

Project 1 Final Presentation

What are the prevailing
attributes of UFO ?

The University of Adelaide
Computer Science (Advanced)

Jiajun Yu (Jason)
a1806320



CONTENTS

01 Research 
Overview



02 Research
Methods 



03 Visualizations
& Code  





Research Overview 01

- The research of Peter Sturrock
- Motivation of the research
- Benefits of the research



Research Overview



Few months ago, the US government released a report about UAP (unidentified aerial phenomena). It is based on the review of 144 significant UAP reports involving observations made by military aviators

The report of the US government



The outcome of the research

However, the task force could only give a concrete explanation for one phenomenon, and the rest remain unexplained (Stieb & Danner 2021)

- Out of curiosity about UFO
- Find the prevailing attributes of UFO

Motivation



Benefits

- Help specialists in this area do some further research.
- It is a social obligation to help people know more about UFO.
- Boost the critical thinking and independent learning skills of people
- Promote the development of philosophy.



Research Methods 02

- All Words Frequencies Counting Code
- Target Words Frequencies Counting Code
- Microsoft Excel Text Filter



Resources Methods



```
In [1]: import pandas as pd
import numpy as np

In [2]: df=pd.read_csv("./Desktop/Project1/UFO_Dateset.csv")

In [3]:
counts={}
for i in range(134421):
    txt=df.Summary.iloc[i]
    for ch in '#%,.?!"()-':
        txt=txt.replace(ch," ")
    txt=txt.lower()
    words=txt.split()

    thisRound={}
    for word in words:
        thisRound[word]=1

    for word in words:
        if thisRound[word]==2:
            break;
        else:
            thisRound[word]=2
            if word not in counts:
                counts[word]=1
            else:
                counts[word]+=1

# Ranking
items=list(counts.items())
items.sort(key=lambda x:x[1],reverse=True)
print("\n");
for i in range(5000):
    word,count=items[i]
    print("{0:<15}{1:>10}".format(word,count))
    print("\n");

in          42446
the         37263
a           33294
and          28976
lights       26242
of           25035
```



```
[24]: import pandas as pd
import numpy as np

[25]: df=pd.read_csv("./Desktop/Project1/UFO_Dateset.csv")

[27]:
count=0
target=['Wife','How','would','you','describe','me',
'Husband','ABCDEFG',
'Wife','What','does','that','mean',
'Husband','Adorable','beautiful','cute',
'delightful','elegant','fashionable','gorgeous']

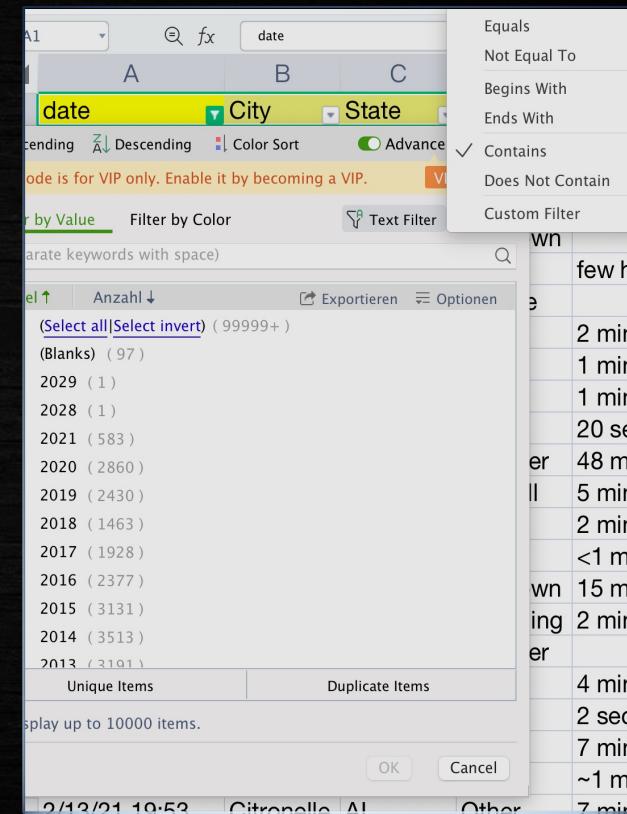
for i in range(134421):
    txt=df.Summary.iloc[i]
    for ch in '#%,.?!"()-':
        txt=txt.replace(ch," ")
    txt=txt.lower()
    words=txt.split()

    allin=[]
    for word in words:
        allin.append(word)

    found=0
    print(allin)
    for i in range(len(target)):
        if target[i] in allin:
            found=1

    if found==1:
        count=count+1

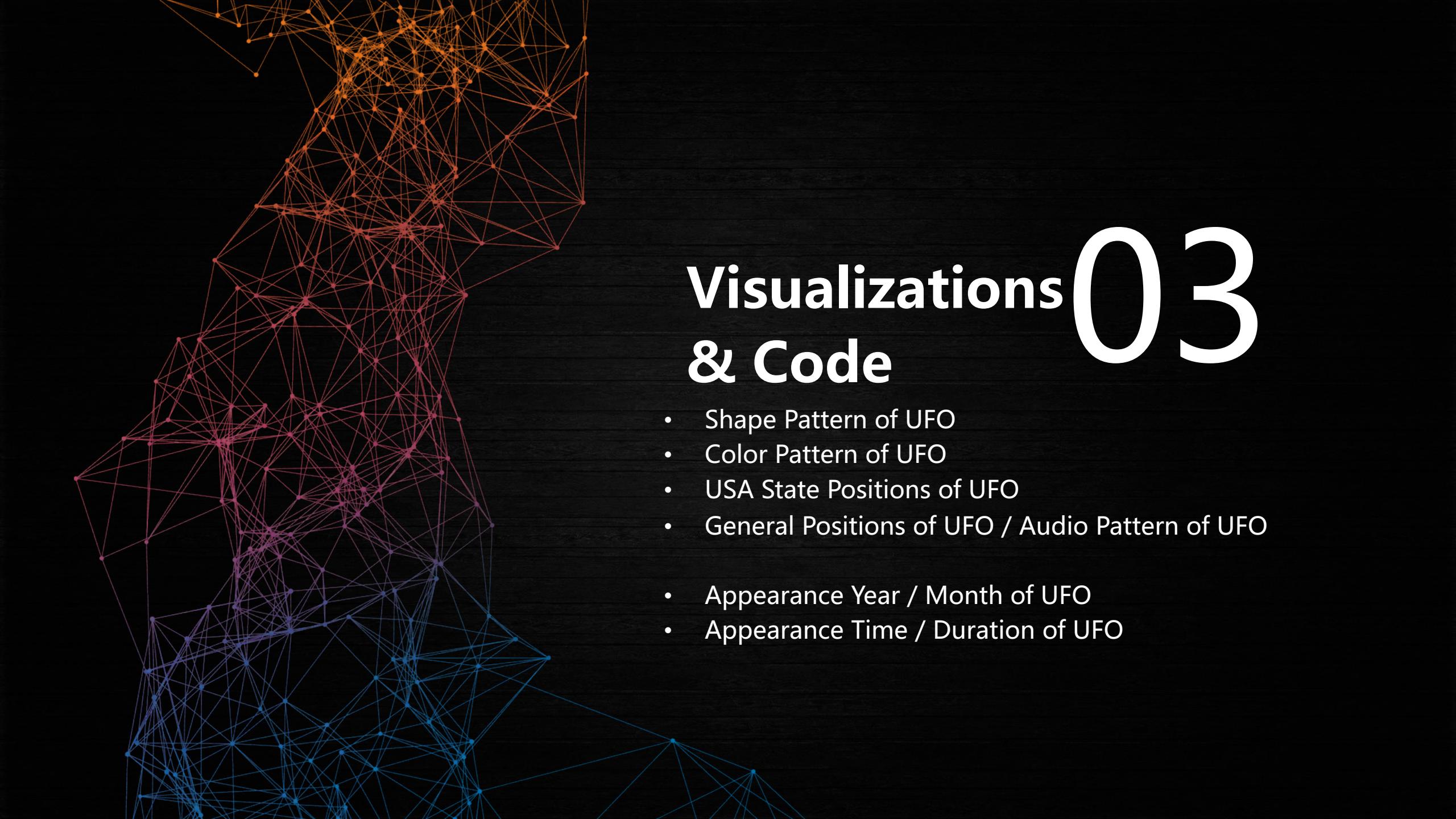
print("The number of records containing these words is :"
The number of records containing these words is : 12585
```



All Words Frequencies Counting

Target Words Frequencies Counting

Microsoft Excel

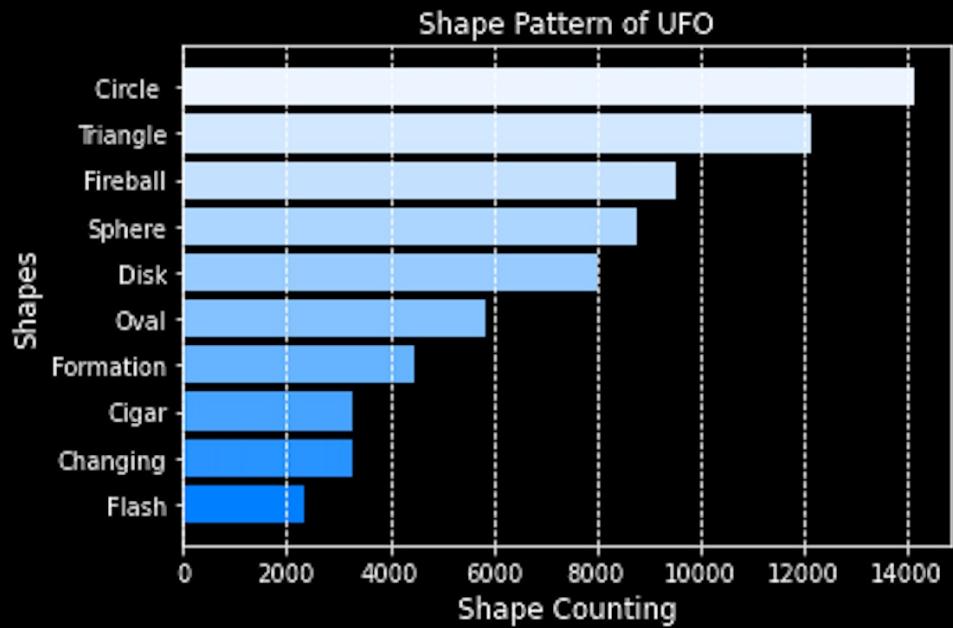
A large, abstract network graph is positioned on the left side of the slide. It consists of numerous small, semi-transparent dots of various colors (orange, red, purple, blue) connected by thin, semi-transparent lines, forming a complex web-like structure.

Visualizations 03 & Code

- Shape Pattern of UFO
- Color Pattern of UFO
- USA State Positions of UFO
- General Positions of UFO / Audio Pattern of UFO

- Appearance Year / Month of UFO
- Appearance Time / Duration of UFO

Shape Pattern of UFO



43%
Records

Top 3 words related to shapes

Circle, triangle and fireball are top 3 words used to describe the shapes of UFO.

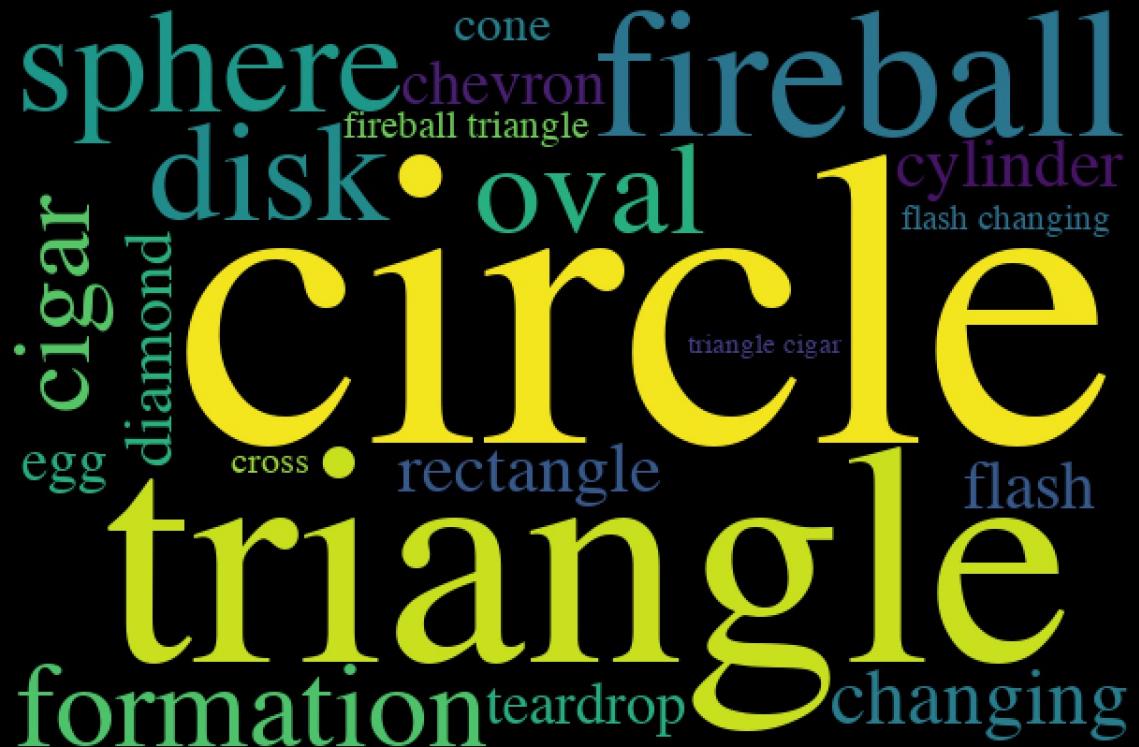
$\approx 57\%$
Records

Ratio of circular shapes

People might be affected by the stereotype of UFO.



 Show all words frequencies by plotting them with various sizes



Shape Pattern of UFO

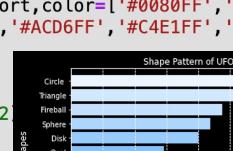
```
import numpy as np
import matplotlib.pyplot as plt
y = np.array(["Flash","Changing","Cigar","Formation","Oval",
              "Disk","Sphere","Fireball","Triangle","Circle"])
x = np.array([2356,3262,3264,4479,5842,8002,8754,9516,12132,14163])

sortIndex = np.argsort(x) # Rank ordering

x_sort = x[sortIndex] # Sorting
y_sort = y[sortIndex]

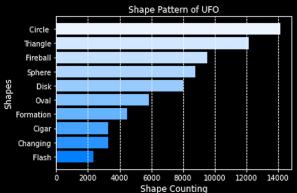
plt.yticks(np.arange(len(y_sort)), y_sort)
a = plt.barh(np.arange(len(y_sort)),x_sort,color=['#0080FF', '#2894FF',
       '#66B3FF', '#84C1FF', '#97CBFF', '#ACD6FF', '#C4E1FF', '#D2E9FF'])

plt.plot(kind='barh')
plt.title('Shape Pattern of UFO')
plt.xlabel('Shape Counting', fontsize=12)
plt.ylabel('Shapes', fontsize=12)
plt.style.use('dark_background')
plt.grid(axis="x",ls='--')
plt.show()



| Shape     | Count |
|-----------|-------|
| Circle    | 14163 |
| Triangle  | 12132 |
| Fireball  | 9516  |
| Sphere    | 8754  |
| Disk      | 8002  |
| Oval      | 5842  |
| Formation | 4479  |
| Cigar     | 3264  |
| Changing  | 3262  |
| Flash     | 2356  |


```



```
import random
import pandas as pd
import numpy as np

file_handle=open("./Desktop/words.txt","w")

index=0
words=[0 for x in range(0,103)]
for i in range(5):
    words[index]="cross"
    index=index+1
for i in range(6):
    words[index]="cone"
    index=index+1
for i in range(13):
    words[index]="teardrop"
    index=index+1
for i in range(13):
    words[index]="egg"
    index=index+1
for i in range(17):
    words[index]="formation"
    index=index+1

for i in range(103):
    words[index]="fireball"
    index=index+1
for i in range(132):
    words[index]="triangle"
    index=index+1
for i in range(154):
    words[index]="circle"
    index=index+1

for i in range(10000):
    random.shuffle(words)

for i in range(103):
    file_handle.write(words[i]+"\n")
```

words.txt ~
n cigar circle other spherical circle circle formation
angle circle other cigar finder cigar cigar cigar re disk sphere cigar changing disk disk oval circle circle circle formation ci on formation disk formation
oval circle disk formati other formation changing cylinder formation formation formation cylinder other angle sphere oval oval form mation circle triangle egg re circle fireball circle changing triangle spher triangle triangle chev ball flash fireball cigar disk circle other triangle triangle oval other rectangle sphere cylinder ball sphere formation fir mation other cylinder cigar sphere disk circle disk s tation fireball oval circle rectangle sphere circle c oval formation sphere chan disk flash oval cigar cig circle circle sphere fire changing other oval fire oval disk fireball flash ar changing fireball fireb er other diamond sphere t other diamond sphere changing other formation dis angle oval formation sph circle flash triangle fire cylinder cigar changing o cylinder fireball disk cle disk sphere disk forma chevron sphere disk format drop sphere other flash ci dder triangle other circle egg sphere cylinder forma ball triangle triangle re gar circle cigar changing ish triangle oval circle t other triangle triangle fo ion changing circle format angle disk changing egg d le sphere sphere triangle a fireball circle formation here teardrop other disk c reball sphere cigar circle val cylinder changing circ al formation triangle fire gle cross circle circle ot disk formation triangle ball disk fireball changi triangle other sphere disk fireball formation sphere re circle fireball cigar s e changing sphere other angle changing triangle f mation rectangle sphere t

```
from wordcloud import WordCloud  
  
with open("./Desktop/word.txt" ,encoding="utf-8")as file:  
    # Read text  
    text=file.read()
```

```
# Some settings
wordcloud=WordCloud(font_path=".~/Library/Fonts/Times.ttc",
background_color="black",width=600,
height=400,max_words=100,relative_scaling=0.3).generate(text)
# Generate Word Cloud Visualizations
image=wordcloud.to_image()
# Show this visualization
image.show()
```



triangle
formation teardrop flash

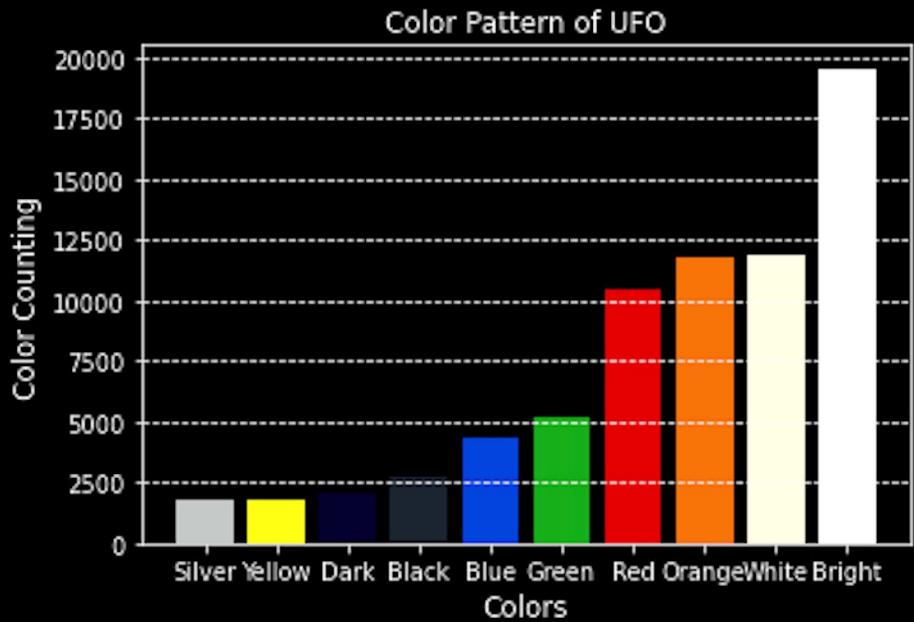
disk cigar changing oval

circle cross rectangle cone

fireballsphere



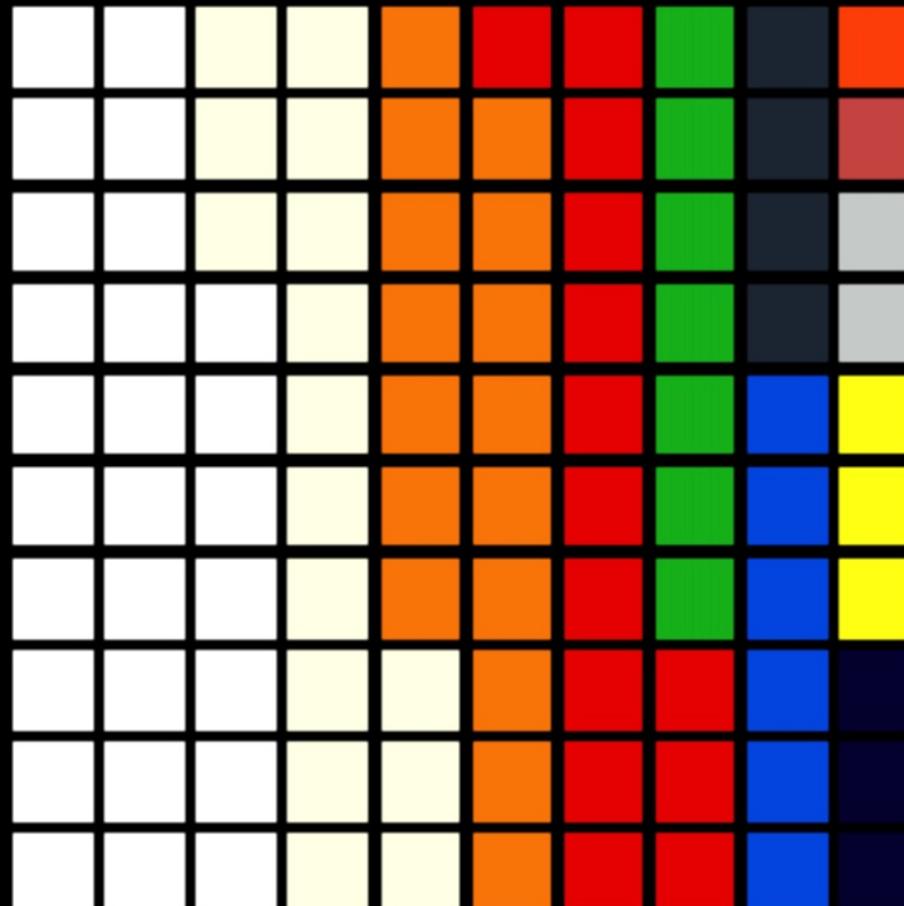
Color Pattern of UFO



Waffle Visualization



Show the prevailing colors of UFO
Show the ratio of them in a more intuitive way



60%
Records

Top 3 words related to colors

Bright, white and orange are top 3 words used to describe the colors of UFO.

≥50%
Records

Ratio of light colors

Most people might witness UFO at night.



Color Pattern of UFO



```
import numpy as np
import matplotlib.pyplot as plt
x = np.array(["Silver", "Yellow", "Dark", "Black", "Blue", "Green", "Red", "Orange", "White", "Bright"])
y = np.array([1736, 1820, 2068, 2676, 4303, 5149, 10420, 11737, 11862, 19535])

sortIndex = np.argsort(y) # Rank ordering

x_sort = x[sortIndex] # Sorting
y_sort = y[sortIndex]

plt.xticks(np.arange(len(x_sort)), x_sort)
a = plt.bar(np.arange(len(x_sort)), y_sort, color=['#c5c9c7', '#fffff14',
['#03012d', '#1b2431', '#0343df', '#15b01a', '#e50000', '#f97306', '#fffffe4

plt.title('Color Pattern of UFO')
plt.ylabel('Color Counting', fontsize=12)
plt.xlabel('Colors', fontsize=12)
plt.style.use('dark_background')
plt.grid(axis="y", ls='--')
plt.show()
```

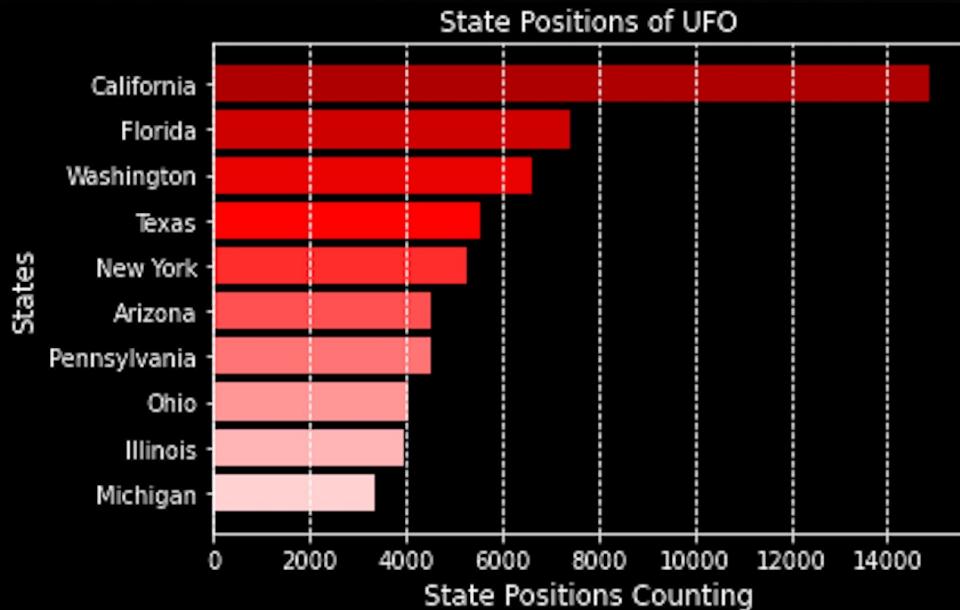
black	2676	0.036877282	3.687728244	#1b2431
blue	4303	0.05929856	5.929855991	#0343df
green	5149	0.070957073	7.095707297	#15b01a
red	10420	0.143595397	14.35953972	#e50000
orange	11737	0.161744643	16.17446427	#f97306
white	11862	0.163467236	16.34672363	#c6fcff
bright	19535	0.269206918	26.92069179	#ffffff

	#613030		#616130		#336666		#484891		#6C
	#743A3A		#707038		#3D7878		#5151A2		#7E
	#804040		#808040		#408080		#5A5AAD		#8F
	#984B4B		#949449		#4F9D9D		#7373B9		#9F
	#AD5A5A		#A5A552		#5CADAD		#8080C0		#A1
	#B87070		#AFAF61		#6FB7B7		#9999CC		#B7
	#C48888		#B9B973		#81C0C0		#A6A6D2		#C0
	#CF9E9E		#C2C287		#95CACAC		#B8B8DC		#C4
	#D9B3B3		#CDCD9A		#A3D1D1		#C7C7E2		#D2
	#E1C4C4		#D6D6AD		#B3D9D9		#D8D8EB		#D4
	#EBD6D6		#DEDEBE		#C4E1E1		#E6E6F2		#E2

```
import matplotlib.pyplot as plt
from pywaffle import Waffle

plt.figure(
    FigureClass=Waffle,
    rows=10,
    columns=10,
    values=[26.92, 16.35, 16.17, 14.36, 7.10, 5.93,
            3.69, 2.85, 2.51, 2.39, 1.06, 0.68],
    colors=("#ffffff", "#ffffe4", "#f97306", "#e50000",
            "#15b01a", "#0343df", "#1b2431", "#03012d",
            "#fffff14", "#c5c9c7", "#c44240", "#fd3c06"),
)
plt.show()
```

USA State Positions of UFO



12%
Records

UFOs were most likely to be witnessed in CA

The ratio of UFOs to be witnessed in California is 12%.

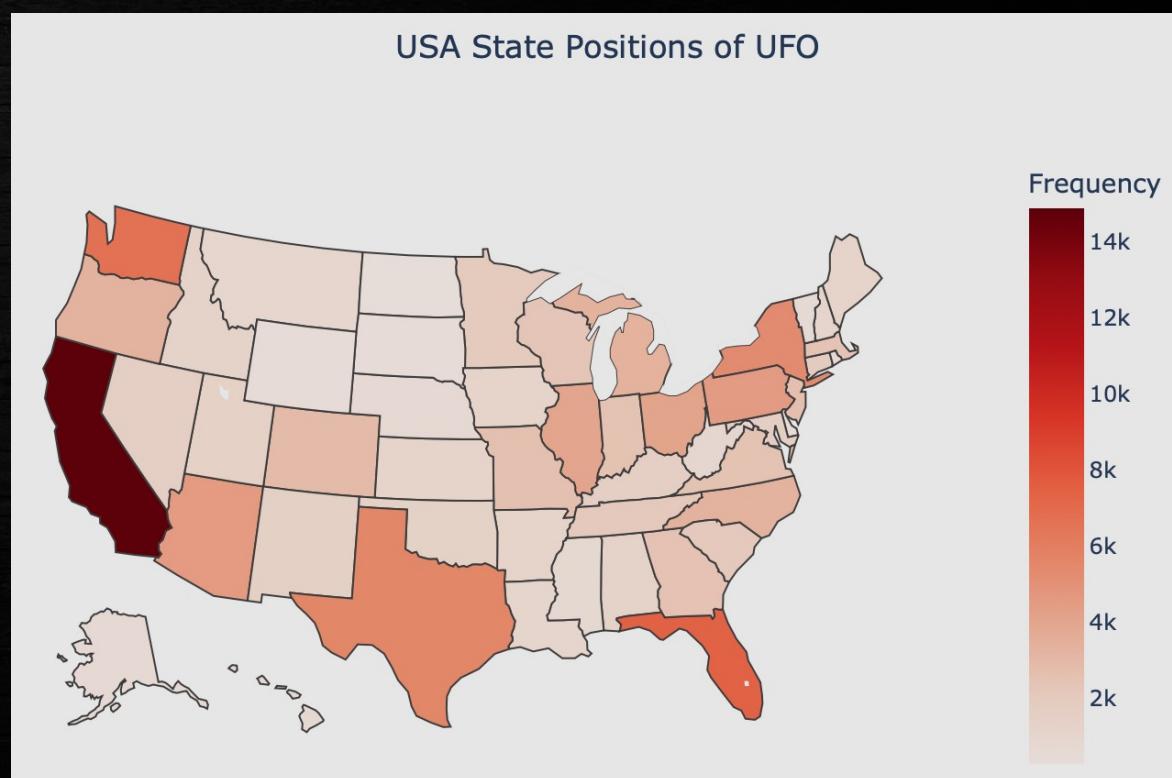
77%
Records

UFOs witnessed in coastal states

77% UFOs were witnessed in 60% states.

Map
Visualization

- Show the distribution of UFO events in the USA
- Use color gradient to represent the overall number of UFO events



USA State Positions of UFO

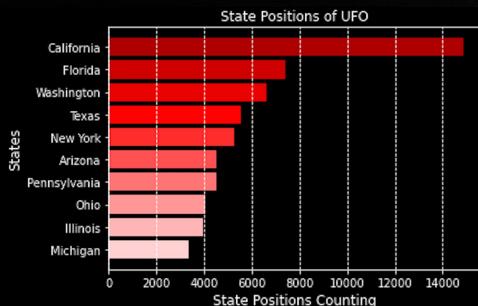
```
import numpy as np
import matplotlib.pyplot as plt
y = np.array(["Michigan", "Illinois", "Ohio", "Pennsylvania", "Arizona",
             "New York", "Texas", "Washington", "Florida", "California"])
x = np.array([3347, 3979, 4050, 4500, 4539, 5286, 5555, 6617, 7398, 14863])

sortIndex = np.argsort(x) # Rank ordering

x_sort = x[sortIndex] # Sorting
y_sort = y[sortIndex]

plt.yticks(np.arange(len(y_sort)), y_sort)
a = plt.barh(np.arange(len(y_sort)), x_sort, color=['#FFD2D2', '#FFB5B5',
'#FF9797', '#FF7575', '#FF5151', '#FF2D2D', '#FF0000', '#EA0000', '#CE0000', '#AE0000'])

plt.plot(kind='barh')
plt.title('State Positions of UFO')
plt.xlabel('State Positions Counting', fontsize=12)
plt.ylabel('States', fontsize=12)
plt.style.use('dark_background')
plt.grid(axis="x", ls='--')
plt.show()
```

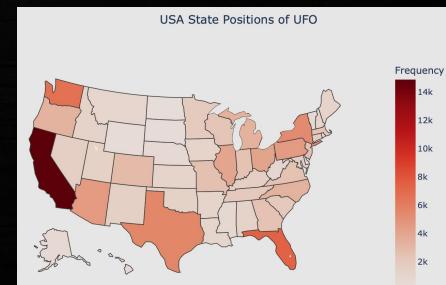


```
In [1]: import plotly.graph_objects as go
import pandas as pd

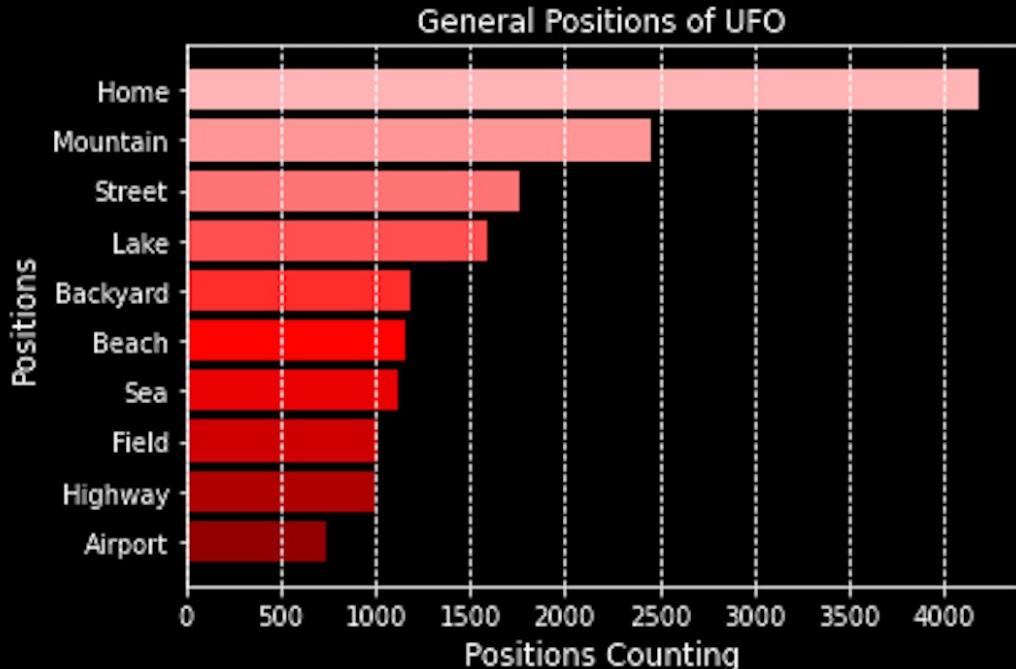
In [2]: df=pd.read_excel("./Desktop/Project1/Research outcomereal.xlsx")

In [3]: fig = go.Figure(data=go.Choropleth(
    locations=df['visualization state'],
    z = df['num4'].astype(float),
    locationmode = 'USA-states',
    colorscale = 'Reds',
    colorbar_title = "Frequency",
))

fig.update_layout(
    title_text = 'USA State Positions of UFO',
    geo_scope='usa',
)
```

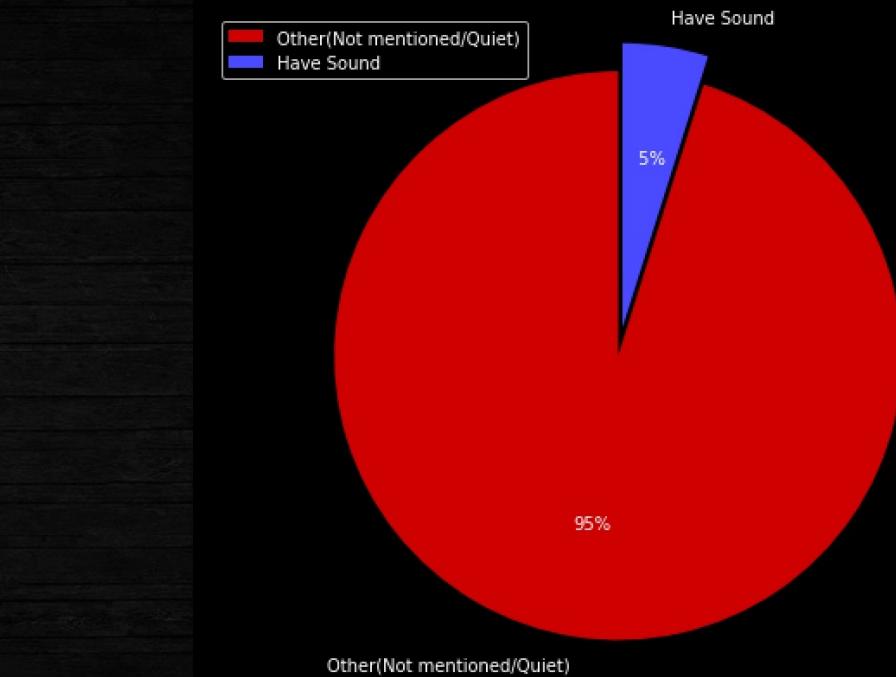


General Positions / Audio Pattern of UFO



General Positions of UFO

UFO were most likely to be witnessed from home.
Top 3 positions are home, mountain and street.



5%
Records

Audio pattern of UFO

There are 5% of eyewitnesses claimed that they heard the sound from UFO.

General Positions / Audio Pattern of UFO

```

import pandas as pd
import numpy as np

df=pd.read_csv("./Desktop/Project1/UFO_Dateset.csv")

counts={}
for i in range(134421):
    txt=df.Summary.iloc[i]
    for ch in '#%,.?!"()-':
        txt=txt.replace(ch," ")
    txt=txt.lower()
    words=txt.split()

    thisRound={}
    for word in words:
        if thisRound[word]==2:
            break;
        else:
            thisRound[word]=2
            if word not in counts:
                counts[word]=1
            else:
                counts[word]+=1

# Ranking
items=list(counts.items())
items.sort(key=lambda x:x[1],reverse=True)
print("\n");
for i in range(5000):
    word,count=items[i]
    print("{0:<15}{1:>10}".format(word,count))
print("\n");

```

in	42446
the	37263
a	33294
and	28976
lights	26242
of	25035
light	24715
sky	22892
object	19825
bright	19535
over	18603
to	17790

home / house /
bedroom /
kitchen
valley / foothills /
mountains / hill
crossing / road
/street / parkway
lake
backyard / yard /
garden
beach
ocean / sea /
field / farm
highway /
freeway
airport
coast / gulf / shore
park
island
woods / forest /
grove
school / college
university

```

y = np.array(["Airport","Highway","Field","Sea","Backyard","Lake","Street","Mountain"])
x = np.array([738,1002,1014,1115,1154,1187,1592,1707])

sortIndex = np.argsort(x) # Rank ordering
x_sort = x[sortIndex] # Sorting
y_sort = y[sortIndex]

plt.yticks(np.arange(len(y_sort)), y_sort)
a = plt.barh(np.arange(len(y_sort)),x_sort,color=[ "#CE0000", "#EA0000", "#FF0000", "#FF2D2D", "#FF5F51", "#FF8C00", "#FFB6C1", "#FFCCBC", "#FFEB3B", "#FFEB3B", "#FFEB3B", "#FFEB3B", "#FFEB3B", "#FFEB3B", "#FFEB3B", "#FFEB3B"])
plt.plot(kind='barh')
plt.title('General Positions of UFO')
plt.xlabel('Positions Counting', fontsize=12)
plt.ylabel('Positions', fontsize=12)
plt.style.use('dark_background')
plt.grid(axis="x",ls='--')
plt.show()

```

```

import pandas as pd
import numpy as np

df=pd.read_csv("./Desktop/Project1/UFO_Dateset.csv")

count=0
target=['home','house','bedroom','kitchen',
'houses','bedrooms','kitchens',]

for i in range(134421):
    txt=df.Summary.iloc[i]
    for ch in '#%,.?!"()-':
        txt=txt.replace(ch," ")
    txt=txt.lower()
    words=txt.split()

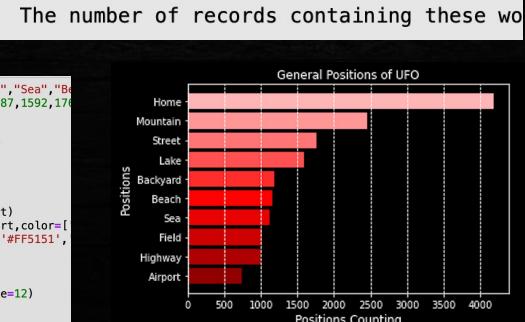
    allin=[]
    for word in words:
        allin.append(word)

    found=0
#    print(allin)
    for i in range(len(target)):
        if target[i] in allin:
            found=1

    if found==1:
        count=count+1

print("The number of records containing these wo

```



```

import pandas as pd
import numpy as np

df=pd.read_csv("./Desktop/Project1/UFO_Dateset.csv")

count=0
target=['rowdy','toot','clang','reverberate',
'clank','clink','creak','clatter',
'clack','echo','echoing','resonate','reso
'resound','buzz','buzzed','blare','blatan
'boisterous','boom','ear','ears','booming
'thundering','vehement','strident','reso
'raucous','vociferous','uproarious','sono
'resonant','piercing','roar','roaring','r
'hear','heard','hearing','listen','listen
'listening','noise','noisy','voice','voic
'veiced','sound','sounds','sounded','audi
'audible','loud','hearable','thunder']

for i in range(134421):
    txt=df.Summary.iloc[i]
    for ch in '#%,.?!"()-':
        txt=txt.replace(ch," ")
    txt=txt.lower()
    words=txt.split()

    allin=[]
    for word in words:
        allin.append(word)

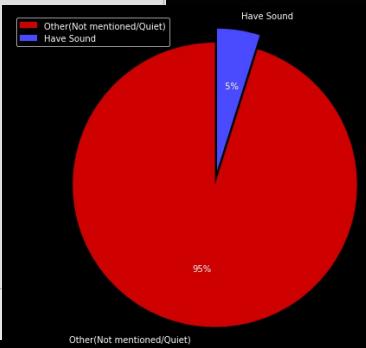
    found=0
    for i in range(len(target)):
        if target[i] in allin:
            found=1

    if found==1:
        count=count+1

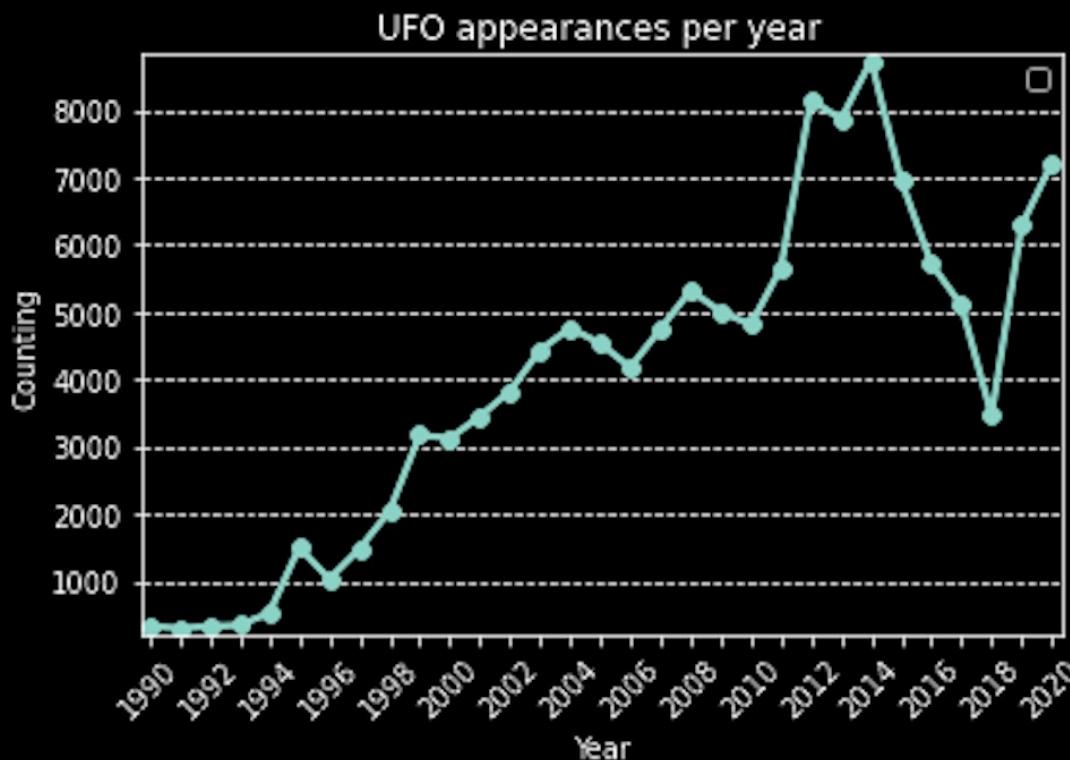
print("The count is :",count)

```

The count is : 6596

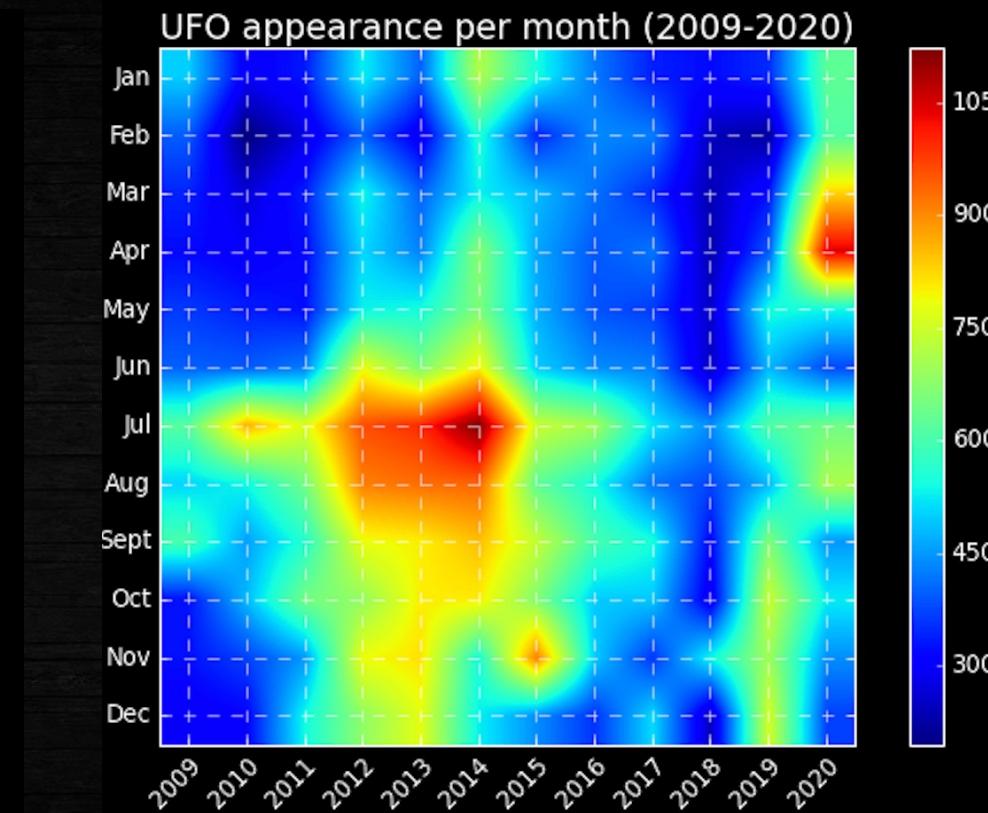


Appearance Year / Month of UFO



UFO appearances per year

The number of UFO events has risen since 1990. In 2014, people witnessed the most UFO events. After that, it dropped rapidly until 2018.



UFO appearance per month(2009-2020)

July and August from 2012 to 2014
March to April of 2020

Appearance Year / Month of UFO

date

5/19/21 20:15

5/19/21 17:25

5/19/21 14:30

5/19/21 10:50

5/19/21 01:00

5/19/21 00:23

5/18/21 00:00

5/18/21 21:25

5/18/21 12:00

5/18/21 11:30

5/18/21 07:40

5/18/21 04:00

5/18/21 02:00

5/18/21 00:30

5/17/21 22:07

5/17/21 21:30

5/17/21 11:00

5/17/21 02:08

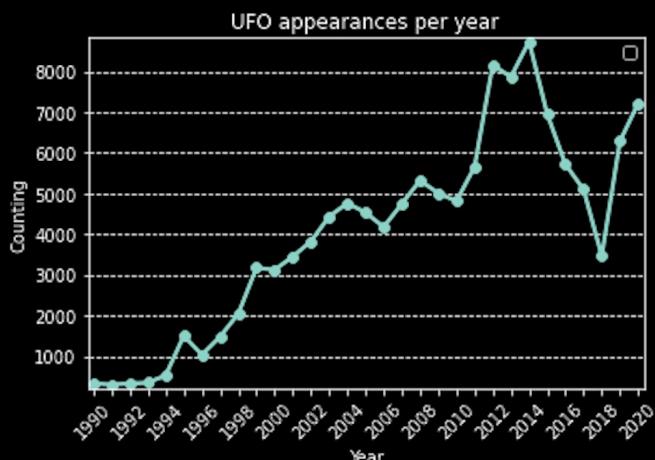
5/17/21 01:20

5/17/21 01:20

```
import matplotlib.pyplot as plt
names = ['1990', ' ', '1992', ' ', '1994', ' ', '1996', ' ', '1998', ' ', '2000', ' ', '2002', ' ', '2004', ' ', '2006', ' ', '2008', ' ', '2010', ' ', '2012', ' ', '2014', ' ', '2016', ' ', '2018', ' ', '2020']
x = range(len(names))
y = [332, 303, 331, 351, 527, 1505, 1038, 1483, 2045, 3175, 3124, 3452, 3826, 4435, 4764, 4534, 4179, 4749, 5323, 5009, 4814, 5652, 8161, 7865, 8739, 6954, 5720, 5099, 3469, 6286, 7213]

plt.plot(x, y, linewidth=2.5,marker='o')
plt.legend()
plt.xticks(x, names, rotation=45)
plt.margins(0.01)

plt.subplots_adjust(bottom=0.15)
plt.xlabel(u"Year")
plt.ylabel("Counting")
plt.title("UFO appearances per year")
plt.style.use('dark_background')
plt.grid(axis="y",ls='--')
plt.show()
```



```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from pandas import Series,DataFrame
import seaborn as sns
import palettable
from sklearn import datasets

df=pd.read_csv("./Desktop/Project1/month.csv",index_col = ['month'])

df

months = ["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sept", "Oct", "Nov", "Dec"]
years = ["2009", "2010", "2011", "2012", "2013", "2014", "2015", "2016", "2017", "2018", "2019", "2020"]

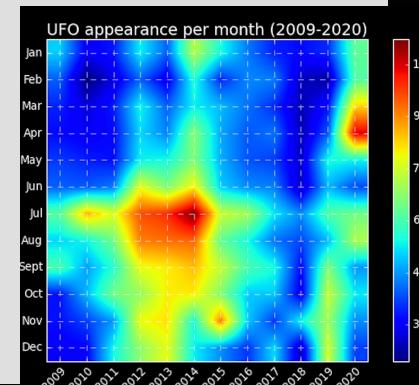
data = np.array([[501, 296, 333, 531, 401, 720, 548, 424, 340, 320, 351, 627], [398, 195, 278, 400, 286, 560, 347, 438, 428, 246, 222, 612], [344, 267, 338, 540, 401, 531, 483, 417, 350, 240, 329, 828], [320, 297, 319, 509, 431, 673, 465, 392, 420, 236, 394, 1045], [364, 339, 324, 532, 546, 663, 473, 388, 367, 256, 549, 541], [400, 394, 423, 780, 661, 796, 499, 440, 428, 250, 480, 379], [620, 862, 768, 959, 994, 1122, 746, 703, 522, 441, 611, 647], [507, 541, 648, 913, 929, 938, 631, 546, 418, 377, 481, 713], [603, 458, 562, 784, 802, 849, 741, 601, 548, 316, 665, 451], [329, 482, 649, 686, 804, 805, 667, 496, 495, 286, 746, 528], [327, 372, 451, 786, 815, 552, 913, 499, 361, 541, 699, 442], [296, 311, 541, 681, 777, 531, 441, 358, 512, 246, 759, 370]])
```

```
fig, ax = plt.subplots()
im = ax.imshow(data)

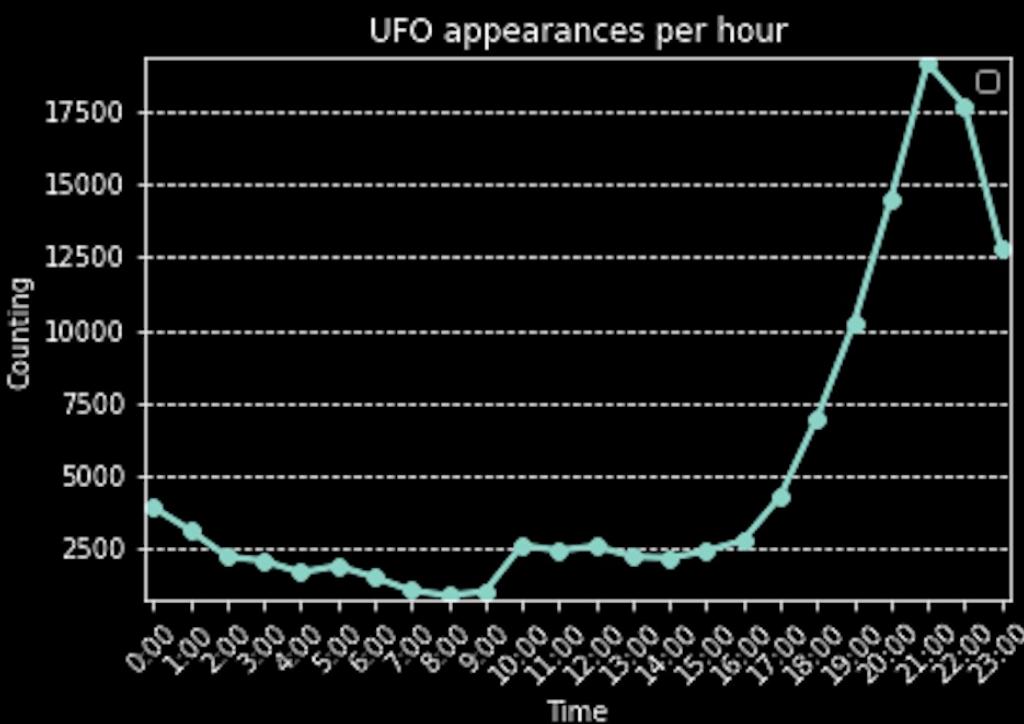
# We want to show all ticks...
ax.set_xticks(np.arange(len(years)))
ax.set_yticks(np.arange(len(months)))
# ... and label them with the respective list entries
ax.set_xticklabels(years)
ax.set_yticklabels(months)

# Rotate the tick labels and set their alignment.
plt.setp(ax.get_xticklabels(), rotation=45, ha="right",
         rotation_mode="anchor")

ax.set_title("UFO appearance per month (2009-2020)")
fig.tight_layout()
plt.colorbar(im)
plt.style.use('dark_background')
plt.show()
```



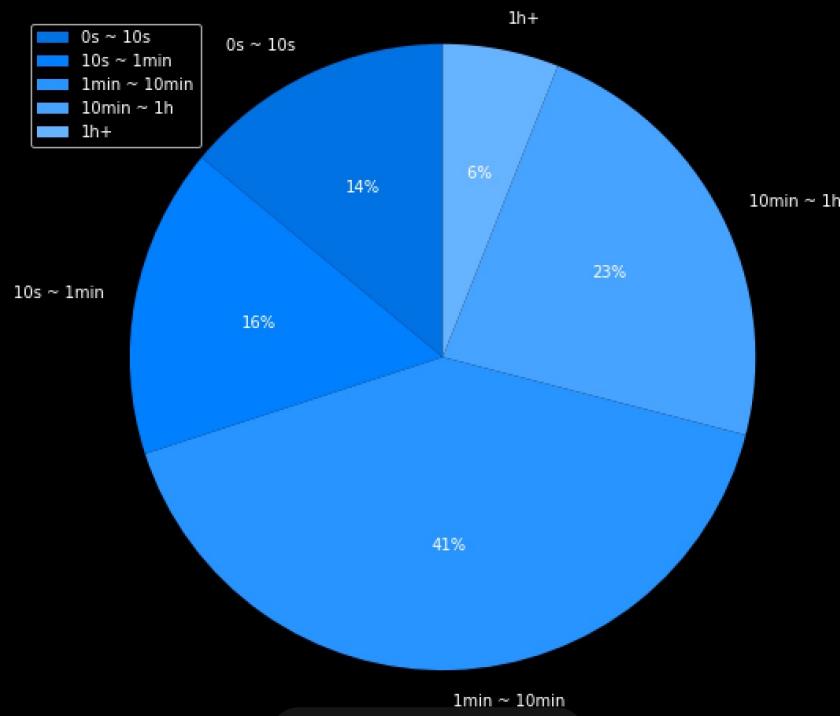
Appearance Time / Duration of UFO



UFO appearances per hour

After 4 pm, the numbers of UFO events increased sharply until 9pm.

Corresponds to previous guess



Durations of UFO appearances

- 0s~10s: 14%
- 0s~1min: 30%
- 1min~10min: 41%

Appearance Time / Duration of UFO

Find what: 21: Replace with: Go To

Within: Sheet Match case

Search for a function: By rows Match entire cell contents

Look in: Smart

Find All Find Previous Find

Workbook	Sheet	Name	Cell	Values
UFO_Dateset.csv	UFO_Dateset	\$A\$9	5/18/21 21:25	
UFO_Dateset.csv	UFO_Dateset	\$A\$17	5/17/21 21:30	
UFO_Dateset.csv	UFO_Dateset	\$A\$32	5/16/21 21:30	
UFO_Dateset.csv	UFO_Dateset	\$A\$73	5/15/21 21:30	
UFO_Dateset.csv	UFO_Dateset	\$A\$74	5/15/21 21:00	
UFO_Dateset.csv	UFO_Dateset	\$A\$84	5/14/21 21:30	
UFO_Dateset.csv	UFO_Dateset	\$A\$92	5/13/21 21:30	
UFO_Dateset.csv	UFO_Dateset	\$A\$93	5/13/21 21:05	
UFO_Dateset.csv	UFO_Dateset	\$A\$94	5/13/21 21:00	
UFO_Dateset.csv	UFO_Dateset	\$A\$98	5/12/2021 21:30	
UFO_Dateset.csv	UFO_Dateset	\$A\$107	5/10/2021 21:30	

```

import matplotlib.pyplot as plt
names = ['0:00', '1:00', '2:00', '3:00', '4:00', '5:00', '6:00', '7:00', '8:00', '9:00',
        '10:00', '11:00', '12:00', '13:00', '14:00', '15:00', '16:00', '17:00', '18:00',
        '19:00', '20:00', '21:00', '22:00', '23:00']
x = range(len(names))
y = [3919, 3131, 2226, 2083, 1676, 1911, 1513, 1061, 892, 1025, 2618, 2429, 2590,
     2229, 2181, 2415, 2778, 4273, 6962, 10217, 14460, 19152, 17704, 12793]
plt.plot(x, y, linewidth=2.5, marker='o')
plt.legend()
plt.xticks(x, names, rotation=45)
plt.margins(0.01)

plt.subplots_adjust(bottom=0.15)
plt.xlabel("Time")
plt.ylabel("Counting")
plt.title("UFO appearances per hour")
plt.style.use('dark_background')
plt.grid(axis="y", ls="--")
plt.show()

```

A

Duration

- 12:00
- 5 minutes
- 3 minutes
- 2 min
- 10-15minutes
- 2 minutes
- 3 min
- 5 minutes
- 3-5 minutes
- 40 seconds
- 3-May
- 7 seconds
- 4 minutes
- 2-3 mins
- 6 minutes around 1.22
- 40 Minutes
- approximately one second
- <10s
- 30 minutes
- 5 minutes
- Seconds to 1 Minute
- Every night for weeks
- 3 min
- 5 minutes
- 30mins
- 5 seconds
- 20 minutes
- 4 minutes
- 2 hours

Find what: &

Replace with: ||

Find what: <

Replace with: ||

Find what: ~

Replace with: ||

Find what: +

Replace with: ||

Find what: ;

Replace with: ||

Find what: ?

Replace with: ||

Find what: ap

Replace with: ||

A

Duration

Descending

mode is fo

Color by Value

Filter by Color

Separate keywords with space

```

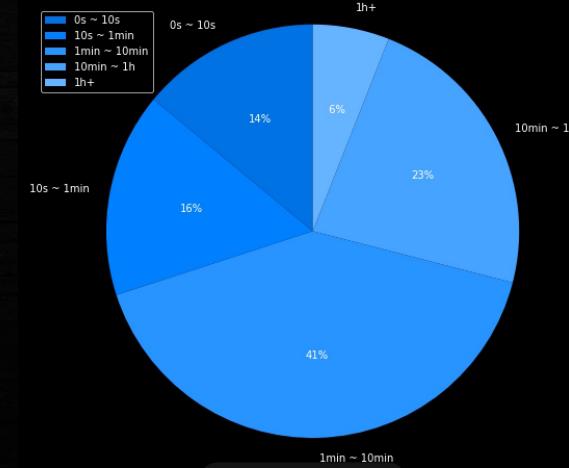
plt.figure(figsize=(8,8))
labels = [u'0s ~ 10s', u'10s ~ 1min', u'1min ~ 10min', u'10min ~ 1h', u'1h+']
sizes = [14,16,41,23,6]
colors = ['#0072E3', '#0080FF', '#2894FF', '#46A3FF', '#66B3FF']

explode = (0,0,0,0,0)

patches, l_text, p_text = plt.pie(sizes, explode=explode, labels=labels, colors=colors,
                                 labeldistance=1.1, autopct='%2.0f%%', shadow=False,
                                 startangle=90, pctdistance=0.6)

for t in l_text:
    t.set_size = 30
for t in p_text:
    t.set_size = 20
plt.axis('equal')
plt.legend(loc='upper left', bbox_to_anchor=(-0.1, 1))
plt.grid()
plt.style.use('dark_background')
plt.show()

```





Workbook

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Colors	Color Counting	Shape Pattern of UFO	num2	State Positions Pattern num3	Audio Pattern of UFO			visualization state	num4		year	num5		during	num6
2	silver	1736 cross		435 NJ	2621 Has sound	4.87%	6553	AK	607		1990	332		0:00	3919	
3	yellow	1820 cone		537 CO	2905 Not mentiond / no sound	95.13%	127826	AL	1235		1991	303		1:00	3131	
4	dark	2068 teardrop		1163 OR	3264			AR	1136		1992	331		2:00	2226	
5	black	2676 egg		1171 NC	3285			AZ	4539		1993	351		3:00	2083	
6	blue	4303 chevron		1537 MI	3347 Michigan			CA	14893		1994	527		4:00	1676	
7	green	5149 diamond		1948 IL	3979 Illinois			CO	2905		1995	1505		5:00	1911	
8	red	10420 cylinder		2091 OH	4050 Ohio			CT	1812		1996	1038		6:00	1513	
9	orange	11737 rectangle		2284 PA	4500 Pennsylvania			DE	381		1997	1483		7:00	1061	
10	white	11862 flash		2356 AZ	4639 Arizona			FL	7398		1998	2045		8:00	892	
11	bright	19535 "changing cigar		3262 NY	5286 New York			GA	2469		1999	3175		9:00	1025	
12		0 formation		3284 TX	5555 Texas			HI	630		2000	3124		10:00	268	
13		oval		4479 WA	6817 Washington			IA	1125		2001	3452		11:00	2429	
14		disk		5842 FL	7398 Florida			ID	1226		2002	3826		12:00	2590	
15		sphere		8002 CA	14863 California			IL	3979		2003	4435		13:00	2229	
16		fireball		8754				IN	2451		2004	4764		14:00	2181	
17		triangle		8516				KS	1087		2005	4834		15:00	2415	
18		circle		12132				KY	1538		2006	4778		16:00	2778	
19				14163				LA	1016		2007	4749		17:00	4273	
20								MA	2400		2008	5323		18:00	6962	
21				82936				MD	1655		2009	5009		19:00	10217	
22								ME	1087		2010	4814		20:00	14460	
23								MI	3347		2011	5652		21:00	19152	
24								MN	1935		2012	8161		22:00	17704	
25								MO	2568		2013	7865		23:00	12793	
26								MS	729		2014	8739		24:00	122938	





Reference



Carvalho, T 2020, *Heatmap Basics with Seaborn*, Towards Data Science, viewed 1 September 2021, <<https://towardsdatascience.com/heatmap-basics-with-pythons-seaborn-fb92ea280a6c>>.

Chauhan, A 2020, *Find frequency of each word in a string in Python*, GeeksforGeeks, viewed 10 August 2021,<<https://www.geeksforgeeks.org/find-frequency-of-each-word-in-a-string-in-python/>>.

Daniela, G et al. 2021, *Unidentified flying object*, Wikipedia, viewed 5 September 2021, <https://en.wikipedia.org/wiki/Unidentified_flying_object>.

Gawande, J 2020, *Plot a pie chart in Python using Matplotlib*, GeeksforGeeks, viewed 23 August 2021, <<https://www.geeksforgeeks.org/plot-a-pie-chart-in-python-using-matplotlib/>>.

Gawande, J 2021, *Bar Plot in Matplotlib*, GeeksforGeeks, viewed 21 August 2021, <<https://www.geeksforgeeks.org/bar-plot-in-matplotlib/>>.

Kaggle 2021, *National UFO Reporting Data*, csv, viewed 7 August 2021, <<https://www.kaggle.com/connerbrew2/national-ufo-reporting-data>>.

Knuth, K 2018, *Are we alone? The question is worthy of serious scientific study*, The Conversation, viewed 4 September 2021, <<https://theconversation.com/are-we-alone-the-question-is-worthy-of-serious-scientific-study-98843>>.

Li, G 2021, *Make Waffle Charts in Python*, Python Repo, viewed 25 August 2021, <<https://pythonrepo.com/repo/gylipyWaffle-python-data-validation>>.

Line plot or line chart in python with legends 2021, Data Science Made Simple, viewed 26 August 2021, <<https://www.datasciencemade-simple.com/line-plot-line-chart-in-python-legends/>>.

Stieb, M & Danner C 2021, *What's Inside the Pentagon's Long-Awaited UFO Report*, Intelligencer, viewed 5 September 2021, <<https://nymag.com/intelligencer/article/pentagon-ufo-report-what-we-know.html>>.

Vu, D 2019, *Generating Word Clouds in Python*, Data camp, viewed 29 August 2021, <<https://www.datacamp.com/community/tutorials/wordcloud-python>>.

Wu, L 2020, *13 lines of Python code to draw a map of the US epidemic*, CSDN, viewed 27 August 2021, <<https://blog.csdn.net/wulishinian/article/details/105539212>>.

Thanks for listening

