```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib as mpl
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: pandas.util.testing is deprecat
  import pandas.util.testing as tm
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6bn6qk8qdgf4n4g3pfee6491h

Enter your authorization code:
··········
Mounted at /content/drive
```

Strip Plot

Strip plot is a scatter plot where one of the variables is categorical.

```
# Recover default matplotlib settings
mpl.rcParams.update(mpl.rcParamsDefault)
%matplotlib inline
sns.set_style("white")
```

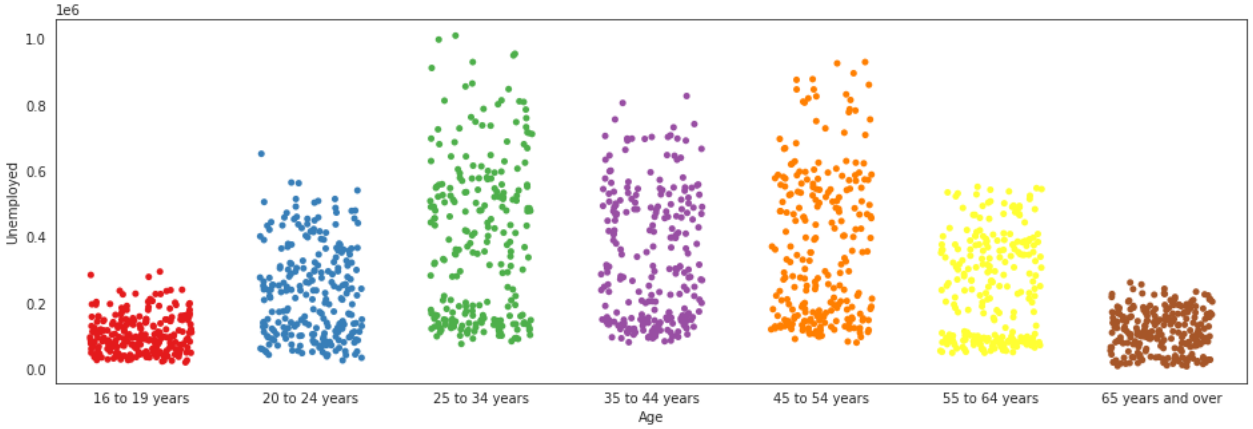
```
employment = pd.read_excel("/content/drive/My Drive/Python DataScience/Visualization/Se
employment.head()
```

	Age	Gender	Period	Unemployed
0	16 to 19 years	Men	2005-01-01	91000
1	20 to 24 years	Men	2005-01-01	175000
2	25 to 34 years	Men	2005-01-01	194000
3	35 to 44 years	Men	2005-01-01	201000
4	45 to 54 years	Men	2005-01-01	207000

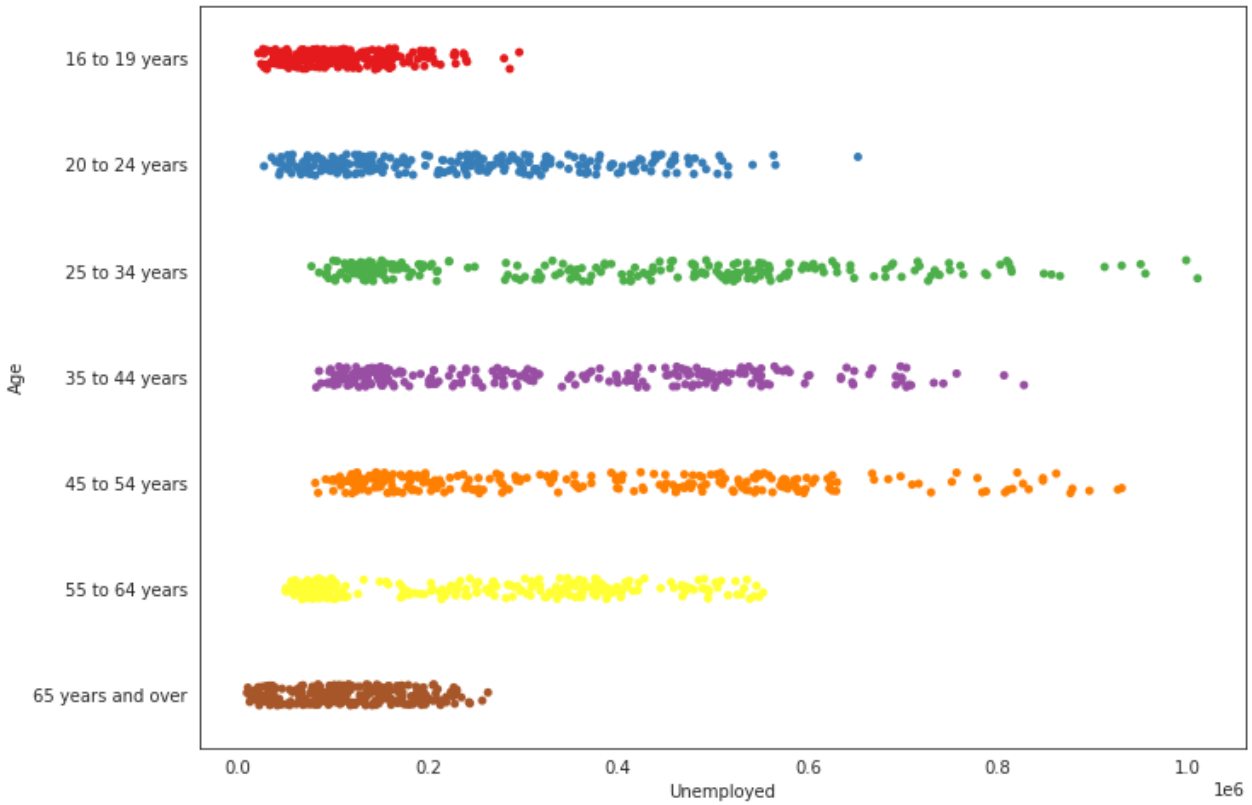
```
plt.figure(figsize=(11,8))
sns.stripplot(x=employment.Unemployed)
plt.show()
```

```
plt.figure(figsize=(16,8))
sns.stripplot(x=employment.Age ,palette="Set1", y = employment.Unemployed)
plt.show()
```

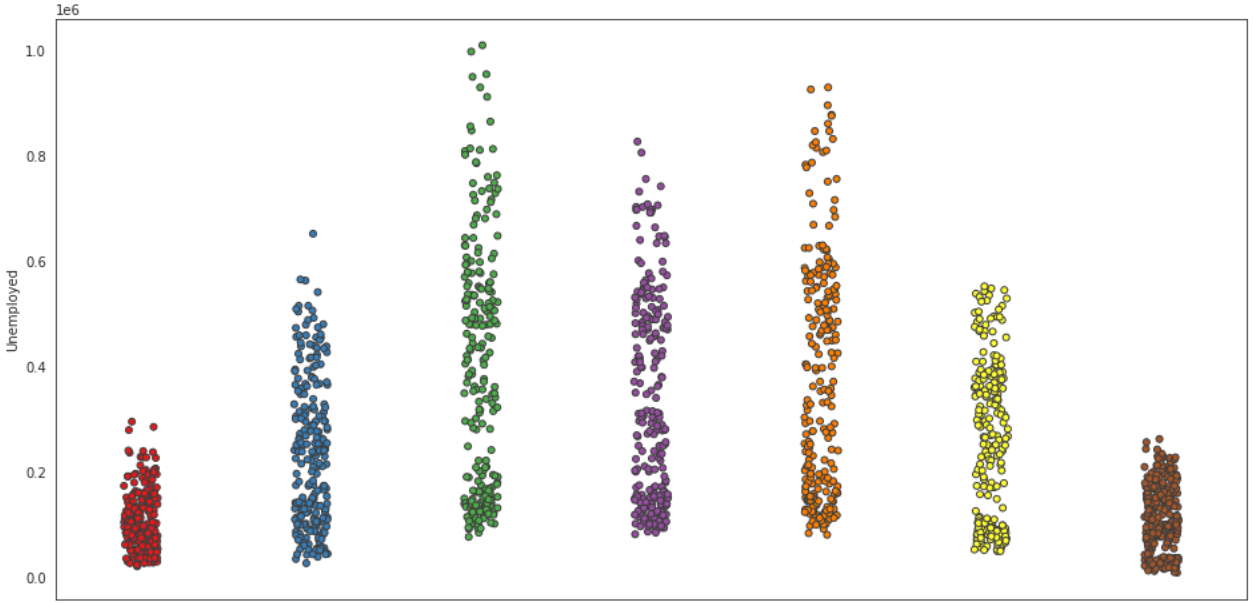
```
""" "Jitter" parameter signifies the amount of jitter to apply.
   This can be extremely useful when we have large clusters of data points"""

plt.figure(figsize=(16,5))
sns.stripplot(x=employment.Age ,palette="Set1", y = employment.Unemployed , jitter=0.3)
plt.show()
```



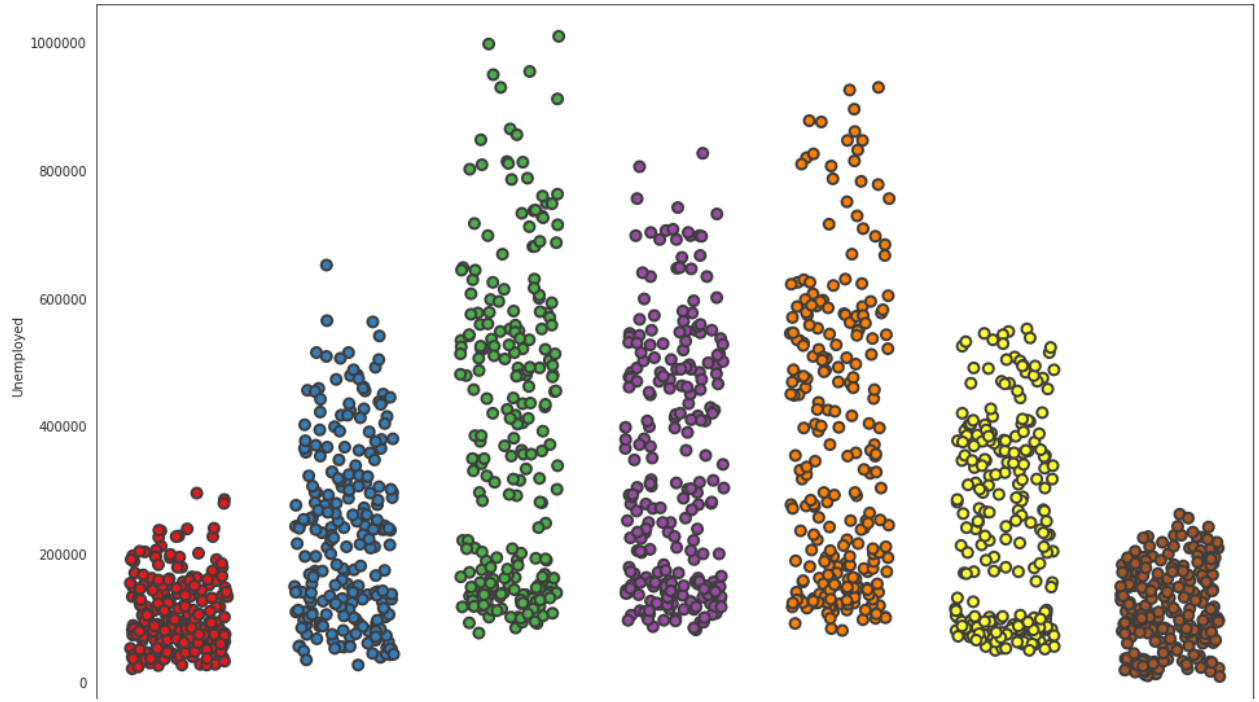

```
# Flip x and y inputs to make a horizontal strip plot
plt.figure(figsize=(11,8))
sns.stripplot(y=employment.Age ,palette="Set1", x = employment.Unemployed , jitter=True
plt.show()
```



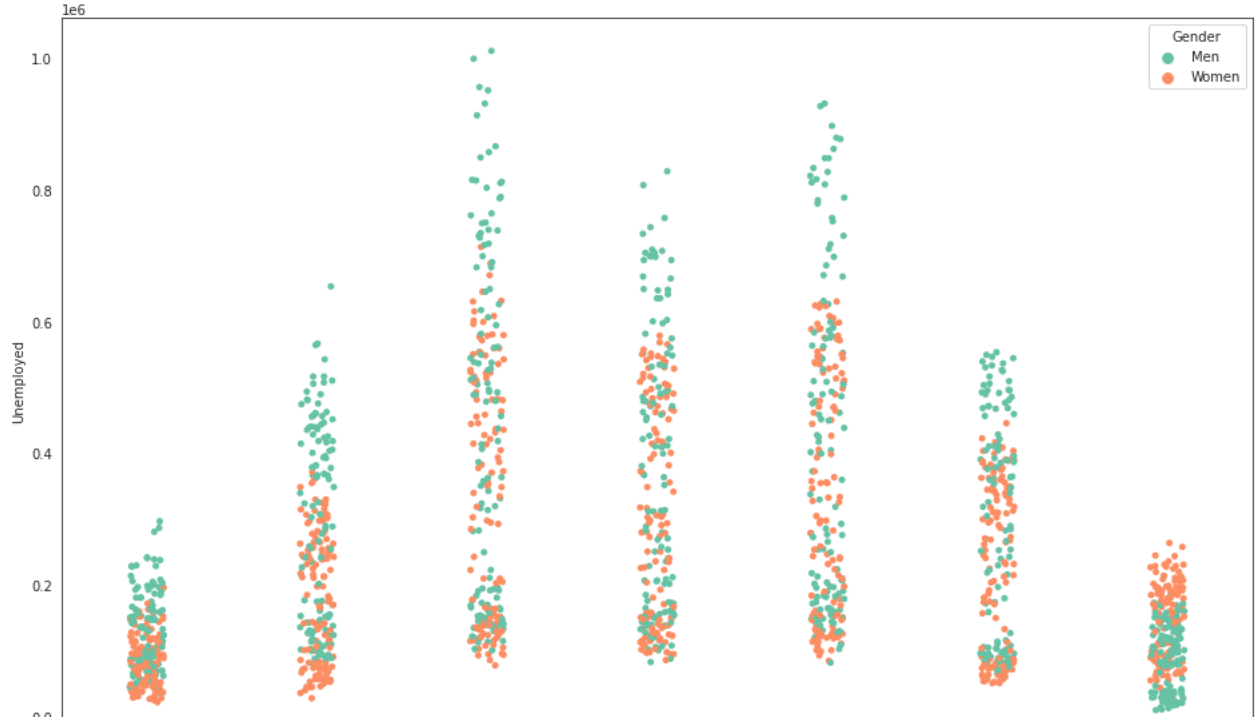

```
plt.figure(figsize=(16,8))
sns.stripplot(x=employment.Age ,palette="Set1", y = employment.Unemployed , linewidth=1
plt.show()
```



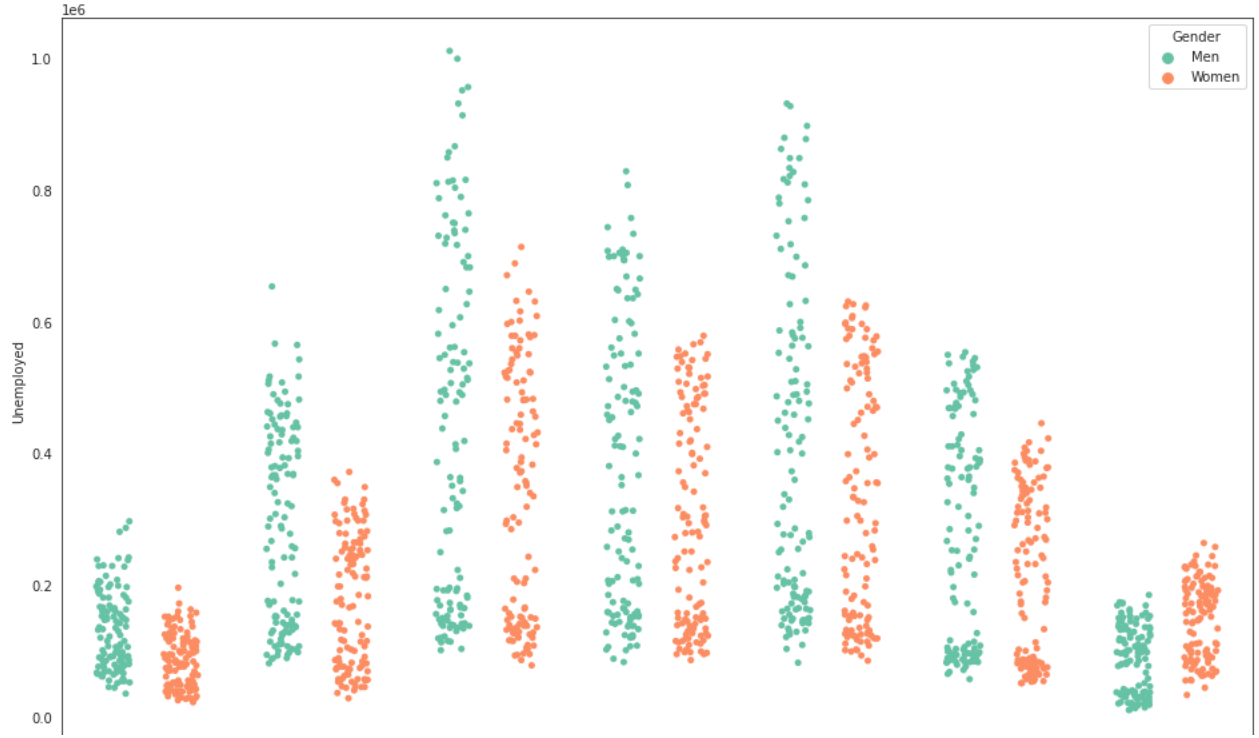

```
# Adjust the linewidth of the edges of the circles using "linewidth" parameter
# Adjust the size of the circles using the "size" parameter
plt.figure(figsize=(16,10))
ax=sns.stripplot(x=employment.Age ,palette="Set1", y = employment.Unemployed , linewidt
ax.ticklabel_format(style='plain', axis='y')
plt.show()
```



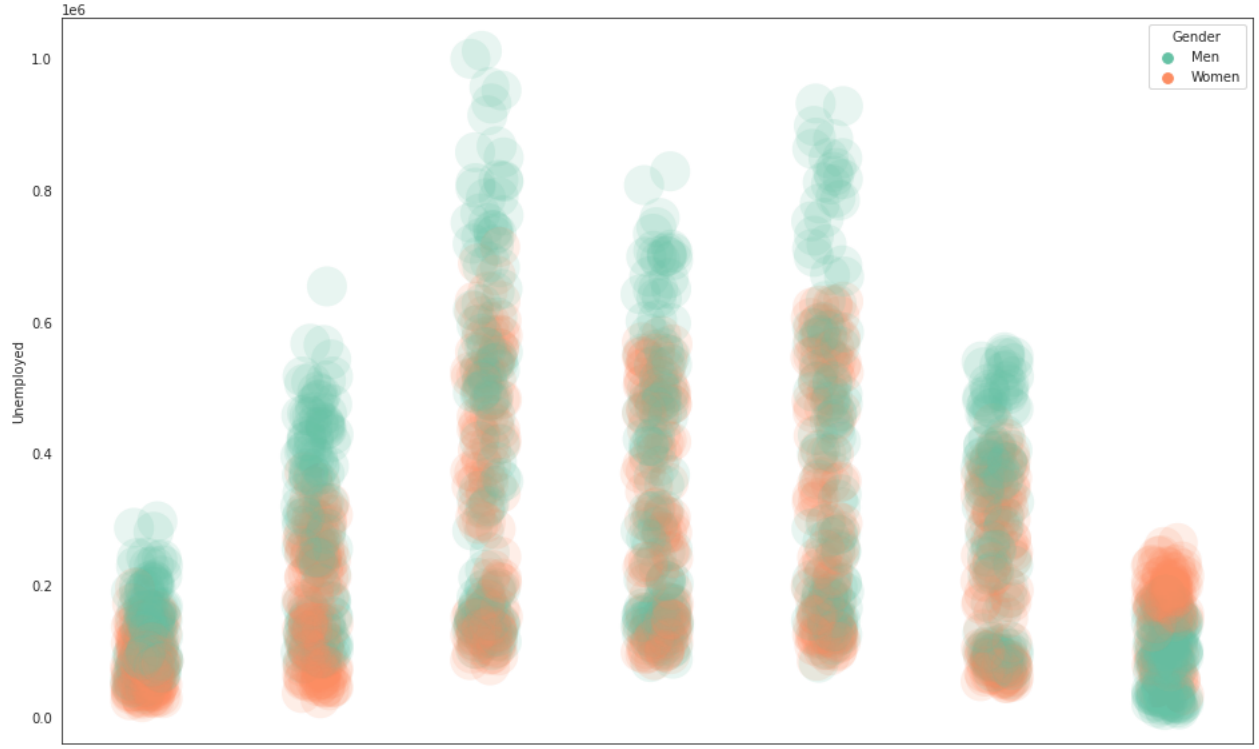

```
# Using set2 pallete
plt.figure(figsize=(16,10))
sns.stripplot(x=employment.Age ,palette="Set2", y = employment.Unemployed , hue=employm
plt.show()
```



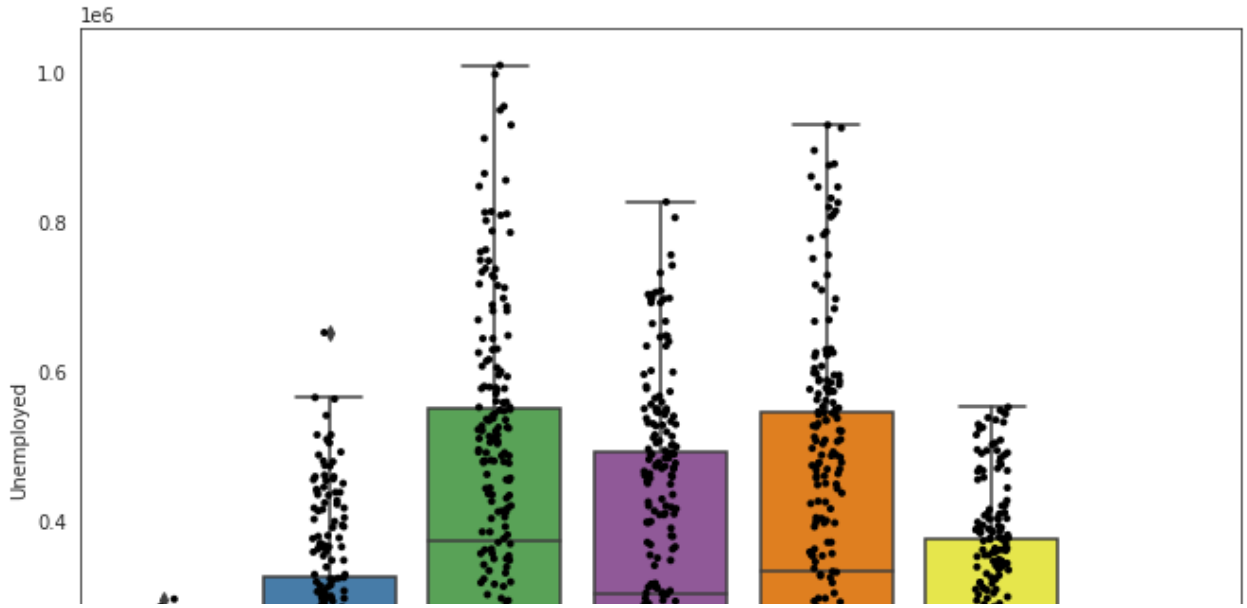

```
# Separate the strips for different hue levels along the categorical axis using "dodge=
plt.figure(figsize=(16,10))
sns.stripplot(x=employment.Age ,palette="Set2", y = employment.Unemployed , hue=employm
plt.show()
```



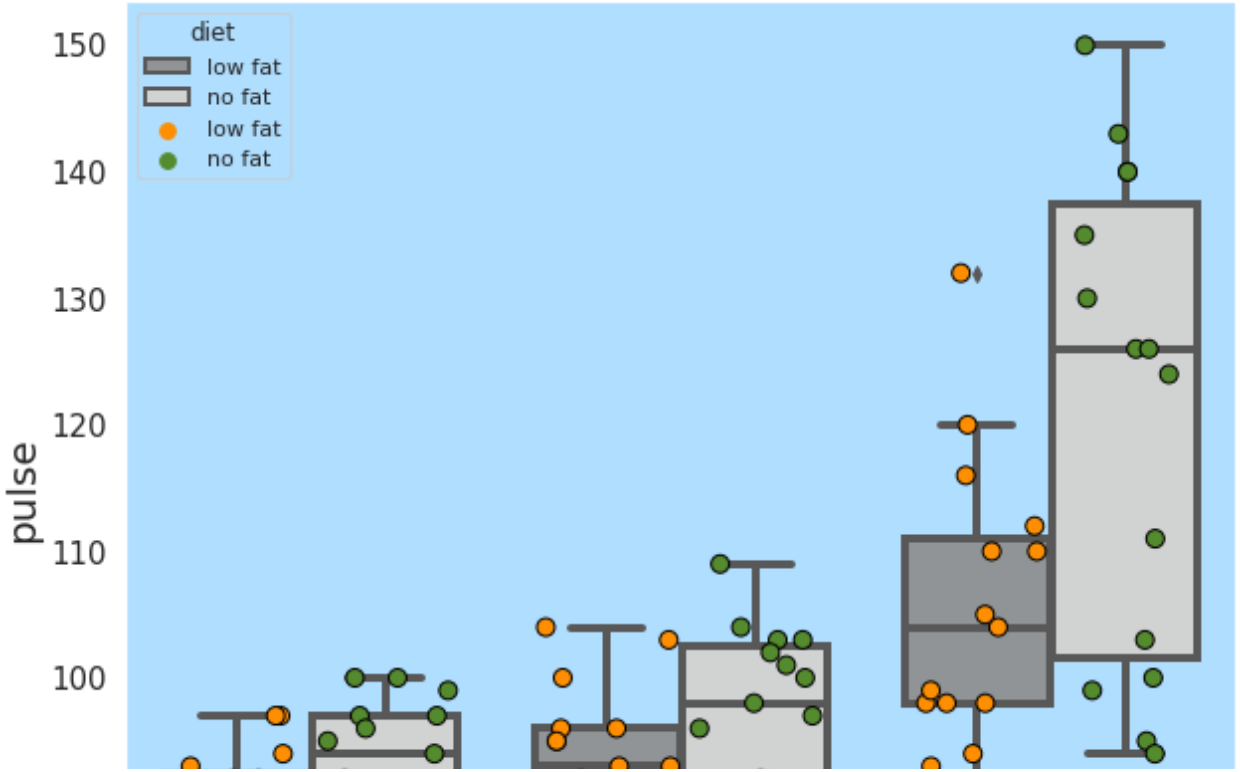

```
plt.figure(figsize=(16,10))
sns.stripplot(x=employment.Age ,palette="Set2", y = employment.Unemployed ,
              hue=employment.Gender, marker = "o" , size = 30 , alpha = .15)
plt.show()
```



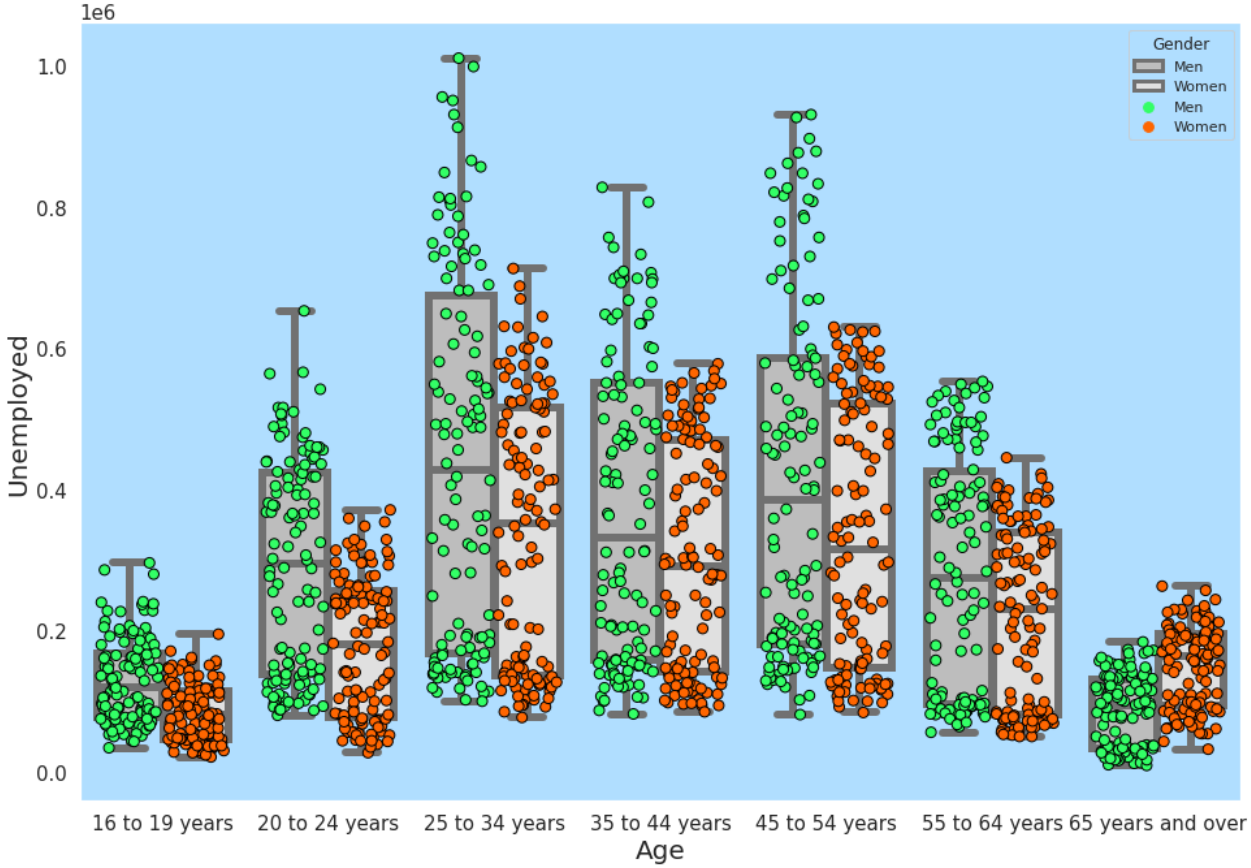

```
# Drawing stripplot on top of a box plot
plt.figure(figsize=(11,8))
sns.stripplot(x=employment.Age, y = employment.Unemployed , jitter=True , color="black"
sns.boxplot(x=employment.Age ,palette="Set1", y = employment.Unemployed , color='black'
plt.show()
```




```
exercise = pd.read_csv("/content/drive/My Drive/Python DataScience/Visualization/Seabor
plt.figure(figsize=(10,9))
sns.set(rc={"axes.facecolor":"#b0deff","axes.grid":False,
            'xtick.labelsize':15,'ytick.labelsize':15,
            'axes.labelsize':20,'figure.figsize':(20.0, 9.0)})
params = dict(data=exercise ,x = exercise.kind ,y = exercise.pulse ,hue=exercise.diet,d
sns.stripplot(**params , size=9,jitter=0.35,palette=['#FF8F00','#558B2F'],edgecolor='bl
sns.boxplot(**params ,palette=['#909497','#D0D3D4'],linewidth=4)
plt.show()
```




```
plt.figure(figsize=(16,11))
sns.set(rc={"axes.facecolor":"#b0deff","axes.grid":False,
            'xtick.labelsize':15,'ytick.labelsize':15,
            'axes.labelsize':20,'figure.figsize':(20.0, 9.0)})
params = dict(data=employment ,x = employment.Age ,y = employment.Unemployed ,hue=emplo
sns.stripplot(**params , size=8,jitter=0.35,palette=['#33FF66','#FF6600'],edgecolor='bl
sns.boxplot(**params ,palette=['#BDBDBD','#E0E0E0'],linewidth=6)
plt.show()
```



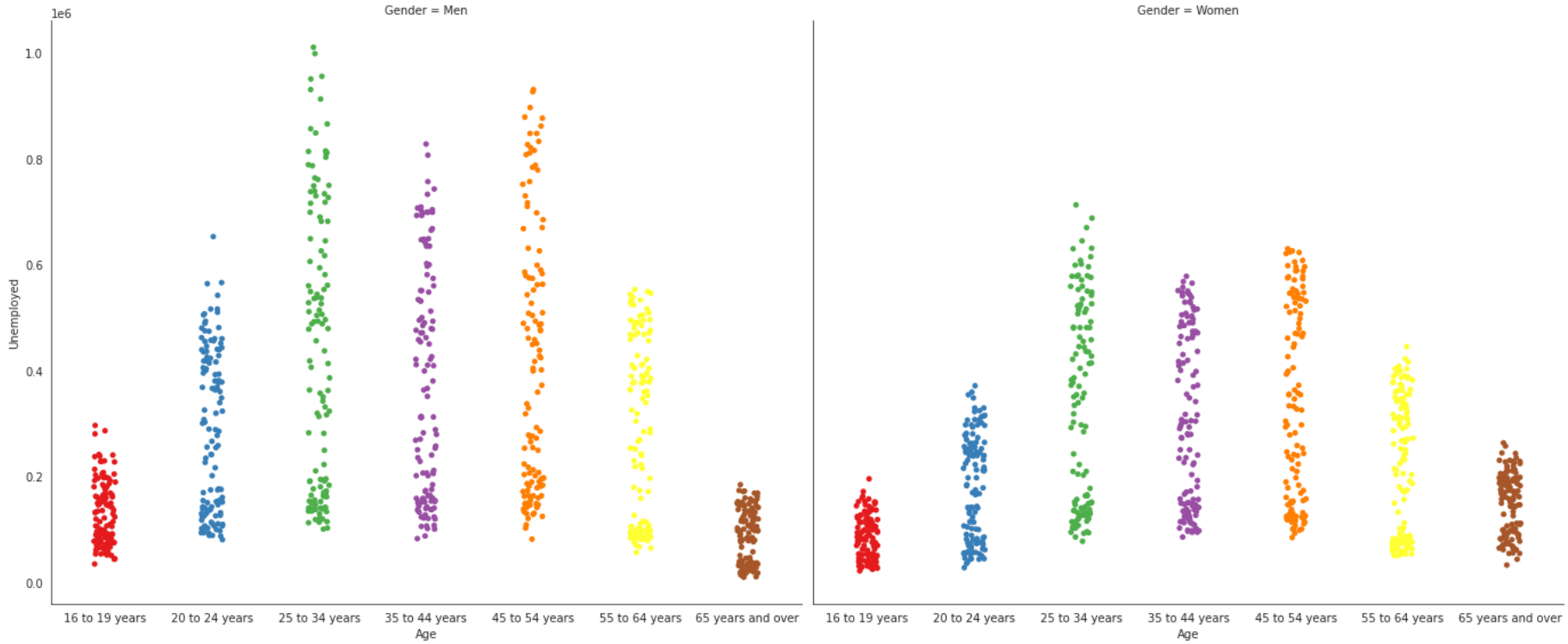

```
# Recover default matplotlib settings
mpl.rcParams.update(mpl.rcParamsDefault)
%matplotlib inline
sns.set_style("white")
```

```
# Drawing stripplot on top of a violin plot
plt.figure(figsize=(14,10))
sns.stripplot(x=employment.Age, y = employment.Unemployed , jitter=True , color="black"
sns.violinplot(x=employment.Age , y = employment.Unemployed , palette="Set1" , scale="c
plt.show()
```

1e6

```
# Facet along the columns to show a categorical variable using "col" parameter
plt.figure(figsize=(11,9))
sns.catplot(x="Age" , y = "Unemployed", col="Gender", kind="strip",palette="Set1" , hei
            data=employment)
plt.show()
```

<Figure size 792x648 with 0 Axes>



```
stdperf = pd.read_csv("/content/drive/My Drive/Python DataScience/Visualization/Seaborn
# Facet along the columns to show a categorical variable using "col" parameter
plt.figure(figsize=(11,9))
sns.catplot(x="parental level of education" , y = "math score", hue= "test preparation
            col="lunch", kind="strip",palette="Set2" , height=8, aspect=1 ,data=stdperf
plt.show()
```

```
/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:2781: UserWarning: The `split` parameter has been renamed
  warnings.warn(msg, UserWarning)
<Figure size 792x648 with 0 Axes>
```