

## 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	0.211678832116788	0.430656934306569	157	2289.03242434015	0
2	78	56	1956	185	0.349475383373688	0.476594027441485	1195	1423.37651183958	1
3	14	70	378	19	0.318718381112985	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	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):
AddressCount    79 non-null int64
CallsCount      79 non-null int64
ClicksCount     79 non-null int64
FirmsCount      79 non-null int64
GeoPart         79 non-null object
MobilePart      79 non-null object
UsersCount      79 non-null int64
Distance        79 non-null object
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 read\_csv.

```
In [20]: data = pd.read_csv("C:/Users/Ice/Downloads/Data_projects.csv", sep=';', 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	714.787238	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    2.696381e+06
CallsCount      6.600109e+07
ClicksCount     1.054623e+09
FirmsCount      1.459638e+05
GeoPart         1.073454e-02
MobilePart      2.135434e-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    1096.430700
CallsCount      4264.819740
ClicksCount     22880.270469
FirmsCount      249.612562
GeoPart         0.083412
MobilePart      0.116609
UsersCount      10080.147733
Distance        1149.126977
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:

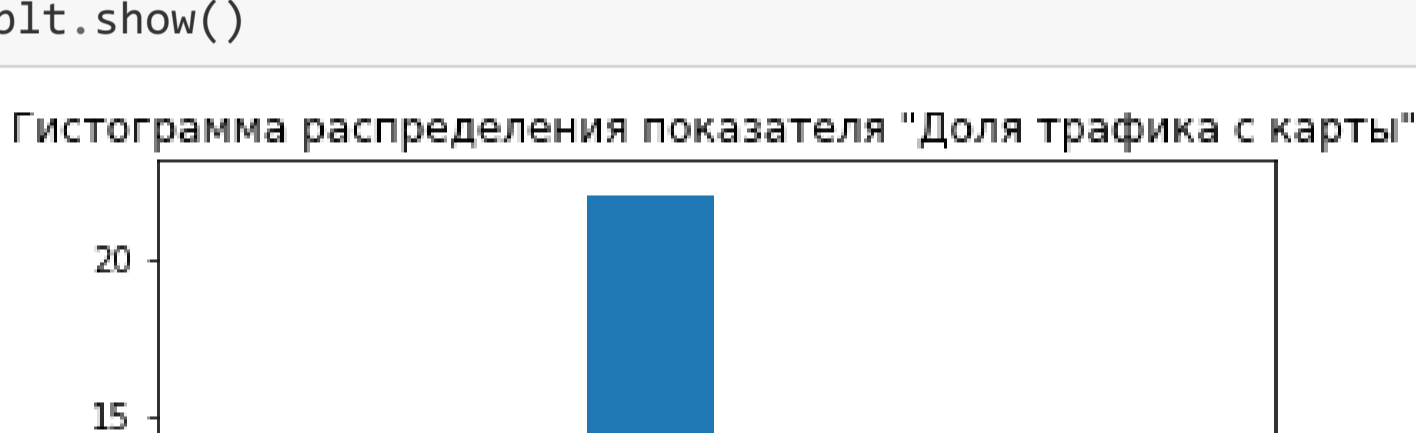
```
In [109]: print("Минимальное значение показателя: ", data['GeoPart'].min())
print("Максимальное значение показателя: ", data['GeoPart'].max())
print("Размах значений показателя: ", data['GeoPart'].max() - data['GeoPart'].min())
print("Среднее арифметическое значение показателя: ", data['GeoPart'].mean())
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: 0.0929166666666667
Максимальное значение показателя: 0.55617545209696
Размах значений показателя: 0.46325878543029325
Среднее арифметическое значение показателя: 0.34264460842316774
Медианное значение показателя: 0.32234151329243393
25-процентный квартиль: 0.2815269423639255
75-процентный квартиль: 0.4169067811429295
Среднеквадратическое отклонение: 0.10360760486576655
Несмещенная дисперсия: 0.010734535786020813
Медианное абсолютное отклонение: 0.08341158161310719
Межквартильный размах: 0.135379838778995
```

Исходя из полученных статистик можем отметить достаточную близость значений среднего арифметического и медианы, однако можно предположить, что распределение может быть немного смещено вправо.

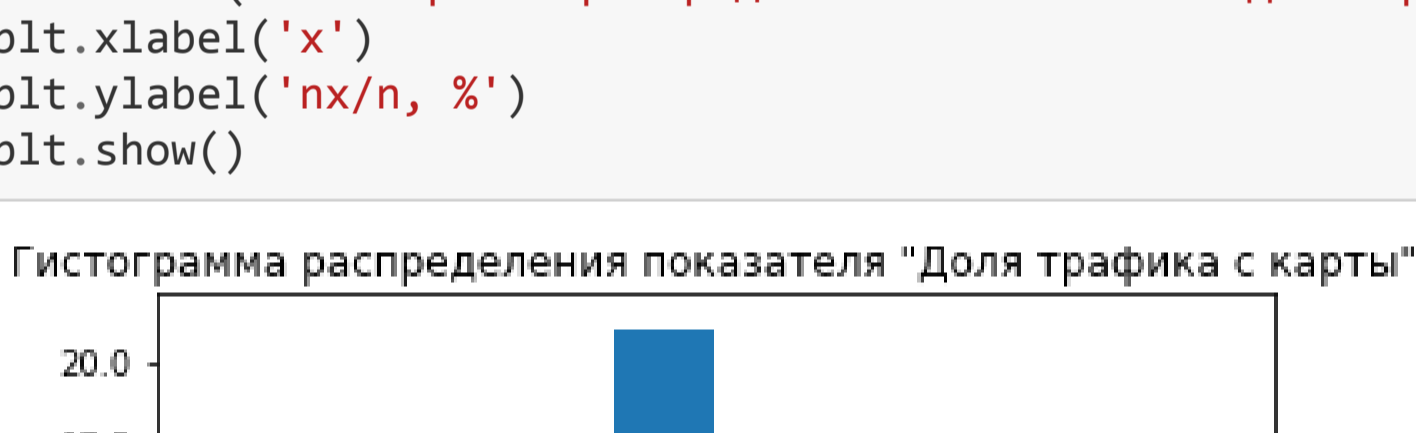
Построение графиков начнем с построения гистограммы признака, в качестве способа определения количества столбцов укажем метод Фридмана-Диакониса.

```
In [39]: plt.hist(data['GeoPart'], bins='fd')
plt.title("Гистограмма распределения показателя \"Доля трафика с карты\"")
plt.xlabel('x')
plt.ylabel('nx/n, %')
plt.show()
```



Также построим гистограмму признака, задав количество столбцов вручную как 10.

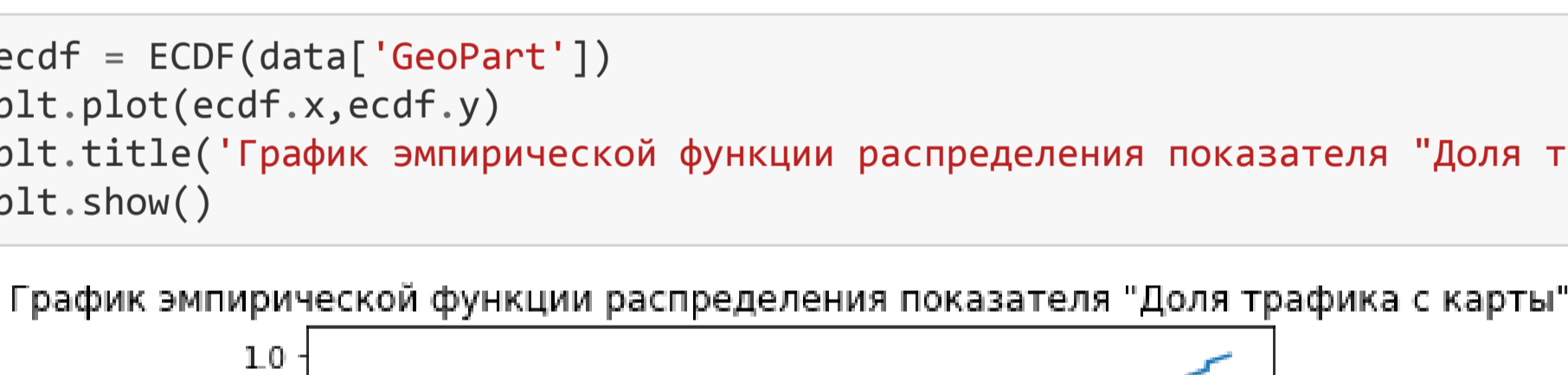
```
In [52]: plt.hist(data['GeoPart'], bins=10)
plt.title("Гистограмма распределения показателя \"Доля трафика с карты\"")
plt.xlabel('x')
plt.ylabel('nx/n, %')
plt.show()
```



Из гистограмм видно, что распределение показателя в целом достаточно похоже на нормальное, однако имеется аномалия в правой части гистограммы - наиболее высокие значения встречаются аномально часто, что может свидетельствовать о большом количестве выбросов, либо о смешении двух выборок с различными характеристиками.

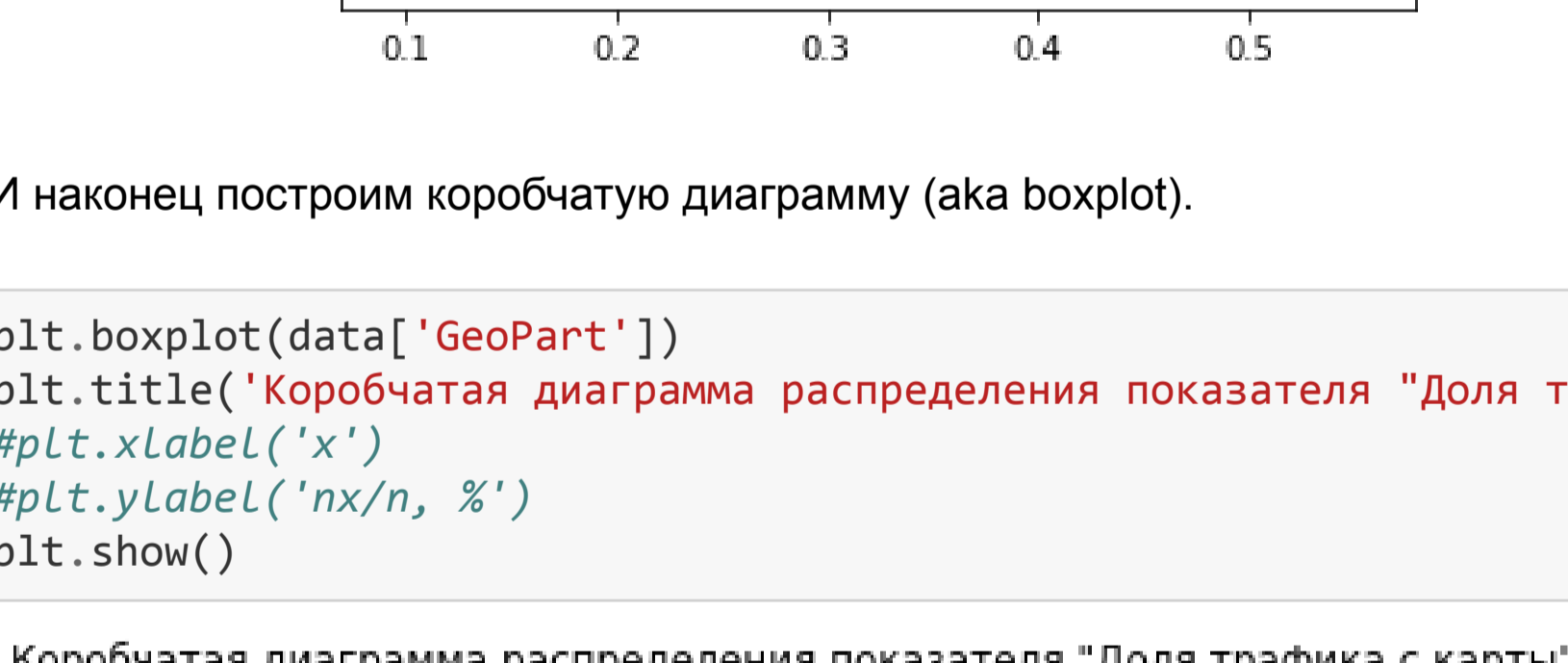
Также построим график эмпирической функции распределения.

```
In [56]: ecdf = ECDF(data['GeoPart'])
plt.plot(ecdf.x,ecdf.y)
plt.title("График эмпирической функции распределения показателя \"Доля трафика с карты\"")
plt.show()
```



И наконец построим коробчатую диаграмму (aka boxplot).

```
In [59]: plt.boxplot(data['GeoPart'])
plt.title("Коробчатая диаграмма распределения показателя \"Доля трафика с карты\"")
#plt.xlabel('x')
#plt.ylabel('nx/n, %')
plt.show()
```

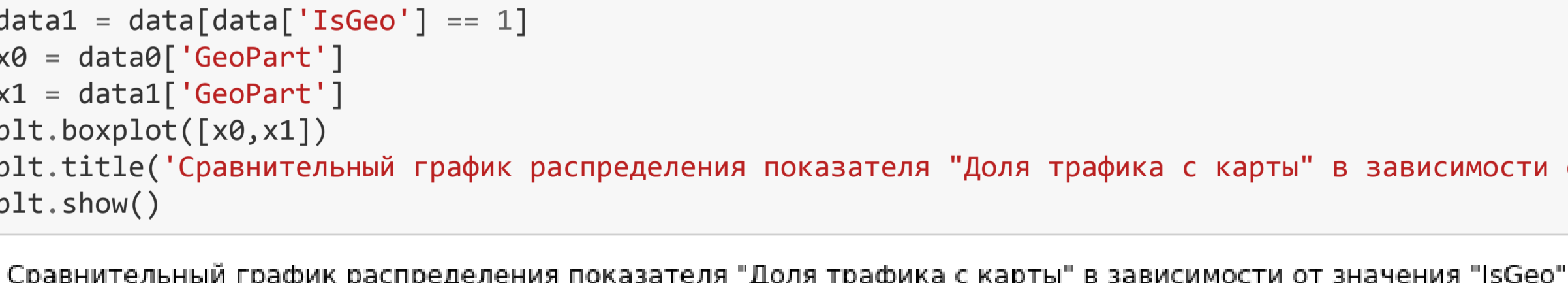


Анализ коробчатой диаграммы показывает отсутствие выбросов, что позволяет предположить, что причиной аномалии на гистограмме является смешение двух выборок с разными характеристиками.

Выделим гипотезу о влиянии либо корреляции параметра геоэависимости и значений показателя "Доля трафика с карты".

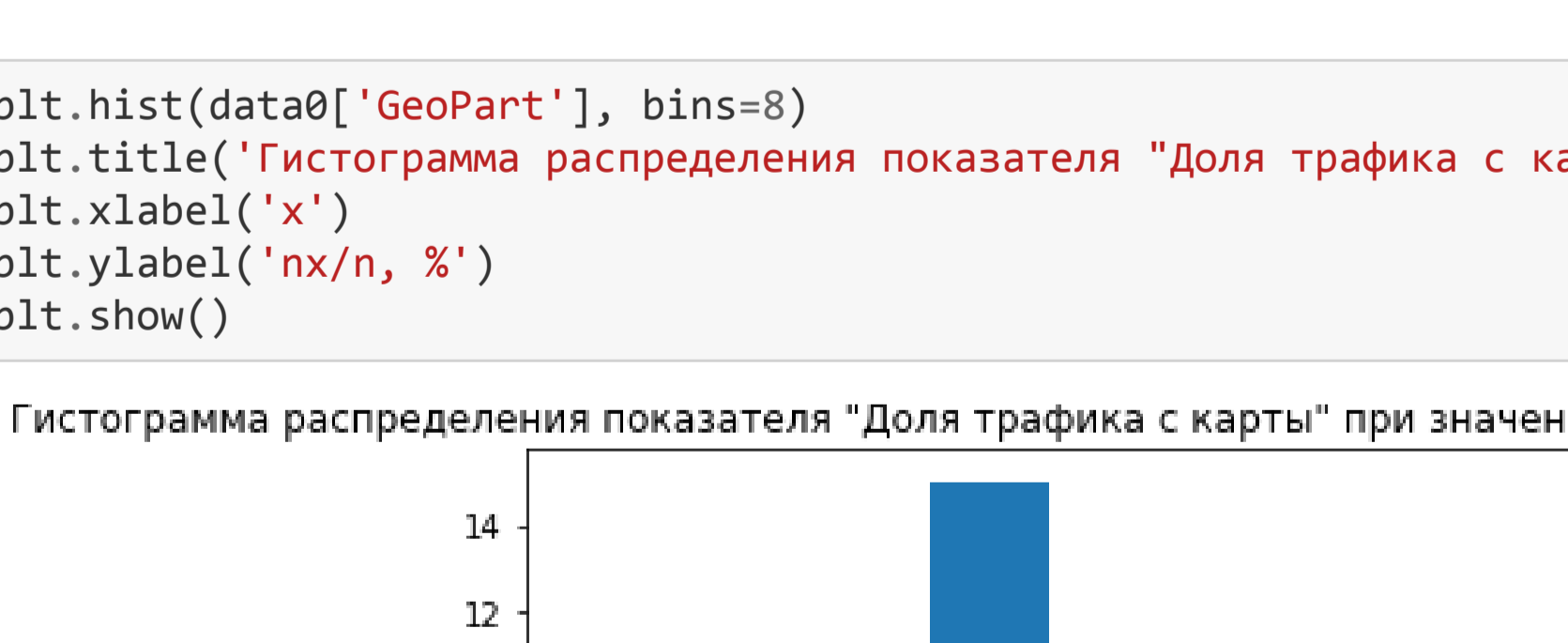
Для проверки данной гипотезы разобьем данные из столбца 'GeoPart' на две группы в зависимости от значений показателя в столбце 'IsGeo' и построим сравнительный график, а также отдельные гистограммы для каждой группы.

```
In [68]: data0 = data[data['IsGeo'] == 0]
data1 = data[data['IsGeo'] == 1]
x0 = data0['GeoPart']
x1 = data1['GeoPart']
plt.boxplot([x0,x1])
plt.title("Сравнительный график распределения показателя \"Доля трафика с карты\" в зависимости от значения \"IsGeo\"")
plt.show()
```



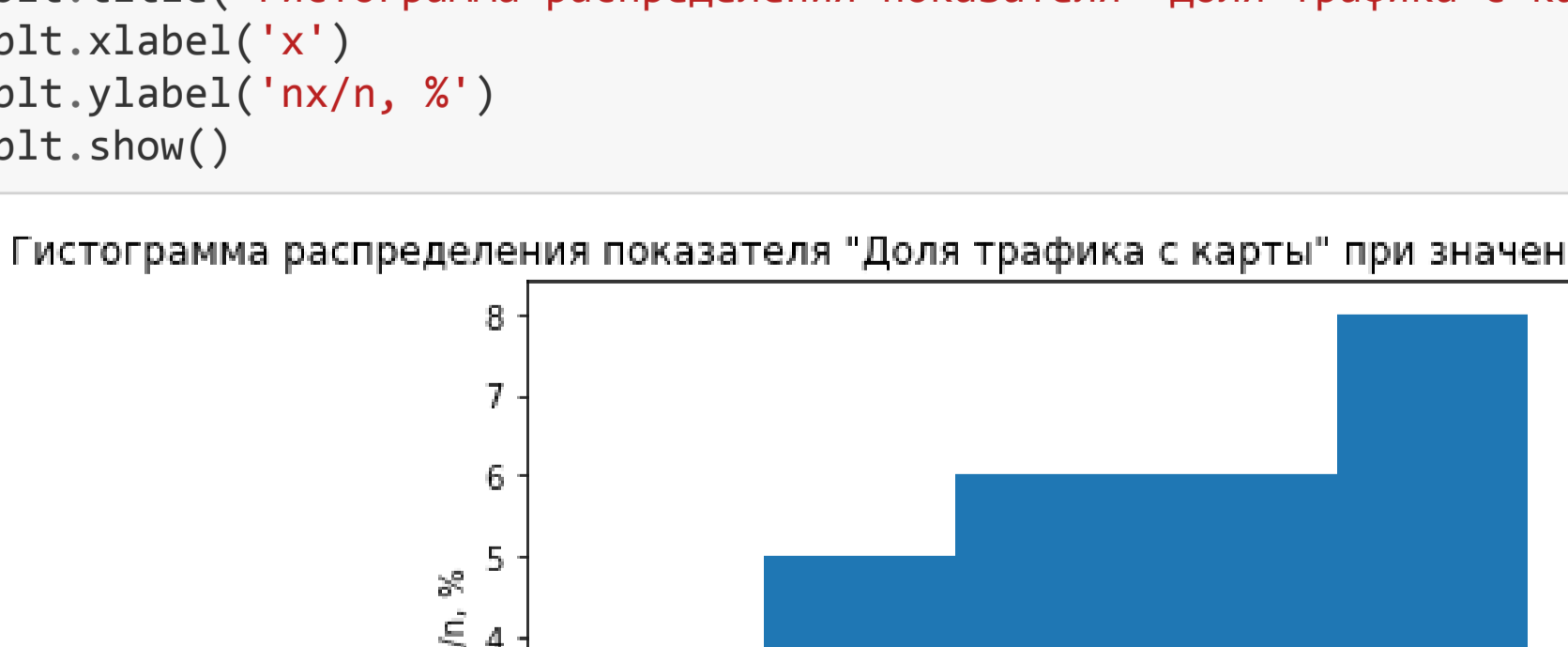
Сравнение коробчатых графиков позволяет сделать вывод о наличии различия в характеристиках выборок в зависимости от значения параметра геоэависимости. Значения второй выборки (IsGeo = 1) заметно смещены в сторону увеличения.

```
In [73]: plt.hist(data0['GeoPart'], bins=8)
plt.title("Гистограмма распределения показателя \"Доля трафика с карты\" при значении показателя \"IsGeo\" = 0")
plt.xlabel('x')
plt.ylabel('nx/n, %')
plt.show()
```



Распределение показателя "Доля трафика с карты" при IsGeo = 0 значительно ближе к нормальному и не содержит каких-либо явно выраженных аномалий.

```
In [81]: plt.hist(data1['GeoPart'], bins=5)
plt.title("Гистограмма распределения показателя \"Доля трафика с карты\" при значении показателя \"IsGeo\" = 1")
plt.xlabel('x')
plt.ylabel('nx/n, %')
plt.show()
```



Распределение показателя "Доля трафика с карты" при IsGeo = 1 имеет ненормальную форму и заметно смещено вправо, однако сильная неправильность формы может быть вызвана слишком малым количеством объектов в выборке.

Также мы можем с достаточной уверенностью сказать, что аномалия на общей гистограмме была вызвана именно наличием смеси двух выборок с разными характеристиками.

Также сравним описательные статистики по двум группам:

```
In [110]: print("Минимальное значение показателя: ", " IsGeo = 0: ", data0['GeoPart'].min(), " IsGeo = 1: ", data1['GeoPart'].min())
print("Максимальное значение показателя: ", " IsGeo = 0: ", data0['GeoPart'].max(), " IsGeo = 1: ", data1['GeoPart'].max())
print("Размах значений показателя: ", " IsGeo = 0: ", data0['GeoPart'].max() - data0['GeoPart'].min(), " IsGeo = 1: ", data1['GeoPart'].max() - data1['GeoPart'].min())
print("Среднее арифметическое значение показателя: ", " IsGeo = 0: ", data0['GeoPart'].mean(), " IsGeo = 1: ", data1['GeoPart'].mean())
print("Медианное значение показателя: ", " IsGeo = 0: ", data0['GeoPart'].median(), " IsGeo = 1: ", data1['GeoPart'].median())
print("25-процентный квартиль: ", " IsGeo = 0: ", data0['GeoPart'].quantile(0.25), " IsGeo = 1: ", data1['GeoPart'].quantile(0.25))
print("75-процентный квартиль: ", " IsGeo = 0: ", data0['GeoPart'].quantile(0.75), " IsGeo = 1: ", data1['GeoPart'].quantile(0.75))
print("Среднеквадратическое отклонение: ", " IsGeo = 0: ", data0['GeoPart'].std(), " IsGeo = 1: ", data1['GeoPart'].std())
print("Несмещенная дисперсия: ", " IsGeo = 0: ", data0['GeoPart'].var(), " IsGeo = 1: ", data1['GeoPart'].var())
print("Медианное абсолютное отклонение: ", " IsGeo = 0: ", data0['GeoPart'].mad(), " IsGeo = 1: ", data1['GeoPart'].mad())
print("Межквартильный размах: ", " IsGeo = 0: ", data0['GeoPart'].quantile(0.75)-data0['GeoPart'].quantile(0.25), " IsGeo = 1: ", data1['GeoPart'].quantile(0.75)-data1['GeoPart'].quantile(0.25))

Минимальное значение показателя: IsGeo = 0: 0.0929166666666667 IsGeo = 1: 0.20337477797513298
Максимальное значение показателя: IsGeo = 0: 0.5317415412470301 IsGeo = 1: 0.5317415412470301
Размах значений показателя: IsGeo = 0: 0.4388248745805634 IsGeo = 1: 0.352800674121827
Среднее арифметическое значение показателя: IsGeo = 0: 0.30949769216316213 IsGeo = 1: 0.4030193258179959
Медианное значение показателя: IsGeo = 0: 0.306822769479299 IsGeo = 1: 0.399805674888025
25-процентный квартиль: IsGeo = 0: 0.248035156898952 IsGeo = 1: 0.323504523431353
75-процентный квартиль: IsGeo = 0: 0.36671895541227806 IsGeo = 1: 0.49360841454392556
Среднеквадратическое отклонение: IsGeo = 0: 0.0885843706808095 IsGeo = 1: 0.1030707510674374
Несмещенная дисперсия: IsGeo = 0: 0.007847190728914433 IsGeo = 1: 0.010623579725605648
Медианное абсолютное отклонение: IsGeo = 0: 0.0672826446240868 IsGeo = 1: 0.08679289141533737
Межквартильный размах: IsGeo = 0: 0.11868379671332605 IsGeo = 1: 0.1700958220079026
```

Описательные статистики свидетельствуют о заметной разности значений между выборками: меры центральной тенденции второй выборки значительно выше. Исходя из разницы в описательных статистиках и разнице в графиках, мы можем считать гипотезу о влиянии либо корреляции параметра геоэависимости и значений показателя "Доля трафика с карты" подтвержденной.

## Вывод:

Сравнение описательных статистик и графиков позволяет нам сделать вывод о том, что между показателями "Доля трафика с карты" и "Признак геоэависимости сферы" (IsGeo) существует зависимость либо корреляция. Для геоэависимой сферы распределение значений показателя "Доля трафика с карты" существенно смещено к более высоким значениям, чем для геоэависимой сферы.