Slide Type Slide

1 Title Slide - Group 3 - Meng Chen, Yi Xiao, Adrian Vojvodic

Slide Type Slide

Slide Type Slide

1 Summary

Core Hypothesis:

Number of cloud to ground lightning strikes over the continental United States have increased over the last 30 years.

Question:

Given the daily cloud to ground lightning strike data from NOAA, is there an increase in overall lightning strike activity over the 30+ years data?

Findings:

We were able to show an increase in cloud to ground lightning activity from 1987 to 2018.

Slide Type Slide

Data:

Data was sourced from Google Cloud Services BigQuery Database, which can be found under the Big Data menu of Google Cloud Platform

The data itself was clean, with a total volume of approximately 3.4GB divided annually from 1987 to present. Each year's data included columns for "Date", "Number_of_Strikes", and "Center_Point"

We initially began to download it directly in .csv format from Google Cloud Platform but encountered multiple errors due to the size of the datasets.

Slide Type: Slide

Data (continued):

We then attempted to use Google Cloud Services utilities to download the data. While educational, it proved ultimately futile. We returned to look at the .csv download method.

At that time, we were successful downloading directly to .csv, realizing that the query needed to be complete before attempting the download.

Next we utilized a GCS API and a %%BIGQUERY cell magic command to automatically download the data.

Slide Type: Slide

Data (continued):

Location data for the lightning strikes was downloaded as 'centerpoint' where latitude and longitude values are in the same field. In using Google Places location mapping, the latitude and longitude values needed to be separate, string values.

There seemed to be a path to use the centerpoint value as-is with Basemap, another python module.

Slide Type: Slide

2 Conclusion & Postmortem

Conclusions & Postmortem:

Data did indicate an increase in cloud to ground lightning strikes from 1987 to 2017.

Concerns:

1. There is an anomalous increase in 2018 that bears more investigation. Upon recalculating the total number of strikes for that year, the same result was found.

2. Analysis regarding the type and number of data collection sites over the time period is necessary to normalize the results.

3. Dataset size challenged:
    a. network throughput
    b. local machine RAM

Slide Type: Slide

3 Code Excerpts

3.1 %%bigquery Code

In [2]:

```
%load_ext google.cloud.bigquery
import os
os.environ["GOOGLE_APPLICATION_CREDENTIALS"]="D:\D DOWNLOADs\DOCUMENTATION\GoogleCloudServices\Analytics Bootcamp Rice U-e97
import datetime

```

executed in 2.64s, finished 21:23:01 2019-07-25

In [2]:

```
%%bigquery lightning_data_complete --verbose
SELECT 2001,2002 FROM `bigquery-public-data.*`

```

executed in 32m 47s, finished 20:00:51 2019-07-25

Slide Type: Slide

3.2 CSV Read-In Loop Code

In [ ]:

```

```

In [1]:

```
%matplotlib inline
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import os
newList = []
filename = []

path = os.path.join('.')

with os.scandir(path) as it:
    for entry in it:
        if entry.name.endswith(".csv") and entry.is_file():
            filename.append(entry.name)

# filename = ['1992_mod.csv','1993_mod.csv','1994_mod.csv']

for file in filename:
```

executed in 2.64s, finished 21:23:01 2019-07-25

In [2]:

Slide Type

```
%%bigquery lightning_data_complete --verbose
SELECT 2001,2002 FROM `bigquery-public-data.*`

```

executed in 32m 47s, finished 20:00:51 2019-07-25

Slide Type Slide

3.2 CSV Read-In Loop Code

In [ ]:

```

```

In [1]:

```
%matplotlib inline
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import os
newList = []
filename = []

path = os.path.join('.')

with os.scandir(path) as it:
    for entry in it:
        if entry.name.endswith(".csv") and entry.is_file():
            filename.append(entry.name)

# filename = ['1992_mod.csv','1993_mod.csv','1994_mod.csv']

for file in filename:
    data = pd.read_csv(file,encoding='utf-8')
    data.columns = ['day', 'number_of_strikes', 'center_point', 'Day', 'Year', 'Month','Latitude', 'Longitude', 'LatitudeR',
    data.drop(columns = ['day','number_of_strikes','center_point',],axis=1,inplace=True)
    newList.append(data)
```

executed in 4.97s, finished 13:32:54 2019-07-30

In [2]:

```
df = pd.concat(newList)
df.shape
```

executed in 42ms, finished 13:32:57 2019-07-30

Out[2]: (4004046, 8)

Slide Type Slide

3.3 Data Splitting Code

In [18]:

```
datetmp=data.day.str.replace("-"," ")
month=datetmp.str.split(" ",expand=True)[0]
```

In [2]:
```
df = pd.concat(newList)
df.shape
```
executed in 42ms, finished 13:32:57 2019-07-30

Out[2]: (4004046, 8)

Slide Type: Slide

3.3 Data Splitting Code

In [18]:
```
datetmp=data.day.str.replace("-"," ")
month=datetmp.str.split(" ",expand=True)[0]
day=datetmp.str.split(" ",expand=True)[1]
year=datetmp.str.split(" ",expand=True)[2]

loctmp=data.center_point.str.replace("POINT\(","")
loctmp=loctmp.str.replace("\)","")
latitude=loctmp.str.split(" ",expand=True)[0]
longitude=loctmp.str.split(" ",expand=True)[1]

strikes=data.number_of_strikes
```
executed in 1m 44.4s, finished 23:23:37 2019-07-26

In [19]:
```
data['Year']=year.astype(float)
data['Month']=month.astype(float)
data['Day']=day.astype(float)
data['Latitude']=latitude.astype(float)
data['Longitude']=longitude.astype(float)
data['LatitudeR']=latitude.astype(float).round(0)
data['LongitudeR']=longitude.astype(float).round(0)
data['Strikes']=strikes
```
executed in 5.73s, finished 23:23:43 2019-07-26

Slide Type: Slide

3.4 Graphing Code

Out[20]:

	day	number_of_strikes	center_point	Year	Month	Day	Latitude	Longitude	LatitudeR	LongitudeR	Strikes
0	1994-01-02	28	POINT(-79.2 30.5)	2.0	1994.0	1.0	-79.2	30.5	-79.0	30.0	28
1	1994-01-02	30	POINT(-78.6 27.8)	2.0	1994.0	1.0	-78.6	27.8	-79.0	28.0	30
2	1994-01-02	58	POINT(-78.8 27.7)	2.0	1994.0	1.0	-78.8	27.7	-79.0	28.0	58
3	1994-01-02	40	POINT(-78 28.1)	2.0	1994.0	1.0	-78.0	28.1	-78.0	28.0	40
4	1994-01-02	24	POINT(-78 28)	2.0	1994.0	1.0	-78.0	28.0	-78.0	28.0	24

In [5]:
```
grouped_df=data.groupby('LatitudeR')
Lat=grouped_df['LatitudeR'].sum()/grouped_df['LatitudeR'].count()
Gstrikes=grouped_df['Strikes'].sum()
```

```
data['Latitude']=latitude.astype(float)
data['Longitude']=longitude.astype(float)
data['LatitudeR']=latitude.astype(float).round(0)
data['LongitudeR']=longitude.astype(float).round(0)
data['Strikes']=strikes
```

executed in 5.73s, finished 23:23:43 2019-07-26

Slide Type: Slide

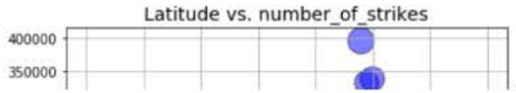
3.4 Graphing Code

Out[20]:

	day	number_of_strikes	center_point	Year	Month	Day	Latitude	Longitude	LatitudeR	LongitudeR	Strikes
0	1994-01-02	28	POINT(-79.2 30.5)	2.0	1994.0	1.0	-79.2	30.5	-79.0	30.0	28
1	1994-01-02	30	POINT(-78.6 27.8)	2.0	1994.0	1.0	-78.6	27.8	-79.0	28.0	30
2	1994-01-02	58	POINT(-78.8 27.7)	2.0	1994.0	1.0	-78.8	27.7	-79.0	28.0	58
3	1994-01-02	40	POINT(-78 28.1)	2.0	1994.0	1.0	-78.0	28.1	-78.0	28.0	40
4	1994-01-02	24	POINT(-78 28)	2.0	1994.0	1.0	-78.0	28.0	-78.0	28.0	24

In [5]:
```
grouped_df=data.groupby('LatitudeR')
Lat=grouped_df['LatitudeR'].sum()/grouped_df['LatitudeR'].count()
Gstrikes=grouped_df['Strikes'].sum()
```

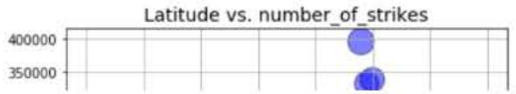
In [6]:
```

plt.scatter(Lat, Gstrikes, marker="o", facecolors="b", label="Suburban",edgecolors="black", s=Gstrikes/1000, alpha=0.5)
plt.title("Latitude vs. number_of_strikes",fontsize=14)
plt.xlabel("Latitude",fontsize=12)
plt.ylabel("number_of_strikes",fontsize=12)
plt.grid()
plt.savefig("LatStrikes.png")
```



Slide Type: Slide

3.5 Heatmap Code

```
plt.grid()
plt.savefig("LatStrikes.png")
```



Slide Type: Slide

3.5 Heatmap Code

```
# # Add Latitude and Longitude information to original data. And convert the datatype.
data['lat']=lat
data['lon']=lon


# # # Extract unique locations
locates=data.drop_duplicates()
# # locates.columns={'Lat','Lon'}

# # lon=locates.iloc[0,0]
# # lat=locates.iloc[0,1]

# # locates
data.shape
data.dtypes
```

executed in 11ms, finished 13:08:40 2019-07-30

In [ ]:

```
# Dependencies
import requests
import json

# Google developer API key
from config import API_KEY

# Extract addresses from GoogleAPI and store in a file
lens=data.shape[0]
# file1=open('Locations.csv',"w")
location = []
mLon = []
mLat = []


for i in range(lens):
    try:

        lon=data.iloc[i,5]
        lat=data.iloc[i,6]
        url="https://maps.googleapis.com/maps/api/geocode/json?latlng="+lat+","+lon+"&key="+API_KEY
        info=requests.get(url).json()
        address=info['results'][0]["formatted_address"]
#         print ("The Latitude is "+lat+ " ;the Longitude is "+ lon+ ";the address is "+ address+"\n")
#         file1.write(lon+"\t"+lat+"\t"+address+"\n")
        mLon.append(lon)
        mLat.append(lat)
        location.append(info['results'][0]["formatted_address"])
    except ValueError:
        try:
            next
        except IndexError:
            next

# file1.close()
```

executed in 1.07s, finished 13:08:42 2019-07-30

In [ ]:

```
info
```

Slide Type: Slide

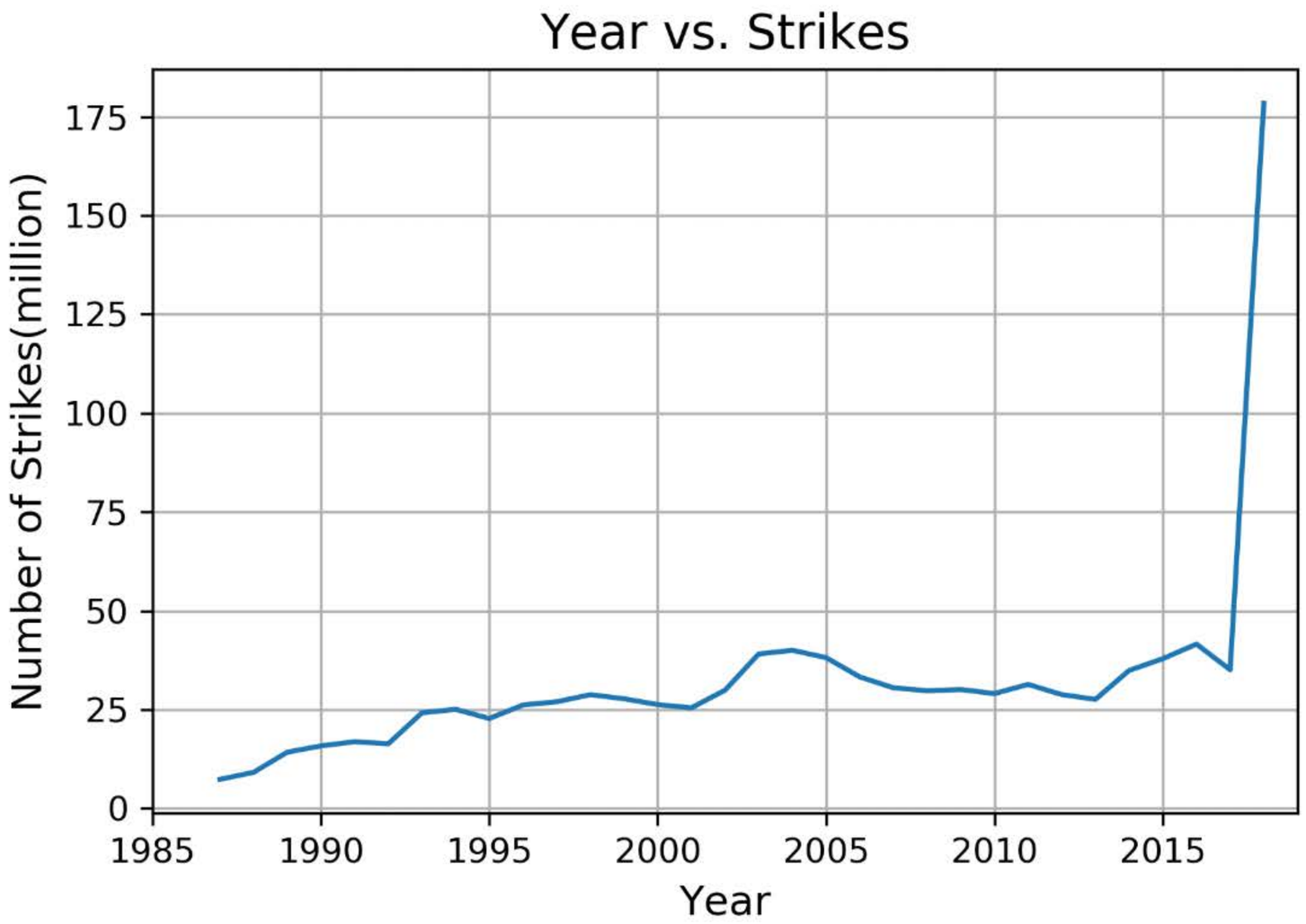
4 Figures

4.1 Year vs Strikes

Slide Type Slide

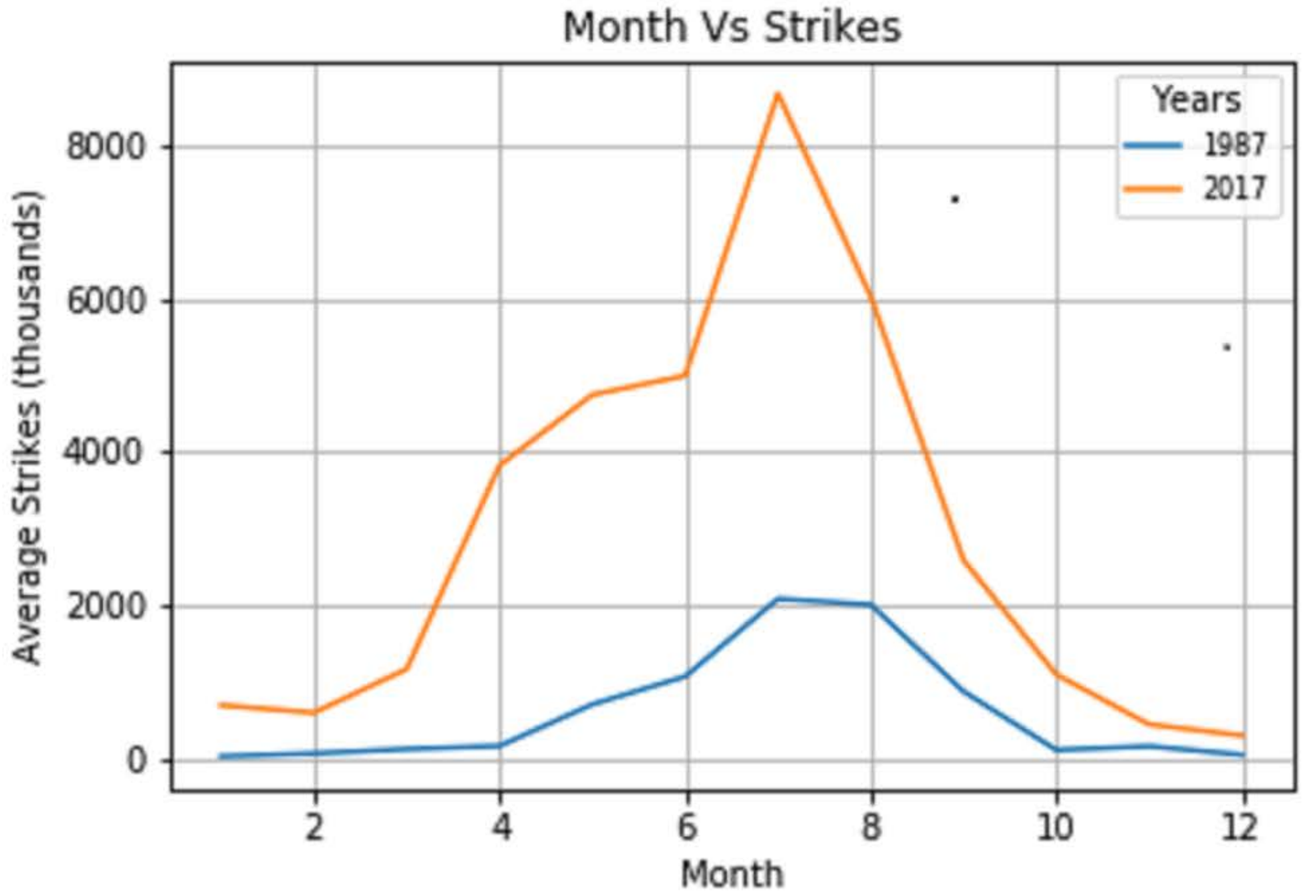
4 Figures

4.1 Year vs Strikes



Slide Type Slide

4.2 Month vs. Strikes



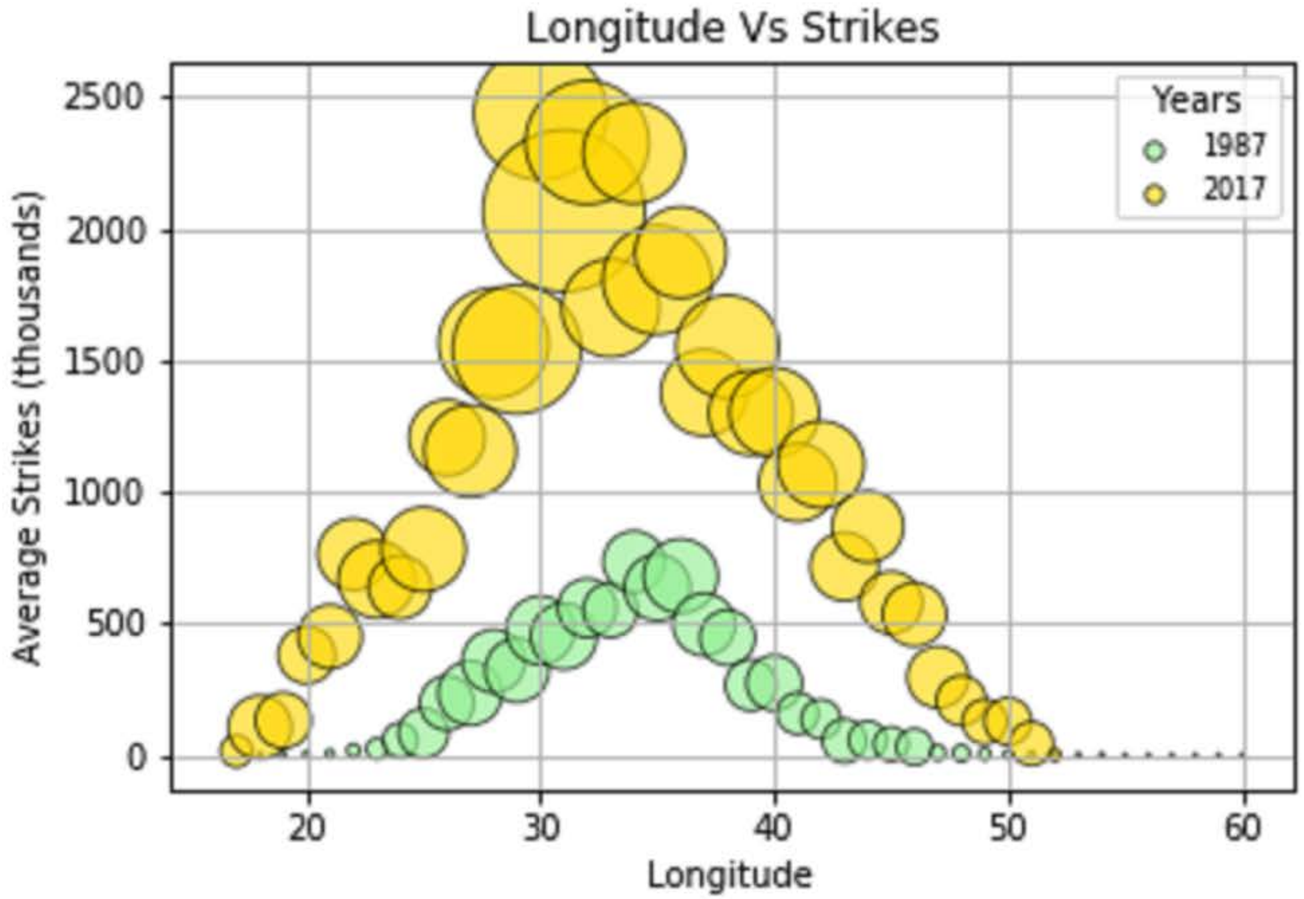
Slide Type Slide

4.3 Longitude vs Strikes

Month

Slide Type Slide

4.3 Longitude vs Strikes

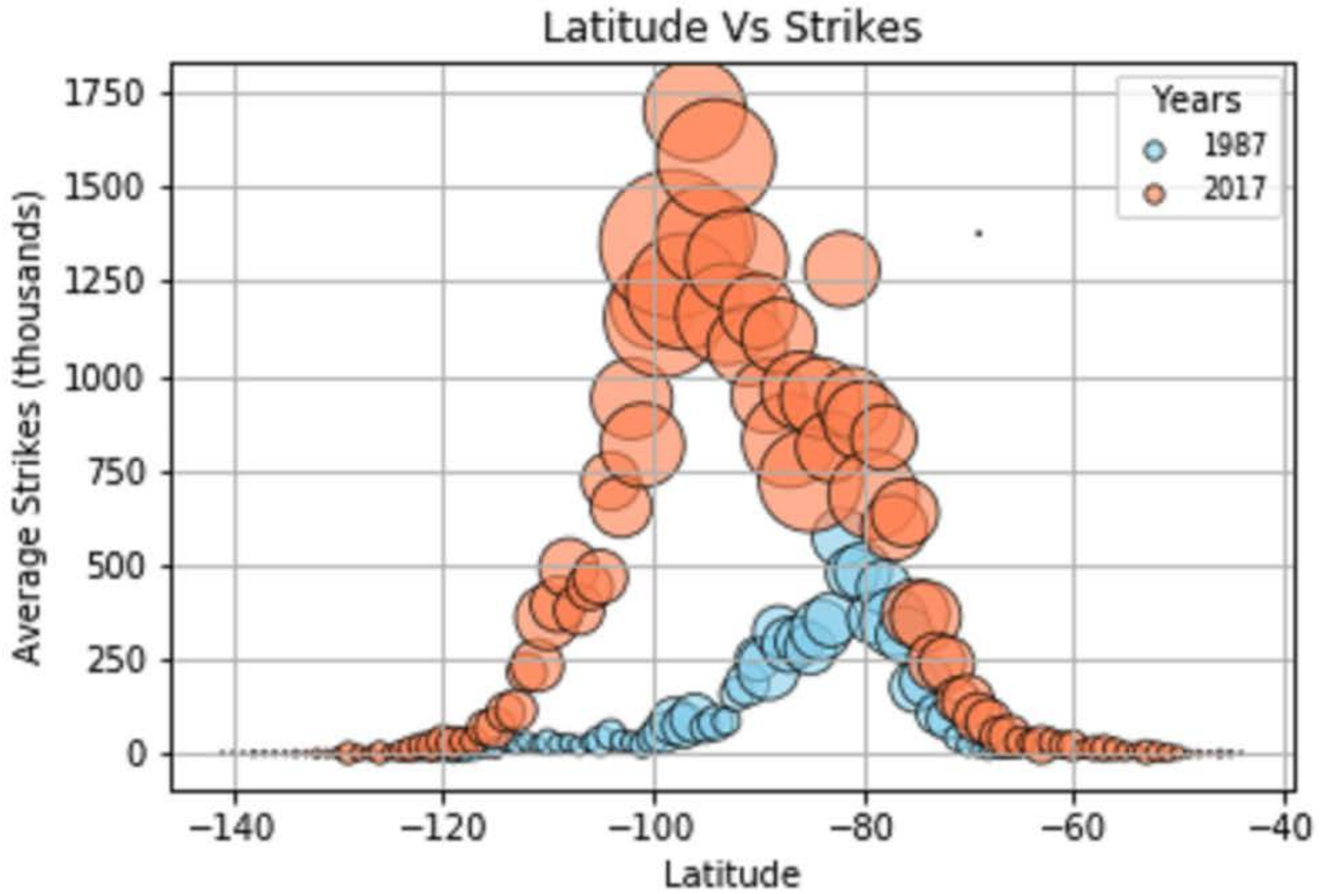


Slide Type Slide

4.4 Latitude vs Strikes

Slide Type Slide

4.4 Latitude vs Strikes



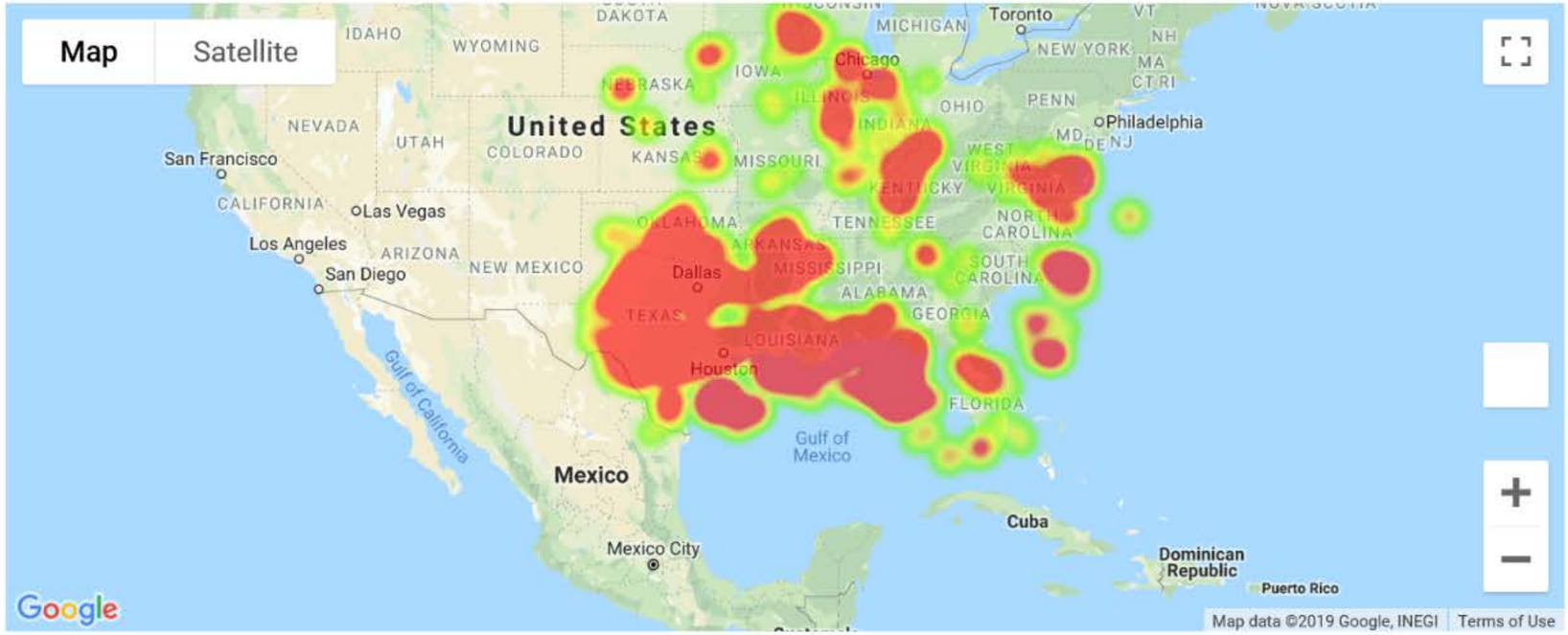
Slide Type Slide

4.5 Spring Lightning Strikes

Latitude

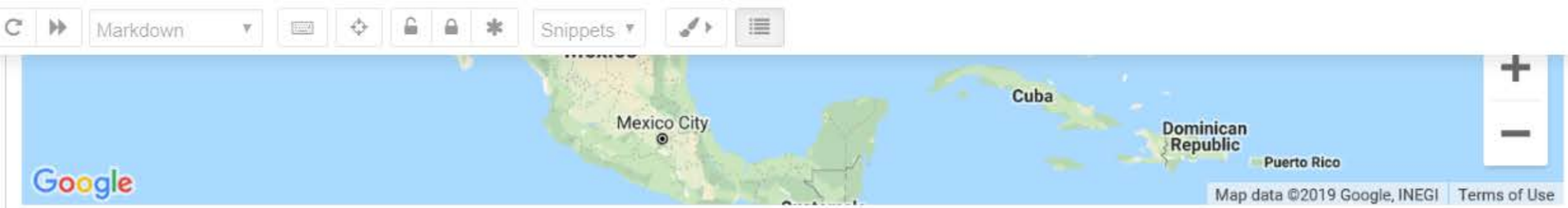
Slide Type Slide

4.5 Spring Lightning Strikes



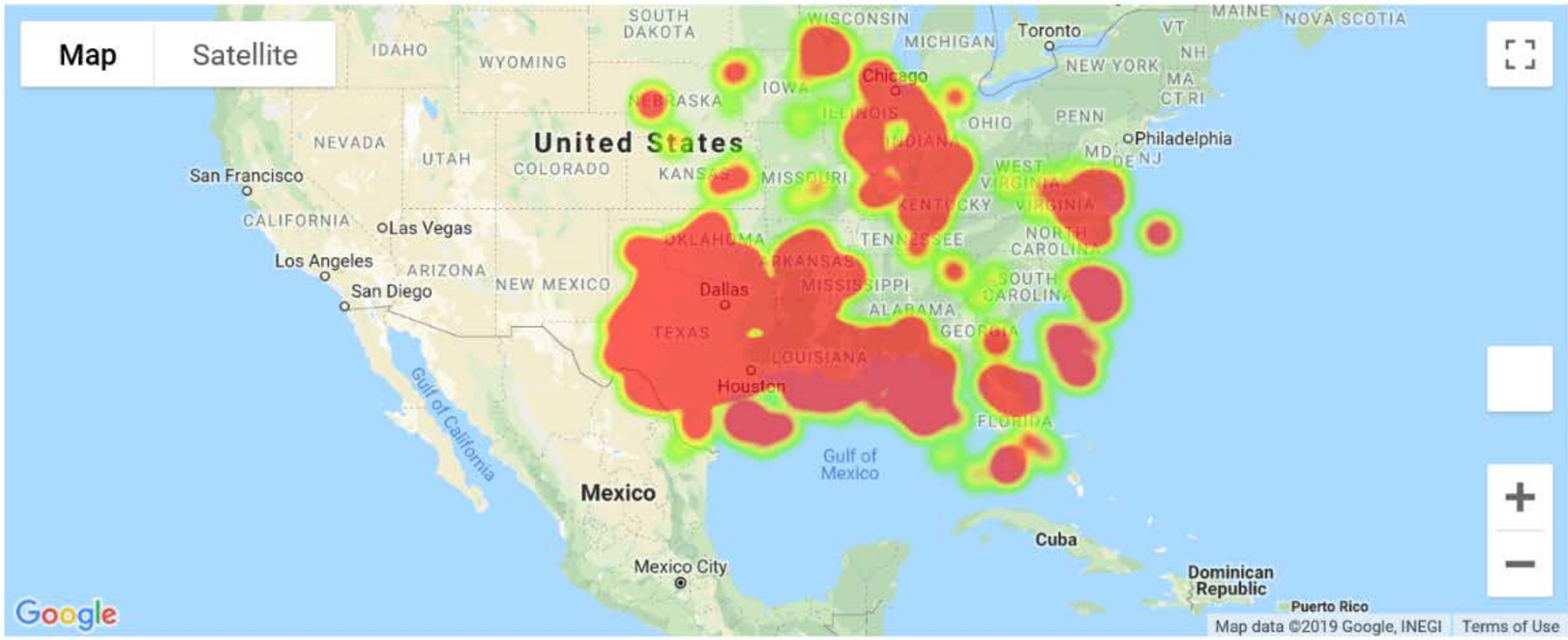
Slide Type Slide

4.6 Summer Lightning Strikes



Slide Type Slide

4.6 Summer Lightning Strikes

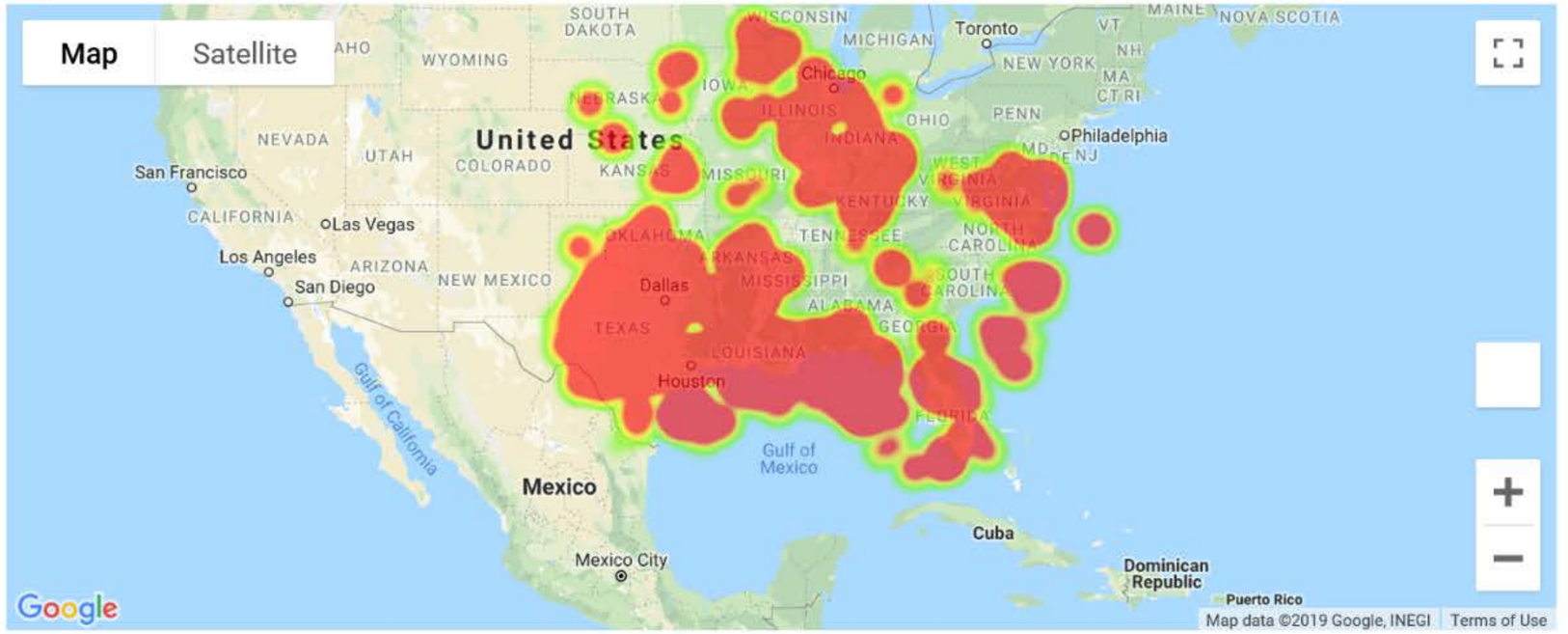


Slide Type Slide

4.7 Fall Lightning Strikes

Slide Type Slide

4.7 Fall Lightning Strikes

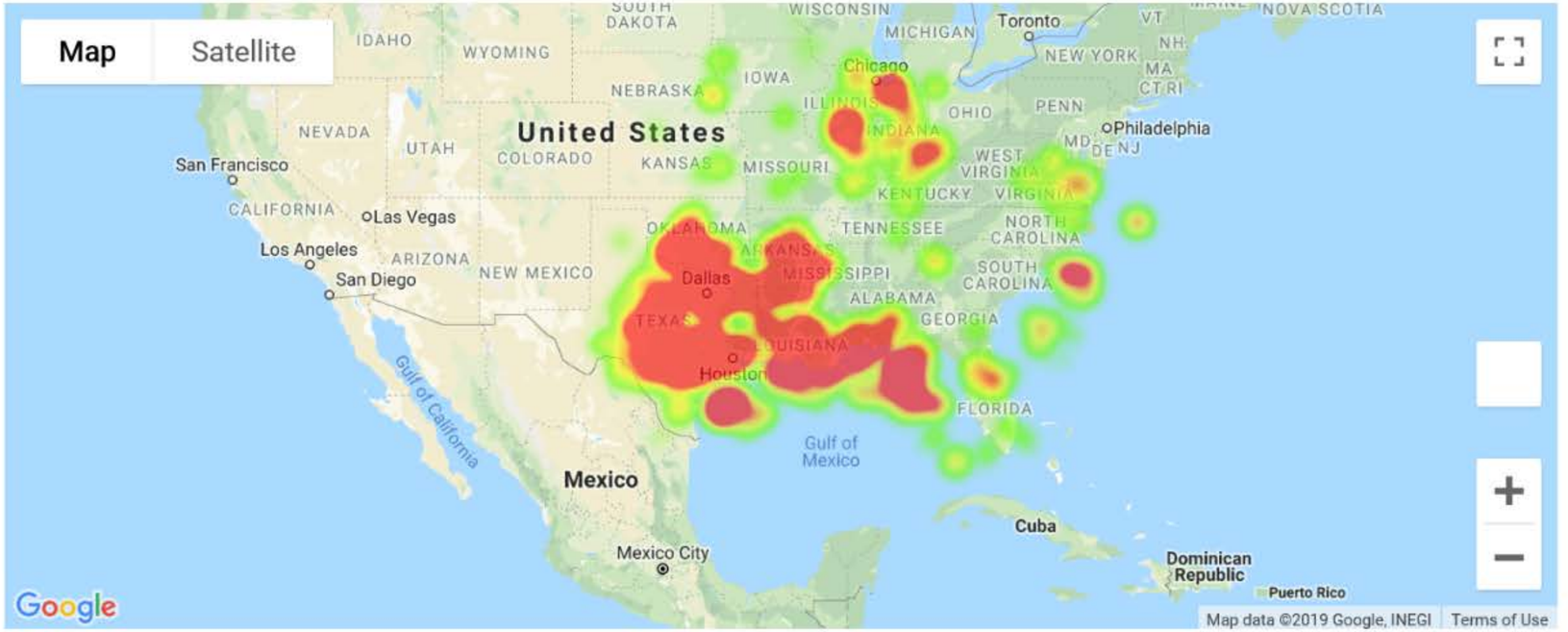


Slide Type Slide

4.8 Winter Lightning Strikes

Slide Type Slide

4.8 Winter Lightning Strikes



Slide Type Slide

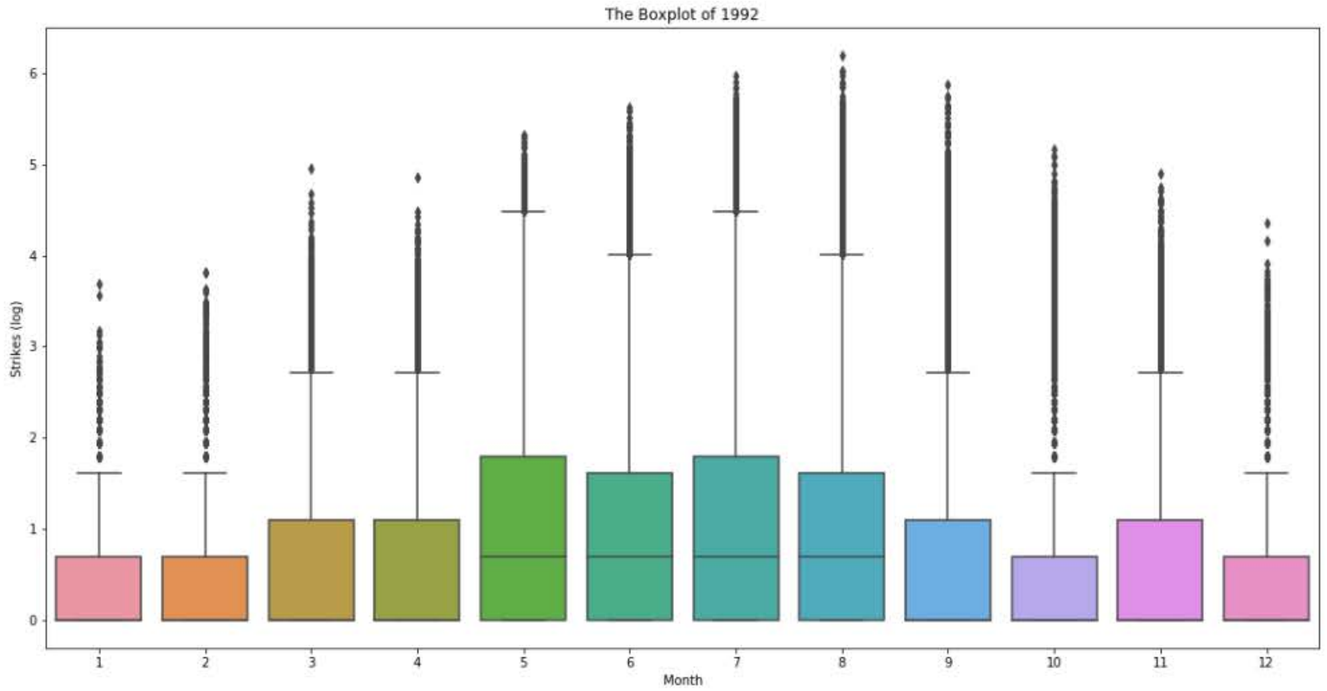
5 Statistical Test

5.1 Boxplot 1992

Slide Type Slide

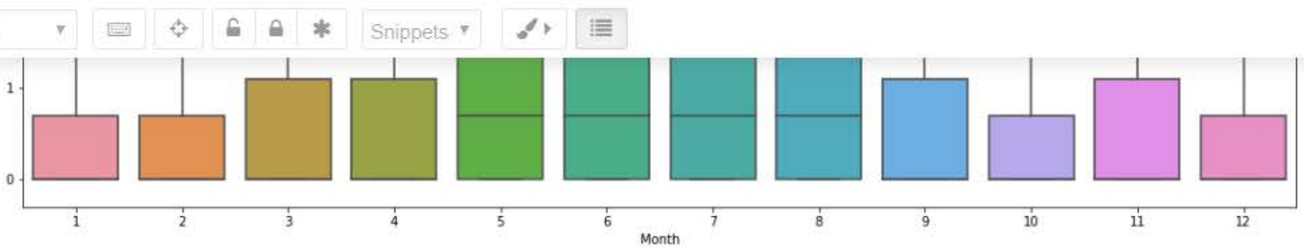
5 Statistical Test

5.1 Boxplot 1992



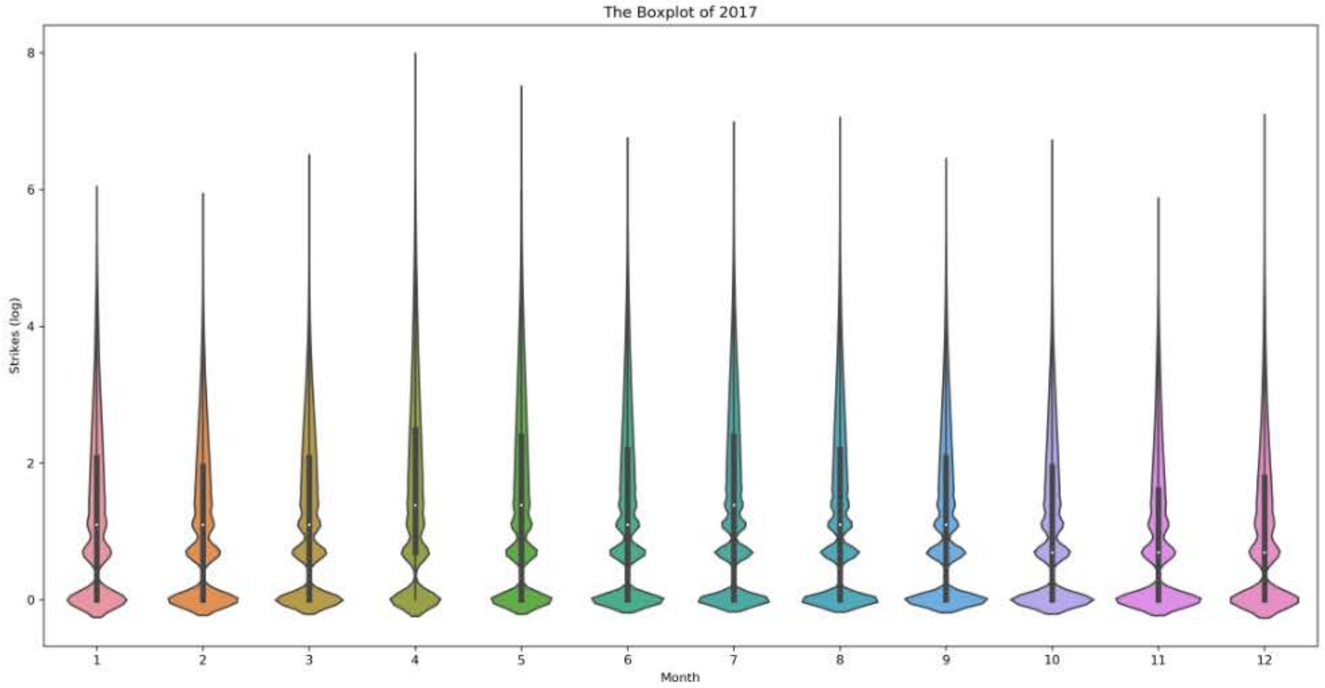
Slide Type Slide

5.2 Boxplot 2017

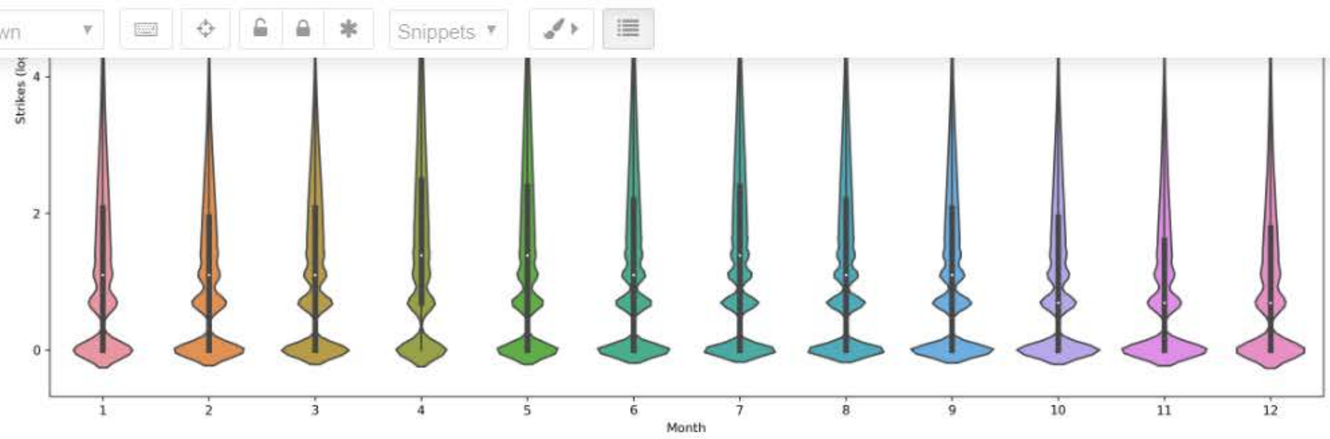


Slide Type Slide

5.2 Boxplot 2017



Slide Type



Slide Type

T-test analysis ¶

```
data87=pd.concat([strike87_m,m87],axis=1)
data87=data87.reset_index(drop=True)

data17=pd.concat([strike17_m,m17],axis=1)
data17=data17.reset_index(drop=True)
# T-test between these two years
stats.ttest_ind(data87["Strikes"],data17["Strikes"],equal_var=False)
```

```
Ttest_indResult(statistic=-2.8389165218500874, pvalue=0.014239012796034685)
```

In [ ]:

Slide Type

```
1
```

In [ ]:

Slide Type

```
1
```

Slide Type

Table of Contents

In [ ]:

Slide Type

1