

Name: Han Hong Tuck from EP0302_01

Title of Data Analysis: How do Global Events influence Singapore's Economic Growth?

Questions to answer to gain deeper insights into dataset:

Question 1: Does the Scatter plot further reaffirms the relationship between GDP Growth, retrenchments and short work week and layoffs?

Question 2: Coming to the end of the Data Analysis, what are some of the key takeaways after analysing all these datasets?

Url of Annual Contribution to Growth in Gross Domestic Product in Chained 2015 dollars Dataset: <https://data.gov.sg/dataset/contribution-to-growth-in-gdp-in-chained-2015-dollars-by-industry-ssic-2015-annual> (<https://data.gov.sg/dataset/contribution-to-growth-in-gdp-in-chained-2015-dollars-by-industry-ssic-2015-annual>)

Url of Annual Retrenched Employees (Topline): <https://data.gov.sg/dataset/retrenched-employees-by-industry-and-occupational-group-annual> (<https://data.gov.sg/dataset/retrenched-employees-by-industry-and-occupational-group-annual>)

Url of Annual Short Work-Week & Temporary Layoffs (Topline): <https://data.gov.sg/dataset/short-work-week-temporary-layoffs-annual> (<https://data.gov.sg/dataset/short-work-week-temporary-layoffs-annual>)

Breaking Down the Nature of Retrenchments in Jobs (Topline) Dataset:

The nature of the dataset for retrenched employees contains the retrenchment of permanent jobs, retrenchments of Term Contract (Temporary) Jobs as well as total retrenchments from the year 1998 to 2019 due to redundancy, sorted by year.

Defining Terminologies:

1. Retrenchment refers to the termination of permanent employees due to redundancy and early termination of term contract employees due to redundancy. Permanent Jobs typically do not have a predetermined end date to employment.
2. Retrenchment of Term Contract Employees refers to early termination of term contract employees due to redundancy. Term Contract (Temporary) Jobs refers Jobs that are fixed-term contracts which can be terminated by employers upon expiry of a specific term or period (such as a date), unless it is renewed.
3. Total Retrenchments refers to Retrenchments inclusive of Permanent Jobs as well as Term Contract (Temporary) Jobs.

Redundancy refers to a situation whereby an employer reduces their workforce in the event that a certain jobs are no longer needed. Such situations may arise due to exterior factors due to the global economic environment which has resulted in businesses closing down, the employer needing to cut expenses, the advent of artificial technology (AI) or other technologies that have made that job unnecessary.

Purpose of Plotting Nature of Retrenchments in Jobs (Topline) Dataset

Since the Nature of Retrenchment in Jobs (By Industry) has been already plotted, the plotting of the Nature of Retrenchments in Jobs (Topline) is to further display the correlation between GDP Growth and Total Retrenchments over the years. By doing so, it will further prove that the hypothesis, analysis and insights gained from analysing the dataset is well-substantiated and justified.

Breaking Down the Nature of Jobs affected by Short Work Week and Temporary Layoff (Topline) Dataset:

The nature of the dataset for short work week and temporary layoff contains the number of jobs that implemented short work week, the number of jobs that have been temporary laid off as well as the total number of jobs from the year 1998 to 2019, sorted by year.

Defining Terminologies:

1. Short Work-Week are employees whose normal number of working days per week has been temporarily reduced due to lack of work, at any time during the reference period.
2. Temporary Layoffs are employees whose services are suspended temporarily due to lack of work, at any time during the reference period. They may or may not have been paid during this period.

Purpose of Nature of Jobs affected by Short Work Week and Temporary Layoff (Topline) Dataset:

Since the Nature of Jobs affected by Short Work Week and Temporary Layoff (By Industry) has been already plotted, the plotting of the Nature of Jobs affected by Short Work Week and Temporary Layoff (Topline) is to further display the correlation between GDP Growth and Total Short Work Week and Layoffs over the years. By doing so, it will further prove that the hypothesis, analysis and insights gained from analysing the dataset is well-substantiated and justified.

Write Python code that uses the pandas package to extract useful statistical or summary information about the data

```

In [2]: import pandas as pd
import numpy as np

#read from file
retrenchments = pd.read_csv("./Datasets/retrenchment-topline.csv",na_values=["-"])
short_work_week_and_temporary_layoffs = pd.read_csv("./Datasets/short-work-week-and-temporary-layoffs.csv",na_values=["-"])

#Merging retrenchments dataframe with short work week and temporary Layoff dataframe
df_all = pd.merge(retrenchments, short_work_week_and_temporary_layoffs, on=['year'])

#Shape of Retrenchments, Short Work Week and Temporary Layoffs Dataset
print(f"Shape of Retrenchments, Short Work Week and Temporary Layoffs by Industry")

#Index of Retrenchments, Short Work Week and Temporary Layoffs Dataset
print(f"Index of Retrenchments, Short Work Week and Temporary Layoffs by Industry")

#Column of Retrenchments, Short Work Week and Temporary Layoffs Dataset
print(f"Columns of Retrenchments, Short Work Week and Temporary Layoffs by Industry")

#Renaming the column names
df_all.rename(
    columns={
        "year": "Year",
        "industry1": "Type_of_Industry",
        "retrench": "Total Retrenchments",
        "retrench_permanent": "Retrenchment of Permanent Jobs",
        "retrench_term_contract": "Retrenchment of Term Contract (Temporary) Jobs",
        "total": "Short Work Week and Temporary Layoffs (Total)",
        "short_work_week": "Short Work Week",
        "temporary_layoff": "Temporary Layoff"}
    ,inplace=True)

#First 5 rows of Retrenchments, Short Work Week and Temporary Layoffs Dataset
print("\n\t\tFirst 5 rows of Retrenchments, Short Work Week and Temporary Layoffs")
display(df_all.head())

#Last 5 rows of Retrenchments, Short Work Week and Temporary Layoffs Dataset
print("\n\t\tLast 5 rows of Retrenchments, Short Work Week and Temporary Layoffs")
display(df_all.tail())

#Display Summary information of Retrenchments, Short Work Week and Temporary Layoffs Dataset
print("\n\t\tSummary Information of Retrenchments, Short Work Week and Temporary Layoffs")
display(df_all.info())

#Display Statistical Information of Retrenchments, Short Work Week and Temporary Layoffs Dataset
print("\n\t\tStatistical information of Retrenchments, Short Work Week and Temporary Layoffs")
display(df_all.set_index("Year").describe(percentiles = [.20,.40,.60,.80]).T)

```

Shape of Retrenchments, Short Work Week and Temporary Layoffs by Industry Dataset:
(22, 7)

Index of Retrenchments, Short Work Week and Temporary Layoffs by Industry Dataset:
RangeIndex(start=0, stop=22, step=1)

Columns of Retrenchments, Short Work Week and Temporary Layoffs by Industry Dataset:

```
Index(['year', 'retrench', 'retrench_permanent', 'retrench_term_contract',
      'total', 'short_work_week', 'temporary_layoff'],
      dtype='object')
```

First 5 rows of Retrenchments, Short Work Week and Temporary Layoffs Dataset

	Year	Total Retrenchments	Retrenchment of Permanent Jobs	Retrenchment of Term Contract (Temporary) Jobs	Short Work Week and Temporary Layoffs (Total)	Short Work Week	Temporary Layoff
0	1998	32800	29090	3720	4450	3690	760
1	1999	15530	14620	910	800	680	120
2	2000	11950	11620	330	540	370	170
3	2001	27570	25840	1730	12500	11730	760
4	2002	20130	19090	1040	5540	4980	570

Last 5 rows of Retrenchments, Short Work Week and Temporary Layoffs Dataset

	Year	Total Retrenchments	Retrenchment of Permanent Jobs	Retrenchment of Term Contract (Temporary) Jobs	Short Work Week and Temporary Layoffs (Total)	Short Work Week	Temporary Layoff
17	2015	15580	13440	2140	810	660	150
18	2016	19170	16810	2360	950	820	130
19	2017	14720	12880	1840	1160	980	180
20	2018	10730	9610	1120	590	480	110
21	2019	10690	9810	880	890	820	80

Summary Information of Retrenchments, Short Work Week and Temporary Layoffs Dataset

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 22 entries, 0 to 21
```

```
Data columns (total 7 columns):
```

#	Column	Non-Null Count	Dtype
0	Year	22 non-null	int64
1	Total Retrenchments	22 non-null	int64

```

2   Retrenchment of Permanent Jobs                22 non-null    int64
3   Retrenchment of Term Contract (Temporary) Jobs 22 non-null    int64
4   Short Work Week and Temporary Layoffs (Total)  22 non-null    int64
5   Short Work Week                               22 non-null    int64
6   Temporary Layoff                              22 non-null    int64
dtypes: int64(7)
memory usage: 1.3 KB

```

None

Statistical information of Retrenchments, Short Work Week and Temporary Layoffs Dataset

	count	mean	std	min	20%	40%	50%	60%	80%
Total Retrenchments	22.0	15236.363636	6197.475327	8590.0	10698.0	11716.0	13010.0	15206.0	18760.0
Retrenchment of Permanent Jobs	22.0	13693.636364	5645.694985	7680.0	9698.0	10688.0	12110.0	13216.0	16720.0
Retrenchment of Term Contract (Temporary) Jobs	22.0	1544.090909	929.088950	330.0	864.0	1034.0	1230.0	1694.0	2120.0
Short Work Week and Temporary Layoffs (Total)	22.0	2637.272727	3681.842424	260.0	624.0	910.0	965.0	1120.0	4170.0
Short Work Week	22.0	2355.454545	3341.938028	150.0	504.0	820.0	915.0	992.0	3560.0
Temporary Layoff	22.0	282.727273	378.671636	20.0	70.0	114.0	140.0	166.0	410.0

```
In [3]: #convert column for total retrenchments to type and Column values to row values
df_melt_retrenchment = df_all.melt(value_vars=df_all[["Total Retrenchments"]],var

#Display top 5 years with Highest Retrenchment Rates
print("\nTop 5 years with Highest Total Number of Retrenchments")
highest_retrenchments = df_melt_retrenchment.sort_values(by="Number",ascending=Fa
#convert column for type to index using pivot
display(highest_retrenchments.pivot(index="Type",columns = "Year"))

#Display top 5 years with Lowest Retrenchment Rates
print("\nTop 5 years with Lowest Total Number of Retrenchments")
lowest_retrenchments = df_melt_retrenchment.sort_values(by="Number",ascending=Fa
#convert column for type to index using pivot
display(lowest_retrenchments.pivot(index="Type",columns = "Year"))

#convert column for total short work week and temporary layoff to type and Column
df_melt_short_work_week_and_temporary_layoff = df_all.melt(value_vars=df_all[["Sh

#Display top 5 years with Highest Total Number of Short Work Week and Temporary L
print("\nTop 5 years with Highest Total Number of Short Work Week and Temporary L
highest_retrenchments = df_melt_short_work_week_and_temporary_layoff.sort_values(
#convert column for type to index using pivot
display(highest_retrenchments.pivot(index="Type",columns = "Year"))

#Display top 5 years with Lowest Total Number of Short Work Week and Temporary La
print("\nTop 5 years with Lowest Total Number of Short Work Week and Temporary La
lowest_retrenchments = df_melt_short_work_week_and_temporary_layoff.sort_values(b
#convert column for type to index using pivot
display(lowest_retrenchments.pivot(index="Type",columns = "Year"))
```

Top 5 years with Highest Total Number of Retrenchments

Year	Number				
	1998	2001	2002	2009	2016
Type					
Total Retrenchments	32800	27570	20130	23430	19170

Top 5 years with Lowest Total Number of Retrenchments

Year	Number				
	2004	2007	2010	2011	2019
Type					
Total Retrenchments	10640	8590	9800	9990	10690

Top 5 years with Highest Total Number of Short Work Week and Temporary Layoffs

Year	Number				
	1998	2001	2002	2003	2009
Type					
Short Work Week and Temporary Layoffs (Total)	4450	12500	5540	4470	13610

Top 5 years with Lowest Total Number of Short Work Week and Temporary Layoffs

Year	Number				
	2000	2007	2010	2014	2018
Type					
Short Work Week and Temporary Layoffs (Total)	540	430	410	260	590

Write Python code that uses Python Visualisation Package (Matplotlib/Seaborn..) to produce useful data visualizations that explain the data.

Data Visualisation 5: Scatterplot (Seaborn and Matplotlib)

```

In [8]: import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker

#set seaborn background to darkgrid
sns.set_style("darkgrid")

#read from gdp growth file
gdp_growth = pd.read_csv("./Datasets/contribution-to-growth-in-gross-domestic-pro

#renaming gdp_growth column names
gdp_growth.rename(columns=
    {"year": "Year",
     "level_1": "GDP Growth in Chained Dollars (Base Year = 2015)",
     "value": "Percentage Growth (%)"},
    inplace=True)

#merging retrenchments, short work week and temporary layoff dataframe with gdp g
df_everything = pd.merge(df_all, gdp_growth, how="inner", on="Year")

#convert column for total retrenchments to type and Column values to row values
df_melt_retrenchment = df_everything.melt(value_vars=df_all[["Total Retrenchments

#convert column for total short work week and temporary layoff to type and Column
df_melt_short_work_week_and_layoff = df_everything.melt(value_vars=df_all[["Short

#declare fig and axes to plot
fig, ax = plt.subplots(2, 1, figsize=(20, 14))

#First Graph for Retrenchments

#declare ax0 to plot on the first subplot
ax0 = plt.subplot(211)

#plot using seaborn scatterplot method
sns.scatterplot(x="Number", y="Percentage Growth (%)", data=df_melt_retrenchment,

#plot using seaborn regplot to plot best fit line
sns.regplot(x="Number", y="Percentage Growth (%)", data=df_melt_retrenchment, color

#set title for first graph plotting
ax0.set_title("Relationship Between GDP Percentage Growth (%) and Retrenchments",

#Second Graph for total short work week and temporary layoff

#declare ax1 to plot on the second subplot
ax1 = plt.subplot(212)

#plot using seaborn scatterplot method
sns.scatterplot(x="Number", y="Percentage Growth (%)", data=df_melt_short_work_w

#plot using seaborn regplot to plot best fit line
sns.regplot(x="Number", y="Percentage Growth (%)", data=df_melt_short_work_week_ar

#set title for second graph plotting
ax1.set_title("Relationship Between GDP Percentage Growth (%) and Short Work Week

```



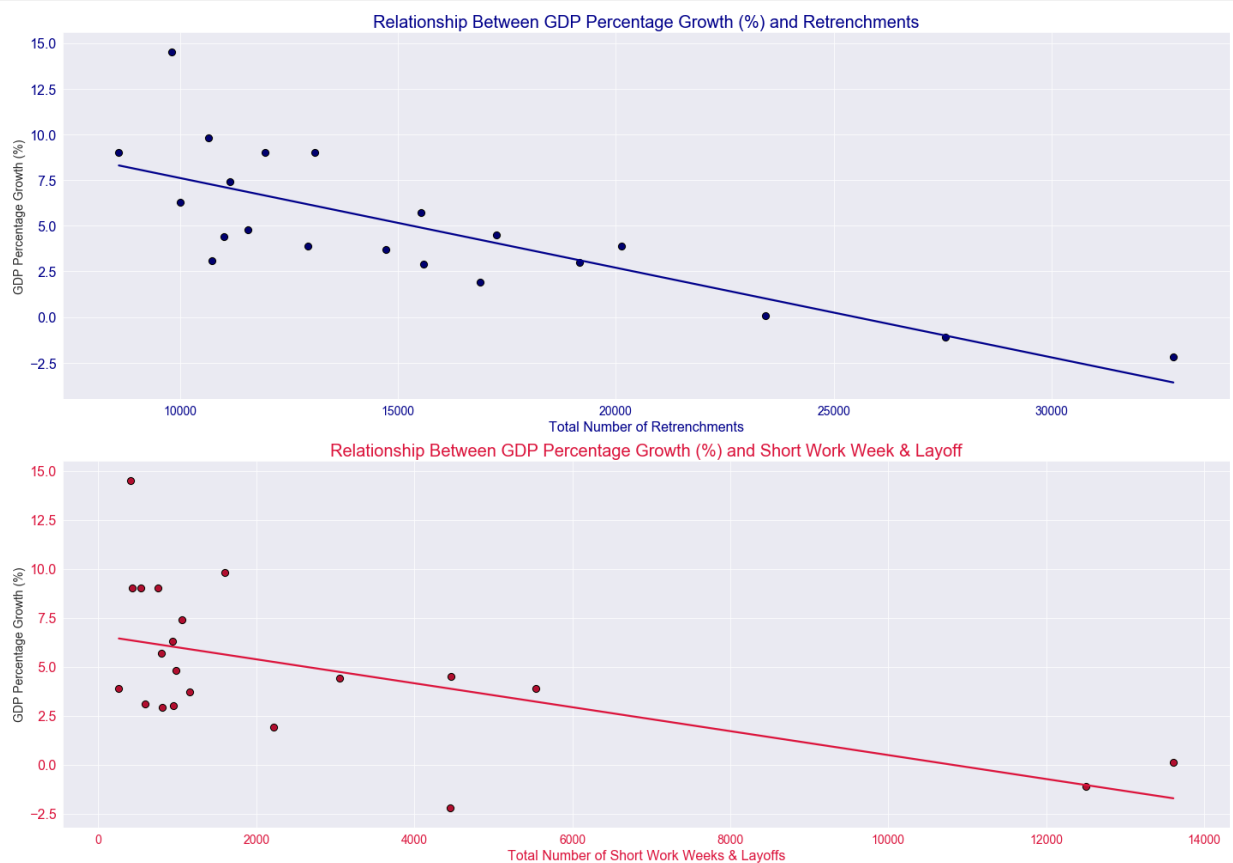
```

#set fontsize, color for x labels, y labels, x ticks and y ticks
ax0.tick_params(axis="x", labelsz=14, colors='darkblue'), ax1.tick_params(axis=
ax0.tick_params(axis="y", labelsz=16, colors='darkblue'), ax1.tick_params(axis=
ax0.set_xlabel('Total Number of Retrenchments', fontsize=16, color='darkblue'), a
ax0.set_ylabel('GDP Percentage Growth (%)', fontsize=14), ax1.set_ylabel("GDP Per

#to display both graph nicely
fig.tight_layout()

#display graph
plt.show()

```



For the chosen datasets, explain the nature of that dataset (i.e. what is in that dataset) or any peculiarities about it you wish to highlight and explain the process you went through to analyse that dataset, . Where possible, you should specifically mention how you used the Pandas, Matplotlib, Seaborn functions to achieve a certain outcome e.g. to transform the data or to produce a certain visualization:

Peculiarities to highlight:

The total number of short work week & layoffs is much lesser than total retrenchments, with many data points hovering between values of 0-2000, which could have resulted in relative weak relationship between both variables. One possible reason for this is because Employers tend to only apply such measures during extreme circumstances, compared to retrenchments.

Process of using Pandas, Matplotlib or Seaborn functions to transform the data:

Using Pandas to transform

Firstly, I read the excel file from retrenchments and short work week and layoffs and store it in two variables. Then, I merge both dataframes into one and store it as `df_all`. After that, I display the shape, index and column names of the `df_all` dataset so that I am able to rename the column names. Moving on, I used the head and tail method to display the first and last five rows. To gather summary information of the data, I used the info method. Then, I display Statistical Information of Retrenchments, Short Work Week and Temporary Layoffs Dataset using the describe method.

In addition to that, I am also interested to know the top 5 years where there are high/low retrenchment rates, high/low total number of short work week and layoff. In order to display this information, I first converted the column for total retrenchments to type and Column values to row values. In order to display the top 5 years where there are high retrenchment rates, I sorted the values by type, in ascending order, using the head method and store it as a variable. Then, I displayed the top 5 retrenchments using the pivot function. Similarly, the method to display top 5 years with lowest retrenchment rates is the same except for using the tail method. The way to display the top 5 years for total short work week and layoff is also the same as the retrenchment rates.

Using Matplotlib or Seaborn functions to transform the data:

Data Visualisation 5: Scatterplot

Firstly, I set style of Seaborn Graph background to dark and read from the gdp growth excel file. After that, I rename the column names for gdp growth and dropped level 1 columns which identifies each row as the GDP Growth Rate. Then, I merged retrenchments, short work week and temporary layoff dataframe with gdp growth on year into one dataframe called `df_all`. In order to display total retrenchments and total short work week and layoff using seaborn, I have to convert column for total retrenchments to type and Column values to row values as well as total short work week and layoff.

Then, I declare figure and axes object to plot. Then, I declare `ax0` on the first plot and plot the graph using seaborn scatterplot and regplot (best fit line) using `df_melt_retrenchment` dataframe. Then, I set the title for the first subplot. Similarly, the process to plot the Second Graph for total short work week and temporary layoff is the same except for specifying `ax1` on the second plot and using a `df_total_short_work_week` and layoff dataframe. In order to differentiate between both graphs, I specify the colors for retrenchments and total_short_week and layoffs to crimson and darkblue respectively for xticks, yticks, xlabel and title. Then, I call `fig.tight_layout` so that both graph will not interfere one another and `plt.show` to display the graph.

For each dataset, highlight the insights you have gained from analysing the data and any conclusions or recommendations you want to make as a result of the analysis:

Data Visualisation 5: Scatterplot

Since the scatterplot is used to distinguish the relationship between two variables, the scatterplot is the last figure plotted for data visualisation to further justify that the insights gained from such analysing the dataset is accurate.

From the graph, I am able to conclude that the hypothesis of a strong correlation between GDP Growth, retrenchments and short work week & layoffs is correct. This is evident as there is strong linear relationship between GDP Percentage Growth and retrenchments, as well as short work week and layoffs. When there are high GDP Growth Rates, the total number of retrenchments is one of the lowest. When there are low GDP Growth Rates, the total number of retrenchments/short work week and layoff is one of the highest. As such, this shows that Singapore is heavily influenced by Global Events as it has a strong impact on the number of retrenchments, short work week and layoff.

Key Takeaways from Data Analysis

- Singapore's Prosperous Economy is largely due to its International financial trade center, which makes it attractive for foreign investments in Singapore.
- Given the image that Singapore portrays to the world, the success of Singapore's Economy is largely intertwined with the Global Economy.
- As such, Singapore is very vulnerable to Global Economic Crisis as it relies heavily on global trade demands.
- Looking forward, Singapore needs to constantly reinvent itself in order to survive for challenges in the decades to come.