**Part 1:** Data Preparation with SQL – Creating a View

**I. Calculating a Subscription’s End Date**

1. **Identify the fields to include in your result set:** You’ll want to include all fields from the original table in your result set, but you’ll also need to create two new ones: date\_start and date\_end. date\_start will be the date\_purchased field from the original table, but date\_end must be calculated.

SELECT

purchase\_id,

student\_id,

plan\_id,

date\_purchased AS date\_start,

??? AS date\_end,

date\_refunded,

FROM

student\_purchases;

1. **Calculate the subscription end date:**You’ll need to use the plan\_id field to calculate the date\_end field. Depending on the plan\_id, you’ll add a different number of months to the date\_purchased field. To handle the varying number of months to add, you can use a CASE statement inside your calculation.

SELECT

purchase\_id,

student\_id,

plan\_id,

date\_purchased AS date\_start,

CASE

WHEN plan\_id = 0 THEN DATE\_ADD(date\_purchased, INTERVAL 1 MONTH)

WHEN plan\_id = 1 THEN DATE\_ADD(date\_purchased, INTERVAL 3 MONTH)

WHEN plan\_id = 2 THEN DATE\_ADD(date\_purchased, INTERVAL 12 MONTH)

WHEN plan\_id = 3 THEN CURDATE()

END AS date\_end,

date\_refunded

FROM

student\_purchases;

**II. Re-Calculating a Subscription’s End Date**

1. **Identify the fields to include in your result set:** Include the necessary fields from the sub-query created in the previous task.

SELECT

purchase\_id,

student\_id,

plan\_id,

date\_start,

??? AS date\_end

FROM

(

-- Sub-query created in the task "I. Calculating a Subscription's End Date"

) a;

1. **Handle refunds:** If a refund occurred, the subscription end date should be the refund date instead of the calculated end date. To handle this, use an IF statement or a similar control structure.

SELECT

purchase\_id,

student\_id,

plan\_id,

date\_start,

IF(date\_refunded IS NULL,

date\_end,

date\_refunded) AS date\_end

FROM

(

-- Sub-query created in the task "I. Calculating a Subscription's End Date"

) a;

**III. Creating Two ‘paid’ Columns and a MySQL View**

1. **Identify the fields to include in your result set:** Select all fields from the sub-query created in II. Re-Calculating a Subscription’s End Date. Include two new fields: paid\_q2\_2021 and paid\_q2\_2022.

SELECT

\*,

??? AS paid\_q2\_2021,

??? AS paid\_q2\_2022

FROM

(

-- Sub-query created in the task "II. Re-Calculating a Subscription's End Date"

) b;

1. **Identify if a student has an active subscription:** Use, for example, a CASE statement and check if a subscription period falls inside Q2—i.e.:
   * If the end date is before April 1, the student will have had a free plan (indicated by 0).
   * If the start date is after June 30, the student will have had a free plan (indicated by 0).
   * In all other cases, the student will have had an active subscription (indicated by 1).

SELECT

\*,

CASE

WHEN date\_end < '2021-04-01' THEN 0

WHEN date\_start > '2021-06-30' THEN 0

ELSE 1

END AS paid\_q2\_2021,

CASE

WHEN date\_end < '2022-04-01' THEN 0

WHEN date\_start > '2022-06-30' THEN 0

ELSE 1

END AS paid\_q2\_2022

FROM

(

-- Sub-query created in the task "II. Re-Calculating a Subscription's End Date"

) b;

1. **Create the view**: Add the following code lines at the beginning of your query to create the view. Refresh the schemas and ensure the view has appeared under Views in the data\_scientist\_project database.

DROP VIEW IF EXISTS purchases\_info;

CREATE VIEW purchases\_info AS

SELECT

\*,

CASE

WHEN date\_end < '2021-04-01' THEN 0

WHEN date\_start > '2021-06-30' THEN 0

ELSE 1

END AS paid\_q2\_2021,

CASE

WHEN date\_end < '2022-04-01' THEN 0

WHEN date\_start > '2022-06-30' THEN 0

ELSE 1

END AS paid\_q2\_2022

FROM

(

-- Sub-query created in the task "II. Re-Calculating a Subscription's End Date"

) b;

Part1.zip

**Part 2:** Data Preparation with SQL – Splitting Into Periods

**I. Calculating Total Minutes Watched in Q2 2021 and Q2 2022**

1. **Identify the necessary fields:** Select the student\_id field from the student\_video\_watched table and calculate the total minutes watched. Round the result to two decimal places.

SELECT

student\_id,

ROUND(SUM(seconds\_watched) / 60, 2) AS minutes\_watched

FROM

student\_video\_watched;

1. **Aggregate by student:** Perform the aggregation such that you calculate the minutes watched for each student.

SELECT

student\_id,

ROUND(SUM(seconds\_watched) / 60, 2) AS minutes\_watched

FROM

student\_video\_watched

GROUP BY student\_id;

1. **Add a condition:** Add a condition that filters 2021 and 2022.

SELECT

student\_id,

ROUND(SUM(seconds\_watched) / 60, 2) AS minutes\_watched

FROM

student\_video\_watched

WHERE

YEAR(date\_watched) = 2021

GROUP BY student\_id;

**II. Creating a ‘paid’ Column**

1. **Identify the tables:** We need the information from the sub-query we created in the previous task (I. Calculating Total Minutes Watched in Q2 2021 and Q2 2022) and the purchases\_info view. Consider joining the sub-query and the view such that you retrieve all records from the sub-query regardless of whether a student has a record in purchases\_info or has never purchased a subscription.

SELECT

???

FROM

(

-- Sub-query created in the task 'I. Calculating Total Minutes Watched in Q2 2021 and Q2 2022'

) a

LEFT JOIN purchases\_info i ON a.student\_id = i.student\_id;

1. **Retrieve the necessary columns and create the paid\_in\_q2 one:** Retrieve the student\_id and miutes\_watched columns. Use an IF statement to check if a student has a record in the purchases\_info view. Aggregate the results such that you obtain one record per student.

SELECT

a.student\_id,

a.minutes\_watched,

IF(

i.date\_start IS NULL,

0,

MAX(i.paid\_q2\_2022)

) AS paid\_in\_q2

FROM

(

-- Sub-query created in the task 'I. Calculating Total Minutes Watched in Q2 2021 and Q2 2022'

) a

LEFT JOIN purchases\_info i ON a.student\_id = i.student\_id

GROUP BY

student\_id;

1. **Create the four data sources:** Filter the data to obtain all four datasets described in the task.

SELECT

a.student\_id,

a.minutes\_watched,

IF(

i.date\_start IS NULL,

0,

MAX(i.paid\_q2\_2022)

) AS paid\_in\_q2

FROM

(

-- Sub-query created in the task 'I. Calculating Total Minutes Watched in Q2 2021 and Q2 2022'

) a

LEFT JOIN purchases\_info i ON a.student\_id = i.student\_id

GROUP BY

student\_id

HAVING paid\_in\_q2 = 1;

Part2.zip

**Part 3:** Data Preparation with SQL – Certificates Issued

**I. Studying Minutes Watched and Certificates Issued**

1. **Create a sub-query:** Select the student\_id column from the student\_certificates table. Then, count the number of certificates each student has been issued and store the result in a column called certificates\_issued.

SELECT

student\_id,

COUNT(certificate\_id) AS certificates\_issued

FROM

student\_certificates

GROUP BY student\_id;

1. **Join tables:** Use an appropriate JOIN clause to join the sub-query with the student\_video\_watched table.

SELECT

???,

???,

???

FROM

(

-- Sub-query

) a

LEFT JOIN

student\_video\_watched w ON a.student\_id = w.student\_id

GROUP BY student\_id;

1. **Select the relevant fields:** Select all records from the sub-query, retrieving the student\_id and certificates\_issued columns. For the students entering the sub-query, calculate the number of minutes watched and store the result in a column called minutes\_watched. Note that a student with an issued certificate shouldn’t have necessarily watched a video, so their record in the minutes\_watched column should be 0.

SELECT

a.student\_id,

IF(w.seconds\_watched IS NULL,

0,

ROUND(SUM(seconds\_watched) / 60, 2)) AS minutes\_watched,

a.certificates\_issued

FROM

(

-- Sub-query

) a

LEFT JOIN

student\_video\_watched w ON a.student\_id = w.student\_id

GROUP BY student\_id;

Part3.zip

**Part 4:** Data Preprocessing with Python – Removing Outliers

**I. Plotting the Distributions**

1. **Importing necessary libraries:** Import the pandas library for data manipulation and matplotlib and seaborn for data visualization.

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

sns.set()

1. **Loading the data:** Load the CSV data files into four separate pandas DataFrames. Remember that pandas has a function read\_csv(), which loads a CSV file and returns a DataFrame.

data1 = pd.read\_csv('minutes\_watched\_2021\_paid\_0.csv')

data2 = pd.read\_csv('minutes\_watched\_2022\_paid\_0.csv')

data3 = pd.read\_csv('minutes\_watched\_2021\_paid\_1.csv')

data4 = pd.read\_csv('minutes\_watched\_2022\_paid\_1.csv')

1. **Initial data exploration:** Examine the loaded data using the head() method on the DataFrames to display the first few rows.

data1.head()

data2.head()

data3.head()

data4.head()

1. **Plotting the distributions:** To plot the distributions, you will use the seaborn method kdeplot(). It takes as an argument the column of the DataFrame you wish to plot. The column is selected using the DataFrame variable followed by the column name in square brackets.

fig, axes = plt.subplots(2, 2, figsize =(15,10))

sns.kdeplot(data1['minutes\_watched'], ax = axes[0, 0]).set(title = 'Free-Plan Students for Q2 2021')

sns.kdeplot(data2['minutes\_watched'], ax = axes[0, 1]).set(title = 'Free-Plan Students for Q2 2022')

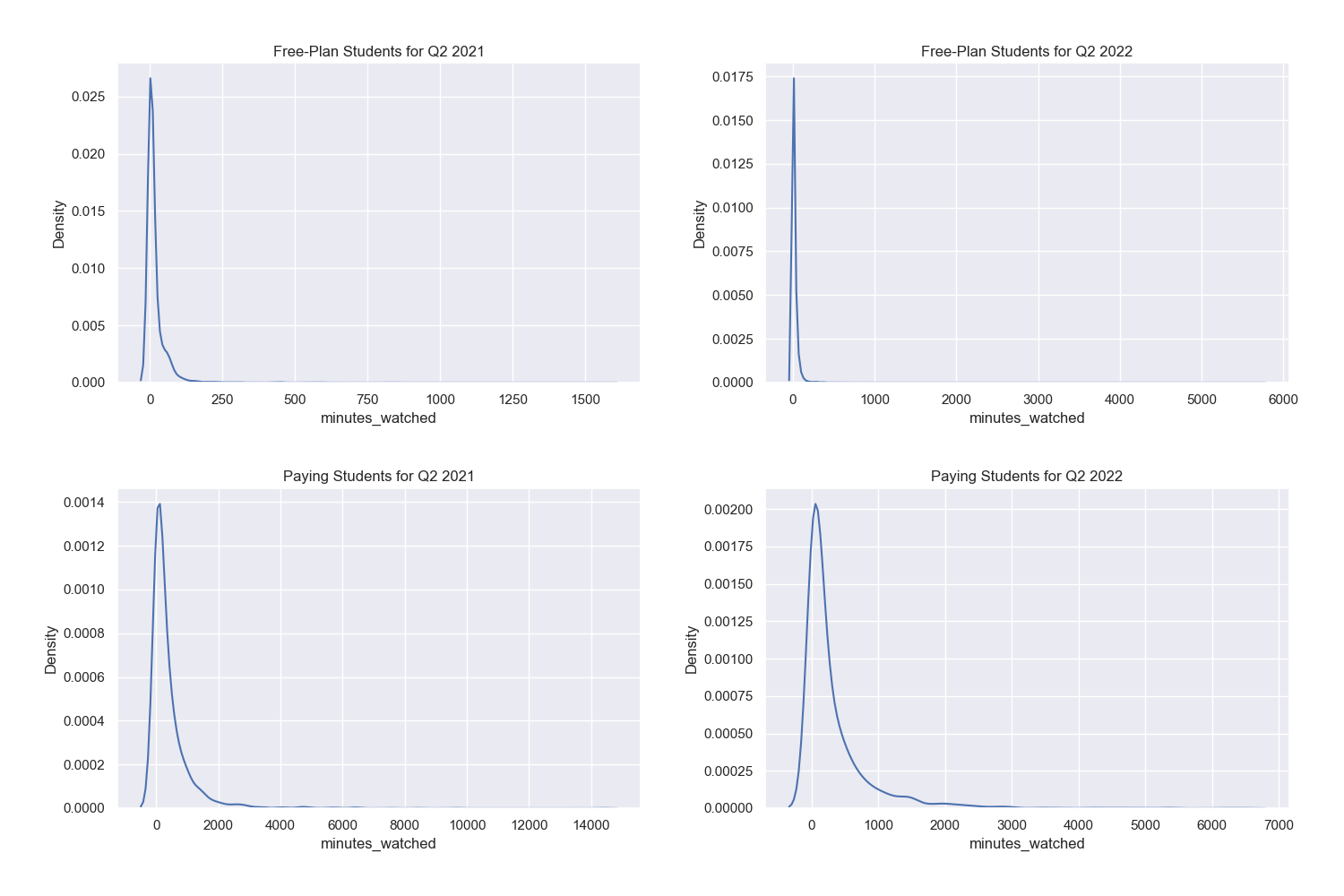
sns.kdeplot(data3['minutes\_watched'], ax = axes[1, 0]).set(title = 'Paying Students for Q2 2021')

sns.kdeplot(data4['minutes\_watched'], ax = axes[1, 1]).set(title = 'Paying Students for Q2 2022')

fig.tight\_layout(pad = 3.0)

plt.show()

By examining the plot, you can check if the distribution is skewed to the left or right (negative or positive skewness, respectively). If the distribution is symmetric around its mean, it’s not skewed. If the distribution is skewed, it tells us that the data is not symmetrically distributed around the mean.



As the plots reveal, all distributions of the minutes students watched are skewed to the right. This suggests some outliers in the data who have watched much more than most of the students on the platform. Their presence in the data will skew all metrics we’ll analyze later, such as the mean, median, and standard deviation.

**II. Removing the Outliers**

1. **Checking for outliers:** First, you can use pandas’ quantile() method to calculate the 99th percentile of the minutes\_watched column.

q1 = data1['minutes\_watched'].quantile(0.99)

q2 = data2['minutes\_watched'].quantile(0.99)

q3 = data3['minutes\_watched'].quantile(0.99)

q4 = data4['minutes\_watched'].quantile(0.99)

1. **Removing outliers:** Now that you have the 99th percentile value, you can use it to filter your DataFrame. You want to keep only those rows where minutes\_watched is less than this value using, for example, conditional filtering.

data\_no\_outliers1 = data1[data1['minutes\_watched']<q1]

data\_no\_outliers2 = data2[data2['minutes\_watched']<q2]

data\_no\_outliers3 = data3[data3['minutes\_watched']<q3]

data\_no\_outliers4 = data4[data4['minutes\_watched']<q4]

1. **Visualizing the filtered data:** After removing the outliers, plot and study the new distributions.

fig, axes = plt.subplots(2, 2, figsize =(15,10))

sns.kdeplot(data\_no\_outliers1['minutes\_watched'],

ax = axes[0, 0]).set(title = 'Free-Plan Students for Q2 2021 (no outliers)')

sns.kdeplot(data\_no\_outliers2['minutes\_watched'],

ax = axes[0, 1]).set(title = 'Free-Plan Students for Q2 2022 (no outliers)')

sns.kdeplot(data\_no\_outliers3['minutes\_watched'],

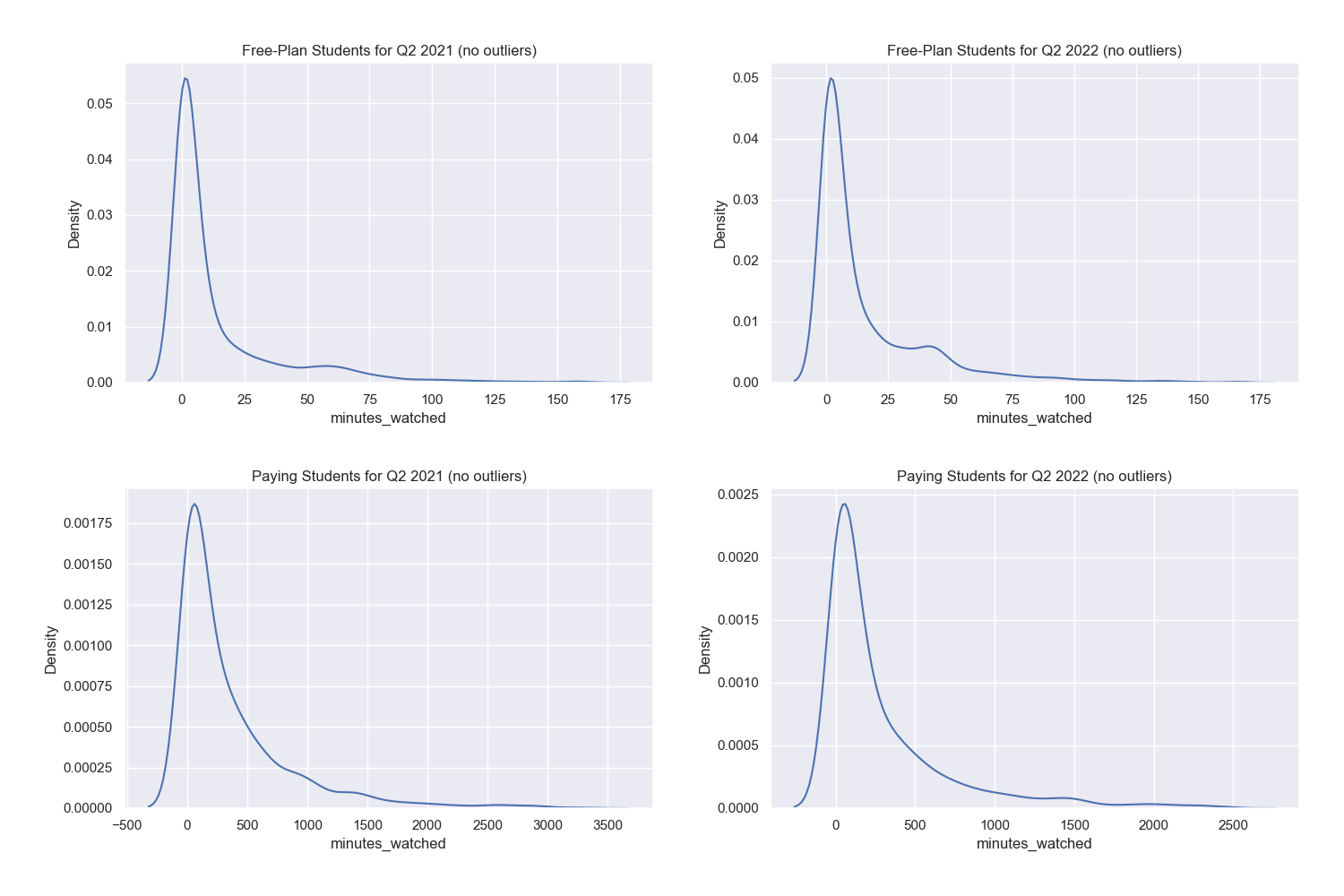
ax = axes[1, 0]).set(title = 'Paying Students for Q2 2021 (no outliers)')

sns.kdeplot(data\_no\_outliers4['minutes\_watched'],

ax = axes[1, 1]).set(title = 'Paying Students for Q2 2022 (no outliers)')

fig.tight\_layout(pad = 3.0)

plt.show()



1. **Saving the data as a CSV file.** Save the filtered data to a CSV file using pandas’ to\_csv() method. Ensure you follow the same steps to process all four datasets. At the end of this exercise, you should have obtained the following CSV files:  
   * minutes\_watched\_2021\_paid\_0\_no\_outliers.csv
   * minutes\_watched\_2022\_paid\_0\_no\_outliers.csv
   * minutes\_watched\_2021\_paid\_1\_no\_outliers.csv
   * minutes\_watched\_2022\_paid\_1\_no\_outliers.csv

data\_no\_outliers1.to\_csv('minutes\_watched\_2021\_paid\_0\_no\_outliers.csv',index=False)

data\_no\_outliers2.to\_csv('minutes\_watched\_2022\_paid\_0\_no\_outliers.csv',index=False)

data\_no\_outliers3.to\_csv('minutes\_watched\_2021\_paid\_1\_no\_outliers.csv',index=False)

data\_no\_outliers4.to\_csv('minutes\_watched\_2022\_paid\_1\_no\_outliers.csv',index=False)

Part4.zip

**Part 5:** Data Analysis with Excel – Hypothesis Testing

**I. Calculating Mean and Median Values**

1. **Insert the data:** In separate Excel tabs, open all four CSV files.
2. **Calculate the mean and median minutes:**Apply the Excel functions AVERAGE and MEDIAN to the minutes\_watched column to compute the mean and median values.

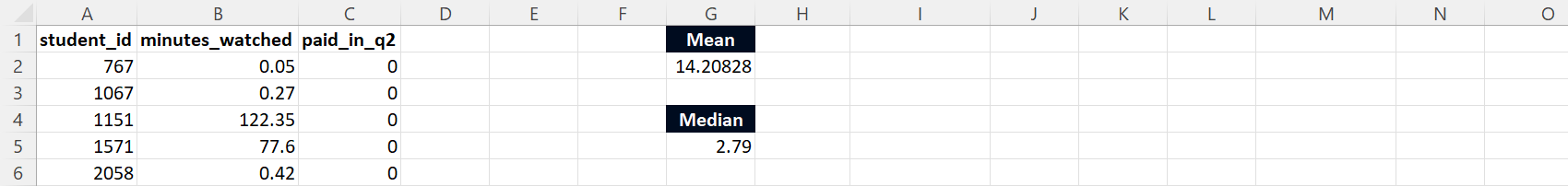
=AVERAGE(B2:B5281)

=MEDIAN(B2:B5281)

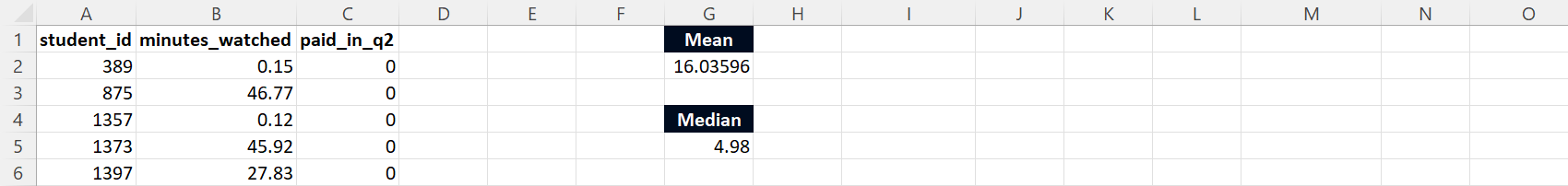
1. **Compare with the distribution plots:**

The mean is a measure of central tendency that calculates the average value of a dataset. In contrast, the median is a measure of central tendency that finds the middle value in a dataset. A significant difference between the mean and median can often indicate the presence of outliers or skew in the distribution since the mean is more sensitive to extreme values than the median.

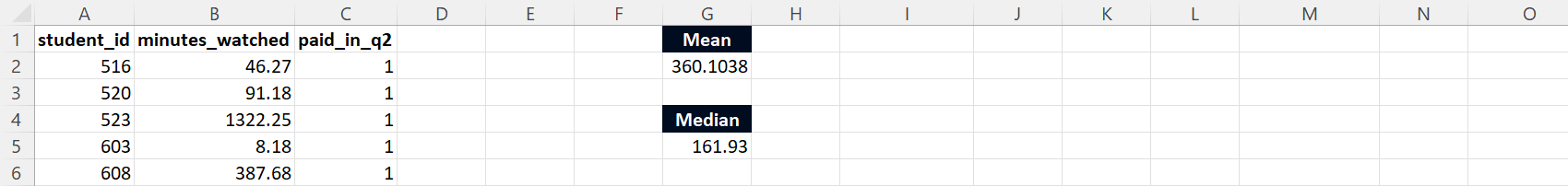
For free-plan students who watched in Q2 2021, the mean minutes watched are significantly higher than the median. This suggests a right-skewed distribution, indicating that a few students watched much more than others.

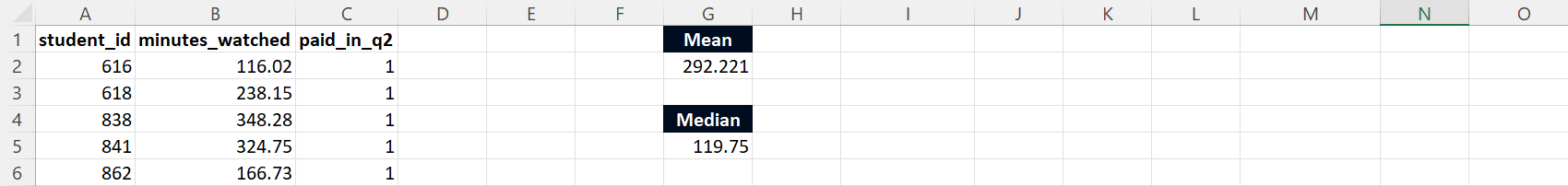


A similar situation is observed for free-plan students who watched in Q2 2022, with the mean being higher than the median, indicating right skewness.



The same applies to paying students who watched in Q2 2021 and those who watched in Q2 2022, where the mean is higher than the median, indicating right skewness.



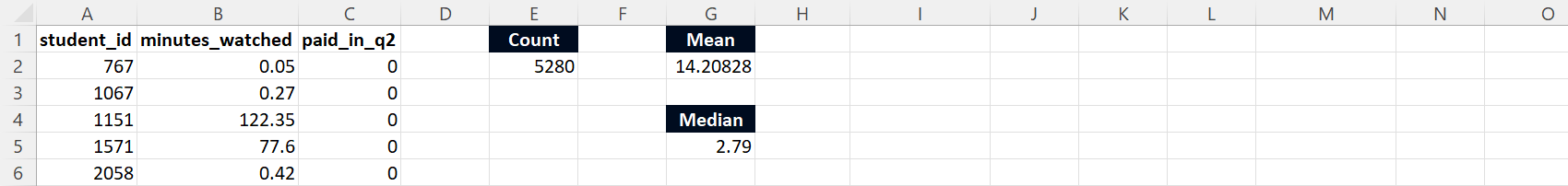


The results meet expectations because they align with the shapes of the data distribution. The right skewness observed in the distributions would lead us to expect a higher mean than median due to the influence of high-value outliers, which these statistics confirm. The difference in these metrics between free-plan and paying subscribers also makes sense because we expect paying students to generally watch more content than free-plan ones, leading to higher means and medians.

**II. Calculating Confidence Intervals**

1. **Determine the size of your sample (n):** Use Excel’s COUNT function to find the number of observations in your data.

=COUNT(B2:B5281)



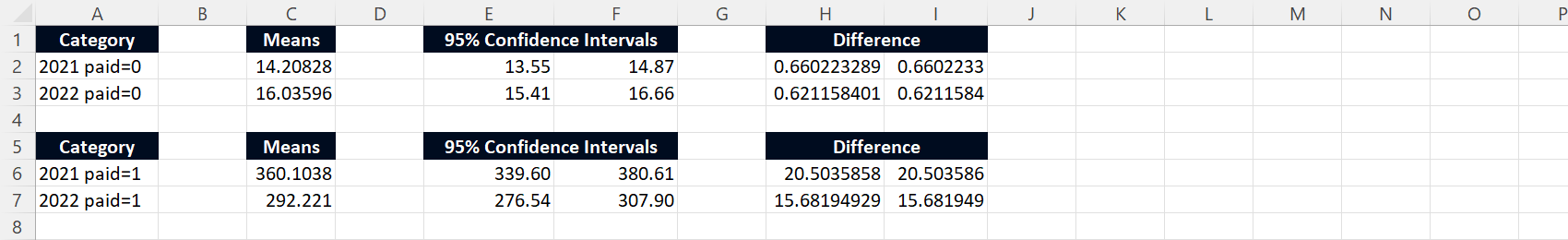
1. **Calculate the standard error:** Calculate this value by dividing the standard deviation (STDEV.S) by the square root (SQRT) of the sample size.

=STDEV.S(B2:B5281)/SQRT(E2)

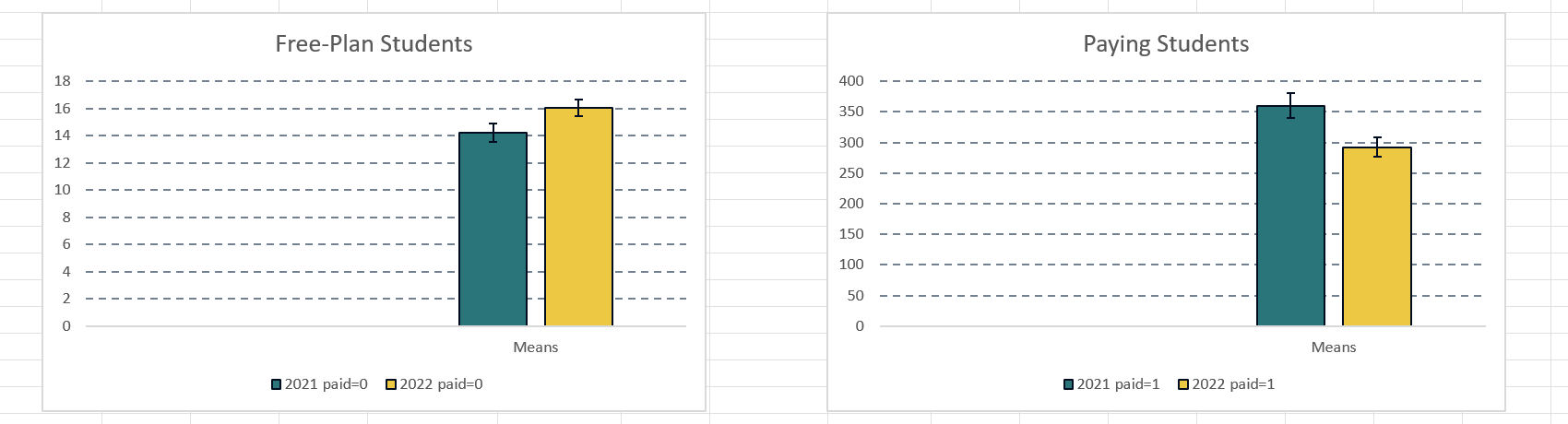
1. **Calculate the margin of error:** This is the critical value—usually a z-score for a normal distribution, 1.96 for a 95% confidence interval—multiplied by the standard error. You can directly multiply by 1.96, assuming a normal distribution.
2. **Calculate the confidence interval:** Subtract the margin of error from the mean to get the lower bound and add it to the mean to get the upper bound.

To create the confidence interval bar chart, execute the steps in the following **instructions**:

1. **Calculate the confidence intervals:** Calculate each group’s mean and confidence intervals per the previous guide. You should have three numbers for each group: the lower limit of the confidence interval, the mean, and the upper limit of the confidence interval.
2. **Prepare data table:** Create a new table with your data, including:
   1. The group names in the first column
   2. The mean values in the second column
   3. The confidence intervals in the third column
   4. The differences in the last column which are calculated as the difference between the mean and lower and the upper confidence limit. This will be used as an ‘error’ for the error bars.



1. **Create bar chart:** Select the data for the bar chart (the groups and mean values). Choose the Column or Bar chart option from the Insert tab, depending if you want a vertical or horizontal orientation.
2. **Add error bars:** Once the chart is created, click on one of the bars in the chart to select all bars. Then, go to the Chart Tools section, click Design, Add Chart Element, Error Bars, and More Error Bars Options.
3. **Customize error bars:** In the pop-up window, you need to customize the error bars. Choose Custom and then click Specify Value. Here you need to select the range of cells that contain the differences calculated in Step 2 (Prepare data table). Do this for both the Positive Error Value and the Negative Error Value.
4. **Format the chart:** Lastly, adjust the formatting of your chart as needed. You can add a title, adjust the axis labels, change colors, etc.



Following these steps should give you a bar chart with error bars representing the confidence intervals. Each bar represents the mean value for a group, and the error bars show the range within which we are 95% confident that the true mean value lies.

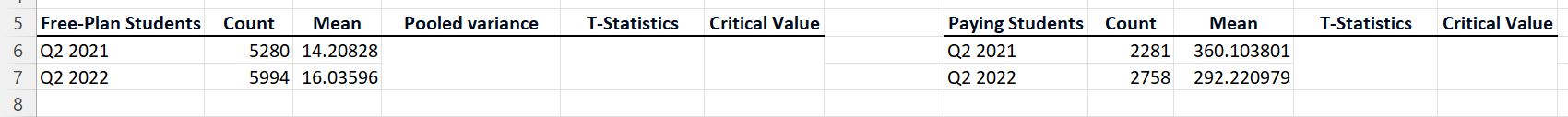
Comparing the four groups, you can observe the following:

* For free-plan students, there’s an increase in engagement from Q2 2021 to Q2 2022, as the confidence interval for the later period (15.41 – 16.66 minutes) is slightly higher than for the earlier one (13.55 – 14.87 minutes).
* Students with paid memberships watch substantially more than those without. This is evident by comparing the confidence intervals of the two groups in Q2 2021: 13.55 – 14.87 minutes for non-subscribers and 339.60 – 380.61 minutes for subscribers.
* Among the paid subscribers, there’s a decrease in engagement from Q2 2021 to Q2 2022, as the confidence interval for the later period (276.54 – 307.90 minutes) is lower than for the earlier one (339.60 – 380.61 minutes).

Please note that these are just interpretations based on the confidence intervals, and actual cause-effect relationships need further investigation. For instance, the fact that paid subscribers watch more doesn’t necessarily mean that having a paid subscription causes them to watch more. Those who watch more are more likely to get a paid subscription. Similarly, the decrease in engagement among paid subscribers from Q2 2021 to Q2 2022 could be due to various factors that need to be explored separately.

**III. Performing Hypothesis Testing**

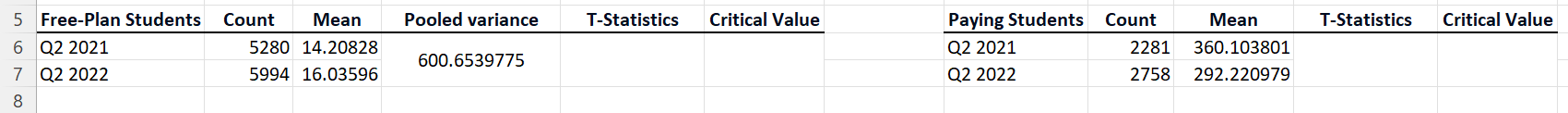
1. **Prepare your data:** Have one row for the metrics of students engaged in Q2 2021 and another for the metrics of students engaged in Q2 2022. Do this separately for free-plan and paying students.



1. **Calculate the pooled variance**σ2�2: Do this according to the formula

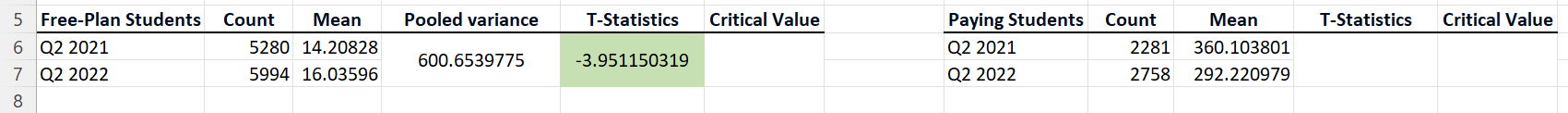
σ2=(n1−1)×σ21+(n2−1)×σ22(n1−1)+(n2−1)�2=(�1−1)×�12+(�2−1)×�22(�1−1)+(�2−1)

Here, n1�1 and n2�2 are the number of observations and σ21�12 and σ22�22 are the sample variances for Q2 2021 and Q2 2022, respectively.

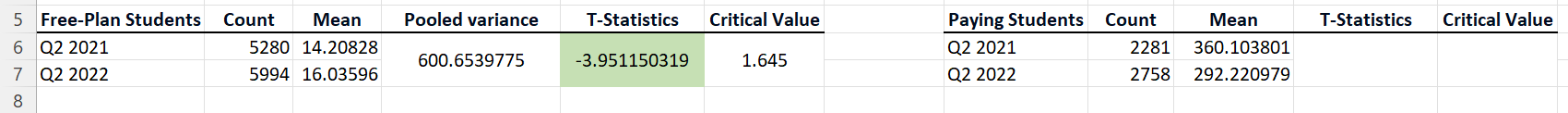


1. **Calculate the t-statistic:** The formula for the t-statistic in a two-sample t-test with equal variances is

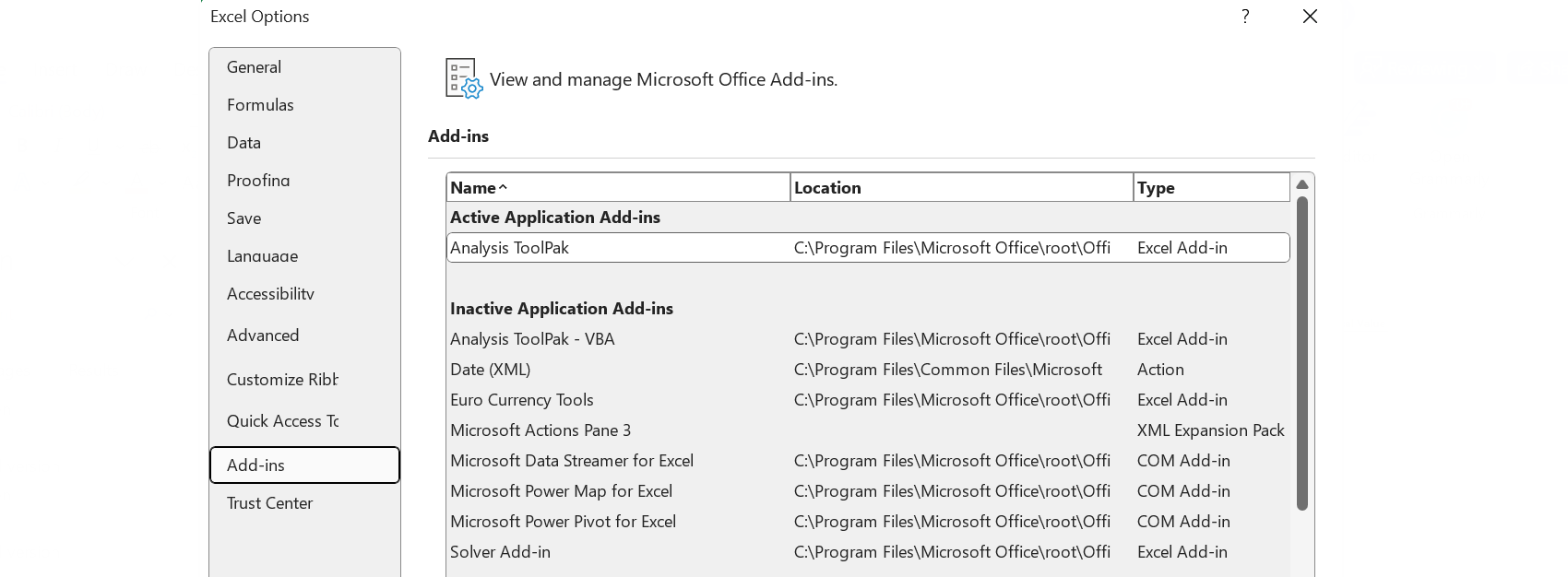
t=μ1−μ2σ√1n1+1n2�=�1−�2�1�1+1�2



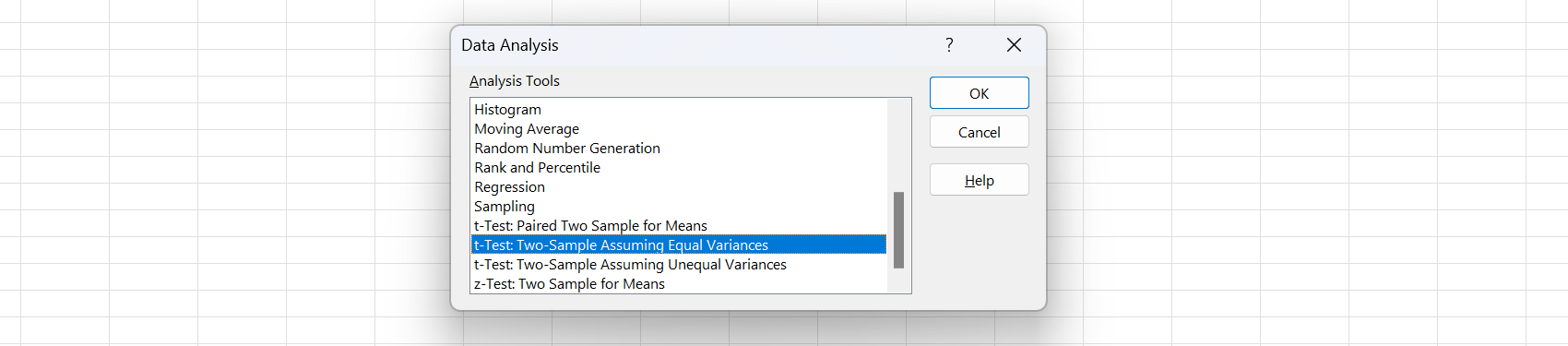
1. **Look up the critical t-value:** Use a t-distribution table (found online) to find the critical t-value (1.645) corresponding to your chosen significance level (commonly 0.05) and your calculated degrees of freedom.



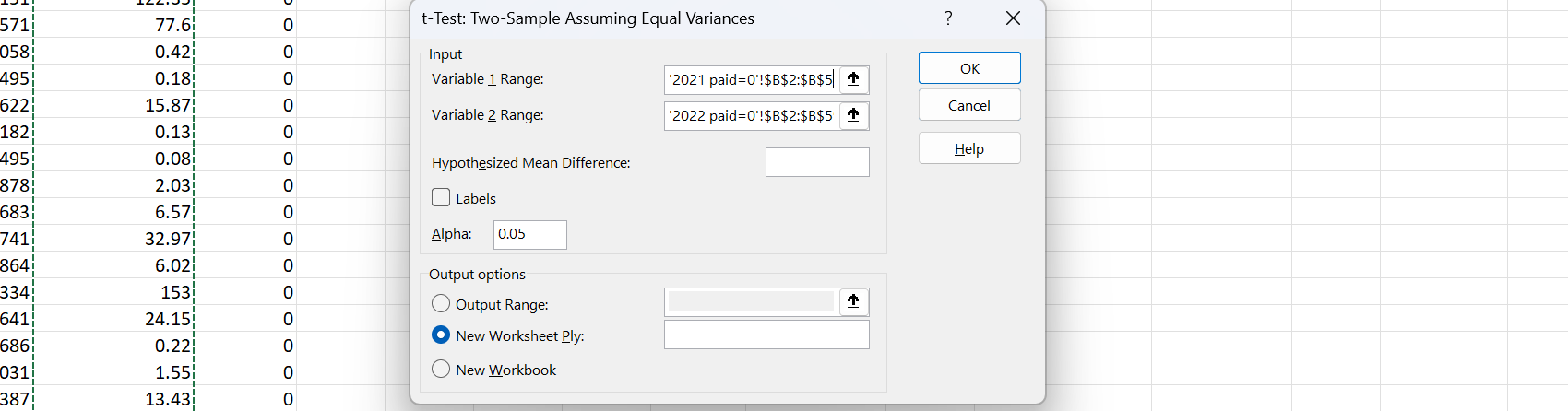
1. **Compare your results to Excel’s built-in t-test (optional):**You’re welcome to double-check your results using Excel’s built-in t-test. First, make sure you have the Analysis ToolPak installed.



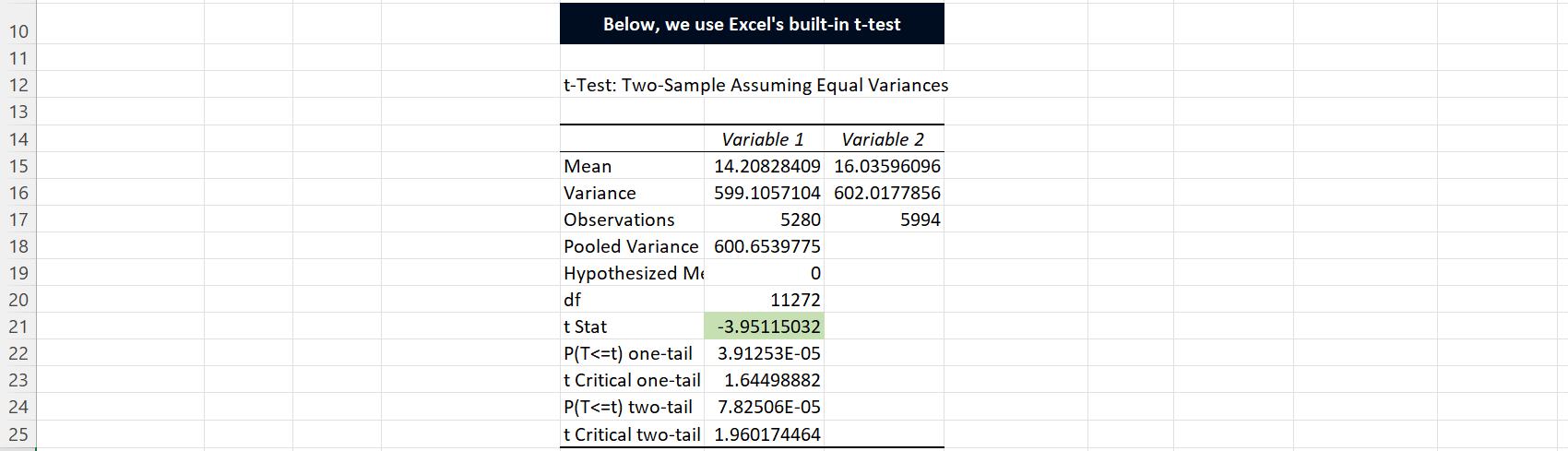
Then, navigate to Data in the Ribbon menu, select Data Analysis, and choose ‘t-Test: Two-Sample Assuming Equal Variances.’



Select the minutes\_watched column from the sheets storing the information on free-plan students.



Finally, select OK. What should appear is a table storing the t-statistic for free-plan students.



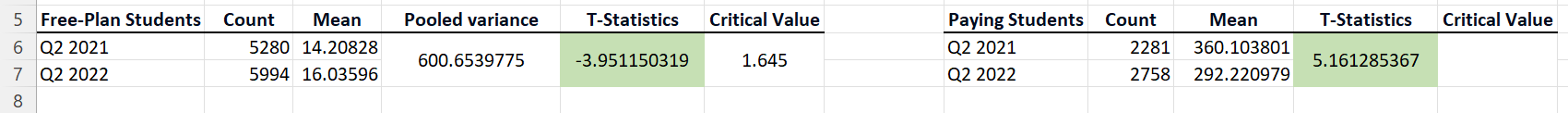
Compare the numbers from the table with the ones you have calculated. Don’t worry if the numbers differ slightly.

1. **Compare t-statistic to critical t-value:** With a t-statistic of -3.951 (less than the critical value of 1.645), you would reject the null hypothesis. This is because the negative t-statistic indicates that μ1�1 (the mean minutes watched by students in Q2 2021) is significantly smaller than μ2�2 (the mean minutes watched by students in Q2 2022), which is contrary to the null, so we reject it. Of course, rejecting the null hypothesis does not confirm the alternative hypothesis. It suggests that the data provide enough evidence against the null hypothesis.

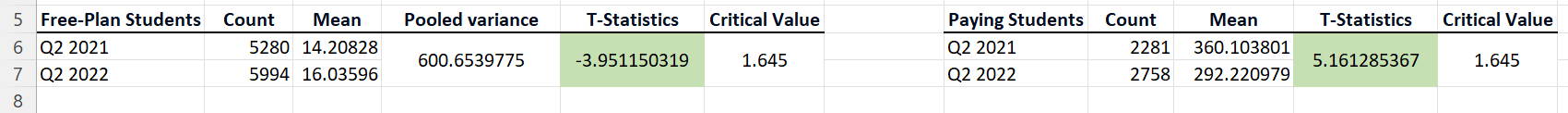
Note the **instructions** below to do it for paying students where the variances are assumed unequal.

1. **Calculate the t-statistic:** The formula for the t-statistic is

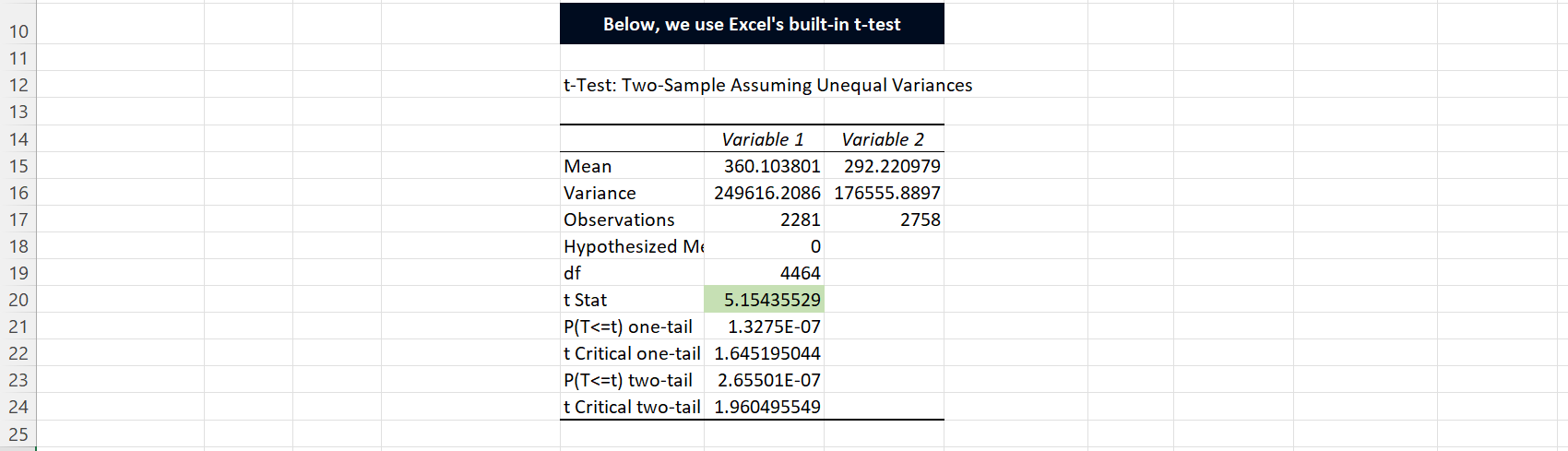
t=μ1−μ2σ√σ1n1+σ2n2�=�1−�2��1�1+�2�2



1. **Look up the critical t-value:** Use a t-distribution table (found online) to find the critical t-value (1.645) corresponding to your chosen significance level (commonly 0.05) and your calculated degrees of freedom.



1. **Compare your results to Excel’s built-in t-test (optional):**You’re welcome to double-check your results using Excel’s built-in t-test. Follow the same steps as the ones outlined for free-plan students. The only difference this time is choosing ‘t-Test: Two-Sample Assuming Unequal Variances’ and using the data for paying students. The table you obtain should look something like this:



1. **Compare t-statistic to critical t-value:** With a t-statistic of 5.161 (greater than the critical value of 1.645), you would fail to reject the null hypothesis. This means there’s not enough evidence to conclude that μ1�1 is smaller than μ2�2. So, the data supports the null hypothesis that μ1�1 is larger than or equal to μ2�2.

Regarding the second part of the question, a Type I error (false positive) occurs when you reject the null hypothesis, which is true. In our case, this would mean concluding that engagement in 2022 is higher when it’s not.

A Type II error (false negative) occurs when you fail to reject the null hypothesis, but it is false. In our case, this would mean concluding that the engagement in 2022 is not higher when it is.

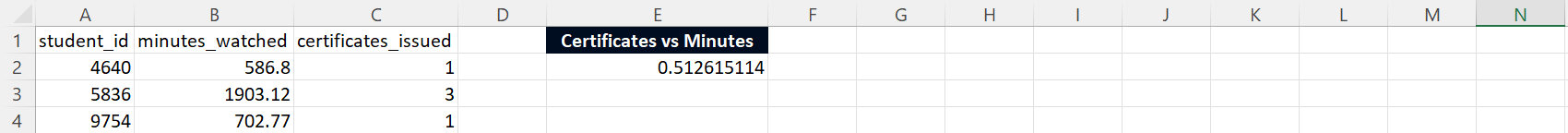
The cost to the company of each type of error would depend on the implications of incorrectly concluding that engagement has increased—potentially leading to over-investment in certain features or complacency about needing to improve features—versus incorrectly concluding that engagement has not increased—potentially missing out on recognizing successful features or identifying areas that need improvement.

Part5.zip

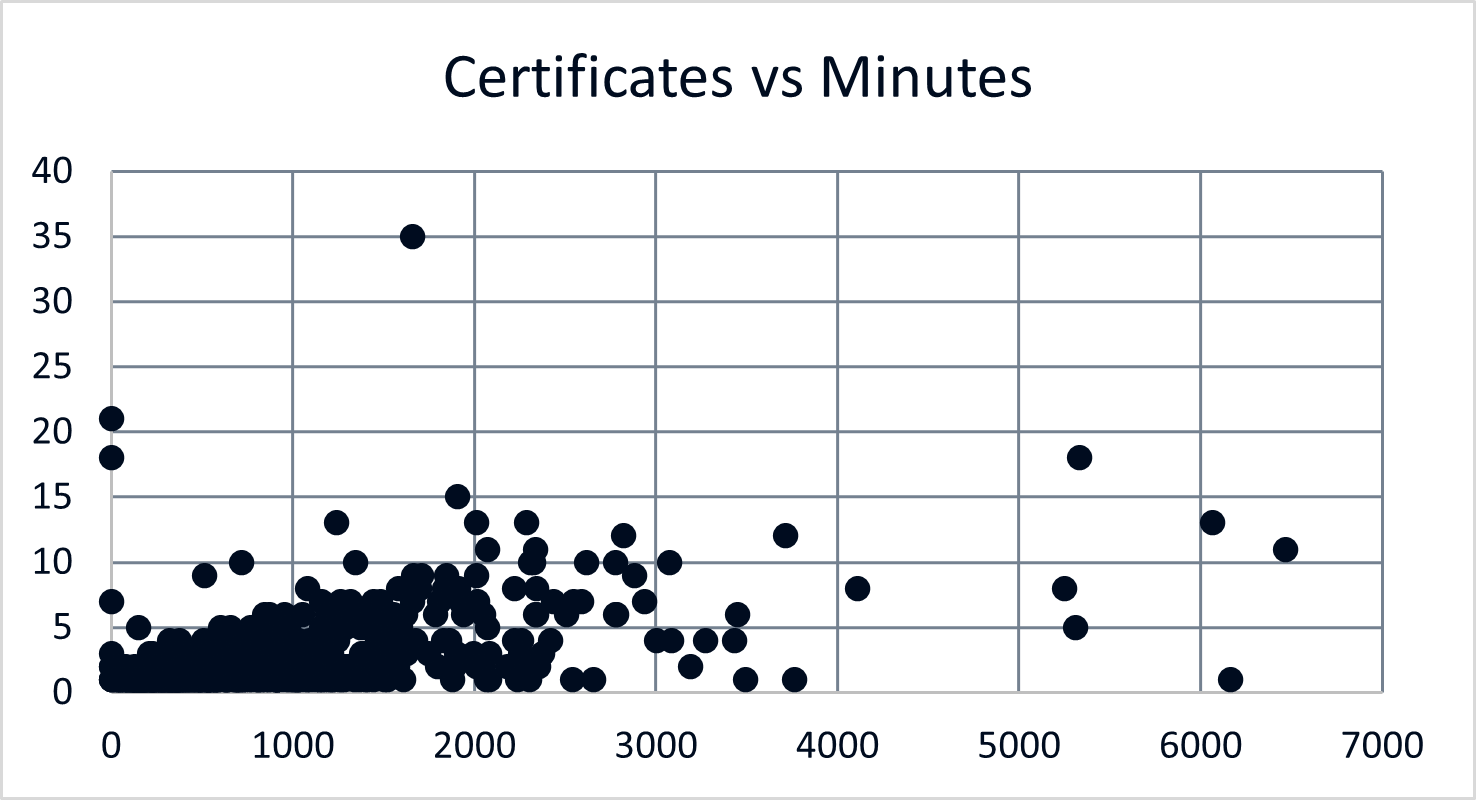
**Part 6:** Data Analysis with Excel – Correlation Coefficients

**I. Calculating Correlation Coefficients**

1. **Load the data into Excel:** First, you must have your CSV data in Excel.
2. **Calculate the correlation coefficient:**Excel has a built-in function called CORREL that calculates the correlation coefficient between two data sets. The general form is CORREL(array1, array2), where array1 and array2 are the two sets of data you want to correlate.



1. **Create a scatter plot (Optional):** You can make a scatter plot to interpret the results better.



1. **Interpret the correlation coefficient:**The correlation coefficients you’ve calculated measure the strength and direction of the linear relationship between two variables. They range from -1 to 1, where:
   * -1 indicates a perfect negative linear relationship.
   * 0 indicates no linear relationship.
   * 1 indicates a perfect positive linear relationship.

The correlation between Certificates and Minutes is approaximately 0.513—a strong positive correlation. It suggests that students who watch more content tend to earn more certificates.

Part6.zip

**Part 7:** Dependencies and Probabilities

**I. Assessing Event Dependencies**

First, we need to define the two events.

* Event A�: A student watched a lecture in Q2 2021.
* Event B�: A student watched a lecture in Q2 2022.

Two events are said to be independent if the occurrence of one does not affect the occurrence of the other. In probability terms, this is expressed as:

P(A∩B)=P(A)×P(B)�(�∩�)=�(�)×�(�)

Where:

* P(A∩B)�(�∩�) is the probability of both A� and B� occurring
* P(A)�(�) is the probability of A� occurring
* P(B)�(�) is the probability of B� occurring

Calculate each of these probabilities using the data and the following formulas:

P(A)=Number of students who watched a lecture in Q2 of 2021Total number of students who watched a lecture=7,63915,840P(B)=Number of students who watched a lecture in Q2 of 2022Total number of students who watched a lecture=8,84115,840P(A∩B)=Number of students who watched a lecture in both Q2 of 2021 and Q2 of 2022Total number of students who watched a lecture=64015,840�(�)=Number of students who watched a lecture in Q2 of 2021Total number of students who watched a lecture=7,63915,840�(�)=Number of students who watched a lecture in Q2 of 2022Total number of students who watched a lecture=8,84115,840�(�∩�)=Number of students who watched a lecture in both Q2 of 2021 and Q2 of 2022Total number of students who watched a lecture=64015,840

The queries you could use to obtain the numbers include the following:

* Calculating the number of students who watched a lecture in Q2 2021

SELECT

COUNT(DISTINCT student\_id)

FROM

student\_video\_watched

WHERE

YEAR(date\_watched) = 2021;

* Calculating the number of students who watched a lecture in Q2 2022

SELECT

COUNT(DISTINCT student\_id)

FROM

student\_video\_watched

WHERE

YEAR(date\_watched) = 2022;

* Calculating the number of students who watched a lecture in Q2 2021 and Q2 2022

SELECT

COUNT(DISTINCT student\_id)

FROM

(SELECT DISTINCT

student\_id

FROM

student\_video\_watched

WHERE

YEAR(date\_watched) = 2021) a JOIN (SELECT DISTINCT

student\_id

FROM

student\_video\_watched

WHERE

YEAR(date\_watched) = 2022) b using(student\_id);

* Calculating the total number of students who watched a lecture

SELECT

COUNT(DISTINCT student\_id)

FROM

student\_video\_watched;

Performing the calculation, we find that

P(A)×P(B)≈0.269�(�)×�(�)≈0.269

and

P(A∩B)≈0.0404�(�∩�)≈0.0404

Therefore,

P(A∩B)≠P(A)×P(B)�(�∩�)≠�(�)×�(�)

and the two events (A� and B�) are dependent, which means that the occurrence of one event has some influence on the occurrence of the other. Since P(A)×P(B)�(�)×�(�) is larger than P(A∩B)�(�∩�), it suggests that those who watched a lecture in Q2 2021 were less likely to watch a lecture in Q2 2022 than anticipated if the two events were independent. This is to be expected. A student who has benefitted from the program in 2021 and has completed their goal is not as likely to return in 2022 and study as much. This is what we refer to as ‘good churn.’

Nevertheless, we run marketing campaigns that ‘resurrect’ students who’ve been registered on the platform for a while but have not been active in a long time. The reason for that is to introduce such students to the new features of the platform as well as the new content in the course library. We always aim to upload new and relevant content and believe that ‘good churn’ students can still benefit the program even after some time.

**II. Calculating Probabilities**

First, we need to define the two events.

* Event A�: A student watched a lecture in Q2 2021.
* Event B�: A student watched a lecture in Q2 2022.

You’re asked to find the probability of A�, given that B� has occurred, i.e., P(A|B)�(�|�).

We can use Bayes’ Rule to solve the task:

P(A|B)=P(B|A)×P(A)P(B)�(�|�)=�(�|�)×�(�)�(�)

In the context of the problem:

* P(A)�(�) is the probability that a student watched a video in Q2 2021.
* P(B)�(�) is the probability that a student watched a video in Q2 2022.
* P(B|A)�(�|�) is the probability that a student watched a video in Q2 2022, given that they watched a video in Q2 2021.

We can calculate these probabilities from the data:

P(A)=Number of students who watched a lecture in Q2 of 2021Total number of students who watched a lecture=7,63915,840P(B)=Number of students who watched a lecture in Q2 of 2022Total number of students who watched a lecture=8,84115,840P(B|A)=Number of students who watched a lecture in both Q2 of 2021 and Q2 of 2022Total number of students who watched a lecture in Q2 of 2021=6407,639�(�)=Number of students who watched a lecture in Q2 of 2021Total number of students who watched a lecture=7,63915,840�(�)=Number of students who watched a lecture in Q2 of 2022Total number of students who watched a lecture=8,84115,840�(�|�)=Number of students who watched a lecture in both Q2 of 2021 and Q2 of 2022Total number of students who watched a lecture in Q2 of 2021=6407,639

Therefore,

P(A|B)=6407,639×7,63915,8408,84115,840=6408,841≈7%�(�|�)=6407,639×7,63915,8408,84115,840=6408,841≈7%

Among the students who watched a lecture in Q2 2022, 7% had also watched a lecture in Q2 2021. Note that this result reduces to the fraction:

Number of students who watched a lecture in both Q2 of 2021 and Q2 of 2022Number of students who watched a lecture in Q2 of 2022Number of students who watched a lecture in both Q2 of 2021 and Q2 of 2022Number of students who watched a lecture in Q2 of 2022

Confirming the result from the previous task, students who watched a lecture in Q2 2022 were unlikely to have also watched one in the same quarter of the previous year.

Part7.zip

**Part 8:** Data Prediction with Python

**I. Creating a Linear Regression**

1. **Import the relevant libraries:**
   * pandas library,
   * matplotlib library,
   * LinearRegression model from the sklearn.linear\_model module,
   * train\_test\_split method from the sklearn.model\_selection module,
   * seaborn library (optional).

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

import seaborn as sns

sns.set()

1. **Data import:** Use the pandas read\_csv() method to import your CSV file containing the dataset. This will return a DataFrame you can store in a variable (e.g., raw\_data).

raw\_data = pd.read\_csv('minutes\_and\_certificates.csv')

1. **Copy the data:** Create a copy to avoid accidentally modifying it; call the new variable data.

data = raw\_data.copy()

1. **Data preview:** You can use the head() method to preview your data quickly, including the column names and the first few rows.

data.head()

1. **Define the input and target variables:** Divide your dataset into features (input variable) and target variable.

target = data['certificates\_issued']

inputs = data['minutes\_watched']

1. **Split the data into training and testing sets:** It’s good practice to split your data into training and testing sets to avoid overfitting and understand how your model will perform on new data. As instructed in the task, allocate 20% of the data to a test set and use 365 as a random seed—ensuring the train-test split is the same each time.

x\_train, x\_test, y\_train, y\_test = train\_test\_split(inputs,

target,

test\_size=0.2,

random\_state=365)

1. **Reshape the data:** Since we create a model with a single feature, the training and test inputs need to be converted to NumPy ndarrays and then reshaped.

x\_train = x\_train.to\_numpy()

x\_test = x\_test.to\_numpy()

y\_train = y\_train.to\_numpy()

y\_test = y\_test.to\_numpy()

x\_train = x\_train.reshape(-1, 1)

x\_test = x\_test.reshape(-1, 1)

1. **Create and train the model:** Create an instance of sklearn’s LinearRegression model and then fit it on the training data.

reg = LinearRegression()

reg.fit(x\_train,y\_train)

1. **Linear equation:** After you fit the regression model to the training data, print the value of the slope (denoted by m�) and the y�-intercept (indicated by b�). The linear equation of the model can then be expressed as

y=mx+b�=��+�

reg.intercept\_, reg.coef\_

The linear equation that explains the behavior of the relationship includes the following:

y=1.056+0.001740×x�=1.056+0.001740×�

1. **Calculating the R-squared:**Print the value of the R-squared metric using the score method.

reg.score(x\_train, y\_train)

The value we obtained is approximately 0.305. This suggests that about 30% of the variability in the target variable (the number of certificates issued) is explained by the input variable (the number of minutes watched). This model does not account for the other 70%.

An R-squared value of 0.305 is not a bad result, but, as we mentioned, it implies that other factors also play a role in the number of certificates issued. Let’s list them one by one.

One such factor, for example, includes different courses with different lengths. Therefore, a student passing three short courses will be issued three certificates, while a student passing one long course—roughly the length of three short ones—will be given only one certificate. Another factor could be that some students pass exams without watching the courses. The reason could be that they are familiar with the subject and only aim for a document proving their proficiency.

The model, therefore, provides some insight into the relationship between these two quantities, but there’s still a large portion of the variance that remains unexplained. The number of minutes watched is reasonable to include when predicting the number of certificates issued but should not be the sole factor considered.

1. **Prediction:** One way to predict the number of certificates issued is to plug the number of minutes watched (namely 1200) in the equation. This yields

y=1.056+0.001740 ×1200 ≈3�=1.056+0.001740 ×1200 ≈3

You can also make predictions using the predict() method.

y\_hat\_test = reg.predict(x\_test)

reg.predict([[1200]])

1. **Visualization:** To visualize the model performance, you can create a scatter plot where the x�-axis represents the actual test values, and the y�-axis is the predicted values.

plt.scatter(y\_test, y\_hat\_test)

plt.xlabel('Targets (y\_test)',size=18)

plt.ylabel('Predictions (y\_hat\_test)',size=18)

plt.show()



Part8.zip