ML 14 - EDA Student Performance Indicator By Virat Tiwari

November 26, 2023

1 EDA Student Performance Indicator By Virat Tiwari

1) Problem Statement

This project understand how the student's performnace (test score) is affected by other variables such as Gender , Ethnicity , Parental Level Of Education , Lunch and Test Preparation course .

2) Data Collection

This Data consist of 8 columns and 1000 rows

Data source - https://www.kaggle.com/datasets/spscientist/students-performance-in-exams

3) Dataset Information

a) Gender : sex of student -> (M / f) ,

b) Race : Ethnicitu of student -> (Group - A,B,C,D,E),

c) Parental Level of education : Parents'final education -> (Bachelors , masters high school etch degrees),

d) lunch : Habving lunch before test (standard / free),

e) Test preparation course : complete or not complete before test ,

f) Math score ,

g) Reading score ,

h) Writing score

[1]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
import warnings
warnings.filterwarnings("ignore")
```

[2]:
```
df=pd.read_csv("Student Performance Indicator.csv")
df.head()
```

[2]:
```
   gender race_ethnicity parental_level_of_education         lunch  \
0  female        group B           bachelor's degree      standard
```

```
1  female        group C               some college      standard
2  female        group B            master's degree      standard
3    male        group A         associate's degree  free/reduced
4    male        group C               some college      standard

  test_preparation_course  math_score  reading_score  writing_score
0                    none          72             72             74
1               completed          69             90             88
2                    none          90             95             93
3                    none          47             57             44
4                    none          76             78             75
```

[3]:
```
df.shape
```

[3]:
```
(1000, 8)
```

[4]:
```
df.columns
```

[4]:
```
Index(['gender', 'race_ethnicity', 'parental_level_of_education', 'lunch',
       'test_preparation_course', 'math_score', 'reading_score',
       'writing_score'],
      dtype='object')
```

2 Data Checks To Perform

We check all these parameters before using the dataset or performing the dataset for analysis

1) Check missing values

2) Check duplicates

3) Check data types

4) Check the number of unique values of each columns

5) Check thh statistics of Dataset

6) Check the various categories presemt in the different categorical columns

3 A) Check Missing Values

[5]:
```
# CHECK THE MISSING VALUES

df.isnull().sum()
```

[5]:
```
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
dtype: int64
```

Q) What Insights Or Observation we get ?

Ans) There is no Missing Values in the Dataset

[6]:
```
# ALTERNATE WAY TO CHECK THE MISSING VALUES

df.isna().sum()
```

[6]:
```
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
dtype: int64
```

Q) What Insights Or Observation we get ?

Ans) There is no Missing Values in the Dataset

4 B) Check Duplicates

[7]:
```
df.duplicated().sum()
```

[7]:
```
0
```

Q) Insights

Ans) There is no duplicate values

[8]:
```
df.duplicated()
```

[8]:
```
0      False
1      False
2      False
3      False
4      False
       …
995    False
996    False
```

```
997    False
998    False
999    False
Length: 1000, dtype: bool
```

Q) Insights

Ans) There is no duplicate values

5 C) Check Data Types

[9]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 8 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   gender                       1000 non-null   object
 1   race_ethnicity               1000 non-null   object
 2   parental_level_of_education  1000 non-null   object
 3   lunch                        1000 non-null   object
 4   test_preparation_course      1000 non-null   object
 5   math_score                   1000 non-null   int64
 6   reading_score                1000 non-null   int64
 7   writing_score                1000 non-null   int64
dtypes: int64(3), object(5)
memory usage: 62.6+ KB
```

6 D) Check the number of unique values of each columns

[10]:
```
df.nunique()
```

[10]:
```
gender                         2
race_ethnicity                 5
parental_level_of_education    6
lunch                          2
test_preparation_course        2
math_score                    81
reading_score                 72
writing_score                 77
dtype: int64
```

7 E) Check thh statistics of Dataset

[11]:
```
df.describe()
```

[11]:
```
       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
```

Q) Insights

Ans) In the data MEANS are very close to each other that is in the range between 66 to 69 AND all the STANDARD DEVIATION is also very close that is in the range between 14.6 to 15.19 AND some STUDENTS get 0 marks in MATHS as well as 100 marks in MATHS also

[12]:
```
# some more data information

df.head()
```

[12]:
```
   gender race_ethnicity parental_level_of_education         lunch  \
0  female        group B           bachelor's degree      standard
1  female        group C                some college      standard
2  female        group B            master's degree      standard
3    male        group A          associate's degree  free/reduced
4    male        group C                some college      standard

  test_preparation_course  math_score  reading_score  writing_score
0                    none          72             72             74
1               completed          69             90             88
2                    none          90             95             93
3                    none          47             57             44
4                    none          76             78             75
```

[13]:
```
# some more data information

df.tail()
```

[13]:
```
     gender race_ethnicity parental_level_of_education         lunch  \
995  female        group E            master's degree      standard
996    male        group C                high school  free/reduced
997  female        group C                high school  free/reduced
998  female        group D               some college      standard
999  female        group D               some college  free/reduced
```

```
    test_preparation_course  math_score  reading_score  writing_score
995               completed          88             99             95
996                    none          62             55             55
997               completed          59             71             65
998               completed          68             78             77
999                    none          77             86             86
```

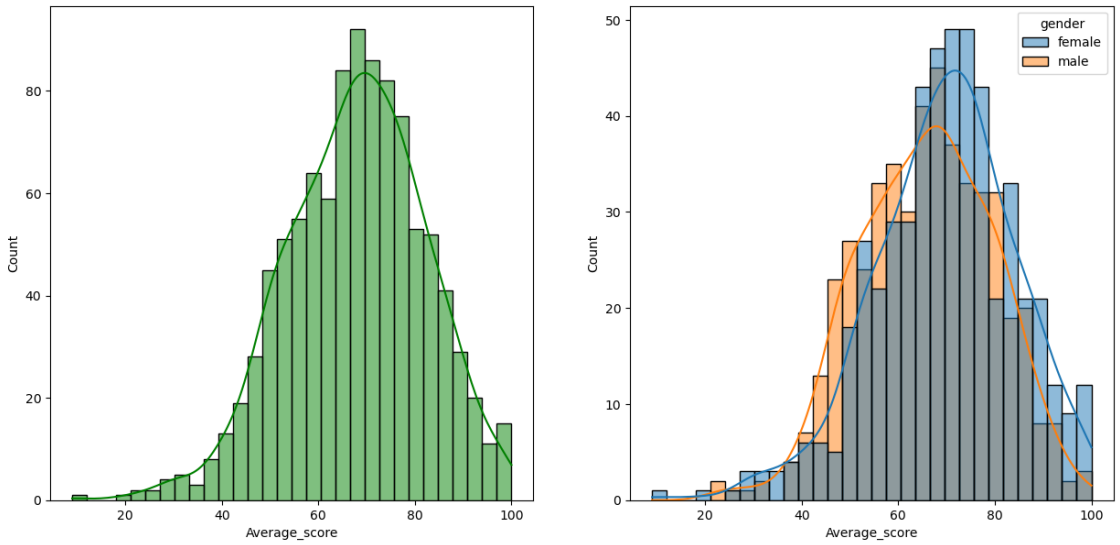
[14]:
```
df["total_score"]=(df["math_score"]+df["reading_score"]+df["writing_score"])
df["Average_score"]=df["total_score"]/3
df.head()
```

[14]:
```
   gender race_ethnicity parental_level_of_education         lunch  \
0  female        group B           bachelor's degree      standard
1  female        group C                some college      standard
2  female        group B             master's degree      standard
3    male        group A          associate's degree  free/reduced
4    male        group C                some college      standard

  test_preparation_course  math_score  reading_score  writing_score  \
0                    none          72             72             74
1               completed          69             90             88
2                    none          90             95             93
3                    none          47             57             44
4                    none          76             78             75

   total_score  Average_score
0          218      72.666667
1          247      82.333333
2          278      92.666667
3          148      49.333333
4          229      76.333333
```

8 EXPLORING MORE VISUALIZATION -

[15]:
```
fig,axis=plt.subplots(1,2,figsize=(15,7))
plt.subplot(121)
sns.histplot(data=df,x="Average_score",bins=30,kde=True,color="g")
plt.subplot(122)
sns.histplot(data=df,x="Average_score",bins=30,kde=True,hue="gender")
```

[15]:
```
<AxesSubplot: xlabel='Average_score', ylabel='Count'>
```

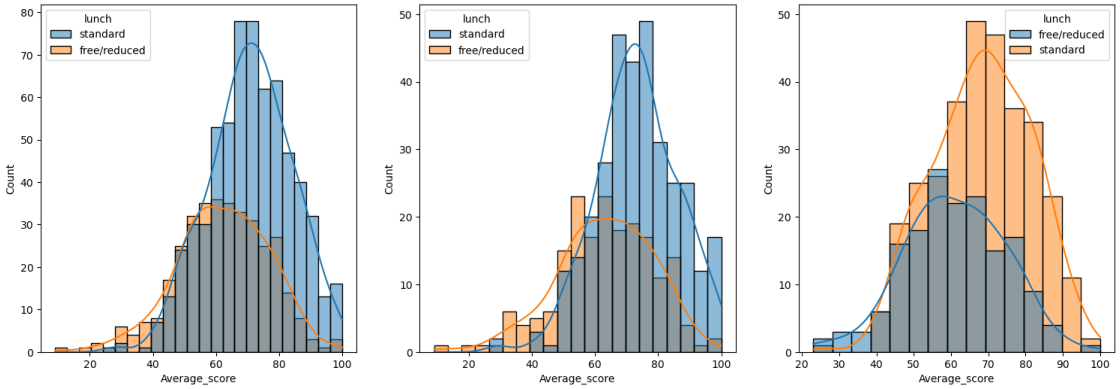


Q) Insights

Ans) Female students perform well comparatively male students

[22]:
```
plt.subplots(1,3,figsize=(25,6))
plt.subplot(141)
sns.histplot(data=df,x="Average_score",kde=True,hue="lunch")
plt.subplot(142)
sns.histplot(data=df[df.
  ↪gender=="female"],x="Average_score",kde=True,hue="lunch")
plt.subplot(143)
sns.histplot(data=df[df.gender=="male"],x="Average_score",kde=True,hue="lunch")
```

[22]:
```
<AxesSubplot: xlabel='Average_score', ylabel='Count'>
```



Q) Insights

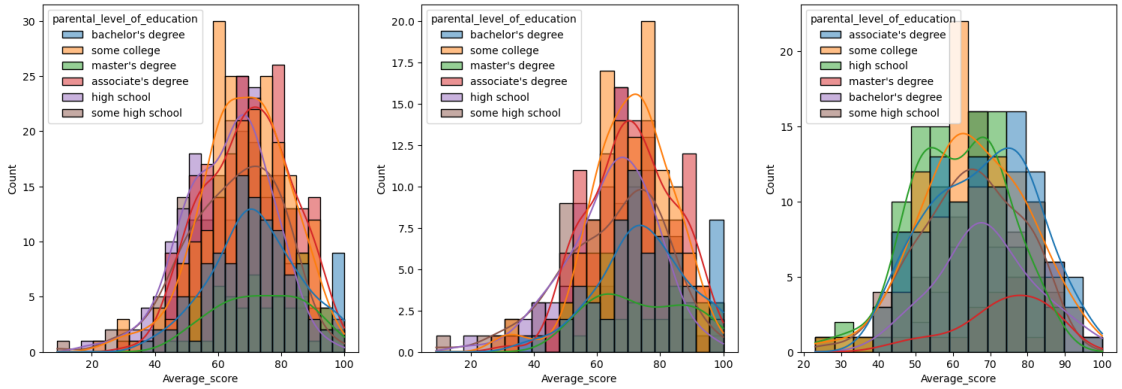
Ans) Standard lunch help students perform well in exams and Standard lunch helps perform well in exams be it a male of female

[23]:
```
df.head()
```

[23]:
```
   gender race_ethnicity parental_level_of_education         lunch  \
0  female        group B           bachelor's degree      standard
1  female        group C                some college      standard
2  female        group B            master's degree      standard
3    male        group A          associate's degree  free/reduced
4    male        group C                some college      standard

  test_preparation_course  math_score  reading_score  writing_score  \
0                    none          72             72             74
1               completed          69             90             88
2                    none          90             95             93
3                    none          47             57             44
4                    none          76             78             75

   total_score  Average_score
0          218      72.666667
1          247      82.333333
2          278      92.666667
3          148      49.333333
4          229      76.333333
```

[24]:
```
plt.subplots(1,3,figsize=(25,6))
plt.subplot(141)
sns.
 ↪histplot(data=df,x="Average_score",kde=True,hue="parental_level_of_education")
plt.subplot(142)
sns.histplot(data=df[df.
 ↪gender=="female"],x="Average_score",kde=True,hue="parental_level_of_education")
plt.subplot(143)
sns.histplot(data=df[df.
 ↪gender=="male"],x="Average_score",kde=True,hue="parental_level_of_education")
```

[24]:
```
<AxesSubplot: xlabel='Average_score', ylabel='Count'>
```

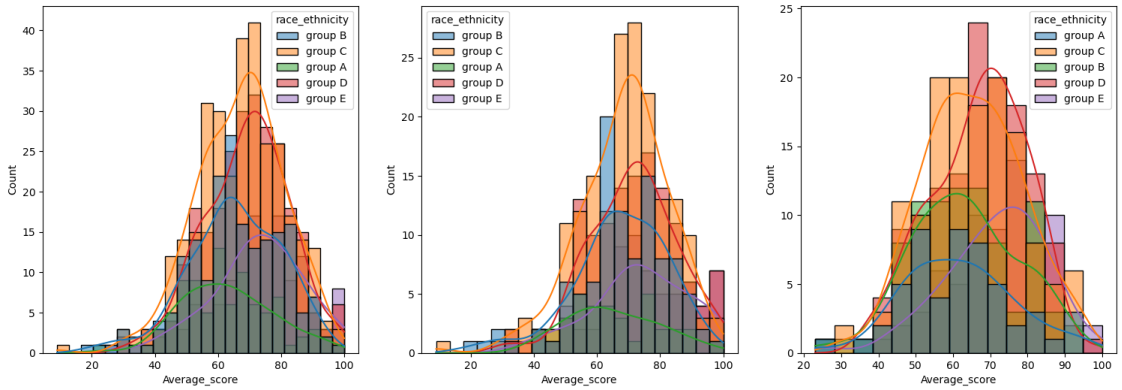


Q) Insights

Ans) "parental_level_of_education" don't help student perform well in exam

[25]:
```
plt.subplots(1,3,figsize=(25,6))
plt.subplot(141)
sns.histplot(data=df,x="Average_score",kde=True,hue="race_ethnicity")
plt.subplot(142)
sns.histplot(data=df[df.
  ↪gender=="female"],x="Average_score",kde=True,hue="race_ethnicity")
plt.subplot(143)
sns.histplot(data=df[df.
  ↪gender=="male"],x="Average_score",kde=True,hue="race_ethnicity")
```

[25]:
```
<AxesSubplot: xlabel='Average_score', ylabel='Count'>
```



Q) Insights

Ans) Student of group A and B tends to perform poorly in exams

THANK YOU SO MUCH !!

YOURS VIRAT TIWARI :)