Python, обработка на данни и НВО по математика за 7 клас

Идеята

Да използвам данните от НВО по математика за 7-ми клас.

Предположения

1. Учениците от малките населени места имат по-слаби резултати от учениците от големите населени места.
2. Резултатите от 2020 година са по-слаби от 2019 година.

Използван софтуер

1. Anaconda 3
2. Jupyter Notebook

Данните

Данните са предоставени от *Център за оценяване в предучилищното и училищното образование* към МОН.

http://copuo.bg/page.php?c=24&d=19

За Jupyter Notebook

$$(x+y)^n = \binom{n}{0}x^n y^0 + \binom{n}{1}x^{n-1}y^1 + \cdots + \binom{n}{n}x^0 y^n = \sum_{k=0}^{n}\binom{n}{k}x^{n-k}y^k$$

In [1]:

```
%matplotlib inline
```

In [2]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

import pickle

from scipy.stats import ttest_ind
from py_functions import plot_total_points, comparison_statistics
```

Изследване на данните от НВО по математика 7 клас 2019 г.

1. Данни 2019

Данните са предоставени от *Център за оценяване в предучилищното и училищното образование* към МОН [1].

1.1. Подготовка на данните

1.1.1. Прочитане и почистване

In [3]:

```
exam_data = pd.read_excel("data/MAT_7_2019.xlsx", usecols = "A:J")
```

Проверка, дали правилно са заредени.

In [4]:

```
exam_data.head()
```

Out[4]:

	Фиктивен номер	Пол	Код на училище	Училище	Област	Нас. Място	Изпит	Общо точки	Точки закрити въпроси	Точки открити въпроси
0	201441	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	Математика НВО	25.5	25	0.5
1	201442	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	Математика НВО	17.0	17	0.0
2	201443	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	Математика НВО	16.0	16	0.0
3	201444	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	Математика НВО	13.5	11	2.5
4	201445	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	Математика НВО	21.0	18	3.0

In [5]:

```
exam_data.shape
```

Out[5]:

```
(55444, 10)
```

It works: (55444, 10) are the expected numbers.

Only the math exam results are expected to be in the table. A check is carried out anyway.

In [6]:

```
exam_data["Изпит"].unique()
```

Out[6]:

```
array(['Математика НВО'], dtype=object)
```

"Фиктивен номер" и "Изпит" не са ни необходими и можем да ги премахнем.

In [7]:

```
exam_data = exam_data.drop(["Фиктивен номер", "Изпит"], axis = 1)
```

In [8]:

```
exam_data.head()
```

Out[8]:

	Пол	Код на училище	Училище	Област	Нас. Място	Общо точки	Точки закрити въпроси	Точки открити въпроси
0	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	25.5	25	0.5
1	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	17.0	17	0.0
2	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	16.0	16	0.0
3	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	13.5	11	2.5
4	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	21.0	18	3.0

Преименуване на колоните в *пайтъновски стил*.

In [9]:

```
exam_data.columns = ["sex", "school_number", "school_name", "district", "place",
                     "total_points", "points_closed_questions", "points_open_questions"
]
```

In [10]:

```
exam_data.head()
```

Out[10]:

	sex	school_number	school_name	district	place	total_points	points_closed_questions	points_open_questions
0	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	25.5	25	0.5
1	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	17.0	17	0.0
2	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	16.0	16	0.0
3	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	13.5	11	2.5
4	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	ГР.БЛАГОЕВГРАД	21.0	18	3.0

Правим проверка за липсващи данни.

In [11]:

```
exam_data.isna().any().any()
```

Out[11]:

```
True
```

Виждаме, че има такива.

In [12]:

```
exam_data.isna().any()
```

Out[12]:

```
sex                        False
school_number              False
school_name                False
district                   False
place                       True
total_points               False
points_closed_questions    False
points_open_questions      False
dtype: bool
```

Нека да ги погледнем по-отблизо.

In [13]:

```
exam_data.loc[exam_data.place.isna()]
```

Out[13]:

	sex	school_number	school_name	district	place	total_points	points_closed_questions	points_open_questions
4971	Ж	200711	СУ "Иван Вазов"	Бургас	NaN	21.50	17	4.50
7027	Ж	400091	ОУ "Св.Кл.Охридски"	Варна	NaN	42.00	28	14.00
7028	Ж	400091	ОУ "Св.Кл.Охридски"	Варна	NaN	37.75	29	8.75
12205	М	500603	СУ "Христо Ботев“	Видин	NaN	20.00	20	0.00
12370	Ж	602084	ОУ "Васил Левски"	Враца	NaN	12.00	11	1.00
13568	Ж	603058	СУ "Христо Ботев"	Враца	NaN	34.00	21	13.00
13570	Ж	603058	СУ "Христо Ботев"	Враца	NaN	30.50	17	13.50
19340	Ж	1209005	СУ "Д. Маринов"	Монтана	NaN	15.00	15	0.00
19680	Ж	1201200	СУ "Д-р П. Берон"	Монтана	NaN	15.75	14	1.75
19707	Ж	1209005	СУ "Д. Маринов"	Монтана	NaN	16.00	16	0.00
31568	Ж	1806108	ОУ "Алеко Константинов"	Русе	NaN	9.00	9	0.00
33360	М	2000402	СУ"Неофит Рилски"	Сливен	NaN	5.00	5	0.00
34706	М	2000132	ОУ"Св.Климент Охридски" Блатец	Сливен	NaN	47.25	47	0.25
35092	М	2100920	СУ"Олимпийски надежди"	Смолян	NaN	14.50	14	0.50
36362	М	2306517	СУ,,Св.Св.Кирил и Методий	София-област	NaN	25.00	15	10.00
48108	Ж	2400105	СУ "Христо Смирненски"	Стара Загора	NaN	13.00	13	0.00
51126	Ж	2500103	III ОУ " П. Р. Славейков"	Търговище	NaN	39.00	39	0.00
51142	Ж	2500103	III ОУ " П. Р. Славейков"	Търговище	NaN	36.00	36	0.00
51329	М	2611005	ОУ "Христо Ботев "	Хасково	NaN	36.00	33	3.00

	sex	school_number	school_name	district	place	total_points	points_closed_questions	points_open_questions
52433	Ж	2604004	СУ "СВ. КЛИМЕНТ ОХРИДСКИ"	Хасково	NaN	11.00	11	0.00

Всъщност местожителството на тези ученици. Но за това изследване то може да бъде заменено с населеното място, където се намира училището. Нека попълним липсващата информация.

Ще бъдат извлечени номерата на училищата и съответните населени места.

In [14]:

```
school_number_place = exam_data.loc[exam_data.place.isna() == False][["school_number", "place"]]
```

In [15]:

```
school_number_place.shape
```

Out[15]:

```
(55424, 2)
```

Подготвяме речник с номерата на училищата и съответните населени места.

In [16]:

```
dict_school_place = pd.Series(school_number_place.place.values, index = school_number_place.school_
number).to_dict()
```

От оригиналния списък ще бъде създаден нов списък с населените места и липсващите данни ще бъдат запълнени със съответното населено място от речника.

In [17]:

```
new_places_list = []
i = 0
for index, row in exam_data.iterrows():
    if row.isna().any():
        for key,val in dict_school_place.items():
            if key == row.school_number:
                new_places_list.append(val)
                i += 1
    else:
        new_places_list.append(row.place)
print(i)
```

```
20
```

Нека да актуализираме колоната с населените места.

In [18]:

```
exam_data["places_updated"] = new_places_list
exam_data = exam_data.drop(["place"], axis = 1)
exam_data = exam_data.rename(columns={"places_updated": "place"})
```

Проверка за липсващи данни.

In [19]:

```
exam_data.isnull().any().any()
```

Out[19]:

```
False
```

Проверка на типовете данни.

In [20]:

```
exam_data.dtypes
```

Out[20]:

```
sex                          object
school_number                 int64
school_name                  object
district                     object
total_points                float64
points_closed_questions       int64
points_open_questions       float64
place                        object
dtype: object
```

1.1.2. Подготовка на нова колона за големи градове.

Под голям град ще разбираме градовете, които имат повече жители от най-малкия областен град. Необходимите данни са взети от [2]. Информацията е предварително обработена и съхранена във файл, който сега директно ще използваме.

In [21]:

```
large_cities = []
with open('data/output/large_cities_expanded.txt', 'rb') as f:
    large_cities = pickle.load(f)
large_cities
```

Out[21]:

```
['ГР.БЛАГОЕВГРАД',
 'ГР.БУРГАС',
 'ГР.ВАРНА',
 'ГР.ВЕЛИКО ТЪРНОВО',
 'ГР.ГОРНА ОРЯХОВИЦА',
 'ГР.ВИДИН',
 'ГР.ВРАЦА',
 'ГР.ГАБРОВО',
 'ГР.ДОБРИЧ',
 'ГР.КЪРДЖАЛИ',
 'ГР.КЮСТЕНДИЛ',
 'ГР.ДУПНИЦА',
 'ГР.ЛОВЕЧ',
 'ГР.МОНТАНА',
 'ГР.ПАЗАРДЖИК',
 'ГР.ПЕРНИК',
 'ГР.ПЛЕВЕН',
 'ГР.ПЛОВДИВ',
 'ГР.АСЕНОВГРАД',
 'ГР.РАЗГРАД',
 'ГР.РУСЕ',
 'ГР.СИЛИСТРА',
 'ГР.СЛИВЕН',
 'ГР.СМОЛЯН',
 'ГР.СОФИЯ',
 'ГР.СТАРА ЗАГОРА',
 'ГР.КАЗАНЛЪК',
 'ГР.ТЪРГОВИЩЕ',
 'ГР.ХАСКОВО',
 'ГР.ДИМИТРОВГРАД',
 'ГР.ШУМЕН',
 'ГР.ЯМБОЛ']
```

Проверяваме кои са големите градове и добавяме нова колона **large_city** към таблицата, в която пише 0 за малък град и 1 за голям град.

In [22]:

```
large_city_boolean = exam_data["place"].isin(large_cities)
```

```
exam_data["large_city"] = large_city_boolean.astype(int)
```

In [23]:

```
exam_data.sample(10)
```

Out[23]:

	sex	school_number	school_name	district	total_points	points_closed_questions	points_open_questions	place
7280	Ж	400034	СУ "Гео Милев"	Варна	71.25	45	26.25	ГР.ВАРНА
16882	Ж	909705	Основно училище " Васил Левски	Кърджали	29.00	28	1.00	С.КОМУНИГА
5179	М	200213	ОУ " Васил Априлов "	Бургас	66.75	41	25.75	ГР.БУРГАС
51009	Ж	2520501	ОУ "Акад. Даки Йорданов"	Търговище	29.25	27	2.25	ГР.ОМУРТАГ
36556	Ж	2308924	ОУ "Христо Максимов"	София-област	11.00	11	0.00	ГР.САМОКОВ
20699	Ж	1307267	ОУ "Любен Каравелов"	Пазарджик	28.00	24	4.00	ГР.ПАЗАРДЖИК
50218	Ж	2403219	VI ОУ Стара Загора	Стара Загора	81.25	38	43.25	ГР.СТАРА ЗАГОРА
31131	М	1802004	СУ "Панайот Волов"	Русе	24.00	24	0.00	ГР.БЯЛА
13412	М	602045	ОУ "Св.св.Кирил и Методий"	Враца	11.00	11	0.00	С.ТРИ КЛАДЕНЦИ
53163	М	2700062	СУ "Св. Паисий Хилендарски"	Шумен	16.75	12	4.75	С.СТАНЯНЦИ

1.1.3. Подоготовка на нова колона за математически гимназии

Ще претърсим колоната **school_name** за "МГ". Ще създадем нова колона **math_school**, съдържаща 1 за математическа гимназия и 0 за останалите.

In [24]:

```
math_schools_boolean = exam_data.school_name.str.contains("МГ", case = True )
exam_data["math_school"] = math_schools_boolean.astype(int)
```

In [25]:

```
exam_data.sample(10)
```

Out[25]:

	sex	school_number	school_name	district	total_points	points_closed_questions	points_open_questions	place
7510	Ж	400018	ОУ "Панайот Волов"	Варна	23.50	20	3.50	ГР.ВАРНА
31464	М	1806209	СУ "Васил Левски"	Русе	24.50	22	2.50	ГР.РУСЕ
52921	Ж	2601010	ОУ "Свети Иван Рилски"	Хасково	8.25	8	0.25	ГР.ХАСКОВО
21491	М	1304930	ОУ"Хр. Смирненски"	Пазарджик	30.00	21	9.00	ГР.ПАЗАРДЖИК
832	Ж	104017	ОУ "Св.св. Кирил и Методий"	Благоевград	47.25	38	9.25	С.РУПИТЕ
6578	Ж	400047	СУХНИ "Константин Преславски"	Варна	48.00	32	16.00	ГР.ВАРНА

	sex	school_number	school_name	district	total_points	points_closed_questions	points_open_questions	place
34407	М	2000115	СУ "К. Константинов"	Сливен	16.00	16	0.00	ГР.СЛИВЕН
25626	Ж	1602502	ОУ "Христо Ботев"	Пловдив	55.75	27	28.75	ГР.РАКОВСКИ
15490	Ж	800009	ОУ "Хан Аспарух"	Добрич	32.00	25	7.00	ГР.ДОБРИЧ
51627	Ж	2601004	ОУ "Любен Каравелов"	Хасково	18.50	14	4.50	ГР.ХАСКОВО

Така подготвените данни ще запазим в scv файл.

In [26]:

```
exam_data.to_csv(r'data/output/math_exam_2019.csv', index = False)
```

2. Анализ на данните

Нека да погледнем корелационната таблица за данните.

In [27]:

```
exam_data.corr()
```

Out[27]:

	school_number	total_points	points_closed_questions	points_open_questions	large_city	math_school
school_number	1.000000	0.024921	0.019711	0.026494	0.149122	0.007470
total_points	0.024921	1.000000	0.926880	0.947685	0.297514	0.290826
points_closed_questions	0.019711	0.926880	1.000000	0.758574	0.217082	0.219015
points_open_questions	0.026494	0.947685	0.758574	1.000000	0.331850	0.318572
large_city	0.149122	0.297514	0.217082	0.331850	1.000000	0.107262
math_school	0.007470	0.290826	0.219015	0.318572	0.107262	1.000000

Очаквано виждаме силна позитивна корелация между **total_points** и **points_closed_questions**, между **total_points** и **points_open_questions**, и **points_closed_questions** и **points_open_questions**. Има слаба позитивна корелация между **large_city** и **points_open_questions**, както и между **large_city** и **total_points**. Между **math_school** и **points_open_questions**, **math_school** и **points_closed_questions** и **math_school** и **total_points** корелациите също са слабо позитивни. Т.е. няма нищо неочаквано.

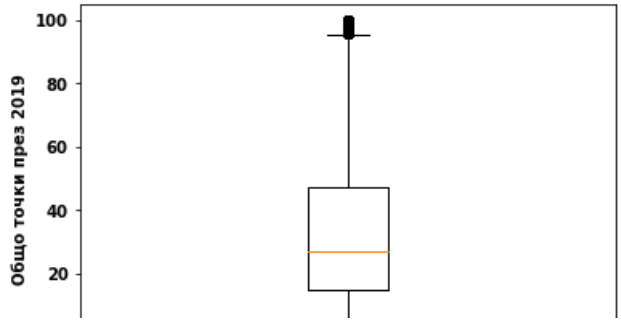
Нека да погледнем статистиката.

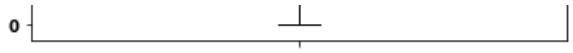
In [28]:

```
plt.boxplot(exam_data.total_points)

plt.xticks([1], [""])
plt.ylabel("Общо точки през 2019")

plt.show()
```





In [29]:

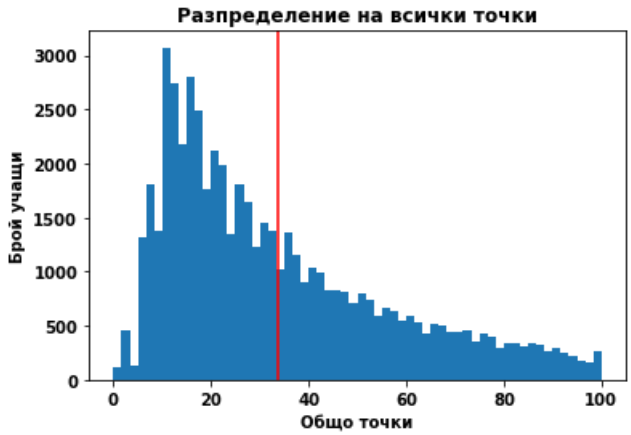
```
print("Общо точки min: {}".format(exam_data.total_points.min()))
print("Общо точки max: {}".format(exam_data.total_points.max()))
print("Общо точки mean: {:.2f}".format(exam_data.total_points.mean()))
print("Общо точки median: {}".format(exam_data.total_points.median()))
```

```
Общо точки min: 0.0
Общо точки max: 100.0
Общо точки mean: 33.61
Общо точки median: 27.0
```

Нека да погледнем цялостното разпределние на точките.

In [30]:

```
plot_total_points(exam_data.total_points, color = "r")
```



Интересни са резултатите с най-много точки. Нека да ги погледнем по-отблизо.

In [31]:

```
best_exams = exam_data[exam_data.total_points >= 95]
```

In [32]:

```
best_exams
```

Out[32]:

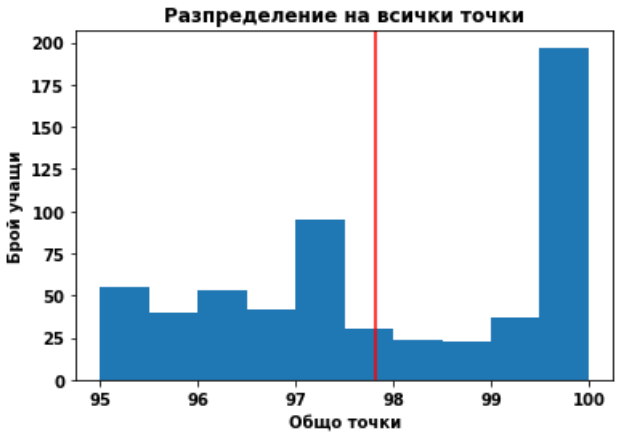
	sex	school_number	school_name	district	total_points	points_closed_questions	points_open_questions	plac
510	Ж	104104	I ОУ "Св. Климент Охридски"	Благоевград	96.25	45	51.25	ГР.САНДАНСК
901	М	109055	СУ "Христо Смирненски"	Благоевград	96.75	47	49.75	С.КОЧА
946	Ж	109055	СУ "Христо Смирненски"	Благоевград	98.00	47	51.00	С.КОЧА
1231	Ж	100200	ПМГ "Акад. Сергей Корольов"	Благоевград	98.50	47	51.50	ГР.БЛАГОЕВГРА
1543	Ж	105201	СУ "Неофит Рилски"	Благоевград	95.75	47	48.75	ГР.БАНСК
...	...	...	...	...	...	...	...	
55061	Ж	2811516	ПМГ "Атанас Радев"	Ямбол	95.50	44	51.50	ГР.ЯМБО
55070	Ж	2811516	ПМГ "Атанас Радев"	Ямбол	97.00	44	53.00	ГР.ЯМБО

	sex	school_number	school_name	district	total_points	points_closed_questions	points_open_questions	plac
55072	Ж	2811516	ПМГ "Атанас Радев"	Ямбол	96.00	47	49.00	ГР.ЯМБО
55141	Ж	2811516	ПМГ "Атанас Радев"	Ямбол	99.00	47	52.00	ГР.ЯМБО
55156	Ж	2811516	ПМГ "Атанас Радев"	Ямбол	97.00	47	50.00	ГР.ЯМБО

596 rows × 10 columns

In [33]:

```
plot_total_points(best_exams.total_points, bins_number = 10, color = "r")
```



Нека да погледнем колко от тези деца са от математически гимназии.

In [34]:

```
len(best_exams[best_exams.math_school == 1])
```

Out[34]:

295

2.1. Сравнение на малки и големи населени места

2.1.1. Сравнение на малки и големи населени места с всички училища

In [35]:

```
exam_data_small_places = exam_data["total_points"][exam_data.large_city == 0]
exam_data_large_places = exam_data["total_points"][exam_data.large_city == 1]
```

In [36]:

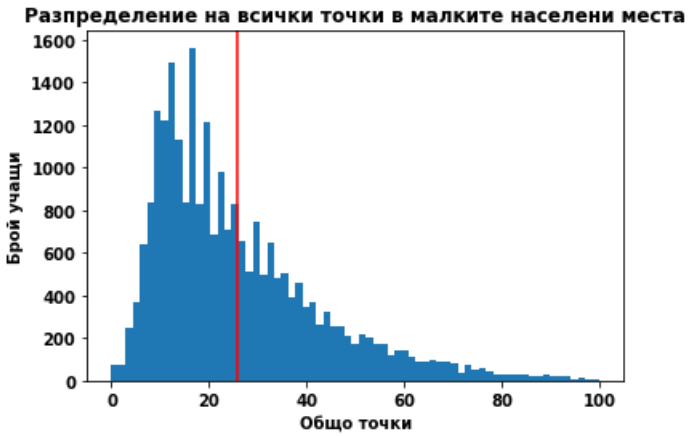
```
comparison_statistics([exam_data_small_places, exam_data_large_places], len(exam_data_small_places
) + len(exam_data_large_places), ["Малки населени места", "Големи населени места"])
```

Имаме 6000 повече ученици от големи градове, участвали в НВО по математика през 2019 г. Очаквано техните резултати са по-добри по всички показатели.

Но нека да погледнем и разпределенията.

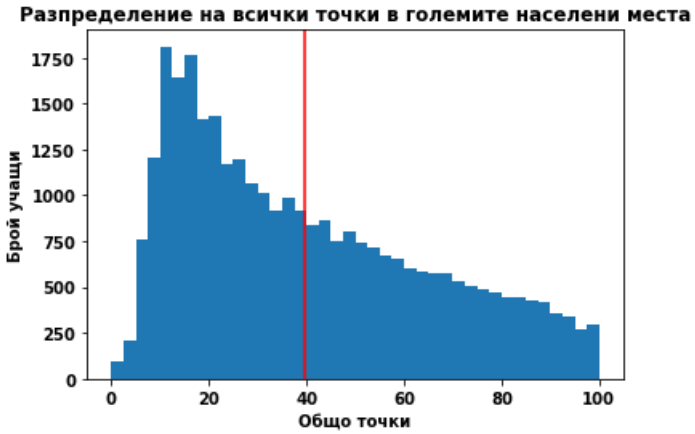
In [37]:

```
plot_total_points(exam_data_small_places, title = "в малките населени места", color = "r")
```



In [38]:

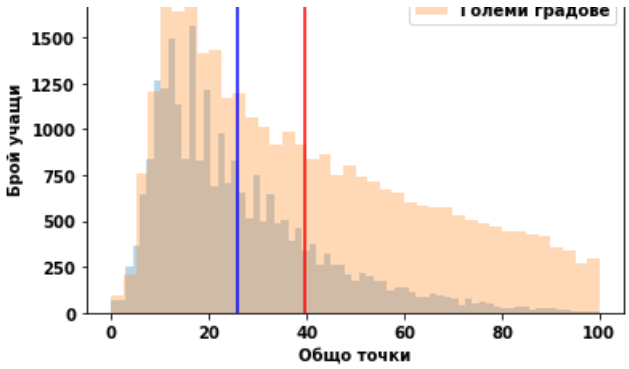
```
plot_total_points(exam_data_large_places, title = "в големите населени места", color = "r")
```



In [39]:

```
plot_total_points(exam_data_small_places, label_name = "Малки градове", show_legend = True,
show_figure = False, alpha_level = 0.3, color = "b")
plot_total_points(exam_data_large_places, label_name = "Големи градове", show_legend = True,
show_figure = False, alpha_level = 0.3,  color = "r")

plt.show()
```



Очаквано разпределнията са различни, но нека да проверим тази хипотеза и с t-test.

2.1.1.1. Подготовка за тестване на хипотеза малки спрямо големи населени места

Дефинираме следните хипотези:

- H_0 : Точките на учениците не зависят от населеното място. **exam_data_small_places** и **exam_data_large_places** са от едно и също разпределение.
- H_1 : **exam_data_small_places** и **exam_data_large_places** са от две различни разпределения.

Тъй като използваме всички данни, избираме доверително ниво $\alpha = 1\%$. Т.е. при $p \leq \alpha$ H_0 следва да бъде орхвърлена.

In [40]:

```
test_result = ttest_ind(exam_data_small_places, exam_data_large_places)
test_result
```

Out[40]:

```
Ttest_indResult(statistic=-73.37566274090423, pvalue=0.0)
```

Явно H_0 може да бъде отхвърлена.

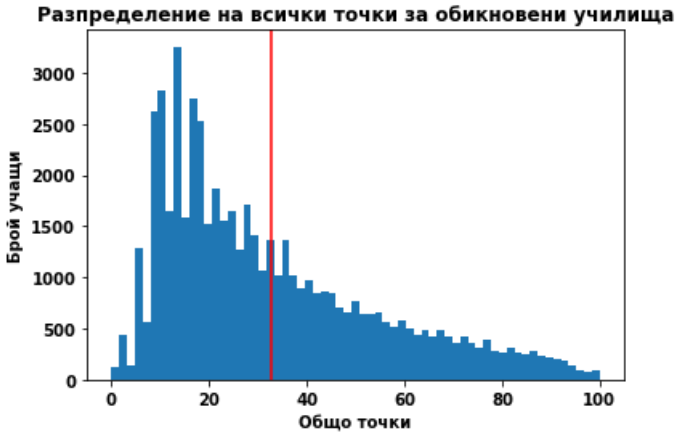
2.1.2. Сравнение на малки и големи населени места, но без математическите гимназии

In [41]:

```
regular_schools_points = exam_data.total_points[exam_data.math_school == 0]
```

In [42]:

```
plot_total_points(regular_schools_points, title = "за обикновени училища", color = "r")
```

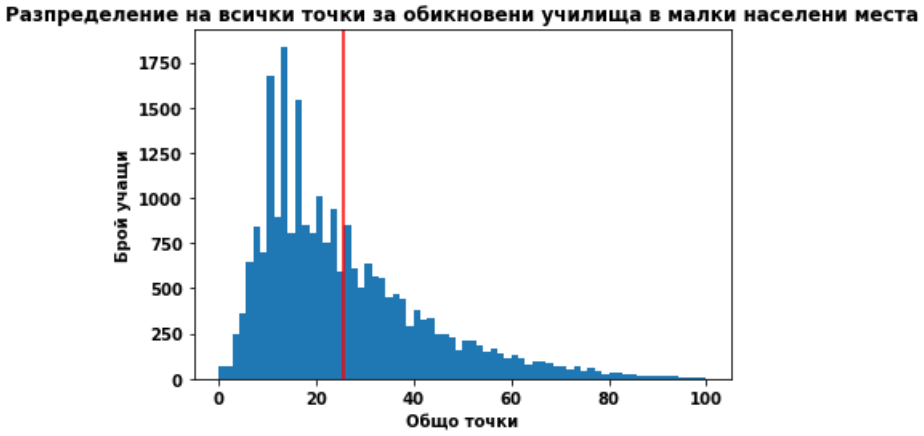


In [43]:

```
regular_schools_small_places = exam_data.total_points[(exam_data.large_city == 0) &
(exam_data.math_school == 0)]
regular_schools_large_places = exam_data.total_points[(exam_data.large_city == 1) &
(exam_data.math_school == 0)]
```

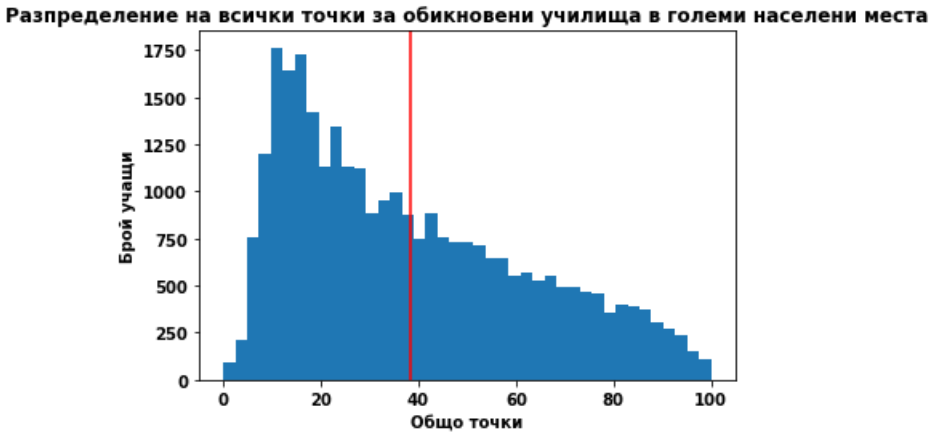
In [53]:

```
plot_total_points(regular_schools_small_places, title = "за обикновени училища в малки населени ме
ста", color = "r")
```



In [54]:

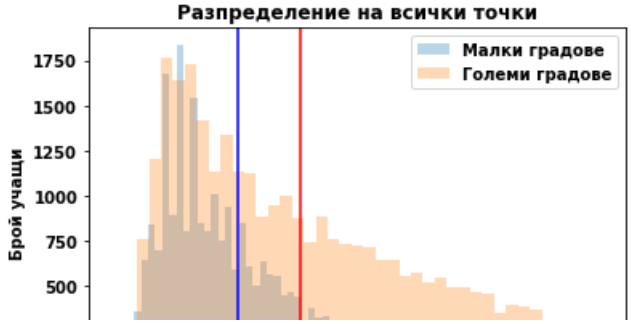
```
plot_total_points(regular_schools_large_places, title = "за обикновени училища в големи населени м
еста",  color = "r")
```

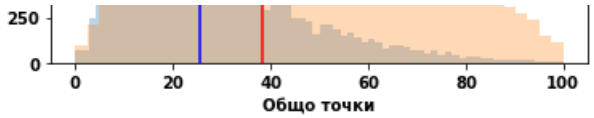


In [46]:

```
plot_total_points(regular_schools_small_places, label_name = "Малки градове", show_legend = True, s
how_figure = False, alpha_level = 0.3, color = "b")
plot_total_points(regular_schools_large_places, label_name = "Големи градове", show_legend = True,
show_figure = False, alpha_level = 0.3,  color = "r")

plt.show()
```





In [47]:

```
comparison_statistics([regular_schools_small_places,
regular_schools_large_places],len(regular_schools_small_places) + len(regular_schools_large_places
), ["Малки населени места", "Големи населени места"])
```

Разликата остава все още голяма.

Да видим, какво ни показва t-test.

2.1.2.1. Perform Hypothesis Testing small vs. Large Places

Дефинираме следните хипотези:

- H_0 : Точките на учениците не зависят от населеното място. **regular_schools_small_places** и **regular_schools_large_places** са от едно и също разпределение.
- H_1 : **regular_schools_small_places** и **regular_schools_large_places** са от две различни разпределения.

Тъй като използваме всички данни, избираме доверително ниво $\alpha = 1\%$. Т.е. при $p \leq \alpha$ H_0 следва да бъде орхвърлена.

In [48]:

```
test_result = ttest_ind(regular_schools_small_places, regular_schools_large_places)
test_result
```

Out[48]:

```
Ttest_indResult(statistic=-68.28046258566107, pvalue=0.0)
```

H_0 може да бъде отхвърлена.

2.2. Math Schools

Нека да погледнем математическите гимназии по-отблизо.

In [49]:

```
math_schools_points = exam_data_total_points[exam_data_math_school == 1]
```

```
math_schools_points = exam_data.total_points[exam_data.math_school == 1]
```

In [50]:

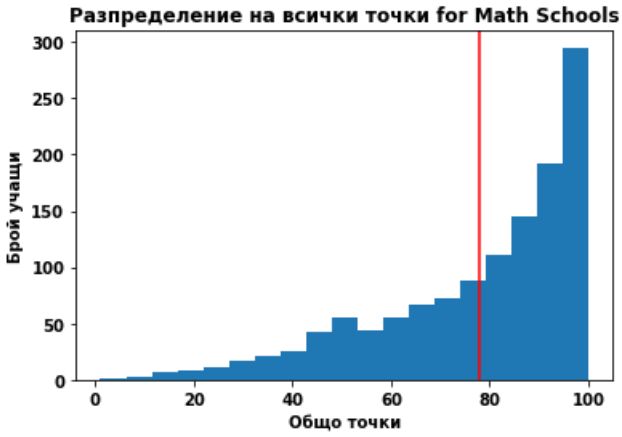
```
len(math_schools_points)
```

Out[50]:

1259

In [51]:

```
plot_total_points(math_schools_points, title = "for Math Schools", color = "r")
```



2.2.1. Math Schools vs. Regular Schools

Едно сравнение на резултатите на математическите гимназии и другите училища.

In [52]:

```
comparison_statistics([regular_schools_points, math_schools_points], exam_data.shape[0],
["Обикновени училища", "Математически гимназии"])
```

In [ ]:

```
```

In [1]:

```
%matplotlib inline
```

In [2]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

import pickle

from scipy.stats import ttest_ind
from py_functions import plot_total_points, comparison_statistics
```

3. Данни 2020

3.1. Подготовка на данннте

3.1.1. Прочитане и почистване на данните

In [3]:

```
exam_data = pd.read_excel("data/MAT_7_2020.xlsx", usecols = "A:J")
```

In [4]:

```
exam_data.shape
```

Out[4]:

```
(56767, 10)
```

In [5]:

```
exam_data["Изпит"].unique()
```

Out[5]:

```
array(['Математика НВО 7'], dtype=object)
```

In [6]:

```
exam_data = exam_data.drop(["Фиктивен номер", "Изпит"], axis = 1)
```

In [7]:

```
exam_data.head()
```

Out[7]:

	Пол	Код на училище	Училище	Област	Населено място	Общо точки	Точки закрити въпроси	Точки открити въпроси
0	Ж	2219049	49 ОУ "Бенито Хуарес"	София-град	София	68.75	44	24.75
1	М	2219049	49 ОУ "Бенито Хуарес"	София-град	София	40.00	33	7.00
2	Ж	2219049	49 ОУ "Бенито Хуарес"	София-град	София	52.25	30	22.25
3	Ж	2219049	49 ОУ "Бенито Хуарес"	София-град	София	58.00	38	20.00

	Пол	Код на училище	Училище	Област	Населено място	Общо точки	Точки закрити въпроси	Точки открити въпроси
4	M	22049	49 ОУ "Бенито Хуарес"	София-град	София	56.50	41	21.50

In [8]:

```
exam_data.columns = ["sex", "school_number", "school_name", "district", "place",
                     "total_points", "points_closed_questions", "points_open_questions"
]
```

In [9]:

```
exam_data.isna().any().any()
```

Out[9]:

```
False
```

In [10]:

```
exam_data.dtypes
```

Out[10]:

```
sex                         object
school_number                int64
school_name                 object
district                    object
place                       object
total_points               float64
points_closed_questions      int64
points_open_questions      float64
dtype: object
```

3.1.2. Подготовка на нова колона за голям град

In [11]:

```
large_cities = []
with open('data/output/large_cities.txt', 'rb') as f:
    large_cities = pickle.load(f)
large_cities
```

Out[11]:

```
['Благоевград',
 'Бургас',
 'Варна',
 'Велико Търново',
 'Горна Оряховица',
 'Видин',
 'Враца',
 'Габрово',
 'Добрич',
 'Кърджали',
 'Кюстендил',
 'Дупница',
 'Ловеч',
 'Монтана',
 'Пазарджик',
 'Перник',
 'Плевен',
 'Пловдив',
 'Асеновград',
 'Разград',
 'Русе',
 'Силистра',
 'Сливен',
 'Смолян',
 'София',
 'Стара Загора',
 'Казанлък',
```

```
 'Търговище',
 'Хасково',
 'Димитровград',
 'Шумен',
 'Ямбол']
```

In [12]:

```
large_city_boolean = exam_data["place"].isin(large_cities)
exam_data["large_city"] = large_city_boolean.astype(int)
```

3.1.3. Подготовка на нова колона за математическа гимназия

In [15]:

```
exam_data.school_name.unique()
```

Out[15]:

```
array(['49 ОУ  "Бенито Хуарес"', '16. ОУ  "Р. Жинзифов"', '102 ОУ', ...,
       'ОУ "Д-р. Петър Берон"', 'БСУ Братислава',
       'БСУ "Д-р П.Берон" Прага'], dtype=object)
```

In [16]:

```
math_schools_boolean = exam_data.school_name.str.contains("МГ", case = True )
exam_data["math_school"] = math_schools_boolean.astype(int)
```

In [17]:

```
exam_data.to_csv(r'data/output/math_exam_2020.csv', index = False)
```

3.2. Анализ на данните

In [18]:

```
exam_data.corr()
```

Out[18]:

	school_number	total_points	points_closed_questions	points_open_questions	large_city	math_school
school_number	1.000000	0.038454	0.037068	0.036059	0.147706	0.018120
total_points	0.038454	1.000000	0.949426	0.951899	0.312465	0.276521
points_closed_questions	0.037068	0.949426	1.000000	0.807548	0.245547	0.224698
points_open_questions	0.036059	0.951899	0.807548	1.000000	0.347318	0.300144
large_city	0.147706	0.312465	0.245547	0.347318	1.000000	0.114826
math_school	0.018120	0.276521	0.224698	0.300144	0.114826	1.000000

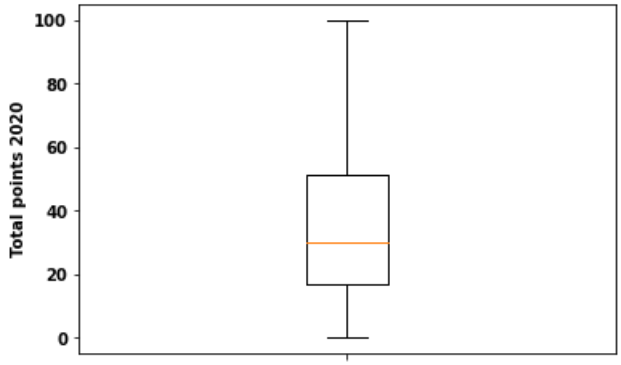
Очаквано виждаме силна позитивна корелация между **total_points** и **points_closed_questions**, между **total_points** и **points_open_questions**, и **points_closed_questions** и **points_open_questions**. Има слаба позитивна корелация между **large_city** и **points_open_questions**, както и между **large_city** и **total_points**. Между **math_school** и **points_open_questions**, **math_school** и **points_closed_questions** и **math_school** и **total_points** корелациите също са слабо позитивни. Т.е. няма нищо неочаквано.

In [19]:

```
plt.boxplot(exam_data.total_points)

plt.xticks([1], [""])
plt.ylabel("Total points 2020")

plt.show()
```



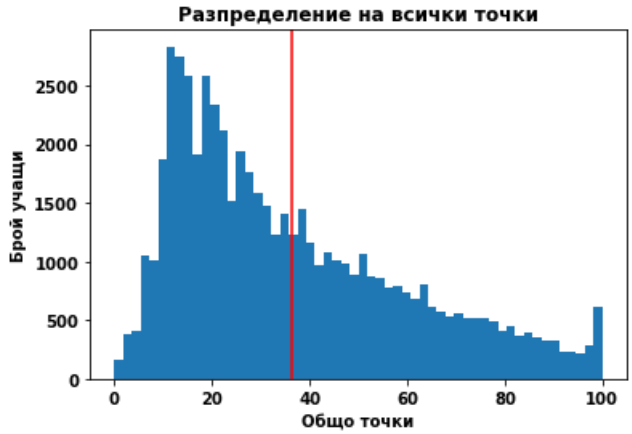
In [21]:

```
print("Общо точки min: {}".format(exam_data.total_points.min()))
print("Общо точки max: {}".format(exam_data.total_points.max()))
print("Общо точки mean: {:.2f}".format(exam_data.total_points.mean()))
print("Общо точки median: {}".format(exam_data.total_points.median()))
```

```
Общо точки min: 0.0
Общо точки max: 100.0
Общо точки mean: 36.33
Общо точки median: 30.0
```

In [22]:

```
plot_total_points(exam_data.total_points, color = "r")
```



In [23]:

```
best_exams = exam_data[exam_data.total_points >= 95]
```

In [24]:

```
best_exams
```

Out[24]:

	sex	school_number	school_name	district	place	total_points	points_closed_questions	points_open_questions	large_city
150	М	2203074	74 СУ "Гоце Делчев"	София-град	София	99.50	50	49.50	1
232	Ж	2202002	2. СУ "Акад.Емилиян Станев"	София-град	София	98.00	50	48.00	1
236	Ж	2202002	2. СУ "Акад.Емилиян Станев"	София-град	София	96.00	50	46.00	1
322	М	2002701	НУККПИИЕК	София-	София	97.00	47	50.00	1

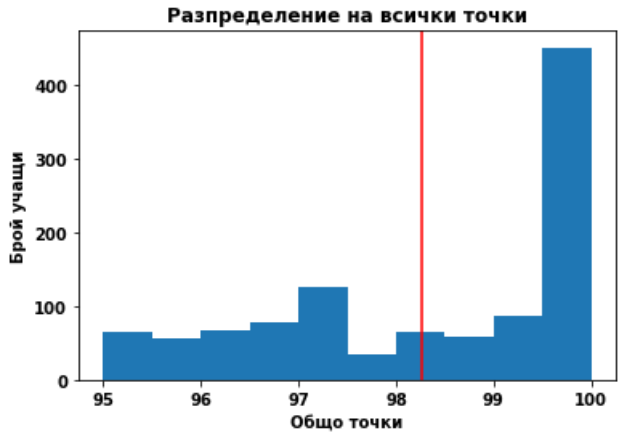
	sex	school_number	school_name	district	place	total_points	points_closed_questions	points_open_questions	large_city
325	M	2902701	НУККЛИИЕК	София-град	София	96.25	47	49.25	1
...	...	...	...	...	...	...	...	...	...
55227	M	800017	СУ "Петко Р. Славейков"	Добрич	Добрич	96.00	50	46.00	1
55542	M	800019	ПМГ "Иван Вазов"	Добрич	Добрич	95.00	50	45.00	1
55609	Ж	800017	СУ "Петко Р. Славейков"	Добрич	Добрич	96.50	47	49.50	1
55632	Ж	800017	СУ "Петко Р. Славейков"	Добрич	Добрич	95.75	50	45.75	1
55937	Ж	800082	СУ "Стефан Караджа"	Добрич	Каварна	99.50	50	49.50	0

1087 rows × 10 columns

През 2020 има около 500 ученика повече с висок брой точки. И това е интересен резултат!

In [25]:

```
plot_total_points(best_exams.total_points, bins_number = 10, color = "r")
```



Нека да погледнем колко от тези деца са от математически гимназии.

In [26]:

```
len(best_exams[best_exams.math_school == 1])
```

Out[26]:

433

Отново около половината от учениците с висок резултат са от математически гимназии.

In [1]:

```
%matplotlib inline
```

In [2]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

import pickle

from scipy.stats import ttest_ind
from py_functions import plot_total_points, comparison_statistics
```

4. Сравнение на НВО 2019 с НВО 2020

4.1. Зареждане на данните

In [3]:

```
exam_data_2020 = pd.read_csv("data/output/math_exam_2020.csv")
```

In [4]:

```
exam_data_2020.head()
```

Out[4]:

	school_number	school_name	district	place	total_points	points_closed_questions	points_open_questions	sex_boolean	large_ci
0	2219049	49 ОУ "Бенито Хуарес"	София-град	София	68.75	44	24.75	1	
1	2219049	49 ОУ "Бенито Хуарес"	София-град	София	40.00	33	7.00	0	
2	2219049	49 ОУ "Бенито Хуарес"	София-град	София	52.25	30	22.25	1	
3	2219049	49 ОУ "Бенито Хуарес"	София-град	София	58.00	38	20.00	1	
4	2219049	49 ОУ "Бенито Хуарес"	София-град	София	62.50	41	21.50	0	

In [5]:

```
exam_data_2019 = pd.read_csv("data/output/math_exam_2019.csv")
```

In [6]:

```
exam_data_2019.head()
```

Out[6]:

	sex	school_number	school_name	district	total_points	points_closed_questions	points_open_questions	place	l
0	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	25.5	25	0.5	ГР.БЛАГОЕВГРАД	
1	Ж	100070	VII СУ "Кузман	Благоевград	17.0	17	0.0	ГР.БЛАГОЕВГРАД	

	sex	school_number	school_name	district	total_points	points_closed_questions	points_open_questions	place	la
2	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	16.0	16	0.0	ГР.БЛАГОЕВГРАД	
3	М	100070	VII СУ "Кузман Шапкарев"	Благоевград	13.5	11	2.5	ГР.БЛАГОЕВГРАД	
4	Ж	100070	VII СУ "Кузман Шапкарев"	Благоевград	21.0	18	3.0	ГР.БЛАГОЕВГРАД	

4.2 Анализ на данните

In [7]:

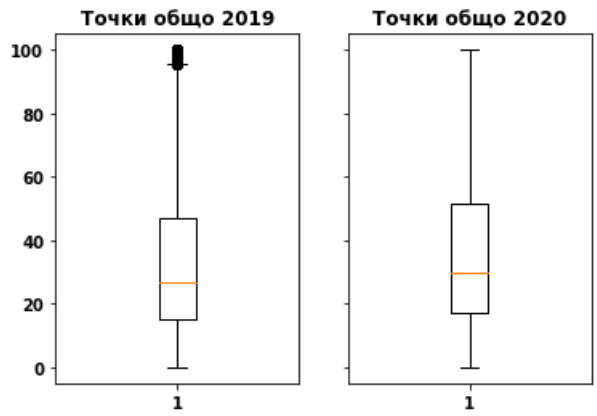
```
comparison_statistics([exam_data_2019.total_points, exam_data_2020.total_points], exam_data_2019.s
hape[0] + exam_data_2020.shape[0] , ["2019", "2020"])
```

In [9]:

```
f, (ax1, ax2) = plt.subplots(1, 2, sharey=True)

ax1.boxplot(exam_data_2019.total_points)
ax1.set_title("Точки общо 2019")

ax2.boxplot(exam_data_2020.total_points)
ax2.set_title("Точки общо 2020")
plt.show()
```

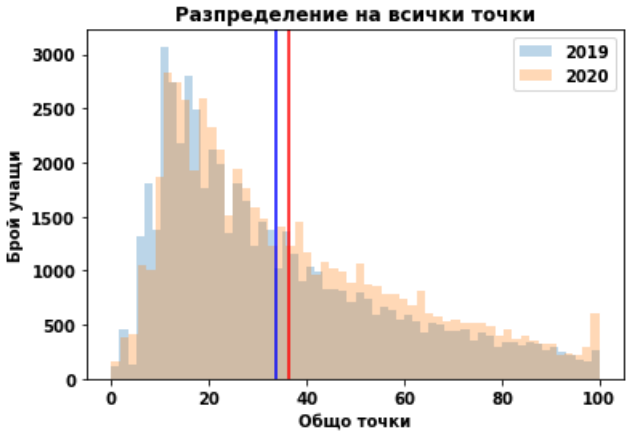


Нетипичните данни през 2019. През 2020 повече ученици са получили максимален брой точки.

In [10]:

```
plot_total_points(exam_data_2019.total_points, label_name = "2019", show_legend = True, show_figure
= False, alpha_level = 0.3, color = "b")
plot_total_points(exam_data_2020.total_points, label_name = "2020", show_legend = True, show_figure
= False, alpha_level = 0.3,  color = "r")

plt.show()
```



Двете разпределния са близки. Поради COVID кризата, аз очаквах по-ниска стойност на медианатапрез 2020. В тази връзка намирам резултатите през 2020 за изненадващи. Интересен е и броят на учениците с максимален брой точки през 2020.

Източници

[1] http://copuo.bg/page.php?c=24&d=19

[2] https://www.nsi.bg/bg/content/2981/%D0%BD%D0%B0%D1%81%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BF%D0%BE-%D0%B3%D1%80%D0%B0%D0%B4%D0%BE%D0%B2%D0%B5-%D0%B8-%D0%BF%D0%BE%D0%BB

Изводи

Благодаря за вниманието!