*Notes on:*

**Forecasting with Univariate Box-Jenkins Models**

by Alan Pankratz

**CHAPTER 1: Overview**

Forecasting is an important tool for business operations, allowing them to predict future environments and make decisions based upon those predictions. This book covers Univariate Box-Jenkins (UBJ) models which examine a single time series, using past data points to forecast future ones. These models were formalized by George Box and Gwilym Jenkins. They are also commonly called ARIMA (autoregressive integrated moving average) models.

The UBJ models are applied best to data series which are:

* Discrete
* Have a sample size of at least 50 observations
* Are stationary (have a constant mean)

The models can help us understand relationships in the historical series. They can also be used for forecasting. When forecasting, they are best at producing short-term forecasts. Many models simply use the past two or three observations. Making long-term forecasts in this way would be very risky and prone to error.

The procedure for developing these models (the Box-Jenkins modeling procedure) consists of:

1. Identification – Choose one or more ARIMA models as candidates
2. Estimation – Estimate the parameters of the models
3. Diagnostic Checking – Evaluate the models adequacy
4. Forecasting – Apply the model to make out-of-sample forecasts

UBJ models are considered by many to be superior to other models because they have a solid foundation in probability theory and statistics. They can also be shown to produce optimal univariate forecasts which are superior to all other univariate forecast models (based on mean-squared forecast error). The optimality of the models is discussed further in Chapter 10.

**CHAPTER 2: Introduction to Box-Jenkins Analysis**

*Summary Statistics and Autocorrelation Functions*

A time series sample is represented as the set . If the underlying distribution from which the sample comes is stationary then it has constant mean, variance, and autocovariances (variances between members of the time series separated by a fixed number of samples). While we never know the theoretical values for these properties, we can form estimates. We distinguish between the theoretical (unobservable) properties of the times series and our estimates with the following symbols:

|  |  |  |
| --- | --- | --- |
| Property | Theoretical | Estimate |
| Mean |  |  |
| Variance |  |  |
| Autocovariance |  |  |
| Autocorrelation |  |  |

The sample mean is found with the usual formula:

|  |  |
| --- | --- |
|  | Sample Mean |

This is our best estimate of the true mean. Since the mean of a stationary series is constant, it is often useful to remove the mean by centering our samples to get:

|  |  |
| --- | --- |
|  | Centered Samples |

The mean of the centered series is guaranteed to be 0. We then compute the variance as:

|  |  |
| --- | --- |
|  | Sample Variance |

NOTE: While this can be estimated in other ways (e.g. dividing by T-1 instead of T), there is a good theoretical reason for doing it this way. Unfortunately, that reason hasn’t been explained to us yet.

The autocovariance between lagged samples is estimated as:

|  |  |
| --- | --- |
|  | Autocovariances |

Note that and .

The autocovariances can then be scaled to give autocorrelations:

|  |  |
| --- | --- |
|  | Autocorrelations |

The summary statistics given above along, especially the estimated autocorrelation function (ACF), are useful in describing a times series. We will find that a plot of the ACF’s is often used to analyze and identify different types of time series.

*Partial Autocorrelation Function*

The autocorrelation function examines correlations between equally spaced samples in the time series. It does not take into account any of the samples in between. Mathematically, this is simply:

A better picture might emerge if we consider the effect of the realized samples between and . That is, we could try to find:

This is estimated by the partial autocorrelation function. This is typically found by doing a linear regression of against the prior terms:

The coefficient then indicates the degree to which contributes to our estimate of , and it is called the partial autocorrelation coefficient. A plot of these values provides an estimate for the partial autocorrelation function (PACF).

This process requires us to do a new linear regression of *k* terms each time we calculate . Other methods that are less accurate (but computationally easier) are also given. These involve recursive formulas where we begin with the estimate , and then keep this fixed while calculating successive values of . In the general method described earlier, we do not assume that the coefficients are fixed after they are calculated and instead perform a new linear regression each time.

**CHAPTER 3: Underlying Statistical Principles**

*Theoretical Distributions and Stationary Time Series*

In theory, a time series is a sample from some underlying probability distribution. The distribution specifies the relationship between all values in the time series and is thus a joint probability distribution over all terms in the series. This is a very inconvenient way to describe the process, especially for time large time series that may contain an infinite number of values. Thus, instead of searching for some theoretical joint distribution underlying our data, we describe properties of that distribution. We begin by assuming that the time series is stationary, meaning that the marginal distribution over any subset of the time series is the same no matter where we are in the time series. In practical terms this means that:

1. The first order marginal distribution, , describing a sample at any time is constant.

 This means that the mean and variance of the distribution are constant for all samples.

1. The second order marginal distributions, , describing any pair of samples separated by a constant lag, *k*, is constant.
 This means that the covariances (and correlations) between samples separated by a constant lag are constant for all samples.

If the process is assumed to have constant marginal distributions (at all orders) for all subsets of the time series, we say the process is **strictly stationary**. In practice, we never work with this assumption, and instead only require that its primary implications – that of a constant mean, variance, covariances (and correlations) – are constant. If the process displays these properties we call it **weakly stationary**. In practice, “stationary” means “weakly stationary.”

*Identifying Simple Models*

The two simplest ARIMA models are:

|  |  |
| --- | --- |
|  | AR(1) Process |
|  |  |
|  | MA(1) Process |

 indicates a random shock to the system (typically some normally distributed “white noise”). The details of AR, MA, ARMA, and ARIMA models will be discussed later, but what is important to note now is that each of these models has a particular signature which can be identified in plots of the ACF and PACF functions for samples of these models.

Theoretical plots for the AR(1) process are shown below:



Note that the AR(1) process is easily identified by its PACF plot. This shows that a linear regression of samples on the previous values completely explains the sampled data, and that values before this are not helpful in explaining our observations. This is exactly what we would expect, knowing that the AR(1) process is indeed a linear function containing only the previous sample. The ACF for this process consistently (and quickly) drops to zero. We will discuss the importance of this later.

Theoretical plots for the MA(1) Process are now shown:



Notice that this is exactly the opposite. We see a single spike in the ACF plot at lag 1 and quickly falling values in the PACF plot. The reasons for this are less obvious from the process definition given above, but they will be discussed later on.

The important point is that if we plot the estimated ACF and PACF for a time series sample and observe a structure similar to those above, we might suspect that the samples were generated from an underlying process described by either AR(1) or MA(1).

*Statistical Inference*

It is important to note that even if we generate artificial data from a process which is perfectly specified by these equations, the summary statistics we generate for the sample will not perfectly match those of the generating process. Just like the sample mean and variance will not match the actual mean and variance of a distribution, the autocorrelations and partial autocorrelations will also not be exact matches. They should be close to the actual values, especially if we have a large sample size, but they cannot be expected to match the theoretical values exactly. Instead, we will observe some noise in our graphs. In the ACF and PACF plots this will show up as small correlations that in fact do not exist in the underlying process. As such, it is useful to develop statistical tests to see if a particular correlation estimate is statistically meaningful.

It has been shown that if the true autocorrelation, , equals 0, then the sampled autocorrelations will be approximately normally distributed given a large enough sampling (large, being about 100 samples). Bartlett provided a ormula estimated the standard error of this distribution as:

We would thus expect autocorrelations within the range to be generated even by processes were the true autocorrelations were zero. If we observe autocorrelations outside of this range, we would be able to reject the hypothesis that at a 95% confidence level and infer that a true autocorrelation does exist (although we could not say exactly what it was).

In general, we generate the test statistic:

For large samples this is approximately normally distributed (although we still call it a *t* statistic instead of a *z* statistic). We would then use the properties of the normal distribution to investigate properties of our statistic.

For partial autocorrelations the distribution is again approximately normal. The corresponding standard error and test statistics are:

|  |  |
| --- | --- |
|  |  |

**CHAPTER 4: Introduction to ARIMA Modeling**

In the previous chapter we saw that the difficult in describing an underlying probability distribution for a given stochastic process does not keep us from building some model to describe the process. We do this by searching for models that display the same characteristics as the observed time series, especially as this pertains to the mean, variance, autocovariances, and autocorrelations. In general a good ARIMA model has the following properties:

1. It is parsimonious (uses the smallest number of coefficients to explain the data)
2. It is stationary (the AR coefficients satisfy specific mathematical inequalities)
3. It is invertible (the MA coefficients satisfy specific mathematical inequalities)
4. It has estimated coefficients (’s and ’s) of high quality
	1. Absolute t-values about 2.0 or larger
	2. ’s and ’s are not too highly correlated
5. It has uncorrelated residuals
6. It fits the available data (the past) well enough to satisfy the analyst
	1. Root-mean-squared error (RMSE) is acceptable
	2. Mean absolute percent error (MAPE) is acceptable
7. It forecasts the future satisfactorily

Properties 2&3 are discussed in chapter 6. The fourth property is discussed in chapter 8, and the fifth in chapter 9.

Two samples are also discussed in the text where we identify an AR(1) model and then an MA(1) model and estimate the coefficients. T-statistics for the estimated coefficients show that they are high quality and are not likely to have occurred by chance. The author then takes the residuals from the model’s predictions (the implied values) and produces an ACF plot for them (a residual ACF). No statistically significant correlations are found, leading us to believe that we have adequately described the data.

**CHAPTER 5: Notation and Interpretation of ARIMA Models**

*Autoregressive and Moving Average Models*

An autoregressive model describes a particular sample as a linear function of past samples. The number of samples used in the model, *p*, is the order of the autoregressive model. We thus have the definition of an AR(*p*) process:

|  |  |
| --- | --- |
|  | AR(*p*) Model |

A moving average model describes a sample as a linear function of sequential random shocks, :

|  |  |
| --- | --- |
|  | MA(*q*) Model |

The minus signs in this expression are a convention, the reason for which will become clear in the next section when we formalize our notation. We can also combine these models to get:

|  |  |
| --- | --- |
|  | ARMA(*p,q*) Model |

This type of model is called an autoregressive moving average model, or ARMA. Notice that the total number of coefficients are , and that all AR models and MA models are subsets of this model which can be obtained by setting the unnecessary coefficients to zero.

Finally, we have noted that we can often make a non-stationary series stationary by taking its differences. When we do this we say that we have formed an “integrated” series. If we apply differences to a series before modeling it with an ARMA model, we have an ARIMA (autoregressive integrated moving average) model. Models of this form are described as ARIMA(*p,d,q*) models where *p* and *q* have the same meanings as before, and *d* indicates the number of times the series was differenced before the ARMA model was applied.

TODO: describe models using centered values and show the difference in coefficients (page 101)

*Backshift Notation*

When relating time series samples to past samples, it can be convenient to use the backshift operator, *B*, which provides us with the sample one observation prior to that upon which it operates:

|  |  |
| --- | --- |
|  | Backshift Operator |

Applying it successively shifts the samples further such that:

This allows us to describe an AR series as:

More commonly, we move all the terms involving to one side of the equation to get:

|  |  |
| --- | --- |
|  | AR(*p*) Model |

Similarly, we can write a MA(*q*) process as:

|  |  |
| --- | --- |
|  | MA(*q*) Model |

To simplify things further, we can represent the terms involving powers of the backshift operator with a single symbol, such that:

This produces:

|  |  |
| --- | --- |
|  | AR(*p*) Model |
|  |  |
|  | MA(*q*) Model |

We can also combine these two into a deceptively simple equation:

|  |  |
| --- | --- |
|  | ARMA(*p,q*) Model |

*Difference Operator*

We can also use the backshift operator to provide differences in a time series:

This is done frequently enough, that it has its own operator: the difference operator, defined as:

|  |  |
| --- | --- |
|  | Difference Operator |

Note that applying the difference operator repeatedly produces the following algebraic expansion:

The algebra obscured what is really going on. Instead, we should just imagine applying the difference operator once to a series to obtain the differences. If we apply it again to the new series, it will give us the differences of the original differences, and so on.

As mentioned, differencing is a common way to transform a non-stationary series into a stationary one. Mathematically, this appears very compactly as . This will produce a series containing the *d*-th differences of the original series. In ARIMA terms this means we have an integrated series of order *d*.

If we difference a time series and then apply an ARMA model, we can describe an ARIMA model by making a simple addition to the ARMA formula to get:

|  |  |
| --- | --- |
|  | ARIMA(*p,d,q*) Model |

*Interpreting ARIMA Models*

AR models are the easiest to understand intuitively. They simply represent forecasts based upon previous samples in the series. This is similar to extrapolating future values from past values by discovering some relationship in the data. The MA models are a little more difficult, but it can be proved that MA models can be represented as AR models using an infinite number of terms. For an MA(1) process we have:

Dividing by and noting that is the sum of a geometric series with common ratio , we find:

Thus the errors in our series are weighted averages of all past samples using weights:

Similarly, we can rewrite an AR(1) process as an MA process with infinite terms:

Again the terms are in a geometric progression with common ratio , giving:

This is the reason that the PACF graph of an MA(1) process shows correlations that quickly die down in a manner similar to a geometric progression and the ACF graph of a AR(1) process shows the same pattern.

Some models also have intuitive explanations. The process:

is a random walk.

An exponentially weighted moving average (EWMA) can also be given as an ARIMA(0,1,1) model, or an MA(1) model applied to the first difference of the series. (TODO: page 110, where this is elaborated).

A system that where that preserves the effects of random shocks, with the lingering effect of each shock varying as time passes would inspire an MA process. If the number of past shocks included in our model (*q*) is large and effects decay geometrically, we might find that an AR(1) process actually serves as a better model since it is exactly equivalent to this type of MA process with a large *q*. In fact, it may even work better since it is more parsimonious and has fewer coefficients to calculate.

**CHAPTER 6: Identifying Stationary (and Invertible) Models**

The chapter begins by showing sample ACFs and PACFs for common ARMA models (AR(1), AR(2), MA(1), MA(2), ARMA(1,1)). These show what you would expect:

* AR(p) models have a PACF with p spikes that fall immediately to zero after lag p. The ACF decays quickly to zero as an exponential or damped sinusoid.
* MA(q) models have an ACF with q spikes that fall immediately to zero after lag q. The PACF decays quickly to zero as an exponential or damped sinusoid.
* ARMA(1,1) models are mixed. Both the ACF and PACF will have decaying spikes.

Once we have identified a possible model and estimated its coefficients, we then have to check that it satisfies certain properties we expect from our model. Two of the most important of these are stationarity and invertibility.

*Stationarity*

It turns out, there is a very interesting way to test if an ARMA model is stationary. If we examine the AR coefficients, writing them using the backshift polynomial:

we can treat B as a variable (which it isn’t) and find the roots of the polynomial. If the roots have a magnitude greater than 1, the process will be stationary. It should be noted that the roots may be imaginary numbers. In this case the requirement for stationarity is the same: their magnitude (or modulus) must be greater than 1. Sometimes this is read as: “The characteristic polynomial of the AR coefficients must have roots lying outside the unit circle.”

For an AR(1) process we have:

The root of this polynomial is . The rule > 1 leads to the stationarity requirement that . Notice that the comparison sign has switched: the root must be greater than 1, but the coefficient must be less than 1.

For an AR(2) process, the stationarity requirements are a little more complicated. The result is

|  |  |
| --- | --- |
|  |  |

Similar checks can be done for AR processes with more coefficients. Note also that MA(*q*) processes are always stationary for any number of coefficients. These processes do not have AR coefficients, and it is only the AR polynomial that is tested to determine if a process is stationary.

In general, we will require that any estimated coefficients for our process satisfy this condition so that they produce a stationary process. If they do not, we will be unable to estimate the mean or variance, and thus will not be able to estimate the AR coefficients either.

*Invertibility*

**Invertibility** is a mathematical term referring to certain properties of the AR and MA coefficients. It turns out that the coefficients can be inverted so that AR coefficients can be expressed as MA coefficients and vice versa. An AR(1) process can be written as an MA process with infinitely many coefficients. It can be shown that the MA coefficients follow a geometric progression, and this is the reason that the ACF and PACF plots for an AR(1) process show a single spike in the PACF and an infinite, geometric decay in the ACF. These graphs are showing us two different representations of the same process. Similarly, an MA(1) process can be written as an AR process within infinitely many coefficients, all decaying geometrically.

Invertibility tell us that if we invert the MA coefficients and write them as AR coefficients, we will have coefficients that decay towards zero instead of exploding towards infinity. This decay is important intuitively, because it tells us that recent samples are more important than samples from the distant past (what happened yesterday is more important than what happened a million years ago, and what happened a million years ago is more important than what happened a trillion years ago). We would thus reject any process which did not display this characteristic as it would not really make sense. There is also a theorem that says an invertible process will have a one-to-one correspondence between its ACF/PACF and its AR/MA coefficients. If the process were not invertible, we could not reliably identify the coefficients.

It turns out the invertibility test is similar to the stationary test except we examine the MA coefficients. Treating the backshift operator, *B*, as a variable and finding the roots of the polynomial, we must have all roots lie outside the unit circle for the process to be invertible. An MA(1) process thus has the requirement that:

An MA(2) process has the same requirements as an AR(2) process if we replace the ’s with ’s.

*Theoretical ACFs and PACFs*

The chapter then continues to describe how we can calculate theoretical ACFs and PACFs for a process with known coefficients. I’m not sure how important this is in practice though.

**CHAPTER 7: Identification: Nonstationary Models**

*Homogeneous Nonstationarity*

We have already mentioned that a stationary process has constant mean and variance. If we examine a graph and have reason to suspect this is not true, we will want to transform the data in order to obtain a stationary series which we can model. In the simplest case, we will have **homogeneous nonstationarity**: this means that the series behaves the same way and has similar properties at all times. Informally, if we draw a rectangle around different parts of the time series, the graphs inside the rectangles should be similar. When this is the case, we can difference the series (1 or more times) and obtain a stationary series with constant mean and variance. The most important thing to look for is a constant variance. When this is present, we can model the mean separately produce a stationary series.

*Nonstationary Variance*

If the variance is not constant, we will have to transform it in a way to make it constant. The **logarithmic transformation** is often a useful tool in doing this. If we take the logarithm of a series, the first differences will then be:

The result is the same as taking the logarithm of percentage changes in the series. For many processes, it will make intuitive sense for changes to be proportionate to the current value. Changes in population are common example. As the values in the raw time series increase, the series will become more and more volatile as small percentage changes have large effects in the raw numbers. Taking the logarithm and then the first difference can then be a way to make the volatility stationary over time.

Other methods to induce a stationary variance include taking the square root (or some other power). The Box-Cox transformation is also a way to estimate an appropriate transformation to induce stationary variance. The details of this method are not discussed in the text.

*Deterministic Trends*

When data is differenced it will often have a mean near zero. In this case we should estimate the process as though it had a mean and then test to see whether that mean is significant. If not, it may be appropriate to simply assume a mean of zero.

If the differenced data has a zero mean, the integrated (original) series will be a random walk. If the differenced data has a single, constant mean, the integrated series will be a random walk with a linear trend or drift. Other trends may also exist.

**CHAPTER 8: Estimation**

TODO (No need to do this manually. Use R or SAS!)

**CHAPTER 9: Diagnostic Checking**

*Randomly Distributed Residuals*

A good model captures all of the identifiable structure in our data set. Given a set of observations we can estimate the random shock () or residuals left after fitting the data. If we have captured all of the structure in the data these residuals should be independent, random variables. They should have no structure, no autocorrelation, and should essentially just be “white noise.”

To simplest test for white noise is to calculate the autocorrelations of the residuals (the residual ACF) and ensure that there are no significant correlations. This is done in the same manner as we would do for our original data. We can also use the portmanteau chi-squared statistic (or modified Box-Ljung statistic) to test all of the autocorrelations together. This test has a null hypothesis that the residuals are white noise with:

We then compute the Box-Ljung statistic:

This should follow a Chi-squared distribution with degrees of freedom, where *m* is the number of parameters used in the model (). Low values indicate that the autocorrelations are significant, whereas high values indicate that the autocorrelations could very well have been caused by a white noise process.

In short, we can analyze the residual time series just like we did the original. If our original model is good, we should not be able to fit any model to the residuals that is statistically significant.

*Checking for Constant Model Parameters*

It should also be noted that we are assuming the model parameters are constant over time. If this is not the case Box-Jenkins analysis simply will not work and alternate analysis methods will be required (State space modeling or Bayesian time series analysis). A quick way to check for model consistency is to break the time series into pieces and model each piece separately. If the estimated models and parameters are roughly the same for each separate time series piece we can assume that the model is acceptable. Pankratz specifically suggest breaking the model in half or dropping the last part of the time series (the last 10% of observations) when fitting a new model.

*Reformulating a Model*

TODO: page 233

**CHAPTER 10: Forecasting**

*Point Forecasts*

The ultimate goal of developing a model is typically to forecast future values. Towards this end we will use the notation:

 is the forecast made at time *t* for future points l periods ahead. That is: we are estimating from time *t*, and of course, we can only use information available at time *t* (which we represent as ).

NOTE: This notation is more useful than what I was using (simply ) because our estimate for time can be different if made at different points in time.

When making a forecast, we note that any ARMA model can easily be rewritten to solve for . If we have:

we will rewrite it as:

We then shift the subscripts to produce an expression for :

We note that we have values for which lie in our sample set (). While we do not have values for , we can estimate them using residuals. We thus have actual values to use for these variables, and when we take an expectation for we will use these actual values when available. If the required values lie outside of our sample set (in the future), we have which allows us to ignore future MA terms when making predictions. Our expectations for outside of the sample set have to be bootstrapped, meaning we first calculate and then use it to calculate , and so on. In general, we thus have:

Continuing the example, we would then have:

Notice that the MA terms drop out when predicting values 2 or more periods in the future. For a simple MA model this means that the prediction at any time beyond t+2 will simply be 0 for centered data or the series mean for uncentered data.

*Forecast Intervals*

If we want to expand our point forecast to include error intervals, a nice analytical solution can be found to do this. We begin by rewriting our equation in **random-shock** form (or psi-form) coefficients:

This is very easy to do if the model is an MA model. If we have AR terms, we note that we take the general ARIMA model:

and substitute the psi-weight definition of :

to get:

If we expand and match like powers of B with those on the right-hand side in , we will have a system of equations which can be solved for the psi coefficients. Typically this is an iterative process in which we can easily find and use this to find and so on. The equations don’t need to be solved simultaneously, and often a simple pattern will emerge in the psi-weight coefficients. We will remember that writing an AR(1) processes in psi-weight form requires infinitely many coefficients. If the process is invertible the coefficients will quickly die down to zero and we can form an approximate representation of the process by including enough psi-weight coefficients. This entire process is typically done by a computer. The important part is that we ultimately can express our model in psi-weight form.

The psi-weight representation is useful for calculating forecast error because we now have the forecasts written as a linear combination of random variables. The lying within the sample set can be estimated so that only those lying in the future remain random. The forecast error in our predictions is:

and our estimate of this error is then:

This is a linear combination of random variables, each of which has zero mean. The variance is then:

We can thus estimate the variance and standard deviation around our forecast values and apply confidence intervals by multiplying by the appropriate z-value. If the are drawn from a stable distribution we will also be able to infer that the same distribution applies to our forecasts.

*Forecasting Data in Logarithmic Form*

If we have transformed our data by taking a logarithm:

we can still forecast values in the transformed space with standard error estimates . However, we cannot transform directly back using the inverse transformation. Instead, we must use:

If we have calculated upper and lower confidence limits U’ and L’ in the transformed space, we can however use the inverse transform directly to get:

*Optimality of ARIMA Forecasts*

It is also noted that ARIMA forecast are optimal in the sense that no other linear combination of historical samples will provide smaller forecast error as measured by . This comes with a warning that better forecast methods may exist if we transform the historical values in some way or construct nonlinear models or if model parameters are changing over time. This presents interesting ideas of how we might try to build different models to improve forecasts.

**CHAPTER 11: Seasonal and Periodic Models**

*Purely Seasonal Models*

We have thus far assumed that time series samples are mostly influenced by the previous 1 or 2 samples. However, we may find that they are influenced by samples at even larger lags. With monthly sales data, we may find that July sales are more influenced by last year’s July sales than they are by June or May sales of the same year. The process may even continue back to previous July’s so that each month essentially requires its own time series model. If we make the simplifying assumption that the models for each month have the same essential structure (albeit with different levels and sample data) we can use a Box-Cox seasonal model for the data.

In this case we would only want to use coefficients that were multiples of the desired seasonal lag. For a seasonal lag *s*, we would have P autoregressive coefficients (at lags *s*, 2*s*, .., P*s*) and Q moving average coefficients (again at lags *s*, 2*s*, …, *Qs*):

Note that we are using capital P and Q for the order of the polynomials and capital phi’s and theta’s for the coefficients. As before, we can simplify this using polynomial notation:

This is a purely seasonal model in that it only has seasonal coefficients. The ACF and PACF for such a model would look just like the regular ARMA plots, except that the spikes would only show up at the seasonal lags and all intervening lags will have zero correlations. Thus if we had an AR(1) process in the seasonal model (with one coefficient, ) we would see a spike at lag s in the PACF and no others. In the ACF we would see a similar spike at lag *s*, with the remaining spikes at multiples of *s* dying down geometrically.

As with non-seasonal models, if the ACF spikes fail to die down quickly we have evidence of a nonstationary model at the seasonal lags and we would need to difference the data. The appropriate difference to apply would be at the lag *s*, and this is done frequently enough that we have the simplifying operators:

and:

where the latter applies the seasonal differencing *D* times. The general description of a purely seasonal ARMA model is thus:

This model is called an ARIMA(*P,D,Q*)s in the same manner as the nonseasonal models, with the subscript *s* added to indicate seasonality.

*Seasonal-Nonseasonal Multiplicative Models*

It is possible to combine seasonal and nonseasonal models, and this will allow the data to depend both on samples immediately prior to it as well as seasonal samples at longer lags. There are two ways to build these models: as multiplicative or additive models. Box and Jenkins suggest starting with multiplicative models, so that is where we will begin.

In a multiplicative model we simply multiply the nonseasonal model terms with the seasonal ones. That is, we combine:

 (the nonseasonal model), and:

 (the seasonal model)

to get:

The nonseasonal model is an ARIMA(p,d,q) while the seasonal is an ARIMA(*P,D,Q*)s. Combining these notations, we express the multiplicative model as ARIMA(*p,d,q*)(*P,D,Q*)s.

Multiplying the seasonal and nonseasonal characteristic polynomials causes the models to intermingle in sometimes complicated ways. As an example, we expand the polynomials:

Notice that in addition to the coefficients at lags 1 and *s* we have a new coefficient at lag *s* + 1. If we compute the theoretical ACF for this model we will find that it has the expected lags at spikes 1 and *s*, but that the spike at lag *s* also comes with companion spikes at lags *s* - 1 and and *s* + 1. These companion spikes are smaller than the primary spike and should be about the same size as each other.

*Seasonal-Nonseasonal Additive Models*

We noted before that the polynomial expansion

Introduces an extra coefficient . In the multiplicative model we have only two parameters and their values will determine the value of this third coefficient. However, we may be able to find a better fit if we allow this parameter to be free and instead simply fit:

This is now an additive model since we simply add in all the coefficients that we think are important to capture the seasonal and nonseaonal relationships.

Because we have more free parameters in the additive model it is guaranteed to produce a better fit than the multiplicative model. However, it will be up to us to determine if the extra parameters are truly necessary or if they are adding unneeded complexity.