Forecasting Princples and Practice

Koji Mizumura

August 10, 2018

Table of Contents

# Resources

[Forecasting: princples and practice](https://otexts.org/fpp2/)  
[Slides](https://robjhyndman.com/seminars/uwa2017/)  
[Github](https://github.com/nealxun/ForecastingPrinciplePractices)

From the Github, we can install the package fpp2.

install.packages("fpp2",dependencies = TRUE)

# Chapter 1: Introduction

<https://github.com/nealxun/ForecastingPrinciplePractices/blob/master/01-intro.Rmd>

## 1.1 What can be forecast?

Forecasting is required in many situations: deciding whether to build another power generation plant in the next five years requires forecasts of future demand; scheduling staff in a call centre next week requires forecasts of call volumes; stocking an inventory requires forecasts of stock requirements. Forecasts can be required several years in advance (for the case of capital investments), or only a few minutes beforehand (for telecommunication routing). Whatever the circumstances or time horizons involved, forecasting is an important aid to effective and efficient planning.

Some things are easier to forecast than others. The time of the sunrise tomorrow morning can be forecast precisely. On the other hand, tomorrow’s lotto numbers cannot be forecast with any accuracy. The predictability of an event or a quantity depends on several factors including:

how well we understand the factors that contribute to it; how much data are available; whether the forecasts can affect the thing we are trying to forecast. For example, forecasts of electricity demand can be highly accurate because all three conditions are usually satisfied. We have a good idea of the contributing factors: electricity demand is driven largely by temperatures, with smaller effects for calendar variation such as holidays, and economic conditions. Provided there is a sufficient history of data on electricity demand and weather conditions, and we have the skills to develop a good model linking electricity demand and the key driver variables, the forecasts can be remarkably accurate.

On the other hand, when forecasting currency exchange rates, only one of the conditions is satisfied: there is plenty of available data. However, we have a limited understanding of the factors that affect exchange rates, and forecasts of the exchange rate have a direct effect on the rates themselves. If there are well-publicised forecasts that the exchange rate will increase, then people will immediately adjust the price they are willing to pay and so the forecasts are self-fulfilling. In a sense, the exchange rates become their own forecasts. This is an example of the “efficient market hypothesis”. Consequently, forecasting whether the exchange rate will rise or fall tomorrow is about as predictable as forecasting whether a tossed coin will come down as a head or a tail. In both situations, you will be correct about 50% of the time, whatever you forecast. In situations like this, forecasters need to be aware of their own limitations, and not claim more than is possible.

Often in forecasting, a key step is knowing when something can be forecast accurately, and when forecasts will be no better than tossing a coin. Good forecasts capture the genuine patterns and relationships which exist in the historical data, but do not replicate past events that will not occur again. In this book, we will learn how to tell the difference between a random fluctuation in the past data that should be ignored, and a genuine pattern that should be modelled and extrapolated.

Many people wrongly assume that forecasts are not possible in a changing environment. Every environment is changing, and a good forecasting model captures the way in which things are changing. Forecasts rarely assume that the environment is unchanging. What is normally assumed is that the way in which the environment is changing will continue into the future. That is, a highly volatile environment will continue to be highly volatile; a business with fluctuating sales will continue to have fluctuating sales; and an economy that has gone through booms and busts will continue to go through booms and busts. A forecasting model is intended to capture the way things move, not just where things are. As Abraham Lincoln said, “If we could first know where we are and whither we are tending, we could better judge what to do and how to do it”.

Forecasting situations vary widely in their time horizons, factors determining actual outcomes, types of data patterns, and many other aspects. Forecasting methods can be simple, such as using the most recent observation as a forecast (which is called the naïve method), or highly complex, such as neural nets and econometric systems of simultaneous equations. Sometimes, there will be no data available at all. For example, we may wish to forecast the sales of a new product in its first year, but there are obviously no data to work with. In situations like this, we use judgmental forecasting, discussed in Chapter 4. The choice of method depends on what data are available and the predictability of the quantity to be forecast.

## 1.2 Forecasting, planning and goals

Forecasting is a common statistical task in business, where it helps to inform decisions about the scheduling of production, transportation and personnel, and provides a guide to long-term strategic planning. However, business forecasting is often done poorly, and is frequently confused with planning and goals. They are three different things.

### *Forecasting*

is about predicting the future as accurately as possible, given all of the information available, including historical data and knowledge of any future events that might impact the forecasts.

### *Goals*

are what you would like to have happen. Goals should be linked to forecasts and plans, but this does not always occur. Too often, goals are set without any plan for how to achieve them, and no forecasts for whether they are realistic.

### *Planning*

is a response to forecasts and goals. Planning involves determining the appropriate actions that are required to make your forecasts match your goals.

Forecasting should be an integral part of the decision-making activities of management, as it can play an important role in many areas of a company. Modern organisations require short-term, medium-term and long-term forecasts, depending on the specific applicatio

### *Short-term forecasts*

are needed for the scheduling of personnel, production and transportation. As part of the scheduling process, forecasts of demand are often also required.

### *Medium-term forecasts*

are needed to determine future resource requirements, in order to purchase raw materials, hire personnel, or buy machinery and equipment.

### *Long-term forecasts*

are used in strategic planning. Such decisions must take account of market opportunities, environmental factors and internal resources.

An organisation needs to develop a forecasting system that involves several approaches to predicting uncertain events. Such forecasting systems require the development of expertise in identifying forecasting problems, applying a range of forecasting methods, selecting appropriate methods for each problem, and evaluating and refining forecasting methods over time. It is also important to have strong organisational support for the use of formal forecasting methods if they are to be used successfully.

## 1.3 Determining what to forecast

In the early stages of a forecasting project, decisions need to be made about what should be forecast. For example, if forecasts are required for items in a manufacturing environment, it is necessary to ask whether forecasts are needed for:

1. every product line, or for groups of products?
2. every sales outlet, or for outlets grouped by region, or only for total sales?
3. weekly data, monthly data or annual data?

It is also necessary to consider the forecasting horizon. Will forecasts be required for one month in advance, for 6 months, or for ten years? Different types of models will be necessary, depending on what forecast horizon is most important.

How frequently are forecasts required? Forecasts that need to be produced frequently are better done using an automated system than with methods that require careful manual work.

It is worth spending time talking to the people who will use the forecasts to ensure that you understand their needs, and how the forecasts are to be used, before embarking on extensive work in producing the forecasts.

Once it has been determined what forecasts are required, it is then necessary to find or collect the data on which the forecasts will be based. The data required for forecasting may already exist. These days, a lot of data are recorded, and the forecaster’s task is often to identify where and how the required data are stored. The data may include sales records of a company, the historical demand for a product, or the unemployment rate for a geographic region. A large part of a forecaster’s time can be spent in locating and collating the available data prior to developing suitable forecasting methods.

## 1.4 Forecasting data and methods

The appropriate forecasting methods depend largely on what data are available.

If there are no data available, or if the data available are not relevant to the forecasts, then **qualitative forecasting** methods must be used. These methods are not purely guesswork—there are well-developed structured approaches to obtaining good forecasts without using historical data. These methods are discussed in Chapter 4.

**Quantitative forecasting** can be applied when two conditions are satisfied:

1. numerical information about the past is available;
2. it is reasonable to assume that some aspects of the past patterns will continue into the future. There is a wide range of quantitative forecasting methods, often developed within specific disciplines for specific purposes. Each method has its own properties, accuracies, and costs that must be considered when choosing a specific method.

Most quantitative prediction problems use either time series data (collected at regular intervals over time) or cross-sectional data (collected at a single point in time). In this book we are concerned with forecasting future data, and we concentrate on the time series domain.

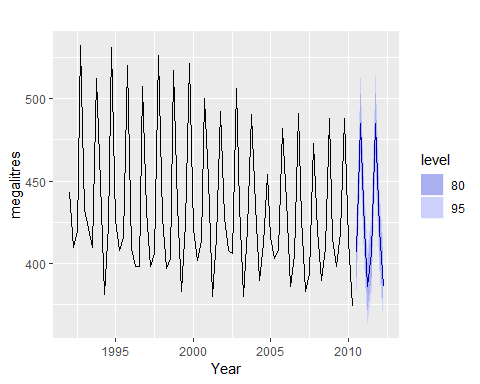
#### 1) Time series forecasting

Examples of time series data include:

* Daily IBM stock prices
* Monthly rainfall
* Quarterly sales results for Amazon
* Annual Google profits

Anything that is observed sequentially over time is a time series. In this book, we will only consider time series that are observed at regular intervals of time (e.g., hourly, daily, weekly, monthly, quarterly, annually). Irregularly spaced time series can also occur, but are beyond the scope of this book.

When forecasting time series data, the aim is to estimate how the sequence of observations will continue into the future. Figure 1.1 shows the quarterly Australian beer production from 1992 to the second quarter of 2010.



Australian quarterly beer production:1992Q1-2010Q2,with two years of forecasts.

The blue lines show forecasts for the next two years. Notice how the forecasts have captured the seasonal pattern seen in the historical data and replicated it for the next two years. The dark shaded region shows 80% prediction intervals. That is, each future value is expected to lie in the dark shaded region with a probability of 80%. The light shaded region shows 95% prediction intervals. These prediction intervals are a useful way of displaying the uncertainty in forecasts. In this case the forecasts are expected to be accurate, and hence the prediction intervals are quite narrow.

The simplest time series forecasting methods use only information on the variable to be forecast, and make no attempt to discover the factors that affect its behaviour. Therefore they will extrapolate trend and seasonal patterns, but they ignore all other information such as marketing initiatives, competitor activity, changes in economic conditions, and so on.

Time series models used for forecasting include decomposition models, exponential smoothing models and ARIMA models. These models are discussed in Chapters 6, 7 and 8, respectively.

#### 2) Predictor variables and time series forecastin

Predictor variables are often useful in time series forecasting. For example, suppose we wish to forecast the hourly electricity demand (ED) of a hot region during the summer period. A model with predictor variables might be of the form

The relationship is not exact — there will always be changes in electricity demand that cannot be accounted for by the predictor variables. The “error” term on the right allows for random variation and the effects of relevant variables that are not included in the model. We call this an **explanatory model** because it helps explain what causes the variation in electricity demand.

Because the electricity demand data form a time series, we could also use a time series model for forecasting. In this case, a suitable time series forecasting equation is of the form.

where, *t* is the present hour, *t-1* is the previous hour, *t-2* is two hours ago, and so on. Here, prediction of the future is based on past values of a variable, but not on external variables which may affect the system. Again, the “error” term on the right allows for random variation and the effects of relevant variables that are not included in the model.

There is also a third type of model which combines the features of the above two models. For example, it might be given by

These types of “mixed models” have been given various names in different disciplines. They are known as dynamic regression models, panel data models, longitudinal models, transfer function models, and linear system models (assuming that  
*f* is linear). These models are discussed in Chapter 9.

An explanatory model is useful because it incorporates information about other variables, rather than only historical values of the variable to be forecast. However, there are several reasons a forecaster might select a time series model rather than an explanatory or mixed model. First, the system may not be understood, and even if it was understood it may be extremely difficult to measure the relationships that are assumed to govern its behaviour. Second, it is necessary to know or forecast the future values of the various predictors in order to be able to forecast the variable of interest, and this may be too difficult. Third, the main concern may be only to predict what will happen, not to know why it happens. Finally, the time series model may give more accurate forecasts than an explanatory or mixed model.

The model to be used in forecasting depends on the resources and data available, the accuracy of the competing models, and the way in which the forecasting model is to be used.

## 1.6 The basic steps in a forecasting task

<https://otexts.org/fpp2/basic-steps.html>

A forecasting task usually involves five basic steps.

#### Step1: Problem definition

Often this is the most difficult part of forecasting. Defining the problem carefully requires an understanding of the way the forecasts will be used, who requires the forecasts, and how the forecasting function fits within the organisation requiring the forecasts. A forecaster needs to spend time talking to everyone who will be involved in collecting data, maintaining databases, and using the forecasts for future planning.

#### Step2: Gather information

There are always at least two kinds of information required: (a) statistical data, and (b) the accumulated expertise of the people who collect the data and use the forecasts. Often, it will be difficult to obtain enough historical data to be able to fit a good statistical model. In that case, the judgmental forecasting methods of Chapter 4 can be used. Occasionally, old data will be less useful due to structural changes in the system being forecast; then we may choose to use only the most recent data. However, remember that good statistical models will handle evolutionary changes in the system; don’t throw away good data unnecessarily.

#### Step3: Preliminary (exploratory) analysis

Always start by graphing the data. Are there consistent patterns? Is there a significant trend? Is seasonality important? Is there evidence of the presence of business cycles? Are there any outliers in the data that need to be explained by those with expert knowledge? How strong are the relationships among the variables available for analysis? Various tools have been developed to help with this analysis. These are discussed in Chapters 2 and 6.

#### Step4: Choosing and fitting models

The best model to use depends on the availability of historical data, the strength of relationships between the forecast variable and any explanatory variables, and the way in which the forecasts are to be used. It is common to compare two or three potential models. Each model is itself an artificial construct that is based on a set of assumptions (explicit and implicit) and usually involves one or more parameters which must be estimated using the known historical data. We will discuss regression models [Chapter 5](https://otexts.org/fpp2/regression.html#regression), exponential smoothing methods [Chapter 7](https://otexts.org/fpp2/expsmooth.html#expsmooth), Box-Jenkins ARIMA models [Chapter 8](https://otexts.org/fpp2/arima.html#arima),Dynamic regression models[Chapter 9](https://otexts.org/fpp2/dynamic.html#dynamic), Hierarchical forecasting [Chapter 10](https://otexts.org/fpp2/hierarchical.html#hierarchical), and several advanced methods including neural networks and vector autoregression in [Chapter 11](https://otexts.org/fpp2/advanced.html#advanced).

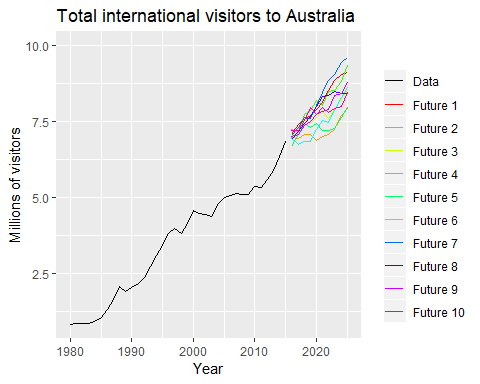
#### Step5: Using an evaluating a forecasting model

Once a model has been selected and its parameters estimated, the model is used to make forecasts. The performance of the model can only be properly evaluated after the data for the forecast period have become available. A number of methods have been developed to help in assessing the accuracy of forecasts. There are also organisational issues in using and acting on the forecasts. A brief discussion of some of these issues is given in Chapter 3. When using a forecasting model in practice, numerous practical issues arise such as how to handle missing values and outliers, or how to deal with short time series. These are discussed in Chapter 12.

## 1.7 The statistical forecasting perspective

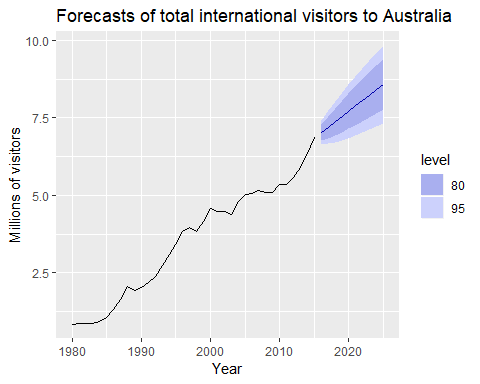
The thing we are trying to forecast is unknown (or we wouldn’t be forecasting it), and so we can think of it as a *random variable*. For example, the total sales for next month could take a range of possible values, and until we add up the actual sales at the end of the month, we don’t know what the value will be. So until we know the sales for next month, it is a random quantity.

Because next month is relatively close, we usually have a good idea what the likely sales values could be. On the other hand, if we are forecasting the sales for the same month next year, the possible values it could take are much more variable. In most forecasting situations, the variation associated with the thing we are forecasting will shrink as the event approaches. In other words, the further ahead we forecast, the more uncertain we are.

We can imagine many possible futures, each yielding a different value for the thing we wish to forecast. Plotted in black in Figure 1.2 are the total international visitors to Australia from 1980 to 2015. Also shown are ten possible futures from 2016–2025. 

When we obtain a forecast, we are estimating the *middle* of the range of possible values the random variable could take. Often, a forecast is accompanied by a **prediction interval** giving a range of values the random variable could take with relatively high probability. For example, a 95% prediction interval contains a range of values which should include the actual future value with probability 95%.

Instead of plotting individual possible futures as shown in Figure 1.2, we usually show these prediction intervals instead. The plot below shows 80% and 95% intervals for the future Australian international visitors. The blue line is the average of the possible future values, which we call the point forecasts.



Total international visitors to Australia (1980-2015) along with 10-year forecasts and 80% and 95% prediction intervanls.

We will use the subscript for time. For example, will denote the observation at time . Suppose we denote all the information we have observed as ${\cal I}$ and we want to forecast . We then write $y\_{t} | {\cal I}$ meaning “the random variable given what we know in ${\cal I}$”. The set of values that this random variable could take, along with their relative probabilities, is known as the “probability distribution” of $y\_{t} |{\cal I}$. In forecasting, we call this the “forecast distribution”.

When we talk about the “forecast”, we usually mean the average value of the forecast distribution, and we put a “hat” over to show this. Thus, we write the forecast of as , meaning the average of the possible values that could take given everything we know. Occasionally, we will use to refer to the *median* (or middle value) of the forecast distribution instead.

It is often useful to specify exactly what information we have used in calculating the forecast. Then we will write, for example, to mean the forecast of taking account of all previous observations . Similarly, means the forecast of taking account of (i.e., an -step forecast taking account of all observations up to time ).

|  |
| --- |
| # Chapter 2: Time series graphics |
| The first thing to do in any data analysis task is to plot the data. Graphs enable many features of the data to be visualised, including patterns, unusual observations, changes over time, and relationships between variables. The features that are seen in plots of the data must then be incorporated, as much as possible, into the forecasting methods to be used. Just as the type of data determines what forecasting method to use, it also determines what graphs are appropriate. But before we produce graphs, we need to set up our time series in R. |
| ## 2.1 ts objects A time series can be thought of as a list of numbers, along with some information about what times those numbers were recorded. This information can be stored as ts object in R. |
| Suppose you have annual observations for the last few years: |
| We turn this into a ts object using the ts() function: |
| ## Time Series: ## Start = 2012 ## End = 2016 ## Frequency = 1 ## [1] 123 39 78 52 110 |
|  |
| If you have annual data, with one observation per year, you only need to provide the starting year (or the ending year). |
| For observations that are more frequent than once per year, you simply add a frequency argument. For example, if your monthly data is already stored as a numerical vector z, then it can be converted to a ts object like this: |
| r y <- ts(z,start=2003,frequency=12) y |
| ## Jan Feb Mar Apr May ## 2003 123 39 78 52 110 |
| Almost all of the data used in this book is already stored as ts objects. But if you want to work with your own data, you will need to use the ts() function before proceeding with the analysis. |
| ### Frequency of a time series{-} |
| The “frequency” is the number of observations before the seasonal pattern repeats. |
| This is the opposite of the definition of frequency in physics, or in Fourier analysis, where this would be called the “period”. |
| When using the ts() function in R, the following choices should be used. |
| Data frequency |

Annual 1 Quartely 4 Monthly 12 Weekly 52

Actually, there are not 52 weeks in a year, but on average, allowing for a leap year every fourth year. But most functions which use ts objects require integer frequency.

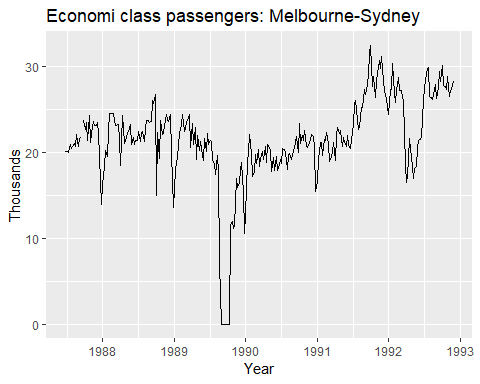
If the frequency of observations is greater than once per week, then there is usually more than one way of handling the frequency. For example, data with daily observations might have a weekly seasonality (frequency) or an annual seasonality (frequency). Similarly, data that are observed every minute might have an hourly seasonality (frequency ), a daily seasonality (frequency), a weekly seasonality (frequency) and an annual seasonality (frequency). If you want to use a ts object, then you need to decide which of these is the most important.

In [chapter 11](https://otexts.org/fpp2/advanced.html#advanced) we will look at handling these types of multiple seasonality, without having to choose just one of the frequencies.

## 2.2 Time plots

For time series data, the obvious graph to start with is a time plot. That is, the observations are plotted against the time of observations, with consecutive observations joined by straight lines. Figure 2.1 below shows the weekly economiy passenger load on Ansett Airlines between Australia’s two largest cities.

autoplot(melsyd[,"Economy.Class"])+  
 ggtitle("Economi class passengers: Melbourne-Sydney")+  
 xlab("Year")+  
 ylab("Thousands")



Weekly economy passenger load on Ansett Airlines.

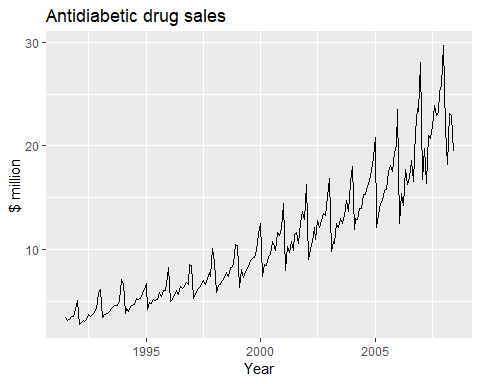
We will use the autopot() command frequently. It automatically produces an appropriate plot of whatever you pass to it in the first argument. In this case, it recognises melsyd[,"Economy.Class"] as a time series and produces a time plot

The time plot immediately reveals some intersting features. - There was a period in 1989 when no passengers were carried — this was due to an industrial dispute. - There was a period of reduced load in 1992. This was due to a trial in which some economy class seats were replaced by business class seats. - A large increase in passenger load occurred in the second half of 1991. - There are some large dips in load around the start of each year. These are due to holiday effects. - There is a long-term fluctuation in the level of the series which increases during 1987, decreases in 1989, and increases again through 1990 and 1991. - There are some periods of missing observations.

Any model will need to take all these features into account in order to effectively forecast the passenger load into the future.

A simpler time series is shown in Figure @ref(fig:a10).

autoplot(a10)+  
 ggtitle("Antidiabetic drug sales")+  
 ylab("$ million")+  
 xlab("Year")



Monthly sales of antidiabetic drugs in Australia.

Here, there is a clear and increasing trend. There is also a strong seasonal pattern that increases in size as the level of the series increases. The sudden drop at the start of each year is caused by a government subsidisation scheme that makes it cost-effective for patients to stockpile drugs at the end of the calendar year. Any forecasts of this series would need to capture the seasonal pattern, and the fact that the trend is changing slowly.

## 2.3 Time series patterns

In describing these time series, we have used words such as “trend” and “seasonal” which need to be defined more carefully.

#### Trend

A trend exists when there is a long-term increase or decrease in the data. It does not have to be linear. Sometimes we will refer to a trend as “changing direction”, when it might go from an increasing trend to a decreasing trend. There is a trend in the antidiabetic drug sales data shown in Figure 2.2.

#### Seasonal

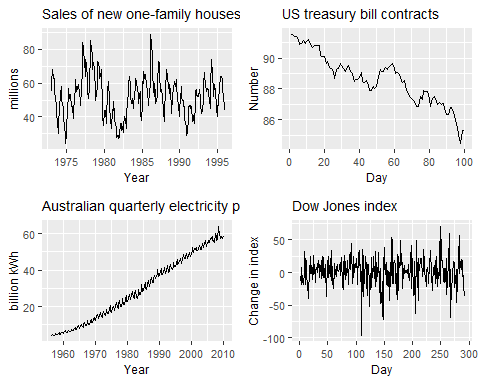
A seasonal pattern occurs when a time series is affected by seasonal factors such as the time of the year or the day of the week. Seasonality is always of a fixed and known frequency. The monthly sales of antidiabetic drugs above shows seasonality which is induced partly by the change in the cost of the drugs at the end of the calendar year.

#### Cyclic

A cycle occurs when the data exhibit rises and falls that are not of a fixed frequency. These fluctuations are usually due to economic conditions, and are often related to the “business cycle”. The duration of these fluctuations is usually at least 2 years.

Many people confuse cyclic behaviour with seasonal behaviour, but they are really quite different. If the fluctuations are not of a fixed frequency then they are cyclic; if the frequency is unchanging and associated with some aspect of the calendar, then the pattern is seasonal. In general, the average length of cycles is longer than the length of a seasonal pattern, and the magnitudes of cycles tend to be more variable than the magnitudes of seasonal patterns.

Many time series include trend, cycles and seasonality. When choosing a forecasting method, we will first need to identify the time series patterns in the data, and then choose a method that is able to capture the patterns properly.

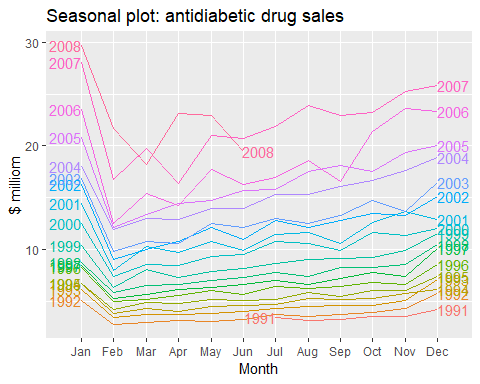
The examples in Figure 2.3 show different combinations of the above components. 

1. The monthly housing sales (top left) show strong seasonality within each year, as well as some strong cyclic behaviour with a period of about 6–10 years. There is no apparent trend in the data over this period.
2. The US treasury bill contracts (top right) show results from the Chicago market for 100 consecutive trading days in 1981. Here there is no seasonality, but an obvious downward trend. Possibly, if we had a much longer series, we would see that this downward trend is actually part of a long cycle, but when viewed over only 100 days it appears to be a trend.
3. The Australian monthly electricity production (bottom left) shows a strong increasing trend, with strong seasonality. There is no evidence of any cyclic behaviour here.
4. The daily change in the Dow Jones index (bottom right) has no trend, seasonality or cyclic behaviour. There are random fluctuations which do not appear to be very predictable, and no strong patterns that would help with developing a forecasting model.

## 2.4 Seasonal plots

A seasonal plot is similar to a time plot except that the data are plotted against the individual “seasons” in which the data were observed. An example is given below showing the antidiabetic drug sales.

ggseasonplot(a10,year.labels = T,year.labels.left = T)+  
 ylab("$ milliom")+  
 ggtitle("Seasonal plot: antidiabetic drug sales")



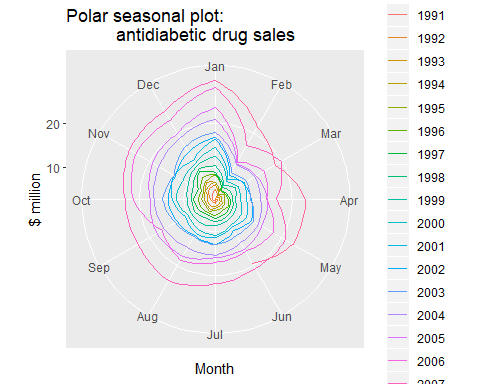
Seasonal plot of monthly antidiabetic drug sales in Australia.

These are exactly the same data as were shown earlier, but now the data from each season are overlapped. A seasonal plot allows the underlying seasonal pattern to be seen more clearly, and is especially useful in identifying years in which the pattern changes.

In this case, it is clear that there is a large jump in sales in January each year. Actually, these are probably sales in late December as customers stockpile before the end of the calendar year, but the sales are not registered with the government until a week or two later. The graph also shows that there was an unusually small number of sales in March 2008 (most other years show an increase between February and March). The small number of sales in June 2008 is probably due to incomplete counting of sales at the time the data were collected.

A useful variation on the seasonal plot uses polar coordinates. Setting polar=TRUE makes the time series axis circular rather than horizontal, as shown below.

ggseasonplot(a10,polar=T)+  
 ylab("$ million")+  
 ggtitle("Polar seasonal plot:  
 antidiabetic drug sales")

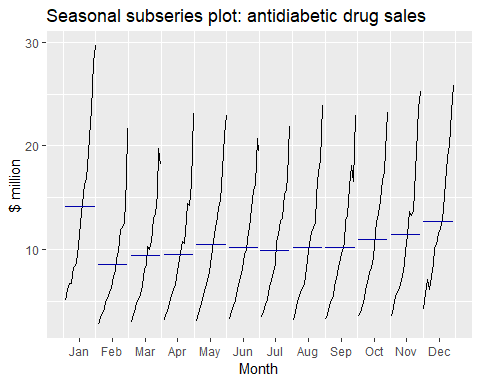


Polar seasonal plot of monthly antidiabetic drug sales in Australia.

## 2.5 Seasonal subseries plots

An alternative plot that emphasizes the seasonal patterns is where the data for each season are collected together in separate mini time plots.

ggsubseriesplot(a10)+  
 ylab("$ million")+  
 ggtitle("Seasonal subseries plot: antidiabetic drug sales")



Seasonal subseries plot of monthly antidiabetic drug sales in Australia.

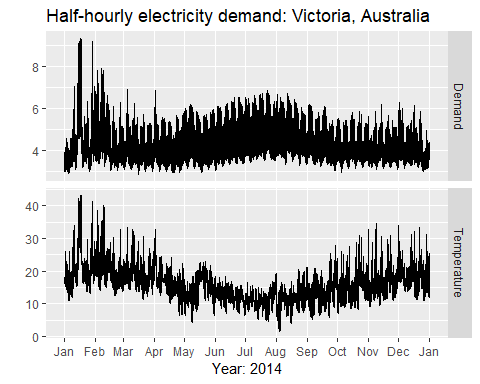
The horizontal lines indicate the means for each month. This form of plot enables the underlying seasonal pattern to be seen clearly, and also shows the changes in seasonality over time. It is especially useful in identifying changes within particular seasons. In this example, the plot is not particularly revealing; but in some cases, this is the most useful way of viewing seasonal changes over time.

## 2.6 Scatterplots

The graphs discussed so far are useful for visualising individual time series. It is also useful to explore relationships *between* time series.

Figure 2.7 shows two time series: half-hourly electricity demand (in Gigawatts) and temperature (in degrees Celsius), for 2014 in Victoria, Australia. The temperatures are for Melbourne, the largest city in Victoria, while the demand values are for the entire state.

month.breaks <- cumsum(c(0,31,28,31,30,31,30,31,31,30,31,30,31)\*48)  
autoplot(elecdemand[,c(1,3)], facet=TRUE) +  
 xlab("Year: 2014") + ylab("") +  
 ggtitle("Half-hourly electricity demand: Victoria, Australia")+  
 scale\_x\_continuous(breaks=2014+month.breaks/max(month.breaks),  
 minor\_breaks=NULL, labels=c(month.abb,month.abb[1]))

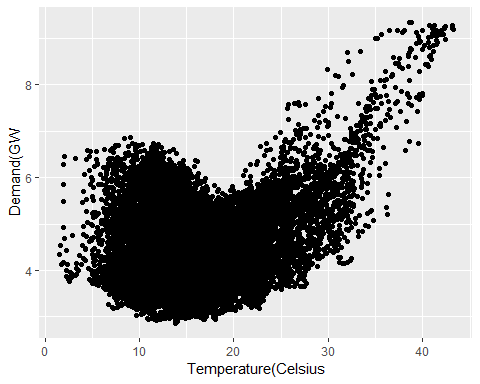


Half hourly electricity demand and temperatures in Victoria, Australia, for 2014.

(The actual code for this plot is little more complidated than what is shown in order to include the months on the x-axis.)

We can study the relationship between demand and temparature by plotting one series against the other.

qplot(Temperature,Demand,data=as.data.frame(elecdemand))+  
 ylab("Demand(GW")+xlab("Temperature(Celsius")

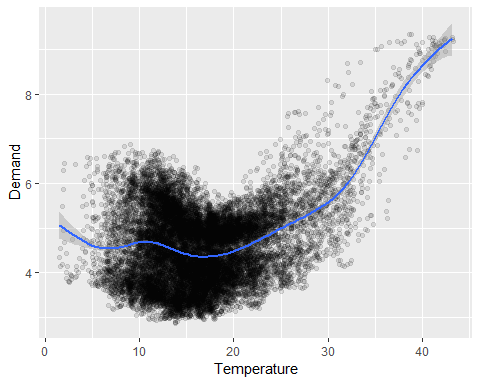


head(data.frame(elecdemand))

## Demand WorkDay Temperature  
## 1 3.914647 0 18.2  
## 2 3.672550 0 17.9  
## 3 3.497539 0 17.6  
## 4 3.339145 0 16.8  
## 5 3.204313 0 16.3  
## 6 3.100043 0 16.6

## ggplot()  
data.frame(elecdemand) %>%   
 ggplot(aes(x=Temperature,y=Demand))+  
 geom\_jitter(alpha=0.1)+  
 geom\_smooth()

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'



This scatterplot helps us to visualise the relationship between the variables. It is clear that high demand occurs when temperatures are high due to the effect of air-conditioning. But there is also a heating effect, where demand increases for very low temperatures.

#### Correlation

It is common to compute *correlation coefficient* to measure the strength of the relationship between two variables. The correlation between variable and is given by:

The value of always lies between and with negative values indicating a negative relationship and positive values indicating a positive relationship. The graphs in Figure 2.9 show examples of data sets with varying levels of correlation.

The correlation coefficient only measures the strength of the linear relationship, and can sometimes be misleading. For example, the correlation for the electricity demand and temperature data shown in Figure 2.8 is , but the non-linear relationship is stronger than that.

The plots in Figure 2.10 all have correlation coefficients of 0.82, but they have very different relationships. This shows how important it is look at the plots of the data and not simply rely on correlation values.

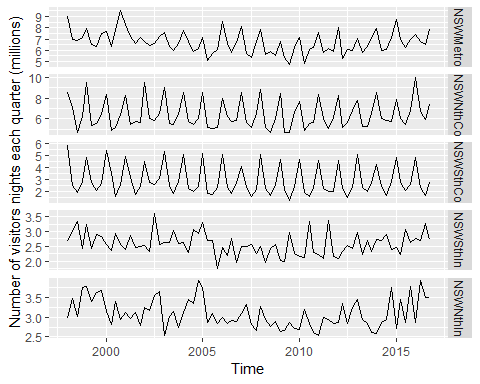
#### Scatterplot matrices

When there are several potential predictor variables, it is useful to plot each variable against each other variable. Consider the five time series shown in Figure 2.11, showing quarterly visitor numbers for five regions of New South Wales, Australia.

# data overview  
data.frame(fpp2::visnights) %>%   
 head()

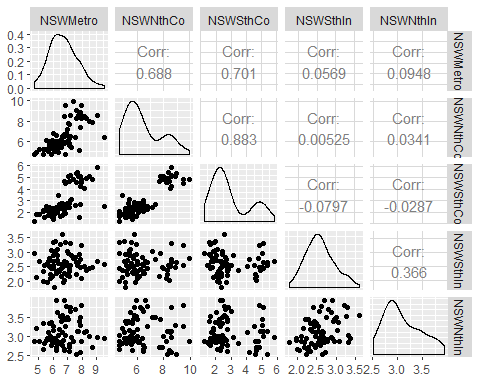
## NSWMetro NSWNthCo NSWSthCo NSWSthIn NSWNthIn QLDMetro QLDCntrl QLDNthCo  
## 1 9.047095 8.565678 5.818029 2.679538 2.977507 12.106052 2.748374 2.137234  
## 2 6.962126 7.124468 2.466437 3.010732 3.477703 7.786687 4.040915 2.269596  
## 3 6.871963 4.716893 1.928053 3.328869 3.014770 11.380024 5.343964 4.890227  
## 4 7.147293 6.269299 2.797556 2.417772 3.757972 9.311460 4.260419 2.621548  
## 5 7.956923 9.493901 4.853681 3.224285 3.790760 12.671942 4.186113 2.483203  
## 6 6.542243 5.401201 2.759843 2.428489 3.395284 9.582965 4.237806 3.377830  
## SAUMetro SAUCoast SAUInner VICMetro VICWstCo VICEstCo VICInner  
## 1 2.881372 2.591997 0.8948773 7.490382 2.4420048 3.381972 5.326655  
## 2 2.124736 1.375780 0.9792509 5.198178 0.9605047 1.827940 4.441119  
## 3 2.284870 1.079542 0.9803289 5.244217 0.7559744 1.351952 3.815645  
## 4 1.785889 1.497664 1.5094343 6.274246 1.2716040 1.493415 3.859567  
## 5 2.293873 2.247684 0.9635227 9.187422 2.3850583 2.896929 4.588755  
## 6 2.197418 1.672802 0.9968803 4.992303 1.3288638 1.547901 4.070401  
## WAUMetro WAUCoast WAUInner OTHMetro OTHNoMet  
## 1 3.075779 3.066555 0.6949954 3.437924 2.073469  
## 2 2.154929 3.334405 0.5576796 2.677081 1.787939  
## 3 2.787286 4.365844 1.0061844 3.793743 2.345021  
## 4 2.752910 4.521996 1.1725514 3.304231 1.943689  
## 5 3.519564 3.579347 0.3981829 3.510819 2.165838  
## 6 3.160430 3.408533 0.5960182 2.871867 1.803940

autoplot(visnights[,1:5],facets=T)+  
 ylab("Number of visitors nights each quarter (millions)")



To see the relationships between these five time series, we can plot each time series against the others. These plots can be arranged in a scatterplot matrix, as shown in Figure 2.12. (This plot requires the GGally package to be installed.)

GGally::ggpairs(as.data.frame(visnights[,1:5]))



For each panel, the variable on the vertical axis is given by the variable name in that row, and the variable on the horizontal axis is given by the variable name in that column. There are many options available to produce different plots within each panel. In the default version, the correlations are shown in the upper right half of the plot, while the scatterplots are shown in the lower half. On the diagonal are shown density plots.

The value of the scatterplot matrix is that it enables a quick view of the relationships between all pairs of variables. In this example, the second column of plots shows there is a strong positive relationship between visitors to the NSW north coast and visitors to the NSW south coast, but no detectable relationship between visitors to the NSW north coast and visitors to the NSW south inland. Outliers can also be seen. There is one unusually high quarter for the NSW Metropolitan region, corresponding to the 2000 Sydney Olympics. This is most easily seen in the first two plots in the left column of Figure 2.12, where the largest value for NSW Metro is separate from the main cloud of observations.

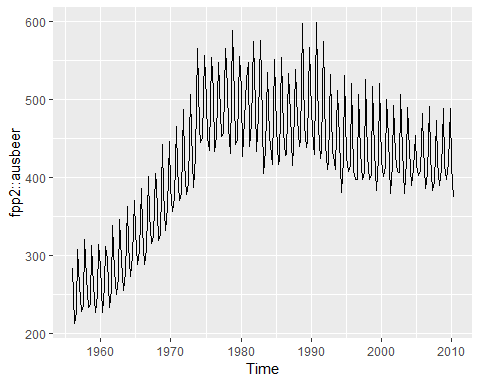
## 2.7 Lag plots

Figure 2.13 displays scatterplots of quarterly Australian beer production, where the horizontal axis shows lagged values of the time series. Each graph shows plotted against for different values of .

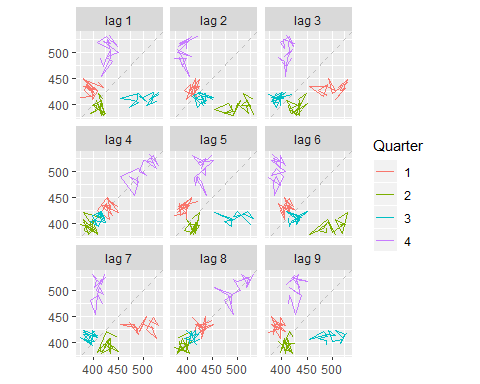
# aus beer overview  
fpp2::ausbeer %>%   
 as.data.frame() %>%   
 head()

## x  
## 1 284  
## 2 213  
## 3 227  
## 4 308  
## 5 262  
## 6 228

autoplot(fpp2::ausbeer)



beer2 <- window(ausbeer,start=1992)  
gglagplot(beer2)



Here the colours indicate the quarter of the variable on the vertical axis. The lines connect points in chronological order. The relationship is strongly positive at lags 4 and 8, reflecting the strong quarterly seasonality in the data. The negative relationship seen for lags 2 and 6 occurs because peaks (in Q4) are plotted against troughs (in Q2)

The window() function used here is very useful when extracting a portion of a time series. In this case, we have extracted the data from ausbeer, beginning in 1992.

## 2.8 Autocorrelation

Just as correlation measures the extent of a linear relationship between two variables, autocorrelation measures the linear relationship between *lagged values* of a time series.

There are sefveral autocorrelation coefficients, corresponding to each panel in the lag plot. For example, measures the relationship between and , measures the relationship between and and so on.

The value of can be written as

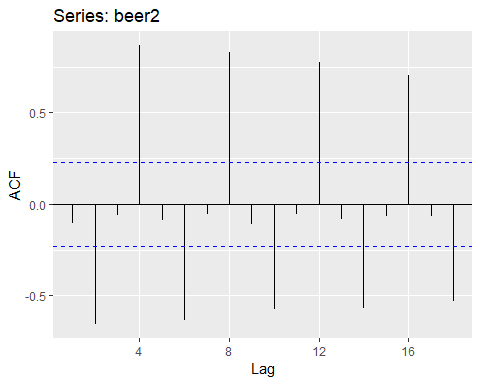
, where is the length of the time series.

The first nine autocorrelation coefficients for the beer production data are given in the following table.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| -0.102 | -0.657 | -0.060 | 0.869 | -0.089 | -0.635 | -0.054 | 0.832 | -0.108 |

These correspond to the nine scatterplots in Figure 2.13. The autocorrelation coefficients are plotted to show the *autocorrelation function* of **ACF**. The plot is also known as a *correlogram*.

ggAcf(beer2)



In this graph: - is higher than for the other lags. This is due to the seasonal pattern in the data: the peaks tend to be four quarters apart and the troughs tend to be two quarters apart. - is more negative than for the other lags because troughs tend to be two quarters behind peaks. - The dashed blue lines indicate whether the correlations are significantly different from zero. These are explained in Section 2.9.

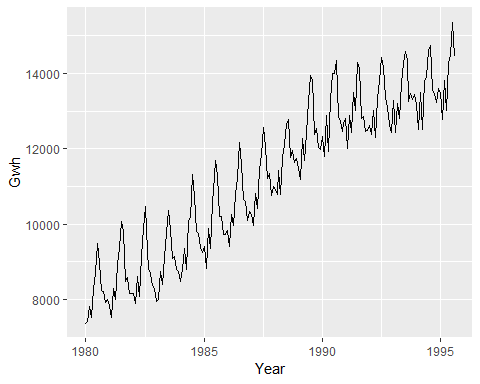
#### Trend and seasonality in ACF plots

When data have a trend, the autocorrelations for small lags trend to be large and positive because observations nearby in time are also nearby in size. Thus the ACF of trended time series tend to have positive values that slowly decrease as the large increase.

When data are seasonal, the autocorrelation will be larger for the seasonal lags(at multiples of the seasonal frequency) than for other lags.

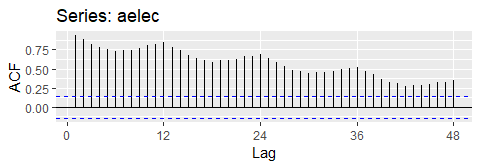
When data are both trended and seasonal, you see a combination of these effects. The monthly, Australian electricity demand series plotted in Figure 2.15 shows both trend and seasonality. Its ACF is shown in Figure 2.16.

aelec <- window(elec,start=1980)  
autoplot(aelec)+xlab("Year")+ylab("Gwh")



Monthly Australian electricity demand from 1980–1995.

ggAcf(aelec,lag=48)



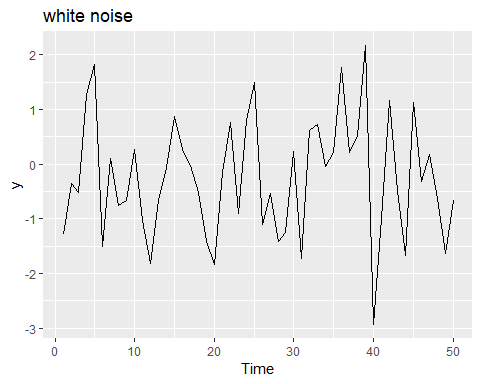
ACF of monthly Australian electricity demand.

The slow decrease in the ACF as the lags increase is due to the trend, while the “scalloped” shape is due the **seasonality**.

## 2.9 White noise

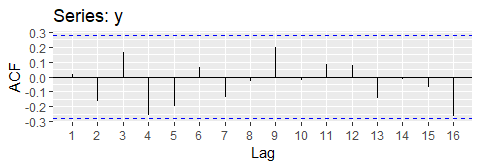
Time series that show no autocorrelation are called **white nose**. Figure 2.17 gives an example of a white noise series.

set.seed(30)  
y <- ts(rnorm(50))  
autoplot(y)+ggtitle("white noise")



A white noise time series.

ggAcf(y)



Autocorrelation function for the white noise series.

For white noise series, we expect each autocorrelation to be close to zero. Of course, they will not be exactly equal to zero as there is some random variation. For a white noise series, we expect 95% of the spikes in the ACF to lie within where is the length of the time series. It is common to plot these bounds on a graph of the ACF (the blue dashed lines above). IF one or more large spikes are outside these bounds, or if substantially more than 5% or spikes are outside these bounds, then the series is probably not white noise.

In this example, T=50 and the bounds are . All of the autocorrelation coefficients lie within these limits, confirming that the data are white noise.

## 2.10 Exercises

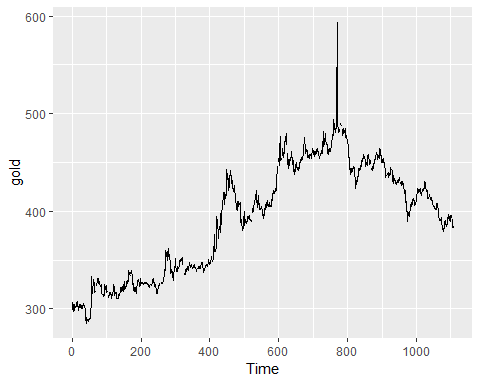
<https://github.com/mldataanalysis/Time-Series-Solutions>

1. Usethe help function to explore what the series gold, woolyrinq and gas represent.

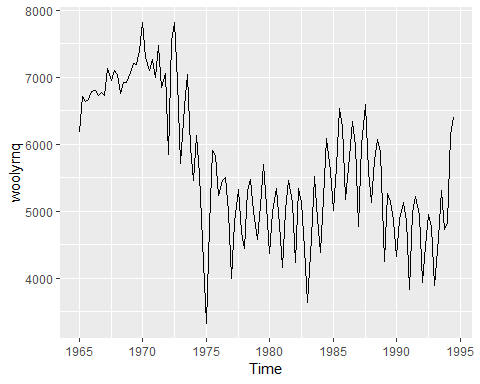
* *gold*: Daily morning gold prices in US dollars. 1 January 1985 – 31 March 1989.
* *woolynrnq*: Quarterly production of woollen yarn in Australia: tonnes. Mar 1965 – Sep 1994
* *gas*:Australian monthly gas production 1956-1995

1. Use autoplot() to plot each of these in separate plots.

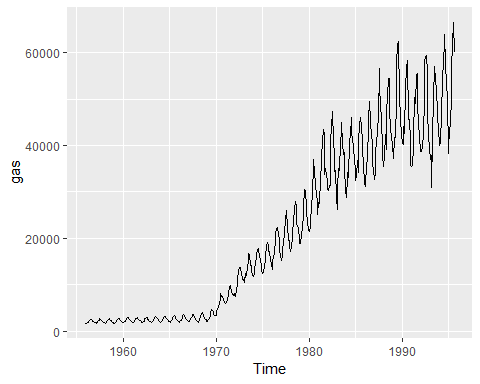
library(fpp2)  
autoplot(gold)



autoplot(woolyrnq)



autoplot(gas)



1. What is the frequency of each series. Hint: apply the frequency function.

frequency(gold)

## [1] 1

frequency(woolyrnq)

## [1] 4

frequency(gas)

## [1] 12

1. Download the file tute1.csv from the [the book website](http://otexts.org/fpp2/extrafiles/tute1.csv), open it in excel and review its contents. You should find four columns of information. Columns B through D each contains a quarterly series,labelled Sales, AdBudget and GDP. Sales contains the quarterly sales for a small company over the period 1981-2005. AdBudget is the advertising budget and GDP is the gross domestic product. All series have been adjusted for inflation.
2. Read the data into R with the following script:

tute1 <- read\_csv("C:/Users/kojikm.mizumura/Desktop/Data Science/4. Forecasting Princples and Practice/tute1.csv")

## Warning: Missing column names filled in: 'X1' [1]

## Parsed with column specification:  
## cols(  
## X1 = col\_character(),  
## Sales = col\_double(),  
## AdBudget = col\_double(),  
## GDP = col\_double()  
## )

head(tute1)

## # A tibble: 6 x 4  
## X1 Sales AdBudget GDP  
## <chr> <dbl> <dbl> <dbl>  
## 1 Mar-81 1020. 659. 252.  
## 2 Jun-81 889. 589 291.  
## 3 Sep-81 795 512. 291.  
## 4 Dec-81 1004. 614. 292.  
## 5 Mar-82 1058. 647. 279.  
## 6 Jun-82 944. 602 254

1. Convert the data to time series

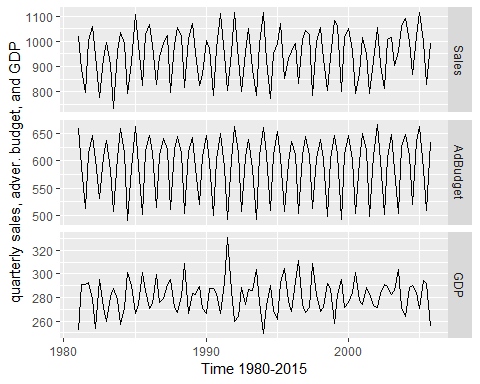
mytimeseries <- ts(tute1[,-1],start=1981,frequency = 4)  
mytimeseries %>% head()

## Sales AdBudget GDP  
## 1981 Q1 1020.2 659.2 251.8  
## 1981 Q2 889.2 589.0 290.9  
## 1981 Q3 795.0 512.5 290.8  
## 1981 Q4 1003.9 614.1 292.4  
## 1982 Q1 1057.7 647.2 279.1  
## 1982 Q2 944.4 602.0 254.0

Note that the [,-1] removes the first column which contains the quarters as we don’t need them now.

1. Construct time series plots of each of the three series

autoplot(mytimeseries,facet=TRUE)+  
 xlab("Time 1980-2015")+  
 ylab("quarterly sales, adver. budget, and GDP")



1. Download some monthly Australian retail data from the book website. These represent retail sales in various categories for different Australian states, and are stored in a MS-Excel file.
2. You can read the data into R with the following script

retaildata <- readxl::read\_excel("C:/Users/kojikm.mizumura/Desktop/Data Science/4. Forecasting Princples and Practice/retail.xlsx",skip=1)  
head(retaildata)

## # A tibble: 6 x 190  
## `Series ID` A3349335T A3349627V A3349338X A3349398A A3349468W  
## <dttm> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1982-04-01 00:00:00 303. 41.7 63.9 409. 65.8  
## 2 1982-05-01 00:00:00 298. 43.1 64 405. 65.8  
## 3 1982-06-01 00:00:00 298 40.3 62.7 401 62.3  
## 4 1982-07-01 00:00:00 308. 40.9 65.6 414. 68.2  
## 5 1982-08-01 00:00:00 299. 42.1 62.6 404. 66   
## 6 1982-09-01 00:00:00 305. 42 64.4 412. 62.3  
## # ... with 184 more variables: A3349336V <dbl>, A3349337W <dbl>,  
## # A3349397X <dbl>, A3349399C <dbl>, A3349874C <dbl>, A3349871W <dbl>,  
## # A3349790V <dbl>, A3349556W <dbl>, A3349791W <dbl>, A3349401C <dbl>,  
## # A3349873A <dbl>, A3349872X <dbl>, A3349709X <dbl>, A3349792X <dbl>,  
## # A3349789K <dbl>, A3349555V <dbl>, A3349565X <dbl>, A3349414R <dbl>,  
## # A3349799R <dbl>, A3349642T <dbl>, A3349413L <dbl>, A3349564W <dbl>,  
## # A3349416V <dbl>, A3349643V <dbl>, A3349483V <dbl>, A3349722T <dbl>,  
## # A3349727C <dbl>, A3349641R <dbl>, A3349639C <dbl>, A3349415T <dbl>,  
## # A3349349F <dbl>, A3349563V <dbl>, A3349350R <dbl>, A3349640L <dbl>,  
## # A3349566A <dbl>, A3349417W <dbl>, A3349352V <dbl>, A3349882C <dbl>,  
## # A3349561R <dbl>, A3349883F <dbl>, A3349721R <dbl>, A3349478A <dbl>,  
## # A3349637X <dbl>, A3349479C <dbl>, A3349797K <dbl>, A3349477X <dbl>,  
## # A3349719C <dbl>, A3349884J <dbl>, A3349562T <dbl>, A3349348C <dbl>,  
## # A3349480L <dbl>, A3349476W <dbl>, A3349881A <dbl>, A3349410F <dbl>,  
## # A3349481R <dbl>, A3349718A <dbl>, A3349411J <dbl>, A3349638A <dbl>,  
## # A3349654A <dbl>, A3349499L <dbl>, A3349902A <dbl>, A3349432V <dbl>,  
## # A3349656F <dbl>, A3349361W <dbl>, A3349501L <dbl>, A3349503T <dbl>,  
## # A3349360V <dbl>, A3349903C <dbl>, A3349905J <dbl>, A3349658K <dbl>,  
## # A3349575C <dbl>, A3349428C <dbl>, A3349500K <dbl>, A3349577J <dbl>,  
## # A3349433W <dbl>, A3349576F <dbl>, A3349574A <dbl>, A3349816F <dbl>,  
## # A3349815C <dbl>, A3349744F <dbl>, A3349823C <dbl>, A3349508C <dbl>,  
## # A3349742A <dbl>, A3349661X <dbl>, A3349660W <dbl>, A3349909T <dbl>,  
## # A3349824F <dbl>, A3349507A <dbl>, A3349580W <dbl>, A3349825J <dbl>,  
## # A3349434X <dbl>, A3349822A <dbl>, A3349821X <dbl>, A3349581X <dbl>,  
## # A3349908R <dbl>, A3349743C <dbl>, A3349910A <dbl>, A3349435A <dbl>,  
## # A3349365F <dbl>, A3349746K <dbl>, ...

The second argument skip=1 is required because the Excel sheet has two header rows.

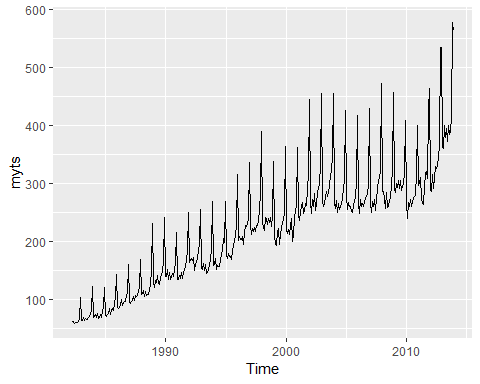
1. Select one of the time series as follows (but replace the column name with your own chosen column):

myts <- ts(retaildata[,"A3349873A"],  
 frequency=12,start=c(1982,4))  
head(myts)

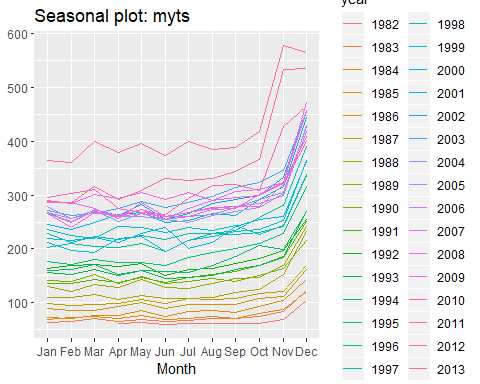
## Apr May Jun Jul Aug Sep  
## 1982 62.4 63.1 59.6 61.9 60.7 61.2

1. Explore your chosen retail time series using the following functions:

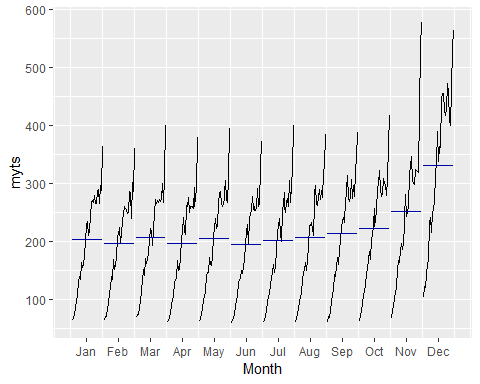
autoplot(myts)



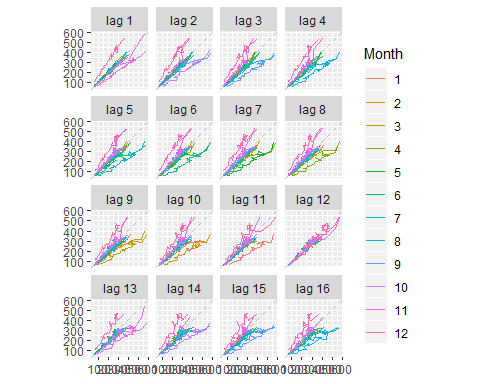
# year-based split  
ggseasonplot(myts)



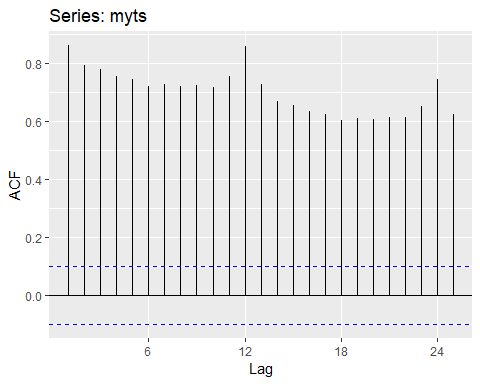
# month-based split  
ggsubseriesplot(myts)



# lag-plot  
# Each graph shows $y\_t$ plotted against $y\_{t−k}$ for different values of $k$.  
gglagplot(myts)



ggAcf(myts)



1. Create time plots of the following time series: bicoal, chicken, dole,usdeaths,lynx,goog,writing,fancy,a10,h02

* usehelp() to fundout about the data in each series.
* For the goog plot, modify the axis label and title.

goog: Closing stock prices of GOOG from the NASDAQ exchange, for 1000 consecutive trading days between 25 February 2013 and 13 February 2017. Adjusted for splits. goog200 contains the first 200 observations from goog.

# help(goog)  
# str(dole)  
glimpse(goog)

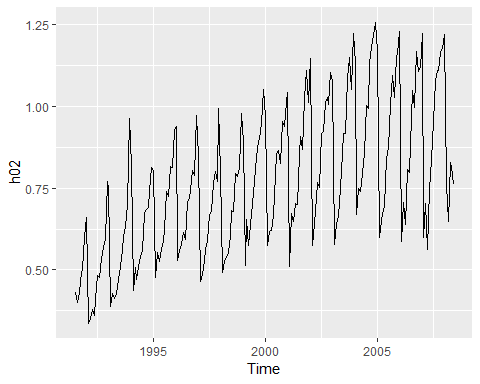
## Time-Series [1:1000] from 1 to 1000: 393 393 397 398 400 ...

head(goog)

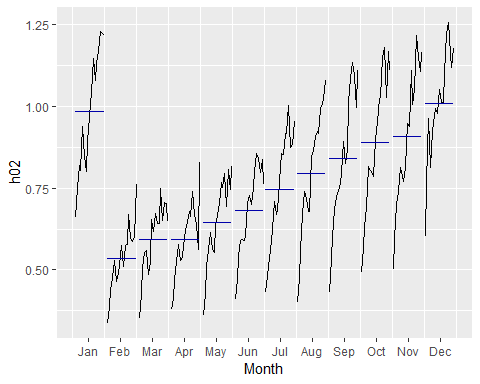
## Time Series:  
## Start = 1   
## End = 6   
## Frequency = 1   
## [1] 392.8300 392.5121 397.3059 398.0113 400.4902 408.0957

**5**. Use the ggseasonplot() and ggsubseriesplot() functions to explore the seasonal patterns in the following time series: writing, fancy, a10, h02. - What can you say about the seasonal patterns? - Can you identify any unusual years?

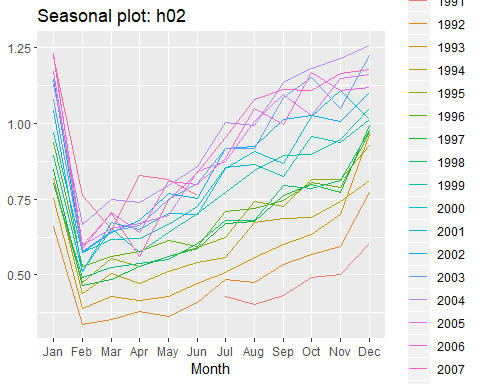
# h02 dataset  
# Monthly corticosteroid drug sales in Australia from July 1991 to June 2008.  
  
autoplot(h02)



ggsubseriesplot(h02)



ggseasonplot(h02)

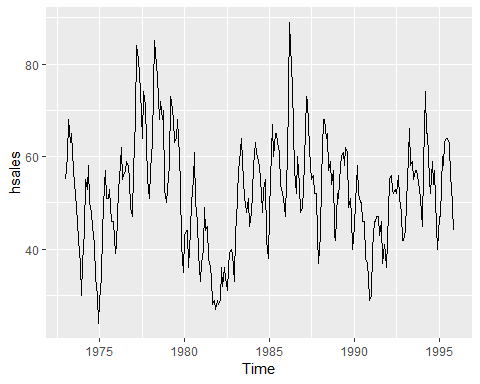


**6.** Use the following graphics functions: autoplot(), ggseasonplot(), ggsubseriesplot(), gglagplot(), ggAcf() and explore features from the following time series: hsales, usdeaths, bricksq, sunspotarea, gasoline. - Can you spot any seasonality, cyclicity and trend? - What do you learn about the series?

# Monthly sales of new one-family houses sold in the USA since 1973.  
head(hsales)

## Jan Feb Mar Apr May Jun  
## 1973 55 60 68 63 65 61

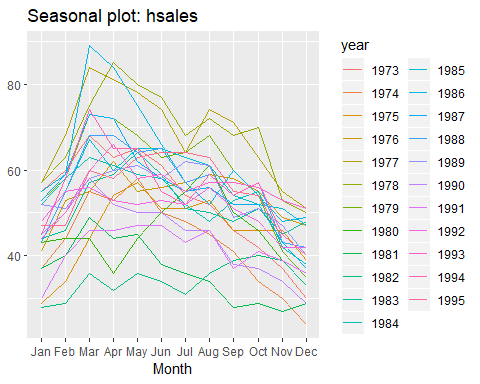
# dataset overview  
autoplot(hsales)



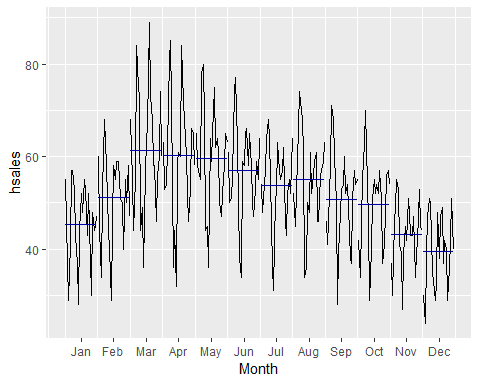
frequency(hsales) # monthly ts data

## [1] 12

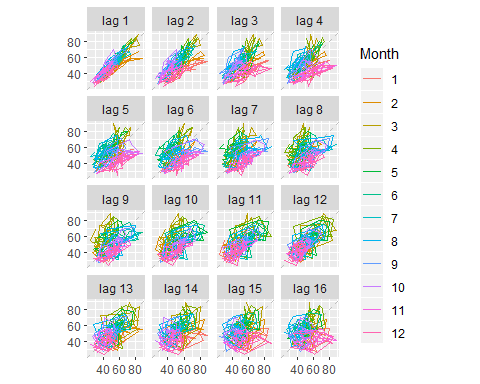
ggseasonplot(hsales)



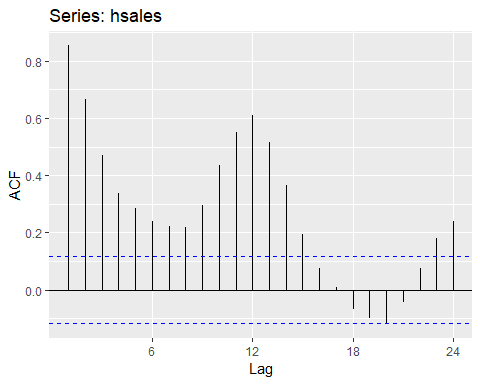
ggsubseriesplot(hsales)



gglagplot(hsales)

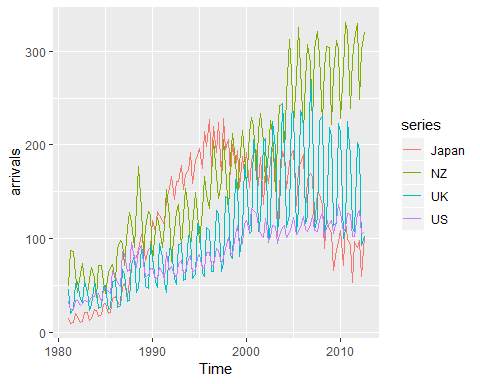


ggAcf(hsales)



1. The arrivals data set comprises quarterly international arrivals (in thousands) to Australia from Japan, New Zealand, UK and the US. Use autoplot(), ggseasonplot() and ggsubseriesplot() to compare the differences between the arrivals from these four countries. Can you identify any unusual observations?

# Quarterly international arrivals (in thousands) to Australia from Japan, New Zealand, UK and the US. 1981Q1 - 2012Q3.  
autoplot(arrivals)



# ggseasonplot(arrivals)  
# ggsubseriesplot(arrivals)

1. Skipped

**9**.The pigs data shows the monthly total number of pigs slaughtered in Victoria, Australia, from Jan 1980 to Aug 1995. Use mypigs <- window(pigs, start=1990) to select the data starting from 1990. Use autoplot and ggAcf for mypigs series and compare these to white noise plots from Figures 2.17 and 2.18.

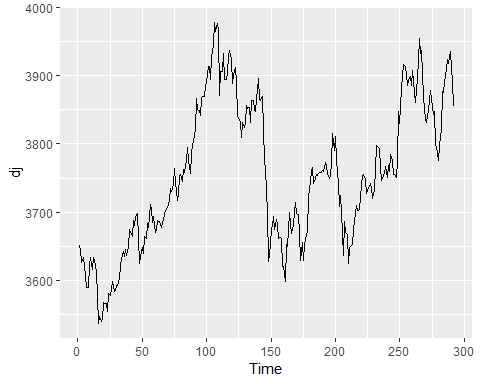
head(pigs)

## Jan Feb Mar Apr May Jun  
## 1980 76378 71947 33873 96428 105084 95741

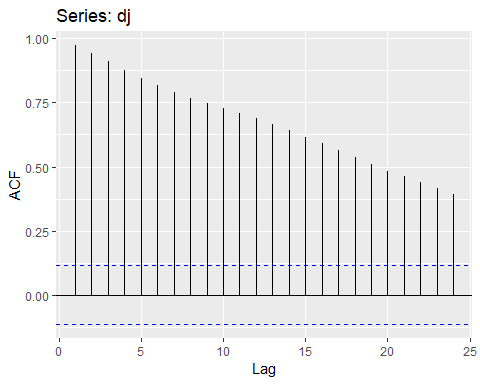
**10**. dj contains 292 consecutive traiding days of the Dow Jones Index. Use ddj<-diff(dj) to compute the daily changes in the index. Plot ddj and its ACF. Do the changes in the Dow Jones Index look like white noise?

dj: Dow-Jones index Dow-Jones index on 251 trading days ending 26 Aug 1994.

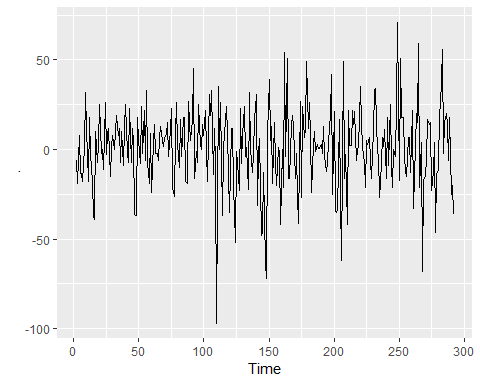
# data overview  
# help(dj)  
autoplot(dj)



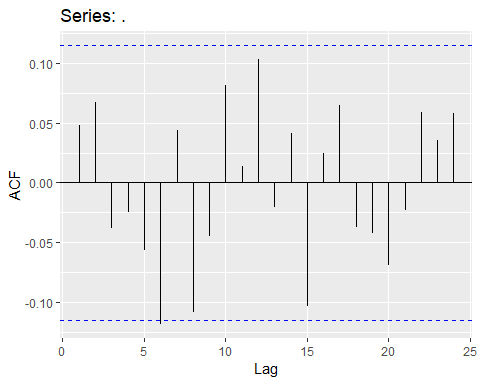
ggAcf(dj)



# DID   
diff(dj) %>%   
 autoplot()



diff(dj) %>%   
ggAcf()



## 2.11 Further reading

* Cleveland (1993) is a classic book on the principles of visualisation for data analysis. While it is more than 20 years old, the ideas are timeless.
* Unwin (2015) is a modern introduction to graphical data analysis using R. It does not have much information on time series graphics, but plenty of excellent general advice on using graphics for data analysis.

Cleveland, W. S. (1993). Visualizing data. Hobart Press. Unwin, A. (2015). Graphical data analysis with R. Chapman; Hall/CRC.

# Chapter 3: The forecaster’s toolbox

In this chapter, we discuss some general tools that are useful for many different forecasting situations. We will describe some benchmark forecasting methods, ways of making the forecasting task simpler using transformations and adjustments, methods for checking whether a forecasting method has adequately utilised the available information, and techniques for computing prediction intervals.

Each of the tools discussed in this chapter will be used repeatedly in subsequent chapters as we develop and explore a range of forecasting methods.

## 3.1 some simple forecasting methods

Some forecasing methods are extremely simple and surprisingly effective. We will use the following four forecasing methods as benchmarks throughout this book.

#### Average method

Here, the forecasts of all future values