Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Системы обработки информации и управления»

“Методы машинного обучения”

ЛАБОРАТОРНАЯ РАБОТА № 1. **«Разведочный анализ данных. Исследование и визуализация данных»**

Студент группы ИУ5-24М

Петропавлов Д.М.

_____ Дата

_____ Подпись

Москва 2020

Цель лабораторной работы: изучение различных методов визуализации данных

Для выполнения данной работы был выбран датасет 1000 * 8,
содержащий статистику по баллам за три экзамена
(math score - математика, writing score - письмо, reading score - чтение)
в зависимости от 5 параметров:
gender (пол),
race/ethnicity - этническая принадлежность (в силу моральных принципов автора сета были поделены на Group AE),
parental level of education (уровень образования),
lunch (стабильное питание),
test preparation course (подготовительные курсы).
Датасет представлен одним файлом Dataset/StudentsPerformance.csv

1. Импорт библиотек и загрузка данных

Подключим необходимые библиотеки, такие как Pandas, Numpy и Seabo команды import.

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

Используя read_csv() загрузим данные и проверим, выведя первые head().

```
data = pd.read_csv('C:/Dataset/StudentsPerformance.csv')
data.head()
```

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

2. Основные характеристики датасета

Основные параметры выборки можно получить командой describe().
Таблице только 3 параметра вместо 8 будут указаны ниже.

```
data.describe()
```

	math score	reading score	writing score
count	1000.00000	1000.000000	1000.000000
mean	66.08900	69.169000	68.054000
std	15.16308	14.600192	15.195657
min	0.00000	17.000000	10.000000
25%	57.00000	59.000000	57.750000
50%	66.00000	70.000000	69.000000
75%	77.00000	79.000000	79.000000
max	100.00000	100.000000	100.000000

Для некоторых методов анализа наличие "пустых" ячеек критично. проверить датасет на наличие параметров Null и либо удалить все самостоятельно заполнить их. В нашем случае таких ячеек нет.

In [6]:
```
for col in data.columns:
    # Количество пустых значений - все значения заполнены
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

gender - 0

race/ethnicity - 0

parental level of education - 0

lunch - 0

test preparation course - 0

math score - 0

reading score - 0

writing score - 0

Анализу можно подвергать не все параметры датасета сразу, а приведен пример анализа множества значений параметра gend использований каждого.

In [7]:
```
data['gender'].value_counts()
```

Out[7]:
```
female    518
male      482
Name: gender, dtype: int64
```

3. Визуальный анализ датасета

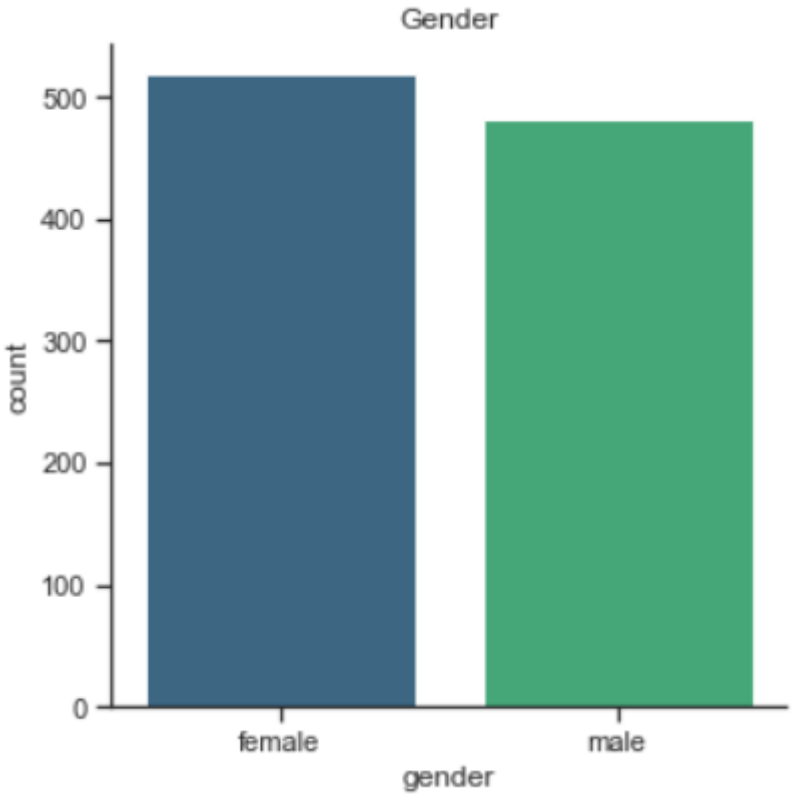
1.Catplot Самый простой и наглядный способ проанализировать выборку - построить на ее области диаграмму.
За это в библиотеке Seaborn отвечает функция catplot.
На том же примере с полом студенов:

In [8]:

```
sns.catplot(x='gender',kind='count',data=data,height=4.5,palette='viridis')
plt.title('Gender')
```

Out[8]:

```
Text(0.5, 1.0, 'Gender')
```

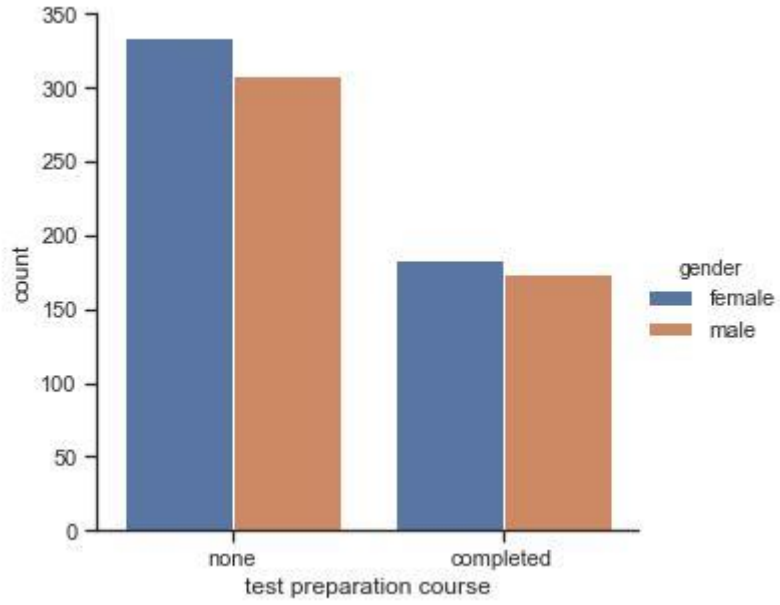


Данная функция обладает множеством параметров для кастомизации гистограмм. Например, введем еще один сортирующий параметр "test preparation course", и получим график распределения студентов по полу относительно факта прохождения подготовительных курсов:

In [7]:

```
sns.catplot(x='test preparation course',
kind='count',data=data,height=4.5,hue='gender')
```

Out[7]:

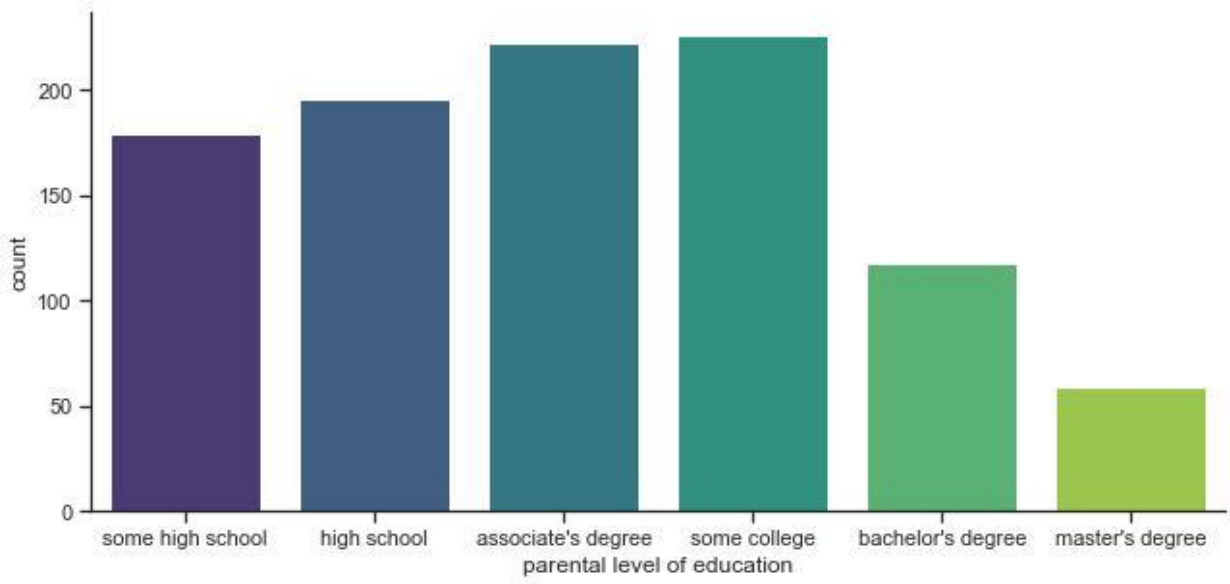


Так же можно отсортировать порядок вывода, толщину столбцов и цветовую палитру.

In [8]:

```
#data["race/ethnicity"].sort_values()
sns.catplot(x='parental level of
education',kind='count',data=data,height=4.5,aspect=2,palette='viridis',
                    order=["some high school","high school","associate's
degree","some college",
                                "bachelor's degree","master's degree"],)
```

Out[8]:



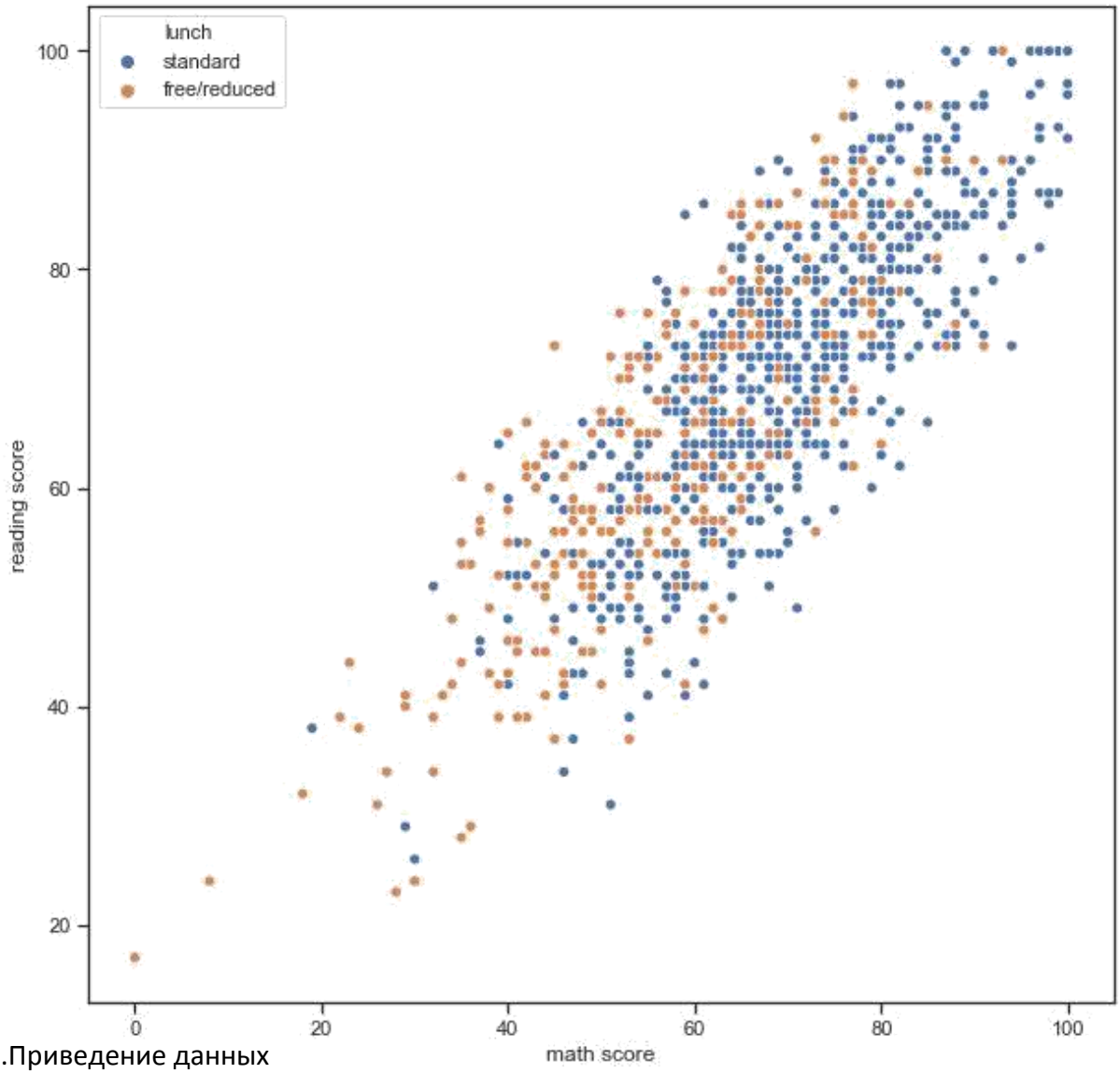
2.Subplots

Диаграмма рассеяния - точечная диаграмма показывающая зависимость значений.
На данном примере по оси X отложены значения тестов по математике, по У - по чтению.
Цветом выделены точки-записи относительно параметра lunch.

In [10]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='math score', y='reading score', data=data, hue='lunch')
```

Out[10]:



3.Приведение данных

Одна из главных проблем анализа данных - их формат.

В данной выборке все параметры кроме целевых, заданы текстовыми значениями.

Для дальнейшего анализа необходимо перевести их в "цифру".

Как это было сделано - показано ниже:

In [12]:

```
data['gender'].replace({'male':'1','female':'0'},inplace=True)
data['race/ethnicity'].replace({'group A':'1','group B':'2', 'group C':'3', 'group D':'4','group E':'5'},inplace=Tru
e)
data['lunch'].replace({'free/reduced':'0','standard':'1'},inplace=True)
data['test preparation course'].replace({'none':'0','completed':'1'},inplace=True)
data['parental level of education'].replace({'some high school':'1','high school':'1',"associate's degree":'2','s
ome college':'3',"bachelor's degree":'4',"master's degree":'5'},inplace=True)

data['gender'] = data['gender'].astype('int64')
data['race/ethnicity'] = data['race/ethnicity'].astype('int64')
data['lunch'] = data['lunch'].astype('int64')
data['test preparation course'] = data['test preparation course'].astype('int64')
```

```
data['parental level of education'] = data['parental level of education'].astype('int64')
```

In [12]:

```
#посмотрим результат изменения значений
data.head()
```

Out[12]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	0	2	4	1	0	72	72	74
1	0	3	3	1	1	69	90	88
2	0	2	5	1	0	90	95	93
3	1	1	2	0	0	47	57	44
4	1	3	3	1	0	76	78	75

In [13]:

```
#убедимся что изменены не только значения, но и тип данных
data.dtypes
```

Out[13]:

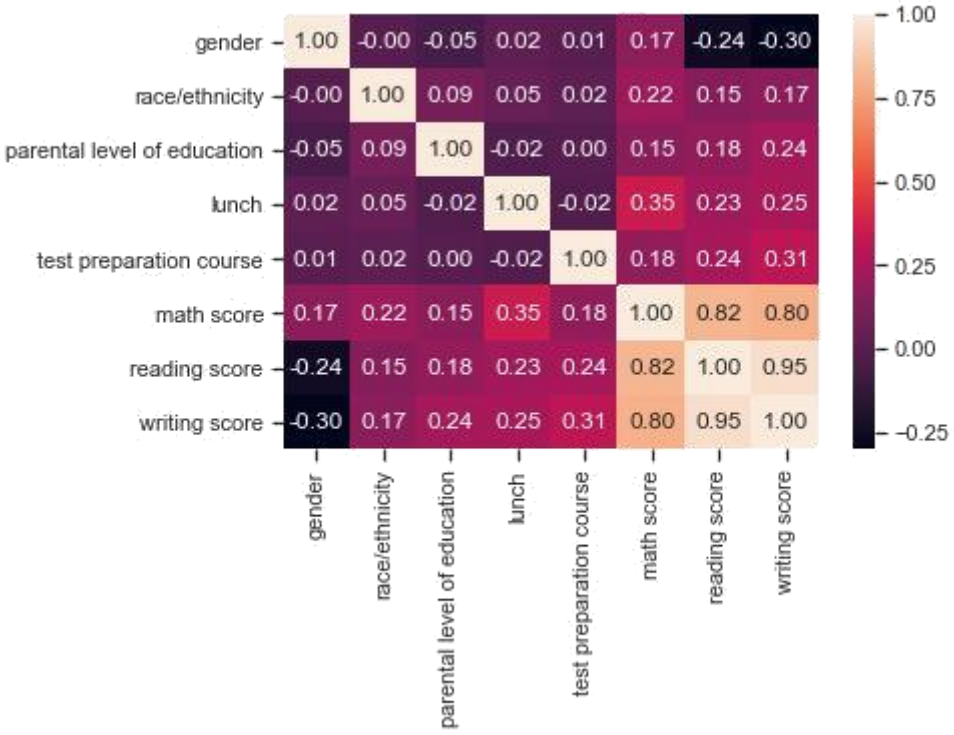
```
gender                         int64
race/ethnicity                 int64
parental level of education    int64
lunch                          int64
test preparation course        int64
math score                     int64
reading score                  int64
writing score                  int64
dtype: object
```

1. Corr Корреляционная - матрицаглавный инструмент при анализе взаимосвяз большой выборке. Значения близкие по модулю к 1 означают высок корреляции, т.е. высокую степень зависимости. Функция heatmap построениятрицыма выделить ячейки-цветоваяцветом карта еще нагляднее показывает степень близости, облегчая визуальный анализ выборо объемом параметров.

2. In [14]:

```
sns.heatmap(data.corr(), annot=True, fmt='.2f')
```

Out[14]:

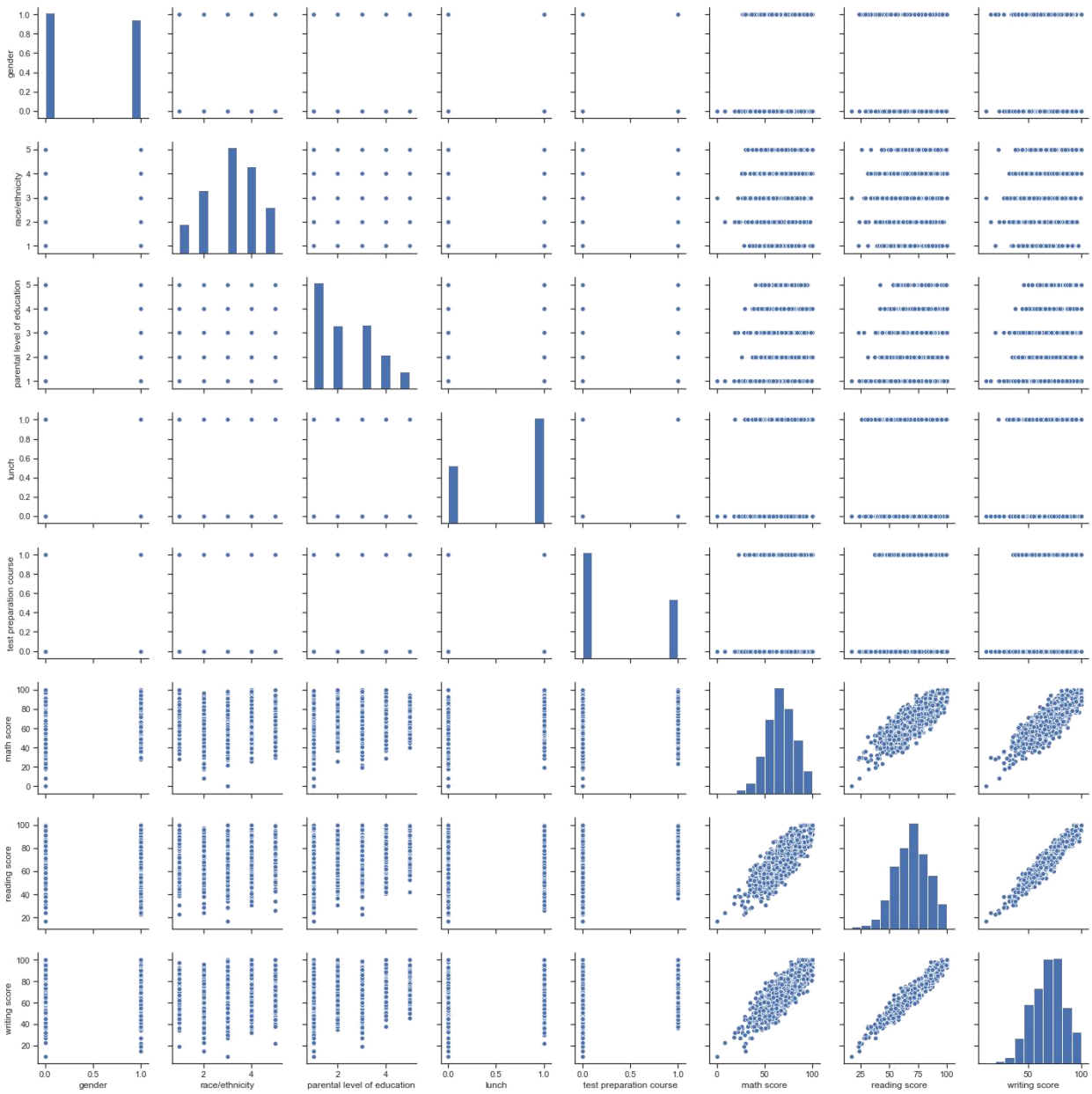


4.Pairplot Данная команда строит матрицу N *-признакNдиагра-ыс**м**м"N.формата "

In [15]:

```
sns.pairplot(data)
```

Out[15]:

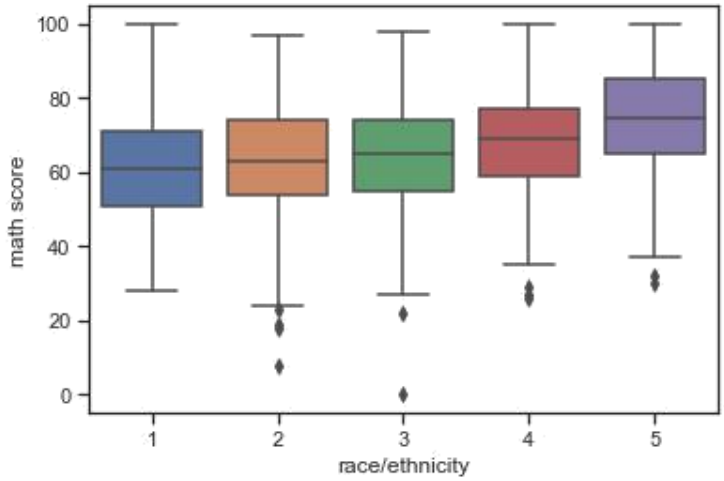


5.Boxplot "Ящик с- своебразныйусами" тип диаграмм отображающий среднее разброс параметра.

In [16]:

```
sns.boxplot(x=data['race/ethnicity'],y=data['math score'])
```

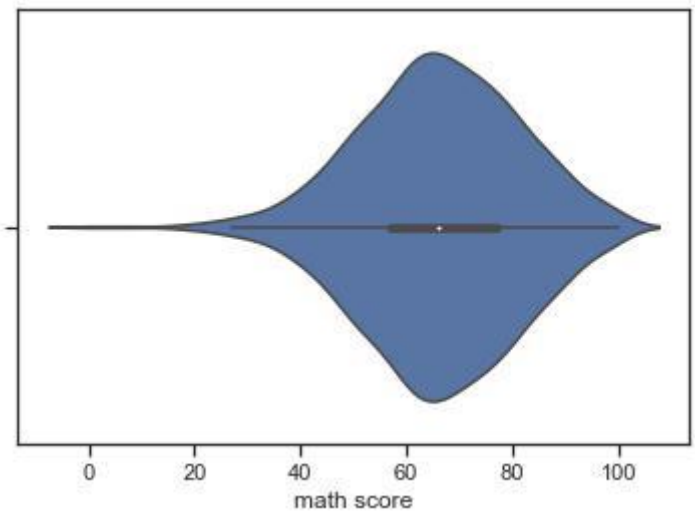
Out[16]:



6.Violinplot Еще одна диаграмма отображающая разброс значений пар
In [17]:

```
sns.violinplot(x=data['math score'])
```

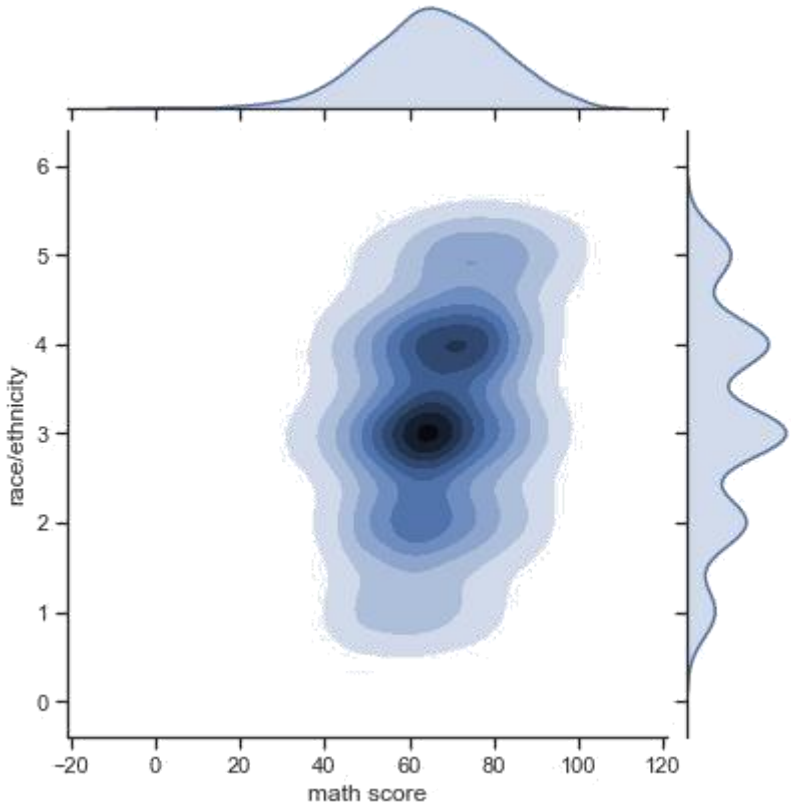
Out[17]:



7.Jointplot Смесь из нескольких диаграммтключа.В зависимостикindзависито тип отр На данном примере нарисована карта глубины.
In [18]:

```
sns.jointplot(x='math score', y='race/ethnicity', data=data, kind="kde")
```

Out[18]:

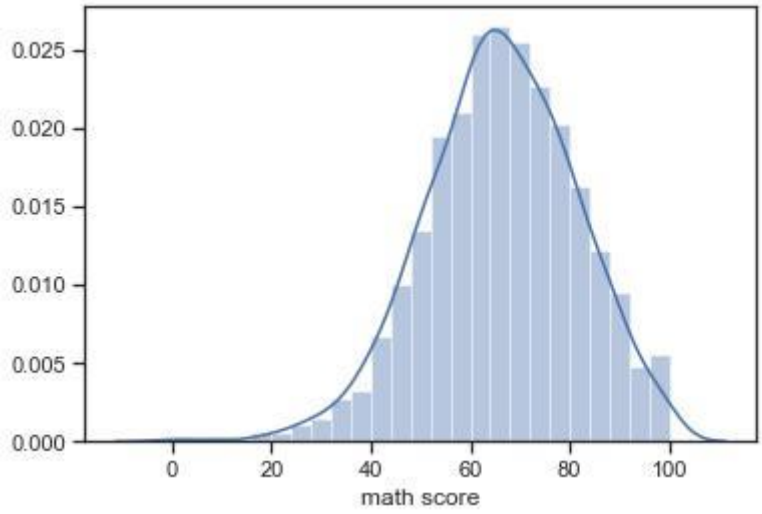


8.Distplot Еще одна диаграммащая отображаюраспределение параметра.

In [19]:

```
sns.distplot(data['math score'])
```

Out[19]:



4. Результаты анализа

Результаты анализа данного датасета оказались неоднозначны.

Во-первых, не подтвердился миф о том, что юношам лучше даются математические науки, а девушкам - гуманитарные.

Во-вторых, от степени подготовки и прохождения дополнительных курсов прямой зависимости так же не наблюдается.

Однако, студенты, успешно сдавшие один из экзаменов, также хорошо сдают и остальные.