Нижегородский государственный университет им. Н.И. Лобачевского

Институт информационных технологий, математики и механики

# *Наглядный вероятностно-статистический анализ данных*

## Практическое задание 3

## «Нахождение статистических числовых характеристик средствами Python»

Пройдакова Екатерина Вадимовна,

доцент кафедры ТВиАД ИИТММ

# Содержание

- Библиотечные функции Python
- Пример: данные о стоимости турпакетов
  - Формат данных
  - Характеристики центрального положения и разброса
  - Обследование распределения данных
- Практическое задание

# 1. БИБЛИОТЕЧНЫЕ ФУНКЦИИ PYTHON

# 1. Библиотечные функции Python

| Характеристика | statistics | numpy | scipy |
|---|---|---|---|
| **Среднее** | mean() | mean() | mean() |
| **Усеченное среднее** | - | - | stats.trim_mean() |
| **Медиана** | median() | median() | median() |
| **Мода** | mode(),<br>multimode() – с версии 3.8 | - | stats.mode() |
| **Дисперсия** | variance() | var() | var() |
| **Среднее квадратическое отклонение** | stdev() | std() | std() |
| **Квантиль** | quantiles() – с версии 3.8 | quantile() | stats.mstats.mquantiles() |
| **Начальный момент** | - | - | stats.moment() |
| **Коэффициент эксцесса** | - | - | stats.kurtosis() |
| **Коэффициент асимметрии** | - | - | stats.skew() |

# 2. ПРИМЕР: ДАННЫЕ О СТОИМОСТИ ТУРПАКЕТОВ

# 2. Пример: данные о стоимости турпакетов
# 2.1 Формат данных

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | | 22622000200030200001 Стоимость турпакетов, реализованных населению | | | | | | |
| 2 | | 2012 г. | 2013 г. | 2014 г. | 2015 г. | 2016 г. | 2017 г. | 2018 г. |
| 3 | Всего | | | | | | | |
| 4 | Российская Федерация | 208117897,60 | 249898028,99 | 243452560,99 | 239554221,80 | 192624377,20 | 281229353,60 | 303737621,49 |
| 5 | Центральный федеральный округ | 82702939,20 | 94748360,99 | 98110495,99 | 110924272,40 | 69591390,70 | 120973363,69 | 130995716,29 |
| 6 | Белгородская область | 727960,50 | 822657 | 916406 | 902520 | 732558,90 | 923334 | 1147512,89 |
| 7 | Брянская область | 476732,50 | 506285 | 487966 | 561214,50 | 500517,10 | 591302,30 | 703124,80 |
| 101 | Гражданам России по территории России | | | | | | | |
| 102 | Российская Федерация | 22745676,70 | 23875845,99 | 25443923 | 50517087,29 | 49165518,09 | 52289929,99 | 59555636,89 |
| 103 | Центральный федеральный округ | 6184777,09 | 6115515 | 5380795 | 24967610 | 10446617,10 | 13196236,20 | 18261917,69 |
| 104 | Белгородская область | 41192,90 | 24773 | 73118 | 209382,50 | 175206,10 | 112518 | 123604,90 |
| 105 | Брянская область | 126868,70 | 123837,99 | 136701 | 81190,80 | 255295,90 | 252913,60 | 199745,20 |
| 199 | Гражданам России по другим странам | | | | | | | |
| 200 | Российская Федерация | 183309232,69 | 222002330,99 | 214308274 | 183970132,09 | 138119905,10 | 222171007,99 | 234054471,49 |
| 201 | Центральный федеральный округ | 75920625,50 | 87963392 | 92316925,99 | 82920954,99 | 56294708,70 | 104035744,49 | 108550368,49 |
| 202 | Белгородская область | 686757,60 | 797755 | 843288 | 693137,50 | 557352,80 | 810816 | 1023908 |
| 203 | Брянская область | 349863,80 | 382447 | 351265 | 480023,70 | 245221,20 | 332367,30 | 503333,60 |
| 297 | Гражданам других стран по территории России | | | | | | | |
| 298 | Российская Федерация | 2062988,20 | 4019851,99 | 3700364 | 5067002,40 | 5338954 | 6768415,60 | 10127513,10 |
| 299 | Центральный федеральный округ | 597536,60 | 669454 | 412775 | 3035707,40 | 2850064,90 | 3741382,99 | 4183430,10 |
| 300 | Белгородская область | 10 | 129 | 0 | 0 | 0 | 0 | 0 |

- Выборочные значения разделены на 4 категории
- Имеются наблюдения за 7 лет (2012-2018 гг.)

# 2. Пример: данные о стоимости турпакетов
# 2.1 Формат данных

- Переводим данные в словарь expenses
- Формат записи:

```
In [81]: expenses['Белгородская область']
Out[81]:
[[727960.5, 822657.0, 916406.0, 902520.0, 732558.9, 923334.0, 1147512.89],
 [41192.9, 24773.0, 73118.0, 209382.5, 175206.1, 112518.0, 123604.9],
 [686757.6, 797755.0, 843288.0, 693137.5, 557352.8, 810816.0, 1023908.0],
 [10.0, 129.0, 0.0, 0.0, 0.0, 0.0, 0.0]]

In [82]: expenses['Белгородская область'][0][2016-2012] # всего по Белгородской области в 2016 году
Out[82]: 732558.9
```

- Подключаем библиотеки для нахождения числовых характеристик

```
31 import statistics
32 import numpy as np
33 from scipy import stats
```

# 2. Пример: данные о стоимости турпакетов
# 2.1 Формат данных

- Формируем срезы данных

```
35 # Массивы данных: года, категории
36 years = np.arange(2012,2019,step=1)
37 categories = range(4)
38
39 # Функция, возвращающая массив стоимостей в млн. руб. по всем областям
40 # для фиксированного года и категории
41 def exp(category, year):
42     res = []
43     for region_data in expenses.values():
44         res.append(region_data[category][year-2012]/1000000)
45     return res
46
47 # Функция, переводящая индекс категории в строку
48 def catToString(category):
49     if category == 0:
50         return "Всего"
51     elif category == 1:
52         return "Гр.РФ по тер.РФ"
53     elif category == 2:
54         return "Гр.РФ по др.странам"
55     elif category == 3:
56         return "Гр.др.стран по тер.РФ"
```

# 2. Пример: данные о стоимости турпакетов
## 2.2 Характеристики центрального положения и разброса

Находим основные числовые характеристики центрального положения и разброса на примере суммарных данным (Категория 0 - Всего) в 2018 году:

```
59 print("Выборочное среднее = " + str(round(statistics.mean(exp(0,2018)), 4)))
60 print("Усеченное среднее = " + str(round(stats.trim_mean(exp(0,2018), 0.1), 4)))
61 print("Выборочная медиана = " + str(round(statistics.median(exp(0,2018)), 4)))
62 print("Выборочная мода для группированных данных = " +
63       str(statistics.mode(round(x) for x in exp(0,2018))))
64 from collections import Counter
65 print("Мультимодальность для группированных данных: " +
66       str(Counter(round(2*x)/2 for x in exp(0,2018)).most_common(2)))
67 print("Выборочная дисперсия = " + str(round(statistics.variance(exp(0,2018)), 4)))
68 print("Среднее квадратическое отклонение = " + str(round(statistics.stdev(exp(0,2018)), 4))
69
```

```
Выборочное среднее = 3.6112
Усеченное среднее = 1.8698
Выборочная медиана = 1.3891
Выборочная мода для группированных данных = 1
Мультимодальность для группированных данных: [(0.5, 14), (1.5, 14)]
Выборочная дисперсия = 134.341
Среднее квадратическое отклонение = 11.5906
```

# 2. Пример: данные о стоимости турпакетов
## 2.2 Характеристики центрального положения и разброса

- Усеченное среднее: меньше влияния выбросов данных
- Наблюдается очевидный «тяжелый» хвост распределения:

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | | 22622000200030200001 Стоимость турпакетов, реализованных населению | | | | | | |
| 2 | | 2012 г. | 2013 г. | 2014 г. | 2015 г. | 2016 г. | 2017 г. | 2018 г. |
| 3 | Всего | | | | | | | |
| 19 | Тамбовская область | 320309,90 | 403161 | 383710 | 356506 | 221577,70 | 367238,50 | 499430,80 |
| 20 | Тверская область | 1136579,10 | 1410133 | 1414155 | 1275488,80 | 813064 | 1526506,40 | 1775544,30 |
| 21 | Тульская область | 695274 | 925618 | 845769 | 546345,40 | 696121,20 | 992542,80 | 1281573,20 |
| 22 | Ярославская область | 590278,90 | 751669 | 545510 | 1354708,50 | 1259140,80 | 1605202,30 | 2459874,40 |
| 23 | Город Москва столица Российской Федерации город федерального значения | 68500672,20 | 76778255,99 | 78283229 | 95963296,49 | 52399873,89 | 101057289,49 | 106831586,40 |

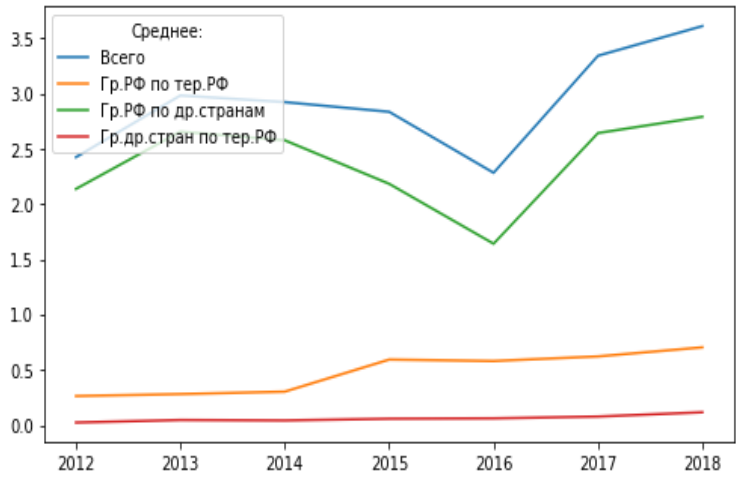
# 2. Пример: данные о стоимости турпакетов
## 2.2 Характеристики центрального положения и разброса

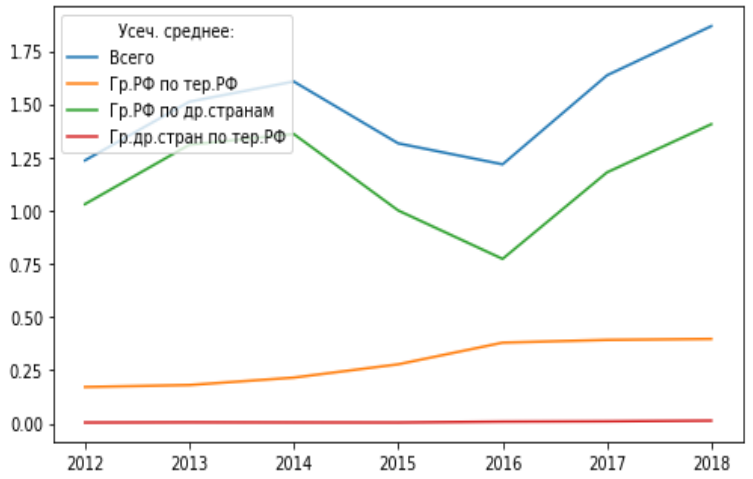
- Динамика числовых характеристик по годам и категориям

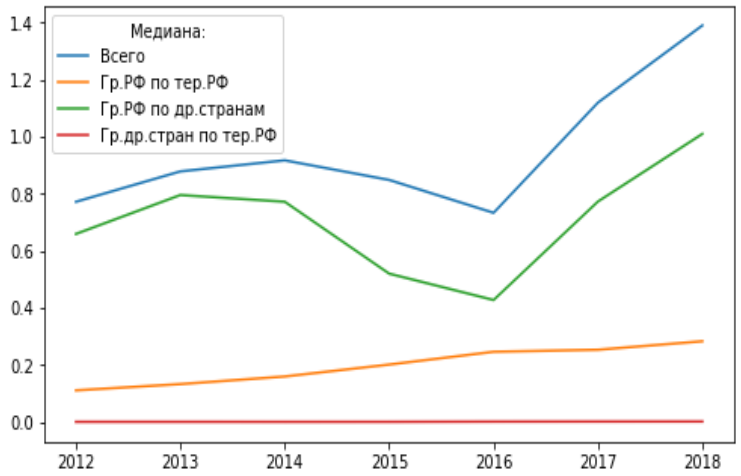
```
from matplotlib import pyplot as plt

# Графики динамики изменения числовых характеристик по годам
fig, axes = plt.subplots(nrows=2, ncols=2, figsize = (18, 10))
for category in categories:
    meanDynamics = [] #среднее
    trimMeanDynamics = [] #усеченное среднее
    medianDynamics = [] #медиана
    varDynamics = [] #дисперсия
    for year in years:
        meanDynamics.append(np.mean(exp(category,year)))
        trimMeanDynamics.append(stats.trim_mean(exp(category,year), 0.1))
        medianDynamics.append(np.median(exp(category,year)))
        varDynamics.append(np.var(exp(category,year)))
    # Отрисовка графиков по точкам
    axes[0][0].plot(years, meanDynamics, label = catToString(category))
    axes[0][1].plot(years, trimMeanDynamics, label = catToString(category))
    axes[1][0].plot(years, medianDynamics, label = catToString(category))
    axes[1][1].plot(years, varDynamics, label = catToString(category))
# Заголовки для легенд
axes[0][0].legend(title='Среднее:', loc = 2)
axes[0][1].legend(title='Усеч. среднее:', loc = 2)
axes[1][0].legend(title='Медиана:', loc = 2)
axes[1][1].legend(title='Дисперсия:', loc = 2)
plt.show() # отображение рисунка на экране
```

# 2. Пример: данные о стоимости турпакетов
## 2.2 Характеристики центрального положения и разброса

- Динамика изменения числовых характеристик по годам и по категориям



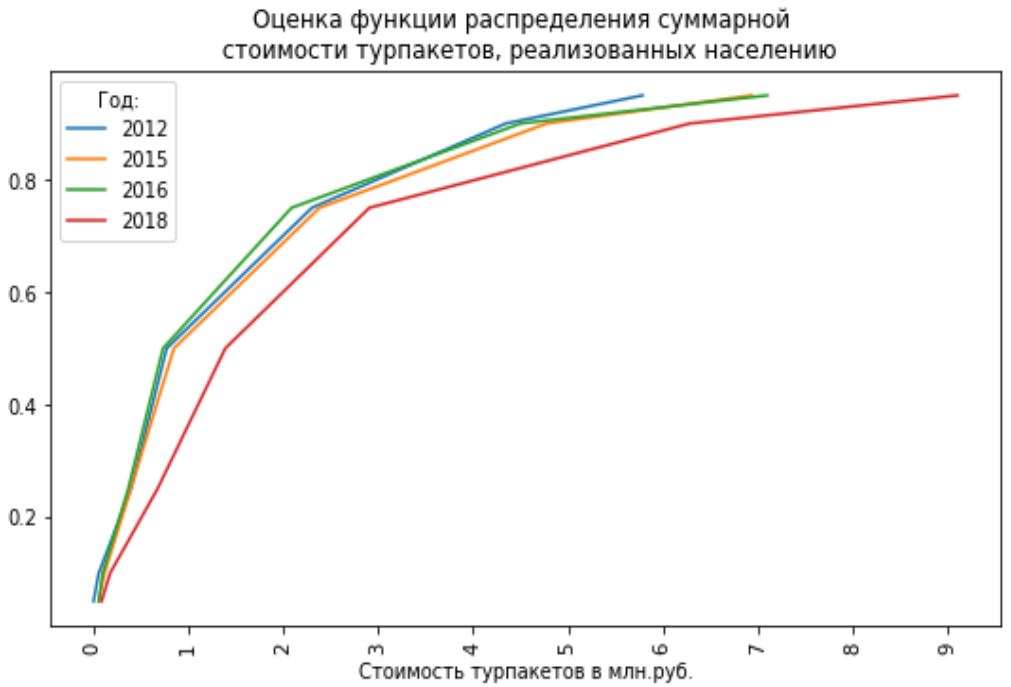




# 2. Пример: данные о стоимости турпакетов
## 2.3 Обследование распределения данных

- Оценка функции распределения

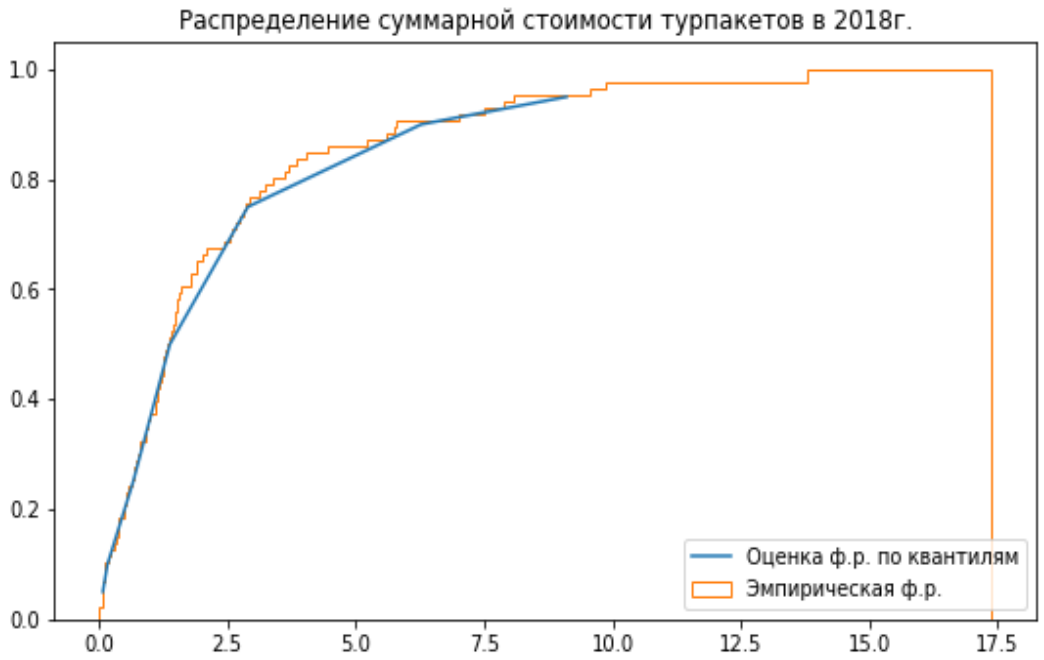
```
95
96 # Массив уровней квантилей
97 qlevels = [0.05, .1, .25, .5, .75, .9, .95]
98 # Функция, возвращающая массив квантилей для данных
99 # по фиксированной категории фиксированного года
100 def quantiles(category, year):
101     res = []
102     for q in qlevels:
103         res.append(np.quantile(exp(category,year), q))
104     return res
```

```
106 plt.figure(figsize = (9, 5))
107 plt.plot(quantiles(0, 2012), qlevels, label='2012')
108 plt.plot(quantiles(0, 2015), qlevels, label='2015')
109 plt.plot(quantiles(0, 2016), qlevels, label='2016')
110 plt.plot(quantiles(0, 2018), qlevels, label='2018')
111 plt.legend(title='Год:')
112 plt.xticks(np.arange(0, quantiles(0, 2018)[-1], step = 1.),
113            rotation = 90) # Метки по горизонтальной оси
114 plt.xlabel("Стоимость турпакетов в млн.руб.") # название горизонтальной оси
115 plt.title("Оценка функции распределения суммарной \n стоимости турпакетов, " +
116           "реализованных населению") # название графика
117 plt.show() # отображение рисунка на экране
```



# 2. Пример: данные о стоимости турпакетов
## 2.3 Обследование распределения данных

- Качественная оценка функции распределения строится по набору квантилей



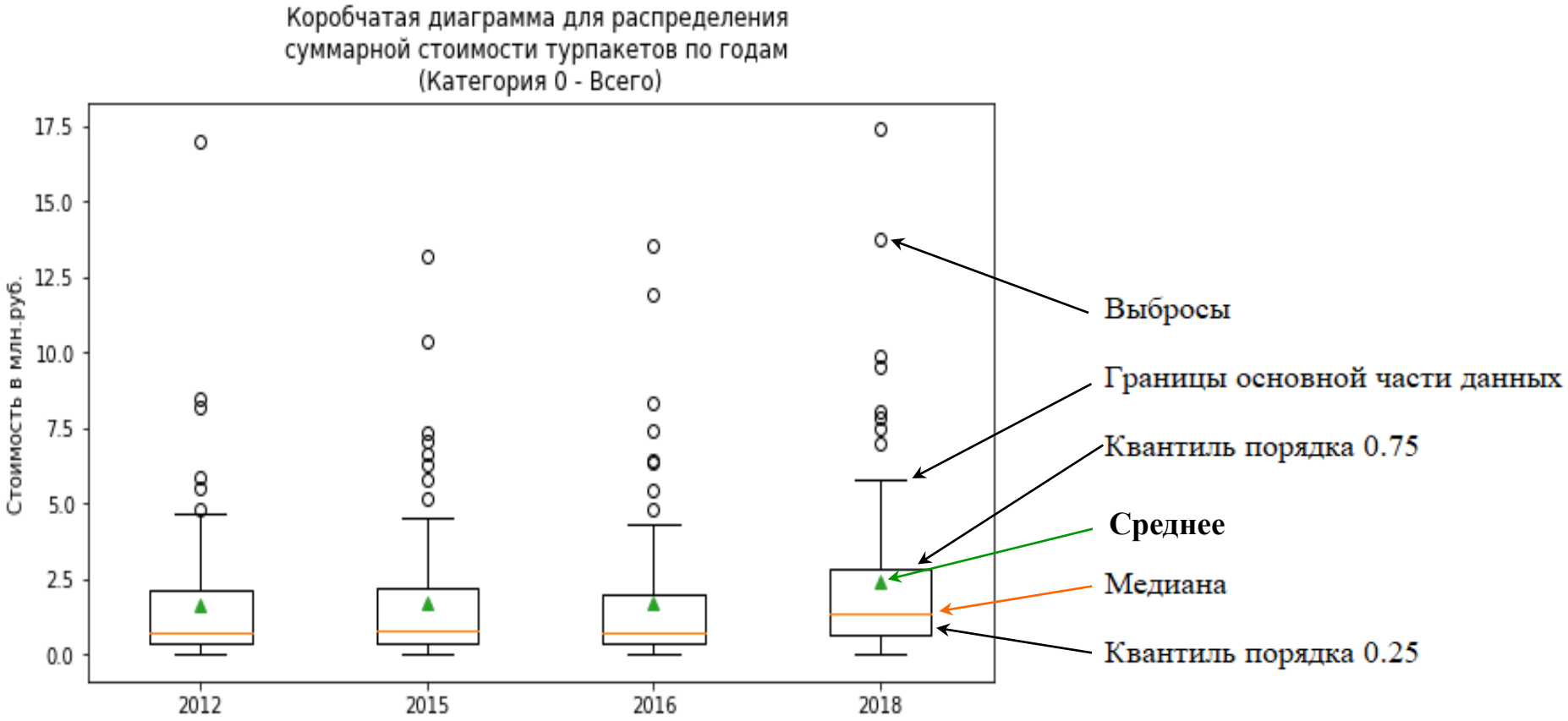

```
118
119 def trimTails(category, year):
120     data = exp(category,year)
121     data.sort()
122     data = data[0:-1]
123     return data
124
125 plt.figure(figsize = (9, 5))
126 plt.plot(quantiles(0, 2018), qlevels, label = "Оценка ф.р. по квантилям")
127 plt.hist(trimTails(0,2018), bins = list(trimTails(0,2018)),
128          density = True, cumulative = True, histtype='step', fill = False,
129          label = "Эмпирическая ф.р.")
130 plt.legend(loc = 4) # Расположение легенды внизу справа
131 plt.title("Распределение суммарной стоимости турпакетов в 2018г.")
132 plt.show()
```

# 2. Пример: данные о стоимости турпакетов
## 2.3 Обследование распределения данных

- **Коробчатая диаграмма (boxplot):** стандарт для визуализации распределения данных

```
134 plt.figure(figsize = (9, 5))
135 plt.boxplot([trimTails(0,2012), trimTails(0,2015), trimTails(0,2016), trimTails(0,2018)],
136              showmeans = True)
137 plt.xticks(np.arange(1, 5, step = 1), labels = [2012, 2015, 2016, 2018])
138 plt.title("Коробчатая диаграмма для распределения \n" +
139           "суммарной стоимости турпакетов по годам \n(Категория 0 - Всего)")
140 plt.ylabel("Стоимость в млн.руб.")
141 plt.show()
142
```

# 2. Пример: данные о стоимости турпакетов
## 2.3 Обследование распределения данных



# 3. СРЕДСТВА MS EXCEL

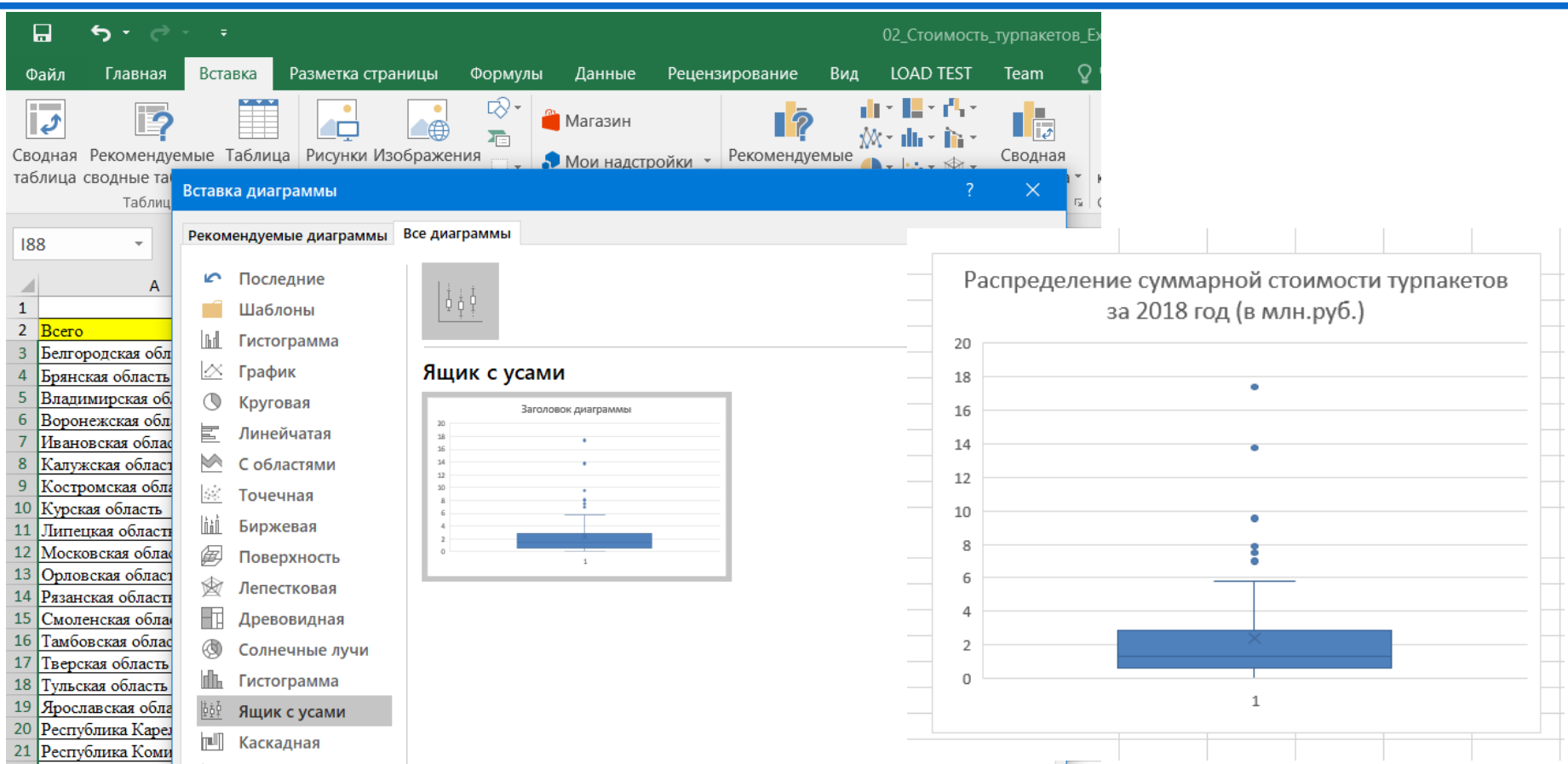
# 3. Средства MS Excel

Простейший анализ данных при помощи числовых характеристик можно провести также и в MS Excel.

Ниже представлены формулы для подсчета основных характеристик в предположении, что выборочные данные находятся в ячейках I4:I90 (суммарные стоимости турпакетов за 2018 г.).

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | | 2018 г. | | | | |
| 2 | Всего | в руб. | в млн.руб. | Характеристика | | Формула |
| 3 | Белгородская область | 1147512,89 | 1,14751289 | Среднее | 3,61 | ОКРУГЛ(СРЗНАЧ(I4:I90); 4) |
| 4 | Брянская область | 703124,80 | 0,7031248 | Медиана | 1,39 | ОКРУГЛ(МЕДИАНА(I4:I90); 4) |
| 5 | Владимирская область | 1545161,40 | 1,5451614 | Дисперсия | 134,341 | ОКРУГЛ(ДИСП.В(I4:I90); 4) |
| 6 | Воронежская область | 977750,50 | 0,9777505 | Ср.кв.откл. | 11,5906 | ОКРУГЛ(КОРЕНЬ(K6); 4) |
| 7 | Ивановская область | 1101764,89 | 1,10176489 | | | |

# 3. Средства MS Excel



# 3. ПРАКТИЧЕСКОЕ ЗАДАНИЕ

# 3. Практическое задание

- Данные – **из практического задания 2**: *02_Автоаварии.xls*

**1. Написать функции для подсчета следующих выборочных числовых характеристик**: a) математическое ожидание, b) медиана, c) усеченное среднее (доля усеченных данных – аргумент функции), d) дисперсия, e) квантиль заданного порядка (уровень квантиля – аргумент функции), f) центральный и начальный момент заданного порядка (порядок – аргумент функции).

Для подсчета каждой характеристики необходима **отдельная функция**. При написании не использовать библиотечные функции вычисления числовых характеристик.

Проверить правильность работы функций, сравнив их выходы с выходами функций библиотек statistics/ numpy/ scipy на примере выборочных данных, полученных по результатам наблюдений за величиной **ξ – видимость дороги в момент совершения аварии (Visibility).**

# 3. Практическое задание

**2. Построить график**, отражающий зависимость среднего значения видимости дороги от степени серьезности аварии. График будет представлять из себя ломанную, построенную по 4 точкам вида ($x$, $y$), где $x$ – степень серьезности аварии, $y$ – среднее значение видимости дороги.

Аналогичный график построить для выборочной медианы и усеченного среднего. Сделать первичные выводы о взаимном влиянии двух указанных величин и наличии выбросов в наблюдениях.

**3. Построить оценки функций распределения** величин $\zeta_1$, $\zeta_2$, $\zeta_3$, $\zeta_4$ – скорость ветра в момент совершения аварии степени серьезности 1, 2, 3, 4 по квантилям порядков {**0.05, 0.1, 0.25, 0.5, 0.75, 0.9, 0.95}.**

Также **построить коробчатые диаграммы** для указанных величин. Определить моду (моды) распределений указанных величин.