ALY 6010 Probability Theory & Introductory Statistics CRN 22162

Final Project Report: Student Performance

Submitted to: Professor Behzad Ahmadi

Northeastern University Silicon Valley Campus

Submitted by: Sourabh D Khot

CPS Analytics
NUID: 002754952

Date: April 1, 2022

Introduction

This final project analyzes the 'StudentsPerformance.csv' dataset from Kaggle. This dataset has students' backgrounds and marks secured by them in various subjects in 2021.

I have created the following storyline for this project:
We are a huge group of schools, and in 2021, we randomly collected a sample data of 1000 students containing their background and exam scores of math, reading, writing subjects. Before the exam, students were provided a preparatory course to complete. Also, from 2020 to 2021, we had revised the math curriculum. Due to limited resources, we will be able to teach only two of the three subjects in 2022.

In this consolidated and condensed report, I will:

- Clean data and perform an exploratory data analysis (EDA)
- Formulate questions related to individual variables and relationship between variables
- Study questions related to individual variables using hypothesis testing
- Study relationship between variables using correlation and regression
- Provide a conclusion from all interpretations

A. Exploratory data analysis

Preparing data

In this section, we will import, clean, and prepare the data.

In [1]:
```
import pandas as pd     #importing libraries
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pingouin as pg
import statsmodels.api as sm
```

In [2]:
```
df = pd.read_csv('Datasets/StudentsPerformance.csv') #importing dataset
```

In [3]:
```
df.info()  #no null values to clean
```

```
<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 [4]:
```
#no columns to drop, renaming headers
df.rename(columns = {'race/ethnicity':'race_ethnicity','lunch':'lunch_plan',\
                     'parental level of education':'parent_education',\
                     'test preparation course':'prep_completed',\
                     'math score':'math','reading score':'reading',\
                     'writing score':'writing'}, inplace = True)
```

In [5]:
```
#finding unique in categorical attributes
for col in df.iloc[:,[0,1,2,3,4]]:
    print(col,df[col].unique())
```

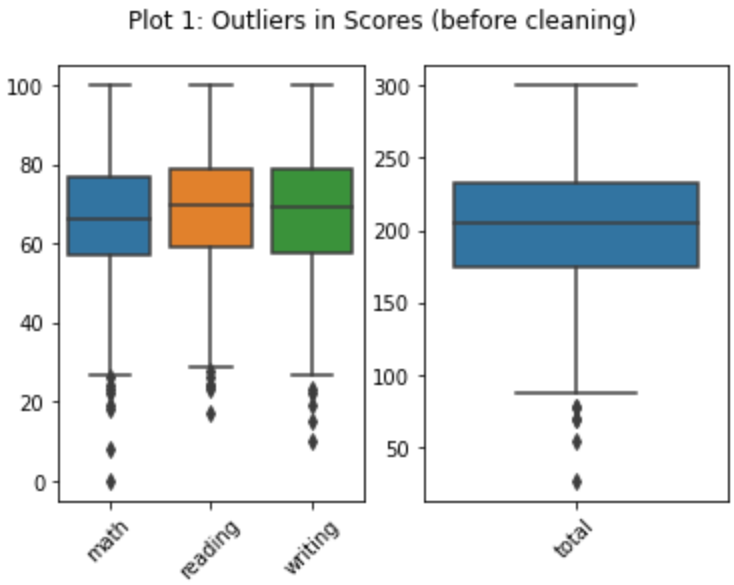
```
gender ['female' 'male']
race_ethnicity ['group B' 'group C' 'group A' 'group D' 'group E']
parent_education ["bachelor's degree" 'some college' "master's degree" "associate's degre
e"
 'high school' 'some high school']
lunch_plan ['standard' 'free/reduced']
prep_completed ['none' 'completed']
```

In [6]:
```
#chanding variable types
df.gender = df.gender.astype('category')
df.race_ethnicity = df.race_ethnicity.astype('category')
df.lunch_plan = df.lunch_plan.astype('category')

#changing binary values to boolean
df.prep_completed = df.prep_completed.map(\
                           {'completed': True,'none':False})

#setting order of parental education from low to high
from pandas.api.types import CategoricalDtype
df.parent_education = df.parent_education.astype(CategoricalDtype(\
            ['some high school','high school','some college',"associate's degree",\
             "bachelor's degree","master's degree"], ordered=True))
```

In [7]:
```
#adding a computed variable of total score
df['total'] = df.math + df.reading + df.writing
```

In [8]:
```
#finding outliers through boxplot
fig, axs = plt.subplots(1,2,sharey=False)
fig.suptitle('Plot 1: Outliers in Scores (before cleaning)')
sns.boxplot(ax=axs[0],data=df.iloc[:,[5,6,7]])
sns.boxplot(ax=axs[1],data=df.iloc[:,[8]])
for tick in axs[0].get_xticklabels():
```

```
 tick.set_rotation(45)
for tick in axs[1].get_xticklabels():
 tick.set_rotation(45)
```



Six outliers of the total score will be removed. Outliers are not removed at subject score level because a student maybe not be good at one subject but good at others. However, those whose total scores are outliers may have a learning disability. Since there is no learning disability attribute, it is better to exclude the total score outliers when we are analzying the data.

In [9]:
```
#removing outlier rows of total score, which maybe special students
Q1 = np.percentile(df.total, 25, interpolation = 'midpoint')
Q3 = np.percentile(df.total, 75, interpolation = 'midpoint')
IQR = Q3 - Q1
upper = np.where(df.total > (Q3+1.5*IQR))
lower = np.where(df.total < (Q1-1.5*IQR))
df.drop(upper[0], inplace = True)
df.drop(lower[0], inplace = True)
```

The data is ready for exploration.

Exploring data

We will analyze the numerical variables for descriptive statistics and categorical variables for proportions.

In [10]:
```
print('\033[1m' + 'Table 1: Glimpse of the Data' +'\033[0m')
df.head()
```

Out[10]:

Table 1: Glimpse of the Data

	gender	race_ethnicity	parent_education	lunch_plan	prep_completed	math	reading	writing	total
0	female	group B	bachelor's degree	standard	False	72	72	74	218
1	female	group C	some college	standard	True	69	90	88	247
2	female	group B	master's degree	standard	False	90	95	93	278
3	male	group A	associate's degree	free/reduced	False	47	57	44	148
4	male	group C	some college	standard	False	76	78	75	229

In [11]:

```
print('\033[1m' + 'Table 2: Descriptive Statistics of Scores' +'\033[0m')
stats = df.iloc[:,[5,6,7,8]].describe()
stats.loc['var'] = df.iloc[:,[5,6,7,8]].var().tolist()
stats.loc['skew'] = df.iloc[:,[5,6,7,8]].skew().tolist()
stats.loc['kurt'] = df.iloc[:,[5,6,7,8]].kurtosis().tolist()
stats.transpose()
```

Table 2: Descriptive Statistics of Scores

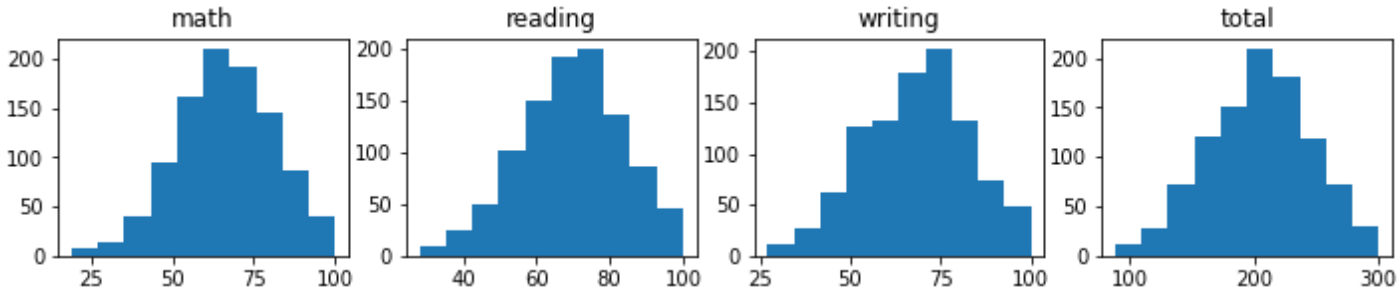
Out[11]:

	count	mean	std	min	25%	50%	75%	max	var	skew	kurt
math	994.0	66.373239	14.731519	19.0	57.0	66.0	77.00	100.0	217.017652	-0.128832	-0.181525
reading	994.0	69.439638	14.216740	28.0	60.0	70.0	79.75	100.0	202.115688	-0.144140	-0.345601
writing	994.0	68.347082	14.757020	27.0	58.0	69.0	79.00	100.0	217.769644	-0.152696	-0.408148
total	994.0	204.159960	41.456338	88.0	175.0	206.0	233.75	300.0	1718.627962	-0.144487	-0.312543

In [12]:

```
#  All scores are approximately normally distributed
print('\033[1m' + 'Plot 2: Distribution of Scores' +'\033[0m')
plt.figure(figsize=(12,2))
plt.subplot(1,4,1); plt.hist(df.math); plt.title('math')
plt.subplot(1,4,2); plt.hist(df.reading); plt.title('reading')
plt.subplot(1,4,3); plt.hist(df.writing); plt.title('writing')
plt.subplot(1,4,4); plt.hist(df.total); plt.title('total');
```

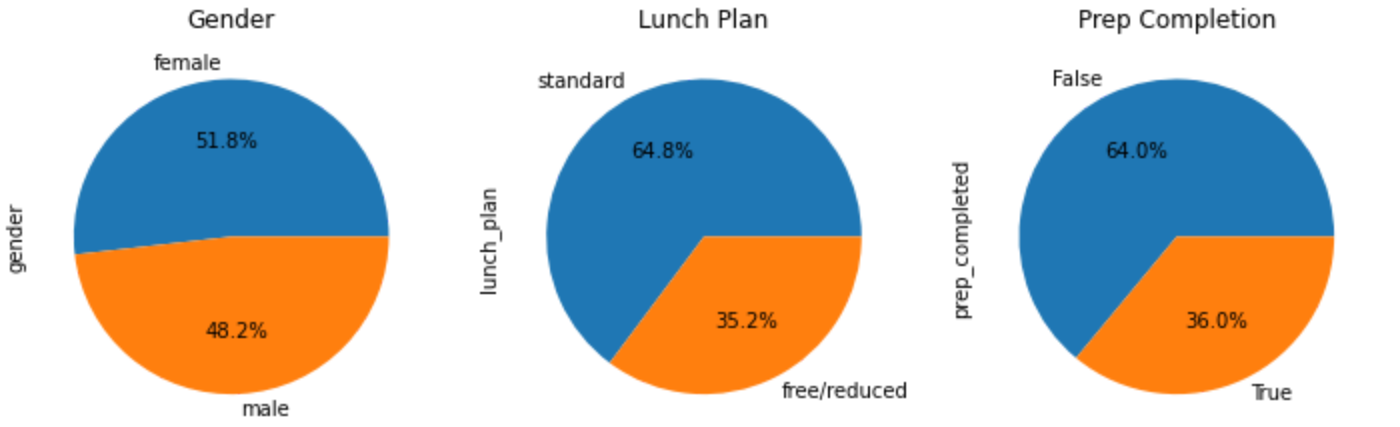
Plot 2: Distribution of Scores



In [13]:

```
print('\033[1m' + 'Plot 3: Proportion of Students by Categories' +'\033[0m')
plt.figure(figsize=(12,4))
plt.subplot(1,3,1);df.gender.value_counts().plot(kind='pie',autopct='%1.1f%%',title='Gende
plt.subplot(1,3,2);df.lunch_plan.value_counts().plot(kind='pie',autopct='%1.1f%%',title='I
plt.subplot(1,3,3);df.prep_completed.value_counts().plot(kind='pie',autopct='%1.1f%%',titl
```

Plot 3: Proportion of Students by Categories

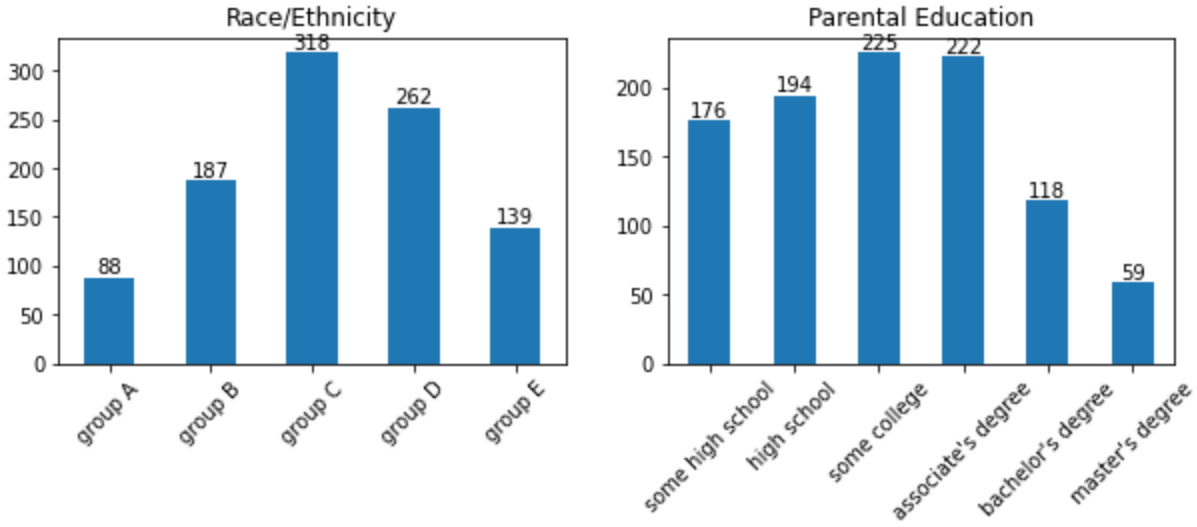


In [14]:

```
print('\033[1m' + 'Plot 4: Count of Students by Categories' +'\033[0m')
plt.figure(figsize=(10,3))
```

```
plt.subplot(1,2,1); ax = df.race_ethnicity.value_counts(sort=False).plot.bar(title='Race/E
ax.bar_label(ax.containers[0]); plt.xticks(rotation=45);
plt.subplot(1,2,2); ax = df.parent_education.value_counts(sort=False).plot.bar(title='Pare
ax.bar_label(ax.containers[0]); plt.xticks(rotation=45);
```

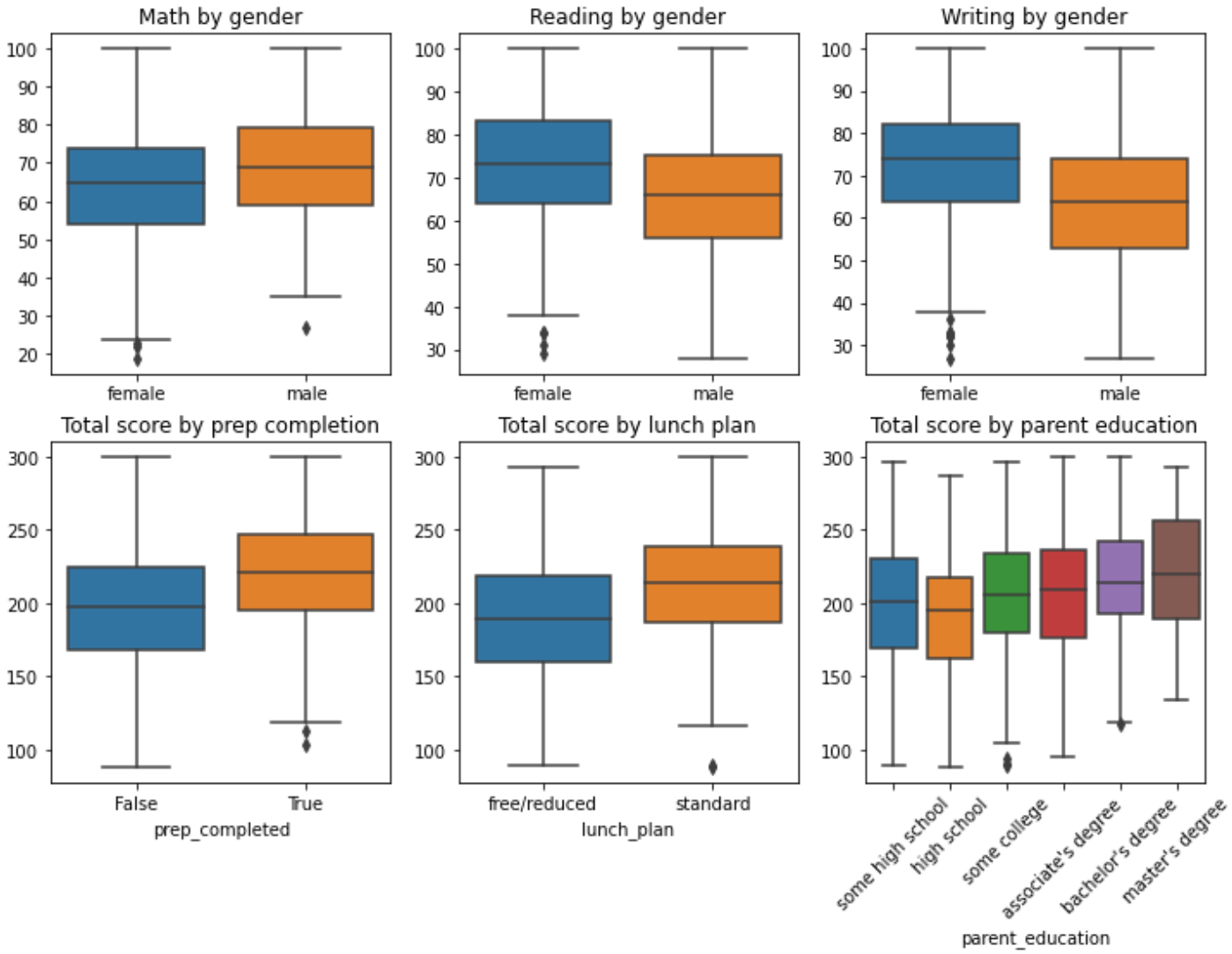
Plot 4: Count of Students by Categories



In [15]:

```
fig, axs = plt.subplots(2,3,figsize=(12,8))
fig.suptitle('Plot 5: Student Scores by Subset of Categories')
sns.boxplot(ax=axs[0,0],x=df.gender,y=df.math).set(xlabel=None,ylabel=None); axs[0,0].titl
sns.boxplot(ax=axs[0,1],x=df.gender,y=df.reading).set(xlabel=None,ylabel=None); axs[0,1].t
sns.boxplot(ax=axs[0,2],x=df.gender,y=df.writing).set(xlabel=None,ylabel=None); axs[0,2].t
sns.boxplot(ax=axs[1,0],x=df.prep_completed,y=df.total).set(ylabel=None); axs[1,0].title.s
sns.boxplot(ax=axs[1,1],x=df.lunch_plan,y=df.total).set(ylabel=None); axs[1,1].title.set_t
sns.boxplot(ax=axs[1,2],x=df.parent_education,y=df.total).set(ylabel=None); axs[1,2].title
plt.setp(axs[1,2].get_xticklabels(), rotation=45);
```

Plot 5: Student Scores by Subset of Categories



Key Insights from EDA

In general, the following observations can be made of the sample from the EDA:

- Math scores are lower than those of reading and writing. Females are good at reading and writing, and males are good at math.
- Only 36% of the students completed the test preparation course, and those who completed have higher total scores than those who did not.
- The sample has the highest representation from 'group C' and those with parent education of 'some college' or 'associate degree'.
- Higher parental education is associated with higher total scores.
- About 65% of students opted for a 'standard' lunch plan and had better total scores than those with a 'free/reduced' plan.

B. Questions to explore

The dataset has complete details about the sample. However, our questions are pertaining to the population, so we will use statistical methods to understand population parameters from sample statistics with a confidence level of 95%.

About individual variables

> (1) Has the population math score changed from 70 due to curriculum revision?

I am assuming that the population mean math score in 2020 was 70, and there was curriculum revision in 2021, whose sample we are analyzing. As per EDA table 2, the sample mean is 66.3 (table 2). I want to test if the score has changed due to curriculum revision and not because of sampling error.

> (2) Do students who completed the test prep course have better total scores than those who did not?

I want to know whether the prep course is beneficial. As per EDA plot 5, those who completed it had a better total score, but I want to verify that it is not due to sampling error.

About relationship between variables

We have only three independent numerical variables- math score, reading score, and writing score. Hence any correlation and regression questions would have to be done between these variables.

> (3) Scores of which two subjects have the highest correlation?

Suppose the school has limited resources and wants to reduce duplication of efforts and teach only two of the three subjects. Then finding the two most correlated subjects and skipping one of them may help achieve this objective, assuming there is causation and no other factors involved.

> (4) By how much writing score will increase for a unit increase in reading score?

As we will see in section D, reading and writing scores have the highest correleation. Assuming there will be no teacher for writing, the reading subject teacher wants to understand how much a student's writing score will increase when there is an improvement in their reading score by one unit, assuming there is causation and no other factors involved.

C. Studying individual variables

This section will address questions related to individual variables using one-sample and two-sample tests.

(1) Testing if math score has changed

Post curriculum revision, we want to test whether the population mean math score has changed from 70, assuming a confidence level of 95%.

Hypothesis

- Null hypothesis, H0: $\mu = 70$
- Alternate hypothesis, H1: $\mu \neq 70$

n = 994 (>30), $\alpha = 0.05$, population variance not known. Hence, we will use one-sample two-sided t-test.

Testing

In [16]:
```
print('\033[1m' + 'Table 3: Testing if math score has changed' +'\033[0m')
pg.ttest(x=df.math,y=70,alternative='two-sided',confidence=0.95)
```

Table 3: Testing if math score has changed

Out[16]:

	T	dof	alternative	p-val	CI95%	cohen-d	BF10	power
T-test	-7.761838	993	two-sided	2.074722e-14	[65.46, 67.29]	0.246191	1.649e+11	1.0

Interpretation

The hypothesized mean (70) is outside the 95% confidence interval (65.46 to 67.29). Hence we have sufficient evidence to reject the null hypothesis. This implies that the population mean math score is no longer the same, and the change in curriculum did impact scores.

(2) Testing if prep course improves total score

We will test if students completing the prep course have better total scores than those who did not, assuming a confidence level of 95%.

In [17]:

```
print('\033[1m' + 'Table 4: Total score statistics by Prep Completion' +'\033[0m')
df.iloc[:,[4,8]].groupby('prep_completed').describe()
```

Table 4: Total score statistics by Prep Completion

Out[17]:

	total							
	count	mean	std	min	25%	50%	75%	max
prep_completed								
False	636.0	196.36478	40.725804	88.0	168.0	197.0	225.0	300.0
True	358.0	218.00838	39.110881	103.0	195.0	220.5	246.5	300.0

Hypothesis

- Null hypothesis, H0: μ.prep = μ.non_prep
- Alternate hypothesis, H1: μ.prep > μ.non_prep

n.prep = 358, n.non_prep = 636 (>30), α = 0.05, population variance is not known and assumed to be unequal. Hence, it is an unpaired two-sample one-sided Welch's t-test.

Testing

In [18]:

```
print('\033[1m' + 'Table 5: Testing if prep course improves total score' +'\033[0m')
pg.ttest(x=df[df.prep_completed==True].total, y=df[df.prep_completed==False].total, alterr
```

Table 5: Testing if prep course improves total score

Out[18]:

	T	dof	alternative	p-val	CI95%	cohen-d	BF10	power
T-test	8.251159	765.459182	greater	3.417960e-16	[17.32, inf]	0.53904	1.927e+13	1.0

Interpretation

The p-value is much less than α (0.05). Hence we have sufficient evidence to reject the null hypothesis. We can say that completing the test preparation course will improve the total score.

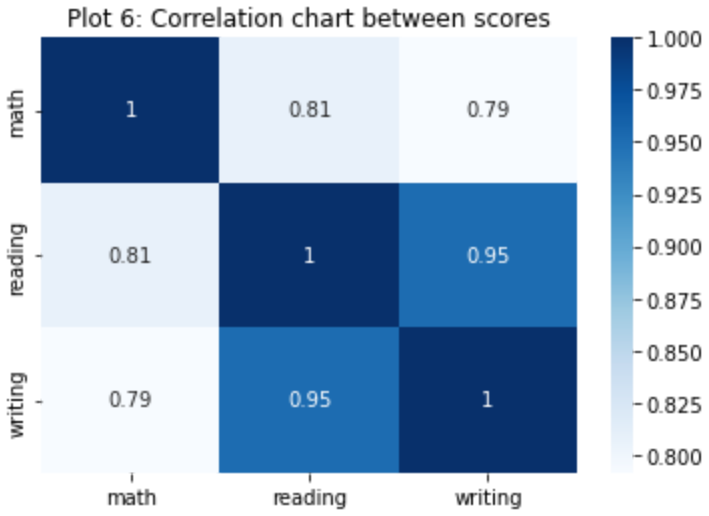
D. Studying relationship between variables

In this section, we will find the correlation between scores, and understand how much one of them affects the other.

(3) Finding highest correlation between scores

Let us use a correlation chart with values embedded to achieve correlation between scores of this sample.

In [19]:
```
plt.figure(figsize=(6,4))
sns.heatmap(df.iloc[:,[5,6,7]].corr(),cbar=True,annot=True,cmap='Blues')
plt.title('Plot 6: Correlation chart between scores');
```



We observe that reading and writing scores in the sample have the highest correlation of 0.95.

Testing for significant correlation

To ensure that the population also has a significant correlation between reading and writing scores, we will perform a correlation coefficient t-test with a significance level of 95% and n = 994.

- Null hypothesis, H0: ρ = 0
- Alternate hypothesis, H1: ρ ≠ 0

In [20]:
```
from scipy import stats
corr_coef, p_value = stats.pearsonr(df.reading, df.writing)
print("Correlation Coefficient =", corr_coef, "with p-value =", p_value)
```

```
Correlation Coefficient = 0.9518364981787317 with p-value = 0.0
```

Interpretation

The p-value is lower than 0.05, so we have sufficient evidence to reject the null hypothesis. Hence reading score is significantly correlated with writing score. The correlation is positive, i.e. writing score increases when reading score increases, and vice versa.

Hence, the school may choose to teach only one among reading and writing subjects, and students will still score in the other subject, assuming there is causation and no other factors involved.

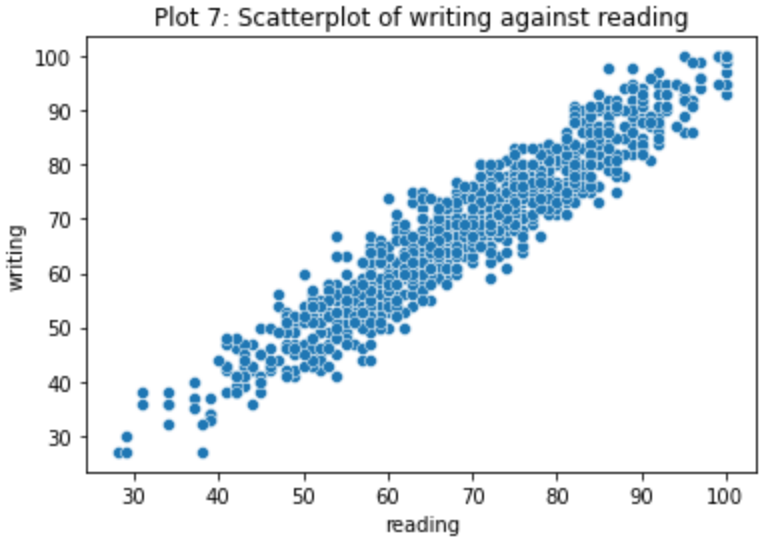
(4) Regression of writing score on reading score

We have assumed the school will have no teacher for writing. The reading subject teacher wants to understand how much a student's writing score will increase, when their reading score is improved by a unit.

Here, reading score is the predictor variable and writing score is the output variable.

Scatter plot

In [21]:
```
sns.scatterplot(x=df.reading, y=df.writing)
plt.title('Plot 7: Scatterplot of writing against reading');
```



From the scatter plot, the relationship looks linear. Hence we will perform a linear regression analysis.

Linear Regression Table

In [22]:
```
x = df.reading.to_numpy().reshape((-1, 1))
y = df.writing.to_numpy()
x = sm.add_constant(x)     #to calculate intercept B0
model = sm.OLS(y, x)       #regression using ordinary least squares
results = model.fit()
results.summary(title='Table 6: Results of OLS regression of writing on reading')
```

Out[22]:

Table 6: Results of OLS regression of writing on reading

Dep. Variable:	y	**R-squared:**	0.906
Model:	OLS	**Adj. R-squared:**	0.906
Method:	Least Squares	**F-statistic:**	9560.
Date:	Mon, 04 Apr 2022	**Prob (F-statistic):**	0.00
Time:	22:37:24	**Log-Likelihood:**	-2910.4
No. Observations:	994	**AIC:**	5825.
Df Residuals:	992	**BIC:**	5835.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-0.2599	0.716	-0.363	0.717	-1.665	1.146
x1	0.9880	0.010	97.777	0.000	0.968	1.008

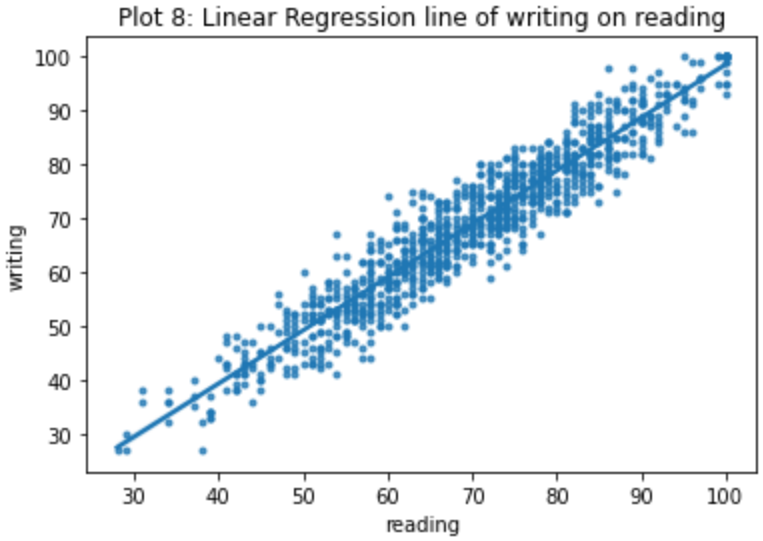
Omnibus:	1.739	**Durbin-Watson:**	1.974
Prob(Omnibus):	0.419	**Jarque-Bera (JB):**	1.805

Skew:	0.082	**Prob(JB):**	0.405
Kurtosis:	2.871	**Cond. No.**	354.

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Linear Regression Line

In [23]:
```
sns.regplot(x=df.reading, y=df.writing, marker='.')
plt.title('Plot 8: Linear Regression line of writing on reading');
```



Interpretation

As per table 6, an R-squared of 0.906 indicates that 90.6% of the variation in writing score is explained by reading score, which is excellent proportion. Following is the best fit linear regression line:

- (writing score) = -0.259 + 0.998(reading score)

Hence, when the reading teacher improves a student's reading score by a unit, the writing score will be improved by 0.988 (95% confidence interval of 0.968 to 1.008), assuming there is causation and no other factors involved.

E. Conclusion

We imported, cleaned, and performed an EDA of the sample dataset. We asked four questions regarding the population and addressed them using statistical tools of hypothesis testing, correlation, and regression analysis with a 95% confidence level.

Here are the key takeaways about the student population:

- Post math curriculum change in 2021, Math scores have significantly dropped, and the subject needs to be focused.
- Those who completed the test preparation course had significantly better total scores than those who did not (although only 36% of the sample completed the prep).

- Scores of reading and writing have the highest correlation. Hence, due to limited resources, among the two subjects, the school may choose to teach only reading subject, and students will still score in writing (assuming there is causation and no other factors involved).
- When the reading subject teacher improves a student's reading score by 1, their writing score will improve between a range of 0.968 to 1.008, with a mean of 0.998.

There were more insights of the sample data, but given the context, they are of lesser importance and hence not tested for the population.

References

[1] northeastern.instructure.com/courses/105627/assignments/1206546
[2] kaggle.com/datasets/spscientist/students-performance-in-exams
[3] stackoverflow.com/questions/25239933/how-to-add-title-to-subplots-in-matplotlib
[4] ibm.com/docs/en/watson-studio-local/1.2.3?topic=notebooks-markdown-jupyter-cheatsheet
[5] seaborn.pydata.org/generated/seaborn.scatterplot.html
[6] seaborn.pydata.org/generated/seaborn.regplot.html
[7] analyticsvidhya.com/blog/2021/08/how-to-perform-exploratory-data-analysis-a-guide-for-beginners/
[8] towardsdatascience.com/an-extensive-guide-to-exploratory-data-analysis-ddd99a03199e
[9] geeksforgeeks.org/detect-and-remove-the-outliers-using-python/
[10] nickmccullum.com/python-visualization/subplots/
[11] statsmodels.org/stable/generated/statsmodels.graphics.mosaicplot.mosaic.html
[12] delftstack.com/howto/matplotlib/how-to-place-legend-outside-of-the-plot-in-matplotlib/
[13] pandas.pydata.org/docs/reference/api/pandas.DataFrame.plot.html
[14] stackoverflow.com/questions/25447700/annotate-bars-with-values-on-pandas-bar-plots
[15] regenerativetoday.com/three-very-useful-functions-of-pandas-to-summarise-data
[16] pingouin-stats.org/generated/pingouin.ttest.html
[17] scribbr.com/frequently-asked-questions/one-sample-t-test-vs-paired-t-test/