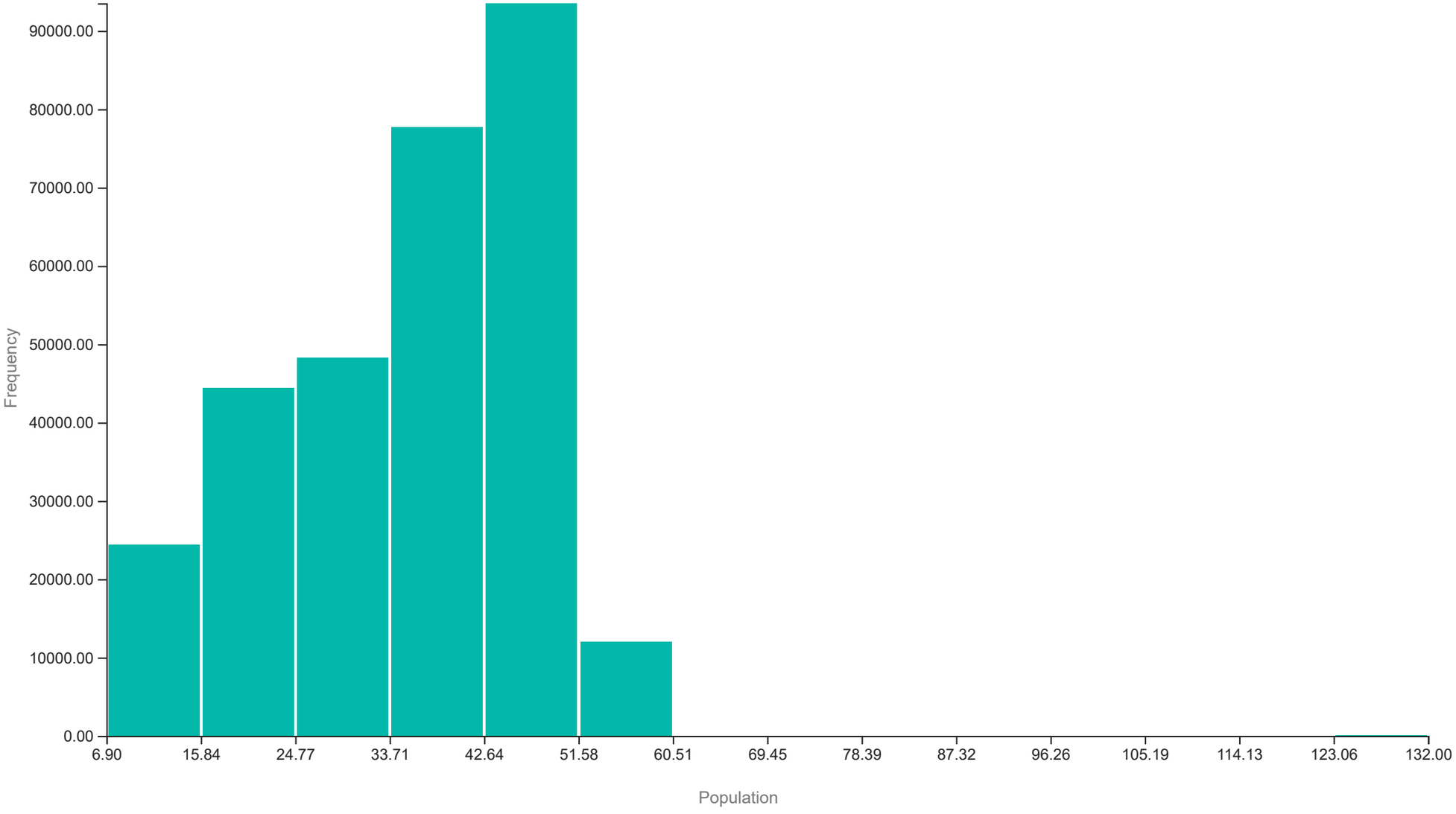
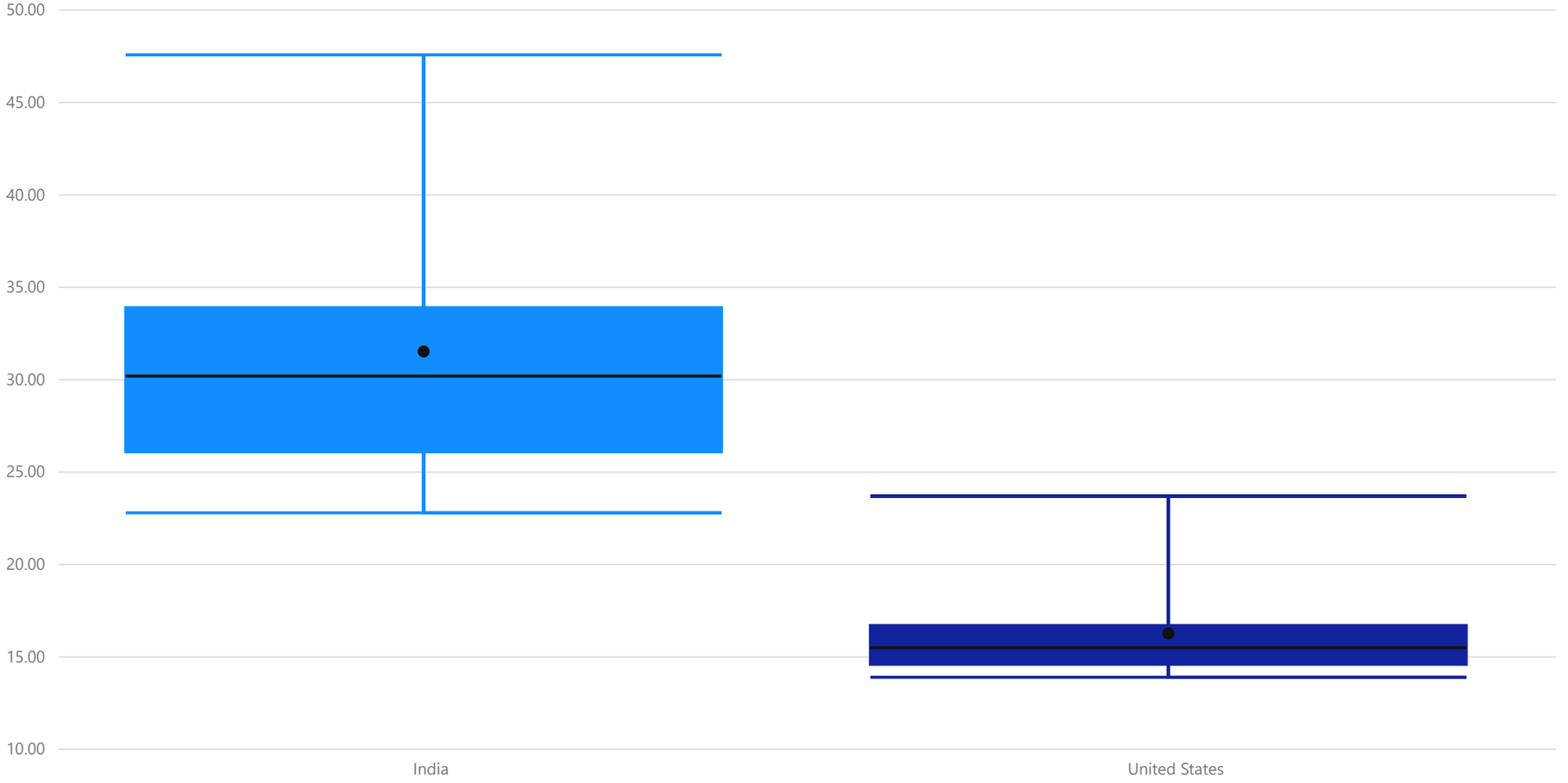
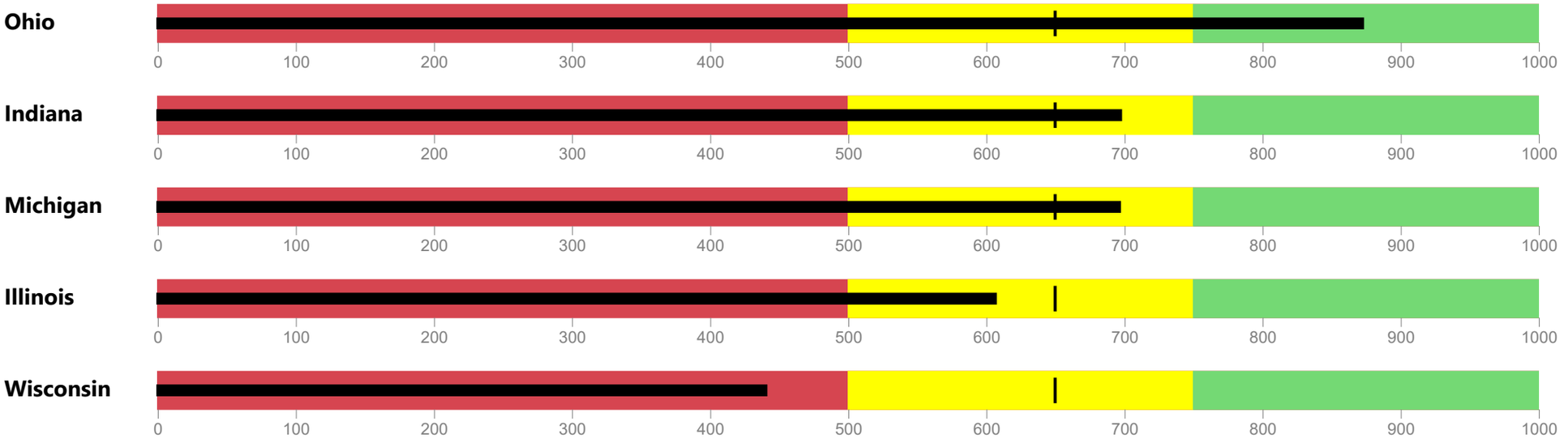
PowerBI_Histogram



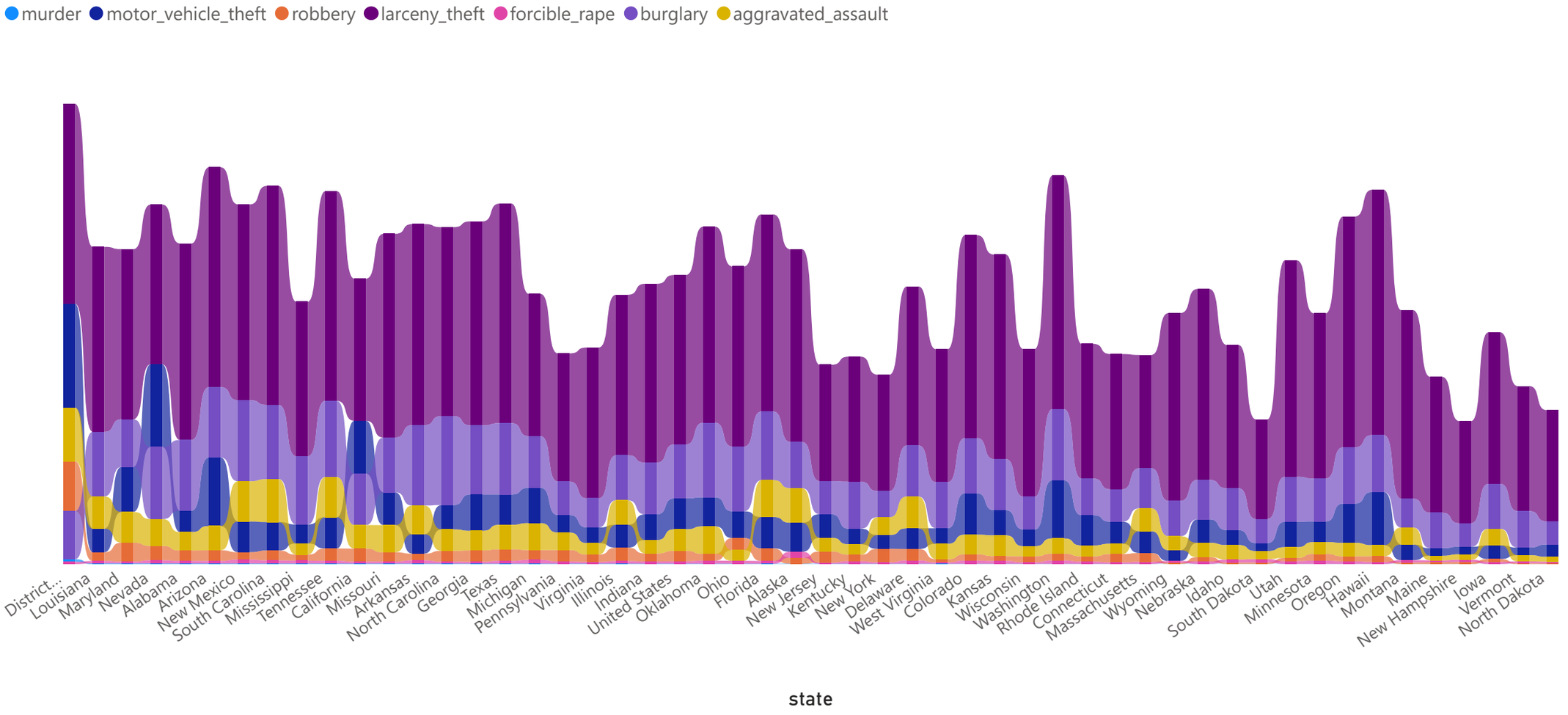
PowerBI-Box Plot



PowerBI-Bullet chart



PowerBI-Ribbon chart



Python Plots

In [1]:
```
# Imports
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
```

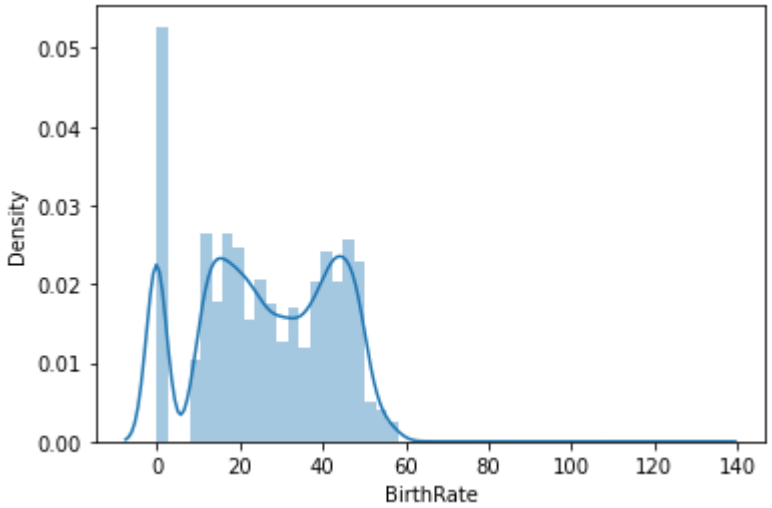
In [17]:
```
education = pd.read_csv('ex6-2/education.csv')
crime = pd.read_csv('ex6-2/crimeratesbystate-formatted.csv')
birthrate = pd.read_csv('ex6-2/birth-rate.csv')

# removing whitespaces
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)
```

Python-Histogram

In [9]:
```
birthrate_hist = pd.melt(birthrate, id_vars="Country", var_name="Year", value_name = 'BirthRate').fillna(0)
birthrate_hist["BirthRate"] = birthrate_hist["BirthRate"].apply(lambda x: math.ceil(x))
sns.distplot( birthrate_hist["BirthRate"] )
```

```
C:\Users\meena\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a de
precated function and will be removed in a future version. Please adapt your code to use either `displot`
(a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

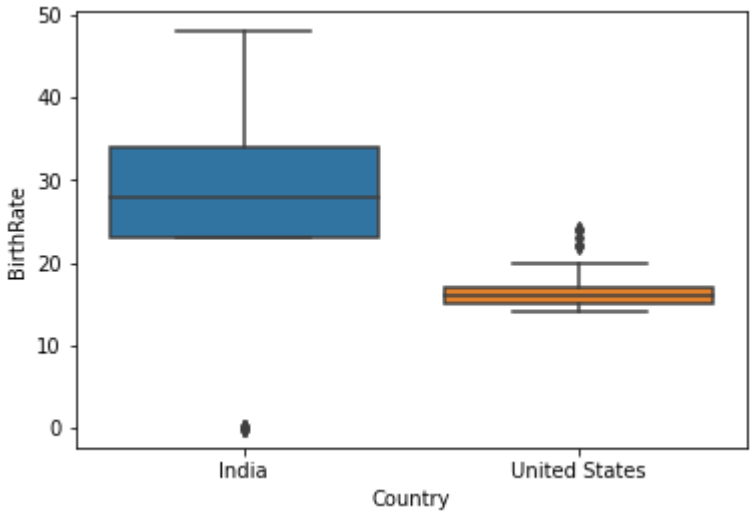
Out[9]: <AxesSubplot:xlabel='BirthRate', ylabel='Density'>



Python-Box plot

In [14]:
```
birthrate_box = birthrate_hist[(birthrate_hist["Country"]=="India") | (birthrate_hist["Country"]=="United St
sns.boxplot(x = birthrate_box["Country"], y=birthrate_box["BirthRate"])
```

Out[14]: <AxesSubplot:xlabel='Country', ylabel='BirthRate'>



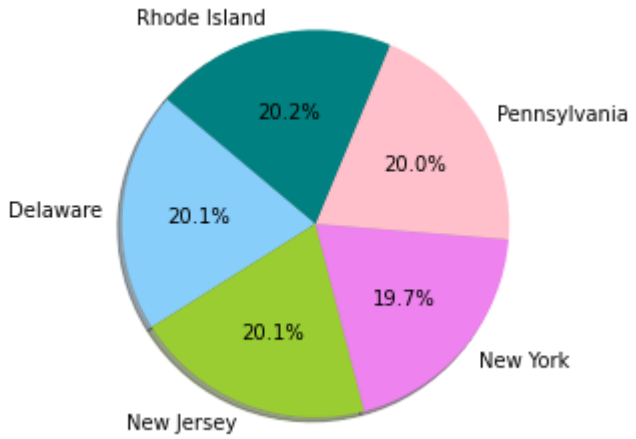
Python-Bullet chart

In [40]:
```
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]
crime_bullet_tuple

limits = [300, 500, 1000]
palette = sns.color_palette("Blues_r", len(limits))
fig, ax = plt.subplots()
ax.set_aspect('equal')
ax.set_yticks([1])

prev_limit = 0
for idx, lim in enumerate(limits):
    ax.barh([1], lim-prev_limit, left=prev_limit, height=75, color=palette[idx])
    prev_limit = lim

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[40]: <matplotlib.lines.Line2D at 0x22396011580>

Python-Pie chart

In [26]:
```
# Pie chart: Comparison of reading numbers between 5 states
education_parallel = education[education['state'].isin(['New York','New Jersey','Delaware','Rhode Island','P
education_pie = education_parallel[['state','reading']]

# set colors
colors = ['lightskyblue', 'yellowgreen', 'violet', 'pink','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()
```



R plots

In [1]:
```
# install.packages("reshape2")
```

In [2]:
```
library('magrittr')
library("reshape2")
source("BulletGraph.R", local=TRUE)
```

In [3]:
```
birthrate <- read.csv('ex6-2/birth-rate.csv')
crime <- read.csv('ex6-2/crimeratesbystate-formatted.csv')
education <- read.csv('ex6-2/education.csv')
```

In [4]:
```
colnames(birthrate)
```

'Country' · 'X1960' · 'X1961' · 'X1962' · 'X1963' · 'X1964' · 'X1965' · 'X1966' · 'X1967' · 'X1968' · 'X1969' · 'X1970' · 'X1971' · 'X1972' · 'X1973' · 'X1974' · 'X1975' · 'X1976' · 'X1977' · 'X1978' · 'X1979' · 'X1980' · 'X1981' · 'X1982' · 'X1983' · 'X1984' · 'X1985' · 'X1986' · 'X1987' · 'X1988' · 'X1989' · 'X1990' · 'X1991' · 'X1992' · 'X1993' · 'X1994' · 'X1995' · 'X1996' · 'X1997' · 'X1998' · 'X1999' · 'X2000' · 'X2001' · 'X2002' · 'X2003' · 'X2004' · 'X2005' · 'X2006' · 'X2007' · 'X2008'

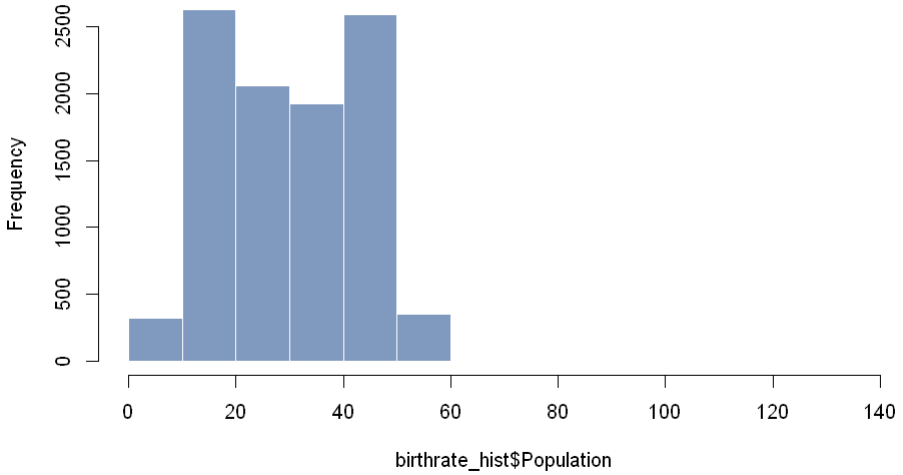
In [5]:
```
colnames(birthrate) <- gsub("X", "", colnames(birthrate))

# check column names
colnames(birthrate)
```

'Country' · '1960' · '1961' · '1962' · '1963' · '1964' · '1965' · '1966' · '1967' · '1968' · '1969' · '1970' · '1971' · '1972' · '1973' · '1974' · '1975' · '1976' · '1977' · '1978' · '1979' · '1980' · '1981' · '1982' · '1983' · '1984' · '1985' · '1986' · '1987' · '1988' · '1989' · '1990' · '1991' · '1992' · '1993' · '1994' · '1995' · '1996' · '1997' · '1998' · '1999' · '2000' · '2001' · '2002' · '2003' · '2004' · '2005' · '2006' · '2007' · '2008'

R-Histogram

In [6]:
```
options(repr.plot.width = 8, repr.plot.height = 5)
```

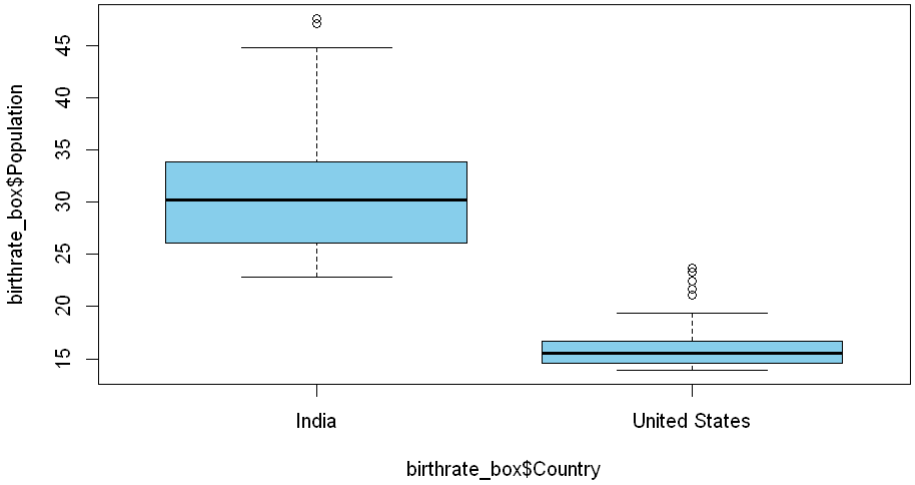
```
birthrate_hist <- reshape2::melt(birthrate, id=c("Country")) %>%
  dplyr::mutate("Country" = as.character(Country),
                "Year" = as.character(variable),
                "Population" = value,
                "Population_int" = ceiling(value)) %>%
  dplyr::select(c("Country","Year","Population", "Population_int"))

hist(birthrate_hist$Population, col=rgb(0,0.2,0.5,0.5) , border=F , main="")
```



R-Box plot

In [7]:

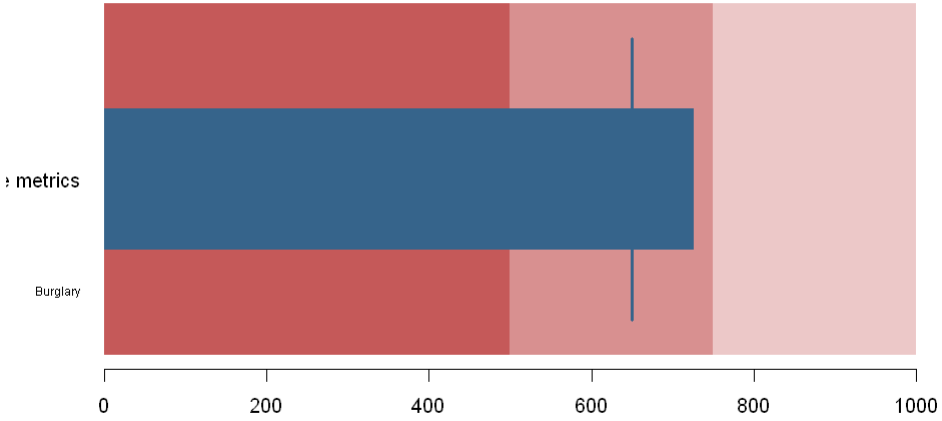
```
birthrate_box <- birthrate_hist %>%
  dplyr::filter(Country %in% c("United States", "India"))

boxplot(birthrate_box$Population ~ birthrate_box$Country , col="skyblue")
```



R-Bullet graph

In [8]:

```
crime_bullet <- crime %>%
  dplyr::filter(stringr::str_trim(state, "both") == "United States") %>%
  dplyr::select(c(state, burglary))
bulletgraph(x=crime_bullet$burglary,ref=650,limits=c(0,500,750,1000),
            name= "USA Crime metrics",subname="Burglary",
            col="steelblue4",shades="firebrick")
```

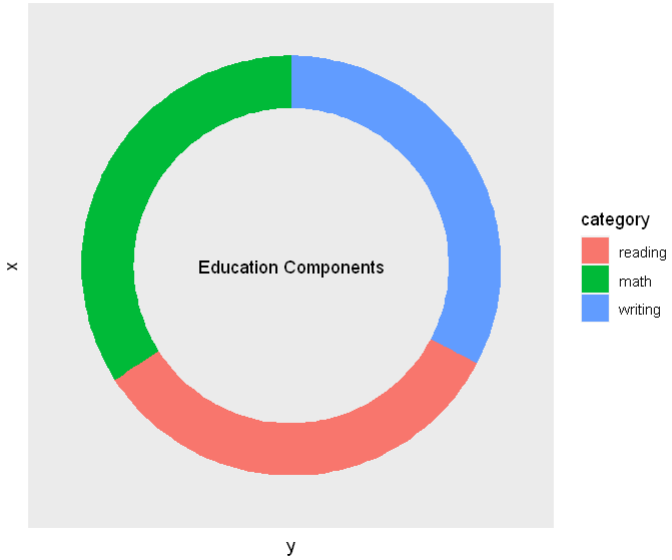


R-Donut chart

In [9]:

```
education_donut <- education %>%
  dplyr::filter(stringr::str_trim(state, "both") == "United States") %>%
  reshape2::melt(id=c("state")) %>%
  dplyr::rename("category" = variable) %>%
  dplyr::filter(category %in% c("reading","math","writing")) %>%
  dplyr::select(-state)

# add addition columns, needed for drawing with geom_rect
education_donut$fraction = education_donut$value / sum(education_donut$value)
education_donut = education_donut[order(education_donut$fraction), ]
education_donut$ymax = cumsum(education_donut$fraction)
education_donut$ymin = c(0, head(education_donut$ymax, n=-1))

# make the plot
ggplot2::ggplot(education_donut, ggplot2::aes(fill=category, ymax=ymax, ymin=ymin, xmax=4, xmin=3)) +
  ggplot2::geom_rect() +
  ggplot2::coord_polar(theta="y") +
  ggplot2::xlim(c(0, 4)) +
  ggplot2::theme(panel.grid=ggplot2::element_blank()) +
  ggplot2::theme(axis.text=ggplot2::element_blank()) +
  ggplot2::theme(axis.ticks=ggplot2::element_blank()) +
```

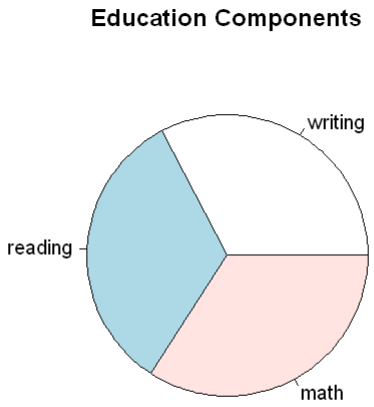
```
ggplot2::annotate("text", x = 0, y = 0, label = "Education Components") +
ggplot2::labs(title="")
```



R-Pie chart

In [10]:

```
# pie chart
slices <- education_donut$value
lbls <- education_donut$category
pie(slices, labels = lbls, main="Education Components")
```


Education Components
writing
reading
math