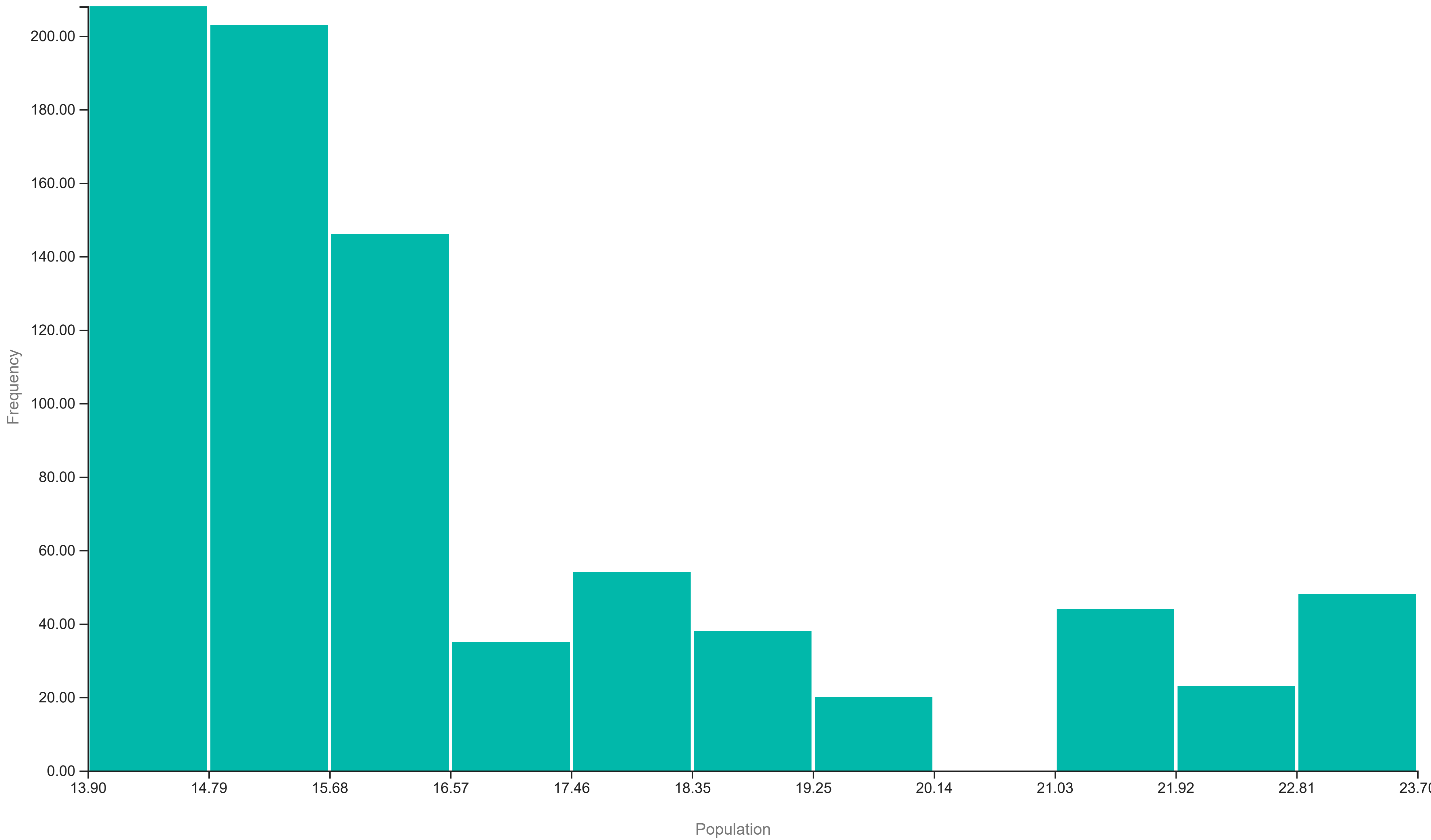
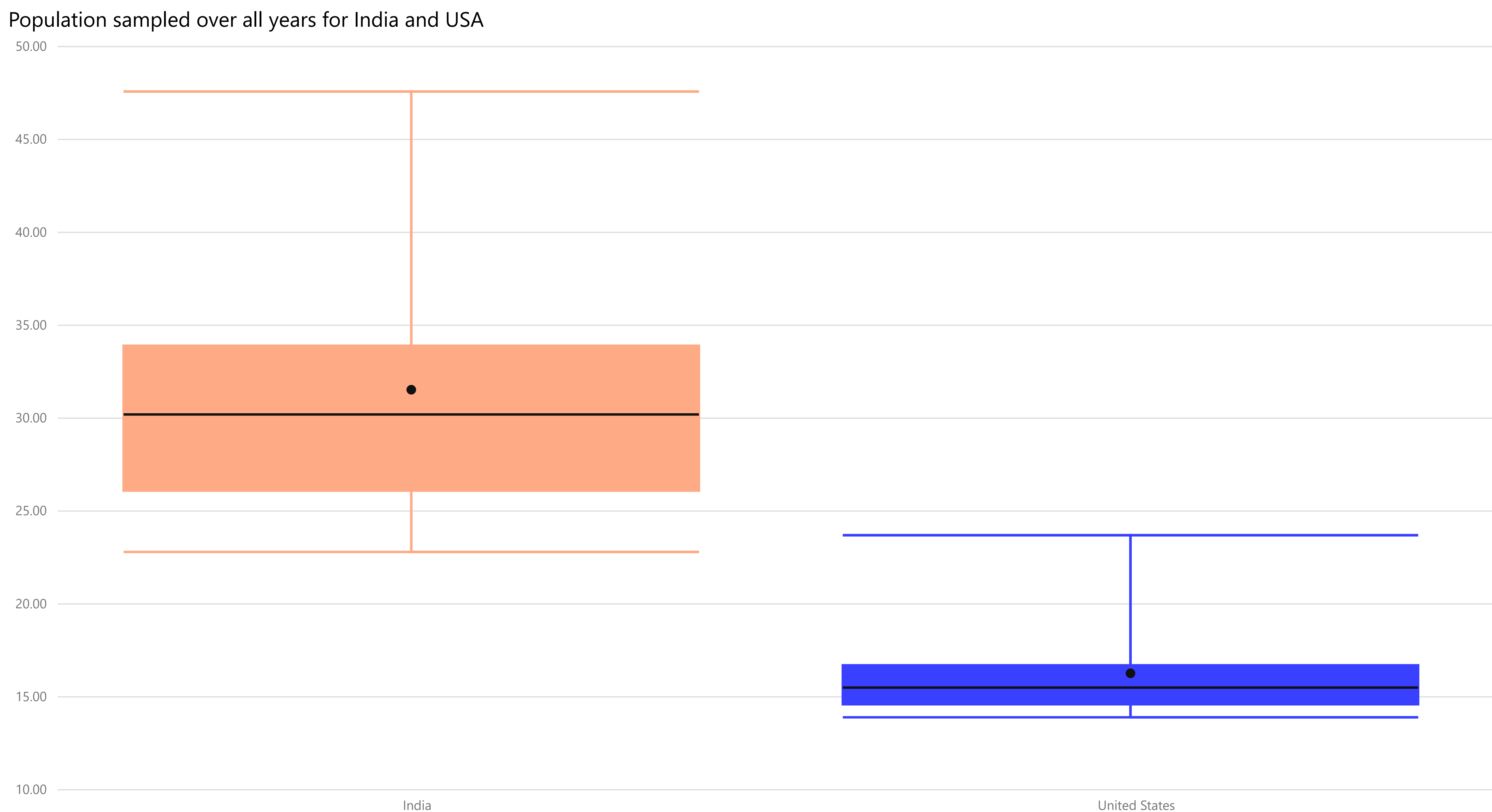
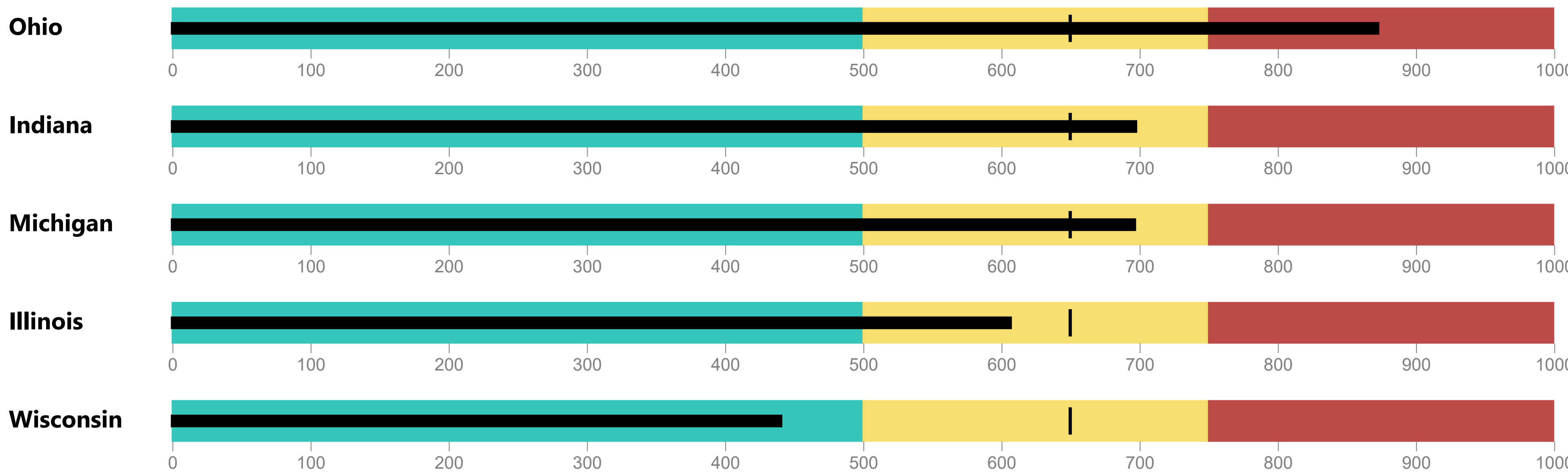
USA Population Rate Changes



Ceiling value of population has been used to reduce the number of groups

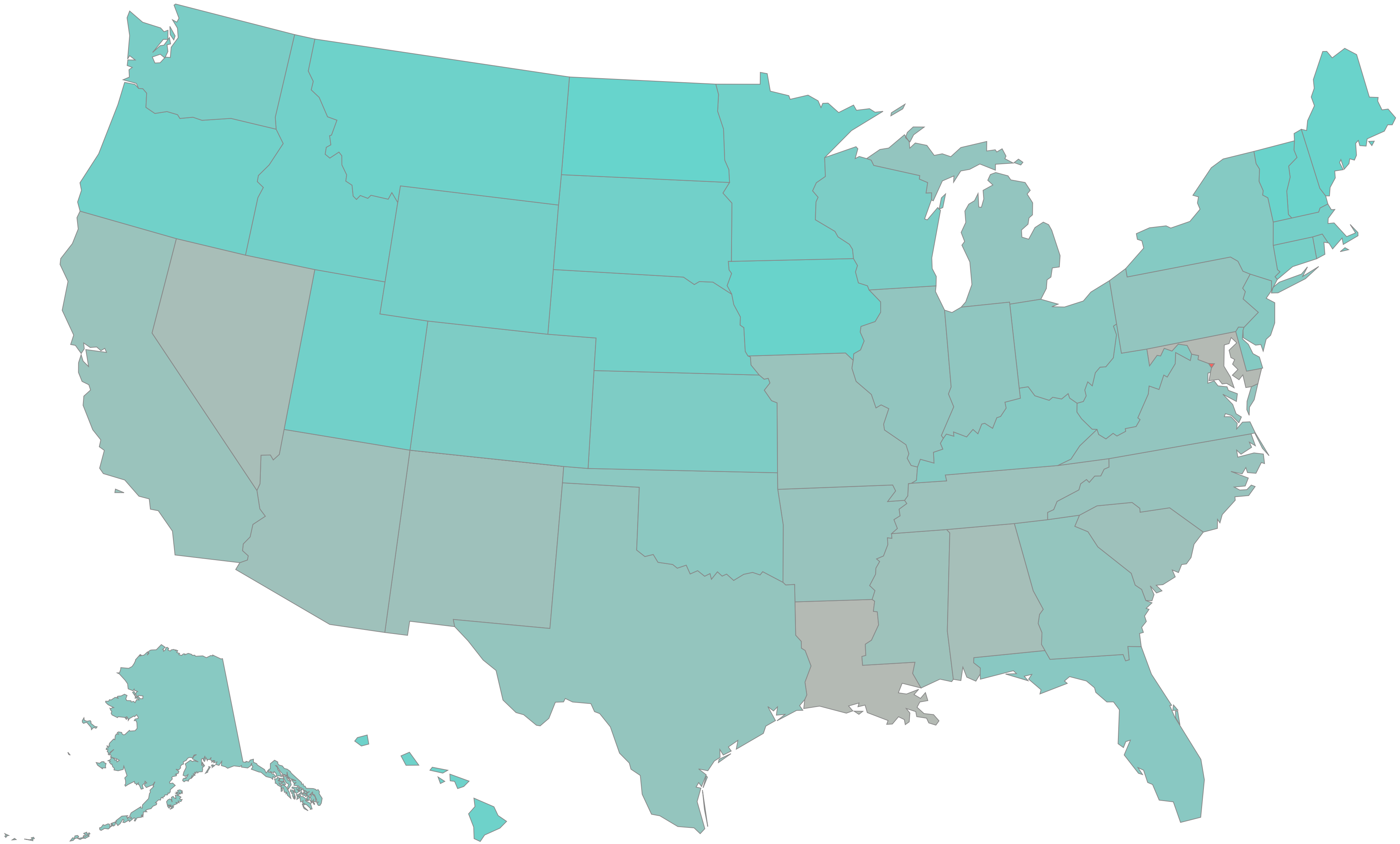

Population sampled over all years for India and USA
50.00
45.00
40.00
35.00
30.00
25.00
20.00
15.00
10.00
India
United States

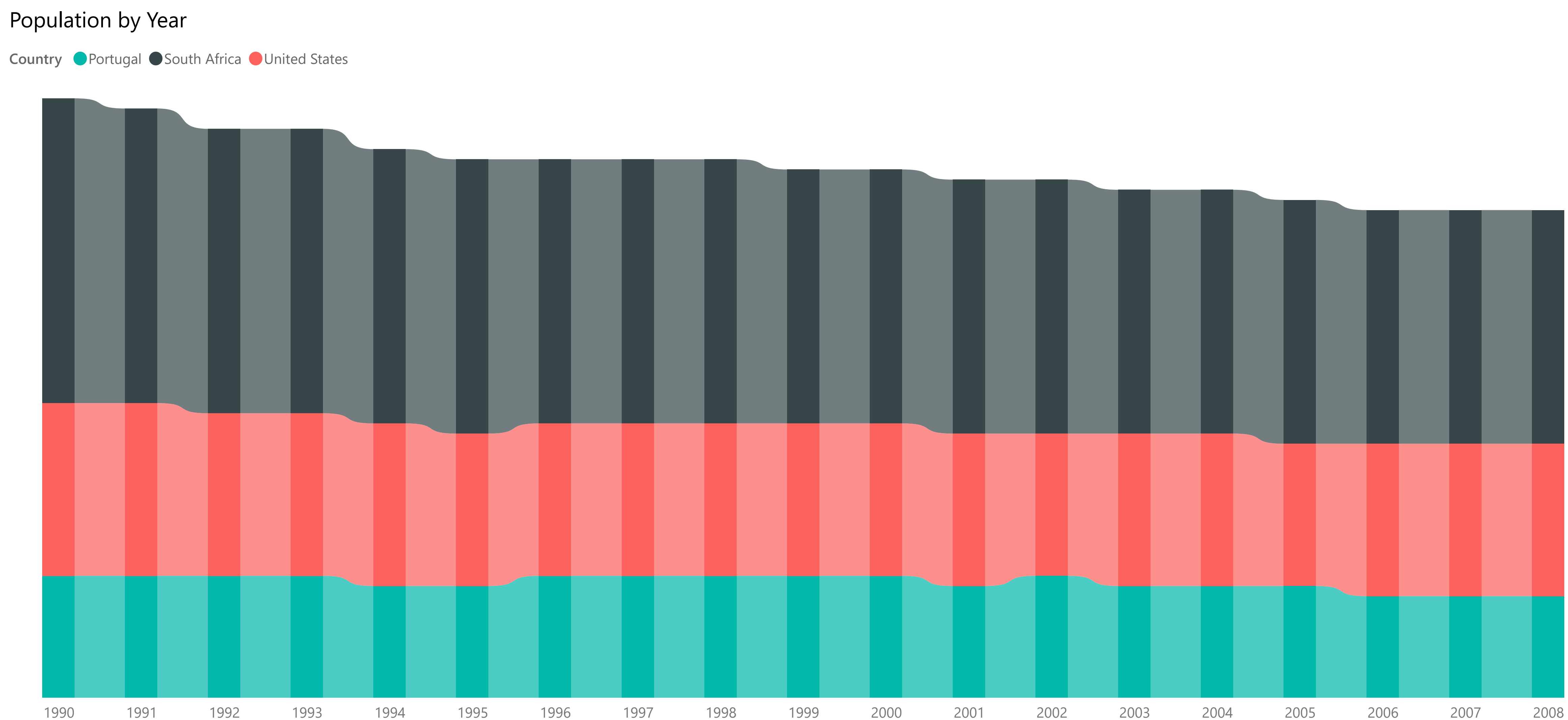
Theft Crime comparison in East North Central States of USA



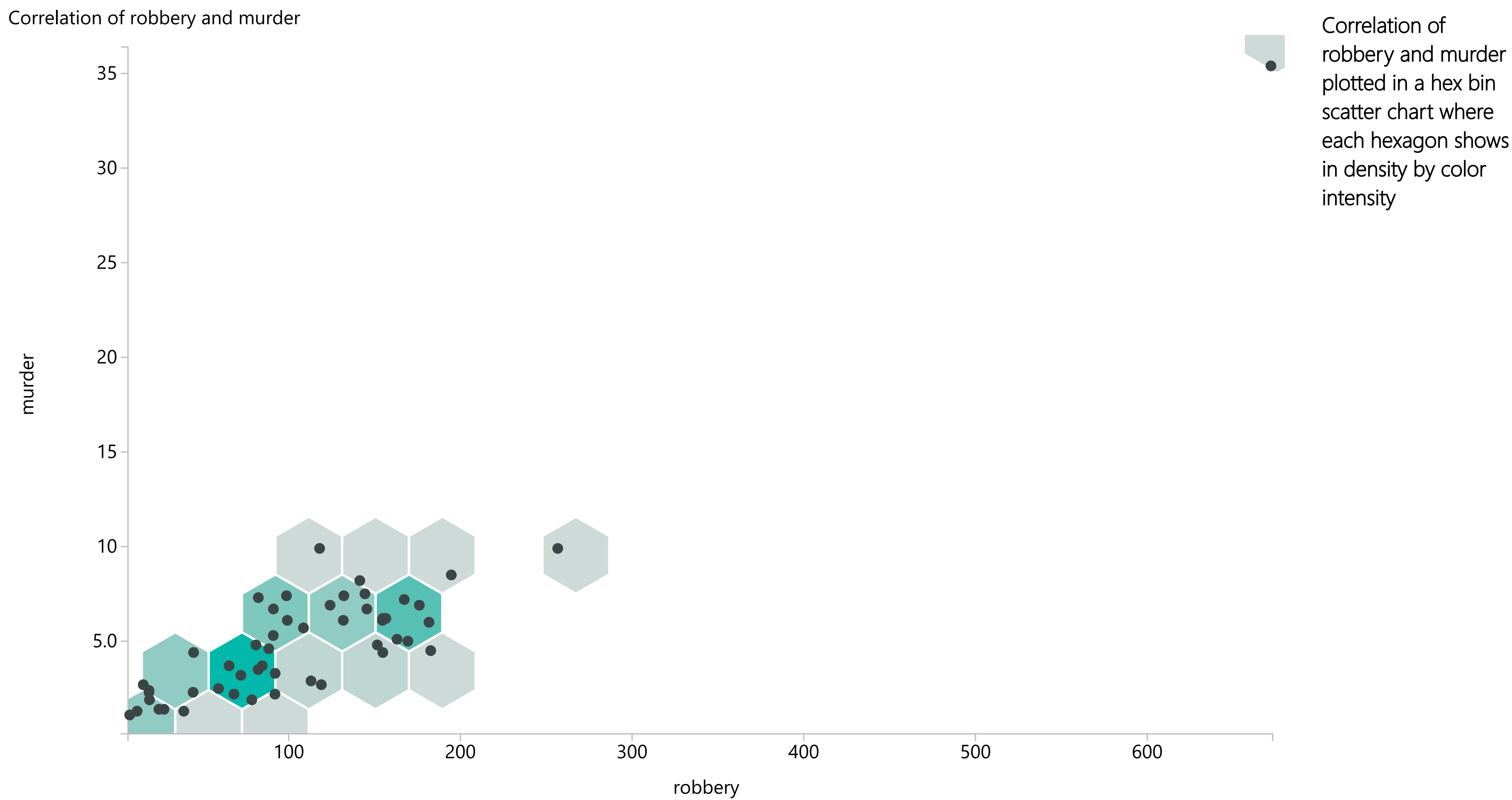
Ohio, Indiana, Michigan and Illinois are the four East North Central states of USA, chosen for the bullet chart, to keep the visualization limited. The green bar shows number of burglary incident marked as 'Safe', yellow represents 'Moderate' and red represents 'Unsafe'. The tick mark is the targeted theft crime index and the black bar is the actual. The further away the bar is from the tick, towards the green part, the safer.

Count of murder crimes by State





Population and population changes over year - Comparison between Portugal, South Africa and USA for the period of 1990 to 2008


Correlation of robbery and murder
murder
35
30
25
20
15
10
5.0
100
200
300
400
500
600
robbery
Correlation of robbery and murder plotted in a hex bin scatter chart where each hexagon shows in density by color intensity

Import modules

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import math
from matplotlib.ticker import FuncFormatter
import plotly
import plotly.figure_factory as ff
from pandas.plotting import parallel_coordinates
import numpy as np

%matplotlib inline
```

Data load and transformation

In [2]:
```
education = pd.read_csv('ex6-2/education.csv')
crime = pd.read_csv('ex6-2/crimeratesbystate-formatted.csv')
birthrate = pd.read_csv('ex6-2/birth-rate.csv')

# remove whitespaces from crime dataset (sine we have already encountered it)
education = education.applymap(lambda x: x.strip() if type(x) is str else x)
crime = crime.applymap(lambda x: x.strip() if type(x) is str else x)
birthrate = birthrate.applymap(lambda x: x.strip() if type(x) is str else x)
```

Histogram

Distribution of birth rate

In [3]:
```
birthrate_hist = pd.melt(birthrate, id_vars="Country", var_name="Year", value_name = 'Birt
birthrate_hist["BirthRate_int"] = birthrate_hist["BirthRate"].apply(lambda x: math.ceil(x)
birthrate_hist.head()
```

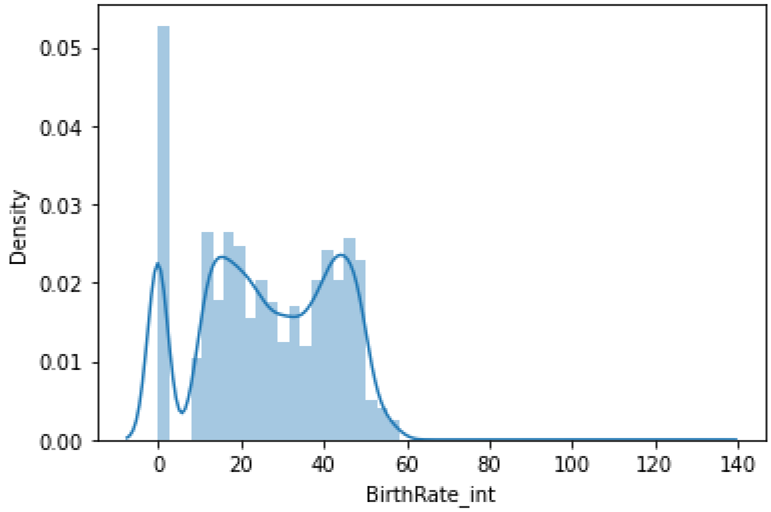
Out[3]:

	Country	Year	BirthRate	BirthRate_int
0	Aruba	1960	36.400	37
1	Afghanistan	1960	52.201	53
2	Angola	1960	54.432	55
3	Albania	1960	40.886	41
4	Netherlands Antilles	1960	32.321	33

In [4]:
```
sns.distplot( birthrate_hist["BirthRate_int"] )
```

```
/Users/veerareddykoppula/opt/anaconda3/lib/python3.9/site-packages/seaborn/distributions.p
y: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 simi
lar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[4]:
```
<AxesSubplot:xlabel='BirthRate_int', ylabel='Density'>
```

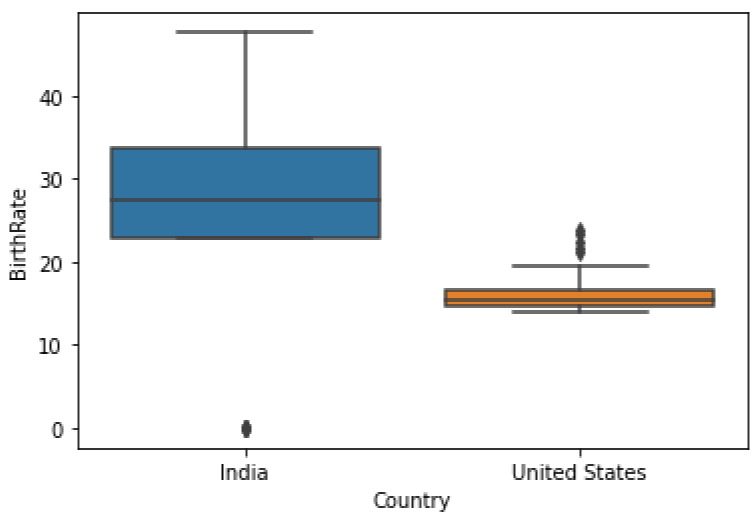


Box plot

Comparison of birthrate betwen India and USA

In [5]:
```
birthrate_box = birthrate_hist[(birthrate_hist["Country"]=="United States") | (birthrate_h
sns.boxplot(x = birthrate_box["Country"], y=birthrate_box["BirthRate"])
```

Out[5]: <AxesSubplot:xlabel='Country', ylabel='BirthRate'>



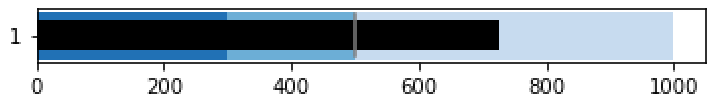
Bullet chart

US burglary statistics against some dummy benchmark

In [6]:
```
# transform data
crime_bullet = crime[crime["state"]=="United States"][["state","burglary"]]
crime_bullet['target'] = 500
crime_bullet_tuple = [tuple(x) for x in crime_bullet.values][0]


# set parameter for bullet chart
limits = [300, 500, 1000]
palette = sns.color_palette("Blues_r", len(limits))
fig, ax = plt.subplots()
ax.set_aspect('equal')
ax.set_yticks([1])
ax.set_yticklabels='United States'

prev_limit = 0
```

```python
for idx, lim in enumerate(limits):
    ax.barh([1], lim-prev_limit, left=prev_limit, height=75, color=palette[idx])
    prev_limit = lim

# draw the value we're measuring
ax.barh([1], crime_bullet_tuple[1], color='black', height=45)

ax.axvline(crime_bullet_tuple[2], color="gray", ymin=0.10, ymax=0.9)
```

Out[6]: <matplotlib.lines.Line2D at 0x7fdf0d76b640>

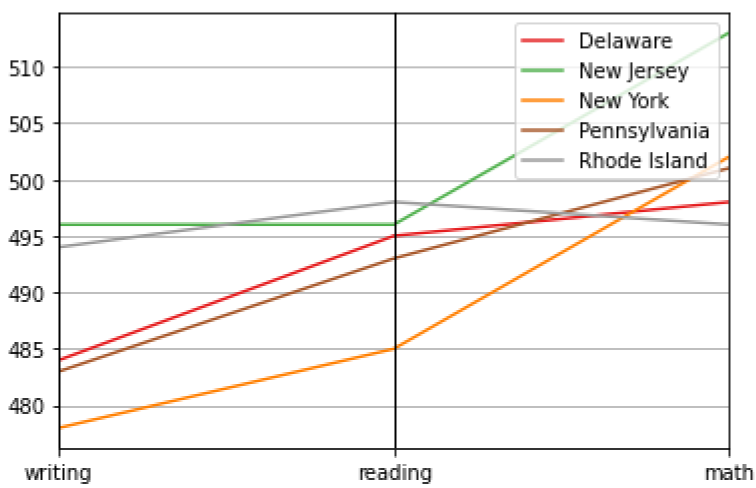


Parallel Coordinate plot

Comparison of reading, writing and math numbers between 5 states

In [7]:

```python
# transform data
education_parallel = education[education['state'].isin(['New York','New Jersey','Delaware'

# make the plot
parallel_coordinates(education_parallel, 'state', colormap=plt.get_cmap("Set1"))
plt.show()
```

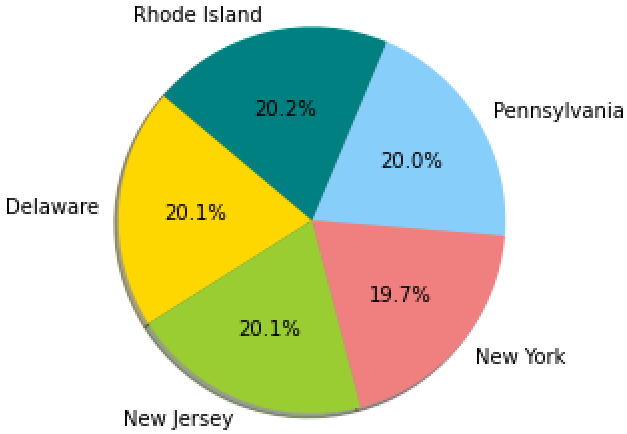


Pie chart

Comparison of reading numbers between 5 states

In [8]:

```python
# transform data
education_pie = education_parallel[['state','reading']]

# set colors
colors = ['gold', 'yellowgreen', 'lightcoral', 'lightskyblue','teal']

# plot
plt.pie(education_pie['reading'], labels=education_pie['state'], colors=colors,
autopct='%1.1f%%', shadow=True, startangle=140)

plt.axis('equal')
plt.show()
```

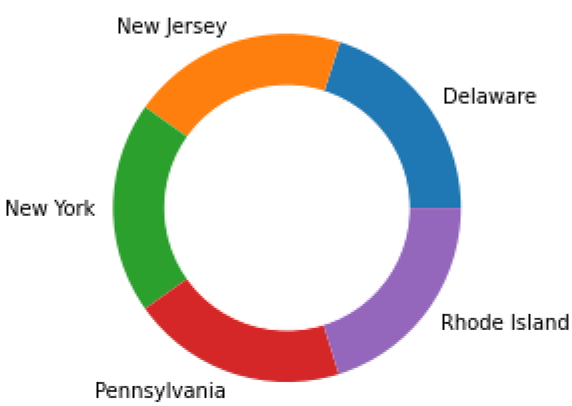


Donought chart

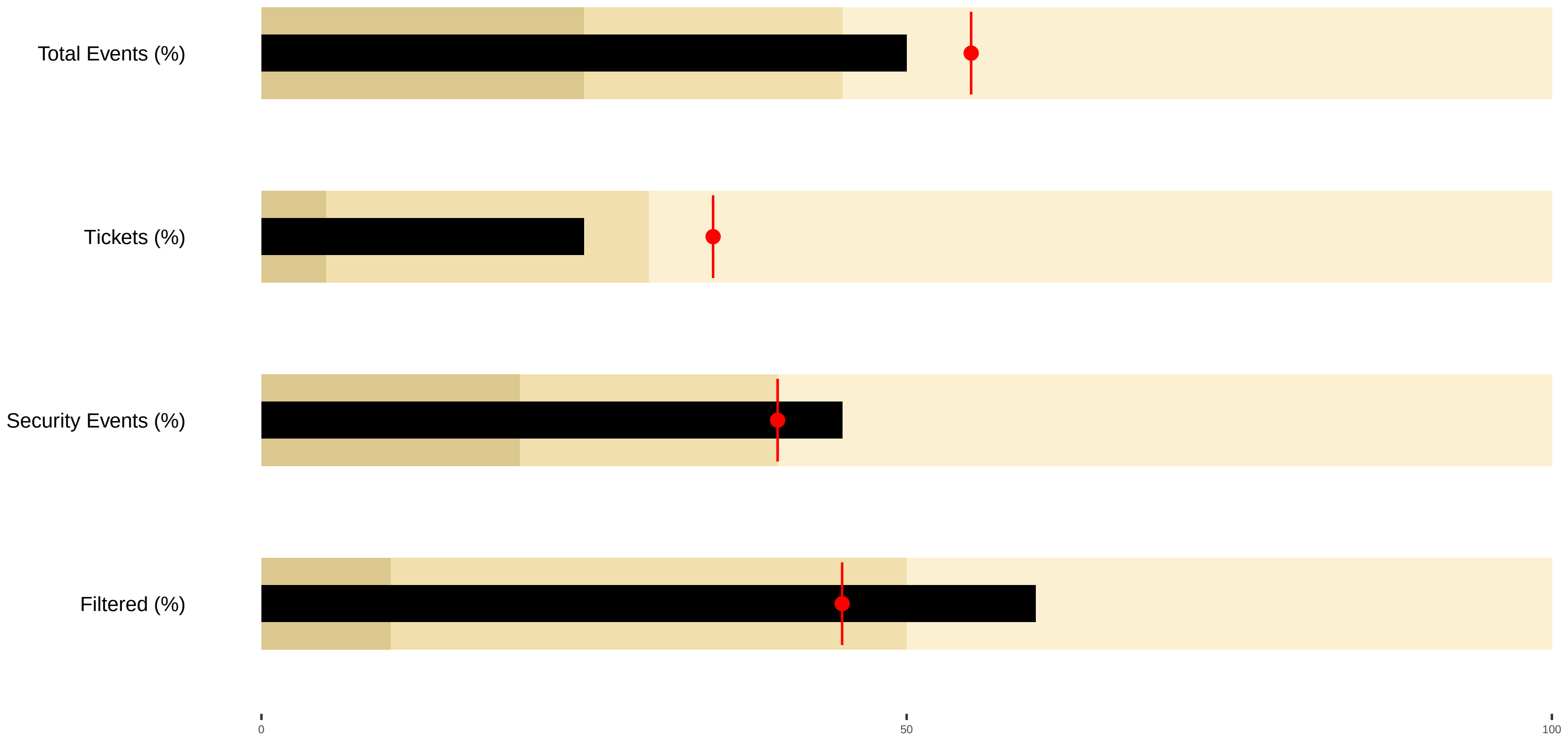
Comparison of reading, writing and math numbers between 5 states

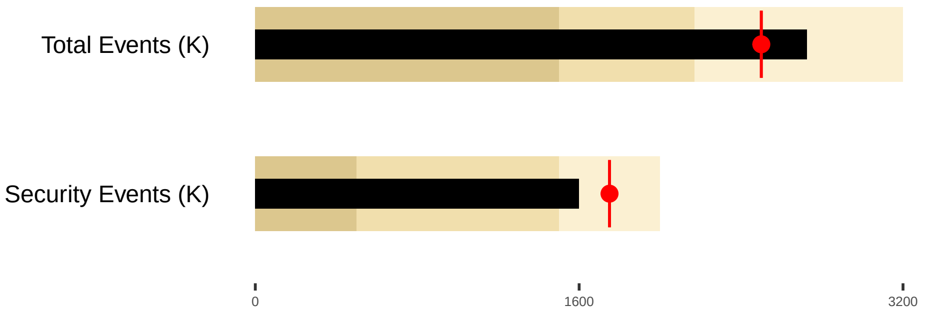
In [9]:

```
# transform data
education_donut = education_pie

# create a pieplot
plt.pie(education_donut['reading'], labels=education_donut['state'])

# add a circle at the center
my_circle=plt.Circle( (0,0), 0.7, color='white')
p=plt.gcf()
p.gca().add_artist(my_circle)

plt.show()
```



In [ ]:


Total Events (%)
Tickets (%)
Security Events (%)
Filtered (%)
0
50
100


Total Events (K)
Security Events (K)
0
1600
3200

Assignment 6.2

Veera Koppula

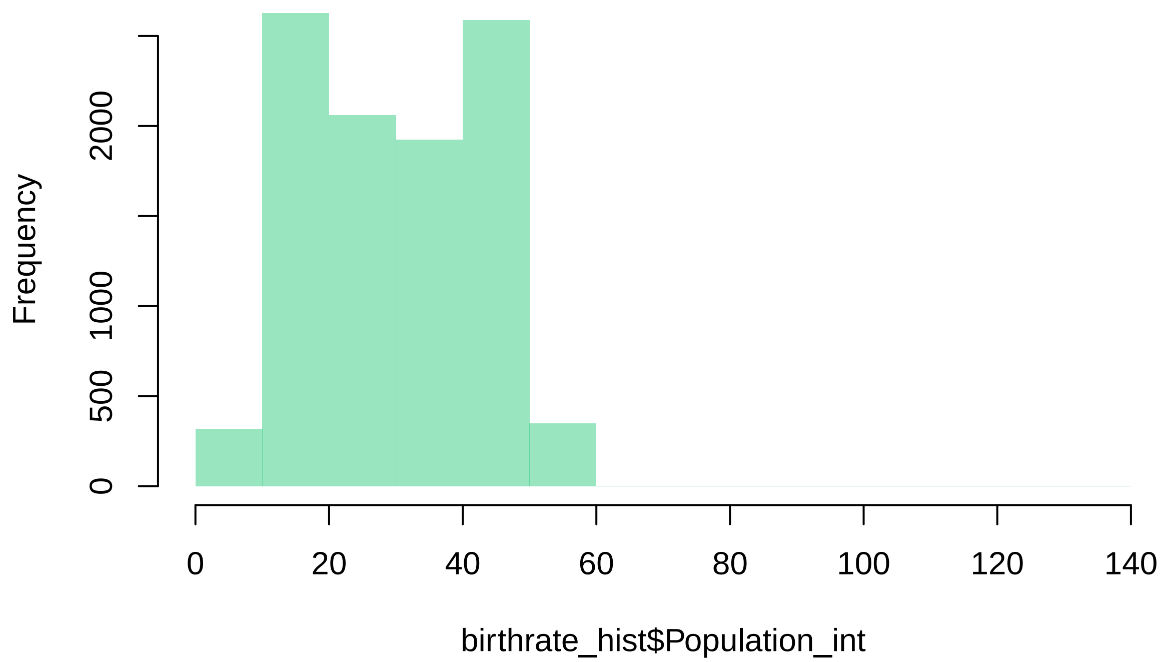
05/23/2022

These two weeks we are going to be focused on histograms, box plots, and bullet charts and using various tools to create these visualizations. You must consolidate all the charts into ONE document with each chart labeled with the type of chart and technology - for example: Python - Bar Chart. Failure to label and consolidate the charts will resort in points being taken off or a 0 for the assignment.

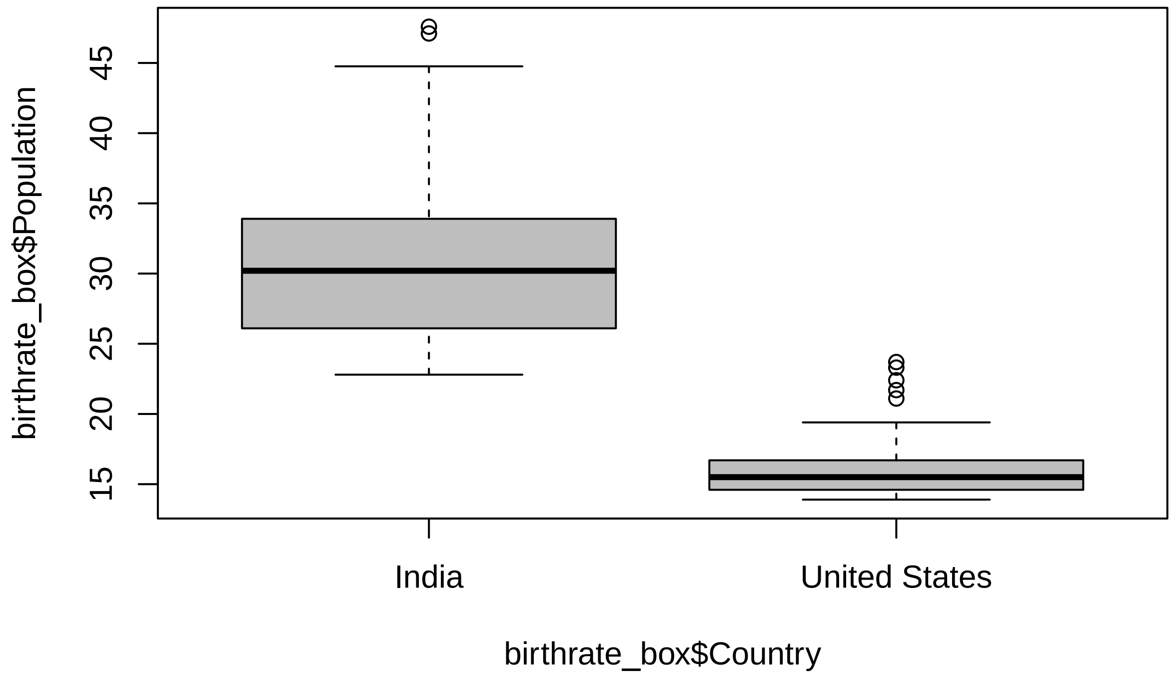
```
##  [1] "Country" "X1960"   "X1961"   "X1962"   "X1963"   "X1964"   "X1965"
##  [8] "X1966"   "X1967"   "X1968"   "X1969"   "X1970"   "X1971"   "X1972"
## [15] "X1973"   "X1974"   "X1975"   "X1976"   "X1977"   "X1978"   "X1979"
## [22] "X1980"   "X1981"   "X1982"   "X1983"   "X1984"   "X1985"   "X1986"
## [29] "X1987"   "X1988"   "X1989"   "X1990"   "X1991"   "X1992"   "X1993"
## [36] "X1994"   "X1995"   "X1996"   "X1997"   "X1998"   "X1999"   "X2000"
## [43] "X2001"   "X2002"   "X2003"   "X2004"   "X2005"   "X2006"   "X2007"
## [50] "X2008"
```

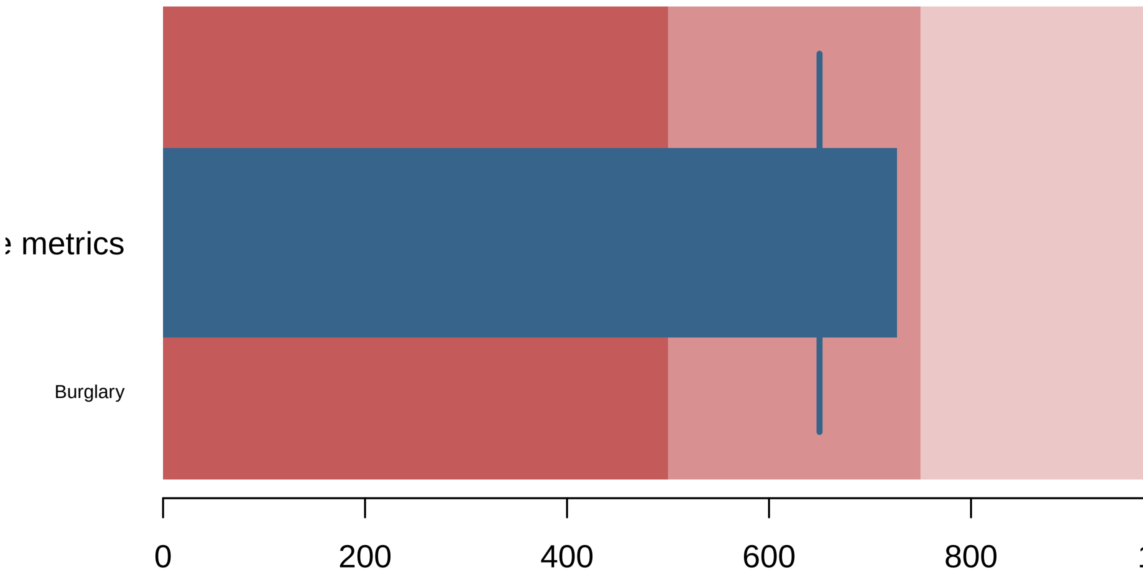
```
##  [1] "Country" "1960"    "1961"    "1962"    "1963"    "1964"    "1965"
##  [8] "1966"    "1967"    "1968"    "1969"    "1970"    "1971"    "1972"
## [15] "1973"    "1974"    "1975"    "1976"    "1977"    "1978"    "1979"
## [22] "1980"    "1981"    "1982"    "1983"    "1984"    "1985"    "1986"
## [29] "1987"    "1988"    "1989"    "1990"    "1991"    "1992"    "1993"
## [36] "1994"    "1995"    "1996"    "1997"    "1998"    "1999"    "2000"
## [43] "2001"    "2002"    "2003"    "2004"    "2005"    "2006"    "2007"
## [50] "2008"
```

Plot1: Histogram

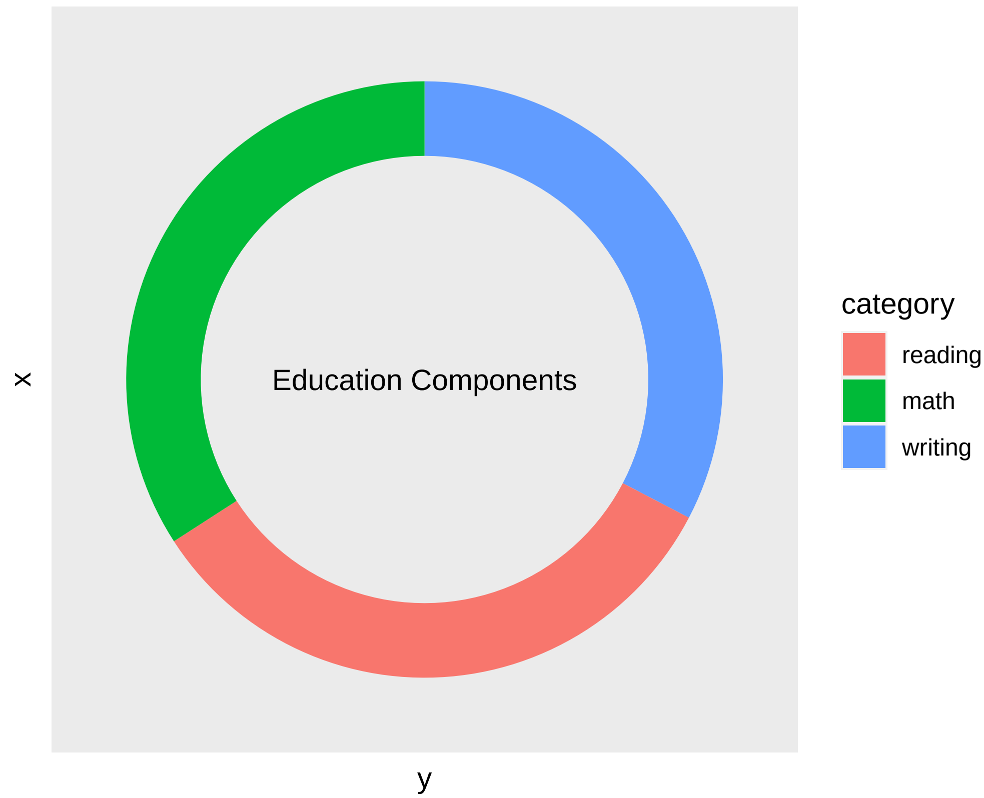


Plot2: Box Plot





Plot3: Bullet Chart



Plot4: Donut Chart

Plot5: Pie Chart