Ryan Long
DSC640
1.2 Exercises: Charts

Datasets used:
hotdog-contest-winners.xlsm
obama-approval-ratings.xls

Summary

I hadn't used Power BI extensively prior to this exercise. I found it to be intuitive and required little effort to create the graphs as it felt familiar to creating graphs in Excel and Word. Creating the graphs in Python was similarly easy as I have been using it primarily throughout this course work. I found R to be the most challenging as I have not used it much beyond the initial classes of this program which included a temporary hiatus of my course work.

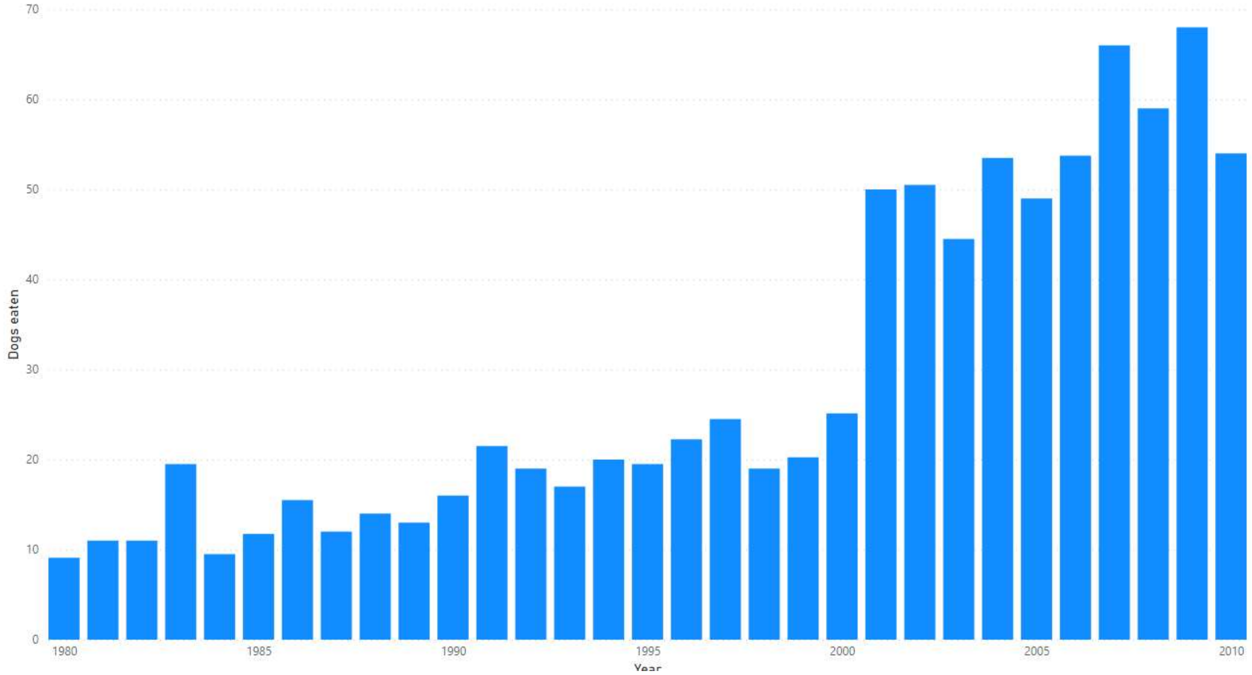
As someone who will most likely use these tools in a casual or supplementary sense in the workplace, I gravitate towards Power BI and Python as future solutions. I believe those entrenched in data science daily either in a professional setting or academics would lean more towards R. Both Python and R offer near unlimited flexibility as compared to the out of the box nature of Power BI.

The following pages contain:

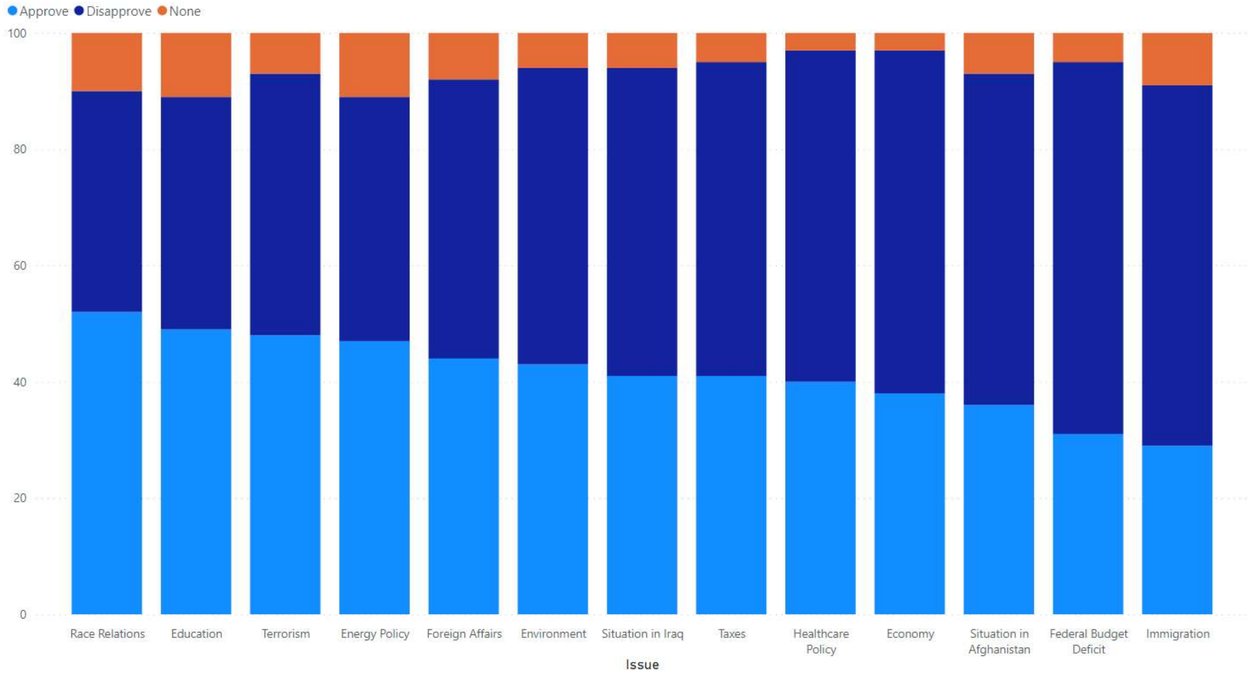
Appendix

Code support for both Python and R notebooks

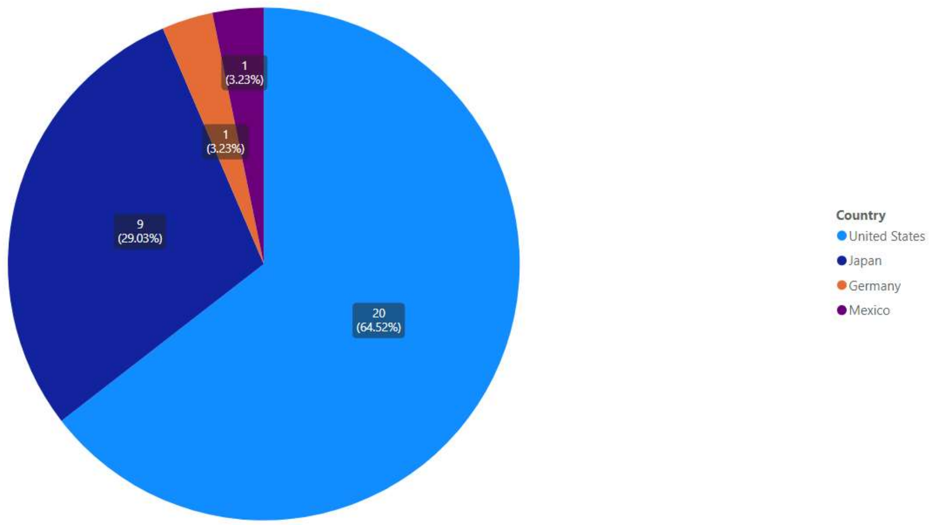
Power BI: Bar chart - Number of Hot Dogs Eaten by Contest Winners



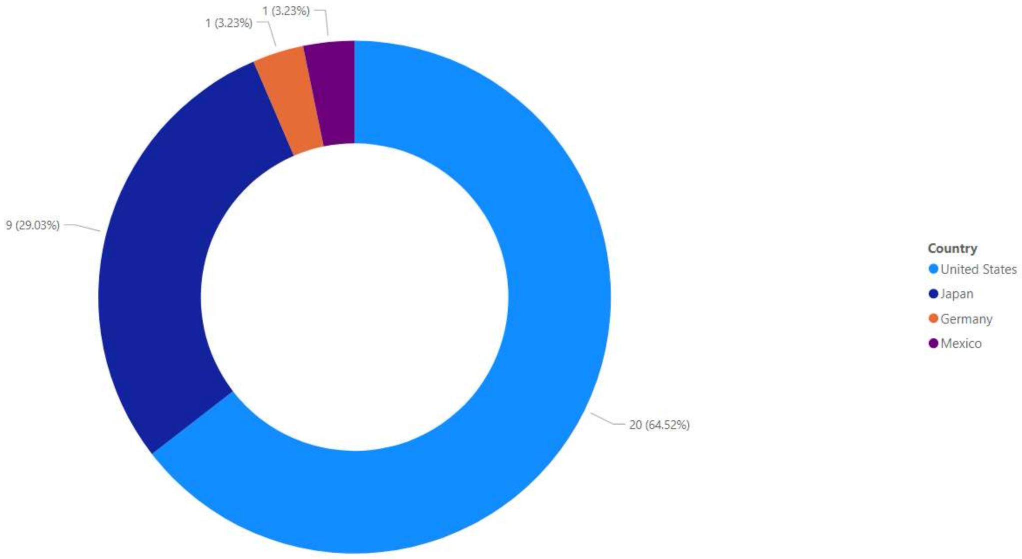
Power BI: Stacked Bar chart - President Obama Approval Ratings by Issue

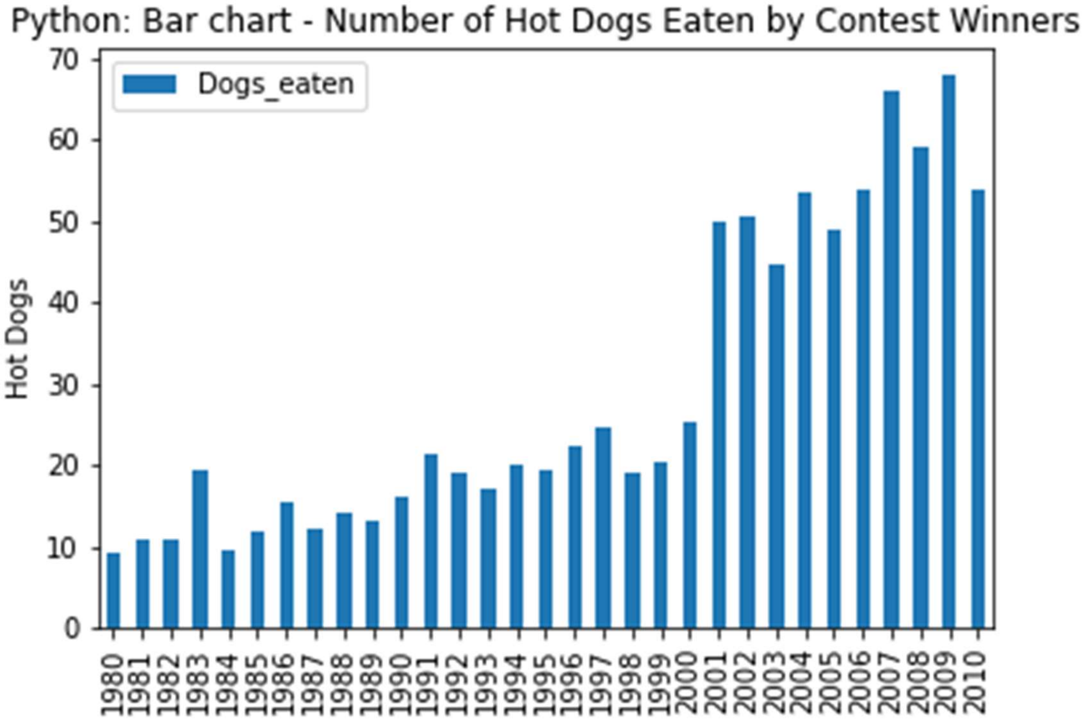


Power BI: Pie chart - 1980-2010 Hot Dog Contest Winners by Country

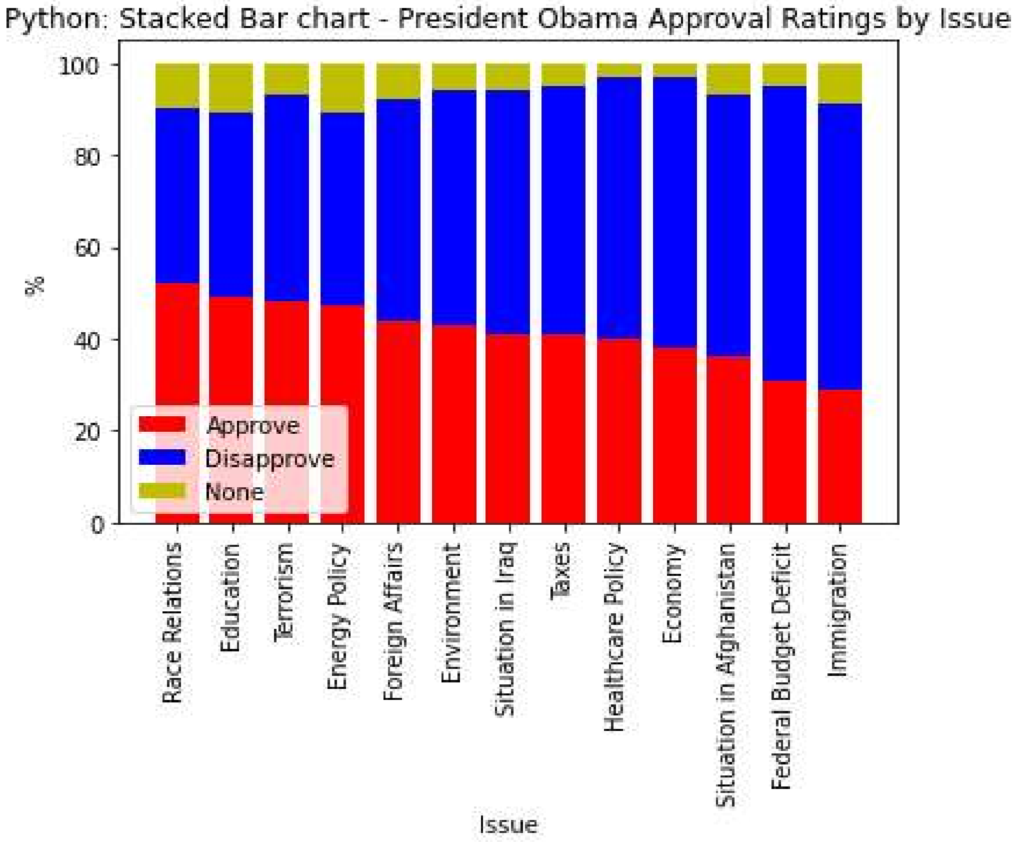


Power BI: Donut chart - 1980-2010 Hot Dog Contest Winners by Country

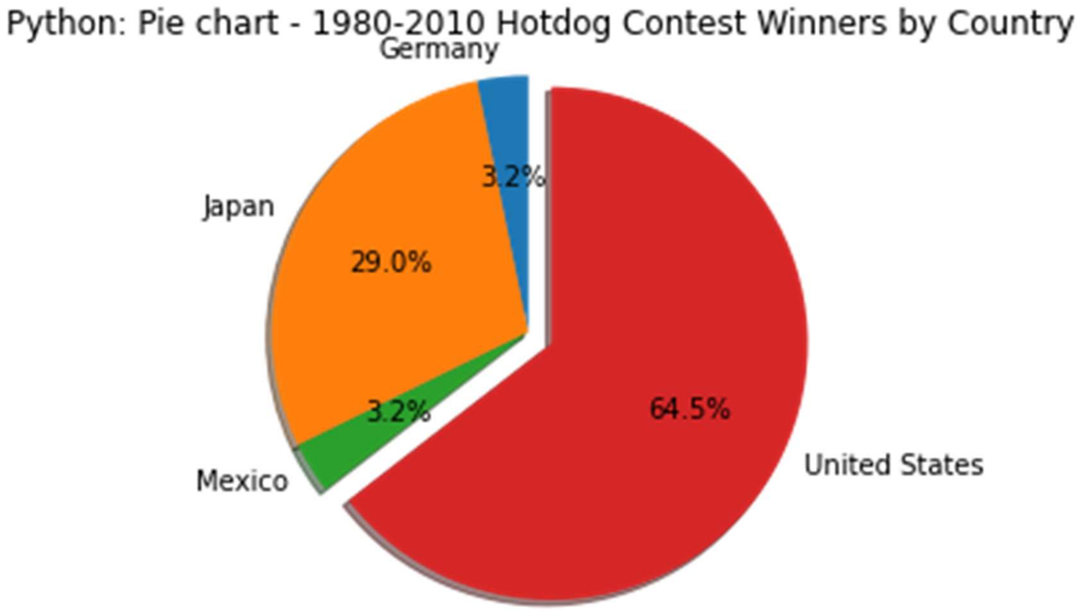


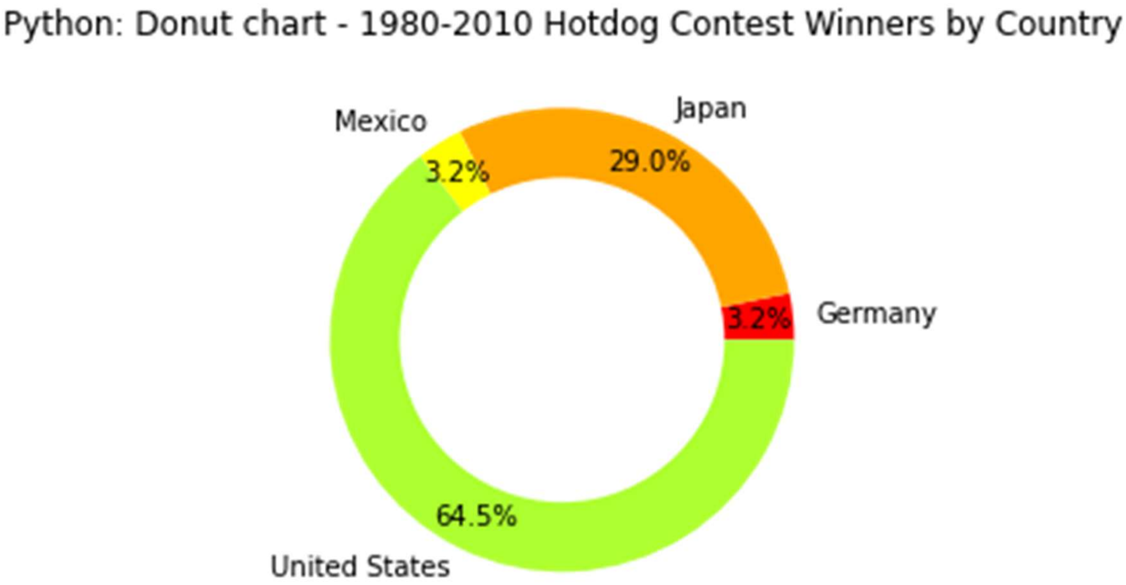


Python: Bar chart - Number of Hot Dogs Eaten by Contest Winners

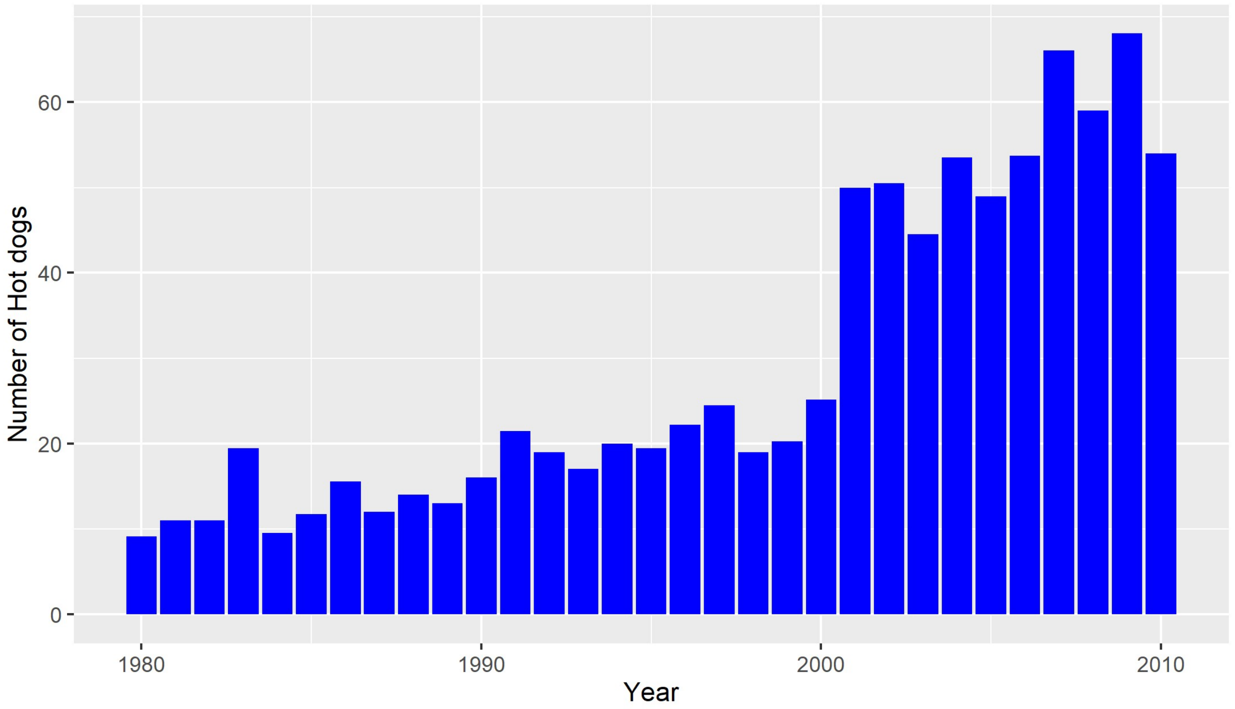


Python: Stacked Bar chart - President Obama Approval Ratings by Issue

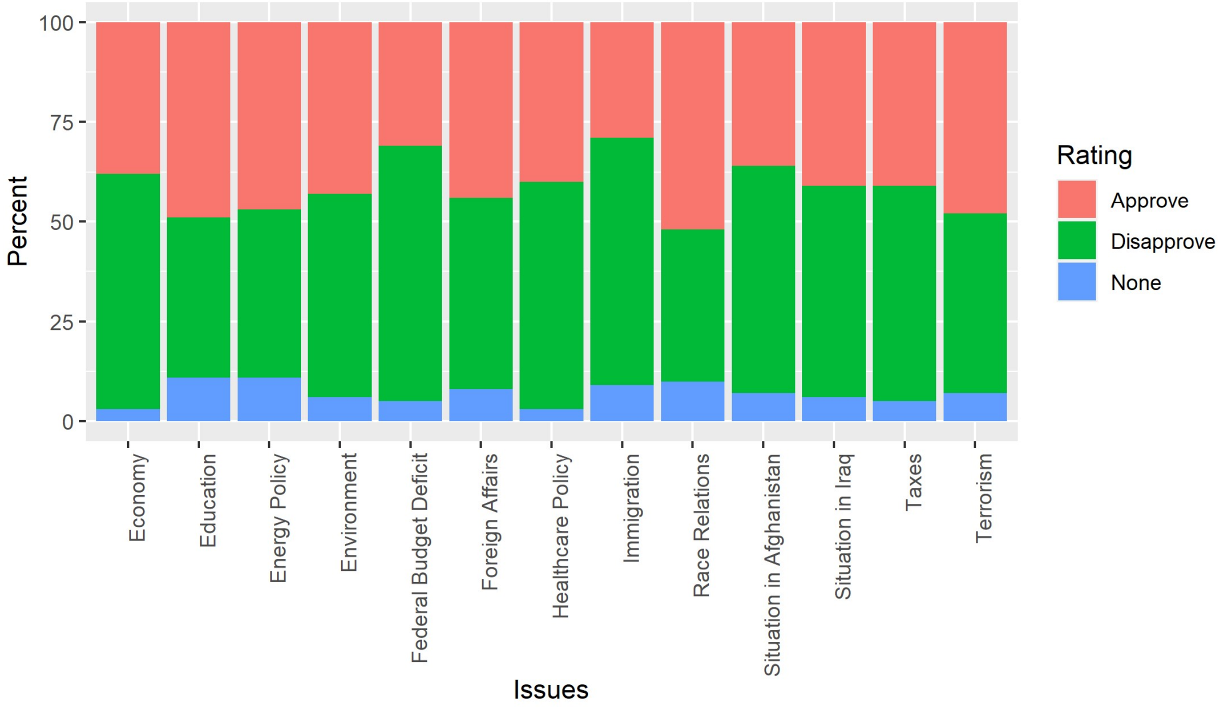

Python: Pie chart - 1980-2010 Hotdog Contest Winners by Country


Python: Donut chart - 1980-2010 Hotdog Contest Winners by Country

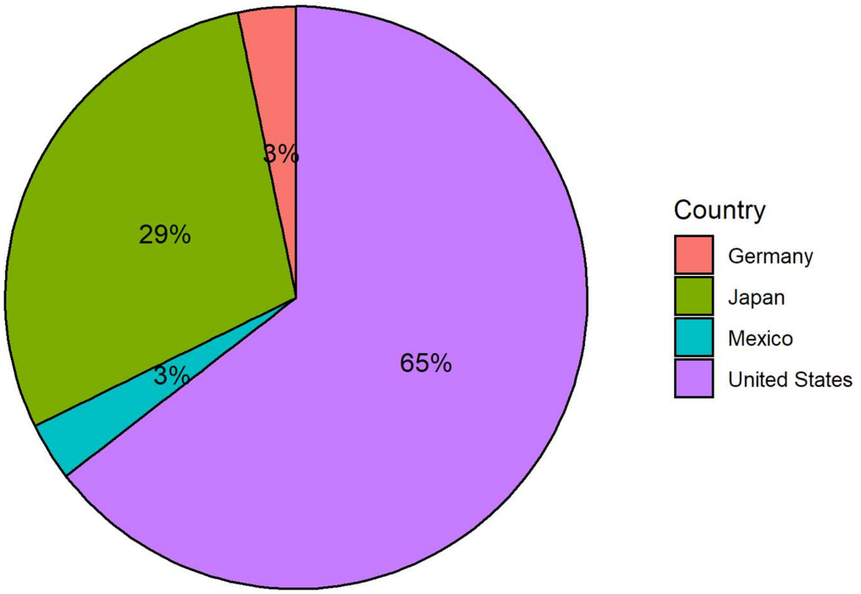
R: Bar chart - Number of Hot Dogs Eaten by Contest Winners



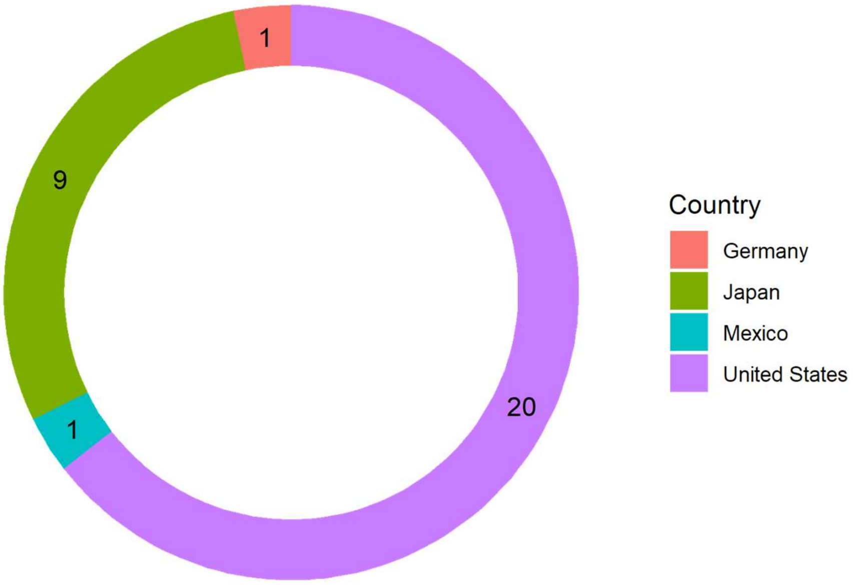
R: Stacked Bar chart - President Obama Approval Ratings by Issue



R: Pie chart - 1980-2010 Hotdog Contest Winners by Country



R: Donut chart - 1980-2010 Hotdog Contest Winners by Country



APPENDIX

In [1]:

```
#load libraries
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:

```
#import data as dataframe
data = pd.read_excel('hotdog-contest-winners.xlsm', index_col=0)
```

In [3]:

```
# review df
data.info()
```

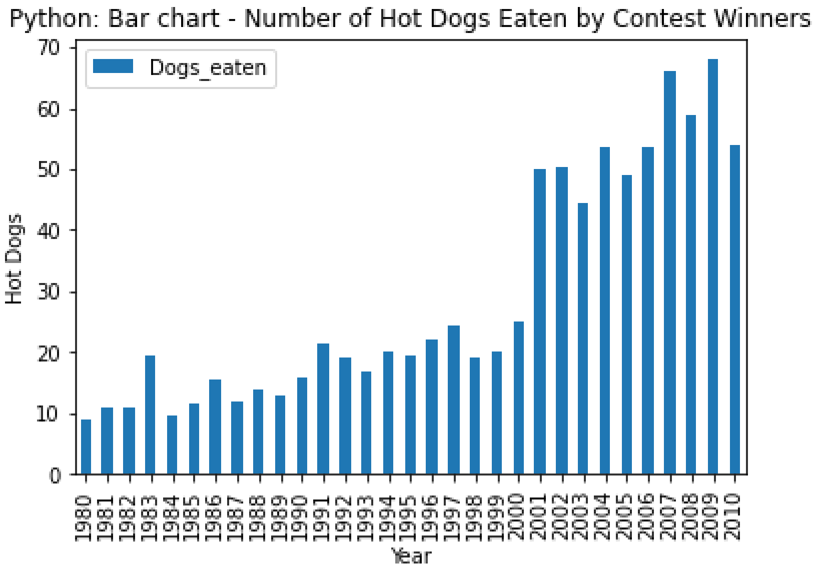
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 31 entries, 1980 to 2010
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Winner      31 non-null     object 
 1   Dogs_eaten  31 non-null     float64
 2   Country     31 non-null     object 
 3   New_record  31 non-null     int64  
dtypes: float64(1), int64(1), object(2)
memory usage: 1.2+ KB
```

In [4]:

```
#data
```

PYTHON BAR CHART

In [5]:
```
# x-axis uses the index, don't define
data.plot(y='Dogs_eaten', kind='bar')
plt.title('Python: Bar chart - Number of Hot Dogs Eaten by Contest Winners')
plt.xlabel('Year')
plt.ylabel('Hot Dogs')

#Save chart file
plt.savefig('PYTHON BAR CHART.png')
```



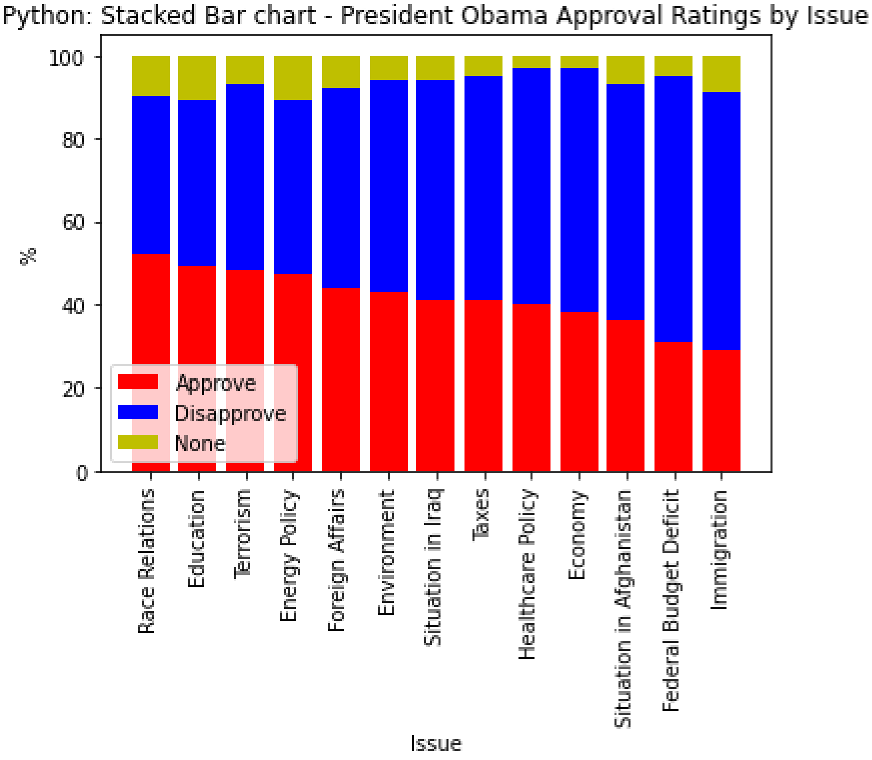
PYTHON STACKED BAR CHART

In [6]:
```
#import data as dataframe
data2 = pd.read_excel('obama-approval-ratings.xls')
```

In [7]:
```
#data2
```

In [8]:

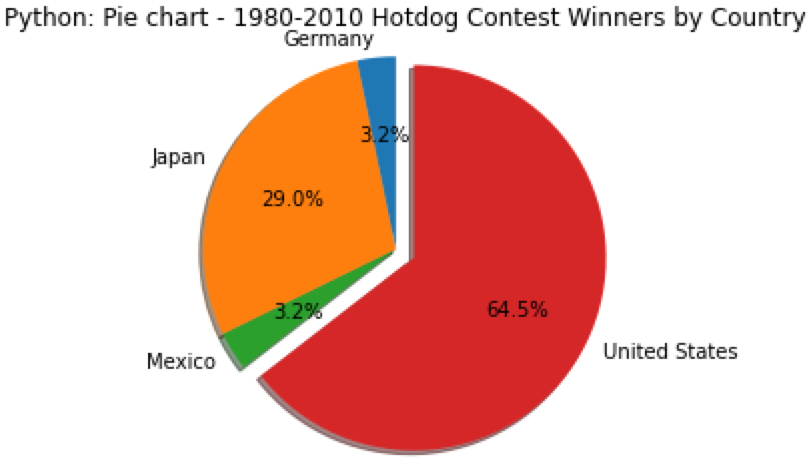
```
#data setup for stacked chart
x=data2['Issue']
y1=data2['Approve']
y2=data2['Disapprove']
y3=data2['None']

# plot
plt.bar(x, y1, color='r')
plt.bar(x, y2, bottom=y1, color='b')
plt.bar(x, y3, bottom=y1+y2, color='y')
plt.xlabel("Issue")
plt.xticks(rotation=90)
plt.ylabel("%")
plt.legend(["Approve", "Disapprove", "None"])
plt.title("Python: Stacked Bar chart - President Obama Approval Ratings by Issue'

#Save chart file
plt.savefig('PYTHON STACKED BAR CHART.png')
```



PYTHON PIE CHART

In [9]:
```
# groupby
piechrt = data.groupby('Country')['Winner'].count().reset_index()
```

In [10]:
```
#REF: https://matplotlib.org/stable/gallery/pie_and_polar_charts/pie_features.htm
# Pie chart, where the slices will be ordered and plotted counter-clockwise:
explode = (0, 0, 0, .1)  # "explode" the 4th slice

fig1, ax1 = plt.subplots()
ax1.pie(piechrt['Winner'], explode=explode, labels=piechrt['Country'], autopct='%
        shadow=True, startangle=90)
ax1.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
plt.title('Python: Pie chart - 1980-2010 Hotdog Contest Winners by Country')

#Save chart file
plt.savefig('PYTHON PIE CHART.png')
```



PYTHON DONUT CHART

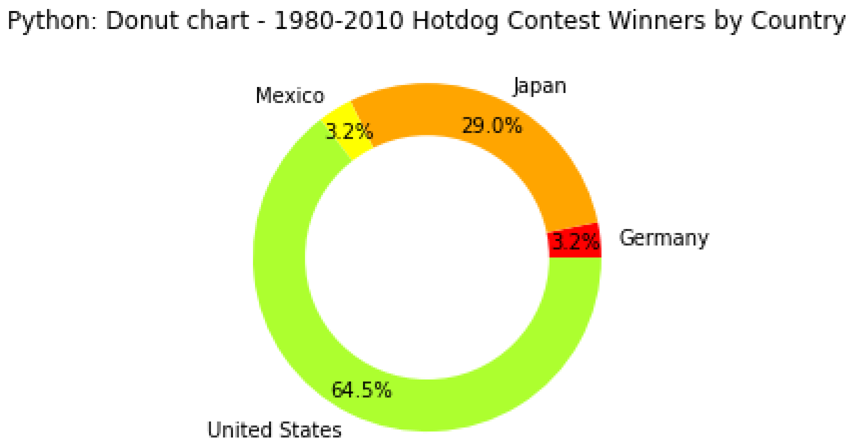
In [11]:
```
# groupby
donut = data.groupby('Country')['Winner'].count().reset_index()
```

In [12]:
```
donut
```

Out[12]:

	Country	Winner
0	Germany	1
1	Japan	9
2	Mexico	1
3	United States	20

In [13]:
```
# resource used: https://www.geeksforgeeks.org/donut-chart-using-matplotlib-in-py
# colors
colors = ['#FF0000', '#FFA500', '#FFFF00', '#ADFF2F']

# Pie Chart
plt.pie(donut['Winner'], colors=colors, labels=donut['Country'],
        autopct='%1.1f%%', pctdistance=0.85)

# draw circle
centre_circle = plt.Circle((0, 0), 0.70, fc='white')
fig = plt.gcf()

# Add Circle in Pie chart
fig.gca().add_artist(centre_circle)

# Title of chart
plt.title('Python: Donut chart - 1980-2010 Hotdog Contest Winners by Country')

#Save chart file
plt.savefig('PYTHON DONUT CHART.png')
```



Week 1 & 2

Code ▾

Hide

```
library(ggplot2)
```

```
Learn more about the underlying theory at https://ggplot2-book.org/
```

Hide

```
library(tidyverse)
```

```
Registered S3 methods overwritten by 'dbplyr':
  method         from
  print.tbl_lazy
  print.tbl_sql
-- Attaching packages --------------------------------------------------------------------------
------------------------------------------------------------- tidyverse 1.3.1 --
v tibble  3.1.6     v dplyr   1.0.7
v tidyr   1.1.4     v stringr 1.4.0
v readr   2.1.0     v forcats 0.5.1
v purrr   0.3.4
-- Conflicts -----------------------------------------------------------------------------------
------------------------------------------------------- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

Hide

```
library(readxl)
library(tidyr)
library(dplyr)
```

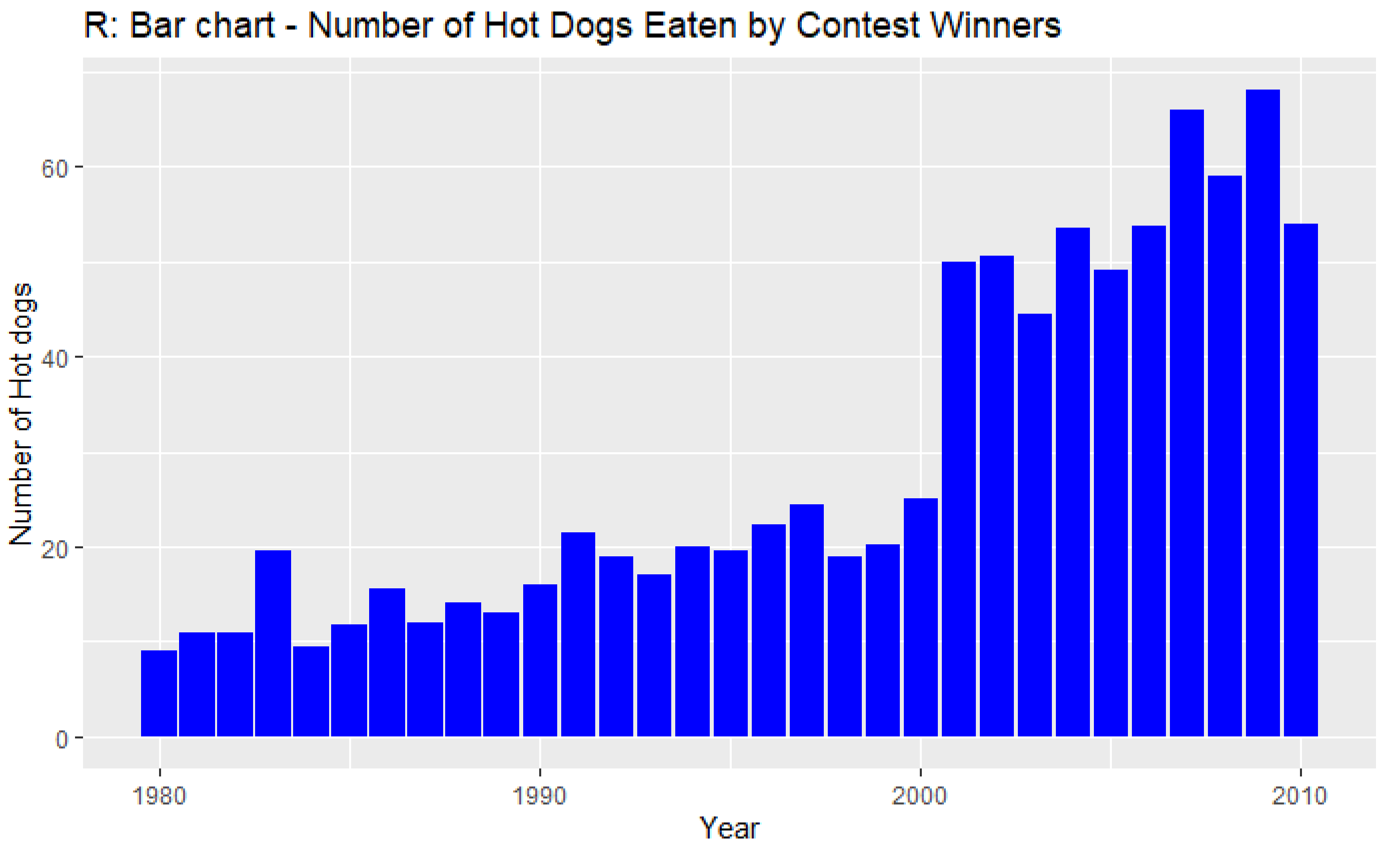
Hide

```
data = read_excel("C:\\Users\\longr\\Documents\\DSC 640\\Week 1 & 2\\hotdog-contest-winners.xls
m")
```

Hide

```
#Bar Plot in R
ggplot(data=data, aes(x=Year, y=Dogs_eaten))+
  geom_bar(stat="identity",fill="blue")+
  ggtitle("R: Bar chart - Number of Hot Dogs Eaten by Contest Winners")+
  xlab("Year")+ylab("Number of Hot dogs")

ggsave('R Bar chart.png')
```

```
Saving 7.29 x 4.5 in image
```



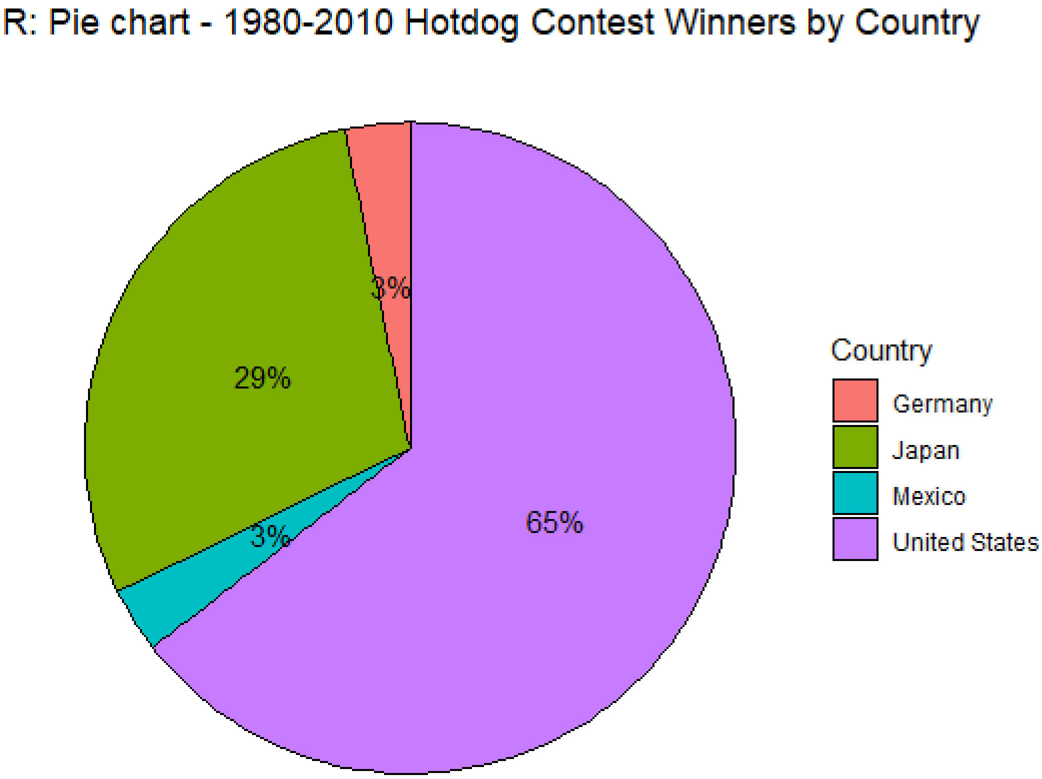
Hide

```
#create table count by country
counts = data %>% count(Country)
```

Hide

```
# Pie chart in R

# reference: https://r-charts.com/part-whole/pie-chart-percentages-ggplot2/
counts <- counts %>%
  mutate(perc = `n` / sum(`n`)) %>%
  arrange(perc) %>%
  mutate(labels = scales::percent(perc))


#plot pie chart
ggplot(counts, aes(x="", y=n, fill=Country)) +
  geom_bar(stat="identity", width=1,color="black") +
  coord_polar("y", start=0)+
  theme_void()+
  geom_text(aes(label = labels),
            position = position_stack(vjust = 0.5))+
  ggtitle("R: Pie chart - 1980-2010 Hotdog Contest Winners by Country")

ggsave('R Pie chart.png')
```

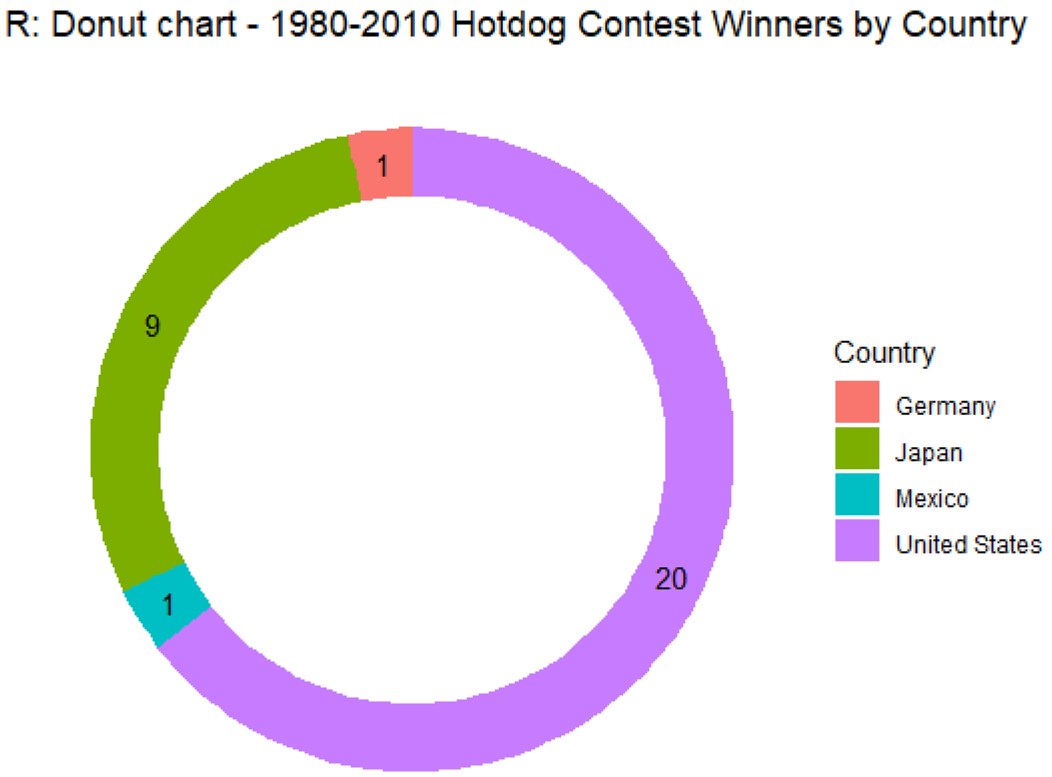
```
Saving 7.29 x 4.5 in image
```



Hide

```
# Donut chart in R

#ref: https://r-charts.com/part-whole/donut-chart-ggplot2/#basic

#hole size
hsize <- 4

counts <- counts %>%
  mutate(x = hsize)

#plot
ggplot(counts, aes(x = hsize, y = n, fill = Country)) +
  geom_col() +
  geom_text(aes(label = n),
            position = position_stack(vjust = 0.5)) +
  coord_polar(theta = "y") +
  xlim(c(0.2, hsize + 0.5)) +
  ggtitle("R: Donut chart - 1980-2010 Hotdog Contest Winners by Country")+
  theme_void()
ggsave('R Donut chart.png')
```

```
Saving 7.29 x 4.5 in image
```



Hide

```
#import the obama data
data2 = read_excel("C:\\Users\\longr\\Documents\\DSC 640\\Week 1 & 2\\obama-approval-ratings.xls")
```

Hide

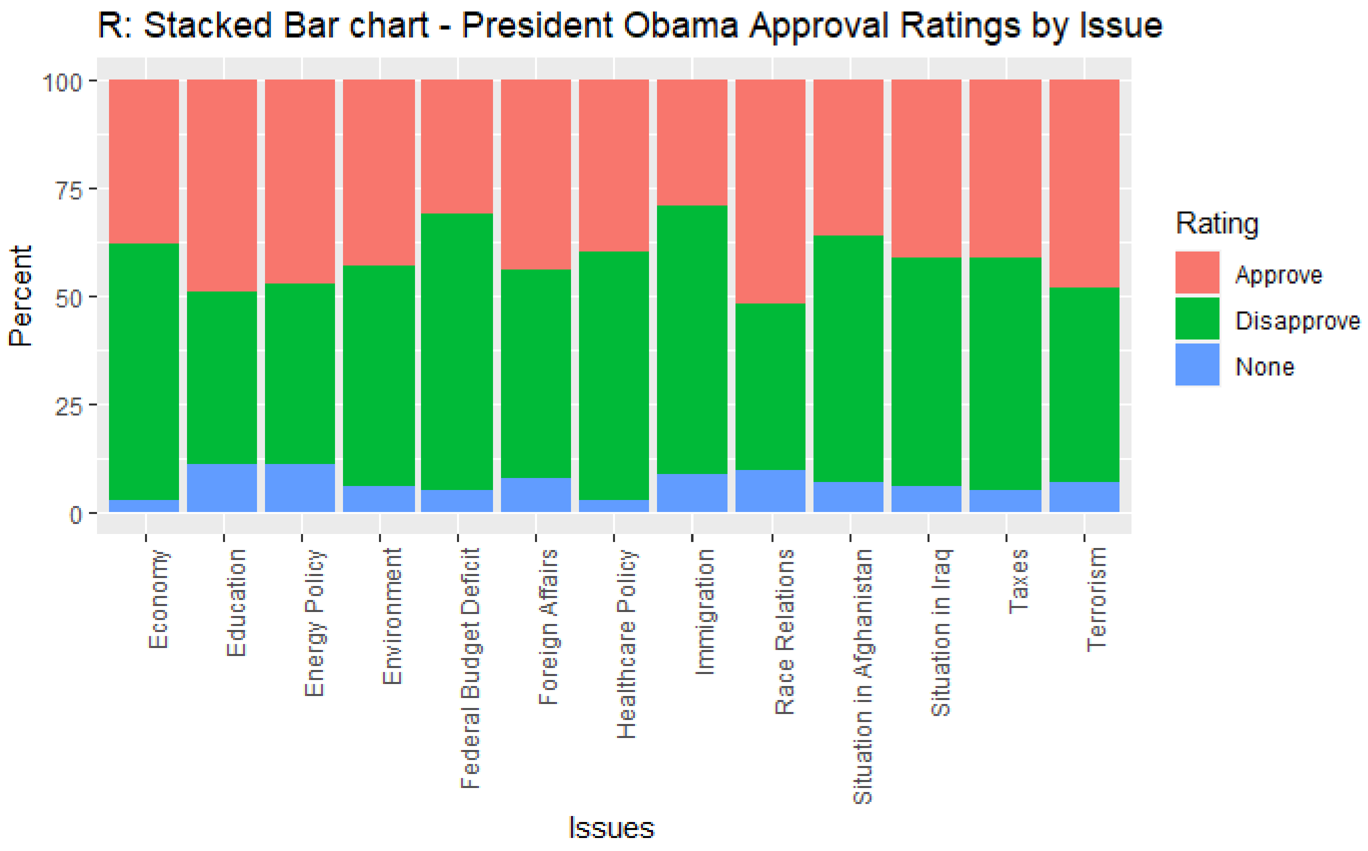
```
#create a pivot for the stacked bar chart
pivot <- pivot_longer(data2, Approve:Disapprove:None, names_to = "Rating", values_to = "Number")
```

```
Warning in x:y :
  numerical expression has 2 elements: only the first used
```

Hide

```
# Stacked Bar Chart in R

# Reference: https://www.r-graph-gallery.com/48-grouped-barplot-with-ggplot2.html

ggplot(pivot, aes(fill=Rating, y=Number, x=Issue)) +
  geom_bar(position="stack", stat="identity") +
  theme(axis.text.x=element_text(angle=90,hjust=1)) +
  xlab("Issues")+ylab("Percent")+
  ggtitle("R: Stacked Bar chart - President Obama Approval Ratings by Issue")
ggsave('R Stacked Bar Chart.png')
```

```
Saving 7.29 x 4.5 in image
```


R: Stacked Bar chart - President Obama Approval Ratings by Issue
Percent
Economy
Education
Energy Policy
Environment
Federal Budget Deficit
Foreign Affairs
Healthcare Policy
Immigration
Race Relations
Situation in Afghanistan
Situation in Iraq
Taxes
Terrorism
Issues
Rating
Approve
Disapprove
None