Introduction

Space missions analysis

The dataset consist of all space missions from 1957 to August 2022, including details on the location, date, and result of the launch, the company responsible, and the name, price, and status of the rocket used for the mission. It is downloaded from Maven Analytics.

Exploration of the universe is the dream of every scientist and to accomplish that dream space missions have taken place since a long time with some successful attempts and today I am here to analyse about space missions that had taken place in past to extract some valuable information.

Questions that I tried to answer in dataset

- How have rocket launches trended across time?
- Which countries have had the most successful space missions?
- Which rocket has been used for the most space missions? Is it still active?
- Success of mission in comparison to the failure of misson is more or less?
- Which company carried most of the space missions?

Data Cleaning and Preparation

In [1]:
```
import pandas as pd # to import data in form of table and to filter it
```

In [2]:
```
import numpy as np # to solve complex calculation if there's any
```

In [3]:
```
import os
os.listdir('/kaggle/input/space-missions')
```

Out[3]:
```
['space_missions.csv', 'space_missions_data_dictionary.csv']
```

In [4]:

```
space_missions_raw_df = pd.read_csv('/kaggle/input/space-missions/space_missi
ons.csv', encoding_errors='ignore')
space_missions_raw_df
```

Out[4]:

	Company	Location	Date	Time	Rocket	Mission	RocketStatus	Price
0	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-10-04	19:28:00	Sputnik 8K71PS	Sputnik-1	Retired	NaN
1	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-11-03	02:30:00	Sputnik 8K71PS	Sputnik-2	Retired	NaN
2	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1957-12-06	16:44:00	Vanguard	Vanguard TV3	Retired	NaN
3	AMBA	LC-26A, Cape Canaveral AFS, Florida, USA	1958-02-01	03:48:00	Juno I	Explorer 1	Retired	NaN
4	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1958-02-05	07:33:00	Vanguard	Vanguard TV3BU	Retired	NaN
...	...	...	...	...	...	...	...	...
4625	SpaceX	SLC-4E, Vandenberg SFB, California, USA	2022-07-22	17:39:00	Falcon 9 Block 5	Starlink Group 3-2	Active	67
4626	CASC	LC-101, Wenchang Satellite Launch Center, China	2022-07-24	06:22:00	Long March 5B	Wentian	Active	NaN
4627	SpaceX	LC-39A, Kennedy Space Center, Florida, USA	2022-07-24	13:38:00	Falcon 9 Block 5	Starlink Group 4-25	Active	67
4628	CAS Space	Jiuquan Satellite Launch Center, China	2022-07-27	04:12:00	Zhongke-1A	Demo Flight	Active	NaN
4629	CASC	LC-3, Xichang Satellite Launch Center, China	2022-07-29	13:28:00	Long March 2D	Yaogan 35 Group 03	Active	29.75

4630 rows × 9 columns

In [5]:
```
space_missions_raw_df.shape #Number of columns and rows in following dataset
```

Out[5]:
```
(4630, 9)
```

In [6]:
```
space_missions_raw_df.describe() # to know basic information about dataset
```

Out[6]:

	Company	Location	Date	Time	Rocket	Mission	RocketStatus	Price
count	4630	4630	4630	4503	4630	4630	4630	1265
unique	62	158	4180	1300	370	4556	2	65
top	RVSN USSR	Site 31/6, Baikonur Cosmodrome, Kazakhstan	1962-04-26	12:00:00	Cosmos-3M (11K65M)	DSP	Retired	450
freq	1777	251	4	52	446	8	3620	136

In [7]:

```
space_missions_raw_df.info() # to know non-null values and data types in whic
h data is stored
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4630 entries, 0 to 4629
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   Company        4630 non-null   object
 1   Location       4630 non-null   object
 2   Date           4630 non-null   object
 3   Time           4503 non-null   object
 4   Rocket         4630 non-null   object
 5   Mission        4630 non-null   object
 6   RocketStatus   4630 non-null   object
 7   Price          1265 non-null   object
 8   MissionStatus  4630 non-null   object
dtypes: object(9)
memory usage: 325.7+ KB
```

Since non-null count for **price** column is too much lower. Hence, it is clear that we cannot get any accurate information from it, due to which I am going to drop that column in further steps.

In [8]:

```
space_missions_raw_df.columns
```

Out[8]:

```
Index(['Company', 'Location', 'Date', 'Time', 'Rocket', 'Mission',
       'RocketStatus', 'Price', 'MissionStatus'],
      dtype='object')
```

In [9]:

```
space_missions_dict_raw_df = pd.read_csv('/kaggle/input/space-missions/space_missions_data_dictionary.csv', index_col = 'Field')
space_missions_dict_raw_df # description of the columns that are present in initial dataset
```

Out[9]:

	Description
Field	
Company	Company responsible for the space mission
Location	Location of the launch
Date	Date of the launch
Time	Time of the launch (UTC)
Rocket	Name of the rocket used for the mission
Mission	Name of the space mission (or missions)
RocketStatus	Status of the rocket as of August 2022 (Active...
Price	Cost of the rocket in millions of US dollars
MissionStatus	Status of the mission (Success, Failure, Parti...

In [10]:

```
space_missions_raw_df2 = space_missions_raw_df.copy() #to make copy of data t
o keep orignal data as it is
space_missions_df = space_missions_raw_df2.drop(columns = ['Price']) #to make
dataset without price column
space_missions_df.drop(columns = ['Time'], inplace = True) #to make dataset w
ithout time column
space_missions_df
```

Out[10]:

	Company	Location	Date	Rocket	Mission	RocketStatus	MissionStatus
0	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-10-04	Sputnik 8K71PS	Sputnik-1	Retired	Success
1	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-11-03	Sputnik 8K71PS	Sputnik-2	Retired	Success
2	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1957-12-06	Vanguard	Vanguard TV3	Retired	Failure
3	AMBA	LC-26A, Cape Canaveral AFS, Florida, USA	1958-02-01	Juno I	Explorer 1	Retired	Success
4	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1958-02-05	Vanguard	Vanguard TV3BU	Retired	Failure
...	...	...	...	...	...	...	...
4625	SpaceX	SLC-4E, Vandenberg SFB, California, USA	2022-07-22	Falcon 9 Block 5	Starlink Group 3-2	Active	Success
4626	CASC	LC-101, Wenchang Satellite Launch Center, China	2022-07-24	Long March 5B	Wentian	Active	Success
4627	SpaceX	LC-39A, Kennedy Space Center, Florida, USA	2022-07-24	Falcon 9 Block 5	Starlink Group 4-25	Active	Success
4628	CAS Space	Jiuquan Satellite Launch Center, China	2022-07-27	Zhongke-1A	Demo Flight	Active	Success
4629	CASC	LC-3, Xichang Satellite Launch Center, China	2022-07-29	Long March 2D	Yaogan 35 Group 03	Active	Success

4630 rows × 7 columns

In [11]:
```
space_missions_df['RocketStatus'].value_counts() #to check if there's any exc
eptional value in dataset
```

Out[11]:
```
Retired    3620
Active     1010
Name: RocketStatus, dtype: int64
```

In [12]:
```
space_missions_df['MissionStatus'].value_counts() #to check if there's any ex
ceptional value in dataset
```

Out[12]:
```
Success              4162
Failure               357
Partial Failure       107
Prelaunch Failure       4
Name: MissionStatus, dtype: int64
```

Data Visualization

In [13]:
```
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns #to visualize the data
```

Trend of rocket launch and about success

To do so first we have to sort data on yearly basis.

In [14]:

```
space_missions_yearly_df = space_missions_df.copy() #to keep orignal data unt
ouched
space_missions_yearly_df['year'] = pd.DatetimeIndex(space_missions_yearly_df.
Date).year #to add a new column of year
space_missions_yearly_df.head(5) #to check if the changes take place
```

Out[14]:

	Company	Location	Date	Rocket	Mission	RocketStatus	MissionStatus	year
0	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-10-04	Sputnik 8K71PS	Sputnik-1	Retired	Success	1957
1	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-11-03	Sputnik 8K71PS	Sputnik-2	Retired	Success	1957
2	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1957-12-06	Vanguard	Vanguard TV3	Retired	Failure	1957
3	AMBA	LC-26A, Cape Canaveral AFS, Florida, USA	1958-02-01	Juno I	Explorer 1	Retired	Success	1958
4	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1958-02-05	Vanguard	Vanguard TV3BU	Retired	Failure	1958

In [15]:

```
space_missions_yearly_df.value_counts(space_missions_yearly_df.year,ascending
= True) #to count total number of missions
```

Out[15]:

```
year
1957      3
1959     20
1958     28
2005     37
2010     37
        ...
1977    114
2018    117
2020    119
1971    119
2021    157
Length: 66, dtype: int64
```

In [16]:

```
#function to make seperate column for particluar value
def split_multicolumn(col_series):
    result_df = col_series.to_frame()
    options = []
    # Iterate over the column
    for idx, value in col_series[col_series.notnull()].iteritems():
        # Break each value into list of options
        for option in value.split(';'):
            # Add the option as a column to result
            if not option in result_df.columns:
                options.append(option)
                result_df[option] = False
            # Mark the value in the option column as True
            result_df.at[idx, option] = True
    return result_df[options]
```

In [17]:

```
success_yearly_df = split_multicolumn(space_missions_yearly_df.MissionStatus)
success_yearly_df['year'] = space_missions_yearly_df.year #to add year column for easy understanding
success_yearly_df.head(5)
```

Out[17]:

	Success	Failure	Partial Failure	Prelaunch Failure	year
0	True	False	False	False	1957
1	True	False	False	False	1957
2	False	True	False	False	1957
3	True	False	False	False	1958
4	False	True	False	False	1958

In [18]:

```
success_yearly2_df = success_yearly_df.groupby('year')[['Success']].sum() #grouping for future visualization
success_yearly2_df
```

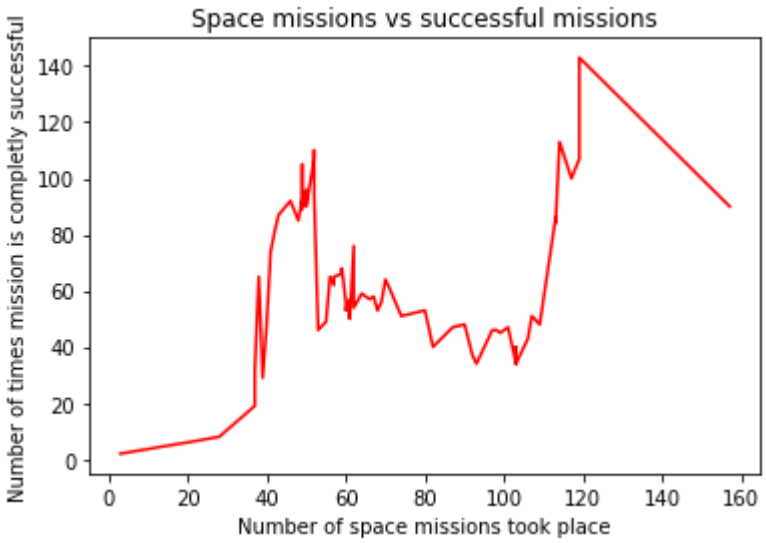
Out[18]:

	Success
year	
1957	2
1958	6
1959	8
1960	19
1961	32
…	…
2018	113
2019	100
2020	107
2021	143
2022	90

66 rows × 1 columns

In [19]:

```
plt.title('Space missions vs successful missions') #for creating title
plt.plot(space_missions_yearly_df.value_counts(space_missions_yearly_df.year,
ascending = True), success_yearly2_df.Success,'-r') #for creating line chart
plt.xlabel('Number of space missions took place') #for labeling x-axis
plt.ylabel('Number of times mission is completly successful'); #for labeling
y-axis
sns.set_style('whitegrid')
```



Country with most successful missions

In [20]:

```
mission_location_df = space_missions_df.copy() #to keep orignal data untouche
d
mission_location_df['Country'] = mission_location_df['Location'].str.split
(',').apply(lambda x: x[-1]) #to split and make new column of country
mission_location_df['year'] = pd.DatetimeIndex(mission_location_df.Date).year
mission_location_df.head(5)
```

Out[20]:

	Company	Location	Date	Rocket	Mission	RocketStatus	MissionStatus	Count
0	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-10-04	Sputnik 8K71PS	Sputnik-1	Retired	Success	Kazak
1	RVSN USSR	Site 1/5, Baikonur Cosmodrome, Kazakhstan	1957-11-03	Sputnik 8K71PS	Sputnik-2	Retired	Success	Kazak
2	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1957-12-06	Vanguard	Vanguard TV3	Retired	Failure	USA
3	AMBA	LC-26A, Cape Canaveral AFS, Florida, USA	1958-02-01	Juno I	Explorer 1	Retired	Success	USA
4	US Navy	LC-18A, Cape Canaveral AFS, Florida, USA	1958-02-05	Vanguard	Vanguard TV3BU	Retired	Failure	USA

In [21]:

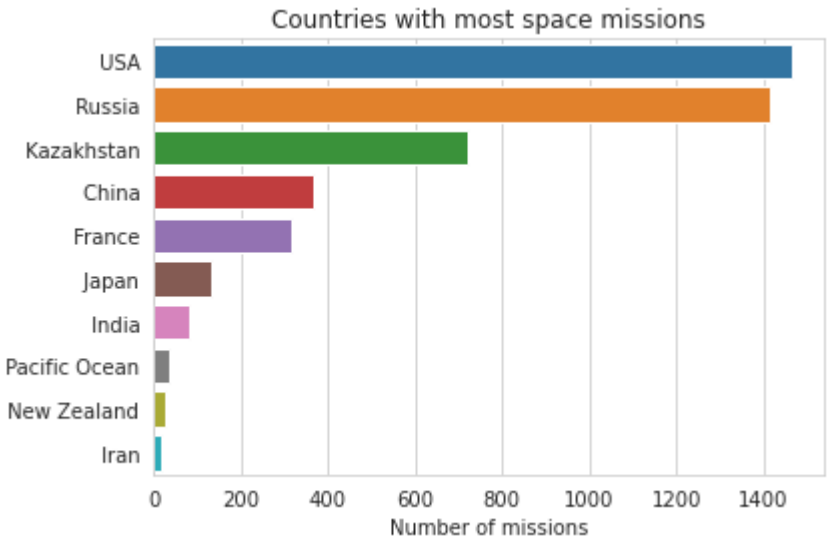
```
country_count = mission_location_df.Country.value_counts().head(10) #to count number of missions each country have done
country_count
```

Out[21]:

```
USA             1467
Russia          1416
Kazakhstan       719
China            365
France           318
Japan            130
India             82
Pacific Ocean     36
New Zealand       28
Iran              16
Name: Country, dtype: int64
```

In [22]:

```
sns.barplot(x = country_count, y = country_count.index) #to make bar graph
plt.title('Countries with most space missions') #to create title
plt.xlabel('Number of missions'); #to label x-axis
```



In [23]:

```
mission_location_df[['Country','MissionStatus']].value_counts().head(10) #to know count of successful missions
```

Out[23]:

```
Country      MissionStatus
 Russia      Success          1323
 USA         Success          1298
 Kazakhstan  Success           625
 China       Success           335
 France      Success           299
 USA         Failure           134
 Japan       Success           117
 Kazakhstan  Failure            72
 India       Success            68
 Russia      Failure            63
dtype: int64
```

As it is clear from the above **Russia** have the most successful mission followed by **USA**.

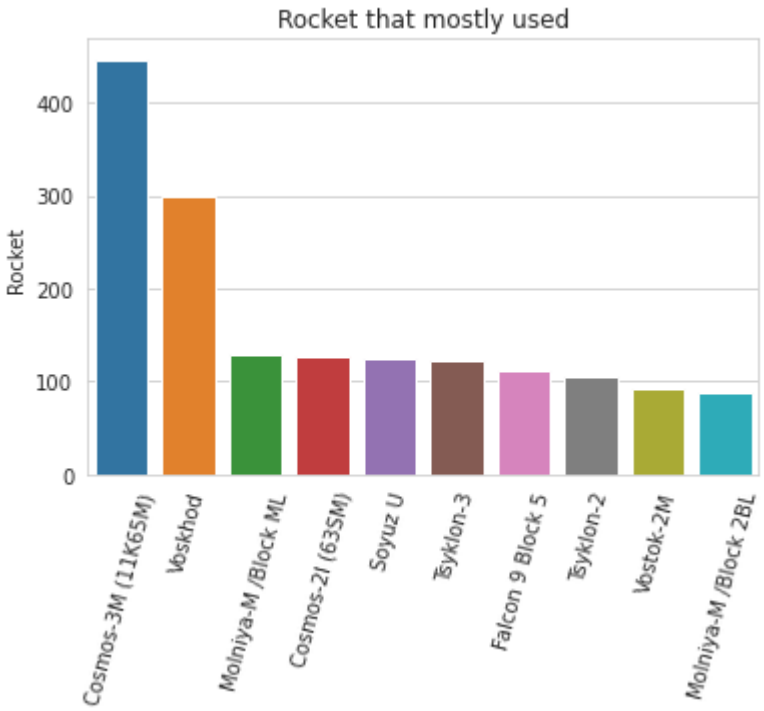
Space rocket that mostly used

In [24]:
```
rockets_df = space_missions_df.Rocket.value_counts().head(10) #to know count
of rockets used in missions
rockets_df
```

Out[24]:
```
Cosmos-3M (11K65M)      446
Voskhod                 299
Molniya-M /Block ML     128
Cosmos-2I (63SM)        126
Soyuz U                 125
Tsyklon-3               122
Falcon 9 Block 5        111
Tsyklon-2               106
Vostok-2M                93
Molniya-M /Block 2BL     87
Name: Rocket, dtype: int64
```

In [25]:

```
plt.title('Rocket that mostly used') #to create title
sns.barplot(x = rockets_df.index, y = rockets_df) #to create bar graph
plt.xticks(rotation = 75); #to rotate x-axis labels for clear visualization
```



In [26]:

```
space_missions_df[['Rocket','RocketStatus']].value_counts().head(10) #to know status of mostly used rockets
```

Out[26]:

Rocket	RocketStatus	
Cosmos-3M (11K65M)	Retired	446
Voskhod	Retired	299
Molniya-M /Block ML	Retired	128
Cosmos-2I (63SM)	Retired	126
Soyuz U	Retired	125
Tsyklon-3	Retired	122
Falcon 9 Block 5	Active	111
Tsyklon-2	Retired	106
Vostok-2M	Retired	93
Molniya-M /Block 2BL	Retired	87

dtype: int64

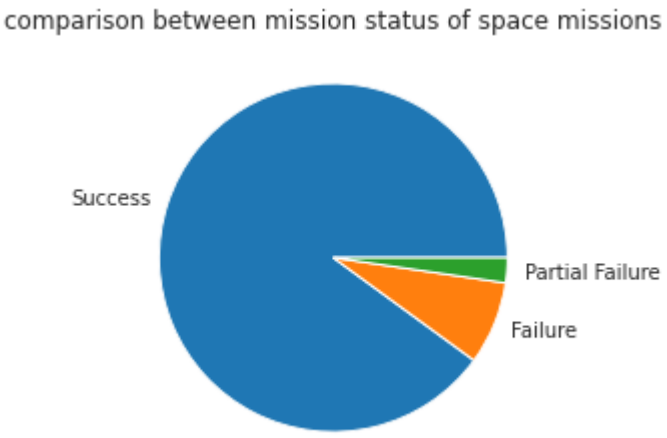
As it can be clearly seen that the mostly used rocket **Cosmos-3M (11K65M)** is **no longer** in service.

Success with failure comparison

In [27]:
```
space_missions_df.MissionStatus.value_counts() #to count mission status
```

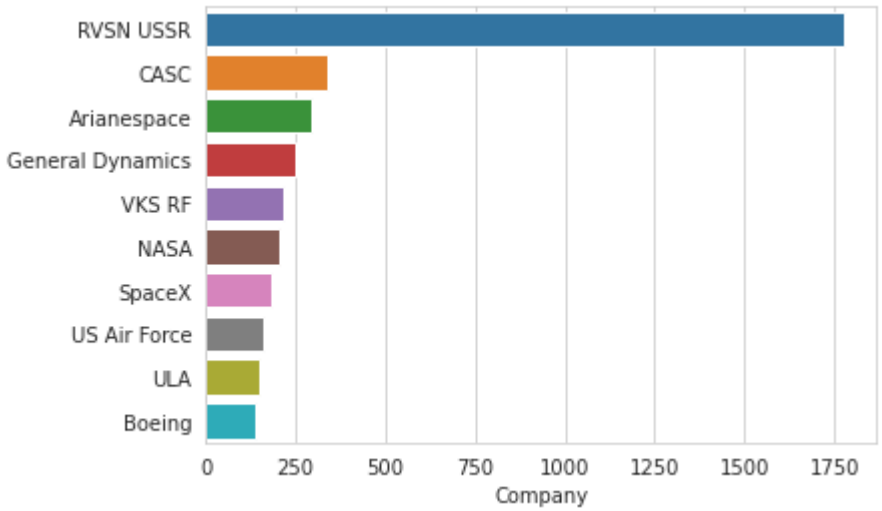
Out[27]:
```
Success              4162
Failure               357
Partial Failure       107
Prelaunch Failure       4
Name: MissionStatus, dtype: int64
```

In [28]:
```
plt.title('comparison between mission status of space missions') #for creatin
g title
plt.pie(space_missions_df.MissionStatus.value_counts().head(3),labels=['Succe
ss','Failure','Partial Failure']); #to create pie chart
```



Company with most missions

In [29]:
```
sns.barplot(x = space_missions_df.Company.value_counts().head(10),
            y = space_missions_df.Company.value_counts().head(10).index); #to
create bar chart for the companies
```



Conslusion

From the processing of data and visualisation of data there are some key points that are extracted.

- With time number of space missions increased but there's no uniform growth in chances of success.
- despite most space missions carried by **USA** throughout the time but **Russia** have more successful missions.
- **Cosmos-3M (11K65M)** is mostly used rocket but is now retired from the services.
- All the missions that have taken place since now **success** of the missions are **far more** than **failure**.
- **RVSN USSR** clearly is dominating company for most of the space missions that are carried.

In [ ]: