# INDIA STARTUP ECOSYSTEM DATA ANALYSIS

Photo by Luke Chesser on Unsplash

***DISCLAIMER: THIS PROJECT WAS DONE BY A GROUP OF STUDENTS ENROLLED IN THE AZUBI AFRICA DATA ANALYTICS PROFESSIONAL PROGRAM THE GROUP NAME IS TEAM SAN-FRANCISCO.***

***FOR EDUCATIONAL PURPOSES ONLY.***

## INTRODUCTION:

India’s startup ecosystem has seen remarkable growth in recent years, with funding pouring in from various investors. As a stakeholder in the industry, it is important to understand the key trends and insights of the Indian startup landscape. In this article, we will answer five critical questions about the Indian startup ecosystem, based on the data collected from 2018 to 2021. The analysis closely follows The Cross Industry Standard Process for Data Mining (CRISP-DM)

## BUSINESS UNDERSTANDING:

Startups are part of a larger business ecosystem; they do not live in a vacuum. The development drivers of the Indian startup ecosystem must thus be understood in light of several variables, including recent market trends, historical economic changes, the influence of technical advancement, and shifting societal views.

India is often described as “the poster child of emerging markets” for its vast commercial potential for startups. In a country with a population of nearly 1.3 billion people, even niche products can have significant market potential. In the 1990s, economic reforms moved India towards a more market-based economic system. Since this liberalization, the overall economic development has been dynamic, and as of 2017, the Indian economy had a GDP of US$2.726 trillion. With a GDP growth of 7.0 percent in 2018, India is one of the fastest-growing large economies in the world. Therefore, the Indian market is perceived as being capable of offering an abundance of opportunities for startups

This article provides insights into the Indian startup ecosystem by answering five key questions based on the data collected from 2018 to 2021. It discusses the highest-funded startups, significant investors, sectors receiving the highest amount of funding, trends in funding over the years, and the difference in funding received by startups in different regions of India

**QUESTIONS**

1. What are the startups that received the highest funding in each year?

2. Who are the major investors in the Indian start-up ecosystem and what is the amount of funding provided by them?

3. Which sectors or industries received the highest amount of funding in India from 2018 to 2021, and which sector received the highest amount of funding during this period?

4. What is the trend in the amount of funding received by Indian startups over the years, and is there a correlation between the year of startup and the amount of funding received?

5. How many new startups are formed yearly, and is there a difference in the amount of funding received by startups in different regions of India?

**Hypothesis:**

- Ho: The amount of funding received by Indian startups has NOT changed over the years.

- H1: The amount of funding received by Indian startups has changed over the years.

## DATA UNDERSTANDING

**Datasets Structure (Columns):**

The datasets contain the following column heading:

1. Year: The year of funding

2. Company/Brand: The name of the startup/company/brand

3. Founded: The year in which the company was founded

4. HeadQuarter: The city in which the company is headquartered

5. Sector: The sector to which the company belongs (e.g., AgriTech, EdTech, FinTech, etc.)

6. What it does: A brief description of what the company does

7. Founders: The names of the founders of the company

8. Investor: The names of the investors who provided the funding

9. Amount($): The amount of funding received by the company in US dollars

10. Stage: The stage of funding received by the company (e.g., Pre-seed, Seed, Series A, etc.)

## DATA PREPARATION

### Importation: python libraries

In this section, we import all the libraries need for this project. Here we explain the need of each library:

1. pandas is a library for data manipulation and analysis. It provides data structures for efficiently storing and manipulating large datasets and tools for working with tabular data, such as dataframes.

2. numpy is a library for numerical computing in Python. It provides tools for working with arrays and matrices, as well as a range of mathematical functions for working with numerical data.

3. summarytools is a library for generating summary statistics and exploratory data analysis (EDA) reports for dataframes. It provides functions to generate tables and plots summarizing the distribution of variables, missing values, correlations, and more.

4. seaborn is a library for data visualization built on top of matplotlib. It provides a high-level interface for creating statistical graphics, such as bar plots, box plots, heatmaps, and more.

5. matplotlib is a library for creating static, animated, and interactive visualizations in Python. It provides a wide range of tools for creating different types of plots and charts, such as line plots, scatter plots, histograms, and more.

6. re is a built-in library for regular expression operations. It provides a set of functions for working with strings, such as searching, replacing, and splitting text based on patterns specified using regular expressions. It is commonly used for data cleaning and preprocessing, especially when dealing with text data.

```
# import relevant libraries
import pandas as pd
import numpy as np
from summarytools import dfSummary
import seaborn as sns
import matplotlib.pyplot as plt

import re

#hypothesis testing
import scipy.stats as stats
import statsmodels.api as sm
from statsmodels.formula.api import ols
```

## Data Loading

- Here we load all the datasets and all additional files that are to be used in the project.

```
# read in the four datasets
df_2018 = pd.read_csv('Dataset\startup_funding2018.csv')
df_2019 = pd.read_csv('Dataset\startup_funding2019.csv')
df_2020 = pd.read_csv('Dataset\startup_funding2020.csv')
df_2021 = pd.read_csv('Dataset\startup_funding2021.csv')
```

## DATA CLEANING

Due to the rigorous data cleaning and preparation steps behind this analysis, I will skip to answering the business questions and hypotheses testing.

However, the link to the notebook has been included for your further use.

Some cleaning techniques used include.

1. Removing missing values

2. striping of values with the rupee symbol and dollar symbol attached

3. Removing duplicates

4. Converting supposed floats columns to float since they were objects

5. we used four datasets collected from 2018 to 2021 since they were collected at different points in time, the columns were different for some datasets, for this reason, some columns were renamed to match the columns in other datasets and finally, the datasets are concatenated. The resulting Dataframe was named df_startup

Etc.

Other steps skipped in this article include

Exploratory Data analysis

Univariate analysis

Multivariate analysis

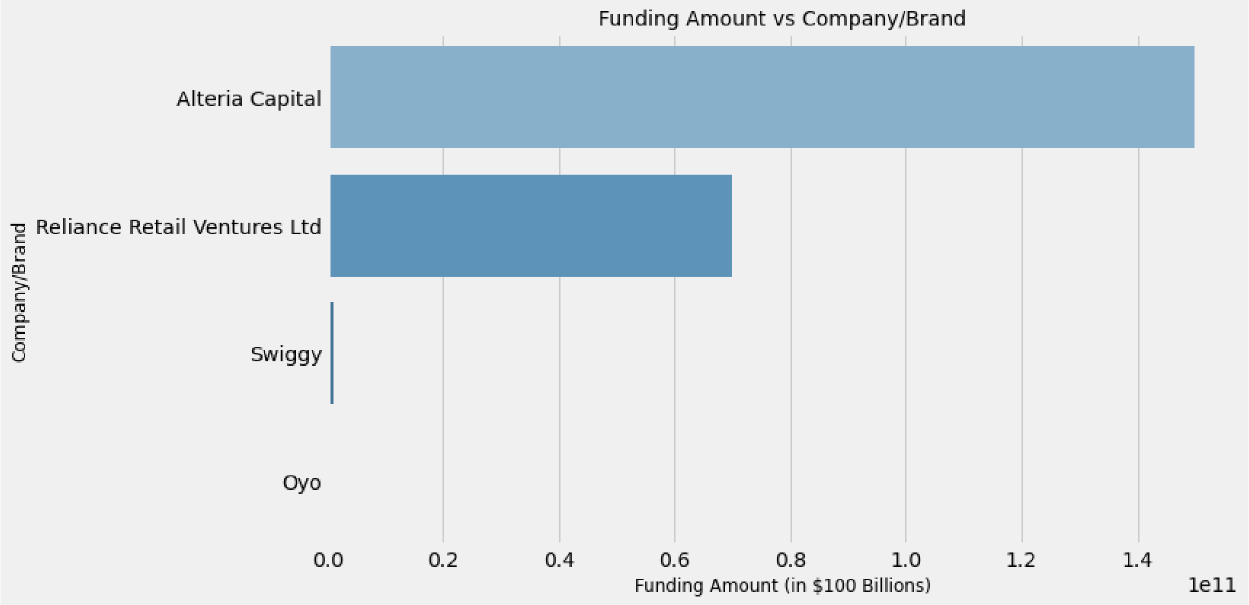
Which can also be found in the notebook linked above

## ANSWERING THE BUSINESS QUESTION

### Q1. What are the startups that received the highest funding each year?

```
find the startups that received the highest funding in each year
df_top_funding = df_startup.sort_values('Amount($)',
ascending=False).groupby('Year').first().sort_values('Amount($)',
ascending=False)
 create barplot using Seaborn
plt.figure(figsize=(8, 6))
```

```python
sns.barplot(x="Amount($)", y="Company/Brand", data=df_top_funding,
color='blue')

# set plot title and axis labels
plt.title("Funding Amount vs Company/Brand", fontsize=14)
plt.xlabel("Funding Amount (in $100 Billions)", fontsize=12)
plt.ylabel("Company/Brand", fontsize=12)

# show the plot
plt.show()
```



The above visualization answers Question 1 by showing the startups that received the highest funding each year. The result is as follows: The exact values can be known by using the pandas head method on the df_top_funding dataframe

1. In the Year 2021 the Company/Brand Alteria Capital received the highest funding of $150,000,000,000.00

2. This was followed by the Year 2020 and the Company/Brand was Reliance Retail Ventures Ltd getting funding of $70,000,000,000.00

3. The third was in the Year 2018 by the Company/Brand called Swiggy receiving funding of $1,000,000,000.00

4. The last was in the Year 2019 by the Company/Brand called Oyo which received funding of $693,000,000.00

**Q2. Who are the major investors in the Indian start-up ecosystem and what is the amount of funding provided by them?**

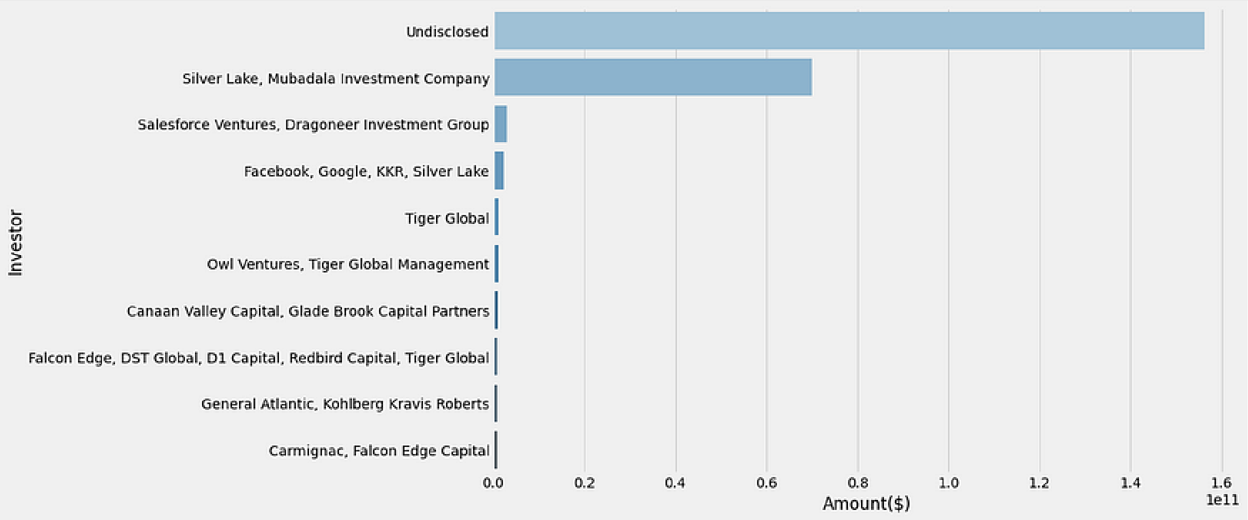
**We need to compute the total amount of funding received by each investor in the df_startup DataFrame.**

- This is achieved by grouping the DataFrame by the “Investor” column using the groupby() method, and then summing the “Amount($)” column for each group using the sum() method.

- The reset_index() method is called to reset the index of the resulting DataFrame.

- Then we sort the values in descending order

```
investor_funding =
df_startup.groupby('Investor')['Amount($)'].sum().reset_index()
investor_funding = investor_funding.sort_values(by='Amount($)',
ascending=False)
create a horizontal bar plot using Seaborn
plt.figure(figsize=(12, 8))
sns.barplot(x="Amount($)", y="Investor", data=investor_funding.head(10),
color='blue')

# set plot title and axis labels
plt.title("Top 10 Investors by Total Funding Amount ($ Billion)",
fontsize=14)
plt.xlabel("Total Funding Amount ($100 B)", fontsize=12)
plt.ylabel("Investor", fontsize=12)

# show the plot
plt.show()
```



The above visualization shows the major investors in the Indian start-up ecosystem and the amount of funding in Billions provided. The first 10 major Investors are as follows:

1. Undisclosed $156,355,472,938.00

2. Silver Lake, Mubadala Investment Company $70,000,000,000.00

3. Salesforce Ventures, Dragoneer Investment Group $3,000,000,000.00

4. Facebook, Google, KKR, Silver Lake $2,200,000,000.00

5. Tiger Global $1,217,000,000.00

6. Owl Ventures, Tiger Global Management $1,200,000,000.00

7. Canaan Valley Capital, Glade Brook Capital Par...
$1,000,000,000.00

8. Falcon Edge, DST Global, D1 Capital, Redbird C... $840,000,000.00

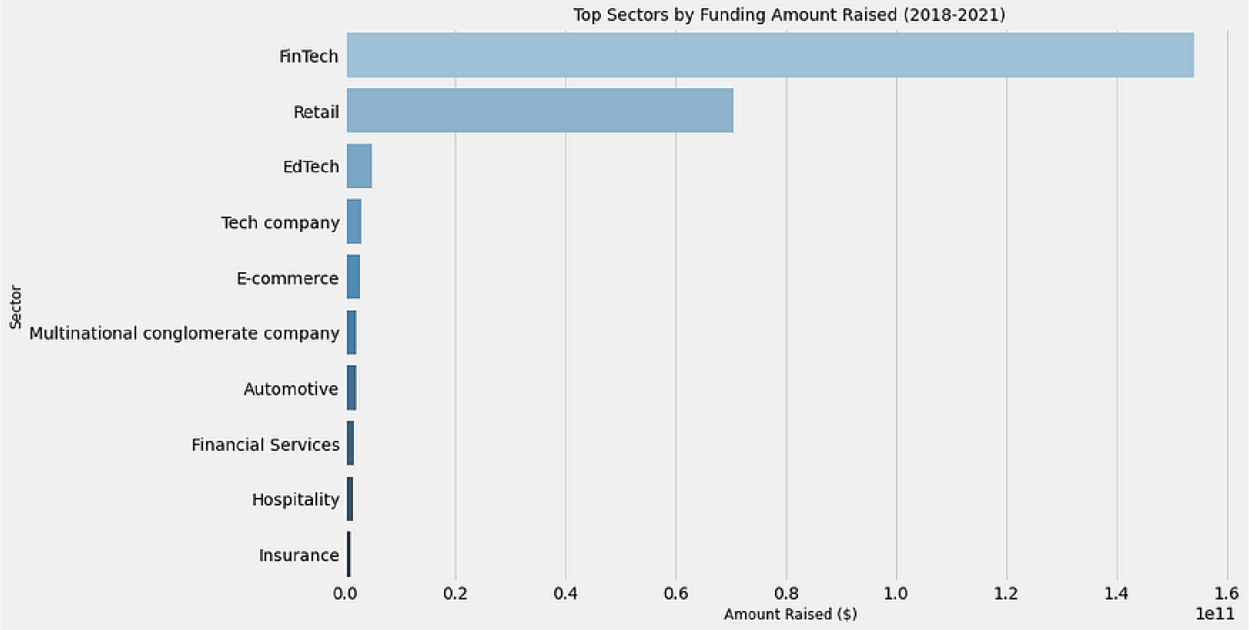
9. General Atlantic, Kohlberg Kravis Roberts $800,000,000.00

110. Carmignac, Falcon Edge Capital $800,000,000.00

## Q3. Which sectors or industries received the highest amount of funding in India from 2018 to 2021, and which sector received the highest amount of funding during this period?

```
# filter the data for the years 2018-2021
df_startup = df_startup[(df_startup["Year"] >= 2018) & (df_startup["Year"] <=
2021)]
# group the data by sector and sum the amount raised by each sector
sector_funding =
df_startup.groupby("Sector")["Amount($)"].sum().reset_index()
# sort the data by the amount raised in descending order
sector_funding = sector_funding.sort_values("Amount($)", ascending=False)
# print the top 5 sectors by funding amount
print(sector_funding.head(10))
# plot the top sectors by funding amount raised using Seaborn's barplot
function
plt.figure(figsize=(12, 8))
```

```
sns.barplot(x="Amount($)", y="Sector", data=sector_funding.head(10),
color='blue')

# set plot title and axis labels
plt.title("Top Sectors by Funding Amount Raised (2018-2021)", fontsize=14)
plt.xlabel("Amount Raised ($100 Billion)", fontsize=12)
plt.ylabel("Sector", fontsize=12)

# show the plot
plt.show()
```



The above visualization answers Question 3. which finds out the sectors or industries that received the highest amount of funding in India from 2018 to 2021, and which sector received the highest amount of funding during this period. Here the Barchart shows the top 10 Sectors as follows:

1. FinTech $153,915,410,000.00

2. Retail $70,547,020,351.00

3. EdTech $4,967,618,330.00

4. Tech company $3,028,700,000.00

5. E-commerce $2,898,052,000.00

6. Multinational conglomerate company $2,200,000,000.00

7. Automotive $2,130,083,041.00

8. Financial Services $1,780,925,146.00

9. Hospitality $1,628,903,099.00

10. Insurance $1,099,650,000.00

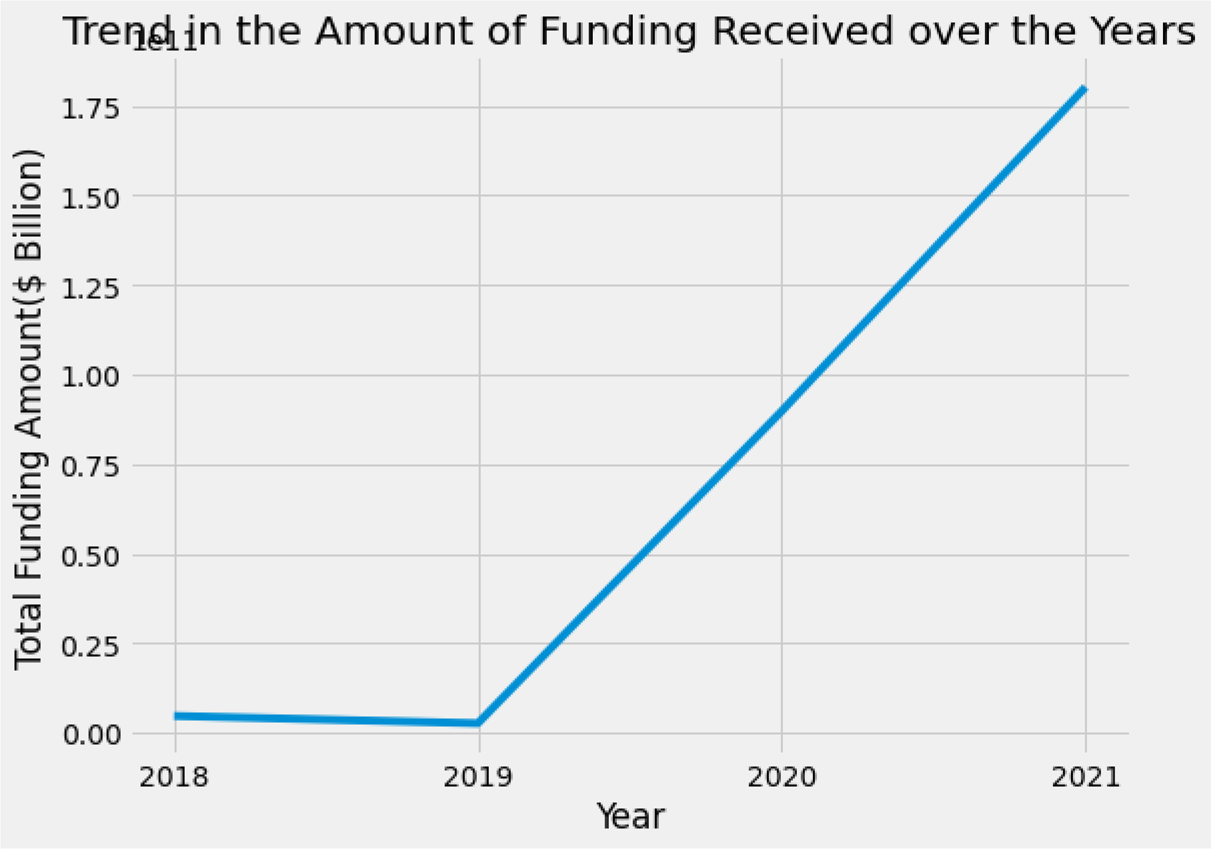
### Q4. What is the trend in the amount of funding received by Indian startups over the years, and is there a correlation between the year of startup and the amount of funding received?

1. First, group the startups by year and calculate the total funding amount received each year using pandas.

2. Visualize the trend in the amount of funding received over the years using a line plot.

### 3. Check for the correlation between the year of startup and the amount of funding received

```
plt.figure(figsize=(8, 6))

# convert 'Year' column to pandas Datetime object
df_startup['Year'] = pd.to_datetime(df_startup['Year'], format='%Y')

# sort the DataFrame by 'Year' column
df_startup = df_startup.sort_values(by='Year')

# group the DataFrame by 'Year' and calculate the total funding amount for
each year
df_funding_by_year = df_startup.groupby('Year')['Amount($)'].sum()

# create a line plot
plt.plot(df_funding_by_year.index.strftime('%Y'), df_funding_by_year.values)

# set the title and axis labels
plt.title("Trend in the Amount of Funding Received over the Years")
plt.xlabel("Year")
plt.ylabel("Total Funding Amount($ Billion)")

# set the x-axis ticks to show only one value per year
plt.xticks(df_funding_by_year.index.strftime('%Y')[::1], rotation=0)

# show the plot
plt.show()
```

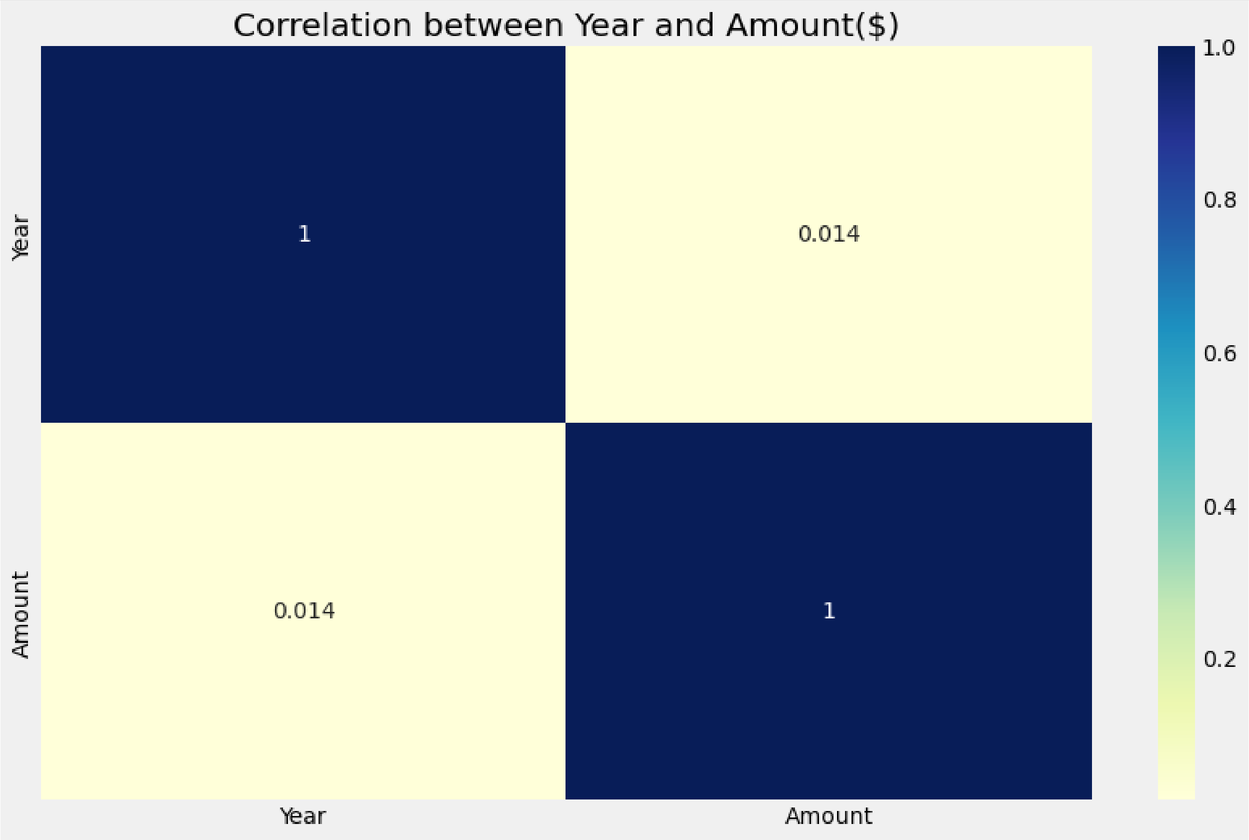


The Line Graph shows the trend in the amount of funding received by Indian startups over the years.

- Here we see that there was a downward trend from 2018 to 2019 but thereafter, a steady upward trend from 2019 to 2021.

```
# calculate the correlation coefficient between 'Year' and 'Amount($)'
corr_coeff = df_startup['Year'].dt.year.corr(df_startup['Amount($)'])

print("Correlation coefficient between Year and Amount($): ", corr_coeff)
# Convert 'Year' column to datetime year only
df_startup['Year'] = pd.to_datetime(df_startup['Year'], format='%Y').dt.year

# Create a new DataFrame with 'Year' and 'Amount($)' columns
df_year_amt = pd.DataFrame({'Year': df_startup['Year'], 'Amount':
df_startup['Amount($)']})
```

```
# Print the new DataFrame
df_year_amt.head()
corr_matrix = df_year_amt.corr()

# Create a heatmap of the correlation in the df_year_amt
sns.heatmap(corr_matrix, annot=True, cmap='YlGnBu')

# add a title to the plot
plt.title('Correlation between Year and Amount($)')

# Show the plot
plt.show()
```



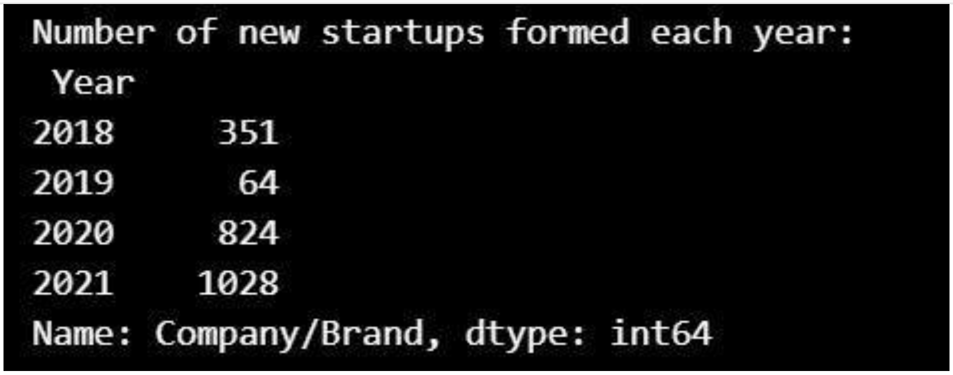
The Correlation Matrix shows that there was a very small positive correlation of only 0.014 between the year of startup and the amount of funding received.

### Q5. How many new startups are formed yearly, and is there a difference in the amount of funding received by startups in different regions of India?

Question 5 can be divided into two parts:

1. We use the pandas groupby() method to group the DataFrame by ‘Year’ and then count the number of unique startups in each year using the nunique() method.

```
# Convert 'Year' column to datetime year only
df_startup['Year'] = pd.to_datetime(df_startup['Year'], format='%Y').dt.year

# group the DataFrame by 'Year' and count the number of unique startups in
each year
num_startups_by_year = df_startup.groupby('Year')['Company/Brand'].nunique()

# print the number of new startups formed each year
print("Number of new startups formed each year:\n", num_startups_by_year)
```



```
Number of new startups formed each year:
 Year
2018     351
2019      64
2020     824
2021    1028
Name: Company/Brand, dtype: int64
```

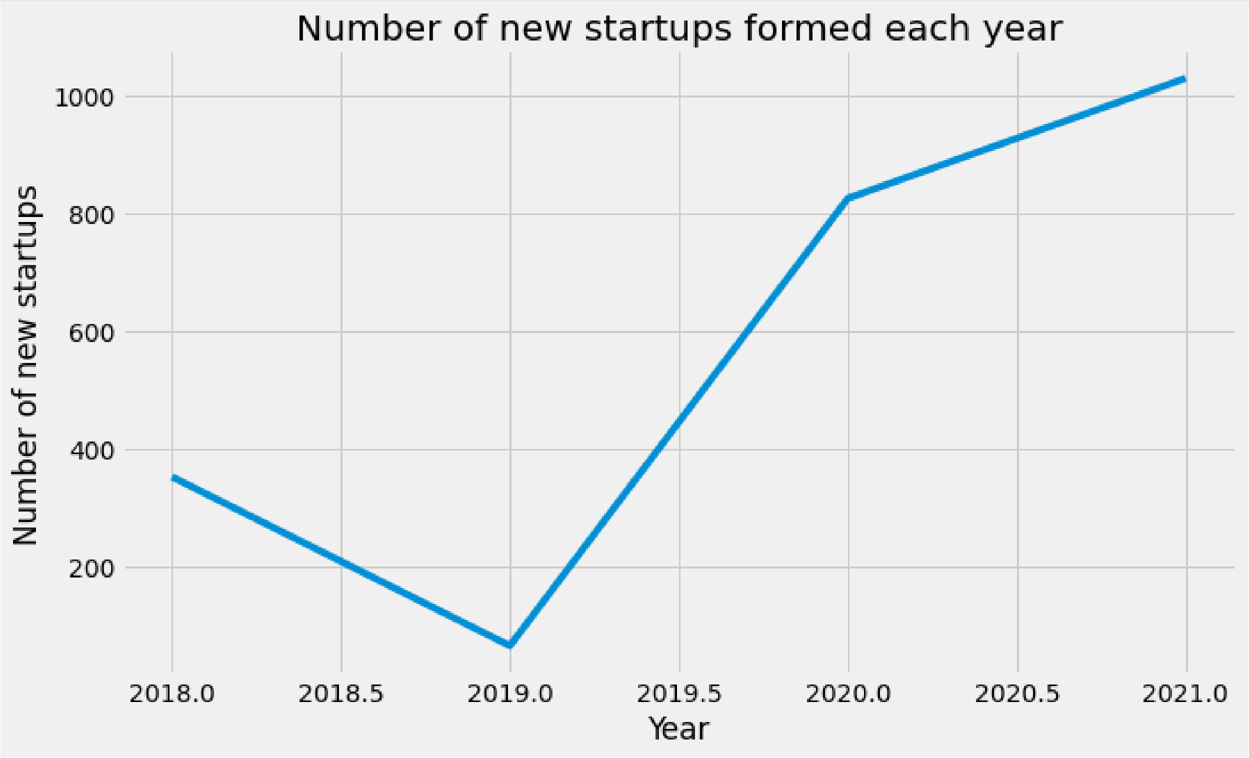
In 2018 we had 351 new Start-ups

- In 2019 we had 64 new Start-ups

- In 2020 we had 824 new Start-ups, and

- In 2021 we had 1028 new Start-ups

```
# plot the number of new startups formed each year
plt.figure(figsize=(8, 6))
plt.plot(num_startups_by_year.loc[2018:2021].index,
num_startups_by_year.loc[2018:2021].values)
plt.xticks(num_startups_by_year.loc[2018:2021].index)
plt.xlabel('Year')
plt.ylabel('Number of new startups')
plt.title('Number of new startups formed each year')

# set x-ticks to show only one value per year
plt.show()
```



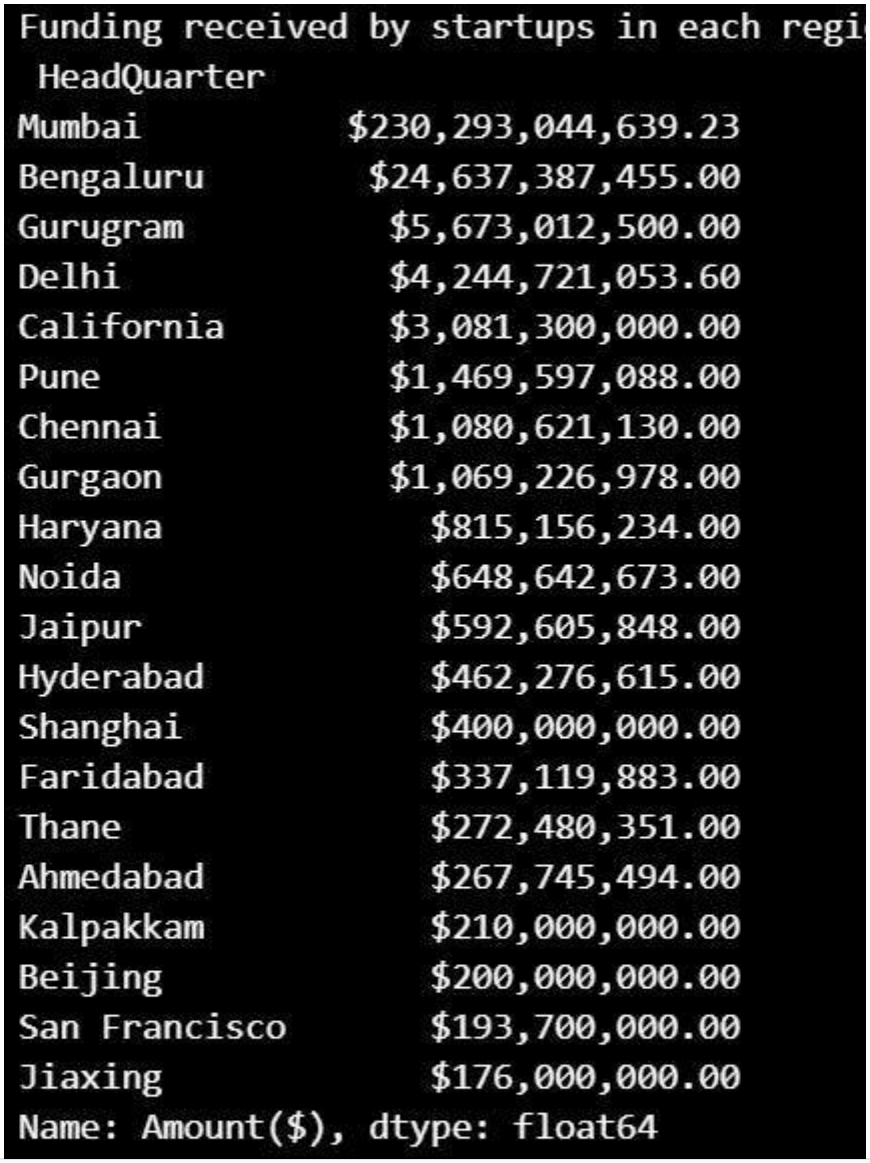
The line graph shows the number of startups formed each year reduced from 2018 to 2019 however there has been an upward trend after the year 2019

2. We group the DataFrame by ‘HeadQuarter’ column and calculate the total funding amount received by startups in each region using the groupby() and sum() methods.

- The HeadQuarters are then sorted out in descending order of the Amount($) to show the top 20.

```
# group the DataFrame by 'HeadQuarter' and calculate the total funding amount received by startups in each region
funding_by_region = df_startup.groupby('HeadQuarter')['Amount($)'].sum()

# sort the regions by total funding amount and take only the top 20
top20 = funding_by_region.sort_values(ascending=False)[:20]

# print the funding received by startups in each region
print("Funding received by startups in each region:\n", top20)
```

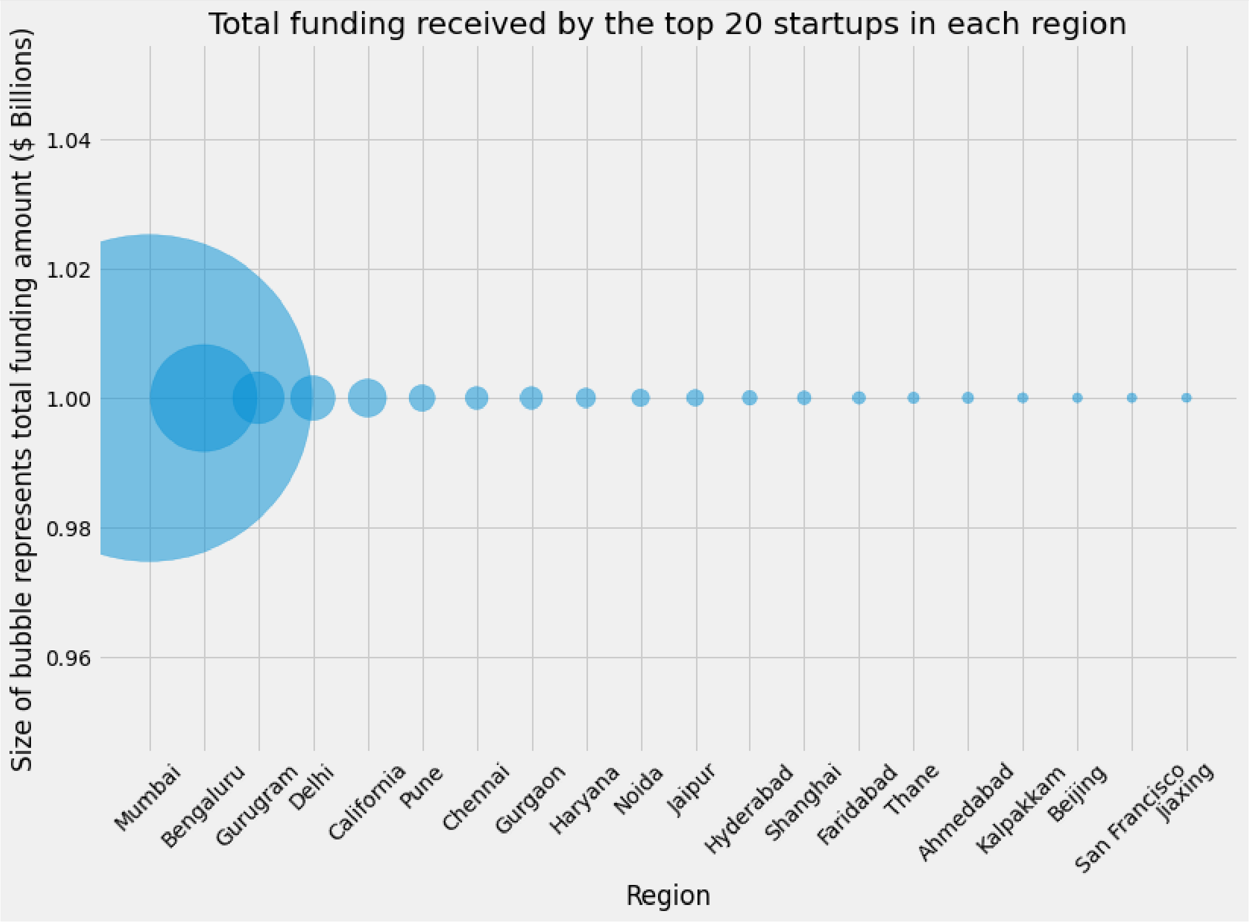
The result is shown below



```
Funding received by startups in each regi
 HeadQuarter
Mumbai          $230,293,044,639.23
Bengaluru        $24,637,387,455.00
Gurugram          $5,673,012,500.00
Delhi             $4,244,721,053.60
California        $3,081,300,000.00
Pune              $1,469,597,088.00
Chennai           $1,080,621,130.00
Gurgaon           $1,069,226,978.00
Haryana             $815,156,234.00
Noida               $648,642,673.00
Jaipur              $592,605,848.00
Hyderabad           $462,276,615.00
Shanghai            $400,000,000.00
Faridabad           $337,119,883.00
Thane               $272,480,351.00
Ahmedabad           $267,745,494.00
Kalpakkam           $210,000,000.00
Beijing             $200,000,000.00
San Francisco       $193,700,000.00
Jiaxing             $176,000,000.00
Name: Amount($), dtype: float64
```

```
# Sort the funding_by_region DataFrame in descending order by the total
funding amount
funding_by_region_sorted = funding_by_region.sort_values(ascending=False)

# Take only the top 20 rows
funding_by_region_top20 = funding_by_region_sorted[:20]

# Define a list of colors corresponding to the regions
# colors = plt.cm.tab20(np.arange(len(funding_by_region_top20)))

plt.figure(figsize=(10, 6))
plt.scatter(funding_by_region_top20.index, [1]*len(funding_by_region_top20),
s=funding_by_region_top20.values/5000000, alpha=0.5, color='blue')#c='colors'
```

```
    plt.xticks(rotation=45)
    plt.xlabel('Region')
    plt.ylabel('Size of bubble represents total funding amount ($ Billions)')
    plt.title('Total funding received by the top 20 startups in each region')
    plt.show()
```



### Testing the Hypothesis:

Ho: The amount of funding received by Indian startups has NOT changed over the years.

H1: The amount of funding received by Indian startups has changed over the years.

**To test the Hypothesis, we shall use:**

1. t-test and

2. ANOVA to find the p-values.

1. Convert the ‘Year’ column in the DataFrame df_startup to datetime format using the pd.to_datetime() function, specifying the format as ‘%Y’.

2. Then extract the year from the datetime values in the ‘Year’ column using the .dt.year attribute.

3. We then divide the data into groups by year using the .groupby() method on the ‘Year’ column, and apply the .apply() method with the ‘Amount($)’ column to create a list of funding amounts for each year.

4. Then perform a t-test between the funding amounts in 2018 and 2021 using the stats.ttest_ind() function, and store the result in the ttest_result variable, then print the t-test result.

5. We shall perform pairwise t-tests between the Amount($) values for each pair of years and print the t-value and p-value for each comparison. If the p-value is less than your chosen significance level (e.g., 0.05), we can reject the null hypothesis of equal means and conclude that the means are significantly different.

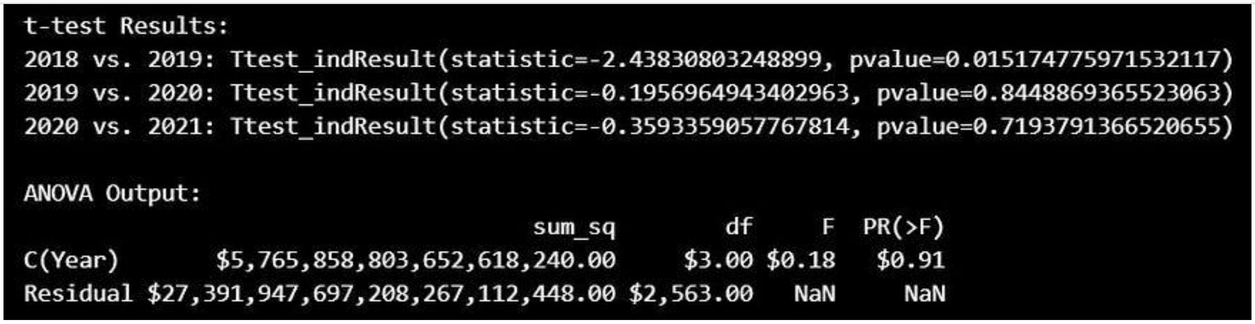
6. We perform an ANOVA to compare the means between all years using the ols() function with the ‘Amount($)’ column as the dependent variable and the ‘Year’ column as the independent variable. We then fit the model using the .fit() method, and store the result in the model variable.

7. Finally we calculate the ANOVA results using the sm.stats.anova_lm() function with the model and specify the type as 2. We then store the result in the anova_result variable and then print the ANOVA results.

**Note:**

Type 2 ANOVA is generally preferred over type 1 ANOVA because it takes into account the interactions between factors, and can be more informative when there are multiple factors in the model. This is why we specified ‘type’ as 2.

```
# Convert 'Year' column to datetime format
df_startup['Year'] = pd.to_datetime(df_startup['Year'], format='%Y')

# extract the year from the datetime values in the 'Year' column
df_startup['Year'] = df_startup['Year'].dt.year

# divide the data into groups by year
df_year = df_startup.groupby('Year')['Amount($)'].apply(list)

# Extract the Amount($) lists for each year:
amounts_2018 = df_year.loc[2018]
amounts_2019 = df_year.loc[2019]
amounts_2020 = df_year.loc[2020]
amounts_2021 = df_year.loc[2021]

# Use the ttest_ind function to compare the means of each pair of years:
ttest_2018_2019 = stats.ttest_ind(amounts_2018, amounts_2019)
ttest_2019_2020 = stats.ttest_ind(amounts_2019, amounts_2020)
ttest_2020_2021 = stats.ttest_ind(amounts_2020, amounts_2021)

# Print the results:
print('t-test Results:')
print('2018 vs. 2019:', ttest_2018_2019)
print('2019 vs. 2020:', ttest_2019_2020)
print('2020 vs. 2021:', ttest_2020_2021)
print()

print('ANOVA Output:')
# perform an ANOVA to compare the means between all years
model = ols('Q("Amount($)") ~ C(Year)', data=df_startup).fit()
anova_result = sm.stats.anova_lm(model, typ=2)
print(anova_result)
```



```
t-test Results:
2018 vs. 2019: Ttest_indResult(statistic=-2.43830803248899, pvalue=0.015174775971532117)
2019 vs. 2020: Ttest_indResult(statistic=-0.1956964943402963, pvalue=0.8448869365523063)
2020 vs. 2021: Ttest_indResult(statistic=-0.3593359057767814, pvalue=0.7193791366520655)

ANOVA Output:
                                         sum_sq        df     F  PR(>F)
C(Year)       $5,765,858,803,652,618,240.00     $3.00 $0.18   $0.91
Residual $27,391,947,697,208,267,112,448.00 $2,563.00   NaN     NaN
```

**From our results, we can conclude that:**

1. Based on the p-values, we can conclude that:

- there is a significant difference between the mean Amount for the years 2018 and 2019 (p-value=0.015)

- However, there is not a significant difference between the mean Amount for the years 2019 and 2020 (p-value=0.845), or between 2020 and 2021 (p-value=0.719).

- Therefore, we can conclude that there was a <b>significant increase in the mean Amount from 2018 to 2019</b>, but there was no significant change from 2019 to 2020, or from 2020 to 2021.

2. The ANOVA result compares the mean funding amount across all years. The PR(>F) value of 0.91 indicates that the overall difference in funding across all years is not statistically significant.

**Therefore this suggests that there has not been a significant change in the amount of funding received by startups over the years. So we CAN NOT reject the Null Hypothesis.**

- The t-test and ANOVA results suggest that there may be a difference in the funding amounts received in 2018 and 2021, but more analysis is needed to confirm this. We could also explore other factors that may influence the funding amounts received by startups, such as location, industry, and business model, to gain a deeper understanding of the startup ecosystem.

## CONCLUSION

The Indian startup ecosystem has been growing at an impressive rate, with significant funding pouring in from various investors. The fintech sector has been the highest-funded sector, with a total of $153.9 billion in funding. The total funding received in the Indian startup ecosystem has been increasing steadily, with $180.2 billion in funding received in 2021. There is no strong correlation between the year of the startup and the amount of funding received by Indian startups. Mumbai was the region with the highest amount of funding, followed by Bengaluru. Overall, the Indian startup ecosystem offers vast opportunities for investors and entrepreneurs alike, and this trend is expected to continue in the future.

***ACKNOWLEDGEMENT:***

*JACOB ORIANG JAROYA*

*KINGSLEY YAW ASIEDU*

*FELIX KIPROTICH*

*RICHARD TEYE LAWER*