EDA Student Performance Indicator

Data Collection

- Dataset Source - https://www.kaggle.com/datasets/spscientist/students-performance-in-exams?datasetId=74977
- The data consists of 8 column and 1000 rows.

Dataset Information

- gender : sex of students -> (Male/female)
- race/ethnicity : ethnicity of students -> (Group A, B,C, D,E)
- parental level of education : parents'
- education -> (bachelor's degree,some college,master's degree,associate's degree,high school)
- lunch : having lunch before test (standard or free/reduced)
- test preparation course : complete or not complete before test
- math score
- reading score
- writing score

Problem statement

This project understands how the student's performance (test scores) is affected by other variables such as Gender, Ethnicity, Parental level of education, Lunch and Test preparation course.

In [40]:
```
# Import all Library

import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [41]:
```
# Read the Dataset

df = pd.read_csv("Student.csv")
```

In [42]:
```
df.head()
```

Out[42]:

	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

In [43]:
```
df.shape
```

Out[43]:
```
(1000, 8)
```

In [44]:
```
df.ndim
```

Out[44]:
```
2
```

In [45]:
```
df.size
```

Out[45]:
```
8000
```

In [46]:
```
df.dtypes
```

Out[46]:
```
gender                       object
race_ethnicity               object
parental_level_of_education  object
lunch                        object
test_preparation_course      object
math_score                    int64
reading_score                 int64
writing_score                 int64
dtype: object
```

Data checks to perform

- check missing values
- check duplicates
- check the datatype
- check the number of unique values of each column
- check statistics of dataset
- check various categories present in the different categorical column

In [47]:
```
## check missing Values
df.isnull().sum()
```

Out[47]:
```
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
```

In [48]:
```
df.isna().sum()
```

Out[48]:
```
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
```

In [49]:
```
## Check Duplicates
df.duplicated().sum()
```

Out[49]:
```
0
```

In [50]:
```
## check datatypes
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
```

In [51]:
```
##Checking the number of uniques values of each columns
df.nunique()
```

Out[51]:
```
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
```

In [52]:
```
## Check the statistics of the dataset
df.describe()
```

Out[52]:

	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

Insights or Observation

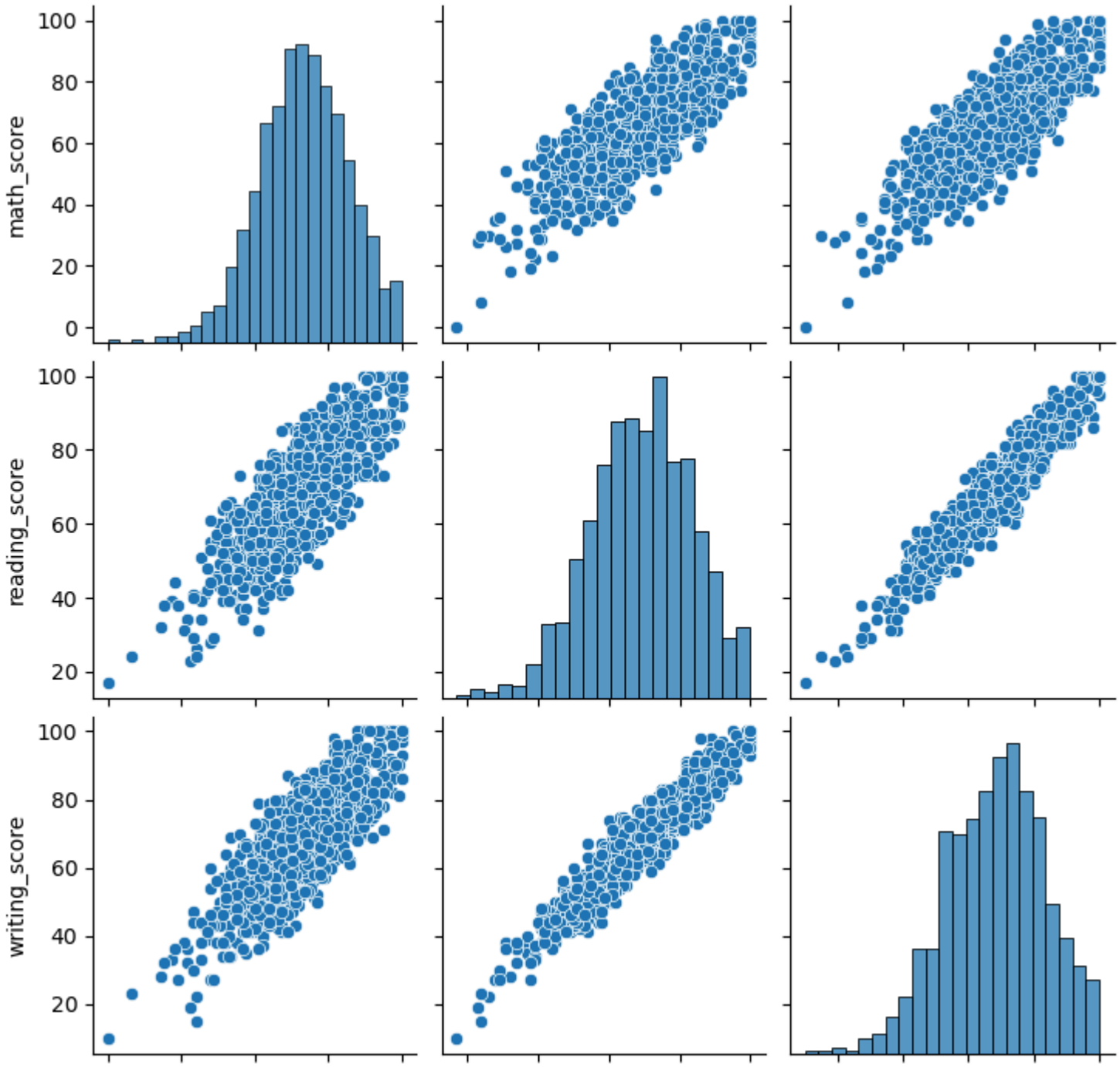
- From the above description of numerical data,all means are very close to each other- between 66 and 69
- All the standard deviation are also close- between 14.6- 15.19
- While there is a minimum of 0 for maths,other are having 17 and 10 value

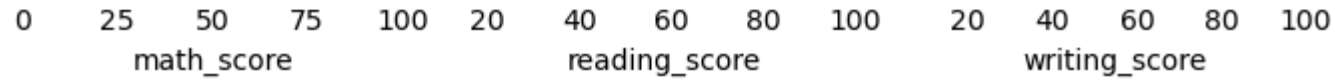
In [53]:
```
## Explore more info about the data
df.head()
```

Out[53]:

	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

In [54]:
```
sns.pairplot(df)
plt.show()
```


math_score
reading_score
writing_score
0
20
40
60
80
100



In [55]:
```
[feature for feature in df.columns if df[feature].dtype=='O']
```

Out[55]:
```
['gender',
 'race_ethnicity',
 'parental_level_of_education',
 'lunch',
 'test_preparation_course']
```

In [56]:
```
#segrregate numerical and categorical features

numerical_features=[feature for feature in df.columns if df[feature].dtype!='O']
categorical_feature=[feature for feature in df.columns if df[feature].dtype=='O']
```

In [57]:
```
numerical_features
```

Out[57]:
```
['math_score', 'reading_score', 'writing_score']
```

In [58]:
```
categorical_feature
```

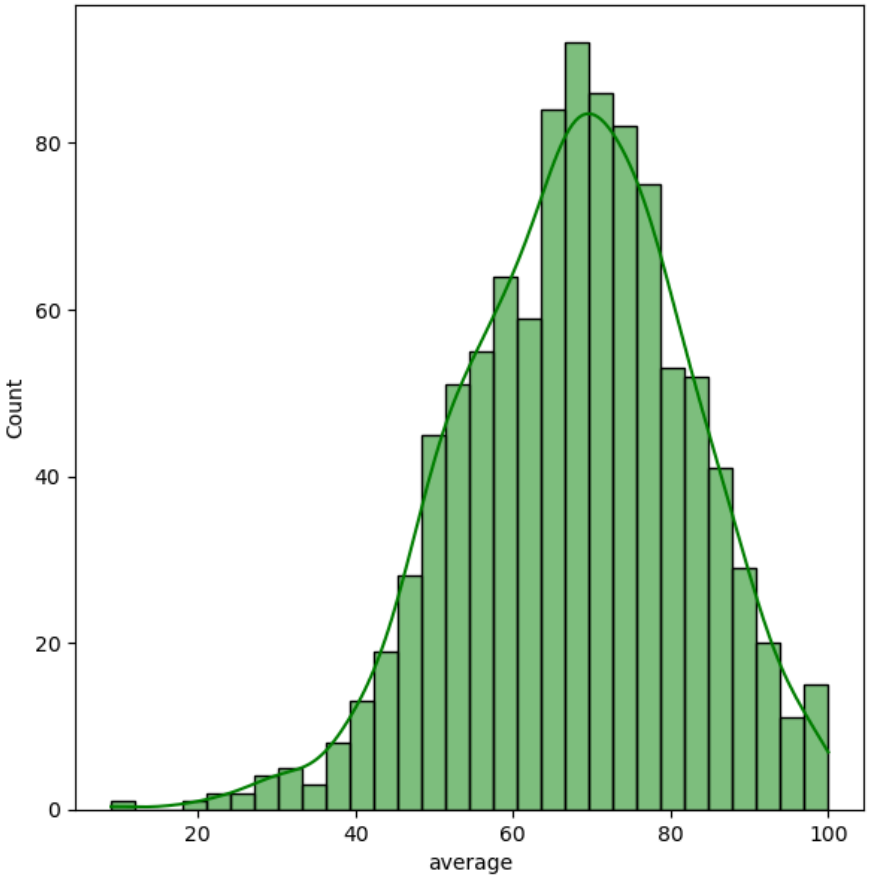
Out[58]:
```
['gender',
 'race_ethnicity',
 'parental_level_of_education',
 'lunch',
 'test_preparation_course']
```

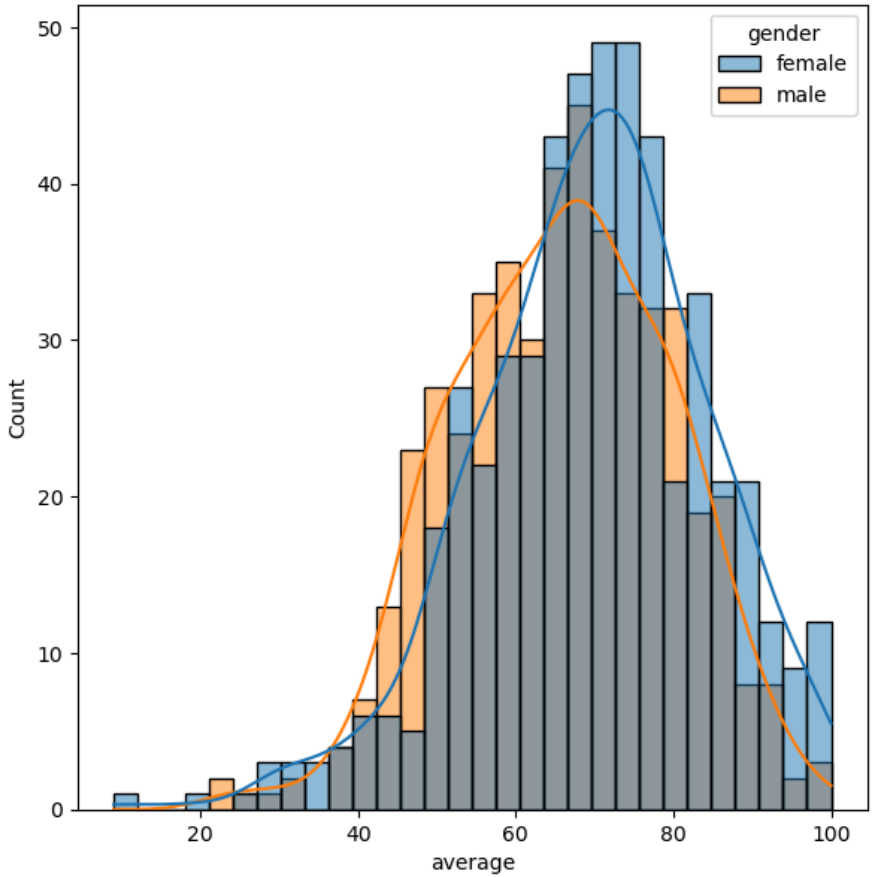
In [59]:
```
## Aggregate the total score with mean

df['total_score']=(df['math_score']+df['reading_score']+df['writing_score'])

df['average']=df['total_score']/3
df.head()
```

Out[59]:

	gender	race_ethnicity	parental_level_of_education	lunch	test_preparation_course	math_score	reading_score	writing_score	tota
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	

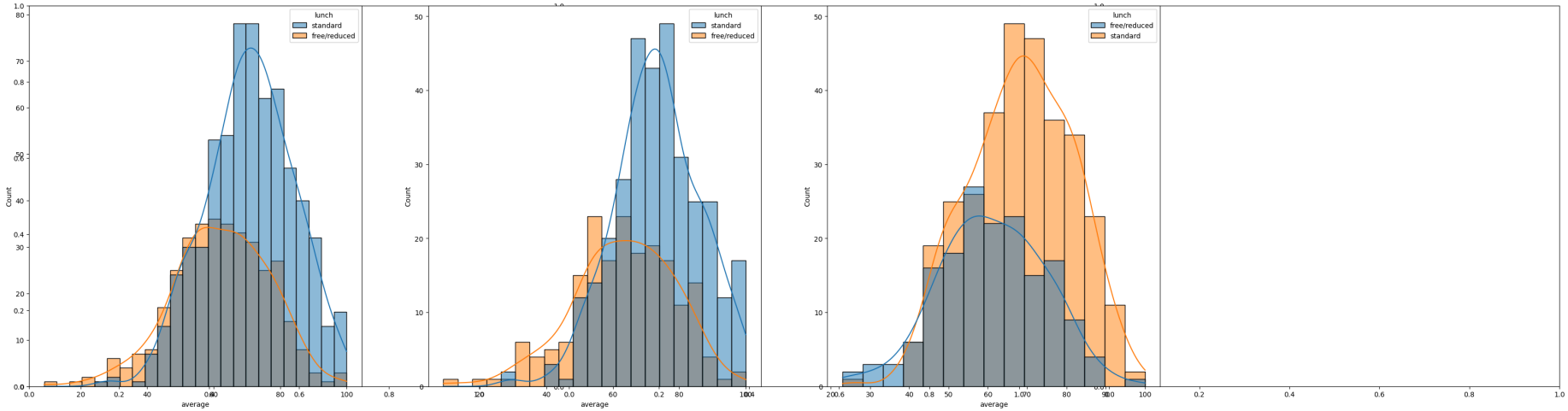
In [60]:
```
fig,axis=plt.subplots(1,2,figsize=(15,7))
plt.subplot(121)
sns.histplot(data=df,x='average',bins=30,kde=True,color='g')
plt.subplot(122)
sns.histplot(data=df,x='average',bins=30,kde=True,hue='gender')
plt.show()
```





Female student tend to perform well than male students

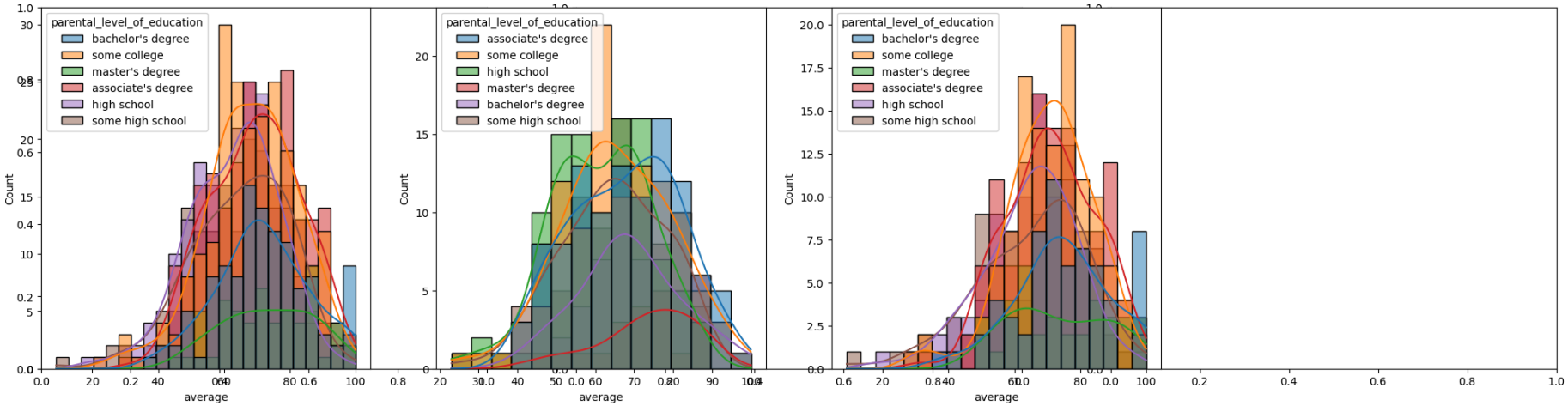
In [61]:
```
plt.subplots(1,3,figsize=(40,10))
plt.subplot(141)
sns.histplot(data=df,x='average',kde=True,hue='lunch')
plt.subplot(142)
sns.histplot(data=df[df.gender=='female'],x='average',kde=True,hue='lunch')
plt.subplot(143)
sns.histplot(data=df[df.gender=='male'],x='average',kde=True,hue='lunch')
plt.show()
```



Insights

- Standard Lunch help students perform well in exams.
- Standard lunch helps perform well in exams be it a male of female

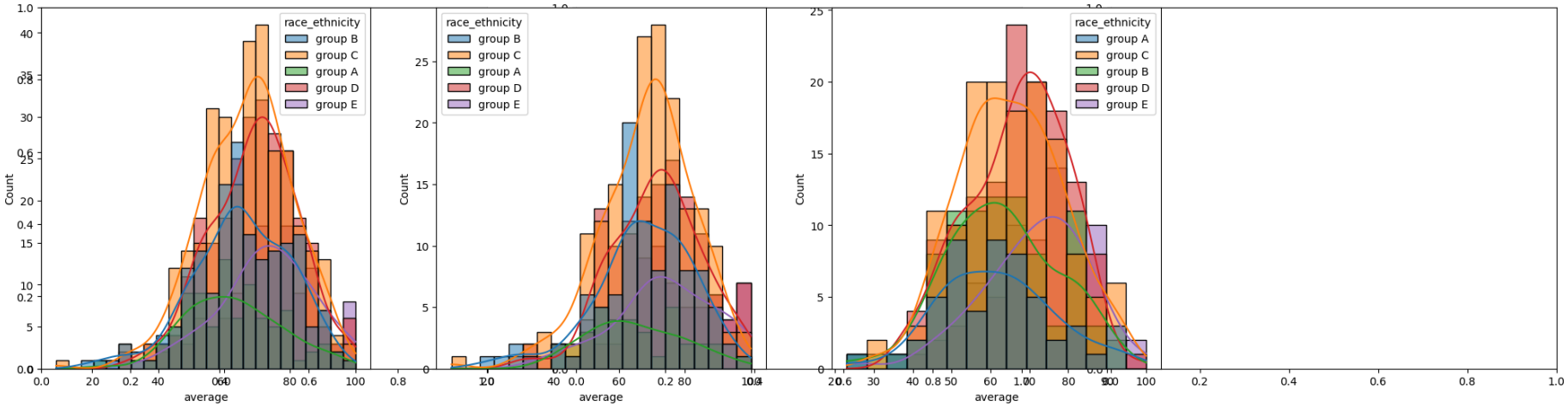
In [62]:
```
plt.subplots(1,3,figsize=(25,6))
plt.subplot(141)
ax =sns.histplot(data=df,x='average',kde=True,hue='parental_level_of_education')
plt.subplot(142)
ax =sns.histplot(data=df[df.gender=='male'],x='average',kde=True,hue='parental_level_of_education')
plt.subplot(143)
ax =sns.histplot(data=df[df.gender=='female'],x='average',kde=True,hue='parental_level_of_education')
plt.show()
```



More Insight

- In general parent's education don't help student perform well in exam.
- 2nd plot shows that parent's whose education is of associate's degree or master's degree their male child tend to perform well in exam
- 3rd plot we can see there is no effect of parent's education on female students.

In [63]:
```
plt.subplots(1,3,figsize=(25,6))
plt.subplot(141)
ax =sns.histplot(data=df,x='average',kde=True,hue='race_ethnicity')
plt.subplot(142)
ax =sns.histplot(data=df[df.gender=='female'],x='average',kde=True,hue='race_ethnicity')
plt.subplot(143)
ax =sns.histplot(data=df[df.gender=='male'],x='average',kde=True,hue='race_ethnicity')
plt.show()
```



Insights

- Students of group A and group B tends to perform poorly in exam.
- Students of group A and group B tends to perform poorly in exam irrespective of whether they are male or female

In [ ]: