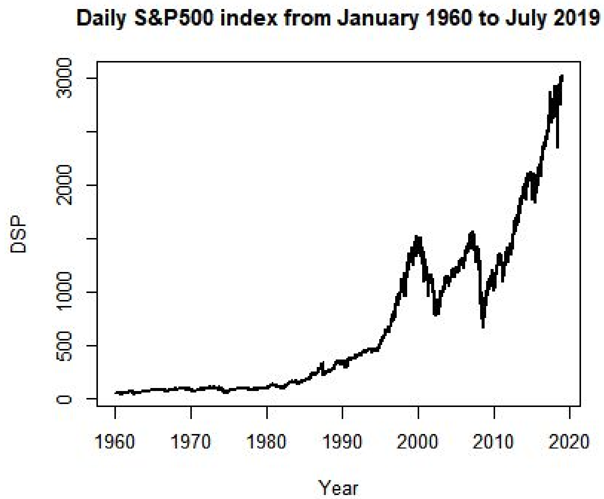
Univariate Data Analysis Case III: S&P 500 Index

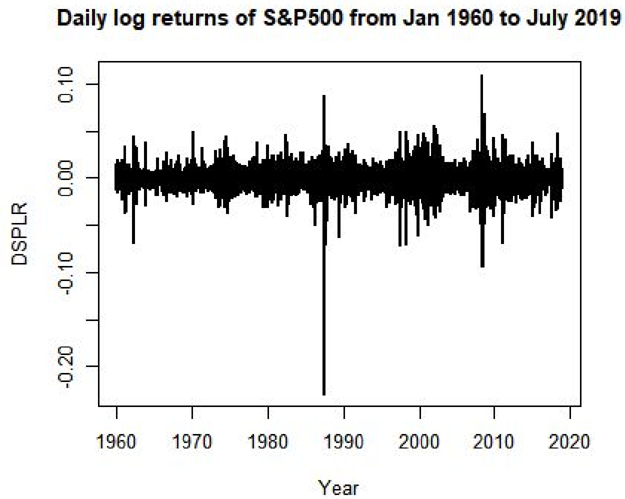
```
> names(SP)
[1] "Date"      "Open"     "High"      "Low"      "Close"     "Adj.Close"
[7] "Volume"
> str(SP)
'data.frame':     14996 obs. of  7 variables:
 $ Date     : Factor w/ 14996 levels "1959-12-31","1960-01-04",..: 1 2 3  ...
 $ Open     : num  59.9 59.9 60.4 60.1 59.7 ...
 $ High     : num  59.9 59.9 60.4 60.1 59.7 ...
 $ Low      : num  59.9 59.9 60.4 60.1 59.7 ...
 $ Close    : num  59.9 59.9 60.4 60.1 59.7 ...
 $ Adj.Close: num  59.9 59.9 60.4 60.1 59.7 ...
 $ Volume   : num  3810000 3990000 3710000 3730000 3310000 3290000  ...
> summary(SP)
         Date             Open             High       
 1959-12-31:    1   Min.   :  52.2   Min.   :  52.2  
 1960-01-04:    1   1st Qu.: 100.0   1st Qu.: 100.9  
 1960-01-05:    1   Median : 327.2   Median : 328.8  
 1960-01-06:    1   Mean   : 679.7   Mean   : 683.8  
 1960-01-07:    1   3rd Qu.:1183.5   3rd Qu.:1190.1  
 1960-01-08:    1   Max.   :3024.5   Max.   :3028.0  
 (Other)   :14990                                    
      Low               Close           Adj.Close     
 Min.   :  51.35   Min.   :  52.2   Min.   :  52.2  
 1st Qu.:  99.25   1st Qu.: 100.0   1st Qu.: 100.0  
 Median : 325.16   Median : 327.5   Median : 327.5  
 Mean   : 675.50   Mean   : 679.9   Mean   : 679.9  
 3rd Qu.:1175.17   3rd Qu.:1183.4   3rd Qu.:1183.4  
 Max.   :3014.30   Max.   :3025.9   Max.   :3025.9  

     Volume         
 Min.   :1.890e+06  
 1st Qu.:1.788e+07  
 Median :1.698e+08  
 Mean   :1.100e+09  
 3rd Qu.:1.594e+09  
 Max.   :1.146e+10  
```

We are interested in the Closing price from the year 1960 to 2019.

```
> plot(DSP_time, DSP, type="l", lwd=2, xlab = "Year", main = "Daily S&P500 
index from January 1960 to July 2019")
```

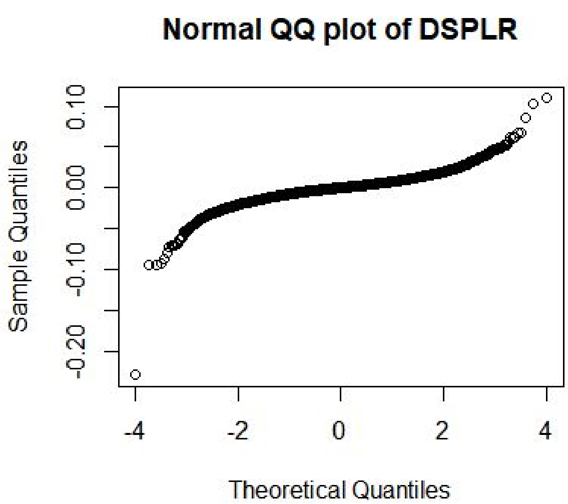
```
> plot(DSP_time[2:length(DSP)], DSPLR, type = "l", lwd=2, xlab = "Year", main =
"Daily log returns of S&P500 from Jan 1960 to July 2019")
```

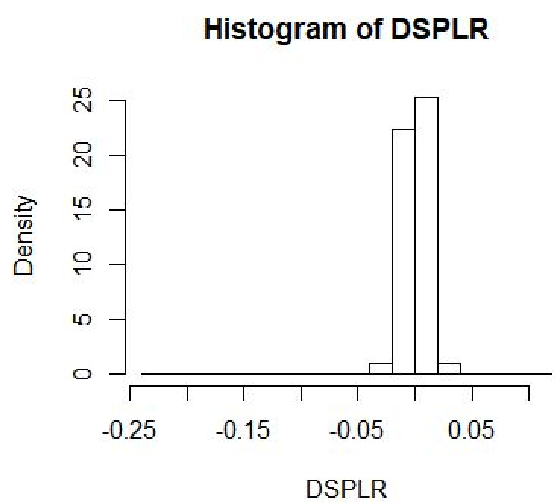




We can see that there is high sequential dependence. Computing the log of the returns, we can observe the mean reversion phenomenon. Now let us see if this can qualify to be a normal model or not.

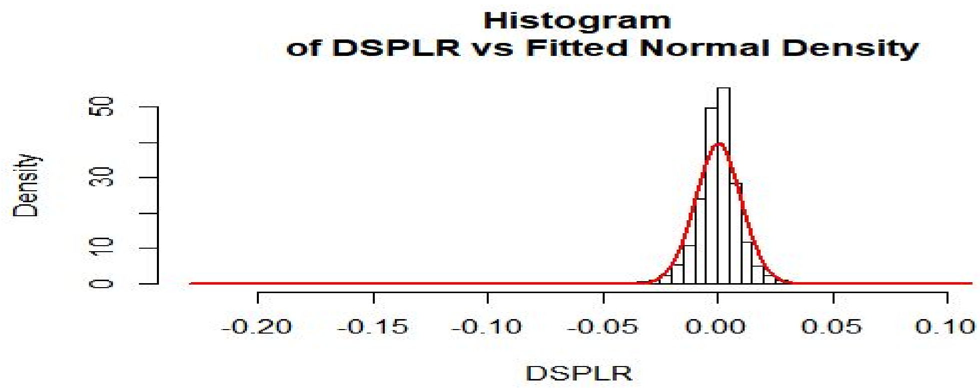
```
> qqnorm(DSPLR, main = "Normal QQ plot of DSPLR")
> hist(DSPLR, freq = FALSE, main = "Histogram of DSPLR")
```



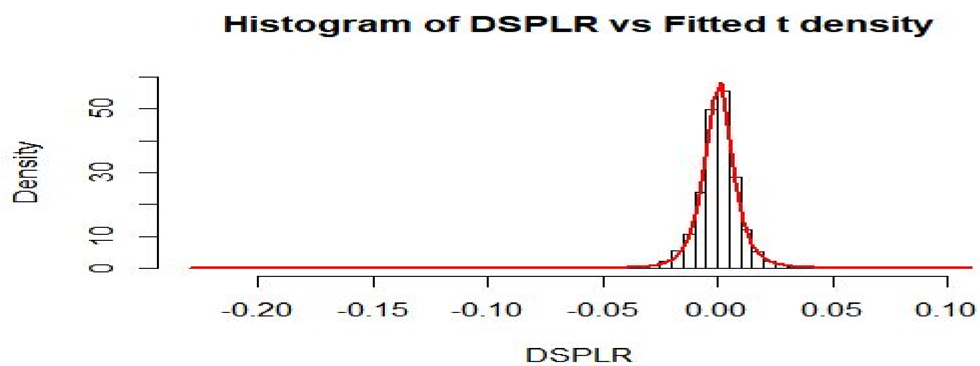


As we can see from the above plot, it is more or less a normal fit. There appears to be an outlier at the left tail of the plot. As the total area of rectangles = 1, the height function is a density and can be viewed as a non-parametric density estimation. But what happens when we try a parametric estimation like normal distribution?

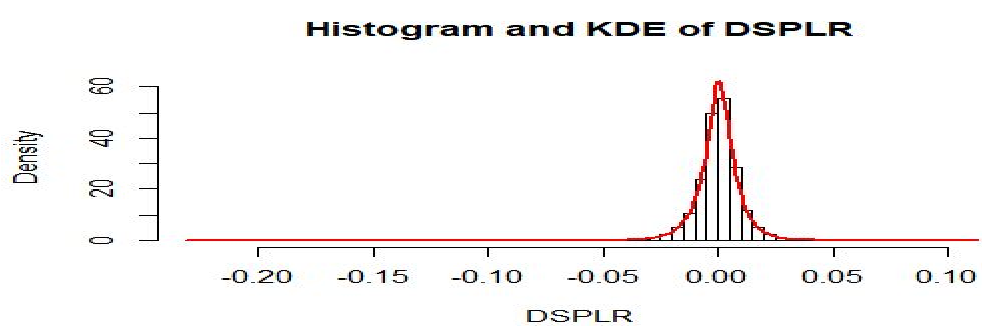
```
> hist(DSPLR, breaks = 100, freq = FALSE, main = "Histogram of DSPLR vs Fitted
Normal Density")
```



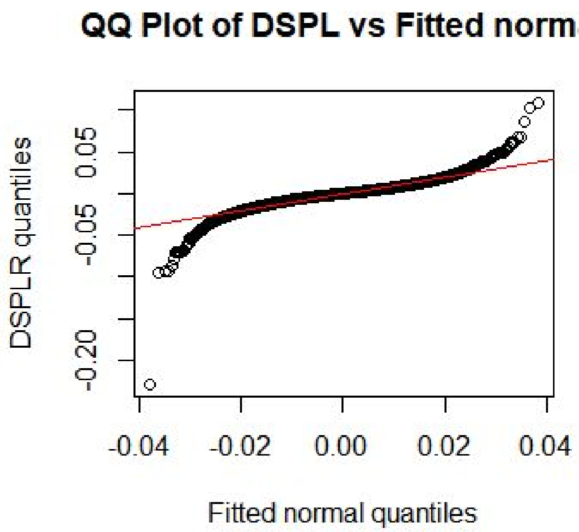
We can see that this fit is not proportionate. Hence we cannot assume a normal model here. This model is very restrictive. It is not the mean or standard deviation that makes this model restrictive. It is the normality assumption. Hence let us try fitting t-model into the data. We will be using the **library(MASS)** for the same.

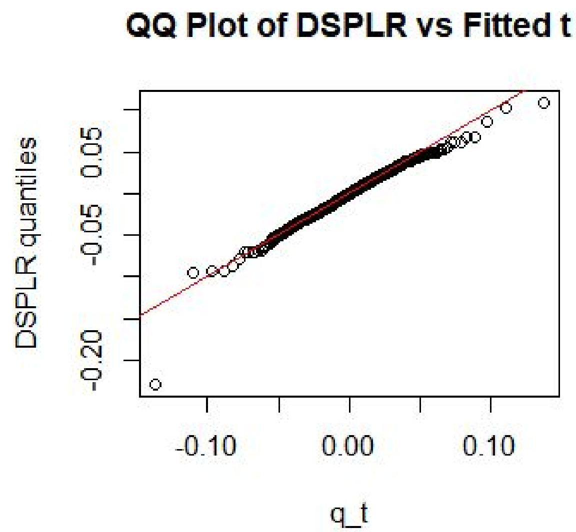


As we can see this is a much better fit and not restrictive model. Let us try Kernel density estimator on this model.



Hence KDE is much smoother although it depends heavily on bandwidth. It is quite hard to understand tail behavior though. Hence this could be very bad for industries which heavily rely on risk and operations management. Heavy tailedness is what most risk applications look for. Let us look into the QQ plot of Fitted normal model and Fitted T model.





Now let us look into the Value at risk computations for various distributions of DSPLR:

```
> c(var_emp, var_normal, var_t)
        1%                      m
0.02694273 0.02298725 0.02663202
```

Hence we can see how the normal model underestimates the VaR.