

Space Missions Analysis_Group 2.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share Settings

+ Code + Text

RAM Disk Editing

Space Missions Analysis

Importing Libraries

```
[1] import pandas as pd #pandas is a data manipulation and analysis tool
import numpy as np #numpy is the library for numerical computation
import seaborn as sns #seaborn is a library for making statistical graphics
import matplotlib.pyplot as plt #Matplotlib is a plotting library
import plotly.express as px #plotly.express provides consistent and interactive figures
```

[2] #Read the csv file

```
df=pd.read_csv('/content/drive/MyDrive/Space_Corrected.csv')
df.head()
```

	Unnamed: 0	Unnamed: 0.1	Company Name	Location	Datum	Detail	Status Rocket	Rocket	Status Mission
0	0	0	SpaceX	LC-39A, Kennedy Space Center, Florida, USA	Fri Aug 07, 2020 05:12 UTC	Falcon 9 Block 5 Starlink V1 L9 & BlackSky	StatusActive	50.0	Success
1	1	1	CASC	Site 9401 (SLS-2), Jiuquan Satellite Launch Ce...	Thu Aug 06, 2020 04:01 UTC	Long March 2D Gaofen-9 04 & Q-SAT	StatusActive	29.75	Success
2	2	2	SpaceX	Pad A, Boca Chica, Texas, USA	Tue Aug 04, 2020 23:57 UTC	Starship Prototype 150 Meter Hop	StatusActive	NaN	Success
3	3	3	Roscosmos	Site 200/39, Baikonur Cosmodrome, Kazakhstan	Thu Jul 30, 2020 21:25 UTC	Proton-M/Briz-M Ekspress-80 & Ekspress-103	StatusActive	65.0	Success
4	4	4	ULA	SLC-41, Cape Canaveral AFS, Florida, USA	Thu Jul 30, 2020 11:50 UTC	Atlas V 541 Perseverance	StatusActive	145.0	Success

Preprocessing

Information of the dataframe

```
[3] # to find the Information of the dataframe#
print("The information of the space dataframe is")
print(df.info())
```

The information of the space dataframe is

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4324 entries, 0 to 4323
Data columns (total 9 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   Unnamed: 0    4324 non-null   int64  
 1   Unnamed: 0.1  4324 non-null   int64  
 2   Company Name 4324 non-null   object  
 3   Location     4324 non-null   object  
 4   Datum        4324 non-null   object  
 5   Detail        4324 non-null   object  
 6   Status Rocket 4324 non-null   object  
 7   Rocket        964 non-null   object  
 8   Status Mission 4324 non-null   object  
dtypes: int64(2), object(7)
memory usage: 304.2+ KB
None
```

rows and columns of the dataframe

```
[4] # to find the rows and columns of the dataframe#
print("The number of rows and columns are")
print(df.shape)
```

The number of rows and columns are

```
(4324, 9)
```

columns present in dataframe

```
[5] #To get the columns present in dataframe#
print("The columns present in dataframe is")
print(df.columns)
```

The columns present in dataframe is

```
Index(['Unnamed: 0', 'Unnamed: 0.1', 'Company Name', 'Location', 'Datum',
       'Detail', 'Status Rocket', 'Rocket', 'Status Mission'],
      dtype='object')
```

```
[6] df.drop('Unnamed: 0.1',axis=1,inplace=True) # Dropping Unnamed: 0.1 column since it does not provide any information
df.head()
```

	Unnamed: 0	Company Name	Location	Datum	Detail	Status Rocket	Rocket	Status Mission
0	0	SpaceX	LC-39A, Kennedy Space Center, Florida, USA	Fri Aug 07, 2020 05:12 UTC	Falcon 9 Block 5 Starlink V1 L9 & BlackSky	StatusActive	50.0	Success
1	1	CASC	Site 9401 (SLS-2), Jiuquan Satellite Launch Ce...	Thu Aug 06, 2020 04:01 UTC	Long March 2D Gaofen-9 04 & Q-SAT	StatusActive	29.75	Success
2	2	SpaceX	Pad A, Boca Chica, Texas, USA	Tue Aug 04, 2020 23:57 UTC	Starship Prototype 150 Meter Hop	StatusActive	NaN	Success

3	3	Roscosmos	Site 200/39, Baikonur Cosmodrome, Kazakhstan	Thu Jul 30, 2020 21:25 UTC	Proton-M/Briz-M Ekspress-80 & Ekspress-103	StatusActive	65.0	Success
4	4	ULA	SLC-41, Cape Canaveral AFS, Florida, USA	Thu Jul 30, 2020 11:50 UTC	Atlas V 541 Perseverance	StatusActive	145.0	Success

```
[7]: df.drop('Unnamed: 0',axis=1,inplace=True) # Dropping Unnamed: 0 column since it does not provide any information
df.head()
```

Company	Name	Location	Datum	Detail	Status	Rocket	Rocket	Status	Mission
0	SpaceX	LC-39A, Kennedy Space Center, Florida, USA	Fri Aug 07, 2020 05:12 UTC	Falcon 9 Block 5 Starlink V1 L9 & BlackSky	StatusActive	50.0	Success		
1	CASC	Site 9401 (SLS-2), Jiuquan Satellite Launch Ce...	Thu Aug 06, 2020 04:01 UTC	Long March 2D Gaofen-9 04 & Q-SAT	StatusActive	29.75	Success		
2	SpaceX	Pad A, Boca Chica, Texas, USA	Tue Aug 04, 2020 23:57 UTC	Starship Prototype 150 Meter Hop	StatusActive	NaN	Success		
3	Roscosmos	Site 200/39, Baikonur Cosmodrome, Kazakhstan	Thu Jul 30, 2020 21:25 UTC	Proton-M/Briz-M Ekspress-80 & Ekspress-103	StatusActive	65.0	Success		
4	ULA	SLC-41, Cape Canaveral AFS, Florida, USA	Thu Jul 30, 2020 11:50 UTC	Atlas V 541 Perseverance	StatusActive	145.0	Success		

▼ Null values in the dataframe

```
[8]: # To find any null values in the dataframe#
df.isna().any()
```

```
Company Name    False
Location        False
Datum           False
Detail          False
Status Rocket   False
Rocket          True
Status Mission  False
dtype: bool
```

▼ Number of null values in Rocket column of the dataframe

```
[9]: # To find the number of null values in Rocket column of the dataframe
df['Rocket'].isna().value_counts()
```

```
True    3360
False   964
Name: Rocket, dtype: int64
```

```
[10]: df.drop('Rocket',axis=1,inplace=True) # Dropping Rocket since it contains lots of null values
df.head()
```

Company	Name	Location	Datum	Detail	Status	Rocket	Status	Mission
0	SpaceX	LC-39A, Kennedy Space Center, Florida, USA	Fri Aug 07, 2020 05:12 UTC	Falcon 9 Block 5 Starlink V1 L9 & BlackSky	StatusActive		Success	
1	CASC	Site 9401 (SLS-2), Jiuquan Satellite Launch Ce...	Thu Aug 06, 2020 04:01 UTC	Long March 2D Gaofen-9 04 & Q-SAT	StatusActive		Success	
2	SpaceX	Pad A, Boca Chica, Texas, USA	Tue Aug 04, 2020 23:57 UTC	Starship Prototype 150 Meter Hop	StatusActive		Success	
3	Roscosmos	Site 200/39, Baikonur Cosmodrome, Kazakhstan	Thu Jul 30, 2020 21:25 UTC	Proton-M/Briz-M Ekspress-80 & Ekspress-103	StatusActive		Success	
4	ULA	SLC-41, Cape Canaveral AFS, Florida, USA	Thu Jul 30, 2020 11:50 UTC	Atlas V 541 Perseverance	StatusActive		Success	

▼ Datatype of all the columns in the dataframe

```
[11]: # To find the datatype of all the columns in the dataframe
df.dtypes
```

```
Company Name    object
Location        object
Datum           object
Detail          object
Status Rocket   object
Status Mission  object
dtype: object
```

▼ Remove the word Status in Status Rocket column Values

```
[12]: # To change the Status Rocket column values from StatusActive to Active by removing the word Status
df['Status Rocket']=df['Status Rocket'].str.replace('Status','')
print(df['Status Rocket'])
```

```
0      Active
1      Active
2      Active
3      Active
4      Active
...
4319    Retired
4320    Retired
4321    Retired
4322    Retired
4323    Retired
Name: Status Rocket, Length: 4324, dtype: object
```

▼ Visualizations

▼ 1. Current Status of the Rockets

```
[13]: #To draw the category plot of Status rocket column
```

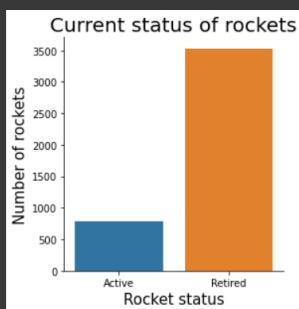
```

sns.catplot('status Rocket',kind='count',data=df,height=4)
plt.title('Current status of rockets',size=20)
plt.xlabel('Rocket status',size=15)
plt.ylabel('Number of rockets',size=15)
plt.show()

/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning:

```

Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error.



Conclusion: Most of the rockets are retired

▼ 2. Mission Status

```

[14] #to get the count of each mission status
data = df['Status Mission'].value_counts().reset_index()
data.columns = ['Status Mission', 'Number of Rockets']
print(data)

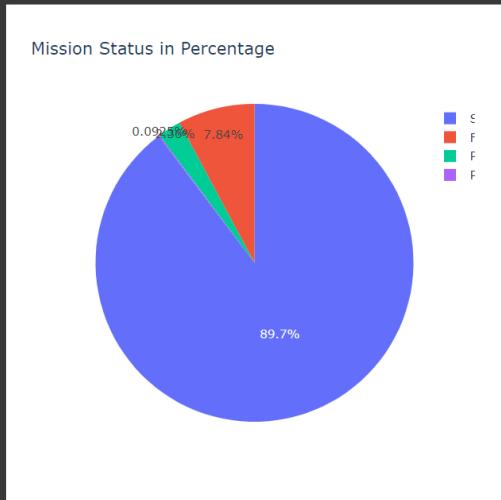
```

	Status Mission	Number of Rockets
0	Success	3879
1	Failure	339
2	Partial Failure	102
3	Prelaunch Failure	4

```

[15] #plotting a pie graph to show the mission status
fig = px.pie(data, values='Number of Rockets', names='Status Mission', title='Mission Status in Percentage',height=500,width=500)
fig.show()

```



Conclusion: Majority of the status mission is success while the minimum is prelaunch failure

▼ 3. Company

```

[16] #To get the number of rockets owned per company
data = df['Company Name'].value_counts().reset_index()
data.columns = ['Company Name', 'Number of Rockets']
print(data)

```

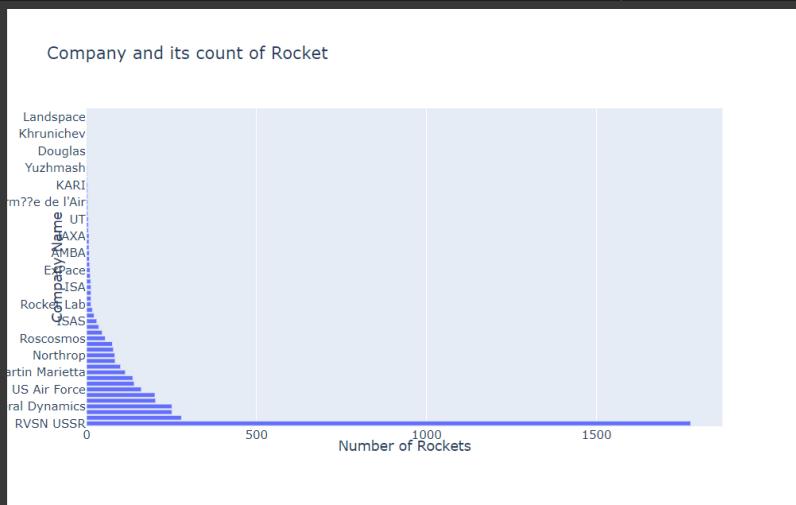
	Company Name	Number of Rockets
0	RVSN USSR	1777
1	Arianespace	279
2	CASC	251
3	General Dynamics	251
4	NASA	203
5	VKS RF	201
6	US Air Force	161
7	ULA	140
8	Boeing	136
9	Martin Marietta	114
10	SpaceX	100
11	MHI	84
12	Northrop	83
13	Lockheed	79
14	ISRO	76
15	Roscosmos	55
16	SAE	45

```

16      IES          46
17  Sea Launch       36
18      ISAS          30
19   Kosmotras        22
20     US Navy         17
21  Rocket Lab         13
22      ESA           13
23  Eurockot           13
24      ISA           13
25  Blue Origin         12
26      IAI            11
27  ExPace             10
28      ASI              9
29      CNES             8
30      AMBA             8
31  Land Launch          7
32      MITT             7
33      JAXA             7
34      CASIC            5
35      KCST              5
36      UT                5
37  CECLES              4
38      Exos              4
39  Arm??e de l'Air          4
40      AEB              3
41      SRC              3
42      KARI              3
43      RAE                2
44  OKB-586              2
45  Yuzhmarsh             2
46      Sandia             1
47      IRGC              1
48      Douglas             1
49  Starsem              1
50  Virgin Orbit             1
51  Khrunichev             1
52      EER              1
53  i-Space              1
54  Landspace              1
55  OneSpace              1

```

```
[17] #plotting the bar graph for company
fig = px.bar(data, x='Number of Rockets', y='Company Name', orientation='h', title='Company and its count of Rocket',height=500,width=800)
fig.show()
```



Conclusion: RVSN USSR owns the highest number of rockets

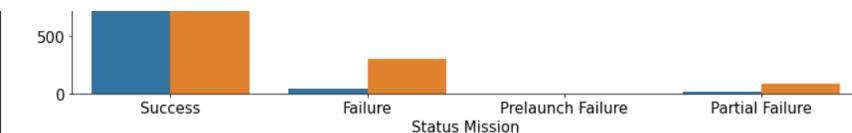
▼ 4. Status Rocket

```
[18] #counting the status mission with respect to active and retired rockets
sns.catplot('Status Mission',kind='count',data=df,hue='Status Rocket',height=6,aspect=2)
plt.xticks(size=15)
plt.xlabel('Status Mission',size=15)
plt.yticks(size=15)
plt.ylabel('Number of Rockets',size=15)
plt.title('Status Mission of active and retired Rockets',size=15)
plt.show()
```

/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning:

Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments without an explicit keyword will result





Conclusion: Retired rockets has highest success rate

▼ 5. Success Rocket Analysis

```
[19] #success rocket analysis count
success=df[df['Status Mission']=='Success']

data = success['Company Name'].value_counts().reset_index()

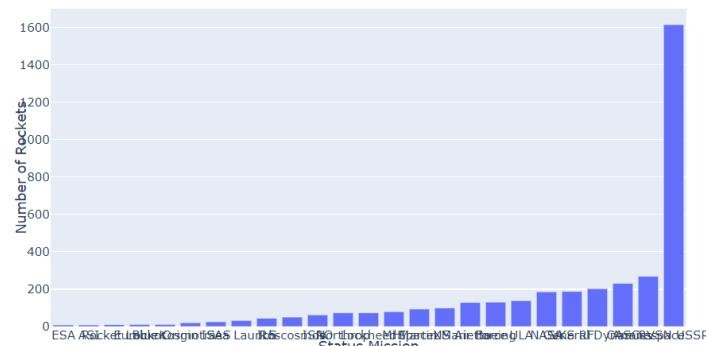
data.columns = [
    'Status Mission',
    'Number of Rockets'
]

data = data.sort_values('Number of Rockets')
print(data)
```

Status Mission	Number of Rockets
KARI	1
i-Space	1
Douglas	1
RAE	1
UT	1
Starsem	1
Khrunichev	1
CECLES	1
IRGC	1
SRC	2
OKB-586	2
KCST	2
US Navy	2
Yuzhnyash	2
Armée de l'Air	3
CASIC	3
ISA	4
AMBA	4
MITT	6
CNES	6
Land Launch	6
JAXA	6
ExPace	9
IAI	9
ESA	9
ASTI	9
Rocket Lab	11
Eurockot	12
Blue Origin	12
Kosmotras	21
ISAS	26
Sea Launch	33
ILS	45
Roscosmos	51
ISRO	63
Northrop	74
Lockheed	74
MHI	80
SpaceX	94
Martin Marietta	100
US Air Force	129
Boeing	131
ULA	139
NASA	186
VKS RF	188
General Dynamics	203
CASC	231
Arianespace	269
RVSN USSR	1614

```
[20] #success rocket analysis visualization
fig = px.bar(data.tail(25),y='Number of Rockets', x='Status Mission', orientation='v', title='Company with highest Success Rockets',height=500,width=800)
fig.show()
```

Company with highest Success Rockets

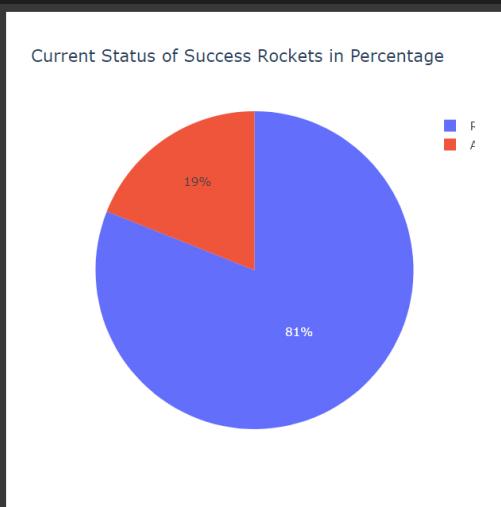


Conclusion: RVSN USSR has highest success rate

```
[21] #To get the current status of successful rockets
data = success['Status Rocket'].value_counts().reset_index()
data.columns = ['Status Rocket', 'Number of Rockets']
print(data)
```

Status Rocket	Number of Rockets
Retired	3143
Active	736

```
[22] #To plot the pie chart for the current status of successful rockets
fig = px.pie(data, values='Number of Rockets', names='Status Rocket', title='Current Status of Success Rockets in Percentage', height=500, width=500)
fig.show()
```



Conclusion: Retired rockets have had highest success while the active rockets have had lesser success

▼ 6. Failure Rocket Analysis

```
[23] #Failure rocket analysis count
Failure=df[df['Status Mission']=='Failure']

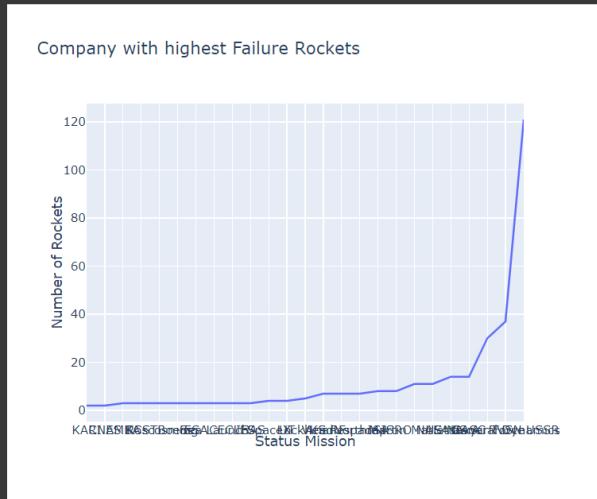
data = Failure['Company Name'].value_counts().reset_index()

data.columns = [
    'Status Mission',
    'Number of Rockets'
]

data = data.sort_values('Number of Rockets')
print(data)
```

	Status Mission	Number of Rockets
42	ExPace	1
30	Sandia	1
32	OneSpace	1
33	Virgin Orbit	1
34	Kosmotras	1
35	Eurockot	1
31	EER	1
37	Exos	1
38	JAXA	1
39	MITT	1
40	Landspace	1
36	SRC	1
41	RAE	1
29	AEB	2
28	MHI	2
27	Rocket Lab	2
26	CASIC	2
25	IAI	2
24	KARI	2
23	CNES	2
22	AMBA	3
21	KCST	3
19	Roscosmos	3
18	Boeing	3
17	ESA	3
16	Sea Launch	3
15	CECLES	3
20	ISAS	3
14	SpaceX	4
13	UT	4
12	Lockheed	5
9	VKS RF	7
11	Arianespace	7
10	Northrop	7
8	ISA	8
7	ISRO	8
6	Martin Marietta	11
5	NASA	11
4	US Navy	14
3	CASC	14
2	US Air Force	30
1	General Dynamics	37
0	AVCO-ML	121

```
[24] # Visualization of failure rocket using scatter
fig = px.line(data.tail(25),y='Number of Rockets', x='Status Mission', title='Company with highest Failure Rockets',height=500,width=600)
fig.show()
```

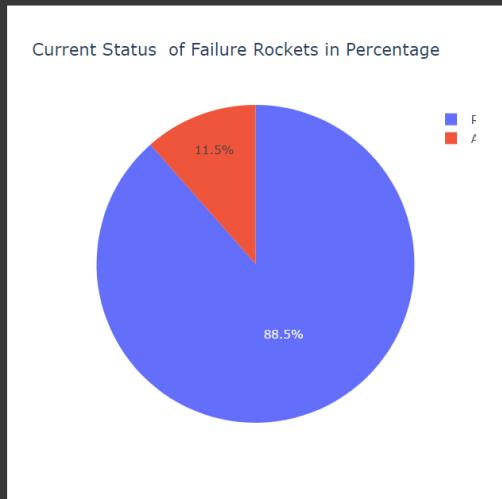


Conclusion: RVSN USSR has highest number of failures

```
[25] #Failure rocket status analysis
data = Failure['Status Rocket'].value_counts().reset_index()
data.columns = ['Status Rocket', 'Number of Rockets']
print(data)
```

Status Rocket	Number of Rockets
Retired	300
Active	39

```
[26] #to plot the pie chart for the current status of failure rockets
fig = px.pie(data, values='Number of Rockets', names='Status Rocket', title='Current Status of Failure Rockets in Percentage',height=500,width=500)
fig.show()
```



Conclusion: Most Failure Rockets are now Retired

7. ISRO analysis

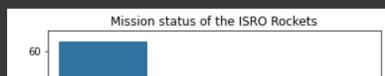
```
[27] # Count Of rockets launched by ISRO
isro=df[df['Company Name']=='ISRO']
print("No of rockets launched by ISRO",isro.shape[0])
```

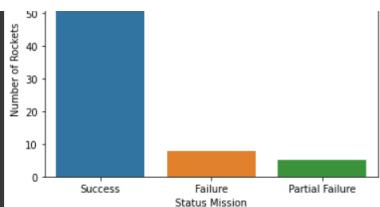
No of rockets launched by ISRO 76

```
[28] # Status Mission of ISRO Rockets
sns.countplot(isro['Status Mission'])
plt.title("Mission status of the ISRO Rockets")
plt.xlabel('Status Mission')
plt.ylabel('Number of Rockets')
plt.show()
```

/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning:

Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result



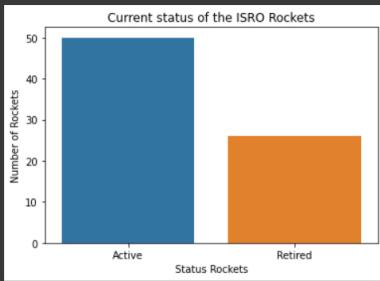


Conclusion: ISRO has given more successful space mission than failure

```
[29] # Current Status of ISRO Rockets
sns.countplot(isro['Status Rocket'])
plt.title("Current status of the ISRO Rockets")
plt.xlabel('Status Rockets')
plt.ylabel('Number of Rockets')
plt.show()
```

/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning:

Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result



Conclusion: Number of active rockets of ISRO is more when compared to retired rockets

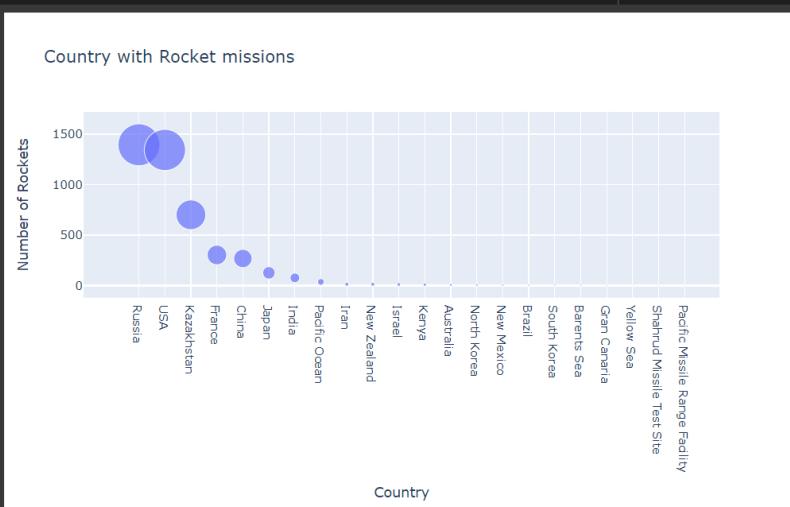
8. Country Analysis

```
[30] #Extracting Country using the location column
df['country'] = df['location'].str.split(', ').str[-1]
df['country'].head()
```

0	USA
1	China
2	USA
3	Kazakhstan
4	USA

Name: country, dtype: object

```
[31] # Count of Each country's Rocket Missions
data = df['country'].value_counts().reset_index()
data.columns = ['Country', 'Number of Rockets']
# Countries with Rocket Missions
fig = px.scatter(data, y='Number of Rockets', x='Country', title='Country with Rocket missions', size='Number of Rockets', size_max=30, height=500, width=800)
fig.show()
```



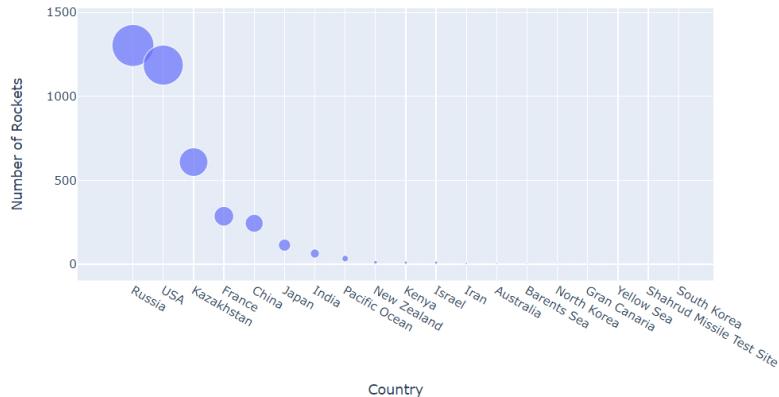
Conclusion: Russia has had the highest number of space missions

```
[32] #Countries with Successful space missions
Success=df[df['Status Mission']=='Success']
data = Success['country'].value_counts().reset_index()
data.columns = ['Country', 'Number of Rockets']
```

```
fig = px.scatter(data, y='Number of Rockets', x='Country', title='Country with most successful space missions', size='Number of Rockets', size_max=30, height=500, width=800)
```

```
fig.show()
```

Country with most successful space missions



Conclusion: Russia has had the highest number of successful space missions

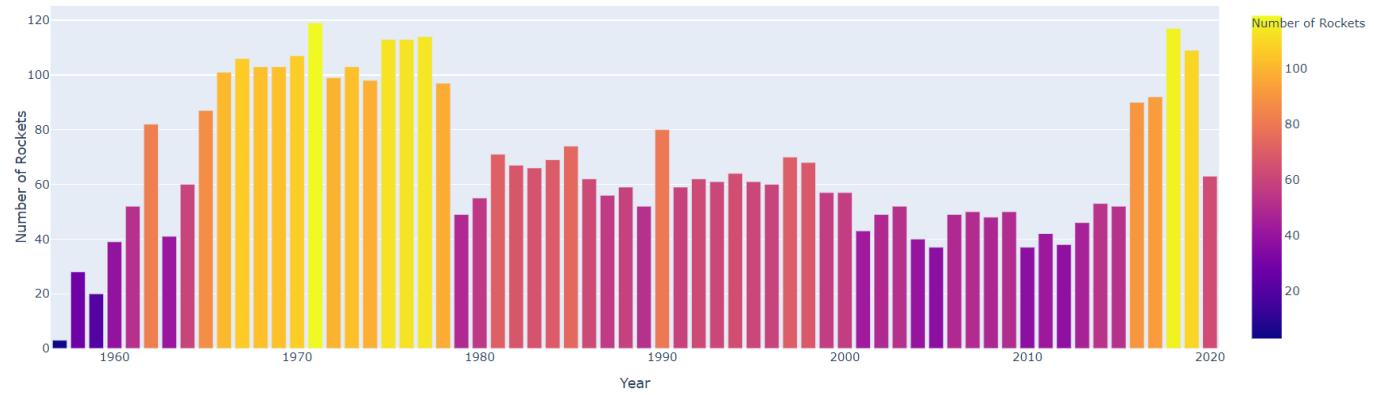
▼ 9. Launch Year

```
[33] # To analyse and plot the number of space mission per year
def get_year(x):
    return x[12:16]
df['Year']=df['Datum'].map(get_year)
df['Year']=df['Year'].astype('int64')

data = df['Year'].value_counts().reset_index()
data.columns = ['Year', 'Number of Rockets']

fig = px.bar(data, y='Number of Rockets', x='Year', orientation='v', title='Year with most space missions', color='Number of Rockets')
fig.show()
```

Year with most space missions



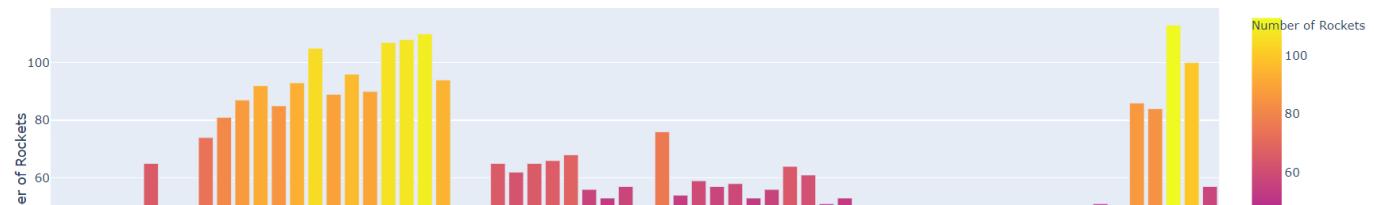
Conclusion: 1971 majority of the space missions took place

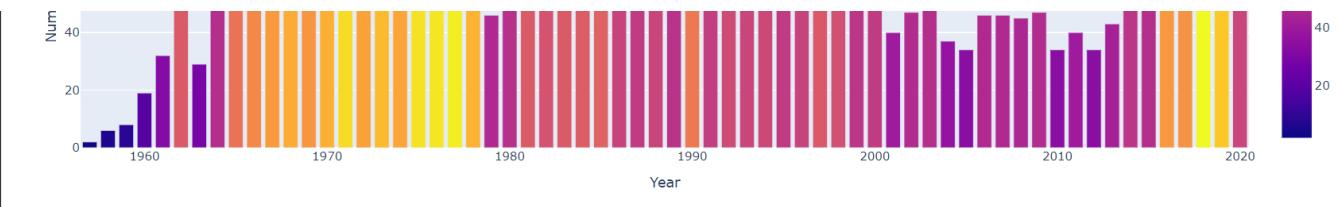
```
[34] # To analyse and plot the number of successful space mission per year
Success=df[df['Status Mission']=='Success']

data = Success['Year'].value_counts().reset_index()
data.columns = ['Year', 'Number of Rockets']

fig = px.bar(data, y='Number of Rockets', x='Year', title='Year with most successful space missions',color='Number of Rockets')
fig.show()
```

Year with most successful space missions





Conclusion: In 2018, maximum successful space missions took place

▼ 10. Launch Month

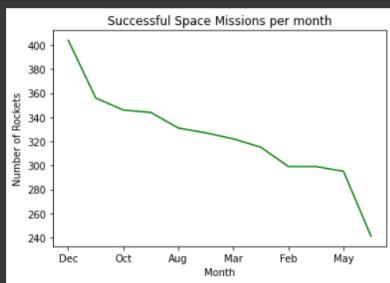
```
[35] # To analyse and plot the number of space mission per month
def get_month(x):
    return x[4:7]
df['Month']=df['Datum'].map(get_month)
data = df['Month'].value_counts().reset_index()
data.columns = ['Month', 'Number of Rockets']

fig = px.line(data, y='Number of Rockets', x='Month', title='Month with most space missions',height=500,width=800)
fig.show()
```



Conclusion: Maximum number of space missions take place in the month of December

```
[36] # To analyse and plot the number of successful space mission per month
Success=df[df['Status Mission']=='Success']
Success['Month'].value_counts().plot(kind='line',color='green')
plt.title('Successful Space Missions per month')
plt.xlabel('Month')
plt.ylabel('Number of Rockets')
plt.show()
```



Conclusion: December has seen maximum successful space missions

▼ 11. Latest Launches per company [2020]

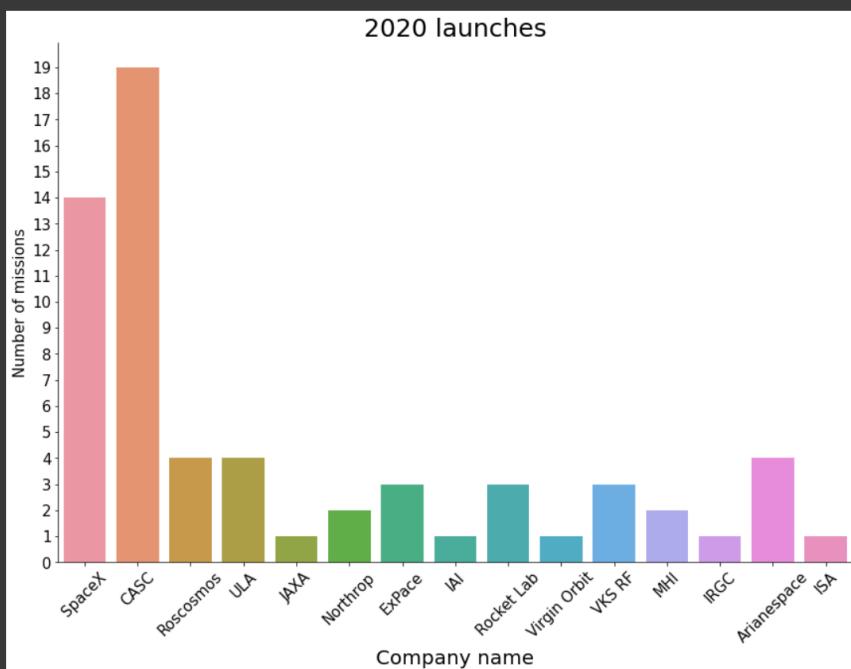
```
[37] #To get the records of latest launch
df_latest=df[df['Year']==2020]

[38] # Plot latest launches using categoryplot
sns.catplot('Company Name',data=df_latest,kind='count',height=8,aspect=1.5)
plt.yticks(np.arange(20))
plt.title('2020 launches',size=25)
plt.xlabel('Company name',size=20)
plt.xticks(size=15,rotation=45)
plt.yticks(size=15)
plt.ylabel('Number of missions',size=15)
plt.show()
```

/use/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning:

```
[38]: /usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:116: FutureWarning:
```

```
Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result
```



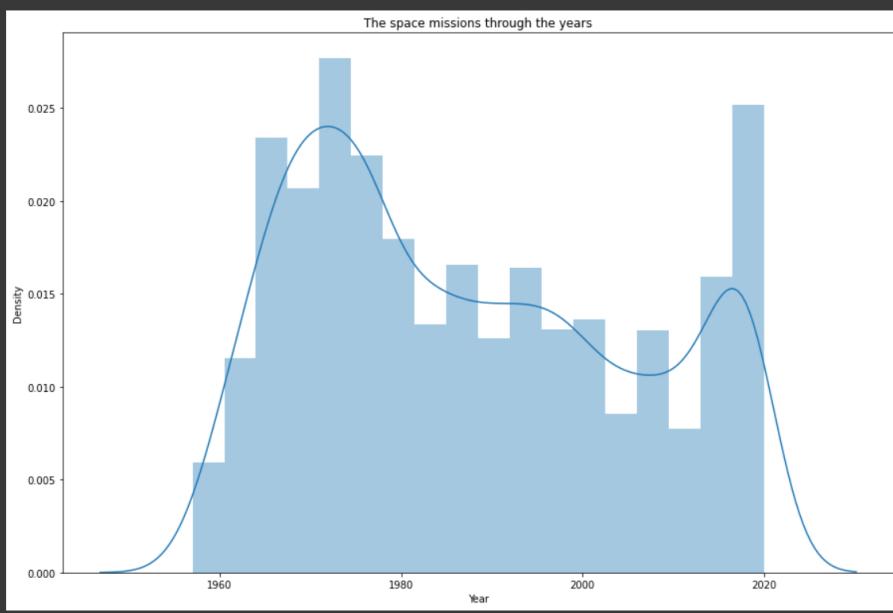
Conclusion: CASC has the maximum latest space missions

▼ 12. The space missions through the years

```
[39] # to plot the space missions through the years
plt.figure(figsize=(15,10))
sns.distplot(df["Year"])
plt.title("The space missions through the years")
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning:
```

```
`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot`
```



✓ 0s completed at 1:40 PM

