In [196]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import math
from scipy import stats
import statsmodels.api as sm
from matplotlib.lines import Line2D
from sklearn.linear_model import LinearRegression
```

In [4]:
```
trans = pd.read_csv("desktop/transactions.csv")
```

In [5]:
```
trans
```

0	1	2547	1.0	3.13
1	2	822	1.0	5.46
2	3	3686	1.0	6.35
3	4	3719	1.0	5.59
4	5	9200	1.0	6.88
...	...	...	...	...
64677	64678	9614	1.0	7.43
64678	64679	3320	2.0	8.59
64679	64680	1666	1.0	6.80
64680	64681	22044	1.0	4.50
64681	64682	20543	1.0	5.19

64682 rows × 4 columns

In [256]:
```
def log_likelihood(Sales_Amount,scale):
    return len(trans["Sales_Amount"])*np.log(scale) - scale*sum(trans["Sale
```

In [257]:
```
scale = np.arange(0,1,1/100)
```

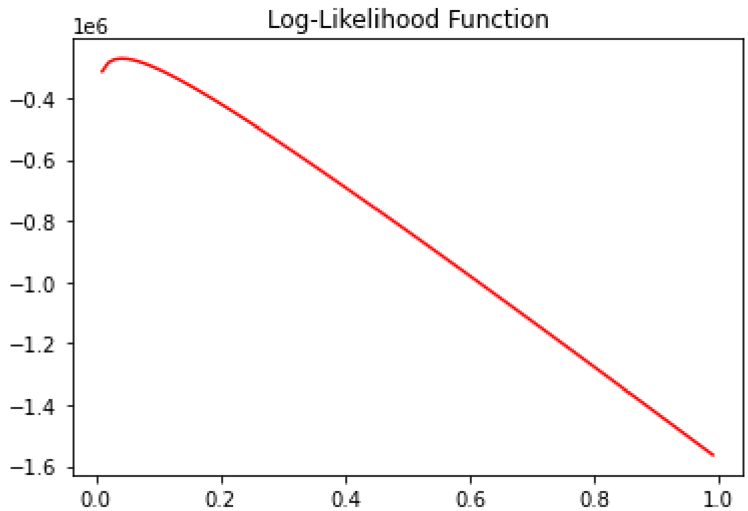
In [263]:
```
L = log_likelihood(trans["Sales_Amount"],scale)
```

In [264]:
```
print(scale[L.argmax()])
```

```
0.04
```

In [266]:
```
plt.plot(scale, L, color = 'red')
plt.title('Log-Likelihood Function', fontsize = 12)
```

Out[266]:
```
Text(0.5, 1.0, 'Log-Likelihood Function')
```



In [65]:
```
mtcars = pd.read_csv("desktop/mtcars.csv")
```

In [66]:
```
mtcars
```

Out[66]:

	model	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
0	Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
1	Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
2	Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
3	Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
4	Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
5	Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
6	Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
7	Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
8	Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
9	Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
10	Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
11	Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
12	Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
13	Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
14	Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
15	Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
16	Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
17	Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
18	Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
19	Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
20	Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
21	Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
22	AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
23	Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
24	Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
25	Fiat X1-9	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
26	Porsche 914-2	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
27	Lotus Europa	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
28	Ford Pantera L	15.8	8	351.0	264	4.22	3.170	14.50	0	1	5	4
29	Ferrari Dino	19.7	6	145.0	175	3.62	2.770	15.50	0	1	5	6
30	Maserati Bora	15.0	8	301.0	335	3.54	3.570	14.60	0	1	5	8
31	Volvo 142E	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2

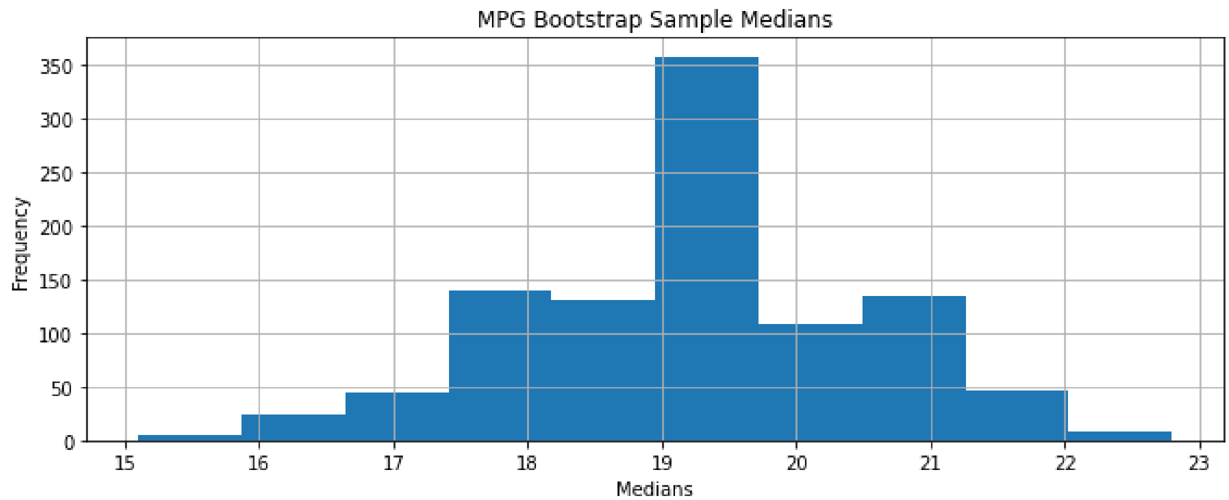
In [70]:
```
mpg_median = np.median(mtcars["mpg"])
```

In [71]:
```
mpg_median
```

Out[71]: 19.2

In [337]:
```
medians = [np.median(np.random.choice(mtcars.mpg,mtcars.shape[0])) for i in
```

In [339]:
```
plt.figure(figsize = (11, 4))
plt.title("MPG Bootstrap Sample Medians")
plt.xlabel("Medians")
plt.ylabel("Frequency")
plt.hist(medians)
plt.grid()
```



In [340]:
```
se = np.std(medians)
print("The Standard Error of the medians is: "+str(se))
```

The Standard Error of the medians is: 1.3027231008545137

In [341]:
```
norm_lower = np.median(mtcars.mpg)-1.96*se
norm_upper = np.median(mtcars.mpg)+1.96*se
```

In [342]:
```
norm_lower
```

Out[342]: 16.646662722325154

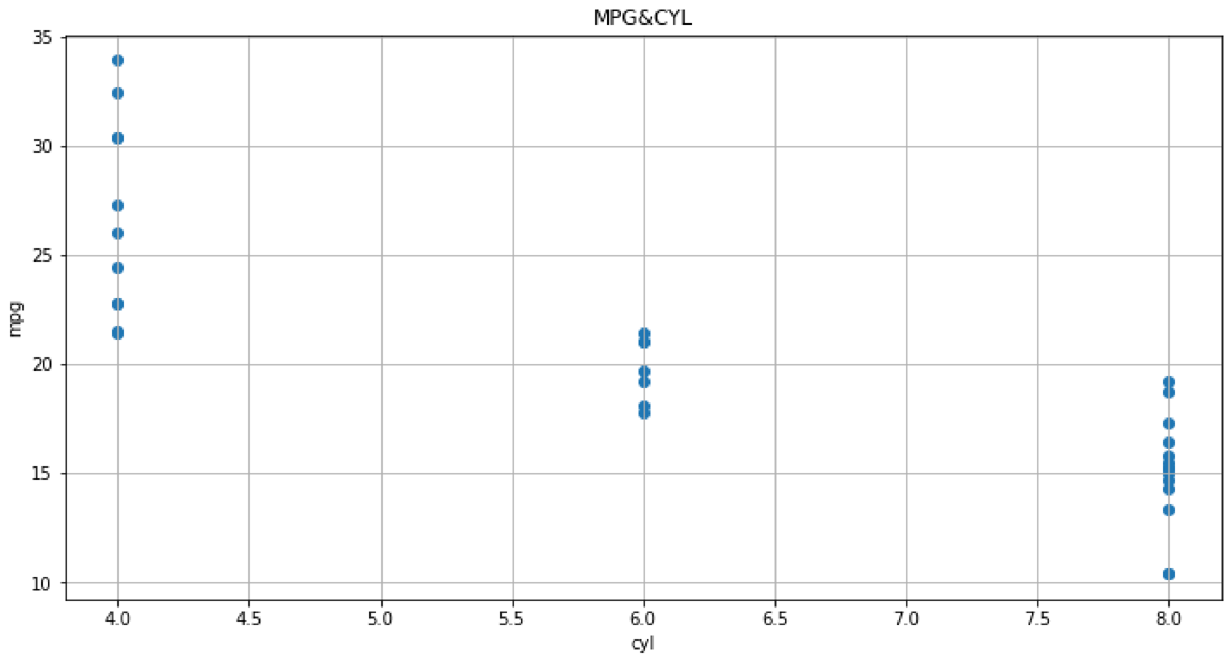
In [343]:
```
norm_upper
```

Out[343]: 21.753337277674845

In [193]:
```
Y = mtcars["mpg"]
X = mtcars["cyl"]
X = sm.add_constant(X)
```

I chose cylinders as the explanatory variable because smaller engines usually have higher mpg, so I expect a strong relationship between mpg and cyl.

In [194]:
```
plt.figure(figsize=(12,6))
plt.title("MPG&CYL")
plt.xlabel("cyl")
plt.ylabel("mpg")
plt.scatter( mtcars["cyl"] , mtcars["mpg"] )
plt.grid()
```



In [344]:
```
ols_model = sm.OLS(Y, X).fit()
ols_model.summary()
```

Out[344]:

OLS Regression Results

Dep. Variable:	mpg	**R-squared:**	0.726
Model:	OLS	**Adj. R-squared:**	0.717
Method:	Least Squares	**F-statistic:**	79.56
Date:	Wed, 20 Oct 2021	**Prob (F-statistic):**	6.11e-10
Time:	18:31:36	**Log-Likelihood:**	-81.653
No. Observations:	32	**AIC:**	167.3
Df Residuals:	30	**BIC:**	170.2
Df Model:	1		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	37.8846	2.074	18.268	0.000	33.649	42.120
cyl	-2.8758	0.322	-8.920	0.000	-3.534	-2.217

Omnibus:	1.007	**Durbin-Watson:**	1.670
Prob(Omnibus):	0.604	**Jarque-Bera (JB):**	0.874
Skew:	0.380	**Prob(JB):**	0.646
Kurtosis:	2.720	**Cond. No.**	24.1

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Looking at the coefficient of CYL, -2.8758: we conclude that as the number of cylinders go up, the mpg goes down. This makes sense because bigger engines usually have lower mpg, while smaller engines usually have higher mpg. The intercept coefficient implies that when there are zero cylinders, the mpg would be 37.8846, however, that does not make intuitive sense because there are no cars with 0 cylinders.

The R^2 is 0.726, which implies that our model explains 72.6% of the variation in mpg.

The p-values are smaller than 0.01, so we reject the null hypothesis that there is no relationship between mpg and cyl. Our variables are statistically significant.

In [ ]: