	0	
1	191.0	327.0	0	1	0	0	0	0	0	
2	258.0	319.0	0	0	1	0	0	0	0	
3	327.0	331.7	0	0	1	0	0	0	0	
4	199.0	150.0	0	0	0	1	0	0	0	
...	...	...	...	...	...	...	...	...	...	
377	273.0	319.3	0	0	0	0	0	0	0	
378	246.0	303.8	0	0	0	0	0	0	0	
379	243.0	297.6	0	0	0	0	0	0	0	

Задача 2

Для набора данных для одного (произвольного) числового признака проведите обнаружение и удаление выбросов на основе межквартильного размаха.

B [7]:

```
df = df[(df['Year'] <= np.quantile(df['Year'], 0.75))&(df['Year'] >= np.quantile(df['Year']
```

B [8]:

```
np.quantile(df['Year'], 0.75)
```

Out[8]:

```
2764.025
```

B [9]:

```
df
```

	Country	City	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1	Albania	Tirana	124.0	125.0	165.0	191.0	263.0	298.0	354.0	327.0	264.0	218.0	127.0	88.0
2	Algeria	Algiers	149.0	165.0	202.0	258.0	319.0	318.0	350.0	319.0	237.0	229.0	165.0	136.0
4	Angola	Luanda	219.0	208.0	213.0	199.0	233.0	223.0	175.0	150.0	145.0	164.0	199.0	212.0
5	Argentina	Buenos Aires	279.0	240.8	229.0	220.0	173.6	132.0	142.6	173.6	189.0	227.0	252.0	266.6
6	Argentina	Córdoba	257.3	229.6	204.6	189.0	170.5	150.0	170.5	204.6	213.0	238.7	255.0	251.1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
373	Vietnam	Da Lat	255.0	234.0	255.0	202.0	190.0	147.0	157.0	136.0	133.0	140.0	172.0	215.0
374	Vietnam	Da Nang	139.0	145.0	188.0	209.0	246.0	239.0	253.0	218.0	176.0	145.0	120.0	103.0
376	Vietnam	Ho Chi Minh City	245.0	246.0	272.0	239.0	195.0	171.0	180.0	172.0	162.0	182.0	200.0	226.0

Задача 3

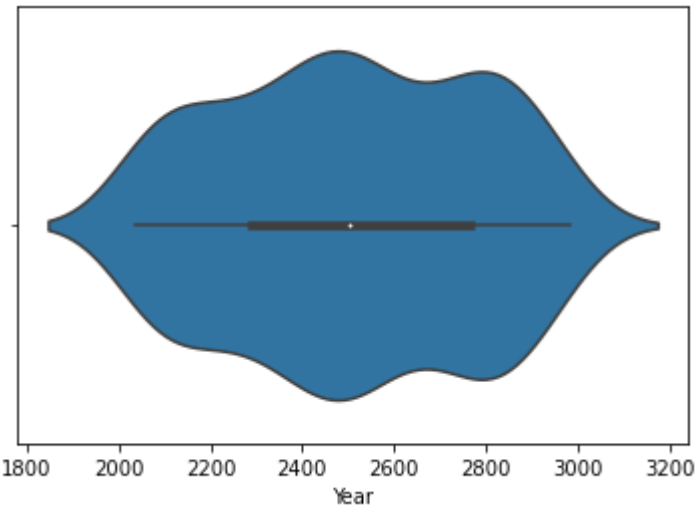
Для студентов группы ИУ5-24М, ИУ5И-24М - для произвольной колонки данных построить график "Скрипичная диаграмма (violin plot)".

B [10]:

```
sns.violinplot(x=df['Year'])
```

Out[10]:

```
<AxesSubplot:xlabel='Year'>
```



B [ ]:

```

```