In [1]:
```
# Dependencies
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
# import locale
# import requests
# import os
import csv
import scipy.stats as stats
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
# Collecting base data...
temp_df = pd.read_csv("Resources/election_all.csv")
pop_df = pd.read_csv("Resources/population2000-2016ALL.csv")
#v2000 = pd.read_csv("Resources/FOR_JINHO_2000.csv.csv")
#v2004 = pd.read_csv("Resources/FOR_JINHO_2004.csv.csv")
v2008 = pd.read_csv("Resources/FOR_JINHO_2008.csv")
v2012 = pd.read_csv("Resources/FOR_JINHO_2012.csv")
#v2016 = pd.read_csv("Resources/FOR_JINHO_2016.csv.csv")
v2008.head()
```

Out[2]:

	Location	total_votes	2008_democrat_votes	2008_republican_votes
0	Alabama, Autauga	23641	6093.0	17403.0
1	Alabama, Baldwin	81413	19386.0	61271.0
2	Alabama, Barbour	11630	5697.0	5866.0
3	Alabama, Bibb	8644	2299.0	6262.0
4	Alabama, Blount	24267	3522.0	20389.0

In [3]:
```
# Creating master State, County details...
master_location_df = temp_df[['state','county','state_po']]
master_location_df = master_location_df.drop_duplicates()
master_location_df['Location'] = master_location_df.state+", "+master_location_df.cc
master_location_df.columns = ['State', 'County', 'State_po', 'Location']
master_location_df = master_location_df.dropna()
master_location_df.to_csv(r'Resources/master_state_info.csv', index=False, header=Tr
print(len(master_location_df))
master_location_df.head()
```

```
3150
```

Out[3]:

	State	County	State_po	Location
0	Alabama	Autauga	AL	Alabama, Autauga
4	Alabama	Baldwin	AL	Alabama, Baldwin
8	Alabama	Barbour	AL	Alabama, Barbour
12	Alabama	Bibb	AL	Alabama, Bibb
16	Alabama	Blount	AL	Alabama, Blount

In [4]:
```
# Collecting required fields...
# pop_df = pop_df[['Location','2000','2004','2008','2012','2016']]
pop_df = pop_df[['Location','2008']]
# print(len(pop_df))
# pop_df.head()
pop_final = pd.merge(pop_df, master_location_df, on='Location')
pop_final
# Creating summary data by state...
pop_state_summary = pop_final.groupby('State')
pop_state_summary = pop_state_summary.sum()

pop_county_summary = pop_final.groupby('Location')
pop_county_summary = pop_county_summary.sum()
pop_county_summary.head()
len(pop_county_summary)
```

Out[4]: 3001

In [5]:
```
# Creating summary for State...
v2008_final = pd.merge(v2008, master_location_df, on='Location')
v2008_final['2008_other_votes'] = v2008_final["total_votes"]-v2008_final["2008_democ
v2008_state_summary = v2008_final.groupby('State')
v2008_state_summary = v2008_state_summary.sum()
# v2008_state_summary.head()
v2008_state_summary['year'] = '2008'
year = v2008_state_summary['year'].iloc[0]
# print(v2008_state_summary['total_votes'].min()/1000, v2008_state_summary['total_vo
#
# merge population data to calcuate avg.
#
v2008_state_summary = v2008_state_summary.merge(pop_state_summary, on='State')
v2008_state_summary['avg_votes'] = round(v2008_state_summary['total_votes']*100/v200
v2008_state_summary['2008_democrat_votes'] = round(v2008_state_summary['2008_democra
v2008_state_summary['2008_republican_votes'] = round(v2008_state_summary['2008_repub
v2008_state_summary['2008_other_votes'] = round(v2008_state_summary['2008_other_vote
hv_st = v2008_state_summary['avg_votes'].max()
highst = v2008_state_summary.loc[v2008_state_summary['avg_votes'] == hv_st]
hv_st = highst.index.tolist()
lv_st = v2008_state_summary['avg_votes'].min()
lowst = v2008_state_summary.loc[v2008_state_summary['avg_votes'] == lv_st]
lv_st = lowst.index.tolist()
v2008_state_summary.head()
```

Out[5]:

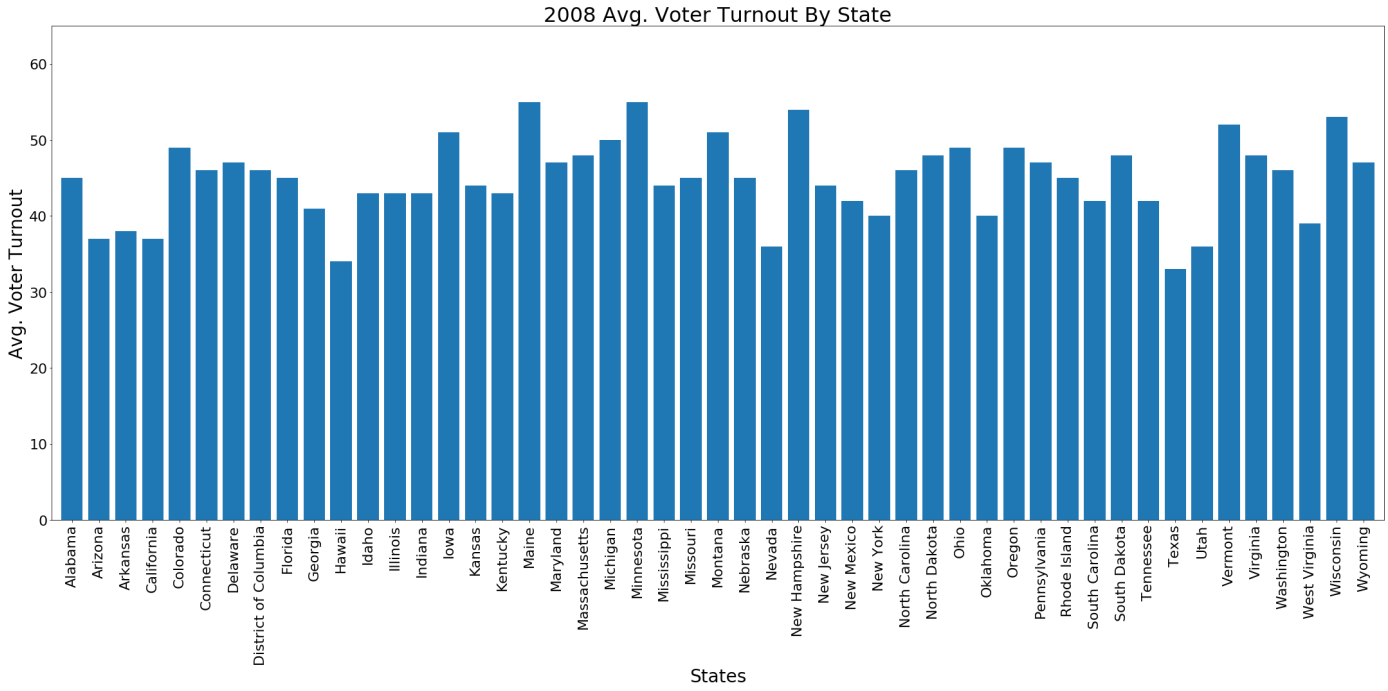
State	total_votes	2008_democrat_votes	2008_republican_votes	2008_other_votes	year	2008	avg_vote
Alabama	2099819	17.0	27.0	0.0	2008	4718206	45.
Arizona	2293475	16.0	20.0	0.0	2008	6280362	37.
Arkansas	1086617	15.0	22.0	1.0	2008	2874554	38.
California	13561900	23.0	14.0	1.0	2008	36604337	37.
Colorado	2401361	26.0	22.0	1.0	2008	4889730	49.

In [6]:
```
# Creating summary for Counties..
v2008_county_summary = v2008_final.groupby('Location', as_index=False)
v2008_county_summary = v2008_county_summary.sum()
# print(v2008_county_summary['total_votes'].min(), v2008_county_summary['total_votes
v2008_county_summary = v2008_county_summary.merge(pop_county_summary, on='Location')
v2008_county_summary['avg_votes'] = round(v2008_county_summary['total_votes']*100/v2
v2008_county_summary['2008_democrat_votes'] = round(v2008_county_summary['2008_democ
v2008_county_summary['2008_republican_votes'] = round(v2008_county_summary['2008_rep
v2008_county_summary['2008_other_votes'] = round(v2008_county_summary['2008_other_vo
hv_cy = v2008_county_summary['avg_votes'].max()
highcy = v2008_county_summary.loc[v2008_county_summary['avg_votes'] == hv_cy]
hv_cy = highcy.Location.tolist()
lv_cy = v2008_county_summary['avg_votes'].min()
lowcy = v2008_county_summary.loc[v2008_county_summary['avg_votes'] == lv_cy]
lv_cy = lowcy.Location.tolist()
v2008_county_summary.head()
```

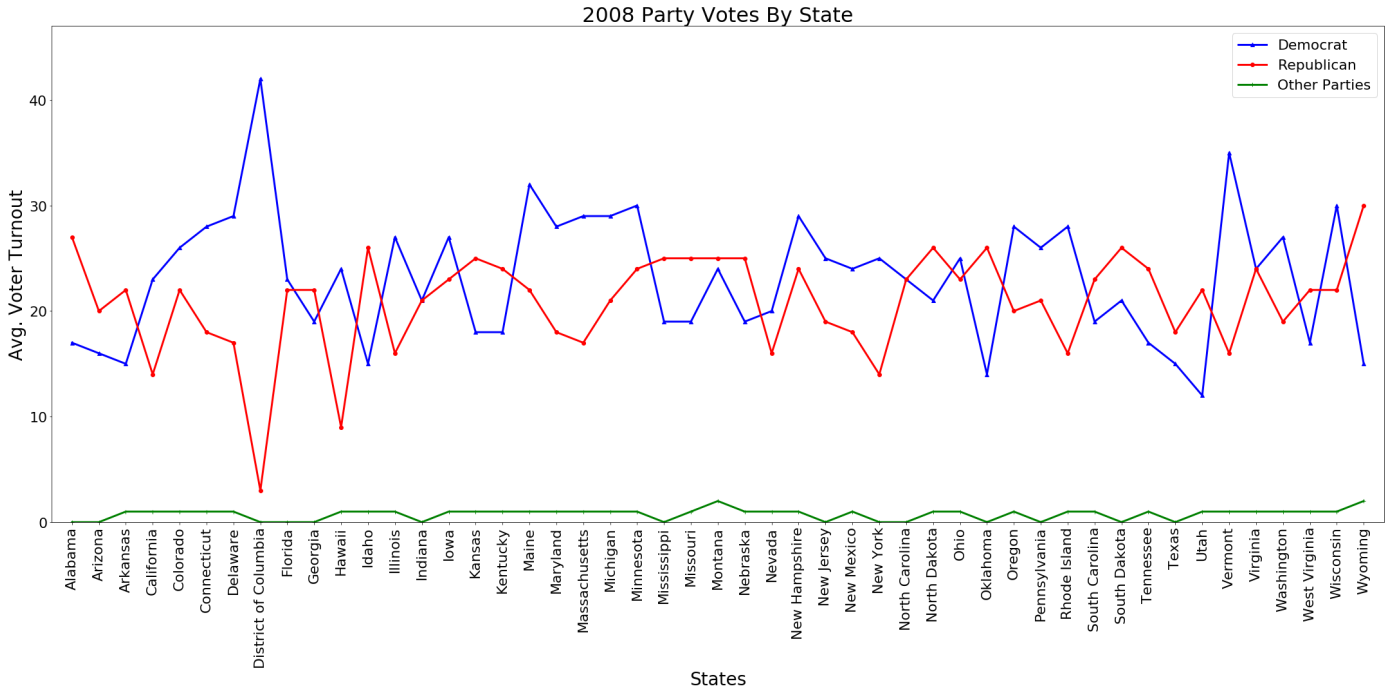
Out[6]:

	Location	total_votes	2008_democrat_votes	2008_republican_votes	2008_other_votes	2008	avg_votes
0	Alabama, Autauga	23641	11.0	33.0	0.0	53277	44.0
1	Alabama, Baldwin	81413	11.0	35.0	0.0	175827	46.0
2	Alabama, Barbour	11630	20.0	21.0	0.0	27808	42.0
3	Alabama, Bibb	8644	10.0	28.0	0.0	22705	38.0
4	Alabama, Blount	24267	6.0	36.0	1.0	57055	43.0

In [7]:
```
# Voter turnout by state...
#
title = year+' Avg. Voter Turnout By State'
pfile = title.replace(" ",'_')+'.png'
pfile = f"Resources/Images/{pfile}"
plt.figure(figsize=(30,15))
plt.rc('axes', titlesize=32)
plt.rc('axes', labelsize=28)
plt.rc('xtick', labelsize=22)
plt.rc('ytick', labelsize=22)
plt.bar(v2008_state_summary.index, v2008_state_summary.avg_votes)
plt.xticks(rotation=90)
plt.yticks(rotation=360)
plt.xlabel('States')
plt.ylabel("Avg. Voter Turnout")
plt.title(title)
plt.xlim(-0.75, len(v2008_state_summary.index)-.25)
plt.ylim(0, v2008_state_summary.avg_votes.max()+10)
plt.tight_layout()
plt.savefig(pfile)
plt.show()
```

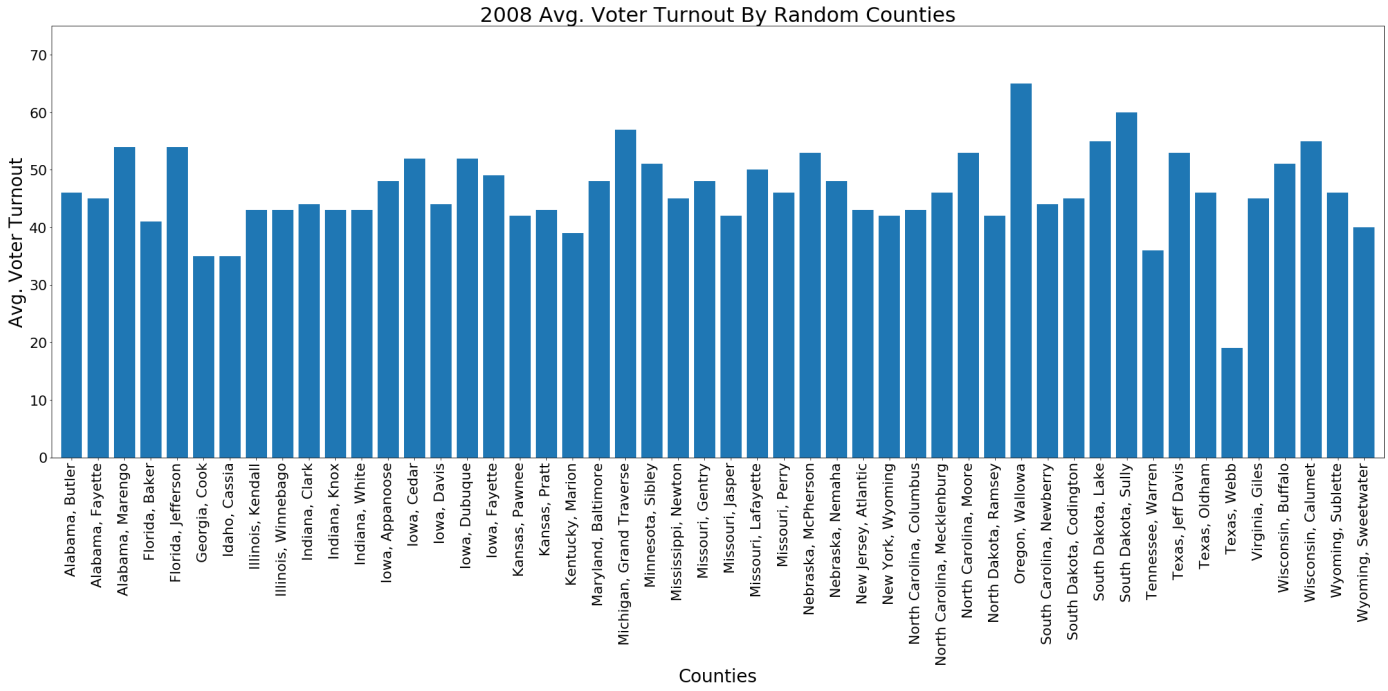


In [8]:
```
# Avg. Party votes by state...
#
plt.figure(figsize=(30,15))
plt.rc('axes', titlesize=32)
plt.rc('axes', labelsize=28)
plt.rc('xtick', labelsize=22)
plt.rc('ytick', labelsize=22)
plt.rc('legend', fontsize=22)
title = year+' Party Votes By State'
pfile = title.replace(" ",'_')+'.png'
pfile = f"Resources/Images/{pfile}"
maxy = v2008_state_summary['2008_democrat_votes'].max()
if maxy < v2008_state_summary['2008_republican_votes'].max():
    maxy = v2008_state_summary['2008_republican_votes'].max()
Democrat = plt.plot(v2008_state_summary.index, v2008_state_summary["2008_democrat_vo
Republican = plt.plot(v2008_state_summary.index, v2008_state_summary["2008_republica
Others = plt.plot(v2008_state_summary.index, v2008_state_summary["2008_other_votes"]
plt.xticks(rotation=90)
plt.yticks(rotation=360)
plt.xlabel('States')
plt.ylabel("Avg. Voter Turnout")
plt.title(title)
plt.xlim(-0.75, len(v2008_state_summary.index)-.25)
plt.ylim(0, maxy+5)
plt.legend(loc="best")
plt.tight_layout()
plt.savefig(pfile)
plt.show()
```

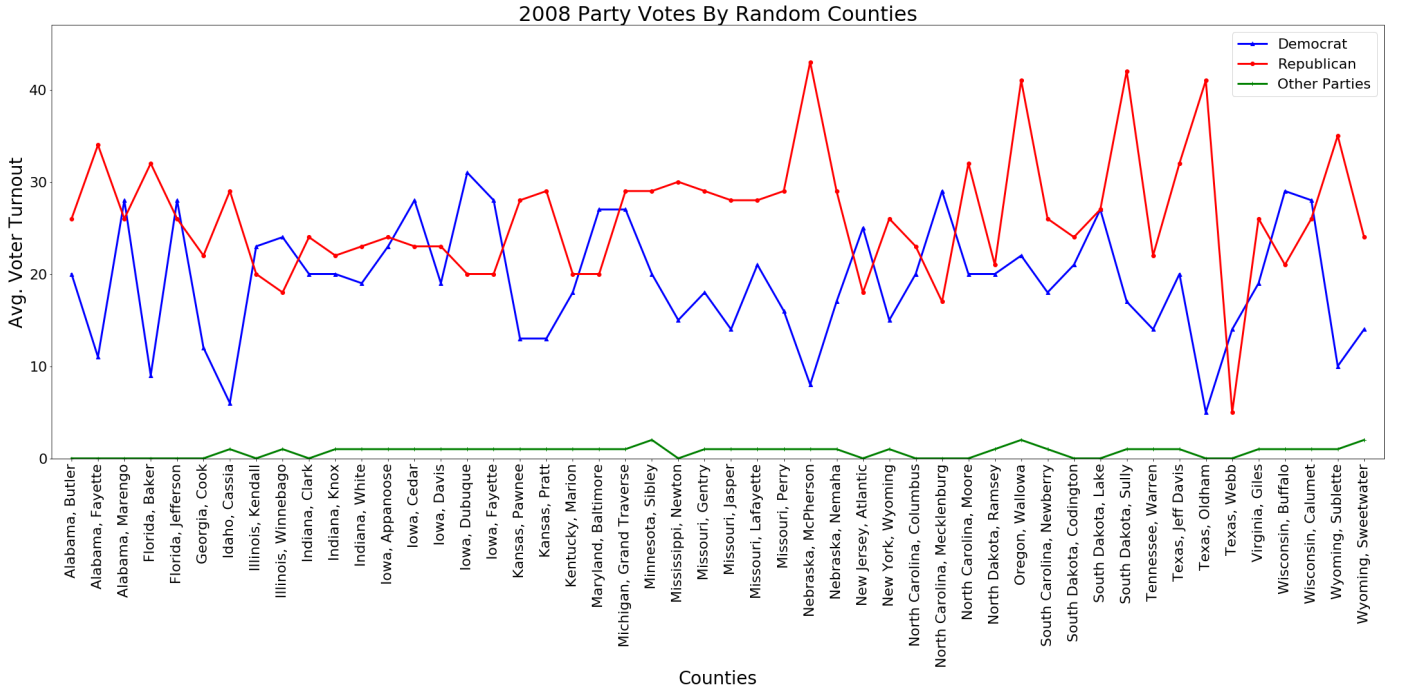


In [9]:
```
# Generating Random 50 counties...
random_sample_county = pd.DataFrame(v2008_county_summary.Location.sample(n=50, randc
random_sample_county.head()
v2008_filtered_county_summary = pd.merge(v2008_county_summary,random_sample_county,
# v2008_filtered_county_summary.head()
```

In [10]:
```
# Avg. Voter turnout by random counties...
#
title = year+' Avg. Voter Turnout By Random Counties'
pfile = title.replace(" ",'_')+'.png'
pfile = f"Resources/Images/{pfile}"
plt.figure(figsize=(30,15))
plt.rc('axes', titlesize=32)
plt.rc('axes', labelsize=28)
plt.rc('xtick', labelsize=22)
plt.rc('ytick', labelsize=22)
plt.bar(v2008_filtered_county_summary.Location, v2008_filtered_county_summary.avg_vo
plt.xticks(rotation=90)
plt.yticks(rotation=360)
plt.xlabel('Counties')
plt.ylabel("Avg. Voter Turnout")
plt.title(title)
plt.xlim(-0.75, len(v2008_filtered_county_summary.Location)-.25)
plt.ylim(0, v2008_filtered_county_summary.avg_votes.max()+10)
plt.tight_layout()
plt.savefig(pfile)
plt.show()
```



In [11]:
```
# Party votes by random counties...
#
plt.figure(figsize=(30,15))
title = year+' Party Votes By Random Counties'
pfile = title.replace(" ",'_')+'.png'
pfile = f"Resources/Images/{pfile}"
plt.rc('axes', titlesize=32)
plt.rc('axes', labelsize=28)
plt.rc('xtick', labelsize=22)
plt.rc('ytick', labelsize=22)
plt.rc('legend', fontsize=22)
maxy = v2008_state_summary['2008_democrat_votes'].max()
if maxy < v2008_state_summary['2008_republican_votes'].max():
    maxy = v2008_state_summary['2008_republican_votes'].max()
Democrat = plt.plot(v2008_filtered_county_summary.Location, v2008_filtered_county_su
Republican = plt.plot(v2008_filtered_county_summary.Location, v2008_filtered_county_
Others = plt.plot(v2008_filtered_county_summary.Location, v2008_filtered_county_summ
plt.xticks(rotation=90)
plt.yticks(rotation=360)
plt.xlabel('Counties')
plt.ylabel("Avg. Voter Turnout")
plt.title(title)
plt.xlim(-0.75, len(v2008_filtered_county_summary.Location)-.25)
plt.ylim(0, maxy+5)
plt.legend(loc="best")
plt.tight_layout()
plt.savefig(pfile)
plt.show()
```



In [12]:
```
# Highest/Lowest voter turnout...
#

print(year + " Highest Voter Turnout for the State  : " + ','.join(hv_st))
print(year + " Lowest Voter Turnout for the State   : " + ','.join(lv_st))
print("")
print(year + " Highest Voter Turnout for the County : " + ','.join(hv_cy))
print(year + " Lowest Voter Turnout for the County  : " + ','.join(lv_cy))
```

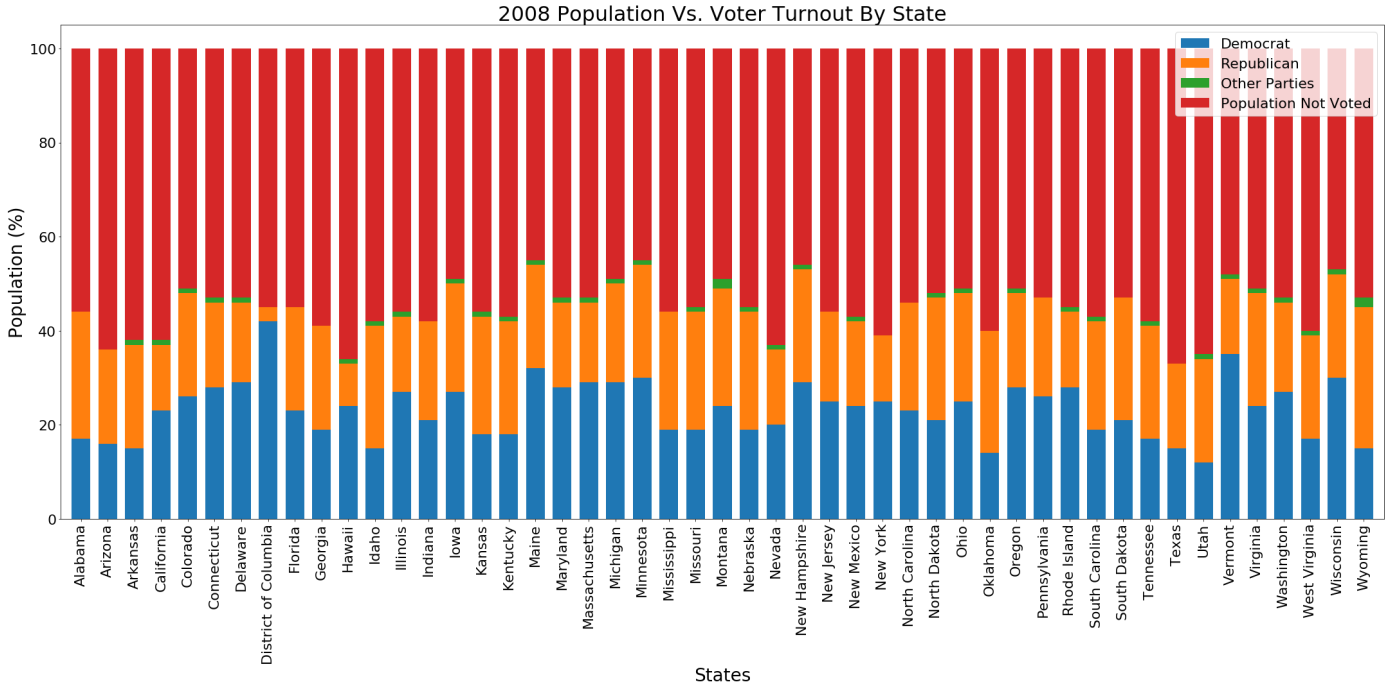
```
2008 Highest Voter Turnout for the State  : Maine,Minnesota
2008 Lowest Voter Turnout for the State   : Texas

2008 Highest Voter Turnout for the County : Texas, Loving
2008 Lowest Voter Turnout for the County  : Georgia, Chattahoochee
```

In [13]:
```
population_votes = v2008_state_summary[['2008_democrat_votes','2008_republican_votes
population_votes.columns = ['2008_democrat_votes', '2008_republican_votes','2008_oth
population_votes['pop_notvoted'] = (100-(population_votes['2008_democrat_votes']+pop
population_votes.head()
```

Out[13]:

State	2008_democrat_votes	2008_republican_votes	2008_other_votes	pop_notvoted
Alabama	17.0	27.0	0.0	56.0
Arizona	16.0	20.0	0.0	64.0
Arkansas	15.0	22.0	1.0	62.0
California	23.0	14.0	1.0	62.0
Colorado	26.0	22.0	1.0	51.0

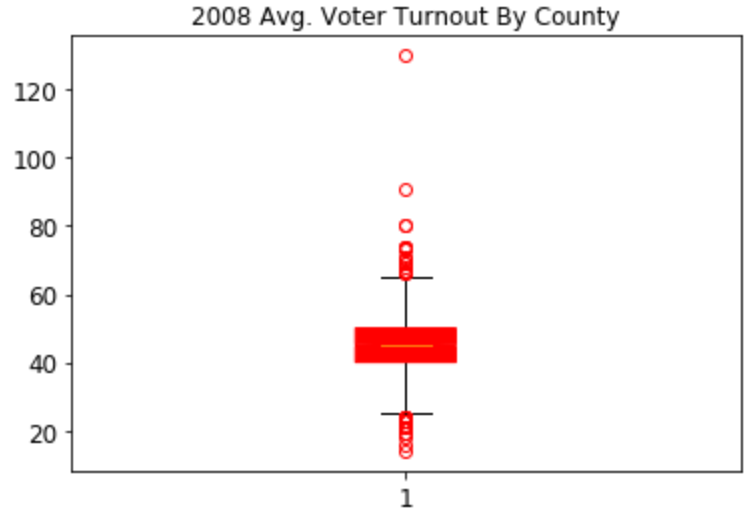
In [14]:
```
# Poplulation vs.Voter turnout by state for the year 2008...
#
title = year+' Population Vs. Voter Turnout By State'
pfile = title.replace(" ",'_')+'.png'
pfile = f"Resources/Images/{pfile}"
plt.figure(figsize=(30,15))
bottom = 0
plt.rc('axes', titlesize=32)
plt.rc('axes', labelsize=28)
plt.rc('xtick', labelsize=22)
plt.rc('ytick', labelsize=22)
label = ['Democrat','Republican','Other Parties','Population Not Voted']
#
for i in population_votes.columns:
    plt.bar(population_votes.index, population_votes[i],  width=0.7, bottom=bottom)
    bottom += population_votes[i]
#
plt.xticks(rotation=90)
plt.yticks(rotation=360)
plt.legend(label)
plt.xlabel('States')
plt.ylabel("Population (%)")
plt.title(title)
plt.xlim(-0.75, len(population_votes.index)-.25)
plt.tight_layout()
plt.savefig(pfile)
plt.show()
```



In [15]:
```
# Outliers
#
quartiles = v2008_county_summary['avg_votes'].quantile([.25,.5,.75])
lowerq = quartiles[0.25]
upperq = quartiles[0.75]
iqr = upperq-lowerq
lower_bound = lowerq - (1.5*iqr)
upper_bound = upperq + (1.5*iqr)

print(f"The lower quartile of county outliers  is :{lowerq}")
print(f"The upper quartile of county outliers  is :{upperq}")
print(f"The interquartile range of county outliers  is :{iqr}")
print(f"The median of county volume  is :{quartiles[0.5]}")
print(f"Values below {round(lower_bound,6)} could be outliers. ")
print(f"Values above {round(upper_bound,6)} could be outliers. ")
print(" ")
```

```
The lower quartile of county outliers  is :40.0
The upper quartile of county outliers  is :50.0
The interquartile range of county outliers  is :10.0
The median of county volume  is :45.0
Values below 25.0 could be outliers.
Values above 65.0 could be outliers.
```

In [16]:
```
title = year+' Avg. Voter Turnout By County'
pfile = title.replace(" ",'_')+'.png'
pfile = f"Resources/Images/{pfile}"
boxplot_data = v2008_county_summary['avg_votes']
fig = plt.figure()
plt.rc('axes', titlesize=12)
plt.rc('axes', labelsize=12)
plt.rc('xtick', labelsize=12)
plt.rc('ytick', labelsize=12)
plt.rc('legend', fontsize=12)
c = "red"
plt.boxplot(boxplot_data,  notch=True, patch_artist=True,
            boxprops=dict(facecolor=c, color=c), flierprops=dict(color=c, markeredge

plt.title(title)
plt.savefig(pfile)
plt.show()
```



In [17]:
```
pop_county = v2008_county_summary['avg_votes']
sample_county = pd.DataFrame(v2008_county_summary.avg_votes.sample(n=1000, random_st

stats.ttest_1samp(sample_county, pop_county.mean())
```

Out[17]:
```
Ttest_1sampResult(statistic=array([0.22383107]), pvalue=array([0.82293452]))
```

In [ ]:
```

```

In [ ]:
```

```