

## Итоговое задание для курса "Введение в данные"

Импорт библиотек Python

```
In [54]: %matplotlib inline
import matplotlib.pyplot as plt
import pandas as pd
from statsmodels.distributions.empirical_distribution import ECDF
```

Загрузка данных из файла в Dataframe

```
In [16]: data = pd.read_csv("C:/Users/Ice/Downloads/Data_projects.csv", sep=';')
```

Отображение верхних пяти строк таблицы

```
In [17]: data.head()
```

```
Out[17]:
```

	AddressCount	CallsCount	ClicksCount	FirmsCount	GeoPart	MobilePart	UsersCount	Distance	IsGeo
0	156	20	1903	176	0.416104363472785	0.535762483130904	1125	749.966084023684	1
1	17	37	258	20	0.21167832116788	0.430656934365669	157	2289.03242434015	0
2	78	56	1956	185	0.34947538373688	0.476594027441485	1195	1423.37651183958	1
3	14	70	378	19	0.31871838112985	0.463743676222597	206	3396.56608856838	0
4	111	90	4089	90	0.55617545209696	0.490573297422085	2934	1576.51415402623	1

Отображение первичных характеристик данных с помощью метода DataFrame.describe() библиотеки Pandas

```
In [18]: data.describe()
```

```
Out[18]:
```

	AddressCount	CallsCount	ClicksCount	FirmsCount	UsersCount	IsGeo
count	79.000000	79.000000	79.000000	79.000000	79.000000	
mean	1048.037975	3648.683544	21826.012658	305.088608	9753.126582	0.354430
std	1642.066119	8124.105402	32474.959513	382.052090	13927.295721	0.481397
min	9.000000	20.000000	258.000000	14.000000	157.000000	0.000000
25%	81.000000	346.000000	2055.000000	71.500000	1167.500000	0.000000
50%	371.000000	931.000000	6921.000000	185.000000	2934.000000	0.000000
75%	1195.000000	2457.500000	30625.500000	402.500000	13265.000000	1.000000
max	9552.000000	48497.000000	167155.000000	2379.000000	61127.000000	1.000000

Функция describe не отобразила характеристики столбцов 'GeoPart', 'MobilePart' и 'Distance', поэтому проверим характеристики таблицы методом info

```
In [19]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 79 entries, 0 to 78
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype  
--- 
 0   AddressCount    79 non-null int64
 1   CallsCount     79 non-null int64
 2   ClicksCount    79 non-null int64
 3   FirmsCount     79 non-null int64
 4   GeoPart        79 non-null object
 5   MobilePart     79 non-null object
 6   UsersCount     79 non-null int64
 7   Distance       79 non-null object
 8   IsGeo          79 non-null int64
dtypes: int64(6), object(3)
memory usage: 5.6+ KB
```

Как оказалось, данные в столбцах 'GeoPart', 'MobilePart' и 'Distance' отнесены Pandas к типу object (вероятная причина - использование запятой вместо точки в качестве десятичного разделителя). Загрузим данные снова, уточнив разделитель в параметрах метода read\_csv.

```
In [20]: data = pd.read_csv("C:/Users/Ice/Downloads/Data_projects.csv", sep=';', decimal=',', decimal=',')
```

```
In [21]: data.head()
```

```
Out[21]:
```

	AddressCount	CallsCount	ClicksCount	FirmsCount	GeoPart	MobilePart	UsersCount	Distance	IsGeo
0	156	20	1903	176	0.416104	0.535762	1125	749.966084	1
1	17	37	258	20	0.211679	0.430657	157	2289.032424	0
2	78	56	1956	185	0.349475	0.476594	1195	1423.376512	1
3	14	70	378	19	0.318718	0.463744	206	3396.566089	0
4	111	90	4089	90	0.556175	0.490573	2934	1576.514154	1

```
In [22]: data.describe()
```

```
Out[22]:
```

	AddressCount	CallsCount	ClicksCount	FirmsCount	GeoPart	MobilePart	UsersCount	Distance	IsGeo
count	79.000000	79.000000	79.000000	79.000000	79.000000	79.000000	79.000000	79.000000	79.000000
mean	1048.037975	3648.683544	21826.012658	305.088608	0.342645	0.445746	9753.126582	2669.426352	0.354430
std	1642.066119	8124.105402	32474.959513	382.052090	0.103608	0.146131	13927.295721	1427.785304	0.481397
min	9.000000	20.000000	258.000000	14.000000	0.092917	0.090000	157.000000	714.787236	0.000000
25%	81.000000	346.000000	2055.000000	71.500000	0.281527	0.357293	1167.500000	1562.103740	0.000000
50%	371.000000	931.000000	6921.000000	185.000000	0.322342	0.463744	2934.000000	2586.503274	0.000000
75%	1195.000000	2457.500000	30625.500000	402.500000	0.416907	0.551654	13265.000000	3575.692331	1.000000
max	9552.000000	48497.000000	167155.000000	2379.000000	0.556175	0.737288	61127.000000	6292.207311	1.000000

Теперь метод describe отображает параметры для всех столбцов таблицы.

Метод describe содержит информацию о следующих мерах центра и мерах разброса данных:

count - количество значений в столбце (при отсутствии пропущенных значений должно быть равно количеству объектов в выборке);

mean - среднее арифметическое значение;

std - среднеквадратическое отклонение;

min - минимальное значение показателя в выборке;

25% - 25- процентный квартиль;

50% - 50- процентный квартиль, он же медиана;

75% - 75- процентный квартиль;

max - максимальное значение показателя в выборке.

С помощью других методов Pandas мы можем также найти несмещенную дисперсию и медианное абсолютное отклонение.

```
In [33]: # Показатель несмещенной дисперсии
data.var()
```

```
Out[33]:
```

```
AddressCount    2.496239e+06
CallsCount      6.600109e+07
ClicksCount     1.054632e+09
FirmsCount      1.459638e+05
GeoPart         1.073454e-02
MobilePart      2.135416e-02
UsersCount      1.939696e+08
Distance        2.038571e+06
IsGeo           2.317429e-01
dtype: float64
```

```
In [35]: # Показатель медианного абсолютного отклонения
data.mad()
```

```
Out[35]:
```

```
AddressCount    1996.430708
CallsCount      4264.819748
ClicksCount     22880.270469
FirmsCount      249.612562
GeoPart         0.083412
MobilePart      0.119069
UsersCount      10000000000.0
Distance        1149.136977
IsGeo           0.457619
dtype: float64
```

Для более подробного рассмотрения выберем показатель 'Доля трафика с карты', представленный в столбце 'GeoPart'.

Приведем описательные статистики для данного показателя:

```
In [109]: print("Минимальное значение показателя: ", data['GeoPart'].min())
print("Максимальное значение показателя: ", data['GeoPart'].max())
print("Размах значений показателя: ", data['GeoPart'].max() - data['GeoPart'].min())
print("Среднее арифметическое значение показателя: ", data['GeoPart'].mean())
print("Медианное значение показателя: ", data['GeoPart'].median())
print("25-процентный квартиль: ", data['GeoPart'].quantile(0.25))
print("75-процентный квартиль: ", data['GeoPart'].quantile(0.75))
print("Среднеквадратическое отклонение: ", data['GeoPart'].std())
print("Несмещенная дисперсия: ", data['GeoPart'].var())
print("Медианное абсолютное отклонение: ", data['GeoPart'].mad())
print("Межквартильный размах: ", data['GeoPart'].quantile(0.75) - data['GeoPart'].quantile(0.25))
```

Минимальное значение показателя: 0.09791665666666667

Максимальное значение показателя: 0.55617545209696

размах значений показателя: 0.463258785309325

Среднее арифметическое значение показателя: 0.34264468042316774

Медианное значение показателя: 0.32234151329243393

25- процентный квартиль: 0.2815269423639255

75- процентный квартиль: 0.4169067811429205

Среднеквадратическое отклонение: 0.1036745369486576655

Несмещенная дисперсия: 0.0834158161310719

Медианное абсолютное отклонение: 0.135379838778995

Межквартильный размах: 0.2815269423639255 - 0.32234151329243