

Topic	Correlation	
Class Description	Students gets introduced to the concept of "correlation" and how to identify if two data sets are correlated visually by drawing plots. Student also learns how to calculate correlation and writes a python program to calculate correlation.	
Class	C106	
Class time	45 mins	
Goal	<ul style="list-style-type: none"> <li>Understand the concept of correlation</li> <li>Identify if two data sets are correlated using visual charts</li> <li>Calculate correlation and write a python program for it</li> </ul>	
Resources Required	<ul style="list-style-type: none"> <li>Teacher Resources               <ul style="list-style-type: none"> <li>Visual Code Studio</li> <li>Laptop with internet connectivity</li> <li>Earphones with mic</li> <li>Notebook and pen</li> </ul> </li> <li>Student Resources               <ul style="list-style-type: none"> <li>Visual Code Studio</li> <li>Laptop with internet connectivity</li> <li>Earphones with mic</li> <li>Notebook and pen</li> </ul> </li> </ul>	
Class structure	Warm Up Teacher-led Activity Student-led Activity Wrap up	5 mins 15 min 15 min 5 min
<div>CONTEXT</div> <ul style="list-style-type: none"> <li>Review central tendency and standard deviation</li> </ul>		
Class Steps	Teacher Action	Student Action

<b>Step 1: Warm Up (5 mins)</b>	<p>Hello! We've been learning statistics and python programming since the last few classes.</p> <p>Do you remember what we have covered in the last few classes?</p>	<p>ESR:</p> <p>We learned how to calculate the central tendency of data - mean, median and mode. We also learned how to calculate standard deviation in data.</p>
	<p>Can you describe in your own words - what is the central tendency of data?</p>	<p>Central Tendency of data is a value which tries to describe the central position (where most of data is centred) of data.</p> <p>There are different ways in which central tendency of data can be calculated. Mean, median and mode are different methods through which we can calculate the central tendency of data.</p>
	<p>Awesome. How would you describe "standard deviation"?</p>	<p>Standard deviation is a measure of how much the members of a data set differ from the mean</p>
	<p>Great! So far, we have been restricted to one data set so far.</p> <p>In this class, we will explore more than one data set and learn to identify if the two data sets are related to each other.</p>	-
<b>Teacher Initiates Screen Share</b>		
<p style="text-align: center;"><b><u>CHALLENGE</u></b></p> <ul style="list-style-type: none"> <li>Look at data containing record of temperature for each day and sales of ice cream and cold-drinks from a public store</li> </ul>		

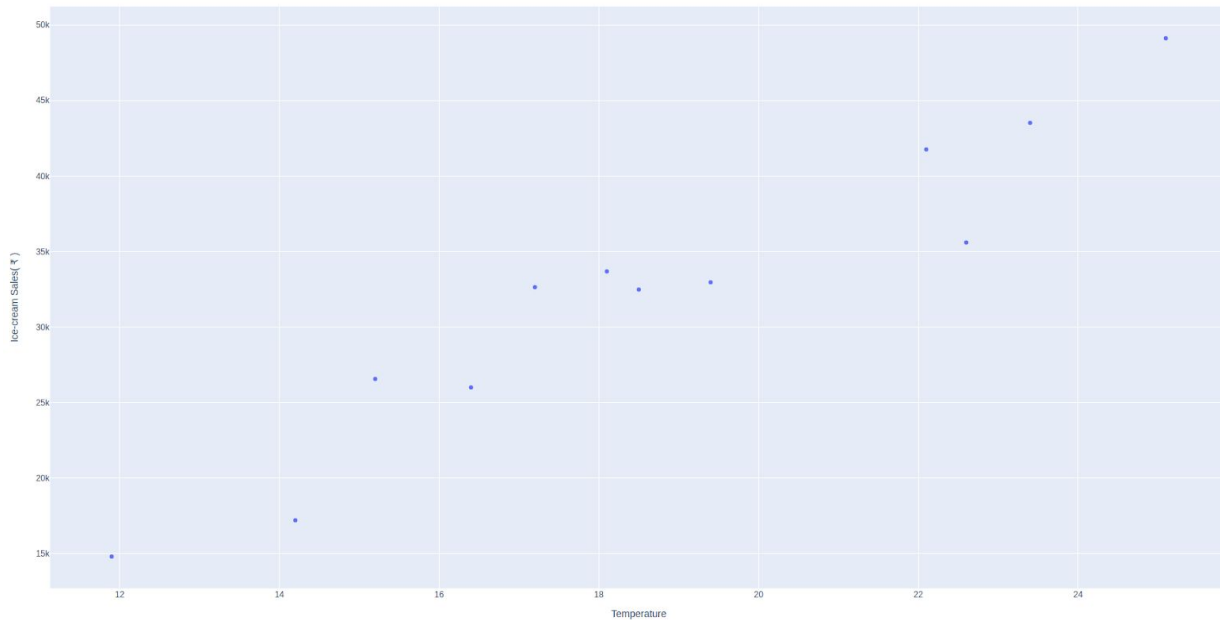
- Visually identify if the data sets are correlated
- Calculate the correlation index on a spreadsheet

<b>Step 2: Teacher-led Activity (15 min)</b>	<p>Have you ever wondered if two different data sets can have some relation to each other?</p> <p>For example - can the number of car accidents in a day and temperature of a particular day be related?</p>	<p>ESR: They might have some relation. People might get more frustrated when the temperature is high. It might lead to more road rage and more accidents!</p>
	<p>What about the temperature of a day and sale of ice-cream in a store?</p>	<p>ESR: High temperature might lead to high sales of ice-cream in the store.</p>
	<p>Let's check one such data. We have some data where temperature of each day is recorded. It also contains data of sales of ice-cream in a public store.</p> <p>Let's plot this data as a scatter plot and see how it looks.</p> <p>Can you help me make the scatter plot? What would be on the X-axis? What would be on the Y-axis?</p>	<p>ESR: X- axis can be temperature Y- axis can be the sales from ice-cream</p>
	<p>Teacher writes code to create a scatter plot for Temperature vs Sales from ice-cream</p> <p>Teacher runs the code to show the data visualization.</p>	<p>Student guides the teacher in writing the code for creating the scatter plot for Temperature vs Sales from ice-cream</p>

```

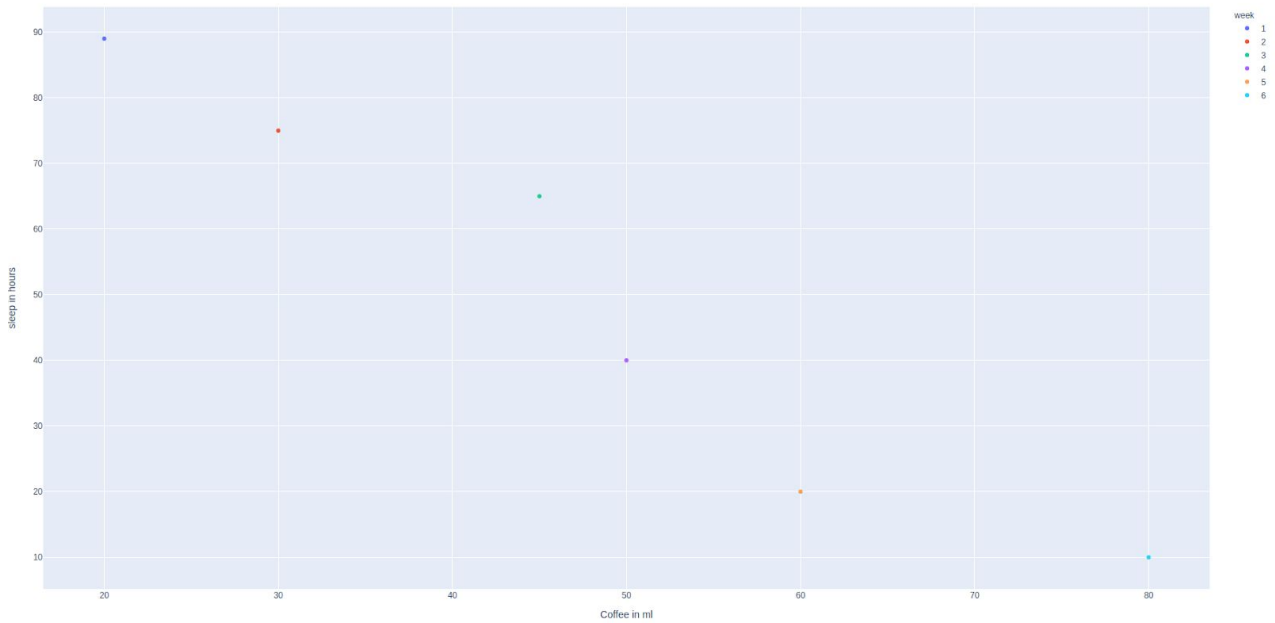
1 import plotly.express as px
2 import csv
3
4 with open("../data/Ice-Cream vs Cold-Drink vs Temperature - Ice Cream Sale vs Temperature data.csv") as csv_file:
5     df = csv.DictReader(csv_file)
6     fig = px.scatter(df, x="Temperature", y="Ice-cream Sales( ₹ )")
7     fig.show()
8

```



	<p>What do you see?</p> <p>What happens when the temperature increases?</p> <p>What happens when the temperature decreases?</p>	<p>ESR:</p> <p>When the temperature increases, the sale of ice-cream goes up.</p> <p>When the temperature reduces, the sale of ice-cream goes down.</p>
	<p>You see the data are not scattered on the graph and are close towards a central line .</p> <p>Such data are set to be highly correlated.</p> <p>Data sets can also be inversely</p>	<p>ESR:</p> <p>One data set increases when the other data set reduces?</p> <p>Temperature data vs sale of warm clothes in a store</p>

	<p>correlated. What do you think that means?</p> <p>Can you think of data sets which might be inversely correlated?</p>	
	<p>Yes! How do you think the scatter plot for such a data set would look like?</p>	<p>ESR: varied</p>
	<p>We have a data set for consumption of coffee vs Hours of sleep. How do you think they might be correlated?</p>	<p>ESR: When cups of coffee decreases hours of sleep increase. or the two data sets are inversely correlated.</p>
	<p>Let's draw a scatter plot and see. Can you help me draw a scatter plot visualization for the data.</p> <p>Teacher writes code to draw the scatter plot.</p>	<p>Student helps the teacher write the code.</p>
<pre>import plotly.express as px import csv  with open("../data/cups of coffee vs hours of sleep.csv") as csv_file:     df = csv.DictReader(csv_file)     fig = px.scatter(df,x="Coffee", y="sleep")     fig.show()</pre>		
	<p>Let's run the code to check the output. What do you see?</p>	<p>ESR: We see a falling graph where the data is still close to the central straight line.</p>



Yes! This is how an inverse correlation looks like.

We can also calculate correlation value.

A correlation of 1 means the two data sets are closely correlated. This will be a rising graph where the data points are close to a central line.

A correlation of -1 means that the two data sets are inversely correlated.

This will be a falling graph where the data points are close to a central line.

A correlation of 0 means that the two data sets are not correlated at all! The data points will be scattered on the graph.

Correlation always lies between -1 and 1

Student ask questions to understand correlation.

	<p>Let's look at how to calculate correlation coefficient using the <code>corrcoef()</code> function in numpy library. WE will calculate the correlation coefficient of temperature and ice-cream sales data</p> <ol style="list-style-type: none"> <li>1. Teacher imports numpy library in code. (Make sure numpy is pre-installed using <code>pip3 install numpy</code> earlier)</li> <li>2. Convert temperature data and ice-cream sales data into arrays. Make sure that each data set is converted into a float value first. (by default each data set is a string)</li> <li>3. Use <code>corrcoef()</code> function and pass the two datasets to it. Store the output in a variable.</li> <li>4. Print the correlation coefficient on the screen.</li> </ol> <p>What is the result?</p>	<p>ESR: 0.95</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------

```
import plotly.express as px
import csv
import numpy as np

def getDataSource(data_path):
    ice_cream_sales = []
    cold_drink_sales = []
    with open(data_path) as csv_file:
        csv_reader = csv.DictReader(csv_file)
        for row in csv_reader:
            ice_cream_sales.append(float(row["Temperature"]))
            cold_drink_sales.append(float(row["Ice-cream Sales( ₹ )"]))

    return {"x" : ice_cream_sales, "y": cold_drink_sales}

def findCorrelation(datasource):
    correlation = np.corrcoef(datasource["x"], datasource["y"])
    print("Correlation between Temperature vs Ice Cream Sales :- \n--->",correlation[0,1])

def setup():
    data_path = "./data/Ice-Cream vs Cold-Drink vs Temperature - Ice Cream Sale vs Temperature data.csv"

    datasource = getDataSource(data_path)
    findCorrelation(datasource)

    |

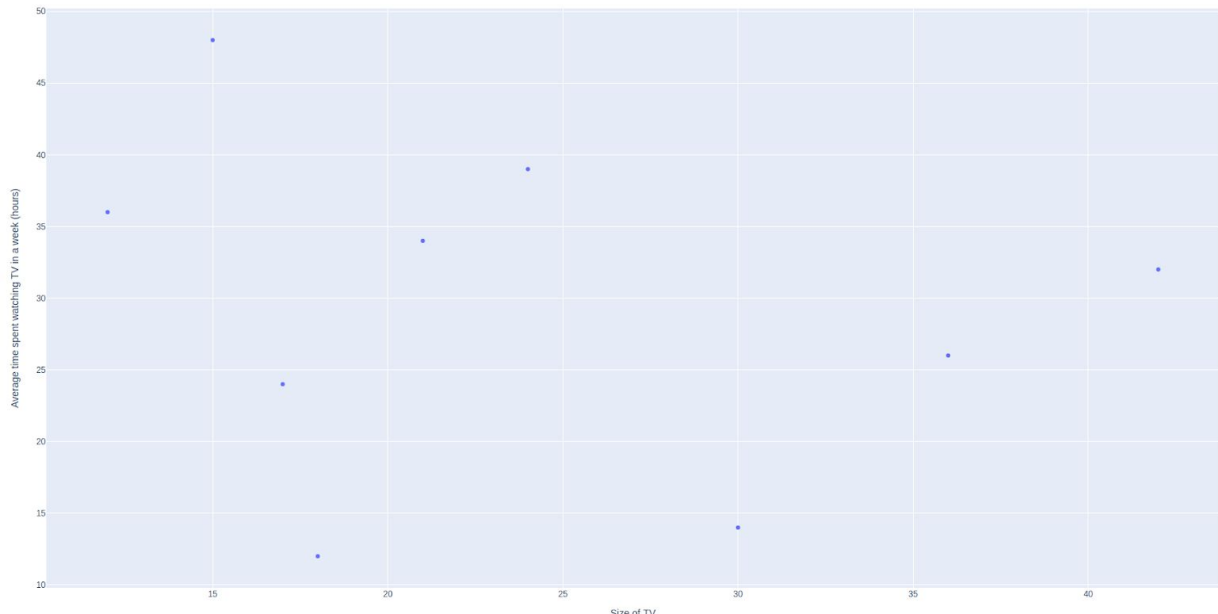
setup()
```

```
$ python3 setup.py
Correlation between Temperature vs Ice Cream Sales :-
---> 0.9575066230015955
```

	What does this tell?	ESR: The two data sets are positively correlated. They also have a high correlation.
	<p>Alright. You now know how to create a visual representation of two data sets and identify if the two data sets are correlated or not.</p> <p>Now, I am going to give you two different sets.</p> <p>1. One data set compares the number of hours of TV watched in a week on</p>	Yes



	<p>average vs the size of television.</p> <p>2. Another data set is of the number of days each student has been present in college in a year vs the percentage of marks scored in the half-yearly exams</p> <p>Can you plot the data for each and visually guess if they are correlated. You can then also use the in-built correlation coefficient function to calculate the correlation.</p>	
<b>Teacher Stops Screen Share</b>		
	Now it's your turn. Please share your screen with me.	
<ul style="list-style-type: none"> <li>• <b>Ask Student to press ESC key to come back to panel</b></li> <li>• <b>Guide Student to start Screen Share</b></li> <li>• <b>Teacher gets into Fullscreen</b></li> </ul>		
<p style="text-align: center;"><b><u>ACTIVITY</u></b></p> <ul style="list-style-type: none"> <li>• <b>Student writes python program to calculate correlation index</b></li> </ul>		
<b>Step 3: Student-Led Activity (15 min)</b>	<p>What do you think would be the relationship between number of hours of TV watched in a week on average vs the size of television?</p> <p>How would their correlation be?</p>	<p>ESR:</p> <p>People should watch more TV in a week as the size of television goes up.</p>

	<p>Let's visually plot the data and check.</p>	<p>Student downloads the data.</p> <p>Student writes code to :</p> <ul style="list-style-type: none"> <li>- read the data</li> <li>- use plotly to draw a scatterplot for the data</li> </ul>
<pre>import plotly.express as px import csv  with open("./data/Size of TV, Average time spent watching TV in a week (hours).csv") as csv_file:     df = csv.DictReader(csv_file)     fig = px.scatter(df,x="Size of TV", y="\tAverage time spent watching TV in a week (hours)")     fig.show()</pre> 		
	<p>Look at the scatter plot. Are the points close to any central line.</p> <p>How do you think is the correlation between the number of hours of TV watched vs size of the television</p>	<p>ESR:</p> <p>No! They are scattered.</p> <p>They both are not correlated at all</p>

	<p>Let's calculate the correlation index and check its value.</p>	<p>Student writes code to calculate the correlation index and finds it is close to 0</p>
<pre> 1 import csv 2 import numpy as np 3 4 5 def getDataSource(data_path): 6     size_of_tv = [] 7     Average_time_spent = [] 8     with open(data_path) as csv_file: 9         csv_reader = csv.DictReader(csv_file) 10        for row in csv_reader: 11            size_of_tv.append(float(row["Size of TV"])) 12            Average_time_spent.append(float(row["Average time spent watching TV in a week (hours)"])) 13 14        return {"x" : size_of_tv, "y": Average_time_spent} 15 16 def findCorrelation(datasource): 17     correlation = np.corrcoef(datasource["x"], datasource["y"]) 18     print("Correlation between Size of Tv and Average time spent watching Tv in a week :- \n--&gt;",correlation[0,1]) 19 20 def setup(): 21     data_path = "./data/Size of TV, Average time spent watching TV in a week (hours).csv" 22 23     datasource = getDataSource(data_path) 24     findCorrelation(datasource) 25 26 setup() 27 </pre> <pre> Correlation between Size of Tv and Average time spent watching Tv in a week :- --&gt; -0.21596489617950243 </pre>		
	<p>Awesome. Now let's take a look at another dataset.</p> <p>We have the number of days students attended college vs the marks they scored in their exams. How do you think these two would be correlated?</p>	<p>ESR: varied</p>

Let's plot them visually and check if your guess is right

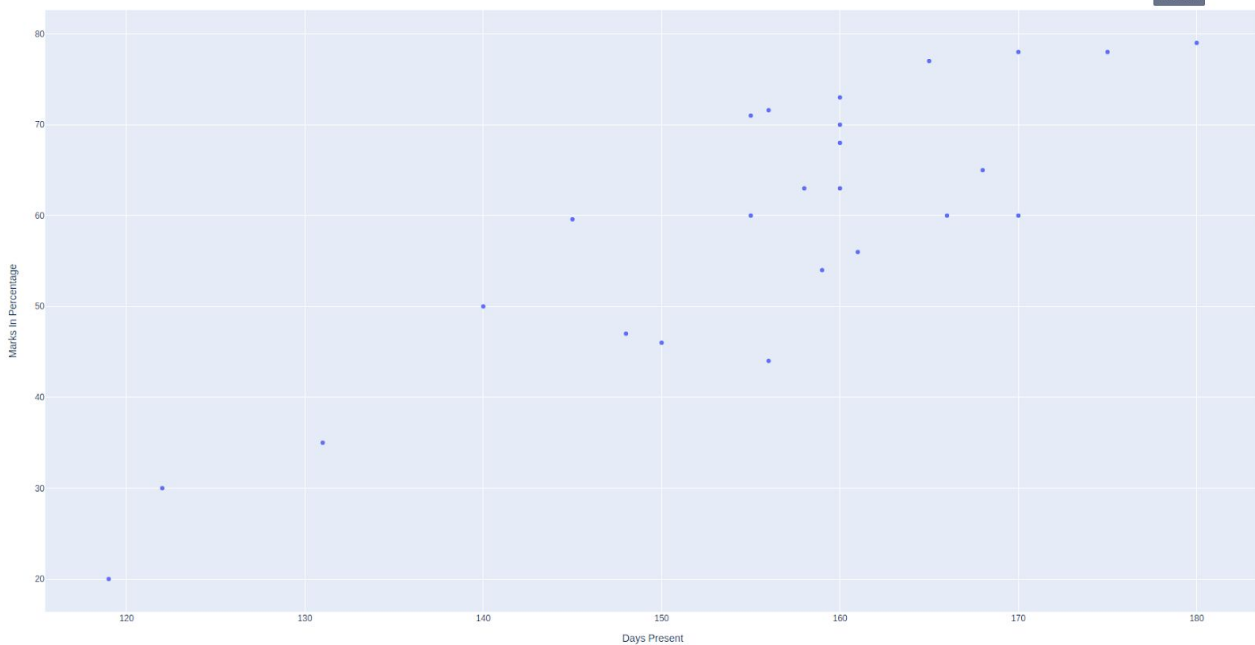
Student downloads the data.

Student writes code to :

- read the data
- use plotly to draw a scatterplot for the data

```
import plotly.express as px
import csv

with open("../data/Student Marks vs Days Present.csv") as csv_file:
    df = csv.DictReader(csv_file)
    fig = px.scatter(df, x="Days Present", y="Marks In Percentage")
    fig.show()
```



	What do you think is the correlation between the number of days one attends the classes vs the percentage of marks scored in the exams?	ESR: The two data are positively correlated.
	Let's calculate the correlation index to verify	Student writes code to calculate the correlation index and finds it is close to 1

```

1  import csv
2  import numpy as np
3
4
5  def getDataSource(data_path):
6      marks_in_percentage = []
7      days_present = []
8      with open(data_path) as csv_file:
9          csv_reader = csv.DictReader(csv_file)
10         for row in csv_reader:
11             marks_in_percentage.append(float(row["Marks In Percentage"]))
12             days_present.append(float(row["Days Present"]))
13
14         return {"x" : marks_in_percentage, "y": days_present}
15
16 def findCorrelation(datasource):
17     correlation = np.corrcoef(datasource["x"], datasource["y"])
18     print("Correlation between Marks in percentage and Days present :- \n-->", correlation[0,1])
19
20 def setup():
21     data_path = "./data/Student Marks vs Days Present.csv"
22
23     datasource = getDataSource(data_path)
24     findCorrelation(datasource)
25
26 setup()
27

```

```

Correlation between Marks in percentage and Days present :-
--> 0.86288947614385

```

	Awesome.	-
<b>Teacher Guides Student to Stop Screen Share</b>		
<p style="text-align: center;"><b><u>FEEDBACK</u></b></p> <ul style="list-style-type: none"> <li>• Appreciate the student for their efforts</li> <li>• Identify 2 strengths and 1 area of progress for the student</li> </ul>		
<b>Step 4: Wrap-Up (5 min)</b>	Can you capture what we learned in today's class?	ESR: We learned about correlation. How to plot data and identify if the data sets have some relation among themselves. We also learned about correlation coefficient and how to calculate it using python program.
	<p>Correlation is a very important in data science. For example lot of data scientists spend number of hours trying to identify data sets to which stock prices might be correlated.</p> <p>You can collect datasets and try to identify if any two datasets are correlated.</p> <p>Correlaton is also an important concept in machine learning models.</p>	-

	<p>It can help us predict future data based on the previously collected data.</p> <p>We'll be learning more about it in the coming classes.</p>	
<div> <b>Teacher Clicks</b> <span>✕ End Class</span> </div>		
<b>Additional Activities</b>	<p>Encourage the student to write reflection notes in their reflection journal using markdown.</p> <p>Use these as guiding questions:</p> <ul style="list-style-type: none"> <li>• What happened today?               <ul style="list-style-type: none"> <li>- Describe what happened</li> <li>- Code I wrote</li> </ul> </li> <li>• How did I feel after the class?</li> <li>• What have I learned about programming and developing games?</li> <li>• What aspects of the class helped me? What did I find difficult?</li> </ul>	<p>The student uses the markdown editor to write her/his reflection in a reflection journal.</p>

Activity	Activity Name	Links
Teacher Activity 1	Solution	<a href="https://github.com/whitehatjr/correlation">https://github.com/whitehatjr/correlation</a>
Student Activity 1	Ice-Cream vs Cold-Drink vs Temperature - Ice Cream Sale vs Temperature data	<a href="https://raw.githubusercontent.com/whitehatjr/correlation/master/data/Ice-Cream%20vs%20Cold-Drink%20vs%20Temperature%20-%20Ice%20C">https://raw.githubusercontent.com/whitehatjr/correlation/master/data/Ice-Cream%20vs%20Cold-Drink%20vs%20Temperature%20-%20Ice%20C</a>

		<a href="#">ream%20Sale%20vs%20Temperatu re%20data.csv</a>
Student Activity 2	Size of TV, Average time spent watching TV in a week (hours)	<a href="https://raw.githubusercontent.com/whitehatjr/correlation/master/data/Size%20of%20TV%2C%09Average%20time%20spent%20watching%20TV%20in%20a%20week%20(hours).csv">https://raw.githubusercontent.com/whitehatjr/correlation/master/data/Size%20of%20TV%2C%09Average%20time%20spent%20watching%20TV%20in%20a%20week%20(hours).csv</a>
Student Activity 3	Student Marks vs Days Present	<a href="https://raw.githubusercontent.com/whitehatjr/correlation/master/data/Student%20Marks%20vs%20Days%20Present.csv">https://raw.githubusercontent.com/whitehatjr/correlation/master/data/Student%20Marks%20vs%20Days%20Present.csv</a>
Student Activity 4	cups of coffee vs hours of sleep	<a href="https://raw.githubusercontent.com/whitehatjr/correlation/master/data/cups%20of%20coffee%20vs%20hours%20of%20sleep.csv">https://raw.githubusercontent.com/whitehatjr/correlation/master/data/cups%20of%20coffee%20vs%20hours%20of%20sleep.csv</a>