Time Series Modeling

Time series data can be modelled as the addition of the following components

1. Trend component (T)
   1. Consistent upward or downward movement of data
2. Seasonal component (S)
   1. Fluctuations from the trend occurring at specific periods
3. Level component (L)
4. Cyclical component ( C )
   1. Fluctuations occurring due to macro-economic changes
5. Irregular component (I)
   1. Noise or random errors that follow Gaussian distributions

**Additive model**

Yt = Tt + St + Ct + It

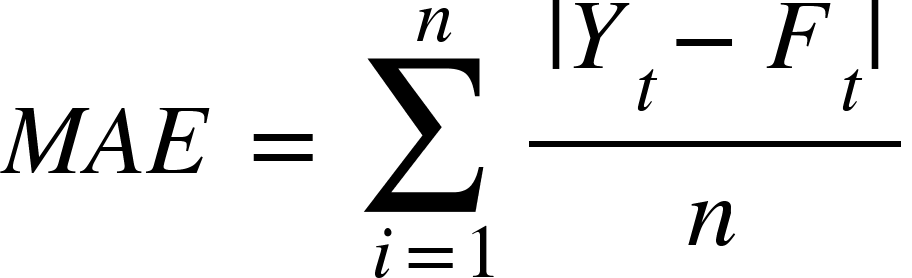
**Multiplicative model**

Yt = Tt \* St \* Ct \* It

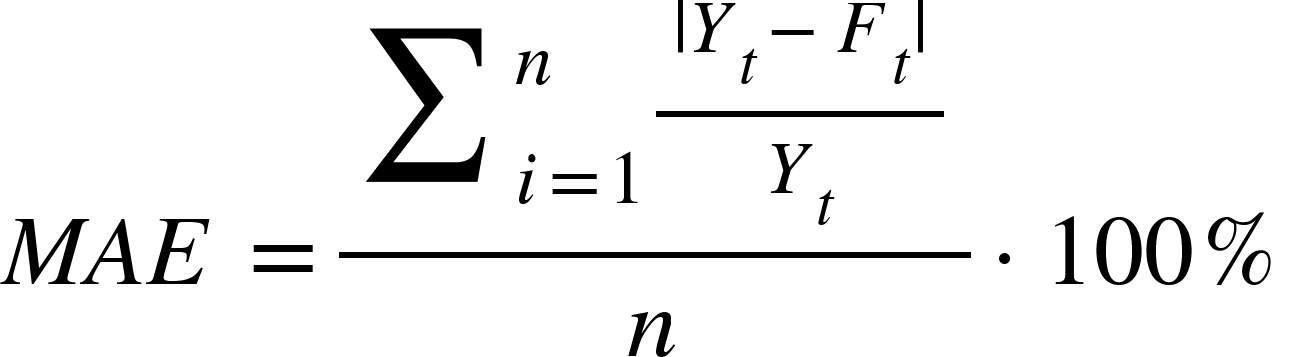
Usually, just trend and seasonality are taken as for C we need a large amount of data.

**Forecasting accuracy measures**

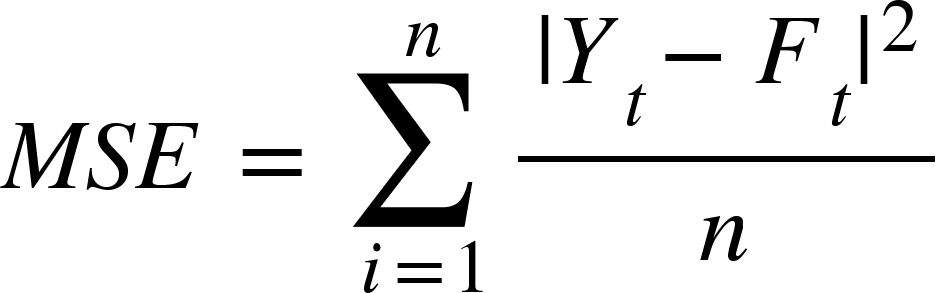
1. Mean absolute error (MAE)



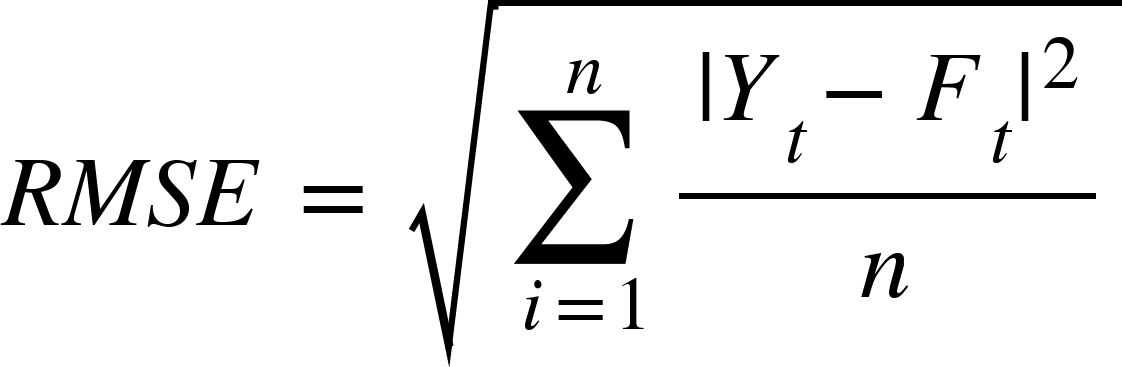
1. Mean Absolute Percentage error (MAPE)



1. Mean Squared Error (MSE)

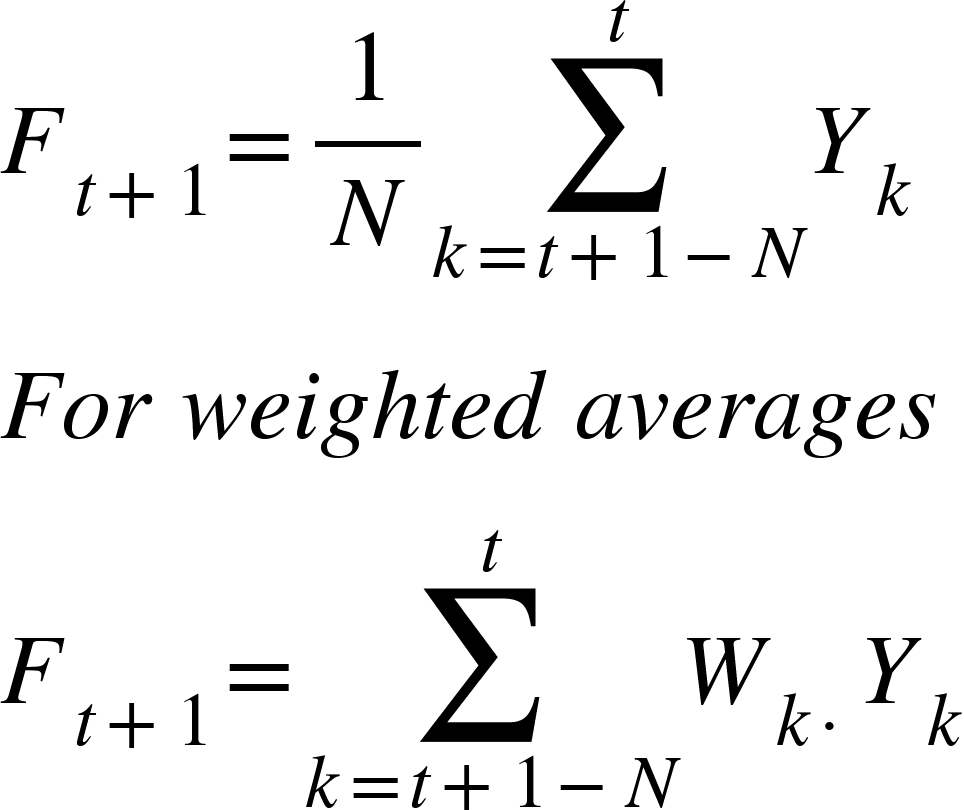


1. Root Mean Squared Error (RMSE)



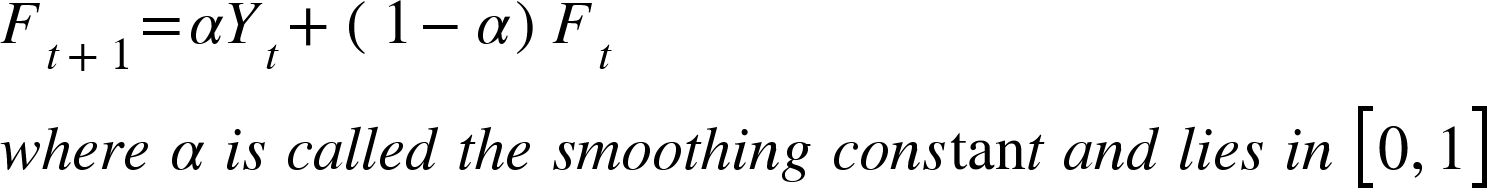
**Forecasting Techniques**

**Moving Average**



Here, the weights are normalised as well.

**Single Exponential Smoothing**



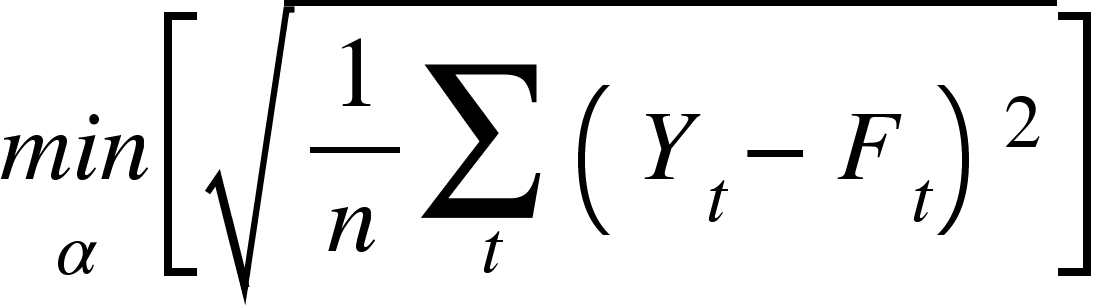
Advantages

1. Uses all the historic data
2. Assigns progressively decreasing weights to older data

Disadvantages

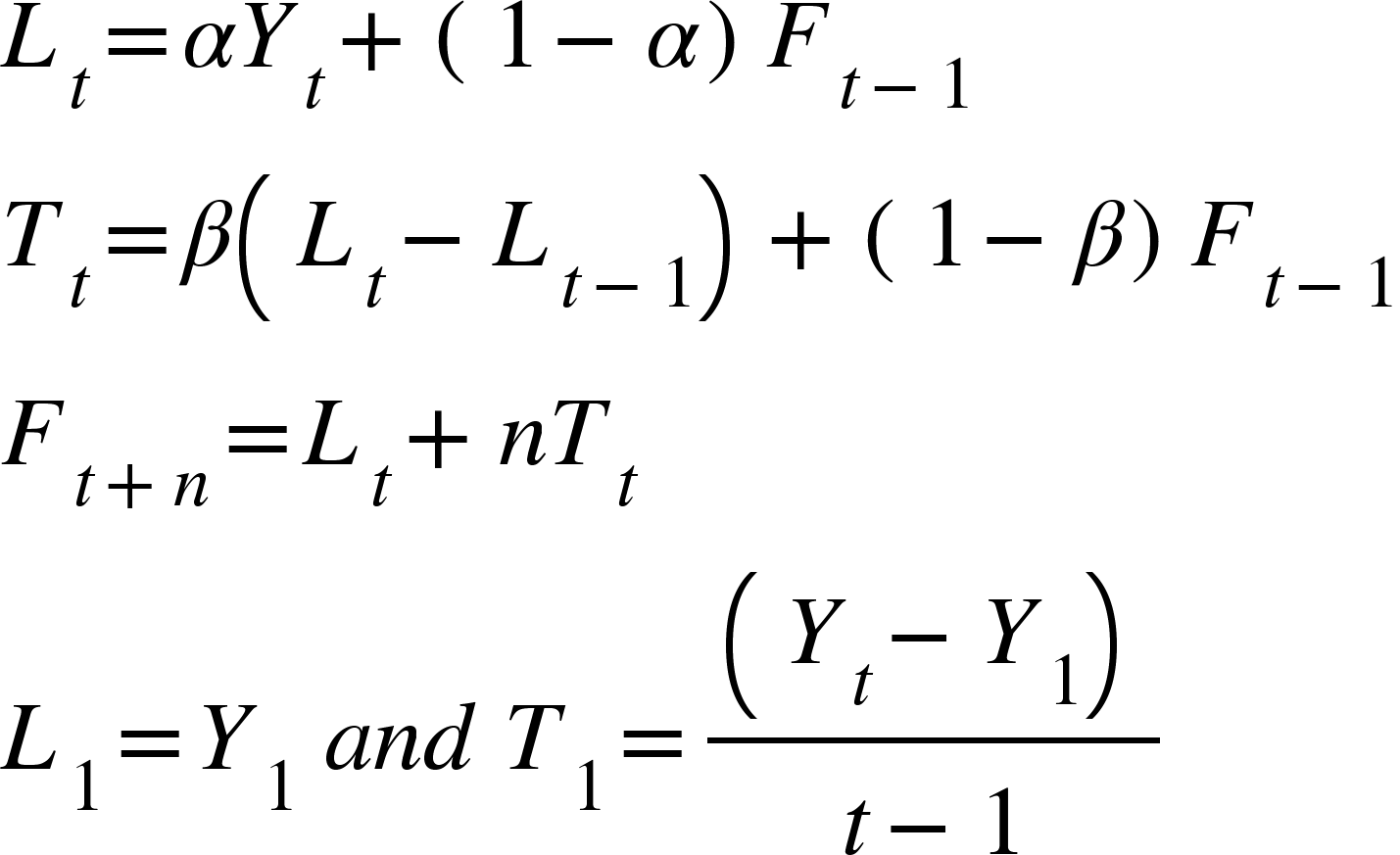
1. Increasing n makes forecast less sensitive to changes in data
2. Always lags behind the trend as it is based on past observations
3. Forecast bias and systematic errors occur when observations have strong trend or seasonal patterns

The optimal smoothing constant is chosen by finding



**Double Exponential Smoothing - Holt’s Model**

The model is given by

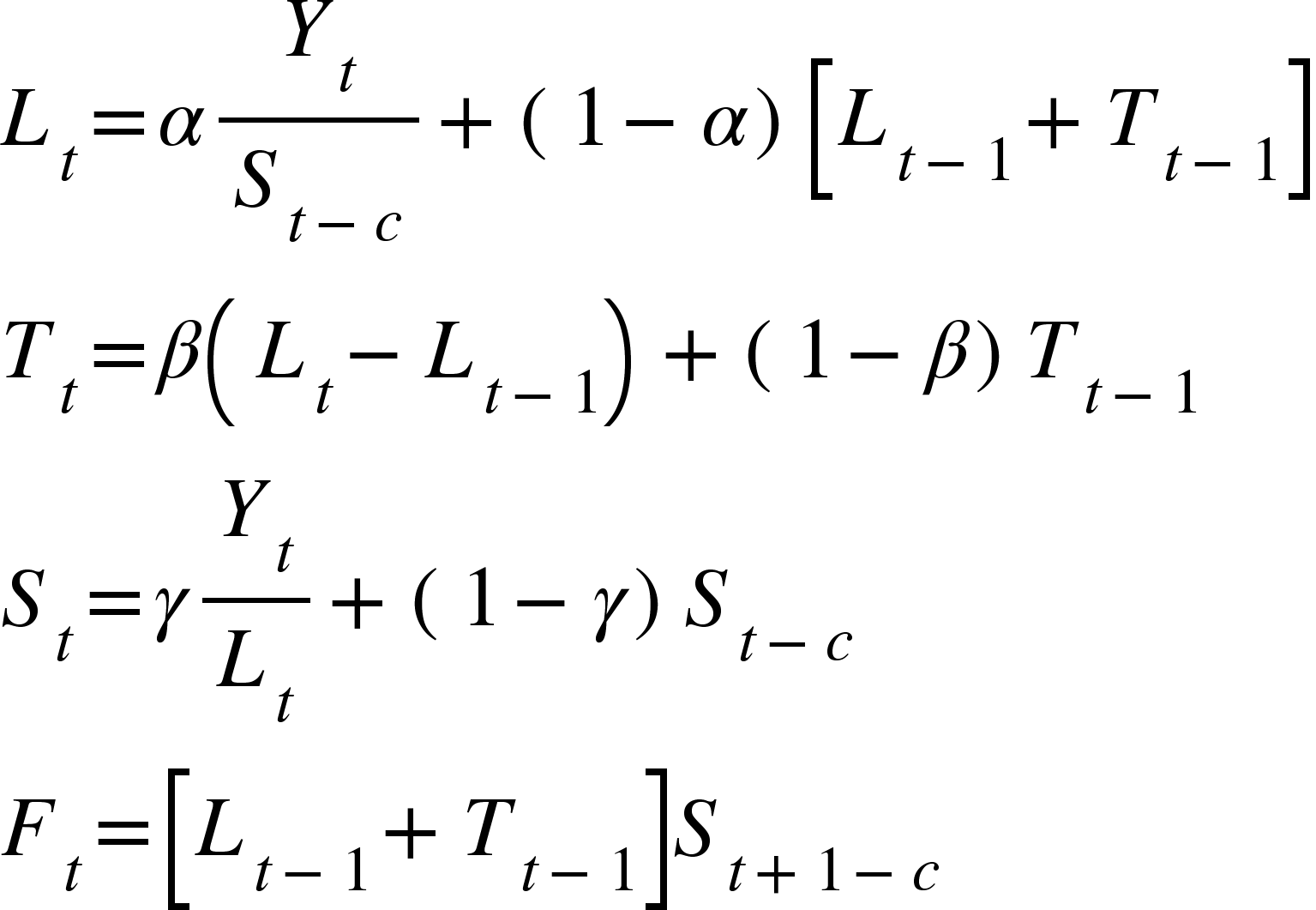


Here alpha and beta can be found by finding the values for which the RMSE is minimized.

The drawback of this model is that it does not capture the influence of seasonality.

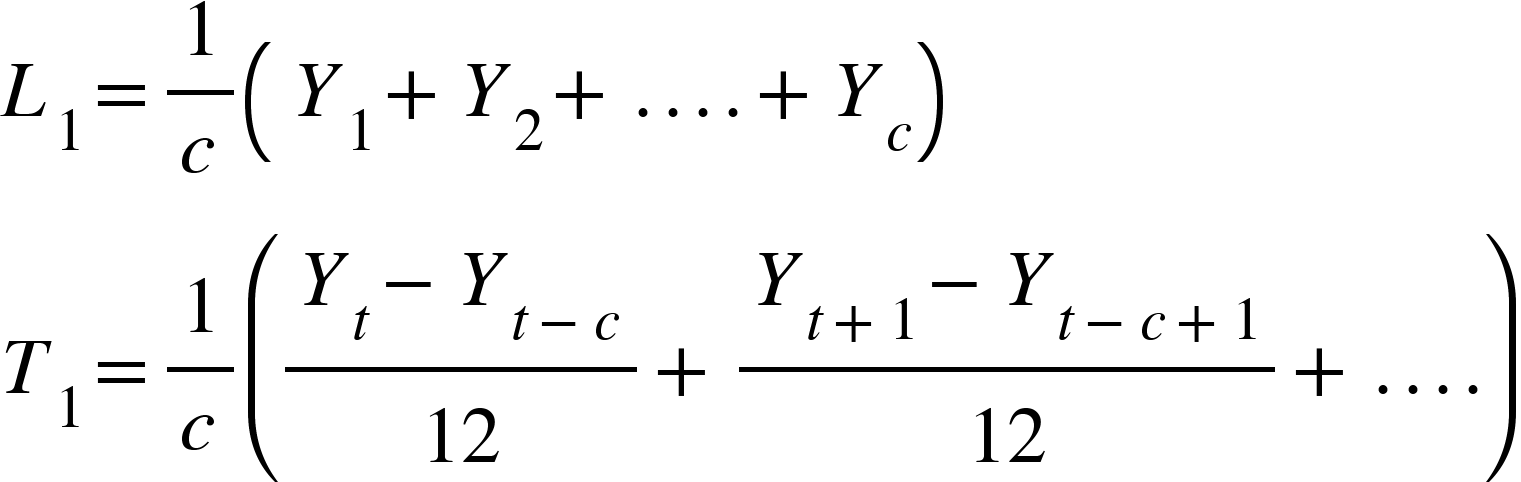
**Triple Exponential Smoothing - Holt-Winters Model**

The equations are given as

****

Where c is the number of seasons.

The initial values are calculated as



**Croston’s Forecasting for Intermittent Demand**

Yt = Demand at time t

Ft = Forecasted demand

TDt = Time between last and previous non-zero demand in period t

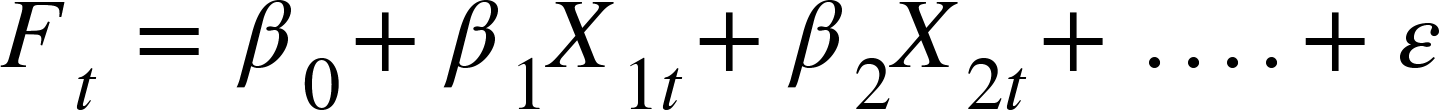
FTDt = Forecasted time between demand at period t

If Yt = 0, then Ft+1 = Ft and FTDt+1 = FTDt.

Else Ft+1 = alpha \* Yt + (1 - alpha)\*Ft and FTDt = beta\*TDt + (1 - beta)\*FTDt

Mean demand per period Dt+1 = Ft+1/FTDt+1

**Regression Model for Forecasting**

****

**Forecasting time series data with Seasonal Variations**

1. Estimate seasonality index
2. Deseasonalize data using additive or multiplicative model
3. Develop a forecasting model on the deseasonalized data
4. Ft+1 = Fd,t + St+1

For a time series data is Yt is stationary

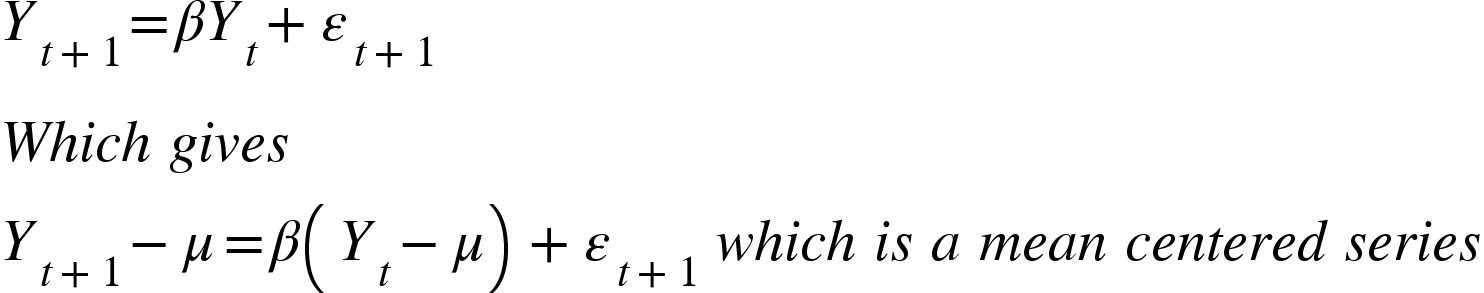
1. The mean values of Yt at different t are constant
2. The variances of Yt at different t are constant (Homoscedastic)
3. The covariances of Yt and Yt-k different lags depend only on k and not on t

Non stationary time series need to be converted to stationary time series to apply models such as AR and MA and ARMA.

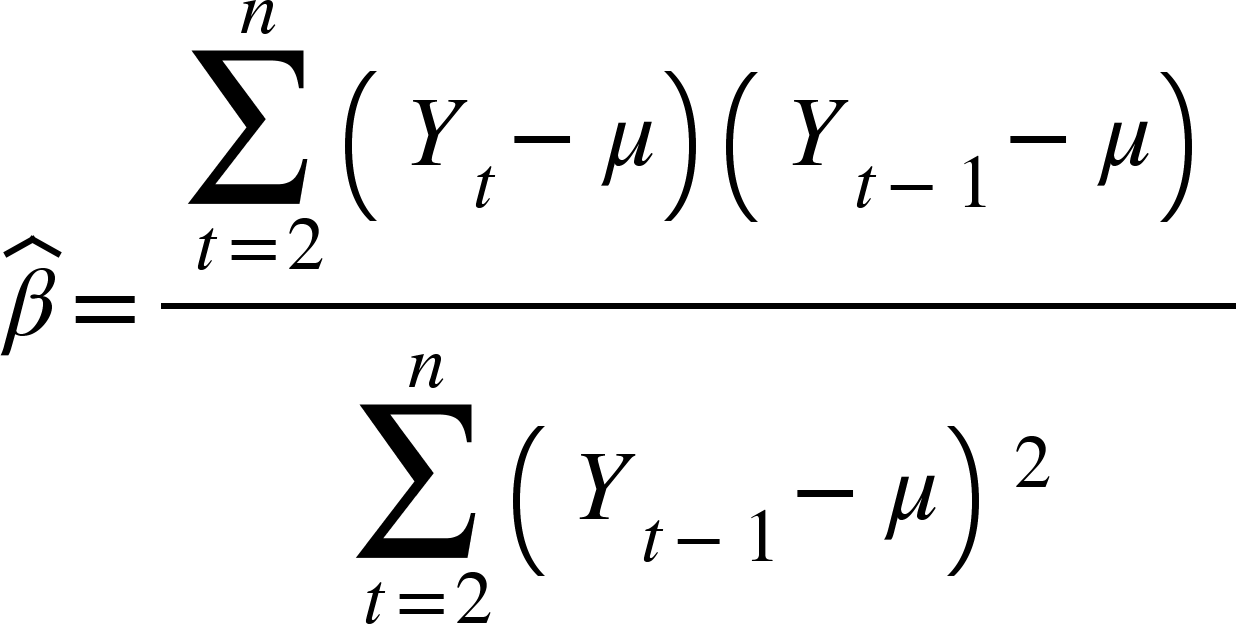
**AutoRegression (AR)**

In AR, we assume the errors follow a white noise, where a white noise is a process of residuals following a Gaussian distribution.

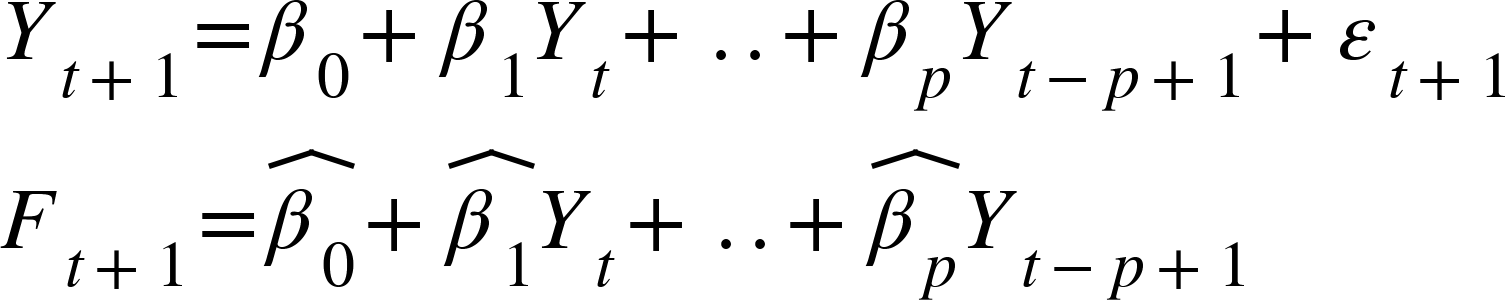
Auto regression is regression of a variable on itself measured at different time points. AR models with lag 1 or AR(1) is given by



Where

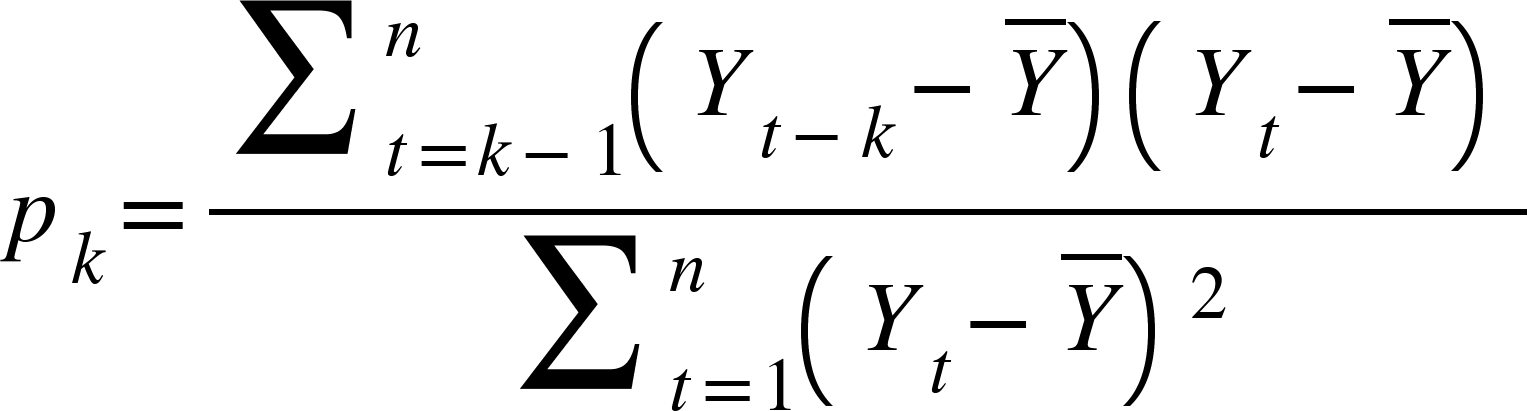


This means that for p lags, AR(p) is given as



**AR Model identification**

The autocorrelation of k lags is given by



The plot of the autocorrelation for different values of k is called the autocorrelation function (ACF).

The partial autocorrelation function is obtained when the effect of the intermediate values between Yt-k and Yt is removed and the resulting coefficient is plotted.

Hypothesis tests can be carried out to check whether the autocorrelation and partial autocorrelation coefficients are different from 0 as

H0 : pk = 0 and HA : pk != 0 where pk is the AC coefficient

H0 : ppk = 0 and HA : ppk != 0 where ppk is the PAC coefficient

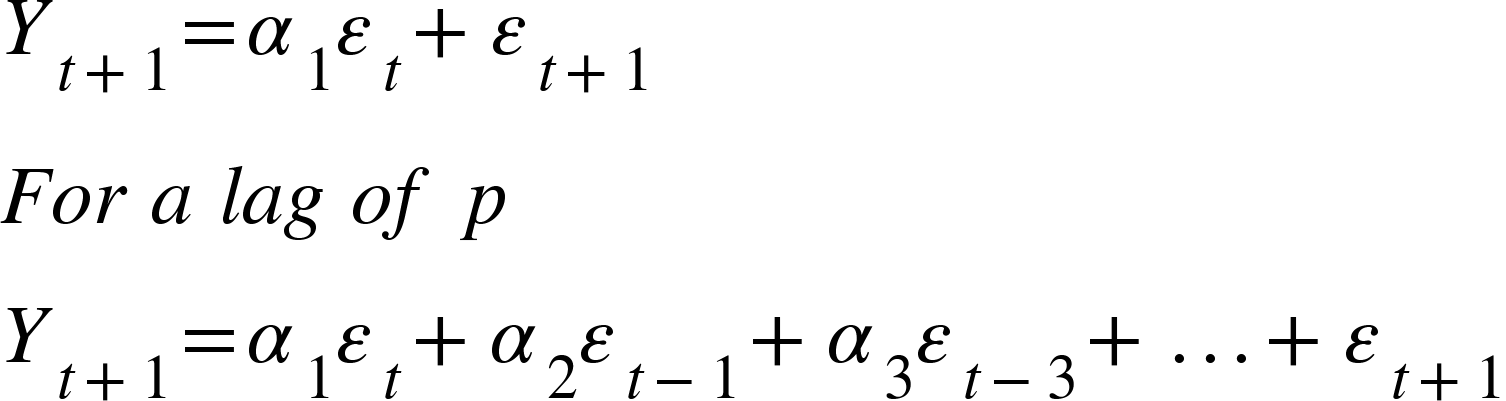
The null hypothesis is rejected when p > 1.96/sqrt(n).

To select appropriate p in AR,

1. The PAC ppk > 1.96/sqrt(n) for first p values and then is 0
2. The ACF pk decreases exponentially.

**Moving Average (MA)**

For a lag of 1



The value of p can be determined as

1. The AC pk > 1.96/sqrt(n) for first p values and then is 0
2. The PACF ppk decreases exponentially.

**AutoRegressive Moving Average (ARMA)**

ARMA(p, q) is given as

<math xmlns="http://www.w3.org/1998/Math/MathML"><msub><mi>Y</mi><mrow><mi>t</mi><mo>+</mo><mn>1</mn></mrow></msub><mo>=</mo><msub><mi>&#x3B2;</mi><mn>1</mn></msub><msub><mi>Y</mi><mi>t</mi></msub><mo>+</mo><msub><mi>&#x3B2;</mi><mn>2</mn></msub><msub><mi>Y</mi><mrow><mi>t</mi><mo>-</mo><mn>2</mn></mrow></msub><mo>+</mo><mo>.</mo><mo>.</mo><mo>+</mo><msub><mi>&#x3B2;</mi><mi>p</mi></msub><msub><mi>Y</mi><mrow><mi>t</mi><mo>-</mo><mi>p</mi><mo>+</mo><mn>1</mn></mrow></msub><mo>&#xA0;</mo><mo>+</mo><mo>&#xA0;</mo><msub><mi>&#x3B1;</mi><mn>1</mn></msub><msub><mi>Y</mi><mi>t</mi></msub><mo>+</mo><msub><mi>&#x3B1;</mi><mn>2</mn></msub><msub><mi>Y</mi><mrow><mi>t</mi><mo>-</mo><mn>2</mn></mrow></msub><mo>+</mo><mo>.</mo><mo>.</mo><mo>.</mo><mo>+</mo><msub><mi>&#x3B1;</mi><mi>q</mi></msub><msub><mi>Y</mi><mrow><mi>t</mi><mo>-</mo><mi>q</mi><mo>+</mo><mn>1</mn></mrow></msub><mo>+</mo><msub><mi>&#x3B5;</mi><mrow><mi>t</mi><mo>+</mo><mn>1</mn></mrow></msub></math>

The parameters are estimated by the Box-Jenkins method

1. The AC pk > 1.96/sqrt(n) for first p values and then is 0
2. The PAC ppk > 1.96/sqrt(n) for first q values and then is 0

**AutoRegressive Integrated Moving Average (ARIMA)**

It has 3 components

1. AR(p)
2. Integration component d
3. MA(q)

The main objective of d is to convert the non-stationary process to a stationary process so AR and MA can be used. The non-stationary condition can be observed by the ACF plot.

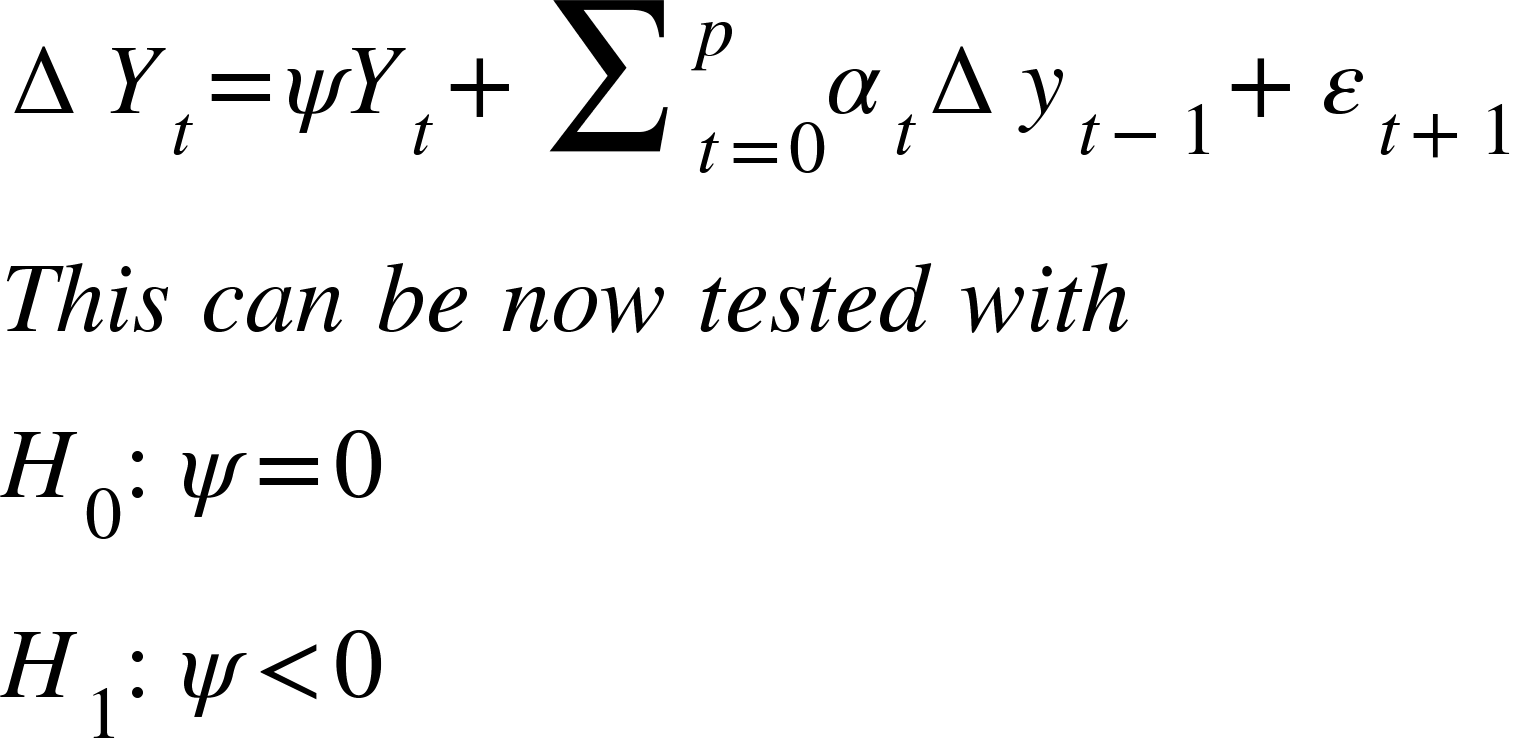
**The Dickey-Fuller Test**

An AR(1) process can become very large when beta > 1 and is non-stationary when it is equal to 1.

<math xmlns="http://www.w3.org/1998/Math/MathML"><msub><mi>H</mi><mn>0</mn></msub><mo>:</mo><mo>&#xA0;</mo><mi>&#x3B2;</mi><mo>=</mo><mn>1</mn><mo>&#xA0;</mo><mi>a</mi><mi>n</mi><mi>d</mi><mo>&#xA0;</mo><msub><mi>H</mi><mi>a</mi></msub><mo>:</mo><mo>&#xA0;</mo><mi>&#x3B2;</mi><mo>&#xA0;</mo><mo>&lt;</mo><mn>1</mn><mspace linebreak="newline"/><mi>T</mi><mi>h</mi><mi>e</mi><mo>&#xA0;</mo><mi>A</mi><mi>R</mi><mfenced><mn>1</mn></mfenced><mo>&#xA0;</mo><mi>c</mi><mi>a</mi><mi>n</mi><mo>&#xA0;</mo><mi>b</mi><mi>e</mi><mo>&#xA0;</mo><mi>w</mi><mi>r</mi><mi>i</mi><mi>t</mi><mi>t</mi><mi>e</mi><mi>n</mi><mo>&#xA0;</mo><mi>a</mi><mi>s</mi><mspace linebreak="newline"/><msub><mi>Y</mi><mrow><mi>t</mi><mo>+</mo><mn>1</mn></mrow></msub><mo>-</mo><msub><mi>Y</mi><mi>t</mi></msub><mo>=</mo><mfenced><mrow><mi>&#x3B2;</mi><mo>-</mo><mn>1</mn></mrow></mfenced><msub><mi>Y</mi><mi>t</mi></msub><mo>+</mo><msub><mi>&#x3B5;</mi><mrow><mi>t</mi><mo>+</mo><mn>1</mn></mrow></msub><mo>=</mo><mi>&#x3C8;</mi><msub><mi>Y</mi><mi>t</mi></msub><mo>+</mo><msub><mi>&#x3B5;</mi><mrow><mi>t</mi><mo>+</mo><mn>1</mn></mrow></msub><mspace linebreak="newline"/><mi>T</mi><mi>o</mi><mo>&#xA0;</mo><mi>m</mi><mi>a</mi><mi>k</mi><mi>e</mi><mo>&#xA0;</mo><mi>t</mi><mi>h</mi><mi>e</mi><mo>&#xA0;</mo><mi>h</mi><mi>y</mi><mi>p</mi><mi>o</mi><mi>t</mi><mi>h</mi><mi>e</mi><mi>s</mi><mi>i</mi><mi>s</mi><mspace linebreak="newline"/><msub><mi>H</mi><mn>0</mn></msub><mo>:</mo><mo>&#xA0;</mo><mi>&#x3C8;</mi><mo>=</mo><mn>0</mn><mo>&#xA0;</mo><mi>a</mi><mi>n</mi><mi>d</mi><mo>&#xA0;</mo><msub><mi>H</mi><mi>a</mi></msub><mo>:</mo><mo>&#xA0;</mo><mi>&#x3C8;</mi><mo>&#xA0;</mo><mo>&lt;</mo><mn>0</mn><mspace linebreak="newline"/><mi>T</mi><mi>h</mi><mi>e</mi><mo>&#xA0;</mo><mi>t</mi><mi>e</mi><mi>s</mi><mi>t</mi><mo>&#xA0;</mo><mi>s</mi><mi>t</mi><mi>a</mi><mi>t</mi><mi>i</mi><mi>s</mi><mi>t</mi><mi>i</mi><mi>c</mi><mo>&#xA0;</mo><mi>i</mi><mi>s</mi><mo>&#xA0;</mo><mi>t</mi><mi>h</mi><mi>u</mi><mi>s</mi><mo>&#xA0;</mo><mi>g</mi><mi>i</mi><mi>v</mi><mi>e</mi><mi>n</mi><mo>&#xA0;</mo><mi>a</mi><mi>s</mi><mo>&#xA0;</mo><mi>D</mi><mi>F</mi><mo>&#xA0;</mo><mo>=</mo><mfrac><mrow><mo>&#xA0;</mo><mi>&#x3C8;</mi></mrow><mrow><mi>E</mi><mi>r</mi><mi>r</mi><mi>o</mi><mi>r</mi><mfenced><mi>&#x3C8;</mi></mfenced></mrow></mfrac></math>

**Augmented DF**

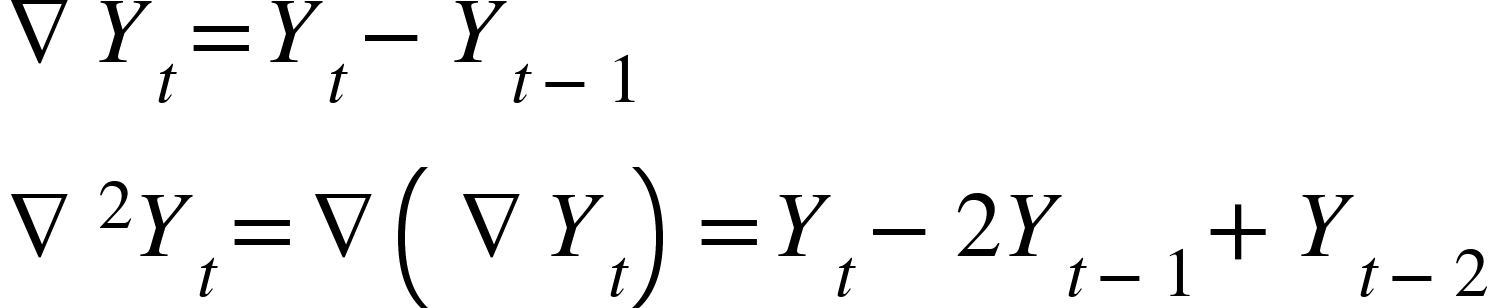
The DF test is correct only when the errors follow a white noise. If they don’t



**Transforming the process into stationary by differencing**

The order of differencing d is used here, which are the number of differences to be taken.

This is for d = 1 and d = 2.



**ARIMA Model building - Box Jenkins**

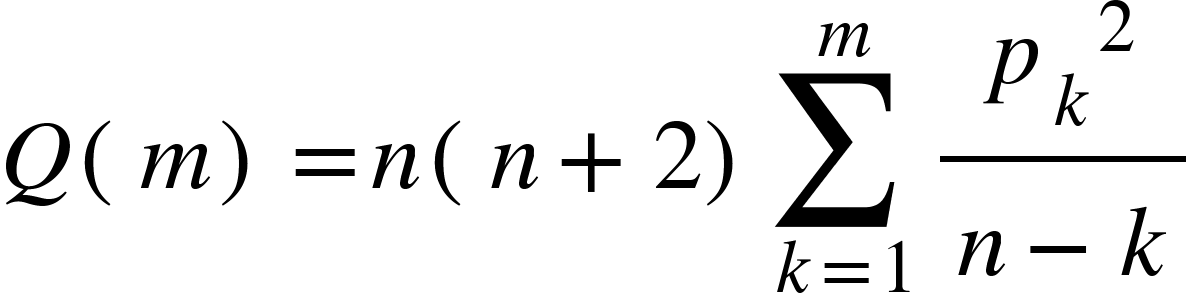
1. Model identification
   1. Plot ACF and PACF
   2. Check if process is stationary for the parameters p,d,q
   3. If d = 0, it is ARMA(p,q)
   4. Else, make process stationary and use the model
2. Parameter estimation and Model selection
   1. Estimate coefficients using OLS
   2. Model selection can be done using
      1. Akalke Information Criteria (AIC) = -2LL + 2K
      2. Bayesian Information Criteria (BIC) = -2LL + Kln(n)
   3. Where LL is the log likelihood and K is the number of parameters estimated
3. Model validation
   1. Goodness of fit test with Ljung-Box test

**Ljung-Box test for AutoCorrelations**

H0: Model does show lack of fit

H1: Model shows lack of fit

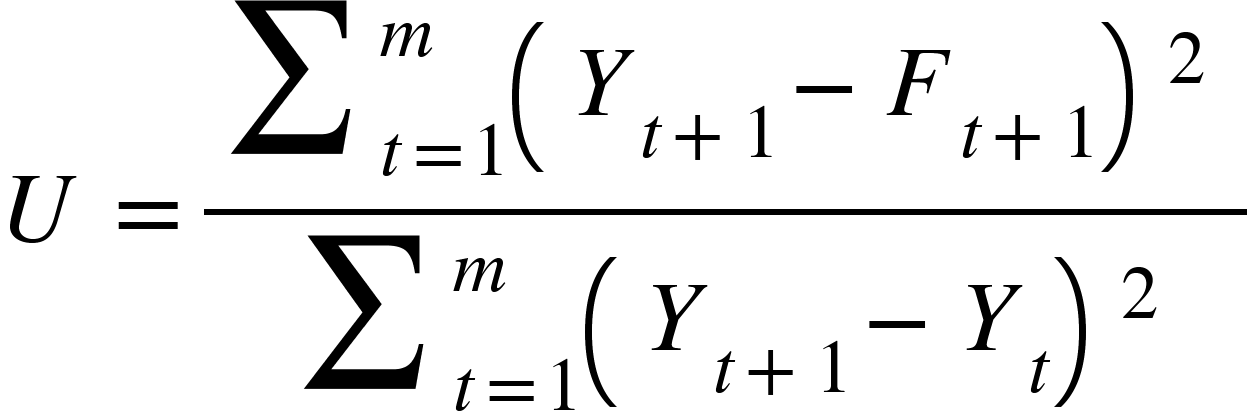
The Q statistic is given by



This is approximately a chi-square distribution with m-p-q degrees of freedom.

**Theill’s coefficient**

The power of a forecasting model is given by this coefficient as



Where it is the ratio between the forecasting model used and the Naive model.