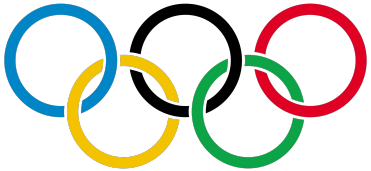
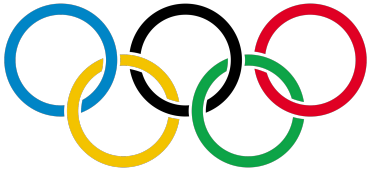


The Olympic Hosting Effect

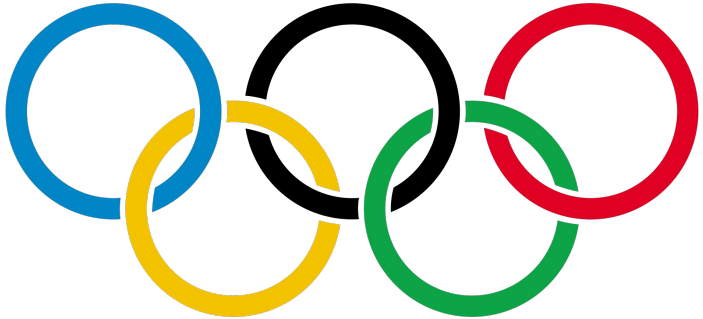
Fatma Butun, Jumaan George, Raj Ravi, Timmy Reynolds



Motivation for our Project

Core Message

- The Olympics is a massive international event that brings together around two hundred countries for an exciting two weeks of athletic events.
- However, our thesis states that hosting the Olympics has much bigger effects that go past just the two week period.
- Our project is focused on the underlying data of the Olympics over the past 100 years, as well as how it may have affected trends in tourism and medal count for the host country.



How does hosting the olympics affect tourism for the host country in terms of international visitors and total expenditures ?

Data Exploration and Cleanup

```
#Get the rows for the last 6 cities that hosted olympics
filtered_data = host_cities_df.iloc[45:]

#Get rid of unnecessary columns
last_six_hosts = filtered_data[['City', 'Country', 'Year']]
```

```
#Rename country in arrivals dataset if different from host cities dataset
arrivals_df = arrivals_df.replace('Russian Federation', 'Russia')

#Rename column to allow merging
arrivals_df = arrivals_df.rename(columns={'Country Name': 'Country'})
```

```
#Merge Datasets
combined_df = last_six_hosts.merge(arrivals_df, on='Country', how='left')

#Get rid of unnecessary columns
combined_df = combined_df.drop(['Country Code', 'Indicator Name', 'Indicator Code'], axis=1)

#Drop NaN columns
combined_df = combined_df.dropna(axis=1)
```

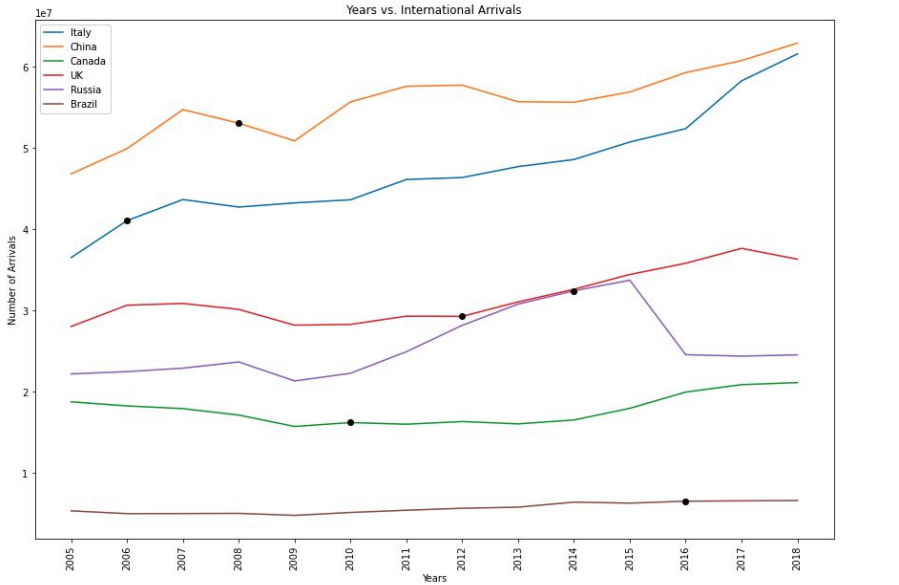
Compare the international arrivals of the host country prior to hosting the olympics and the year they host

Data Analysis

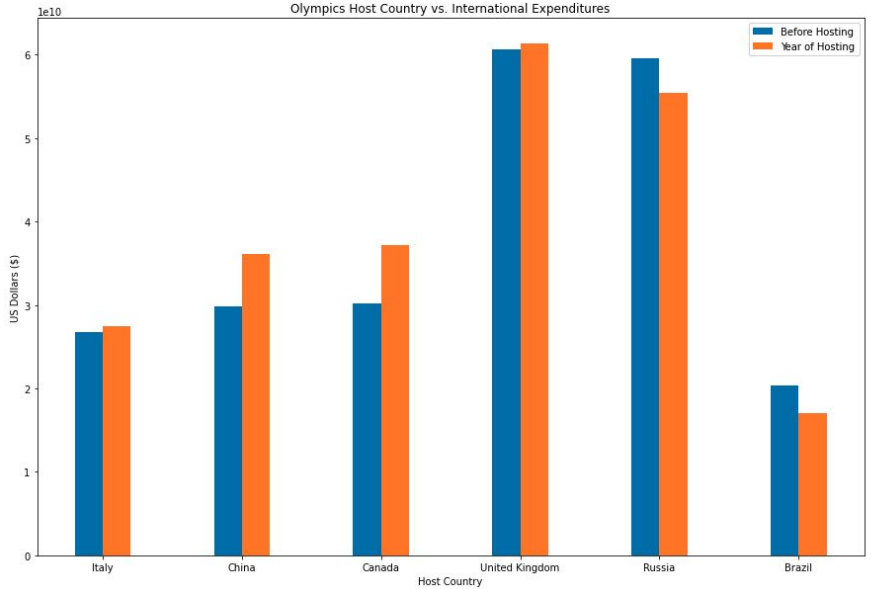
```
#Create a function that calculates percent change
def percent_change(value_1, value_2):
    return (value_2 - value_1)/value_1 * 100

#Create a new data frame that holds the host countries and the arrival data for the year prior to hosting and the year
arrival_change_df = pd.DataFrame({
    'Country': combined_df['Country'],
    'Host Year': combined_df['Year'],
    'Arrivals before Host Year': '',
    'Arrivals on Host Year': '',
    'Percent Change': ''

})

arrival_data_prior_host_year = [combined_df.iloc[0, 13], combined_df.iloc[1, 15], combined_df.iloc[2, 17], \
                                combined_df.iloc[3, 19], combined_df.iloc[4, 21], combined_df.iloc[5, 23]]

arrival_data_on_host_year = [combined_df.iloc[0, 14], combined_df.iloc[1, 16], combined_df.iloc[2, 18], \
                             combined_df.iloc[3, 20], combined_df.iloc[4, 22], combined_df.iloc[5, 24]]

arrival_change_df['Arrivals before Host Year'] = arrival_data_prior_host_year

arrival_change_df['Arrivals on Host Year'] = arrival_data_on_host_year

arrival_change_df['Percent Change'] = percent_change(arrival_change_df['Arrivals before Host Year'], arrival_change_df[

arrival_change_df
```

Insights and Conclusion

	Country	Host Year	Arrivals before Host Year	Arrivals on Host Year	Percent Change
0	Italy	2006	36513000.0	41058000.0	12.447621
1	China	2008	54720000.0	53049000.0	-3.053728
2	Canada	2010	15737000.0	16219000.0	3.062846
3	United Kingdom	2012	29306000.0	29282000.0	-0.081894
4	Russia	2014	30792000.0	32421000.0	5.290335
5	Brazil	2016	6306000.0	6547000.0	3.821757



	Country	Host Year	Expenditures before Host Year	Expenditures on Host Year	Percent Change
0	Italy	2006	2.676400e+10	2.744900e+10	2.559408
1	China	2008	2.978600e+10	3.615700e+10	21.389243
2	Canada	2010	3.022500e+10	3.722500e+10	23.159636
3	United Kingdom	2012	6.060800e+10	6.132300e+10	1.179712
4	Russia	2014	5.950400e+10	5.538300e+10	-6.925585
5	Brazil	2016	2.035600e+10	1.706800e+10	-16.152486



Does the Olympics Inspire Host Citizens to Travel?

(Timmy)

- A very quick description of exploration and cleanup process.

Starting point.

```
In [6]: #this allows me to focus on each city that hosted the games, for future merging
cities_only = olympics_country_df.drop_duplicates(subset = ['City'])
cities_only.head()
```

Out[6]:

	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sport	Event	Medal	Country
0	1	A Dijiang	M	24.0	180.0	80.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	NaN	Spain
1	2	A Lamusi	M	23.0	170.0	60.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightweight	NaN	United Kingdom
4	3	Gunnar Nielsen Aaby	M	24.0	NaN	NaN	Denmark	DEN	1920 Summer	1920	Summer	Antwerp	Football	Football Men's Football	NaN	Belgium
5	4	Edgar Lindenau Aabye	M	34.0	NaN	NaN	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-Of-War	Gold	France
7	5	Christine Jacoba Aaftink	F	21.0	185.0	82.0	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed Skating	Speed Skating Women's 500 metres	NaN	Canada

```
In [7]: #check if it's the correct number
len(cities_only)
```

Out[7]: 42

```
In [8]: #the only columns important to me
edited_cities_df = cities_only[["City", "Country","Year", "Season"]]
edited_cities_df = edited_cities_df.reset_index(drop=True)
```

- In the end everything worked out, but I could probably have used less steps.

Cleaning and narrowing down.

```
In [11]: #basically there is no data til 1995 for departures which is a bummer, so we are dropping coloumns
departures_df.drop(departures_df.iloc[:, 4:39], inplace = True, axis = 1)
departures_df.head()
```

Out[11]:

	Country Name	Country Code	Indicator Name	Indicator Code	1995	1996	1997	1998	1999	2000	...	2011	2012	2013	2014	2015	2016
0	Aruba	ABW	International tourism, number of departures	ST.INT.DPRT	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN
1	Afghanistan	AFG	International tourism, number of departures	ST.INT.DPRT	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN
2	Angola	AGO	International tourism, number of departures	ST.INT.DPRT	3000.0	3000.0	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN
3	Albania	ALB	International tourism, number of departures	ST.INT.DPRT	NaN	NaN	NaN	NaN	NaN	NaN	...	4120000.0	3959000.0	3928000.0	4146000.0	4504000.0	4852000.0
4	Andorra	AND	International tourism, number of departures	ST.INT.DPRT	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN

5 rows × 30 columns

```
In [12]: #rename coloumn for future merging
departures_df
departures_df = departures_df.rename(columns={'Country Name': 'Country'})
```

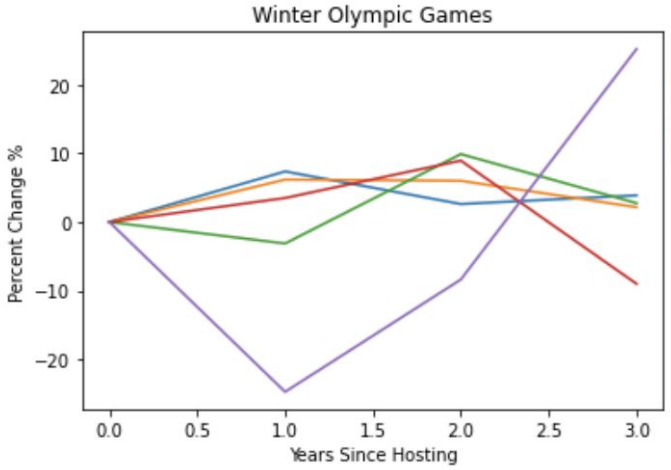
```
In [13]: #change country names in data so it matches
games_with_data = games_with_data.replace(['Russia', 'West Germany', 'South Korea', 'Soviet Union'],
                                          ['Russian Federation', 'Germany', 'Korea, Rep.', 'Russian Federation'])
```

```
In [14]: #reseting the index for finding specific rows
games_with_data = games_with_data.reset_index(drop=True)
```

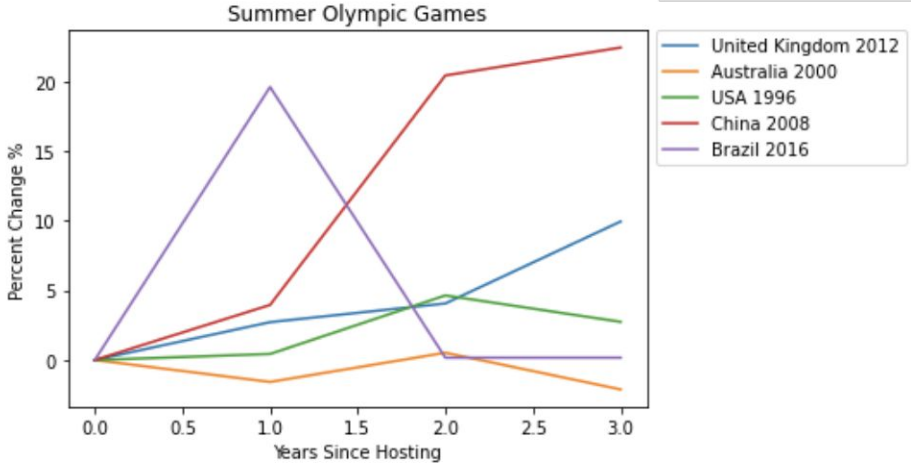
```
In [15]: #no data on Yugoslavia so bye
games_with_data = games_with_data.drop([16])
```

International Departures Post hosting the Olympics.

- The start date is the year each country hosted, the legend has the specific year.
- The reason for measuring the percent change was to be able to compare each timeline on one graph.
- Noteworthy aspect of my code are bbox_to_anchor, which basically says move the legend outside the graph. (Thanks Google.)
- Sadly, contrary to my thesis there seems to be no discernible pattern of increasing, or even decreasing rates for departures for the host citizens. (History is ever present).

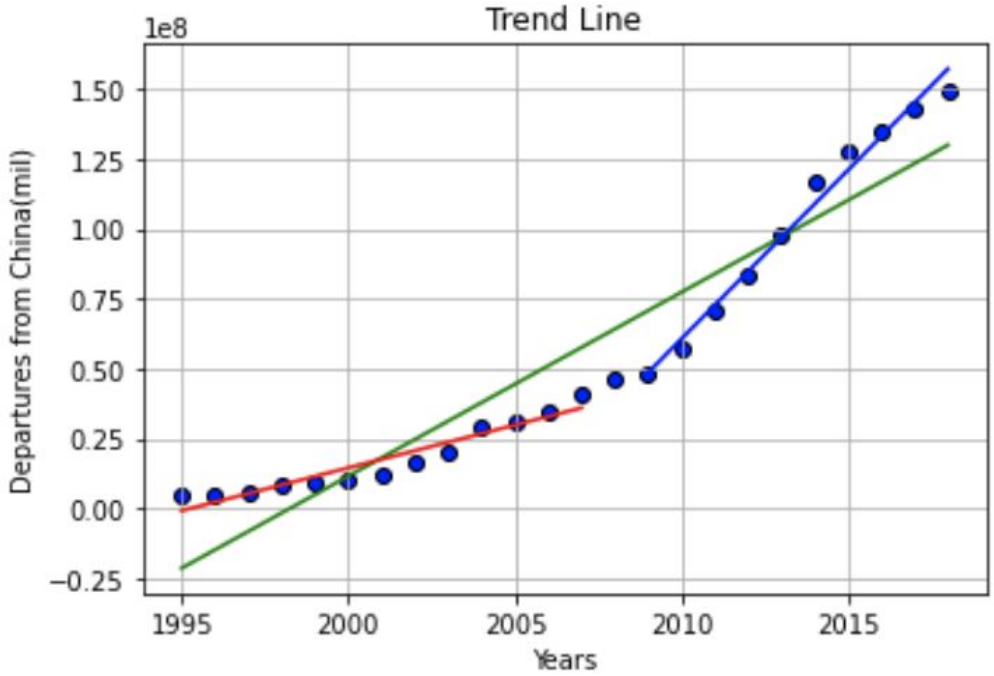



```
#pecercent change graph of when each country hosted the oylmpics
x_axis = range(len(uk_pct_hoststart))


plt.plot(x_axis, ita_pct_hoststart)
plt.plot(x_axis, can_pct_hoststart)
plt.plot(x_axis, usa_1_pct_hoststart)
plt.plot(x_axis, jap_pct_hoststart)
plt.plot(x_axis, rus_pct_hoststart)

plt.legend(['Italy 2006','Canada 2010','USA 2002','Japan 1998', 'Russia 2014'],
           loc='center left', bbox_to_anchor=(1, 0.82))

plt.title("Winter Olympic Games")
plt.ylabel("Percent Change %")
plt.xlabel("Years Since Hosting")
plt.savefig("graphs_Timmy/pct_change_winter")
plt.show()
```



Linear Regression for China

- The decision to focus on China was that they will host the 2022 Winter Olympics in a few months. It would be interesting to see if the linear regression will hold true. (Though probably not.)
- Additionally, their hosting of the Olympics is roughly in the middle of years we have data for, which paints the largest picture without hindrance.
- This graph shows there might be substantial evidence of how the Olympics can impact the rate of citizens traveling international!
- Post hosting the Olympics the rate increased 3.9 time compared to before.



Least and Most proud parts of my code.

Least.

In [24]:
```
#figureout how to do a for loop for this
# do pct chabfe for each and also fill in the first cell since it will be NA

#summer game
uk = for_graph.iloc[0]
uk_years = uk[1 : 25]
uk_pct_hoststart = uk[18: 22].pct_change() * 100
uk_pct_hoststart = uk_pct_hoststart.fillna(0)

#winter game
usa_1 = for_graph.iloc[1]
usa_1_years = usa_1[1 : 25]
usa_1_pct_hoststart = usa_1[8: 12].pct_change() * 100
usa_1_pct_hoststart = usa_1_pct_hoststart.fillna(0)

#summer game
aus = for_graph.iloc[2]
aus_years = aus[1 : 25]
aus_pct_hoststart = aus[6 : 10].pct_change() * 100
aus_pct_hoststart = aus_pct_hoststart.fillna(0)

#summer game
usa_2 = for_graph.iloc[3]
usa_2_years = usa_2[1 : 25]
usa_2_pct_hoststart = usa_2[2: 6].pct_change() * 100
usa_2_pct_hoststart = usa_2_pct_hoststart.fillna(0)

#winter game
rus = for_graph.iloc[4]
rus_years = rus[1 : 25]
rus_pct_hoststart = rus[20 : 24].pct_change() * 100
rus_pct_hoststart = rus_pct_hoststart.fillna(0)

#winter game
jap = for_graph.iloc[5]
jap_years = jap[1 : 25]
jap_pct_hoststart = jap[4 : 8].pct_change() * 100
jap_pct_hoststart = jap_pct_hoststart.fillna(0)

#winter game
ita = for_graph.iloc[6]
ita_years = ita[1 : 25]
ita_pct_hoststart = ita[12 : 16].pct_change() * 100
ita_pct_hoststart = ita_pct_hoststart.fillna(0)

#summer game
chi = for_graph.iloc[7]
chi_years = chi[1 : 25]
chi_pct_hoststart = chi[14 : 18].pct_change() * 100
chi_pct_hoststart = chi_pct_hoststart.fillna(0)
```

Most!

```
x_values = china_df['Total Years']
y_values = china_df['Total Departs']


plt.scatter(x_values, y_values, marker = "o", color = "blue",edgecolor = "black")

#overall
(slope, intercept, rvalue, pvalue, stderr) = linregress(x_values, y_values)
regress_values = x_values * slope + intercept
line_eq = "y = " + str(round(slope,2)) + "x + " + str(round(intercept,2))

#early years1
(slope1, intercept1, rvalue1, pvalue1, stderr1) = linregress(early_years, early_departs)
regress_values1 = early_years * slope1 + intercept1
line_eq1 = "y = " + str(round(slope,2)) + "x + " + str(round(intercept,2))

#later years 2
(slope2, intercept2, rvalue2, pvalue2, stderr2) = linregress(post_years, post_departs)
regress_values2 = post_years * slope2 + intercept2
line_eq2 = "y = " + str(round(slope,2)) + "x + " + str(round(intercept,2))


#plt.scatter(x_values,y_values)
plt.plot(x_values,regress_values,"g-")
plt.plot(early_years,regress_values1,"r-")
plt.plot(post_years,regress_values2,"b-")

plt.grid()
plt.xlabel("Years")
plt.ylabel("Departures from China(mil)")
plt.title("Trend Line")
plt.savefig("graphs_Timmy/China_rate_change")
plt.show()
```

Take it away Raj!

Is there a statistically significant difference between medals won prior to hosting olympics and after hosting the olympics?

Data clean-up

- Merging two dataframes to get host country names
- Determine which countries hosted the olympics and remove all the other teams from the list

```
host_team = olympics_country_df.loc[olympics_country_df["Team"] ==
olympics_country_df["Country"]]

only_teams_that_hosted =
olympics_country_df[olympics_country_df["Team"].isin(olympics_country_df["Country"])]
```

Data analysis

- Do medal counts for countries before and after they hosted the olympics and draw bar graphs (All of the olympics, only for summer olympics)

  Different color bar when it is the hosting year.

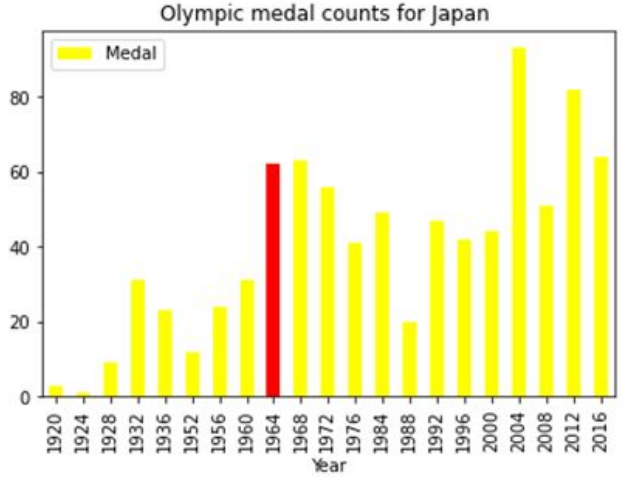
```
team = only_teams_that_hosted_summer["Team"].unique()
team = team.tolist()
team

for team in team:

    teams = year_medal_count_summer.loc[team]

    bar = host_year_medal_count_summer.loc[host_year_medal_count_summer["Team"] == team ]["Year"].tolist()

    colors = ["red" if x in bar else "yellow" for x in teams.index]

    teams.plot.bar(y="Medal", title=f"Olympic medal counts for {team} ", color=colors)

    plt.show()
```

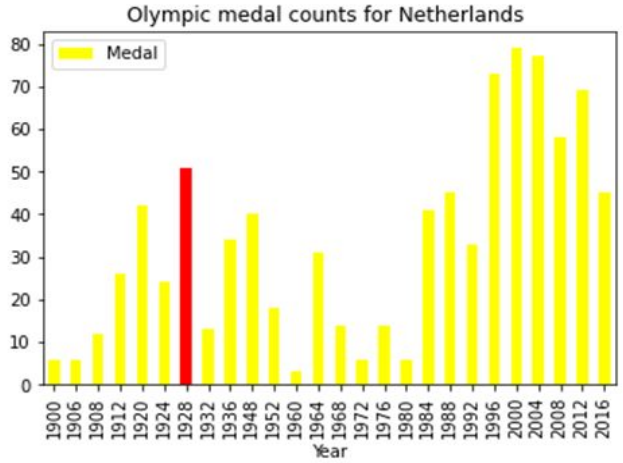
- Do hypothesis testing on the summer olympics data (Removing countries that hosted more than ones to keep the number of groups at 2 for each test)

H_0 : The mean of medal win for a country before and after hosting the Olympics is same

```
teams = []

for i in range(len(team_list_after)):


    test = stats.ttest_ind(team_list_after[i].Medal, team_list_before[i].Medal, equal_var=False)

    teamss = team_list_after[i]["Team"].unique().tolist()

    teams.append(teamss)


    print(f"Ttest result for {teams[i]}: {test}")
```

Summer Olympics Medal counts



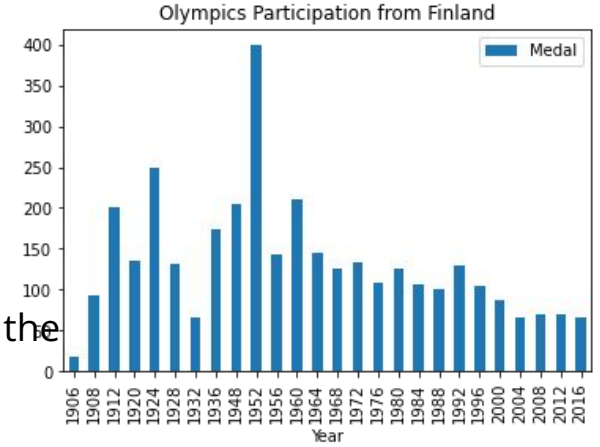


$p < 0.05$

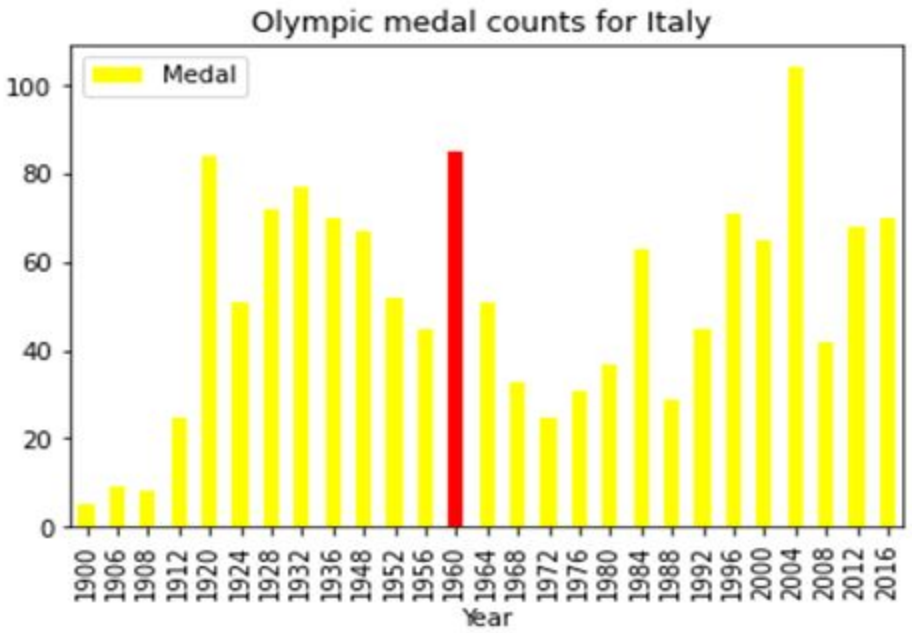
Increased medal counts after hosting the olympics

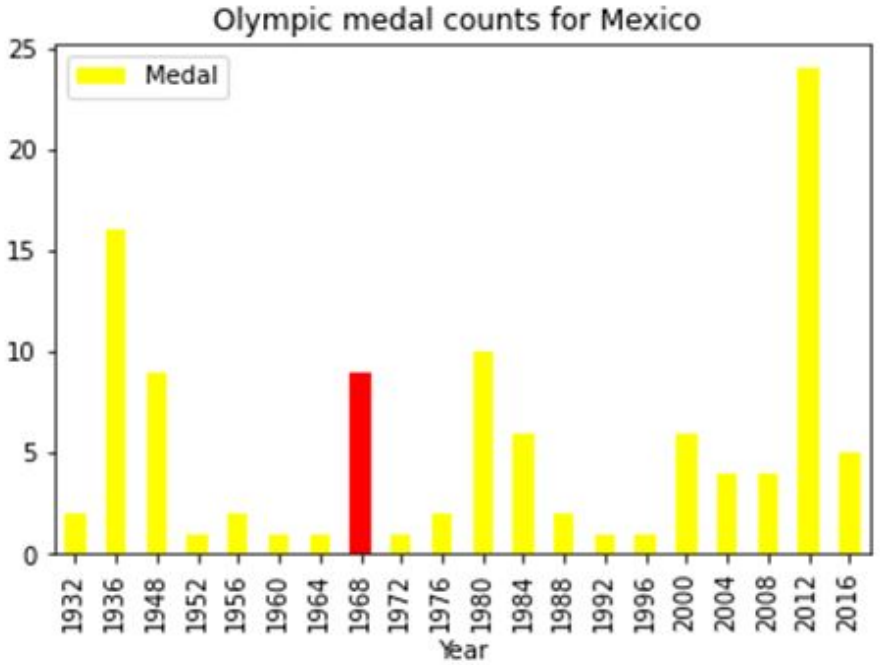
$p < 0.05$

Decreased medal counts after hosting the olympics



$P > 0.05$


Olympic medal counts for Italy
Medal
100
80
60
40
20
0
1900
1906
1908
1912
1920
1924
1928
1932
1936
1948
1952
1956
1960
1964
1968
1972
1976
1980
1984
1988
1992
1996
2000
2004
2008
2012
2016
Year


Olympic medal counts for Mexico
Medal
25
20
15
10
5
0
1932
1936
1948
1952
1956
1960
1964
1968
1972
1976
1980
1984
1988
1992
1996
2000
2004
2008
2012
2016
Year

$p > 0.05$

H_0 : The mean medal win for a country before and after hosting the Olympics is same

```
['Belgium']:      pvalue=0.03973552655429237 6
['Canada'] :      pvalue=0.00254534223708469 9
['China']:        pvalue=0.00215345425266248 6
['Finland']:      pvalue=0.00083888034248622 28
['Italy']:        pvalue=0.60541150053897
['Japan']:        pvalue=4.507169218578653e-05
['Mexico']:       pvalue=0.751101243592695 1
['Netherlands']:  pvalue=0.04807252085244786
['South Korea']:pvalue=0.00021286145825913618
['Spain']:        pvalue=0.00066711064773430 75
```

We were able to reject our H_0 for Belgium, Canada, China, Finland, Germany,Japan, Netherlands, South Korea and Spain. The decrease of Finland's Medal counts does not seem to be due to a decrease in participation.
We failed to reject the H_0 for Italy and Mexico.
We did not have enough data to perform the test on Brazil.

Conclusion and Open-ended Discussion Time

- Limitations in dataset might have affected the percent changes for international tourism and expenditures. Furthermore, external factors have an increased influence on the host cities.
- The Olympics might inspire some host citizens to internationally travel, or it could be more business related for the cause of increase of departures at least for China. In the end more research should be done.
- It looks like hosting the olympics has a positive effect on winning future games.

Thanks for Listening!