1.2 Data Report

Sadie Harper

12/4/2021

DSC630-T302 Predictive Analytics (2223-1)

This report provides a summary of population, employment, and labor data for Alaska, US. The data was retreived from the Bureau of Labor Statistics at the following link: https://data.bls.gov/cgi-bin/surveymost (https://data.bls.gov/cgi-bin/surveymost)

Table of Contents:

In [1]:
```
# Import libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:
```
# Read in data
data = pd.read_excel('Alaska_BLS.xlsx', skiprows=10)
```

Data Summary

Civilian Noninstitutional Population

In [3]:
```
# Summary statistics - civilian noninstitutional population
print('Civilian Noninstitutional Population Summary Statistics')
data[["civilian noninstitutional population"]].describe()
```

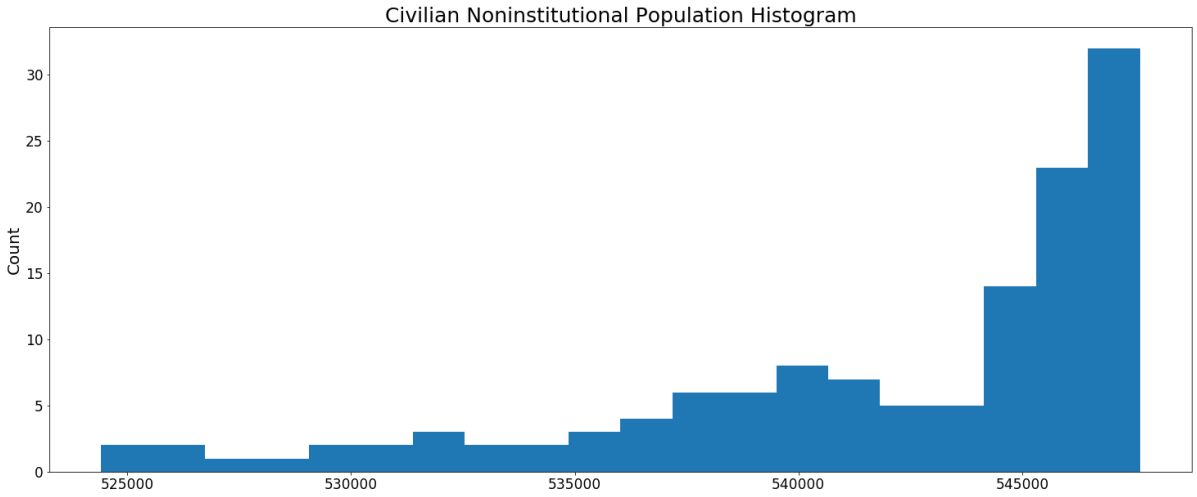
```
Civilian Noninstitutional Population Summary Statistics
```

Out[3]:

	civilian noninstitutional population
count	130.000000
mean	541982.069231
std	5841.954993
min	524423.000000
25%	539194.000000
50%	544306.000000
75%	546380.250000
max	547623.000000

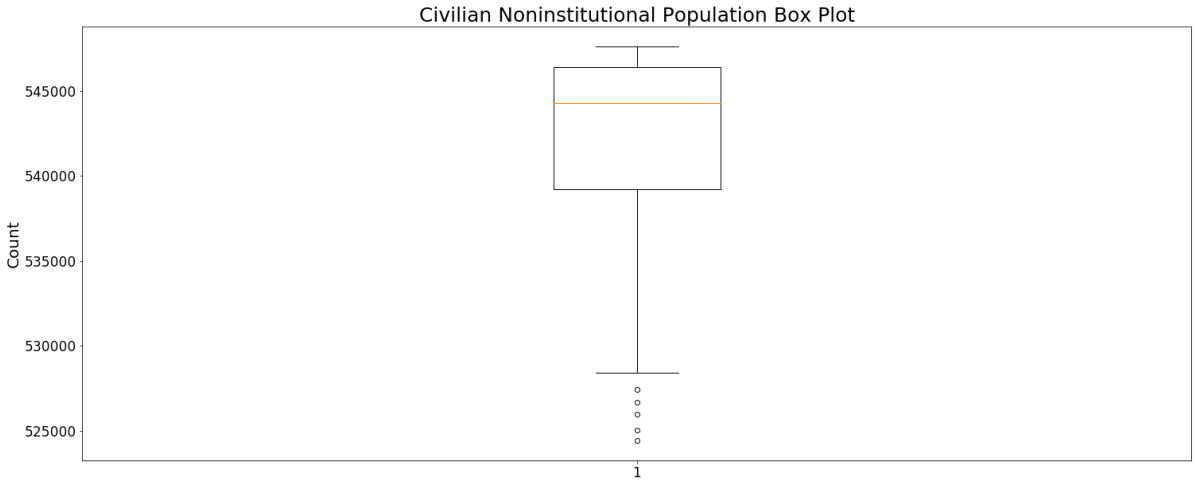
The range for the civilian noninstitutional population doesn't seem to be that large. The jump from the minimum value to 25% is the largest increase.

In [4]:
```
# Histogram - civilian noninstitutional population
plt.rcParams['figure.figsize'] = (25, 10)
plt.hist(data['civilian noninstitutional population'], bins=20)
plt.title('Civilian Noninstitutional Population Histogram', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



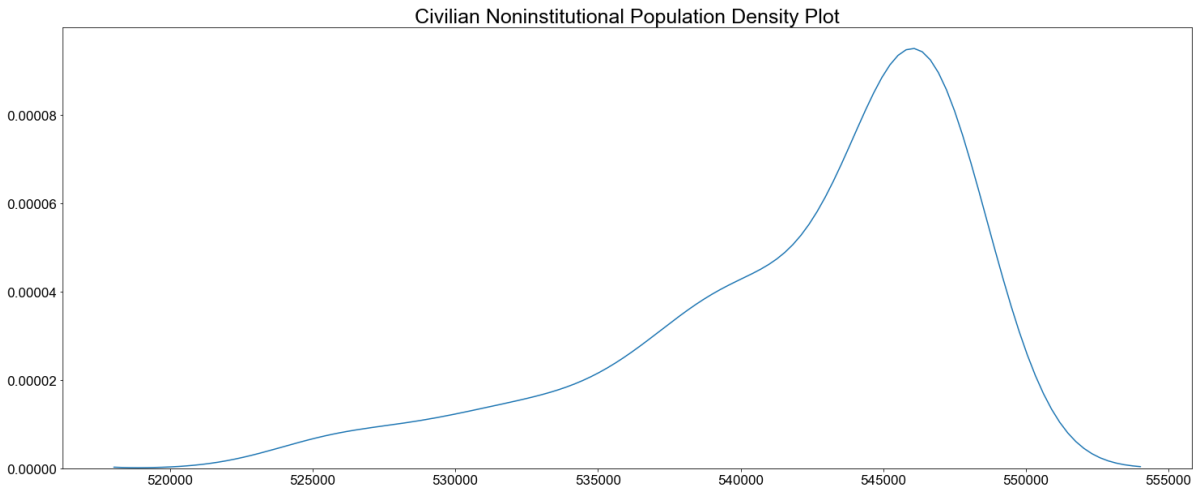
This heavily left skewed histogram shows what we were seeing with the summary statistics, that there are more large values in the data.

In [5]:
```
# Box plot - civilian noninstitutional population
plt.rcParams['figure.figsize'] = (25, 10)
plt.boxplot(data['civilian noninstitutional population'])
plt.title('Civilian Noninstitutional Population Box Plot', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



The low population numbers are appearing as outliters because the increase in population happened at a semi-fast pace.

In [6]:
```
# Density Plot - civilian noninstitutional population
plt.rcParams['figure.figsize'] = (25, 10)
plt.title('Civilian Noninstitutional Population Density Plot', fontsize=25)
plt.tick_params(labelsize=17)
sns.set_style('whitegrid')
sns.kdeplot(np.array(data['civilian noninstitutional population']))
plt.show()
```



This density plot tells the same story as the previous histogram did. We have data skewed towards the higher population numbers.

Labor Force Participation Rate

In [7]:
```
# Summary statistics - labor force participation rate
print('Labor Force Participation Rate Summary Statistics')
data[["labor force participation rate"]].describe()
```

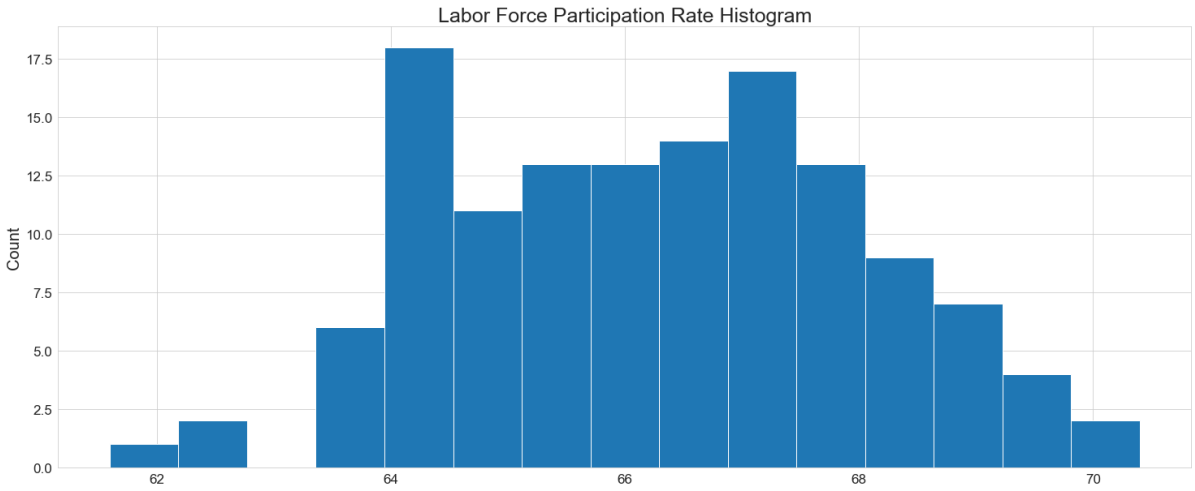
```
Labor Force Participation Rate Summary Statistics
```

Out[7]:

	labor force participation rate
count	130.000000
mean	66.300769
std	1.791474
min	61.600000
25%	64.825000
50%	66.350000
75%	67.775000
max	70.400000

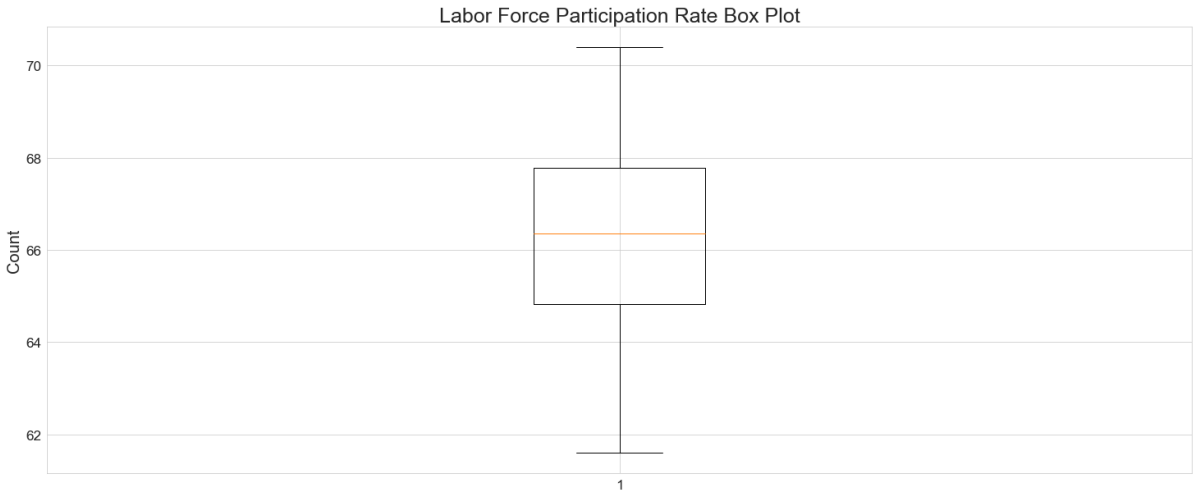
The labor force participation rate doesn't have a huge amount of variation. The lowest value comes in at 61.6 while the highest is 70.4.

In [8]:
```
# Histogram - labor force participation rate
plt.rcParams['figure.figsize'] = (25, 10)
plt.hist(data['labor force participation rate'], bins=15)
plt.title('Labor Force Participation Rate Histogram', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



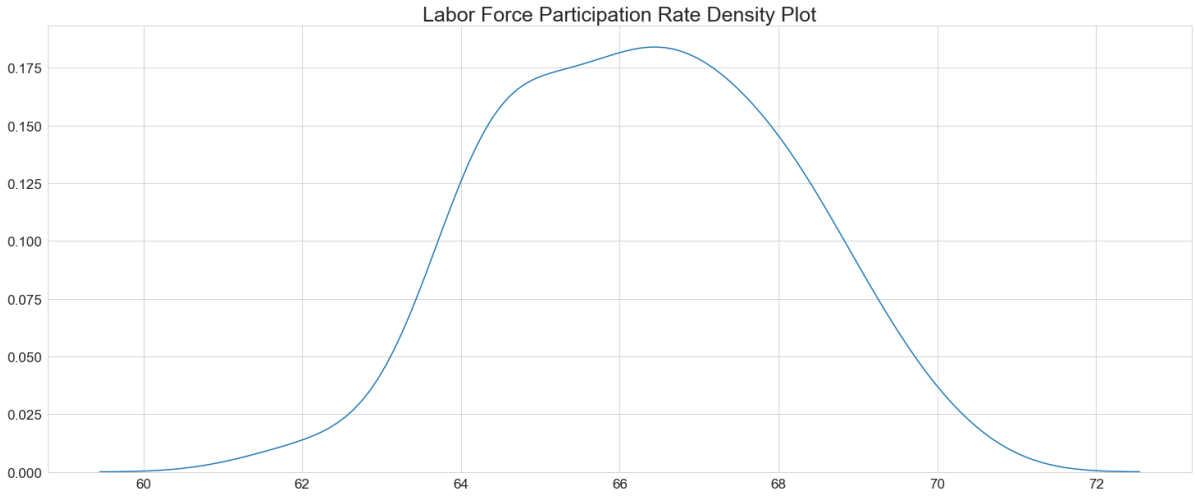
This is a somewhat normal distribution, there appear to be more data points with a higher rate than a lower one excluding around 64-64.5.

In [9]:
```
# Box plot - labor force participation rate
plt.rcParams['figure.figsize'] = (25, 10)
plt.boxplot(data['labor force participation rate'])
plt.title('Labor Force Participation Rate Box Plot', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



There aren't any data points that appear to be outliers for the labor force participation rate.

In [10]:
```
# Density Plot - labor force participation rate
plt.rcParams['figure.figsize'] = (25, 10)
plt.title('Labor Force Participation Rate Density Plot', fontsize=25)
plt.tick_params(labelsize=17)
sns.set_style('whitegrid')
sns.kdeplot(np.array(data['labor force participation rate']))
plt.show()
```



The density plot shows a more normal distribution than the histogram did, this tracks with the other statisics displayed previously.

Employment-Population Ratio

In [11]:
```
# Summary statistics - employment-population ratio
print('Employment-Population Ratio Summary Statistics')
data[["employment-population ratio"]].describe()
```

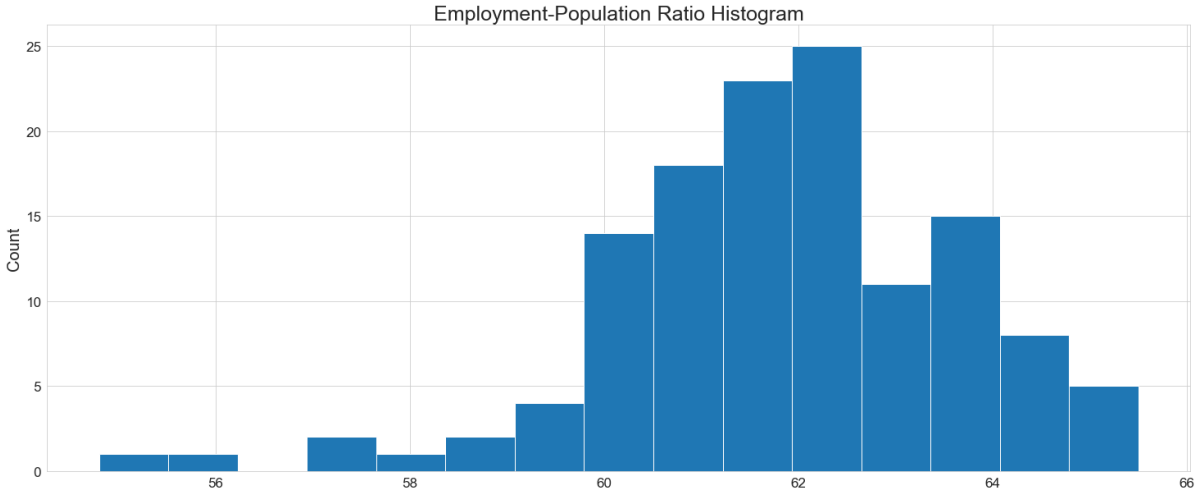
```
Employment-Population Ratio Summary Statistics
```

Out[11]:

	employment-population ratio
count	130.000000
mean	61.851538
std	1.778783
min	54.800000
25%	61.000000
50%	61.900000
75%	62.975000
max	65.500000

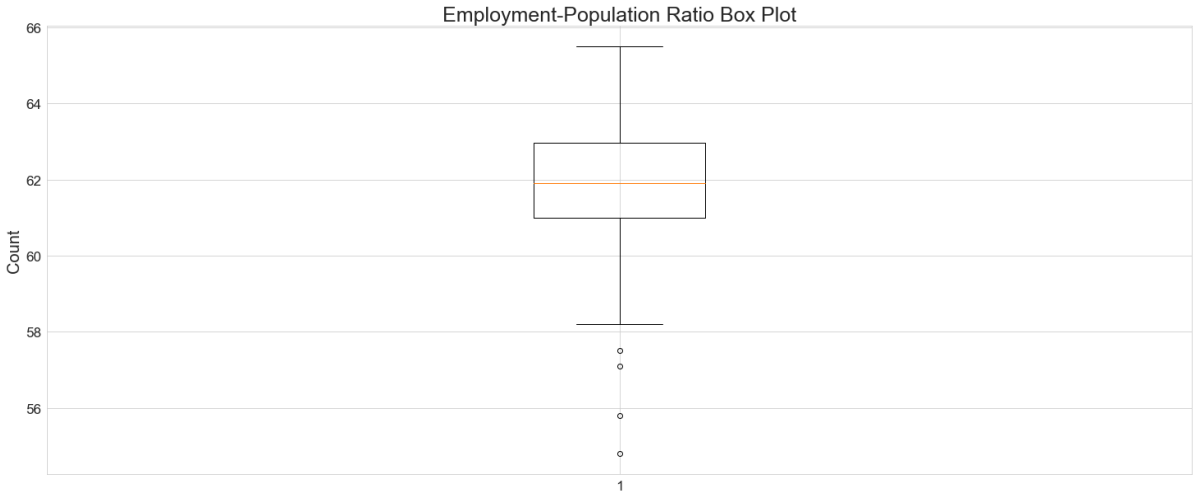
The employment-population ratio ranges from 54.8 to 65.5 so there's also not too much variation in this variable.

In [12]:
```
# Histogram - employment-population ratio
plt.rcParams['figure.figsize'] = (25, 10)
plt.hist(data['employment-population ratio'], bins=15)
plt.title('Employment-Population Ratio Histogram', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



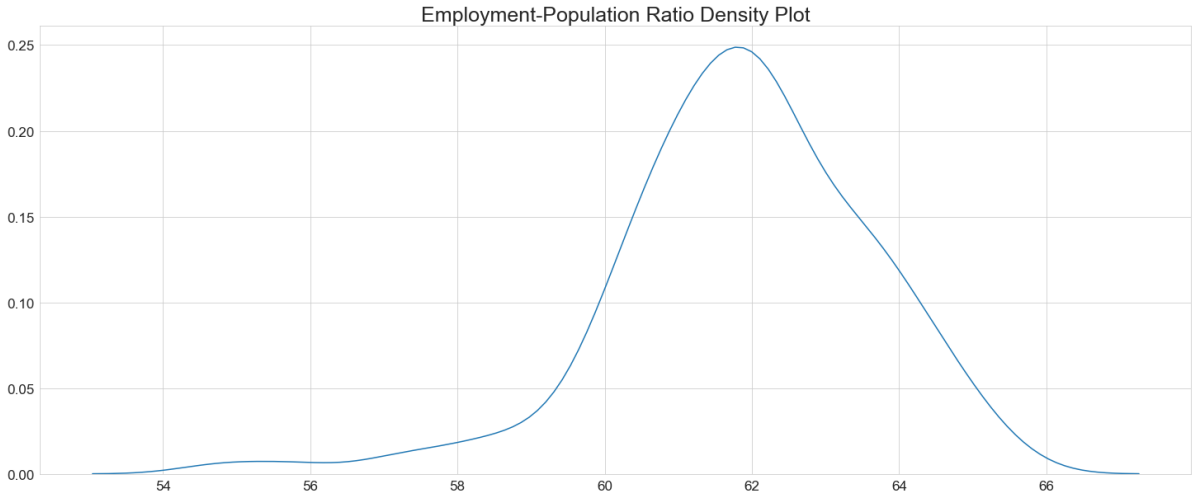
This variable has a slight left skew.

In [13]:
```
# Box plot - employment-population ratio
plt.rcParams['figure.figsize'] = (25, 10)
plt.boxplot(data['employment-population ratio'])
plt.title('Employment-Population Ratio Box Plot', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



The employment-population ratio has some observations that appear as outliers which are the lower ratios. These can be seen in the histogram as well because the lower end has fewer observations.

In [14]:
```
# Density Plot - employment-population ratio
plt.rcParams['figure.figsize'] = (25, 10)
plt.title('Employment-Population Ratio Density Plot', fontsize=25)
plt.tick_params(labelsize=17)
sns.set_style('whitegrid')
sns.kdeplot(np.array(data['employment-population ratio']))
plt.show()
```



This density plot doesn't anything different than the previous histogram for employment-population ratio.

Labor Force

In [15]:
```
# Summary statistics - labor force
print('Labor Force Summary Statistics')
data[["labor force"]].describe()
```

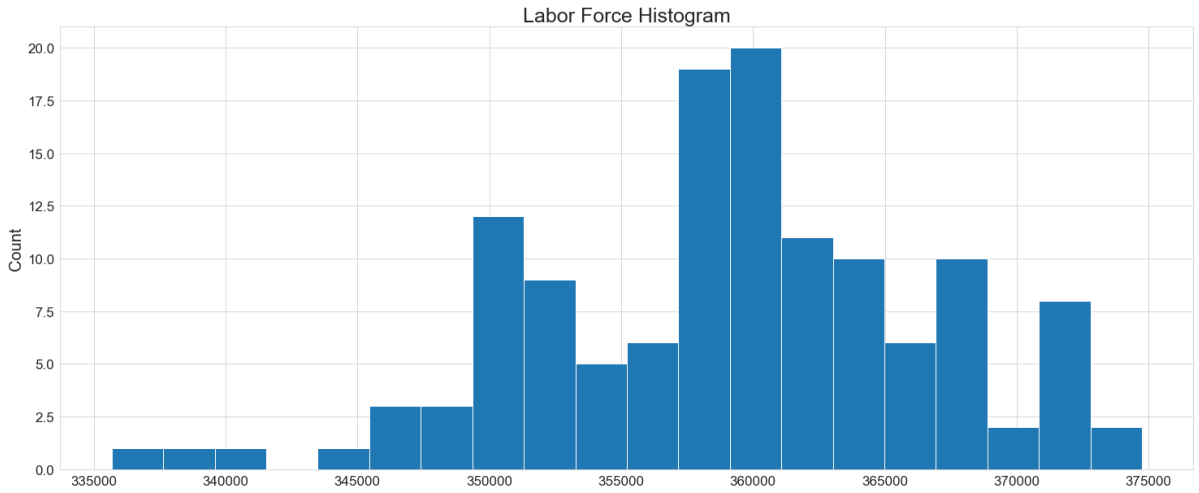
```
Labor Force Summary Statistics
```

Out[15]:

	labor force
count	130.000000
mean	359248.507692
std	7405.853566
min	335688.000000
25%	354197.250000
50%	359492.000000
75%	364107.250000
max	374747.000000

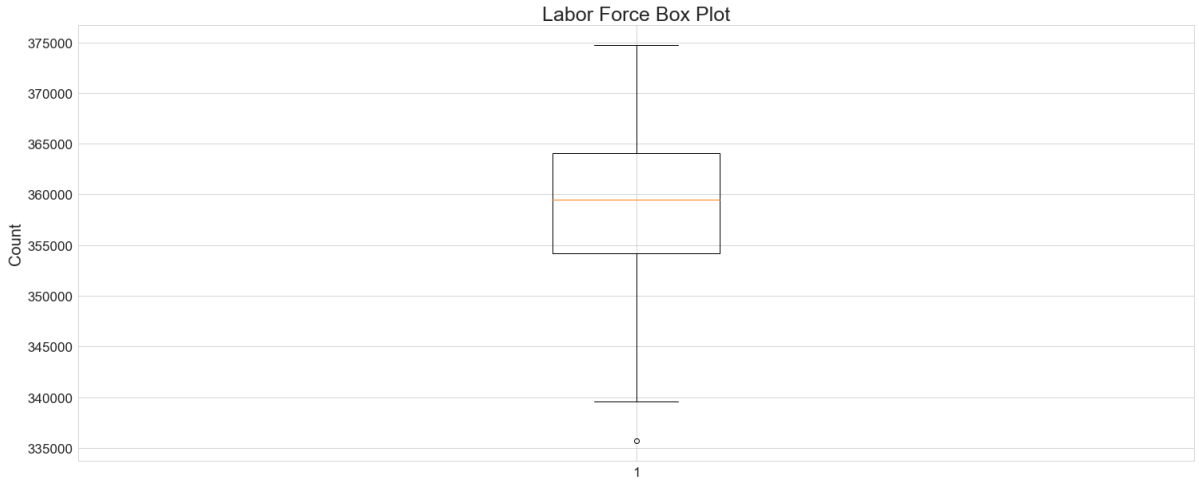
The range for the labor force numbers seems to be a bit higher than previous variables.

In [16]:
```
# Histogram - labor force
plt.rcParams['figure.figsize'] = (25, 10)
plt.hist(data['labor force'], bins=20)
plt.title('Labor Force Histogram', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



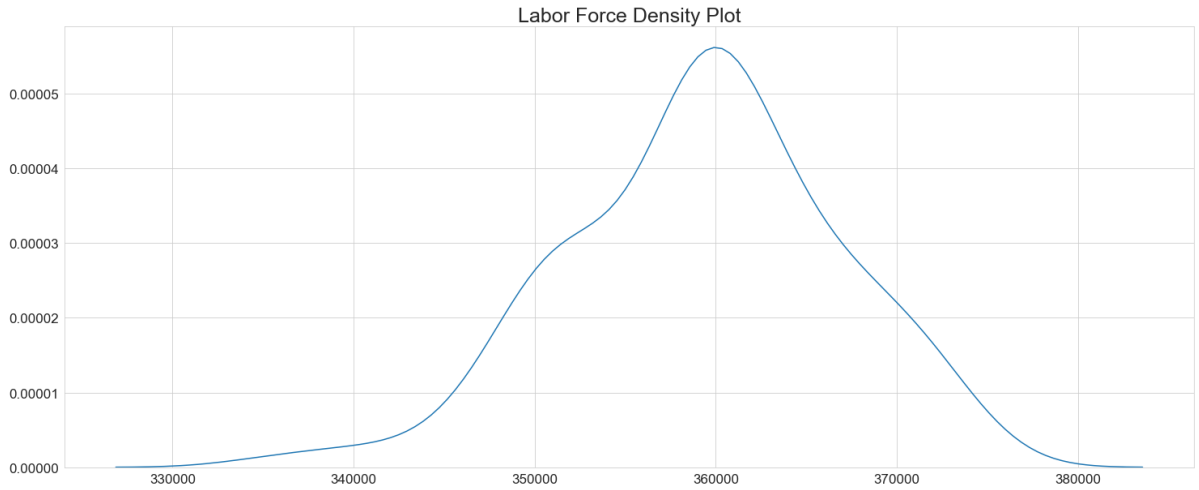
This histogram shows a fairly normal distribution except for the outliers on the lower end of labor force.

In [17]:
```
# Box plot - labor force
plt.rcParams['figure.figsize'] = (25, 10)
plt.boxplot(data['labor force'])
plt.title('Labor Force Box Plot', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



The boxplot shows the same outlier we saw in the histogram.

In [18]:
```
# Density Plot - Labor force
plt.rcParams['figure.figsize'] = (25, 10)
plt.title('Labor Force Density Plot', fontsize=25)
plt.tick_params(labelsize=17)
sns.set_style('whitegrid')
sns.kdeplot(np.array(data['labor force']))
plt.show()
```



The density plot again shows how close the labor force is to a normal distribution.

Employment

In [19]:
```
# Summary statistics - employment
print('Employment Summary Statistics')
data[["employment"]].describe()
```

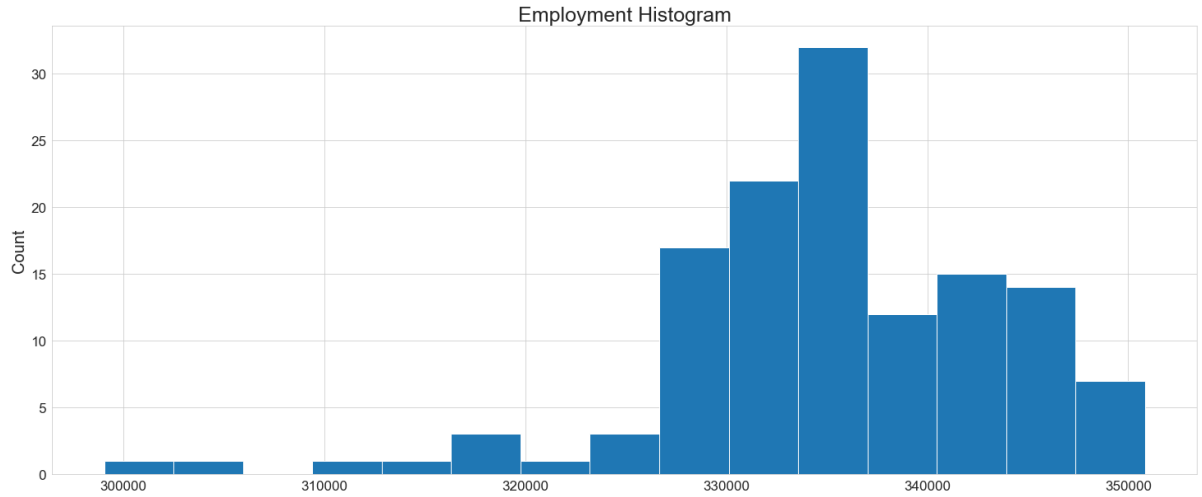
```
Employment Summary Statistics
```

Out[19]:

	employment
count	130.000000
mean	335191.115385
std	8451.357241
min	299048.000000
25%	331401.000000
50%	334871.000000
75%	340839.750000
max	350829.000000

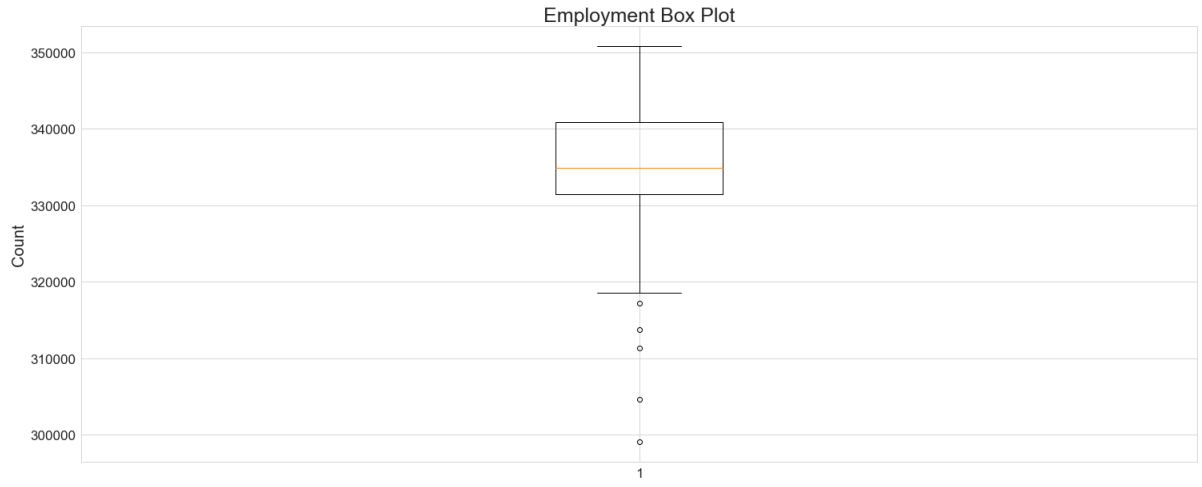
The employment statisitcs have 299,048 as the minimum value and the next 25% quartile value jumps up to 331,401. This variable is going to have some outliers on the lower end of employment numbers.

In [20]:
```
# Histogram - employment
plt.rcParams['figure.figsize'] = (25, 10)
plt.hist(data['employment'], bins=15)
plt.title('Employment Histogram', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



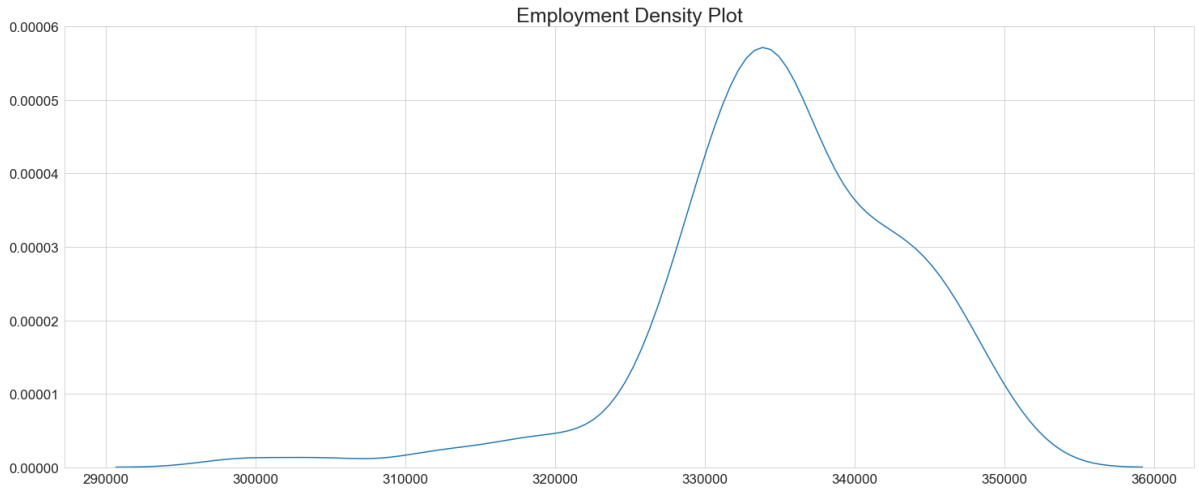
As expected, there is a left skew to employment with some outliers.

In [21]:
```
# Box plot - employment
plt.rcParams['figure.figsize'] = (25, 10)
plt.boxplot(data['employment'])
plt.title('Employment Box Plot', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



The box plot shows multiple outliers within unemployment, indiciating it has not been common for Alaska to have low employment numbers since 2011.

In [22]:
```
# Density Plot - employment
plt.rcParams['figure.figsize'] = (25, 10)
plt.title('Employment Density Plot', fontsize=25)
plt.tick_params(labelsize=17)
sns.set_style('whitegrid')
sns.kdeplot(np.array(data['employment']))
plt.show()
```



The density plot makes the left skew more evidant than the histogram did.

Unemployment

In [23]:
```
# Summary statistics - unemployment
print('Unemployment Summary Statistics')
data[["unemployment"]].describe()
```

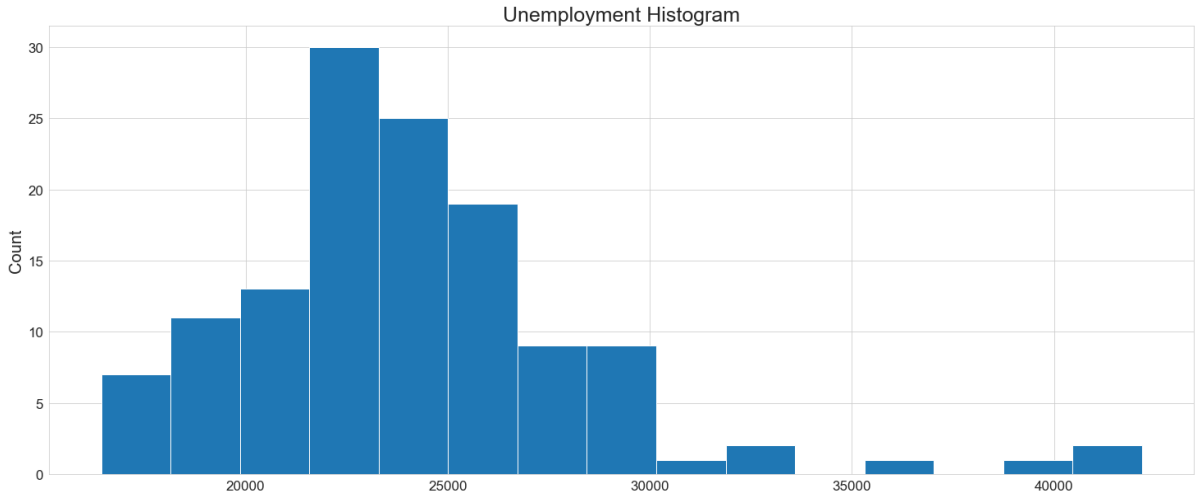
```
Unemployment Summary Statistics
```

Out[23]:

	unemployment
count	130.000000
mean	24057.392308
std	4314.453032
min	16424.000000
25%	21714.750000
50%	23568.500000
75%	25592.250000
max	42175.000000

Unemployment appears to be the opposite of the employment variable. The max value is much higher than the other values.

In [24]:
```
# Histogram - unemployment
plt.rcParams['figure.figsize'] = (25, 10)
plt.hist(data['unemployment'], bins=15)
plt.title('Unemployment Histogram', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```

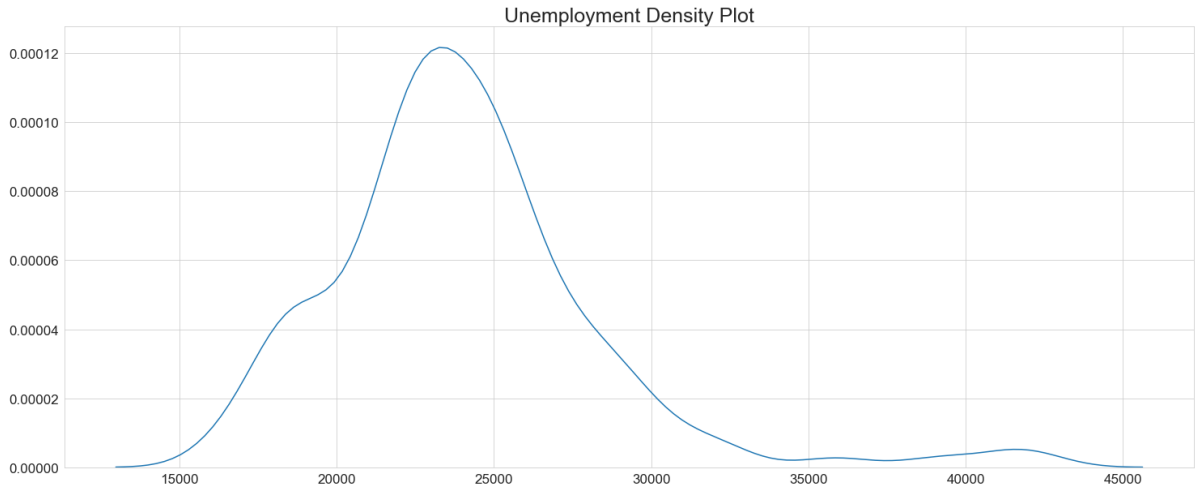


This varaible has a right skew whichis unlike many of the other varibles, there are also multiple points that appear to be outliers.

In [25]:
```
# Box plot - unemployment
plt.rcParams['figure.figsize'] = (25, 10)
plt.boxplot(data['unemployment'])
plt.title('Unemployment Box Plot', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```

The box plot shows all the outliers for the higher values of unemployment.

In [26]:
```
# Density Plot - unemployment
plt.rcParams['figure.figsize'] = (25, 10)
plt.title('Unemployment Density Plot', fontsize=25)
plt.tick_params(labelsize=17)
sns.set_style('whitegrid')
sns.kdeplot(np.array(data['unemployment']))
plt.show()
```



The density plot for unemployment shows the same outliers that have been observed in the previous visuals.

Unemployment Rate

In [27]:
```
# Summary statistics - unemployment rate
print('Unemployment Rate Summary Statistics')
data[["unemployment rate"]].describe()
```

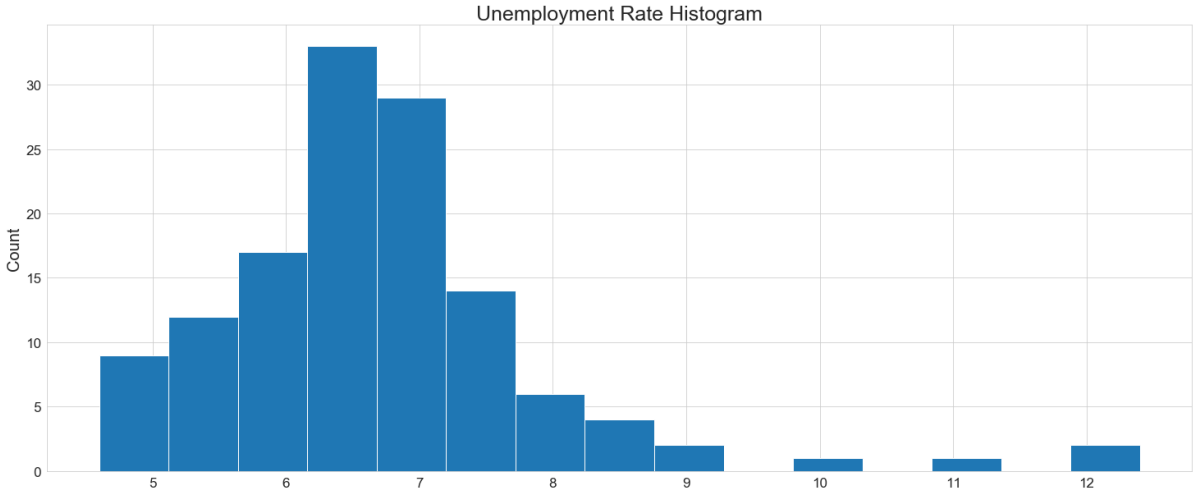
```
Unemployment Rate Summary Statistics
```

Out[27]:

	unemployment rate
count	130.000000
mean	6.698462
std	1.235206
min	4.600000
25%	6.000000
50%	6.500000
75%	7.100000
max	12.400000

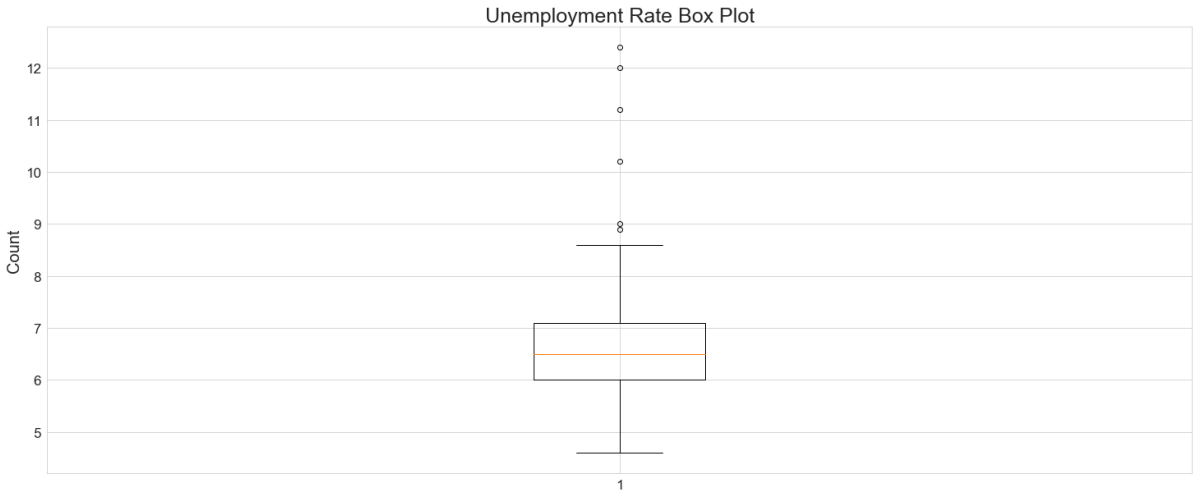
The unemployment rate ranges from 4.6 to 12.4, this is a slightly larger range than other rate variables have had.

In [28]:
```
# Histogram - unemployment rate
plt.rcParams['figure.figsize'] = (25, 10)
plt.hist(data['unemployment rate'], bins=15)
plt.title('Unemployment Rate Histogram', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



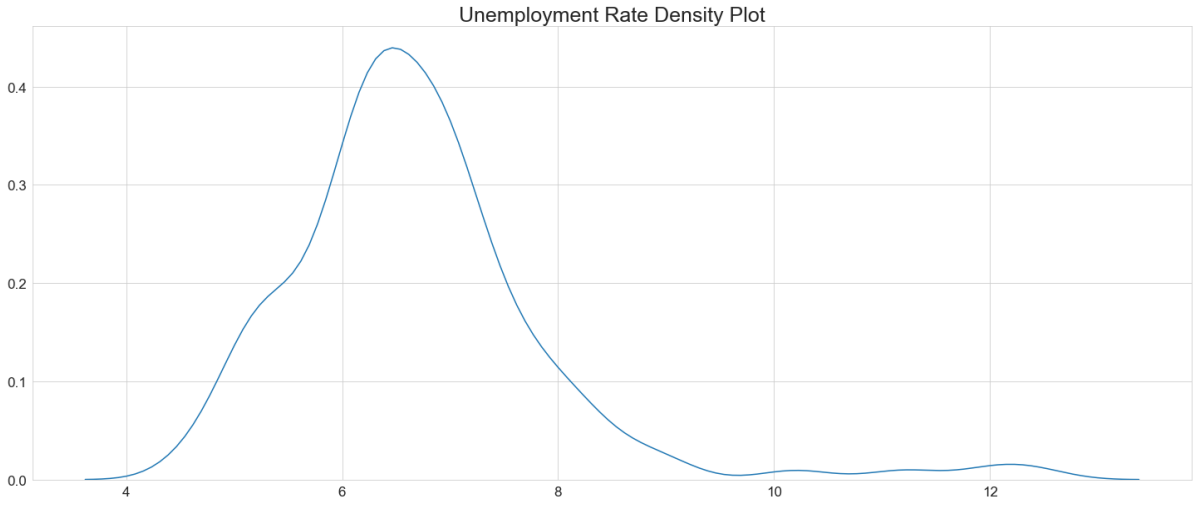
The histogram shows a right skew in the data with a few outliers on the higher end of unemployment rates.

In [29]:
```
# Box plot - unemployment rate
plt.rcParams['figure.figsize'] = (25, 10)
plt.boxplot(data['unemployment rate'])
plt.title('Unemployment Rate Box Plot', fontsize=25)
plt.ylabel('Count', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



The above box plot breaks down the outliers more so showing that values from around 9 to around 12 look like outliers.

In [30]:
```
# Density Plot - unemployment rate
plt.rcParams['figure.figsize'] = (25, 10)
plt.title('Unemployment Rate Density Plot', fontsize=25)
plt.tick_params(labelsize=17)
sns.set_style('whitegrid')
sns.kdeplot(np.array(data['unemployment rate']))
plt.show()
```



The density plot shows what has been seen with the previous plots.

Data Structure and Types

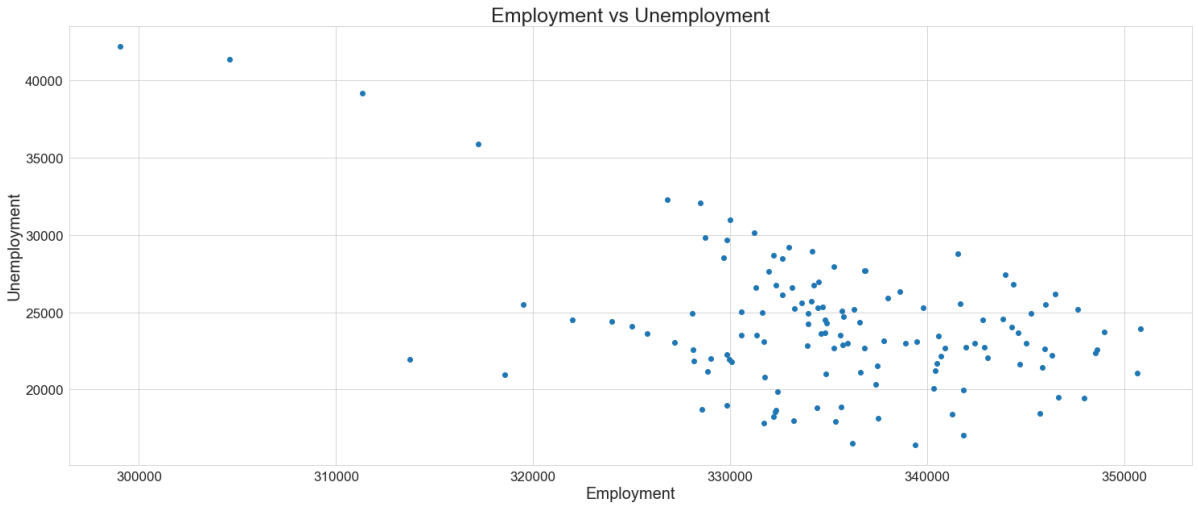
In [31]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 130 entries, 0 to 129
Data columns (total 9 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Year                                  130 non-null    int64
 1   Period                                130 non-null    object
 2   civilian noninstitutional population  130 non-null    int64
 3   labor force participation rate        130 non-null    float64
 4   employment-population ratio           130 non-null    float64
 5   labor force                           130 non-null    int64
 6   employment                            130 non-null    int64
 7   unemployment                          130 non-null    int64
 8   unemployment rate                     130 non-null    float64
dtypes: float64(3), int64(5), object(1)
memory usage: 9.3+ KB
```

Here we can see that the Alaska BLS data is structured as a pandas dataframe so it is in a tabular format. Additionally, the Period variable is the only object, all other variables are integers or floats. The float variables represent percentages and ratios while the integers represent whole number counts.

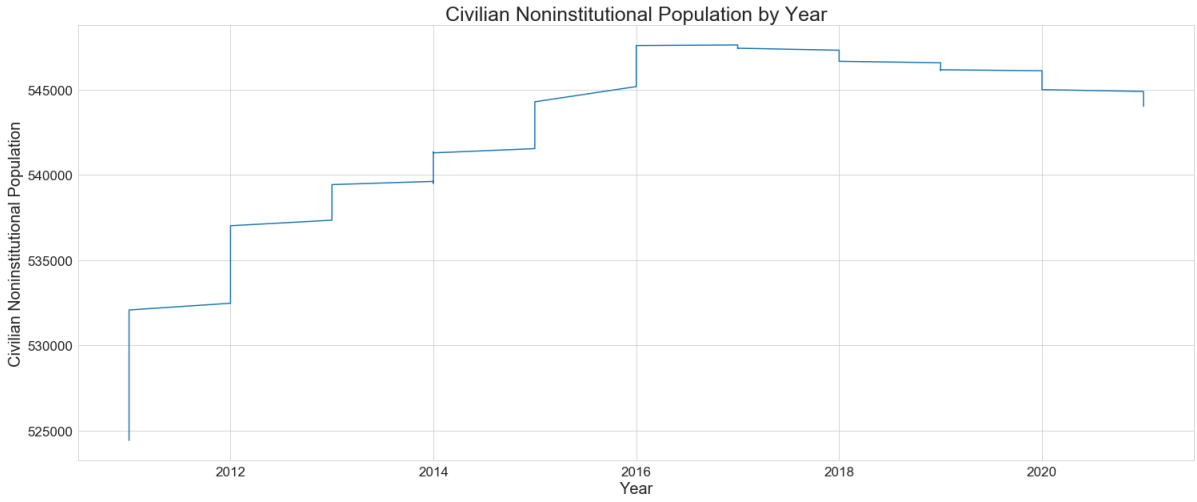
Results

In [32]:
```
# Bivariate plot - employment vs unemployment
plt.rcParams['figure.figsize'] = (25, 10)
plt.scatter(data['employment'], data['unemployment'])
plt.title('Employment vs Unemployment', fontsize=25)
plt.xlabel('Employment', fontsize=20)
plt.ylabel('Unemployment', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



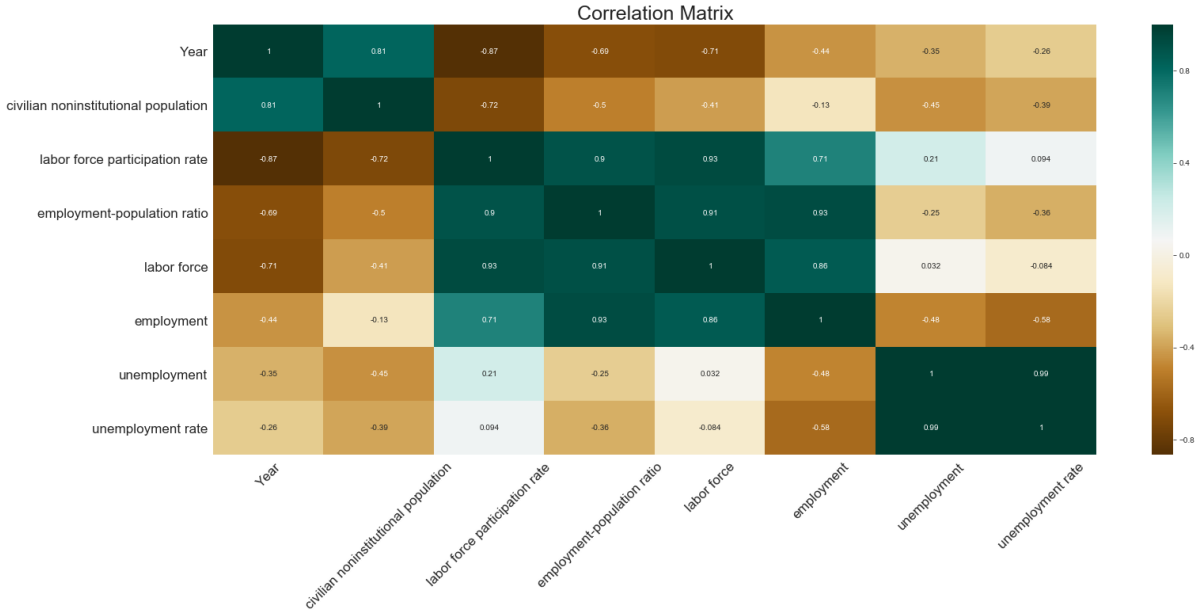
Looking at employment vs unemployment we can see that as the number of employed people in Alaska increase the unemployment numbers generally decrease. There is still some variation though which could potentially indicate that there are people moving in and out of the state causing the total population numbers to flucuate.

In [33]:
```
# Bivariate plot - civilian noninstitutional population by year
plt.rcParams['figure.figsize'] = (25, 10)
plt.plot(np.array(data['Year']), np.array(data['civilian noninstitutional popu
lation']))
plt.title('Civilian Noninstitutional Population by Year', fontsize=25)
plt.xlabel('Year', fontsize=20)
plt.ylabel('Civilian Noninstitutional Population', fontsize=20)
plt.tick_params(labelsize=17)
plt.show()
```



The civilian noninstitutional population is increasing from 2011 to 2016. After 2016 the numbers start to slightly drop off. There could be a number of reasons as to why the population is going down.

In [34]:
```
# Correlation matrix
c = data.corr()
plt.figure(figsize=(25,10))
sns.heatmap(c,cmap='BrBG',annot=True)
plt.title('Correlation Matrix', fontsize=25)
plt.xticks(rotation=45)
plt.tick_params(labelsize=17)
plt.show()
```



The correlation matrix contains a lot of information. There are a few strong relationships between the variables in this data. The labor force population rate has a strong positive correlation with both the labor force and employment-population rate. It also has a fairly strong positive relationship with employment. This indicates that as the participation increases those rates and counts increase. That explains the obvious in a way though because if you have higher employment numbers then the labor force and employment ratios are going to increase. Similarily, unemployment is highly positivly correlated with the unemployment rate, each of those variables are measuring unemployment in very similar ways. There is also an interesting negative relationship in the data. Civilian noninstitutional population has a fairly strong negative relationship with the labor force participation rate. The more people that are 16 years or older in Alaska the lower the labor force participation.

In [35]:
```
# Fix month name sort
month_sort = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep',
'Oct', 'Nov', 'Dec']
data['Period'] = pd.CategoricalIndex(data['Period'], categories=month_sort, or
dered=True)
```

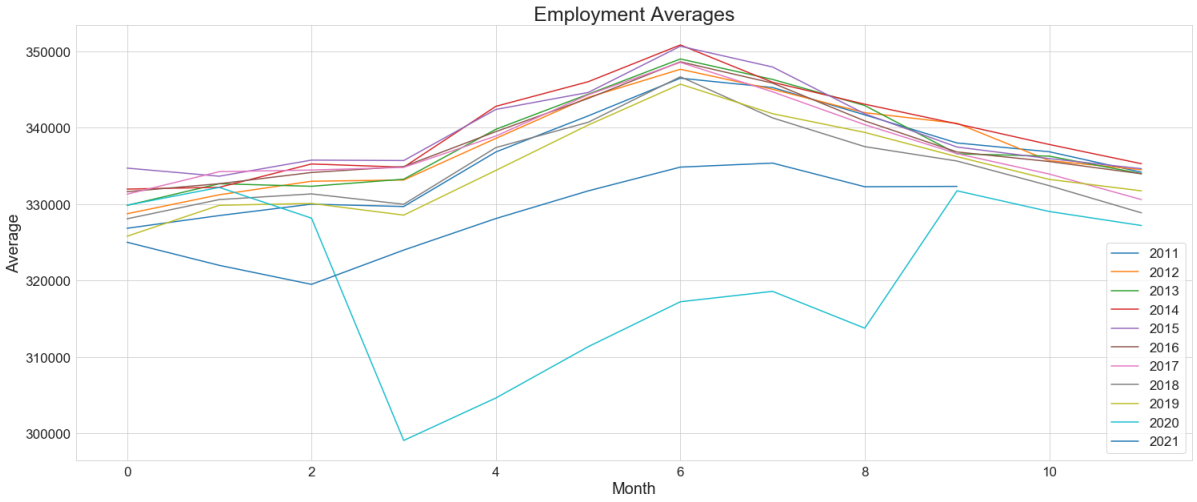
In [36]:
```
# Cross table - employment
print('Average employment per month and year:')
employment_avg = pd.crosstab(data['Period'], data['Year'], values=data['employ
ment'], aggfunc='mean').round(0)
employment_avg
```

```
Average employment per month and year:
```

Out[36]:

Year	2011	2012	2013	2014	2015	2016	2017	2018	201
Period									
Jan	326825.0	328726.0	329822.0	331954.0	334704.0	331617.0	331289.0	328067.0	325783.
Feb	328492.0	331215.0	332657.0	332193.0	333642.0	332663.0	334240.0	330584.0	329835.
Mar	329986.0	332985.0	332324.0	335244.0	335754.0	334130.0	334454.0	331329.0	330081.
Apr	329674.0	333148.0	333255.0	334826.0	335693.0	334907.0	334797.0	329972.0	328549.
May	336821.0	338599.0	339791.0	342795.0	342386.0	339484.0	338905.0	337393.0	334407.
Jun	341539.0	343958.0	344375.0	346012.0	344607.0	343829.0	344303.0	340704.0	340339.
Jul	346476.0	347646.0	348993.0	350829.0	350664.0	348623.0	348549.0	346662.0	345694.
Aug	345254.0	345026.0	346314.0	345941.0	347941.0	345836.0	344693.0	341279.0	341827.
Sep	341676.0	341962.0	342897.0	343070.0	341819.0	340885.0	340391.0	337515.0	339376.
Oct	337996.0	340572.0	336580.0	340499.0	337462.0	336819.0	336605.0	335632.0	336203.
Nov	336846.0	335681.0	336269.0	337800.0	335950.0	335568.0	333926.0	332406.0	333227.
Dec	334163.0	334494.0	333968.0	335280.0	334627.0	333960.0	330584.0	328847.0	331726.

In [37]:
```
# Cross table plot - unemployment
plt.rcParams['figure.figsize'] = (25, 10)
plt.plot(np.array(employment_avg))
plt.title('Employment Averages', fontsize=25)
plt.xlabel('Month', fontsize=20)
plt.ylabel('Average', fontsize=20)
plt.tick_params(labelsize=17)
plt.legend(employment_avg, fontsize=17)
plt.show()
```



Employment in Alaska seems to follow a similar trend for most years. The table shows the exact numbers but the graph makes it much more clear, 2020 and 2021 have by far the lowest employment in Alaska since 2011, likely due to the Covid-19 pandemic. It does appear that 2021 might make it back up towards normal numbers by the end of the year but that data is currently unknown. This visual also shows that June tends to be when employment is at it's highest for each year in Alaska. It would be interesting to get more labor data to see why this is.

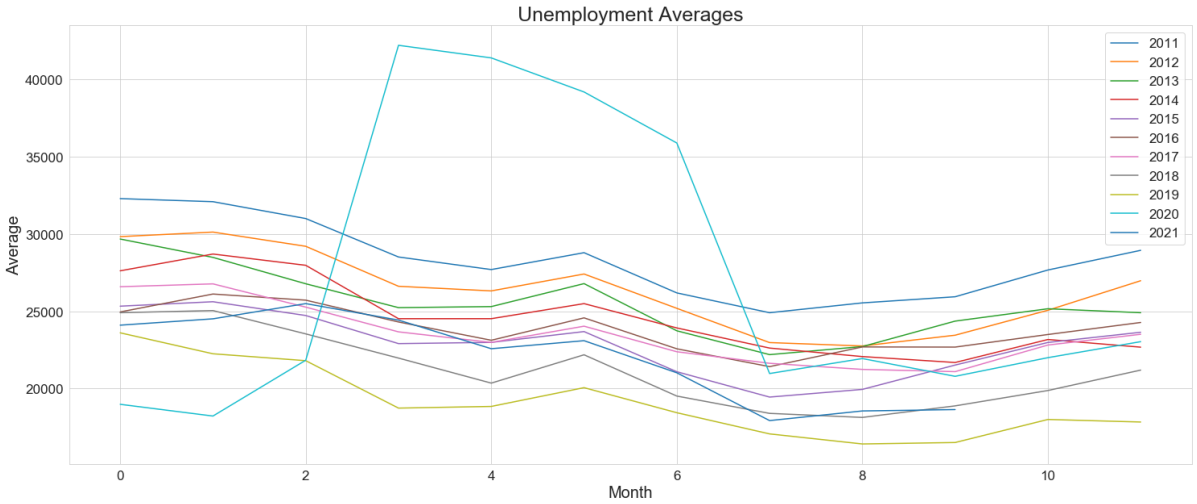
In [38]:
```
# Cross table - unemployment
print('Average unemployment per month and year:')
unemployment_avg = pd.crosstab(data['Period'], data['Year'], values=data['unemployment'], aggfunc='mean').round(0)
unemployment_avg
```

```
Average unemployment per month and year:
```

Out[38]:

Year	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
Period										
Jan	32272.0	29816.0	29662.0	27608.0	25324.0	24951.0	26580.0	24903.0	23603.0	18985.0
Feb	32073.0	30110.0	28475.0	28692.0	25611.0	26108.0	26765.0	25040.0	22249.0	18227.0
Mar	30985.0	29192.0	26772.0	27964.0	24721.0	25713.0	25263.0	23534.0	21804.0	21832.0
Apr	28501.0	26609.0	25229.0	24515.0	22906.0	24303.0	23665.0	21971.0	18740.0	42175.0
May	27689.0	26310.0	25297.0	24516.0	22984.0	23122.0	22984.0	20354.0	18848.0	41361.0
Jun	28779.0	27402.0	26781.0	25489.0	23683.0	24569.0	24031.0	22184.0	20060.0	39160.0
Jul	26181.0	25185.0	23734.0	23918.0	21089.0	22571.0	22380.0	19518.0	18442.0	35864.0
Aug	24901.0	22972.0	22194.0	22618.0	19452.0	21419.0	21634.0	18399.0	17074.0	20973.0
Sep	25536.0	22753.0	22714.0	22069.0	19948.0	22690.0	21240.0	18138.0	16424.0	21941.0
Oct	25932.0	23449.0	24359.0	21685.0	21520.0	22681.0	21095.0	18884.0	16517.0	20793.0
Nov	27660.0	25060.0	25158.0	23171.0	22980.0	23494.0	22814.0	19876.0	18006.0	21998.0
Dec	28931.0	26968.0	24902.0	22676.0	23639.0	24271.0	23507.0	21195.0	17841.0	23032.0

In [39]:
```
# Cross table plot - unemployment
plt.rcParams['figure.figsize'] = (25, 10)
plt.plot(np.array(unemployment_avg))
plt.title('Unemployment Averages', fontsize=25)
plt.xlabel('Month', fontsize=20)
plt.ylabel('Average', fontsize=20)
plt.tick_params(labelsize=17)
plt.legend(unemployment_avg, fontsize=17)
plt.show()
```



The above unemployment table and visual show similar results as the previos employment analysis. The year 2020 clearly has the highest unemployment rates, likely due to the pandemic as I mentioned previously. It also shows that 2021 is still higher than previous years but it's much closer to them where the employment rate for 2021 was much higher than previous years. We can also see that the beginning of each year is a high point for unemployment in Alaska.

Return to Top