MOHAMMAD ATTALA RAJAFAR
2019104824

NYTimes Covid-19 2020 with EDA

Data Mining Technique 2 (Exercise 2)

Latar Belakang

Covid-19 merupakan pandemi yang muncul pada akhir 2019 dan mulai tersebar ke berbagai wilayah di Dunia pada tahun 2020, salah satunya Amerika Serikat. Disini saya ingin melihat seberapa tingkat keparahan dari Covid-19 di berbagai wilayah Amerika Serikat serta menganalisa dan memvisualisasikannya menggunakan data yang saya dapat dari Kaggle dengan tools Jupyter Lab (Conda).

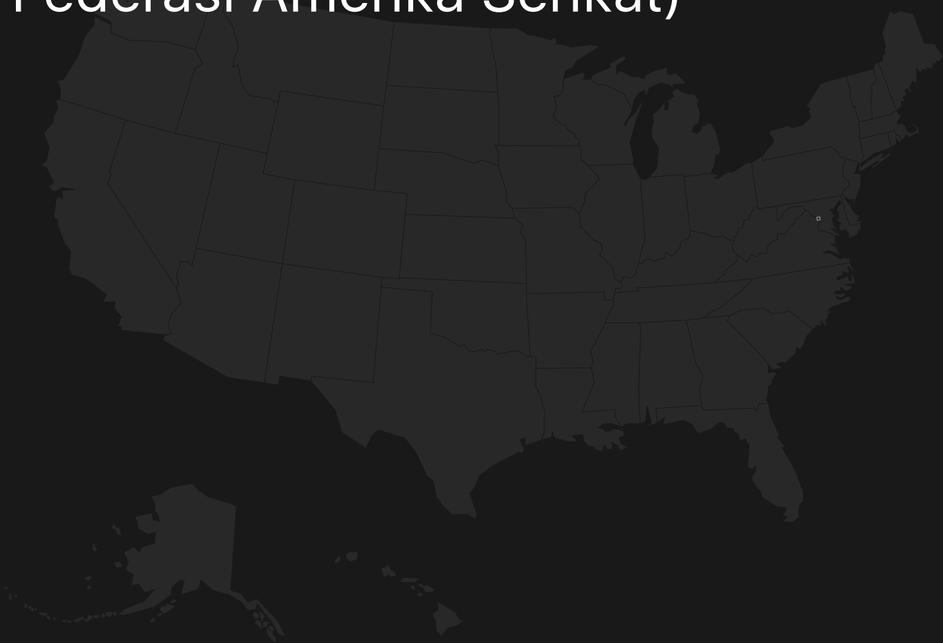
Link Data:

https://www.kaggle.com/ringhilterra17/enrichednytimescovid19?select=covid19_us_county.csv

Varibel (Fitur Data)

Data yang saya pakai ada 2 yakni US Country untuk EDA dan Visualisasi serta US POP Shapes untuk geospasial, berikut fiturnya:

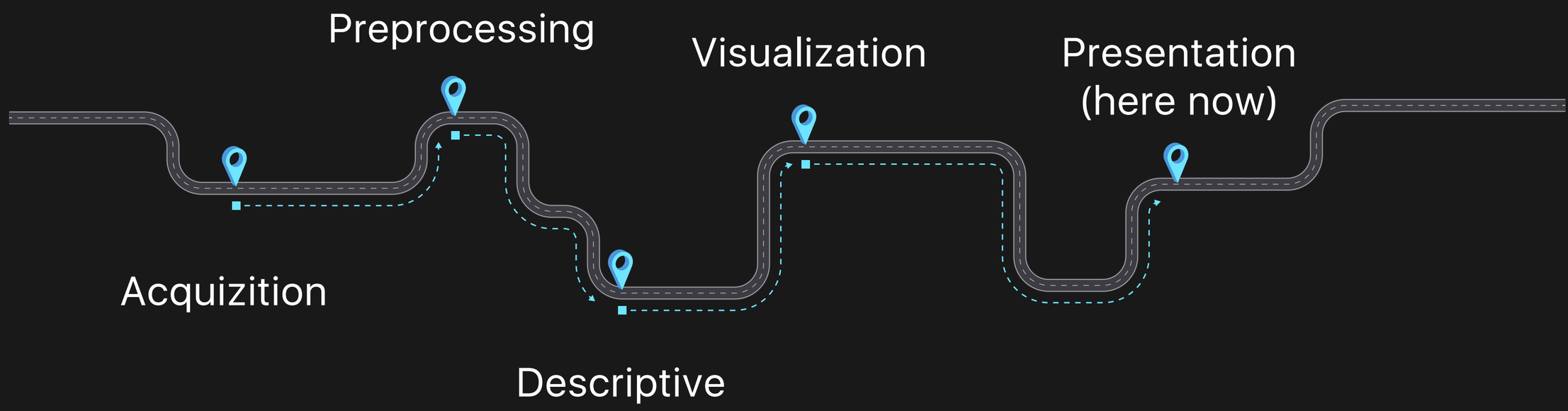
- date = Tanggal
- county = Daerah
- state = Negara Bagian
- fips = Federal Information Processing Standard (ID dari Federasi Amerika Serikat)
- state_fips = ID untuk Negara Bagian
- county_fips = ID untuk Daerah
- cases = Kasus Covid-19
- deaths = Kematian akibat Covid-19
- new_day_cases = Kasus harian baru
- new_day_deaths = Kematian harian baru



Varibel (Fitur Data)

- cases_per_capita_100k = Kasus perkapita/populasi 100k
- deaths_per_capita_100k = Kematian perkapita/populasi 100k
- new_day_cases_per_capita_100k = Kasus harian baru perkapita/populasi 100k
- new_day_deaths_per_capita_100k = Kematian harian baru perkapita/populasi 100k
- county_pop_2019_est = Estimasi populasi di daerah tahun 2019
- pop_per_sq_mile_2010 = Populasi per 16 Kilometer persegi tahun 2010

Alur Analisis



Library

- Import Library

Saya menggunakan numpy, pandas, geopandas, datetime, dan wkt untuk preprocessing, warnings untuk memfilter warning, dan untuk visualisasi menggunakan folium, matplotlib, seaborn, dan plotly

Import Library yang dibutuhkan

```
# Processing
import numpy as np # algoritma numerik
import pandas as pd # data processing
import geopandas as gpd # geo data processing
import warnings
import datetime as dt

# Visualization
import folium
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import plotly.offline as py

from shapely import wkt
```

Mengatur Warning, Print & Display

```
# Ignore the filter warning, suppress print option, & others
warnings.filterwarnings(action='ignore')
np.set_printoptions(suppress=True)
pd.set_option('display.float_format', lambda x: '%.5f' % x)
pd.options.display.max_rows = 999
```

- Warning, Print, & Display

Disini saya memfilter warning dan men set print serta display dari kolom

Exploratory Data Analysis

Negara Amerika Serikat

Acquisition -> Preprocessing -> Descriptive -> Analysis

Acquizition

- Import data

Saya menggunakan lib. pandas untuk read data dan menampilkannya.

Import & Show data teratas

```
# Read Data
usa = pd.read_csv ('covid19_us_county.csv')
usa.head()
```

	date	county	state	fips	state_fips	county_fips	cases	deaths	new_day_cases	new_day_deaths	cases_per_capita_100k	deaths_per_capita_100k	new_day_cases_per_capita_100k	n
0	2020-01-21	Snohomish	Washington	53061	53	61	1	0	0.00000	0.00000	0.12164	0.00000	0.00000	
1	2020-01-22	Snohomish	Washington	53061	53	61	1	0	0.00000	0.00000	0.12164	0.00000	0.00000	
2	2020-01-23	Snohomish	Washington	53061	53	61	1	0	0.00000	0.00000	0.12164	0.00000	0.00000	
3	2020-01-24	Cook	Illinois	17031	17	31	1	0	0.00000	0.00000	0.01942	0.00000	0.00000	
4	2020-01-24	Snohomish	Washington	53061	53	61	1	0	0.00000	0.00000	0.12164	0.00000	0.00000	

Acquizition

- Show info

Melihat shape, null, dan tipe data

Info pada data (shape, null, dan tipe data)

```
# Info
usa.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 231306 entries, 0 to 231305
Data columns (total 16 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   date                            231306 non-null  object
 1   county                          231306 non-null  object
 2   state                           231306 non-null  object
 3   fips                            231306 non-null  int64
 4   state_fips                      231306 non-null  int64
 5   county_fips                     231306 non-null  int64
 6   cases                           231306 non-null  int64
 7   deaths                          231306 non-null  int64
 8   new_day_cases                   231306 non-null  float64
 9   new_day_deaths                  231306 non-null  float64
 10  cases_per_capita_100k           231306 non-null  float64
 11  deaths_per_capita_100k          231306 non-null  float64
 12  new_day_cases_per_capita_100k   231306 non-null  float64
 13  new_day_deaths_per_capita_100k  231306 non-null  float64
```

Preprocessing

- Cleaning

Mendrop data duplikat dan Null/NaNN

Drop duplikat

```
# Drop Duplicate (checks)
duplikat = usa[usa.duplicated()]
print("Data Duplikat (Baris, Kolom) :", duplikat.shape)
```

```
Data Duplikat (Baris, Kolom) : (0, 16)
```

Terlihat bahwa data sudah bersih, sehingga proses ini selesai

Drop NaNN

```
# Drop Null (checks)
usa.isnull().sum()
```

```
date                                0
county                              0
state                               0
fips                                0
state_fips                          0
county_fips                         0
cases                               0
deaths                              0
new_day_cases                       0
new_day_deaths                      0
cases_per_capita_100k               0
deaths_per_capita_100k              0
new_day_cases_per_capita_100k       0
new_day_deaths_per_capita_100k      0
county_pop_2019_est                 0
pop_per_sq_mile_2010                0
dtype: int64
```

Terlihat bahwa tidak ada data NaNN atau Null sehingga proses selesai

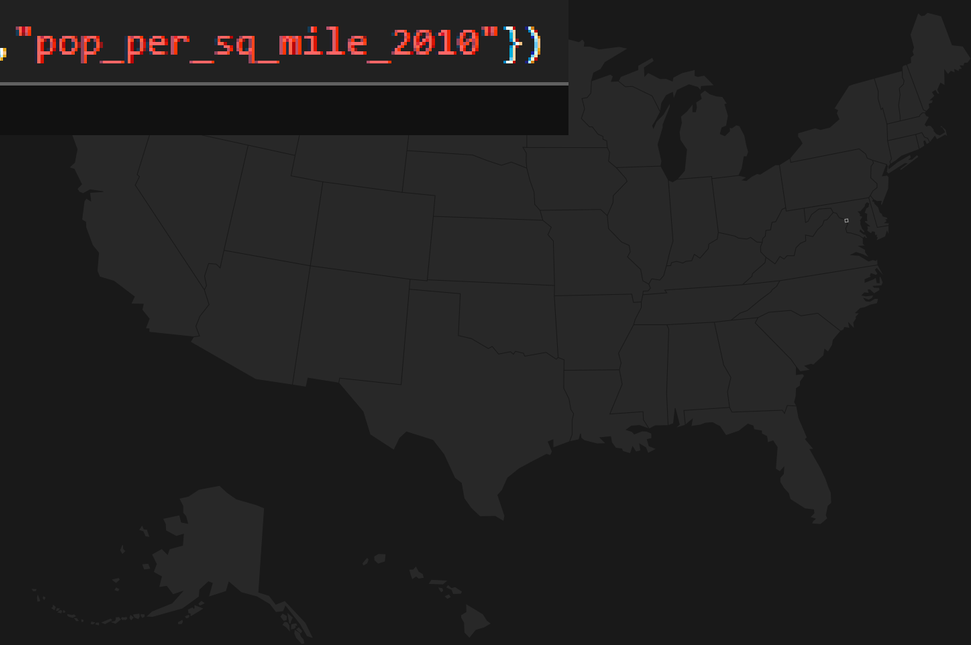
Preprocessing

- Selection

Menyeleksi fitur apa saja yang terpakai dan tidak

Mendrop Fitur tak terpakai

```
# Selection Feature
usa = usa.drop(columns = {"state_fips","county_fips","pop_per_sq_mile_2010"})
```



Descriptive

- Deskriptif Analis

Menganalisa data menggunakan 5 summary

Summary data di data US (Count, Mean, Standard Deviation, Minimum, Q1, Q2, Q3, & Max)

```
# Describe
usa.describe()
```

	fips	cases	deaths	new_day_cases	new_day_deaths	cases_per_capita_100k	deaths_per_capita_100k	new_day_cases_per_capita_100k	new_day_deaths_per_capita_100k
count	231306.00000	231306.00000	231306.00000	231306.00000	231306.00000	231306.00000	231306.00000	231306.00000	231306.00000
mean	30086.73207	391.81216	22.07716	8.94557	0.50059	194.38991	7.78846	5.26205	0.20580
std	15341.24617	3677.65002	330.64643	75.27381	7.71354	455.35770	20.75328	30.14548	1.23066
min	1001.00000	1.00000	0.00000	0.00000	0.00000	0.00996	0.00000	0.00000	0.00000
25%	18085.00000	5.00000	0.00000	0.00000	0.00000	23.95443	0.00000	0.00000	0.00000
50%	29061.00000	21.00000	0.00000	0.00000	0.00000	65.71502	0.00000	0.00000	0.00000
75%	45041.00000	100.00000	3.00000	3.00000	0.00000	181.90955	5.94689	4.61794	0.00000
max	56045.00000	214242.00000	21551.00000	8021.00000	1221.00000	12920.95002	354.73572	7949.30876	106.42072

Descriptive

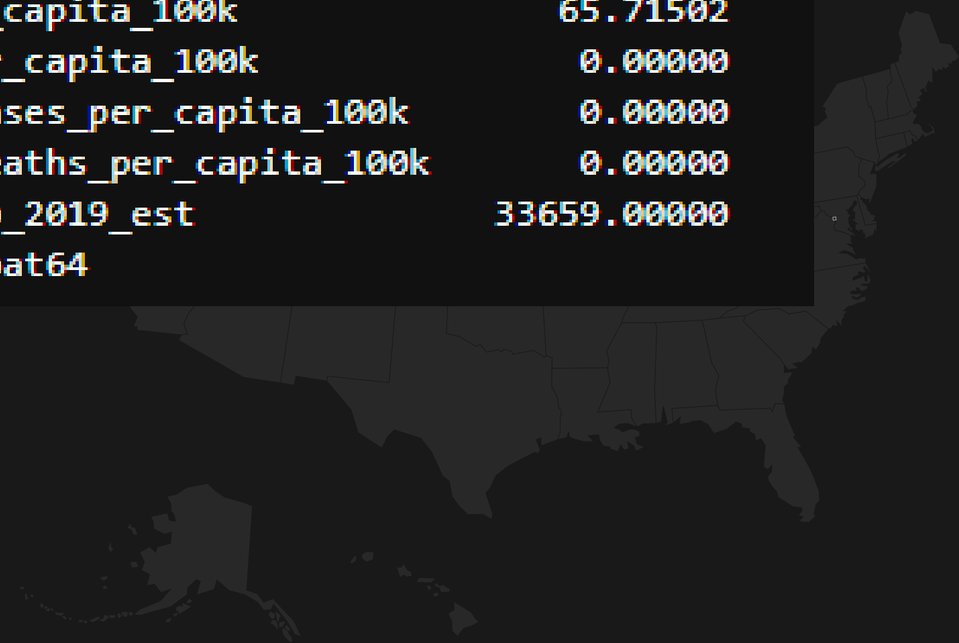
Median

Nilai Tengah pada setiap kolom yang bertipe data numerik

Median Pada data US

```
# Median
usa.median()
```

```
fips                              29061.00000
cases                                21.00000
deaths                                0.00000
new_day_cases                         0.00000
new_day_deaths                        0.00000
cases_per_capita_100k                65.71502
deaths_per_capita_100k                0.00000
new_day_cases_per_capita_100k         0.00000
new_day_deaths_per_capita_100k        0.00000
county_pop_2019_est               33659.00000
dtype: float64
```



Descriptive

IQR

Melihat ukuran variabilitas yang didasarkan pada pembagian kumpulan data menjadi kuartil di setiap kolom bertipe numerik

IQR

```
# IQR
Q1 = usa.quantile(0.25)
Q3 = usa.quantile(0.75)
IQR = Q3 - Q1

IQR
```

```
fips                              26956.00000
cases                                95.00000
deaths                                3.00000
new_day_cases                         3.00000
new_day_deaths                        0.00000
cases_per_capita_100k               157.95512
deaths_per_capita_100k                5.94689
new_day_cases_per_capita_100k         4.61794
new_day_deaths_per_capita_100k        0.00000
county_pop_2019_est               78048.00000
dtype: float64
```

Descriptive

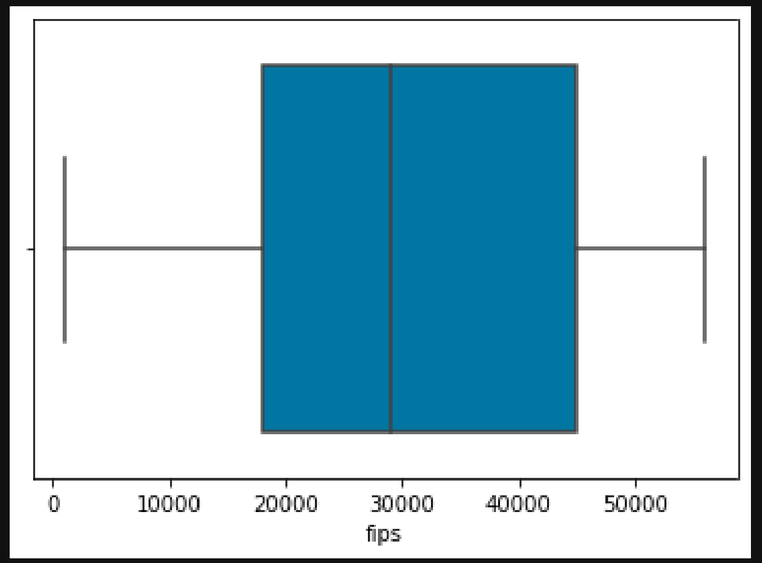
- Distribusi data

Menggunakan boxplot + outliers detection pada salah satu fitur (FIPS)

FIPS (Federal Information Processing Standards)

```
# Boxplot
sns.boxplot(x = usa['fips'])
```

```
<AxesSubplot:xlabel='fips'>
```



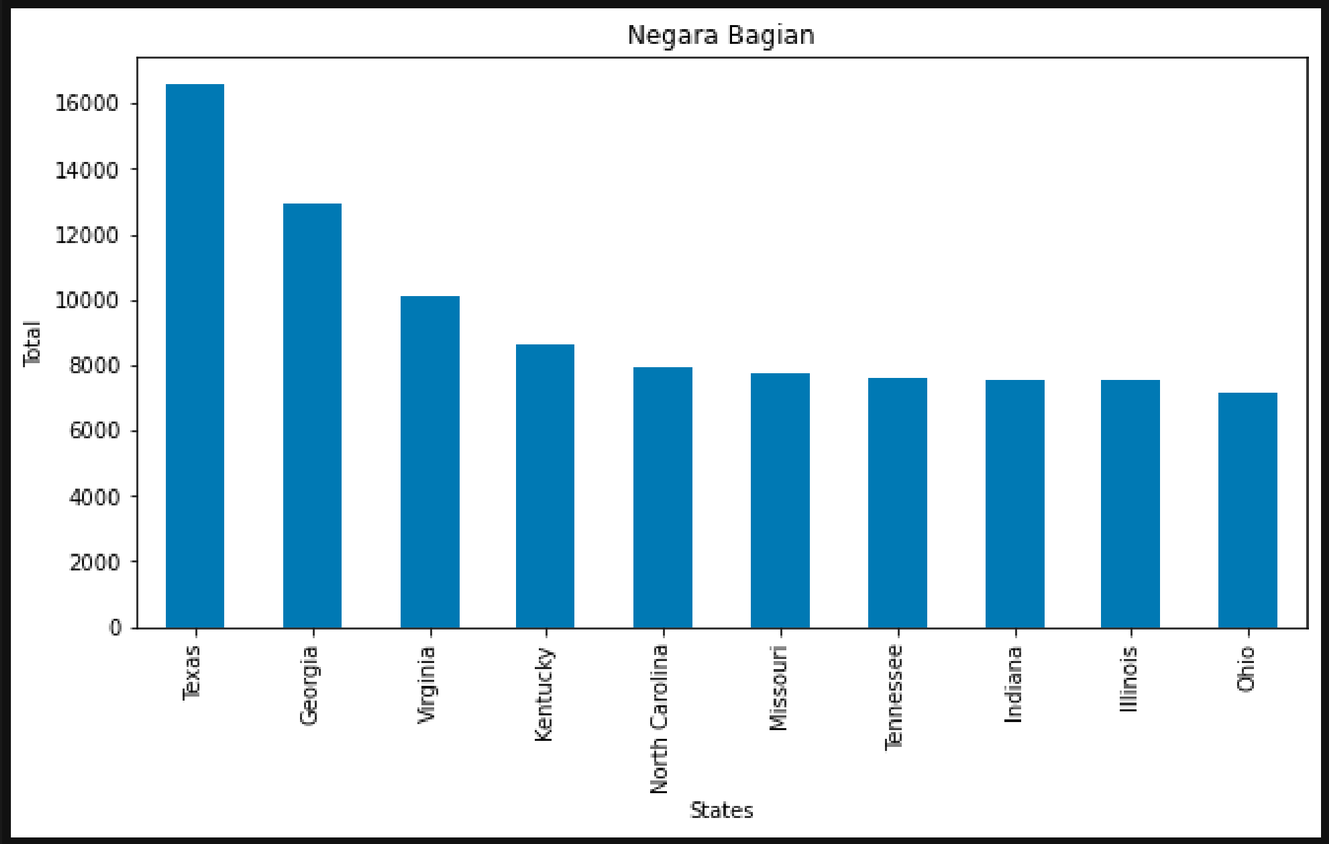

```
# Outliers Detection

Q1 = usa["fips"].quantile(0.25)
Q3 = usa["fips"].quantile(0.75)
IQR = Q3 - Q1

lower_bound = Q1 - (1.5 * IQR)
upper_bound = Q3 + (1.5 * IQR)

print("Batas Bawah = ", lower_bound)
print("Batas Atas = ", upper_bound)
print("Jadi apapun yang berada diluar", lower_bound, " dan", upper_bound, " merupakan Outlier")
```

```
Batas Bawah =  -22349.0
Batas Atas =  85475.0
Jadi apapun yang berada diluar -22349.0  dan 85475.0  merupakan Outlier
```

Descriptive

Menggunakan histogram pada salah satu fitur (State : Negara Bagian)

```
Text(0, 0.5, 'Total')
```

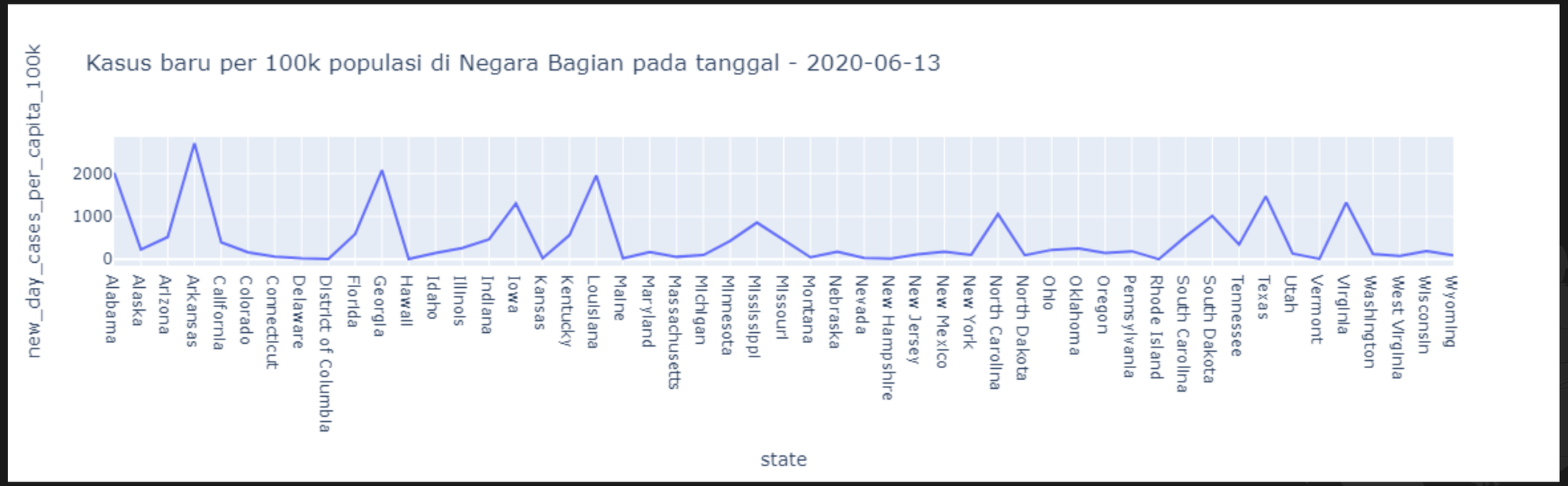


Terlihat bahwa Texas merupakan negara bagian dengan frekuensi kemunculan variabel terbanyak

Visualization

- Line chart

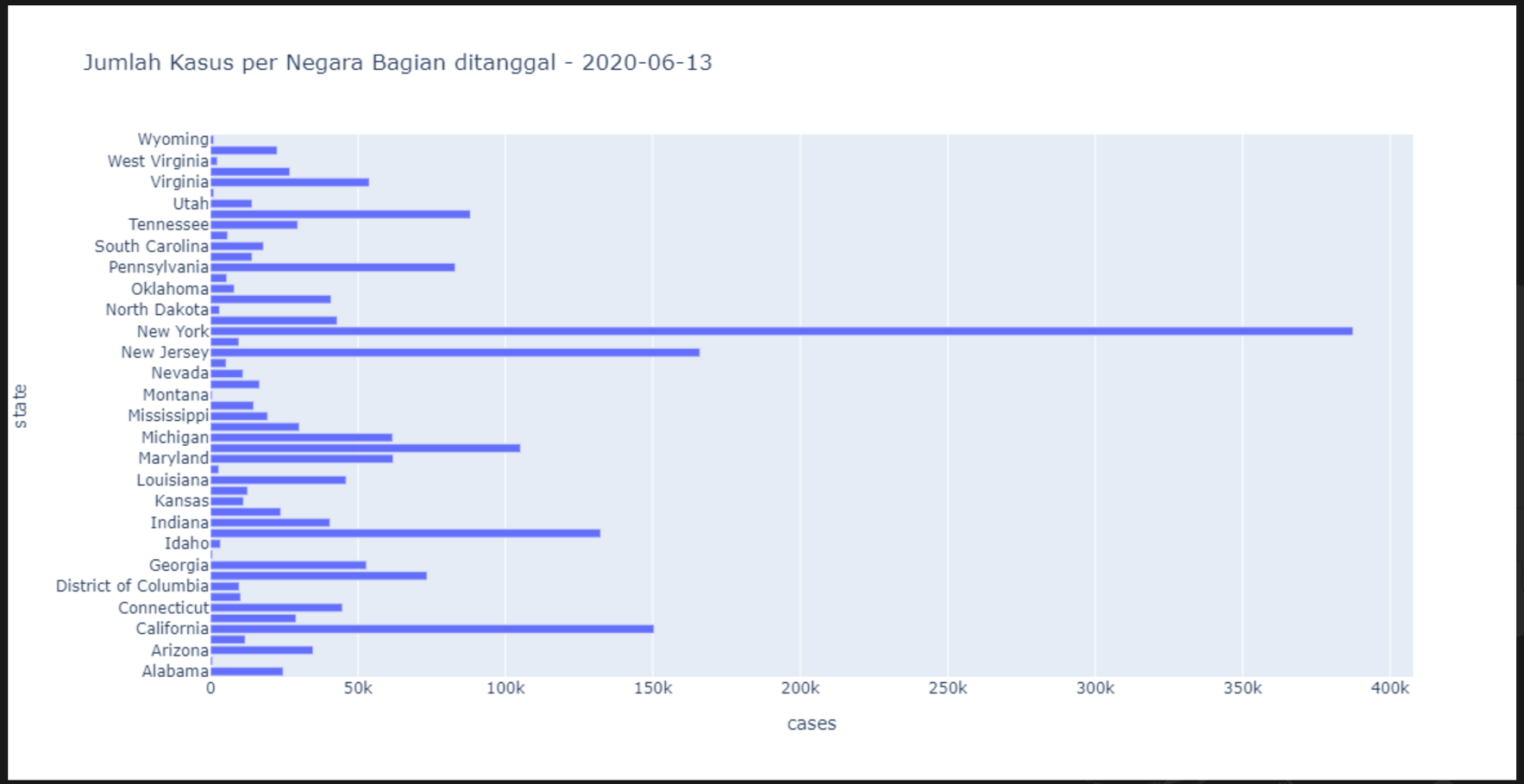
Melihat Negara Bagian terbanyak dengan kasus harian baru terbanyak per kapita 100k



Visualization

- Bar chart

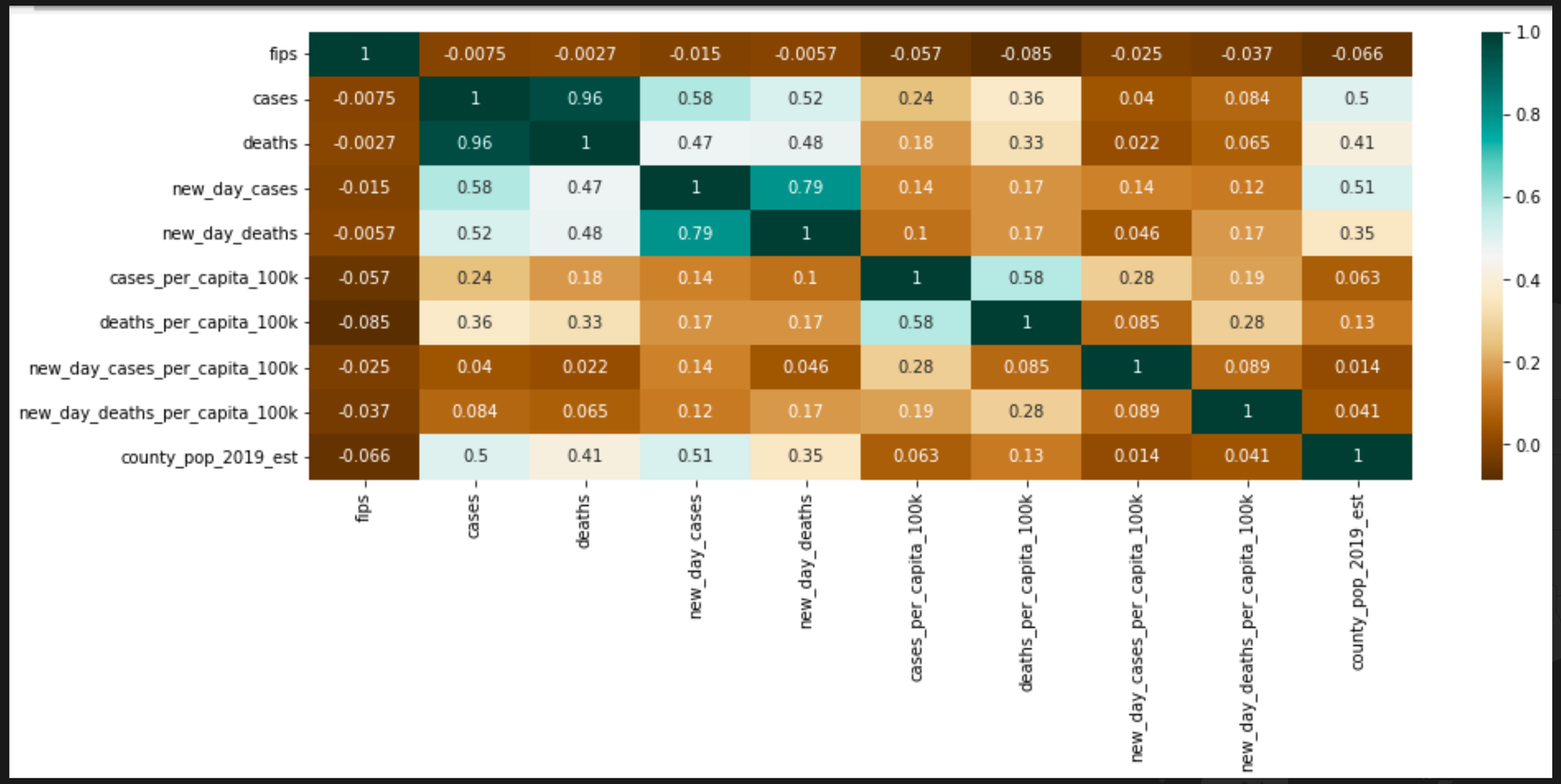
Melihat Negara bagian dengan kasus terbanyak



Visualization

- Heatmaps

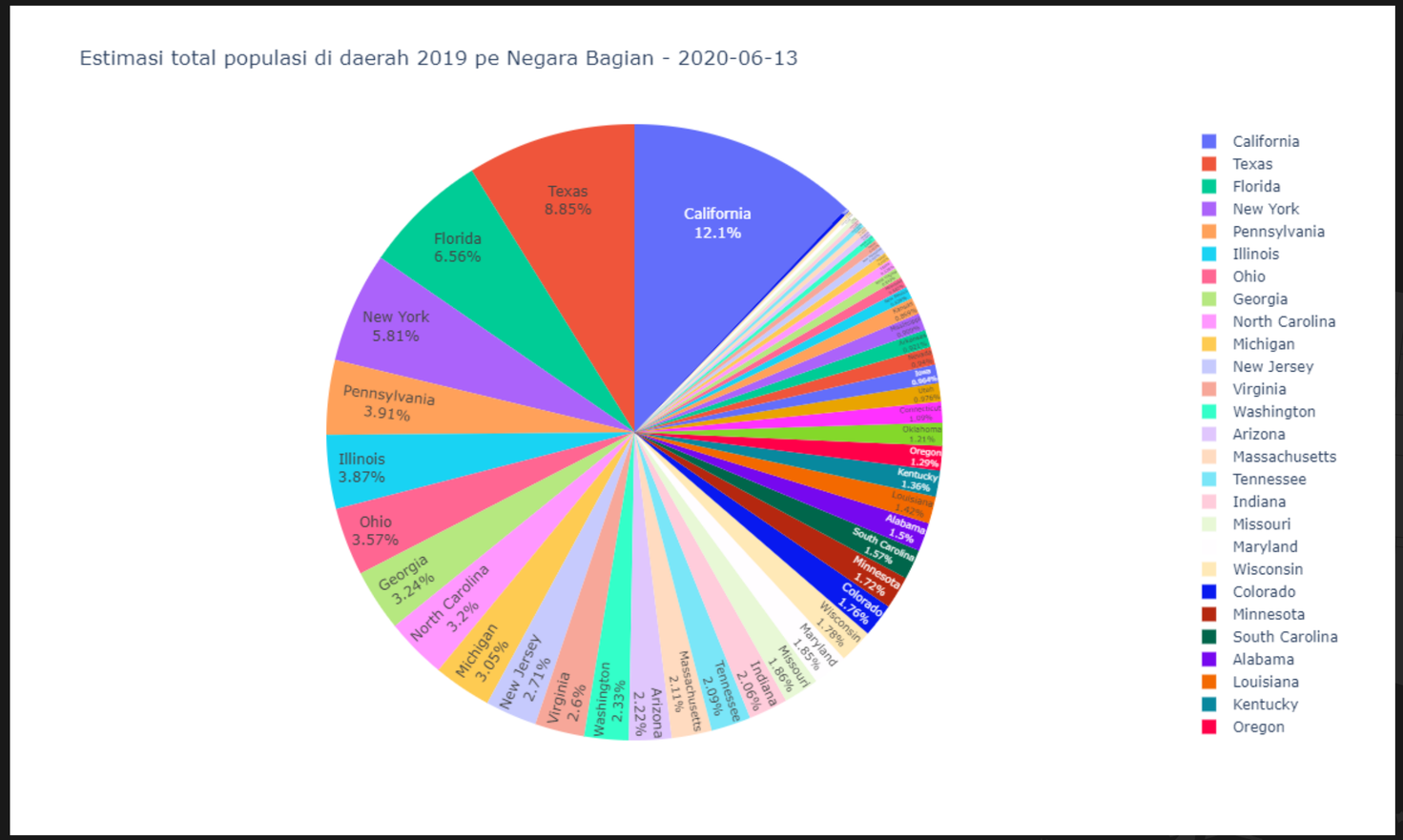
Melihat korelasi antar kolom pada dataframe



Visualization

- Pie Chart

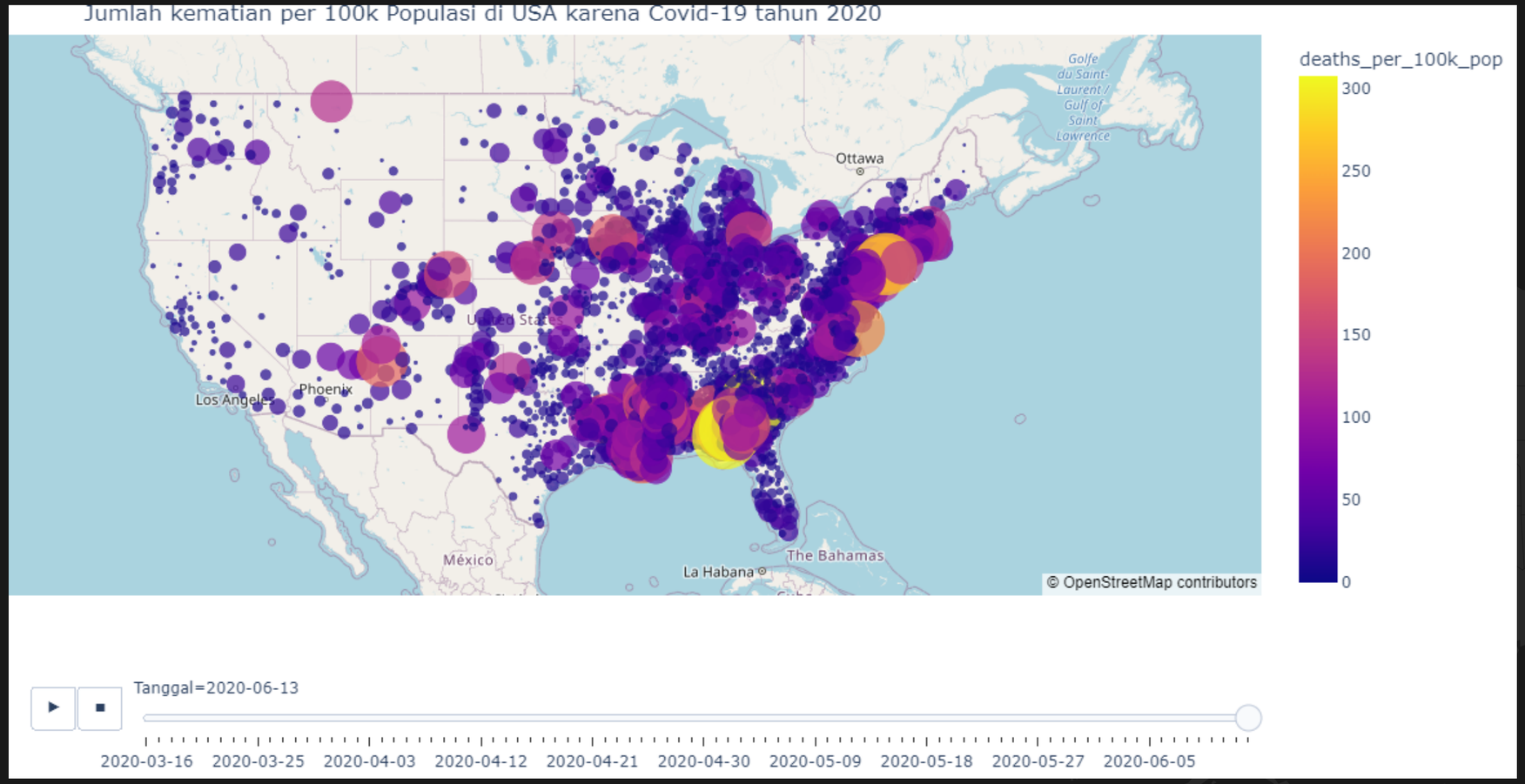
Melihat Estimasi populasi di daerah tahun 2019 di Negara Bagian



Visualization

- Scatter Geomap (Plotly)

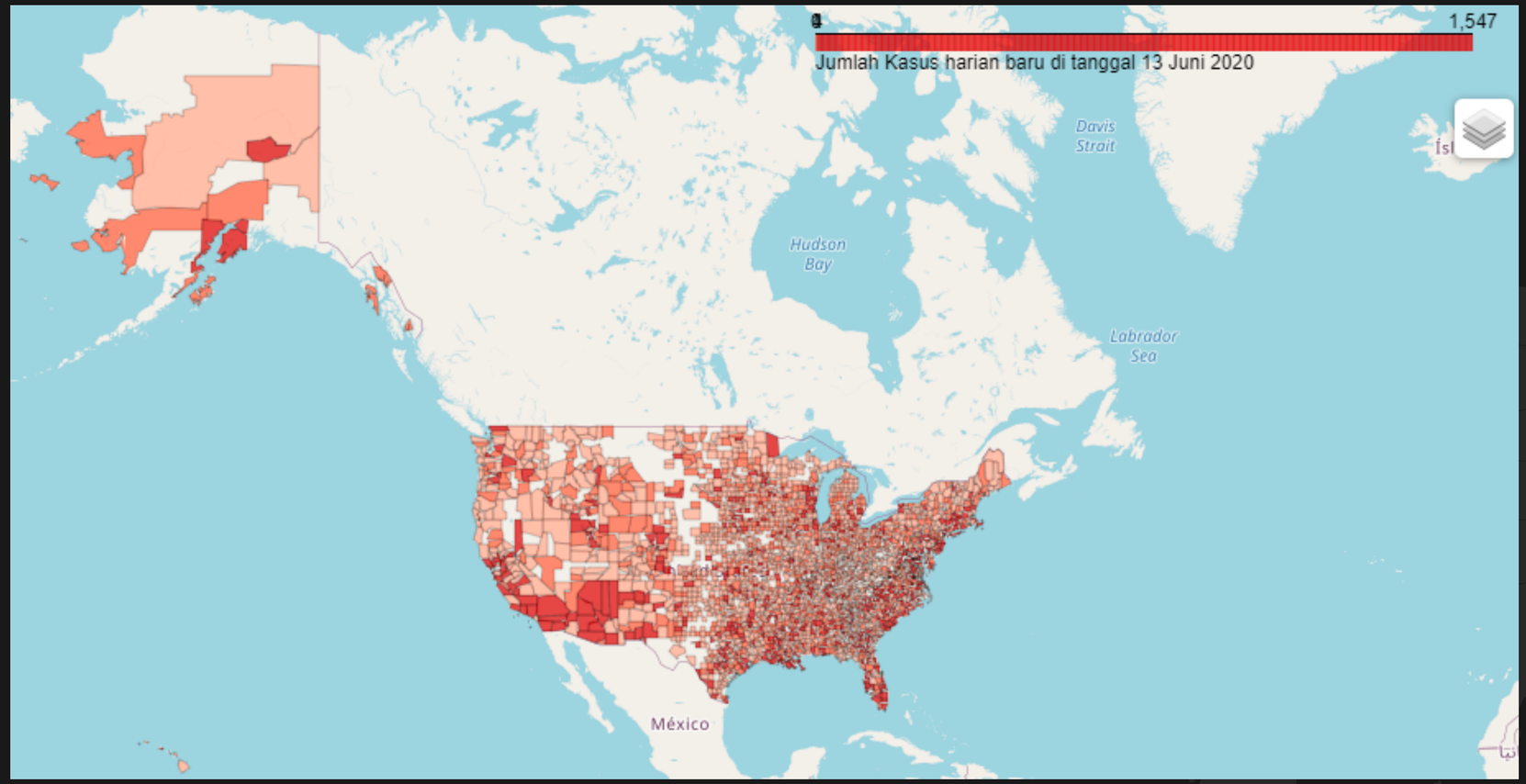
Melihat kematian per 100k populasi di US berdasarkan Negara Bagian



Visualization

- Scatter Choropleth (Folium)

Melihat kasus harian baru ditanggal 13 Juni 2020




TERIMA
KASIH