

## Libraries

```
In [1]: import pandas as pd  
import os  
import us  
from us import states  
import plotly.graph_objects as go  
import plotly.express as px  
import requests  
import dotenv  
  
os.chdir('C:\\Users\\nguye\\Documents\\UVA\\Term 3\\Bayesian Machine Learning\\Project\\Data')  
os.getcwd()
```

```
Out[1]: 'C:\\Users\\nguye\\Documents\\UVA\\Term 3\\Bayesian Machine Learning\\Project\\Data'
```

```
In [2]: dotenv.load_dotenv('.env')
```

```
Out[2]: True
```

```
In [3]: censuskey = os.getenv('censuskey')
```

## Loading Data

```
In [4]: cdc_lyme = pd.read_csv('LD-Case-Counts-by-County-00-19.csv', encoding = 'latin1')
```

```
In [5]: cdc_lyme
```

```
Out[5]:
```

	Ctyname	Sname	STCODE	CTYCODE	Cases2000	Cases2001	Cases2002	Cases2003	Cases2004	Cases2005	Cases2010	Cases2011
0	Autauga County	Alabama	1	1	0	0	0	0	0	0	0	0
1	Baldwin County	Alabama	1	3	1	0	1	0	0	0	0	1
2	Barbour County	Alabama	1	5	0	0	0	0	0	0	0	1
3	Bibb County	Alabama	1	7	0	0	0	0	0	0	0	0
4	Blount County	Alabama	1	9	0	0	0	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...
3188	Teton County	Wyoming	56	39	0	0	1	0	1	1	0	0
3189	Uinta County	Wyoming	56	41	0	0	0	0	0	0	0	0
3190	Washakie County	Wyoming	56	43	0	0	0	0	0	0	0	0
3191	Weston County	Wyoming	56	45	0	0	0	0	0	0	0	0
3192	Wyoming	Wyoming	56	999	0	0	0	0	0	0	0	1

3193 rows × 24 columns

## Data Wrangling

Per the [CDC](#), code '999' represents unknown county within a single state so we'll compare the total number cases per county each year and see how much of it represents county.

```
In [6]: cdc_lyme[cdc_lyme['CTYCODE'] == '999']
```

```
Out[6]:
```

	Ctyname	Sname	STCODE	CTYCODE	Cases2000	Cases2001	Cases2002	Cases2003	Cases2004	Cases2005	Cases2010	Cases2011

0 rows × 24 columns

```
In [7]: cdc_lyme[cdc_lyme['CTYCODE'] != '999']
```

```
Out[7]:
```

	Ctyname	Sname	STCODE	CTYCODE	Cases2000	Cases2001	Cases2002	Cases2003	Cases2004	Cases2005	Cases2010	Cases2011
0	Autauga County	Alabama	1	1	0	0	0	0	0	0	0	0
1	Baldwin County	Alabama	1	3	1	0	1	0	0	0	0	1
2	Barbour County	Alabama	1	5	0	0	0	0	0	0	0	1
3	Bibb County	Alabama	1	7	0	0	0	0	0	0	0	0
4	Blount County	Alabama	1	9	0	0	0	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...
3188	Teton County	Wyoming	56	39	0	0	1	0	1	1	0	0
3189	Uinta County	Wyoming	56	41	0	0	0	0	0	0	0	0
3190	Washakie County	Wyoming	56	43	0	0	0	0	0	0	0	0
3191	Weston County	Wyoming	56	45	0	0	0	0	0	0	0	0
3192	Wyoming	Wyoming	56	999	0	0	0	0	0	0	0	1

3193 rows × 24 columns

```
In [8]: cdc_lyme['CTYCODE'] = cdc_lyme['CTYCODE'].astype(str).str.zfill(3)  
cdc_lyme['STCODE'] = cdc_lyme['STCODE'].astype(str).str.zfill(2)  
cdc_lyme['FIPS'] = cdc_lyme['STCODE'] + cdc_lyme['CTYCODE']  
cdc_lyme['stabbr'] = cdc_lyme['STCODE'].map(us.states.mapping('fips', 'abbr'))  
state_abbr = cdc_lyme['stabbr'].unique()
```

```
Out[8]:
```

	Ctyname	Sname	STCODE	CTYCODE	Cases2000	Cases2001	Cases2002	Cases2003	Cases2004	Cases2005	Cases2010	Cases2011
0	Autauga County	Alabama	1	1	0	0	0	0	0	0	0	0
1	Baldwin County	Alabama	1	3	1	0	1	0	0	0	0	1
2	Barbour County	Alabama	1	5	0	0	0	0	0	0	0	1
3	Bibb County	Alabama	1	7	0	0	0	0	0	0	0	0
4	Blount County	Alabama	1	9	0	0	0	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...
3188	Teton County	Wyoming	56	39	0	0	1	0	1	1	0	0
3189	Uinta County	Wyoming	56	41	0	0	0	0	0	0	0	0
3190	Washakie County	Wyoming	56	43	0	0	0	0	0	0	0	0
3191	Weston County	Wyoming	56	45	0	0	0	0	0	0	0	0
3192	Wyoming	Wyoming	56	999	0	0	0	0	0	0	0	1

3193 rows × 24 columns

```
In [9]: state_count_2000 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2000'].sum() for st in state_abbr]  
state_count_2001 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2001'].sum() for st in state_abbr]  
state_count_2002 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2002'].sum() for st in state_abbr]  
state_count_2003 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2003'].sum() for st in state_abbr]  
state_count_2004 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2004'].sum() for st in state_abbr]  
state_count_2005 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2005'].sum() for st in state_abbr]  
state_count_2006 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2006'].sum() for st in state_abbr]  
state_count_2007 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2007'].sum() for st in state_abbr]  
state_count_2008 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2008'].sum() for st in state_abbr]  
state_count_2009 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2009'].sum() for st in state_abbr]  
state_count_2010 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2010'].sum() for st in state_abbr]  
state_count_2011 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2011'].sum() for st in state_abbr]  
state_count_2012 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2012'].sum() for st in state_abbr]  
state_count_2013 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2013'].sum() for st in state_abbr]  
state_count_2014 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2014'].sum() for st in state_abbr]  
state_count_2015 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2015'].sum() for st in state_abbr]  
state_count_2016 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2016'].sum() for st in state_abbr]  
state_count_2017 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2017'].sum() for st in state_abbr]  
state_count_2018 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2018'].sum() for st in state_abbr]  
state_count_2019 = [cdc_lyme[cdc_lyme['stabbr'] == st]['Cases2019'].sum() for st in state_abbr]
```

```
In [10]: cases_year = ['Cases2000', 'Cases2001',  
                  'Cases2002', 'Cases2003', 'Cases2004', 'Cases2005', 'Cases2006',  
                  'Cases2007', 'Cases2008', 'Cases2009', 'Cases2010', 'Cases2011',  
                  'Cases2012', 'Cases2013', 'Cases2014', 'Cases2015', 'Cases2016',  
                  'Cases2017', 'Cases2018', 'Cases2019']  
sum = []
```

```
In [11]: count_summary = pd.DataFrame({'State': state_abbr,  
                                    '2000': state_count_2000,  
                                    '2001': state_count_2001,  
                                    '2002': state_count_2002,  
                                    '2003': state_count_2003,  
                                    '2004': state_count_2004,  
                                    '2005': state_count_2005,  
                                    '2006': state_count_2006,  
                                    '2007': state_count_2007,  
                                    '2008': state_count_2008,  
                                    '2009': state_count_2009,  
                                    '2010': state_count_2010,  
                                    '2011': state_count_2011,  
                                    '2012': state_count_2012,  
                                    '2013': state_count_2013,  
                                    '2014': state_count_2014,  
                                    '2015': state_count_2015,  
                                    '2016': state_count_2016,  
                                    '2017': state_count_2017,  
                                    '2018': state_count_2018,  
                                    '2019': state_count_2019})
```

```
In [12]: count_summary
```

```
Out[12]:
```

	State	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
0	AL	6	10	11	8	6	3	38	13	9	...	2	24	25	24	64	25	38	41	36	66
1	AK	2	2	3	3	3	4	15	10	6	...	7	11	10	14	8	9	15	10	11	3
2	AZ	2	3	4	4	13	10	13	2	8	...	2	15	13	32	21	12	13	28	7	10
3	AR	7	4	3	0	0	0	2	1	0	...	0	0	0	0	0	0	0	2	6	4
4	CA	96	95	97	86	48	95	134	75	74	...	129	92	70	112	73	98	134	145	104	144
5	CO	0	0	1	0	0	0	0	0	3	...	3	0	0	0	0	0	0	4	3	8
6	CT	3773	3597	4631	1403	1348	1810	1748	3058	3896	...	3068	3039	2657	2925	2360	2541	1748	2051	1859	1233
7	DE	167	152	194	212	339	646	506	715	772	...	656	873	669	509	417	435	506	608	520	641
8	DC	11	17	25	14	16	10	103	116	74	...	42	0	0	35	40	121	103	84	79	100
9	FL	54	43	79	43	46	47	216	30	88	...	84	115								