Release notes Files ×

..

sample_data

UniversalBank.xlsx

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

```
data=pd.read_excel('UniversalBank.xlsx')
```

```
data.head()
```

	ID	Age	Experience	Income	ZIP Code	Family	CCAvg	Education	Mortgage	Perso L
0	1	25	1	49	91107	4	1.6	1	0	
1	2	45	19	34	90089	3	1.5	1	0	
2	3	39	15	11	94720	1	1.0	1	0	
3	4	35	9	100	94112	1	2.7	2	0	
4	5	35	8	45	91330	4	1.0	2	0	

Next steps: Generate code with data View recommended plots New interactive sheet

```
data.shape
```

```
(5000, 14)
```

```
data.dtypes
```

	0
ID	int64
Age	int64
Experience	int64
Income	int64
ZIP Code	int64
Family	int64
CCAvg	float64
Education	int64
Mortgage	int64
Personal Loan	int64
Securities Account	int64
CD Account	int64
Online	int64
CreditCard	int64

dtype: object

```
mean=data['Age'].mean()
```

```
std=data['Age'].std()
```

```
from scipy.stats import norm,kstest
```

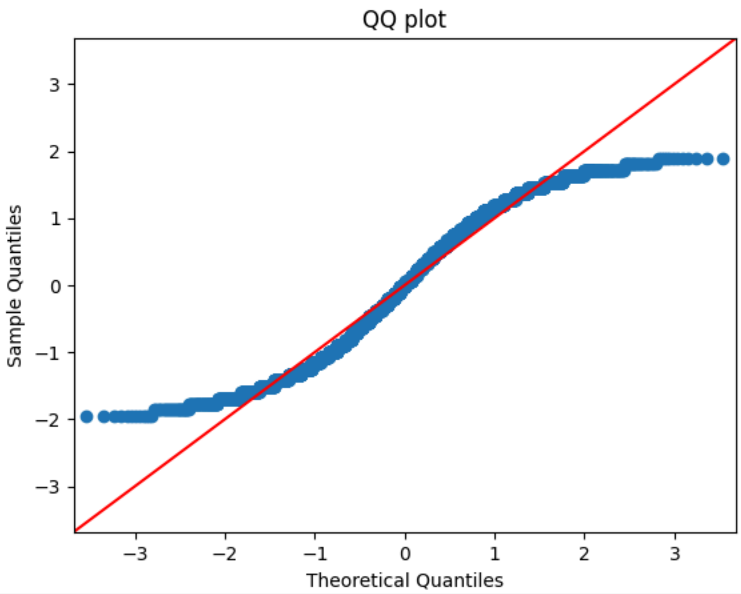
```
kstest(data['Age'],'norm',args=(mean,std))
```

```
KstestResult(statistic=0.07185631043280638, pvalue=6.78166112328824e-23,
statistic_location=54, statistic_sign=-1)
```

```
import statsmodels.api as sm
```

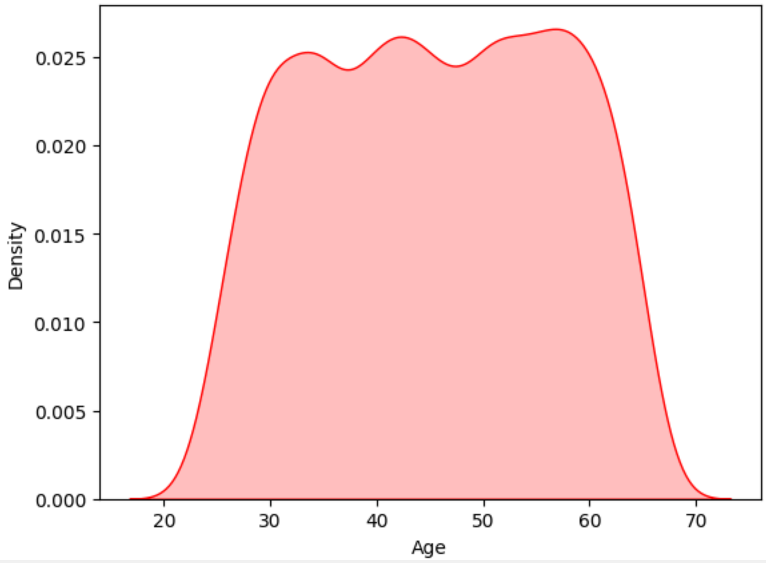
```
sm.qqplot(data['Age'],norm,fit=True,line='45')
plt.title('QQ plot')
```

```
Text(0.5, 1.0, 'QQ plot')
```



```
sns.kdeplot(data['Age'],color='red',fill=True)
```

```
<Axes: xlabel='Age', ylabel='Density'>
```



```
mean=data['Experience'].mean()
```

```
std=data['Experience'].std()
```

```
from scipy.stats import norm,kstest
```

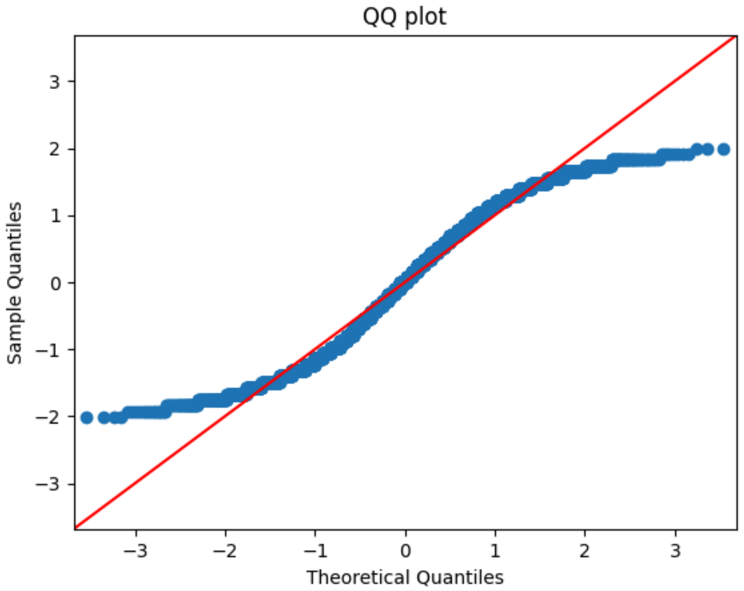
Start coding or generate with AI.

```
kstest(data['Experience'],'norm',args=(mean,std))
```

```
KstestResult(statistic=0.0686725405315961, pvalue=6.019240598975531e-21,
statistic_location=10, statistic_sign=1)
```

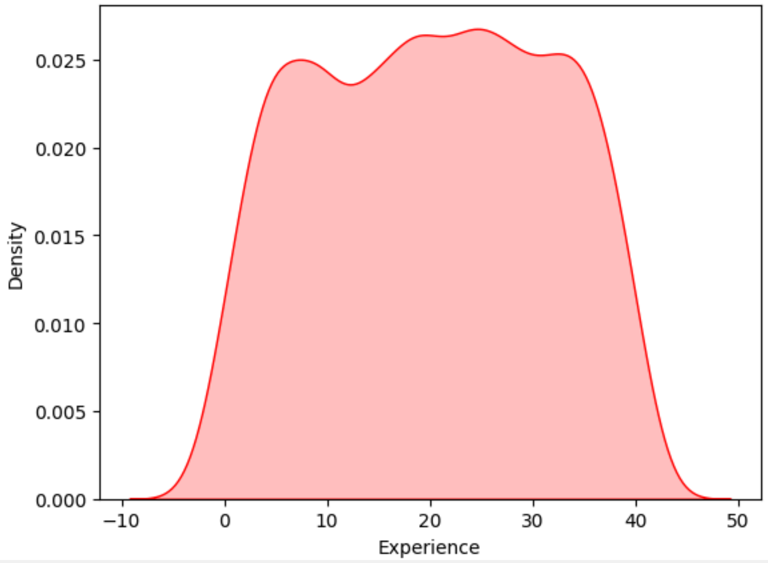
```
import statsmodels.api as sm
sm.qqplot(data['Experience'],norm,fit=True,line='45')
plt.title('QQ plot')
```

```
Text(0.5, 1.0, 'QQ plot')
```



```
sns.kdeplot(data['Experience'],color='red',fill=True)
```

```
<Axes: xlabel='Experience', ylabel='Density'>
```

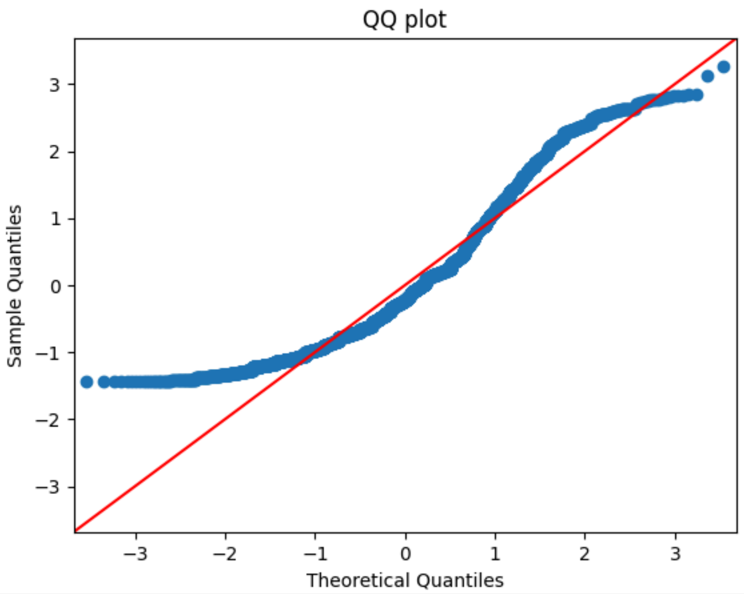


```
mean=data['Income'].mean()
std=data['Income'].std()
from scipy.stats import norm,kstest
kstest(data['Income'],'norm',args=(mean,std))
```

```
KstestResult(statistic=0.09846950047412628, pvalue=1.182660921916301e-42,
statistic_location=85, statistic_sign=1)
```

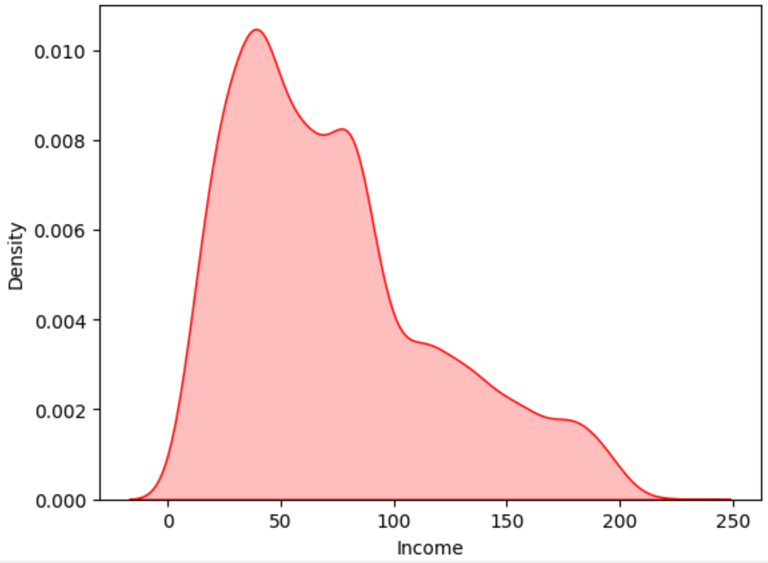
```
import statsmodels.api as sm
sm.qqplot(data['Income'],norm,fit=True,line='45')
plt.title('QQ plot')
```

```
Text(0.5, 1.0, 'QQ plot')
```



```
sns.kdeplot(data['Income'],color='red',fill=True)
```

```
<Axes: xlabel='Income', ylabel='Density'>
```

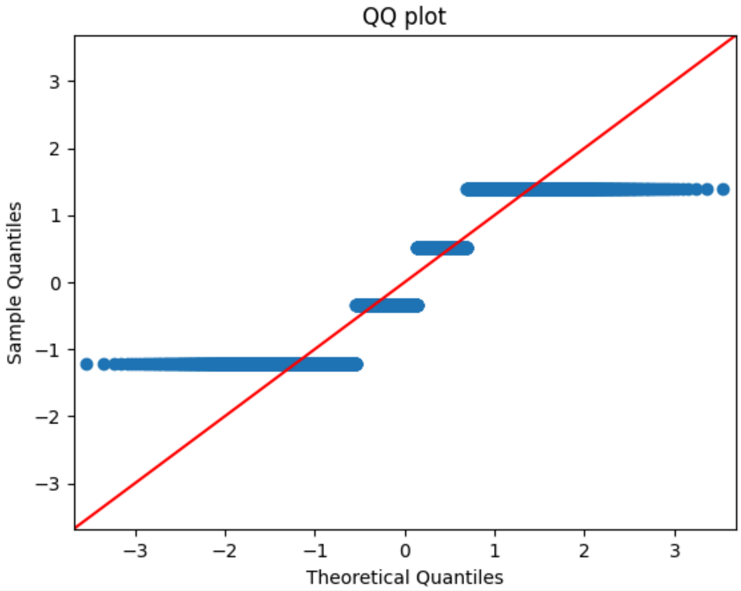


```
mean=data['Family'].mean()
std=data['Family'].std()
from scipy.stats import norm,kstest
kstest(data['Family'],'norm',args=(mean,std))
```

```
KstestResult(statistic=0.1887022392422289, pvalue=2.287412483249719e-156,
statistic_location=2, statistic_sign=1)
```

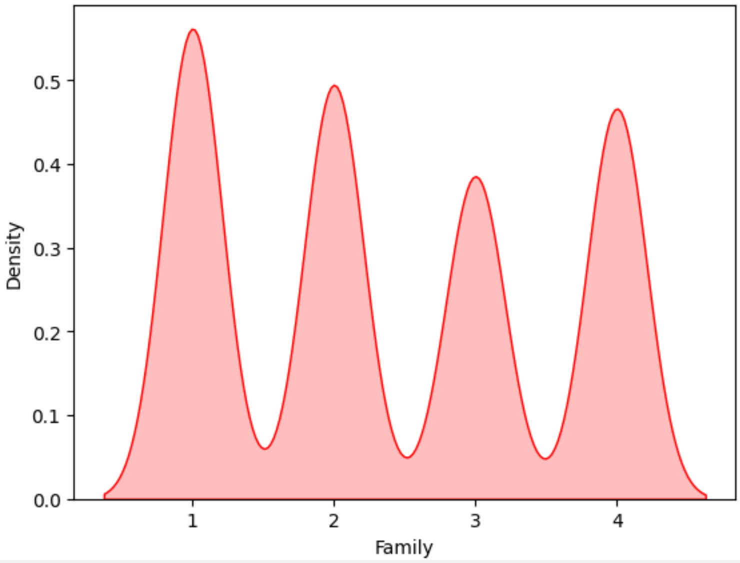
```
import statsmodels.api as sm
sm.qqplot(data['Family'],norm,fit=True,line='45')
plt.title('QQ plot')
```

```
Text(0.5, 1.0, 'QQ plot')
```



```
sns.kdeplot(data['Family'],color='red',fill=True)
```

```
<Axes: xlabel='Family', ylabel='Density'>
```

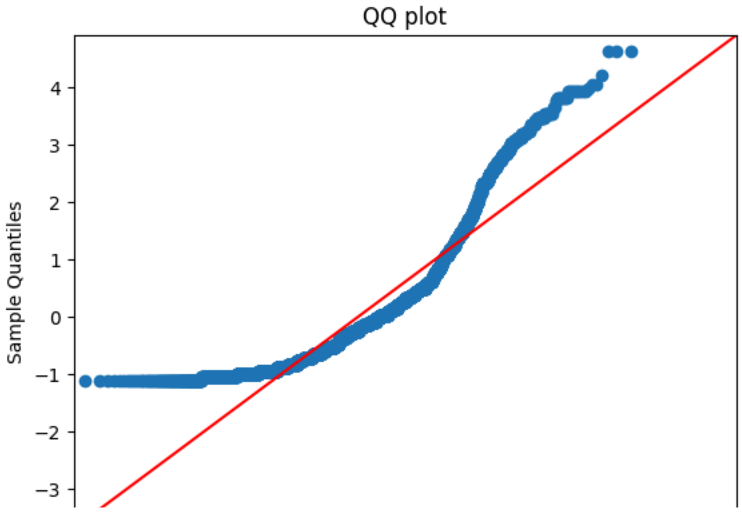


```
mean=data['CCAvg'].mean()
std=data['CCAvg'].std()
from scipy.stats import norm,kstest
kstest(data['CCAvg'],'norm',args=(mean,std))
```

```
KstestResult(statistic=0.13579658592805466, pvalue=7.024604372821239e-81,
statistic_location=2.2, statistic_sign=1)
```

```
import statsmodels.api as sm
sm.qqplot(data['CCAvg'],norm,fit=True,line='45')
plt.title('QQ plot')
```

```
Text(0.5, 1.0, 'QQ plot')
```



Disk 76.66 GB available