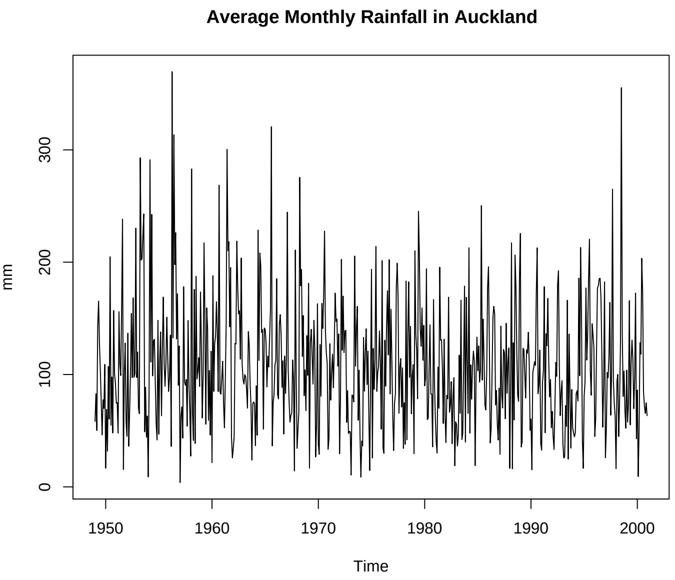
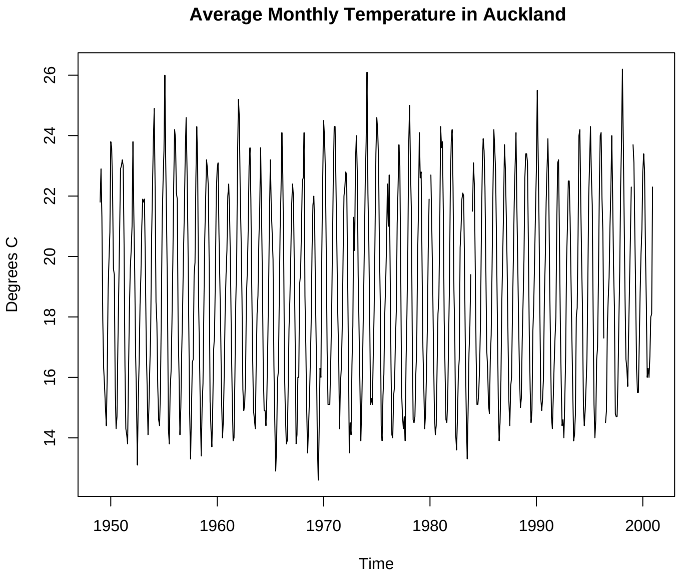
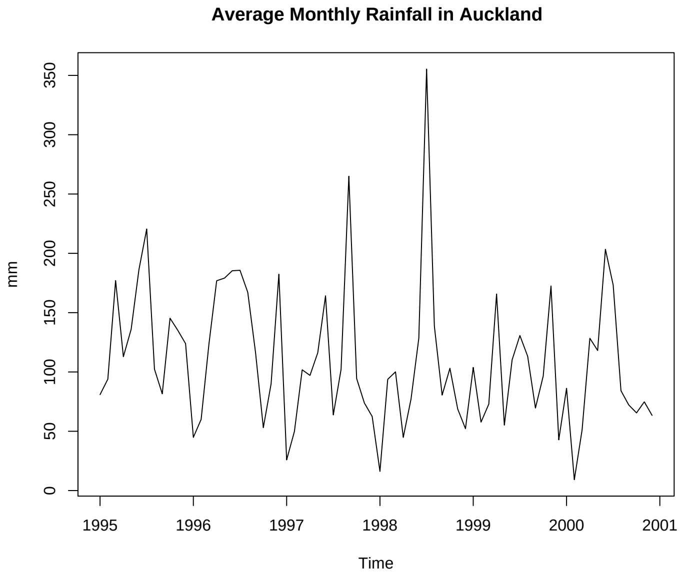
Statistics 120
Displaying Time Series Data

Time Series

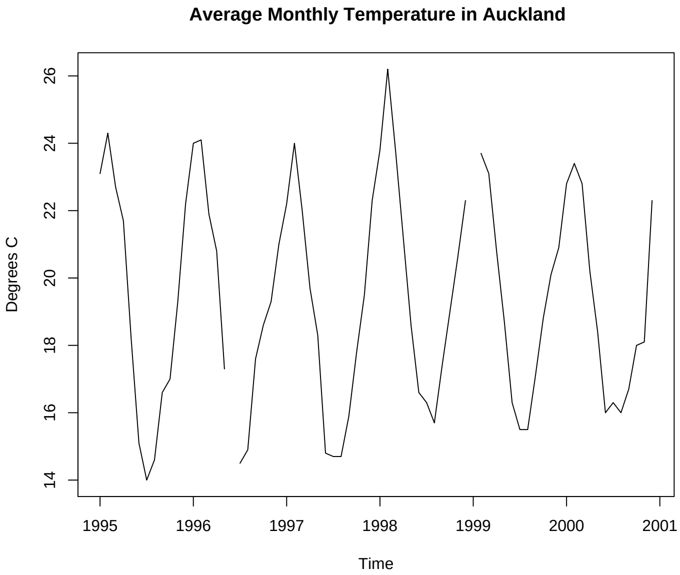
- A *time series* is a set of observations made at equally spaced points in time.
- Time series observations are usually numerical measurements, but occasionally categorical time series are encountered.
- Time series observations are typically not (statistically) independent.
- This means that the time order the observations is crucial to their analysis.

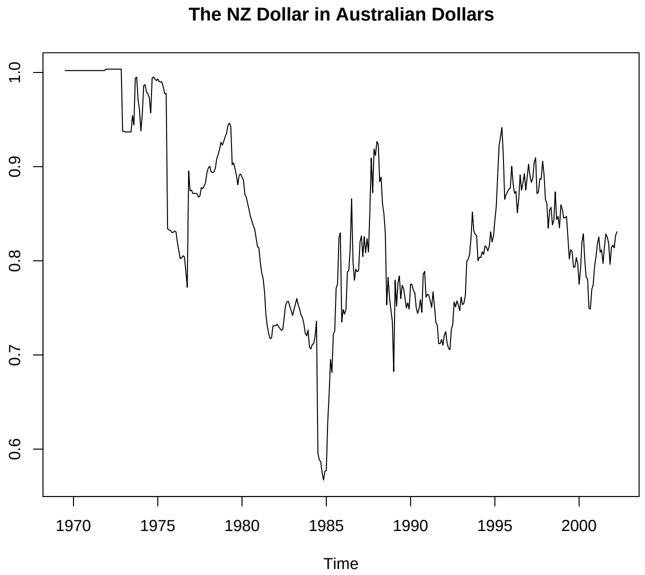

Average Monthly Rainfall in Auckland
mm
0
100
200
300
1950
1960
1970
1980
1990
2000
Time


Average Monthly Temperature in Auckland
Degrees C
26
24
22
20
18
16
14
1950
1960
1970
1980
1990
2000
Time



Average Monthly Rainfall in Auckland
mm
0
50
100
150
200
250
300
350
1995
1996
1997
1998
1999
2000
2001
Time


Average Monthly Temperature in Auckland
Degrees C
14
16
18
20
22
24
26
1995
1996
1997
1998
1999
2000
2001
Time


The NZ Dollar in Australian Dollars
1.0
0.9
0.8
0.7
0.6
1970
1975
1980
1985
1990
1995
2000
Time

Time Series in R

- The function `ts` can be used to turn an ordinary vector into a special time series object.
- It does this by specifying parameters which describe when the observations were made.
- The parameter `frequency` describes how many observations are made per unit time.
- The parameter `start` describes when sampling started.

Example – Creating the Rain Series

- Suppose that the Auckland rainfall values have been read into a vector called `rainvalues`.
- The values are monthly values with the first value sampled in January 1949.

```
> rain = ts(rainvalues, frequency = 12,
            start = c(1949, 1))
```

- R interprets the value 1949 as “the start of 1949” so we could use the simpler form.

```
> rain = ts(rainvalues, frequency = 12,
            start = 1949)
```

Simple Operations on Time Series

- Arithmetic operations can be carried out on time series just as you might expect.

```
> lograin = log(rain)
```

- Subsetting is done by focusing on the values of a time series which fall within a given time *window*.

```
> rain2000 = window(rain,
                    start = c(2000, 1),
                    end   = c(2000, 12))
```

Printing Time Series

- Time series are printed in a special way.

```
> window(rain, start = c(1999, 1),
            end = c(2000, 12))
       Jan   Feb   Mar   Apr   May   Jun
1999 103.8  57.7  72.9 165.7  55.2 110.2
2000  86.2   9.2  51.2 128.4 118.1 203.4
       Jul   Aug   Sep   Oct   Nov   Dec
1999 130.7 113.2  69.6  96.6 172.5  42.8
2000 173.2  84.2  72.2  65.5  74.8  63.4
```

- The frequency values 12 and 4 are recognised as special and taken to correspond to monthly and quarterly observations.

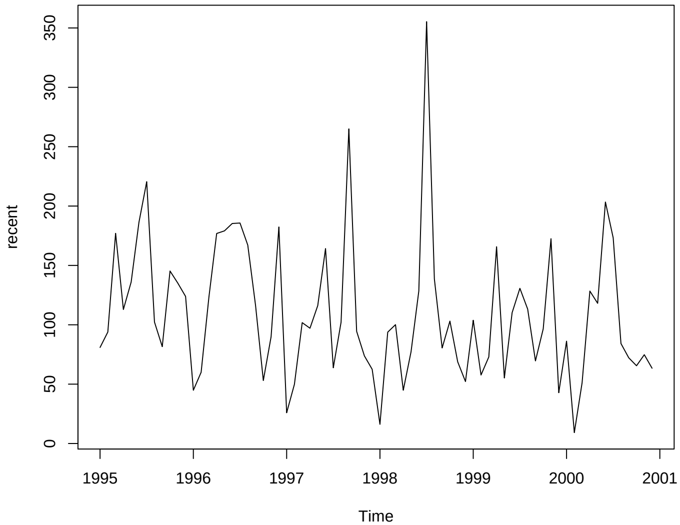
Time Series Plots

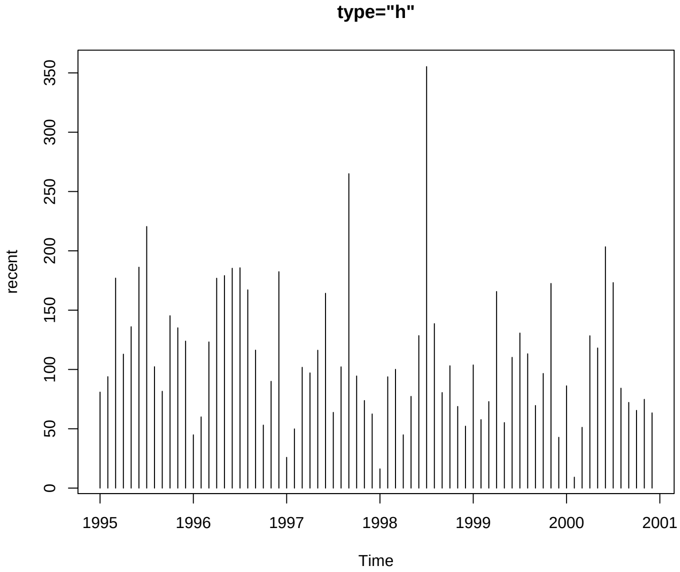
- The plot function recognises time series and plots them in an appropriate way.

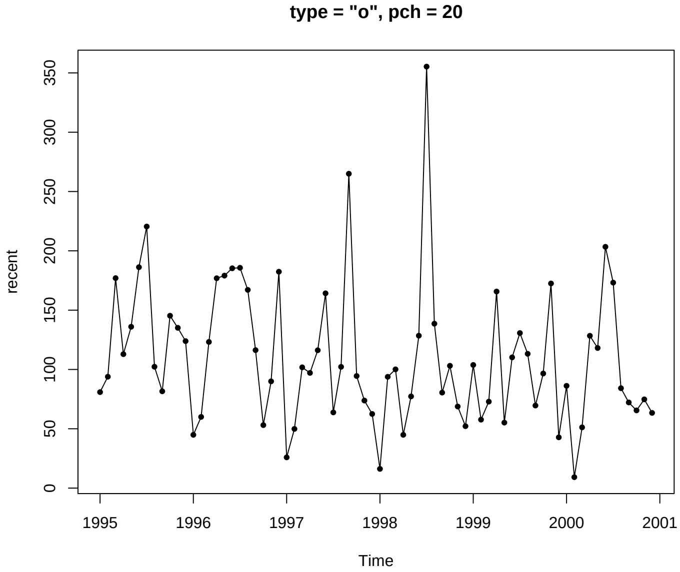
- The default plotting method is to "join up the dots," but this and other aspects of the plot can be customised.

```
> recent = window(rain, start = c(1995, 1),
                 end = c(2000, 12))
> plot(recent)
> plot(recent, type = "h")
> plot(recent, type = "o", pch=20)
```

The Default Time Series Plot




type="h"
recent
0
50
100
150
200
250
300
350
1995
1996
1997
1998
1999
2000
2001
Time

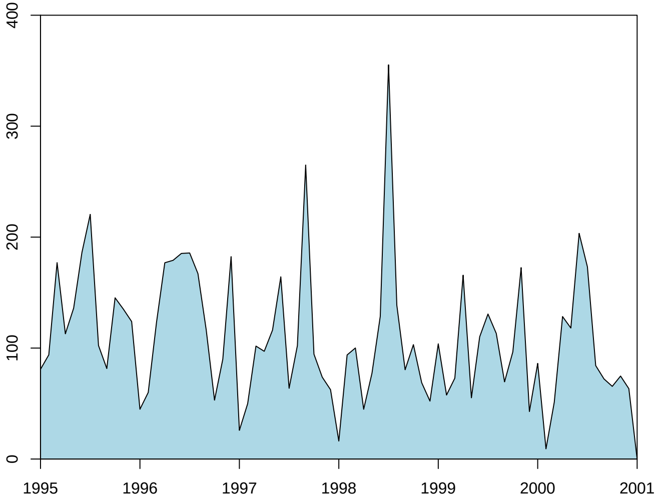


type = "o", pch = 20
recent
0
50
100
150
200
250
300
350
1995
1996
1997
1998
1999
2000
2001
Time

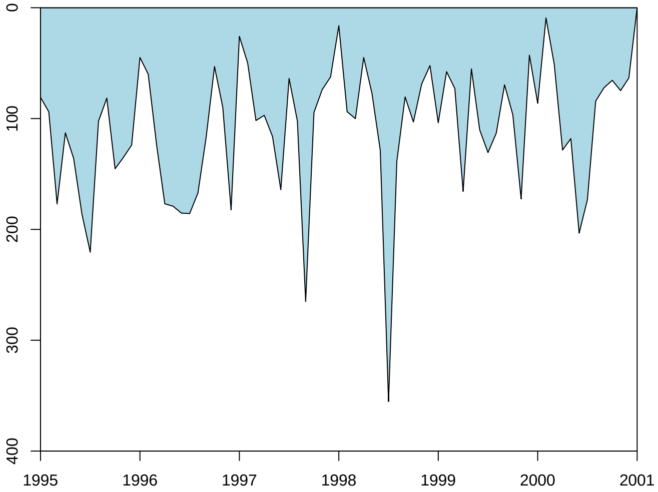
Time Series Plots

- We saw earlier in the course that it is easy to produce filled time series plots using `polygon`.

```
> plot.new()
> plot.window(c(1995, 2001), xaxs = "i",
              c(0,400), yaxs = "i")
> x = c(1995, time(recent), 2001)
> y = c(0, recent, 0)
> polygon(x, y, col = "lightblue")
> axis(1); axis(2); box()
```



400
300
200
100
0
1995
1996
1997
1998
1999
2000
2001


0
100
200
300
400
1995
1996
1997
1998
1999
2000
2001

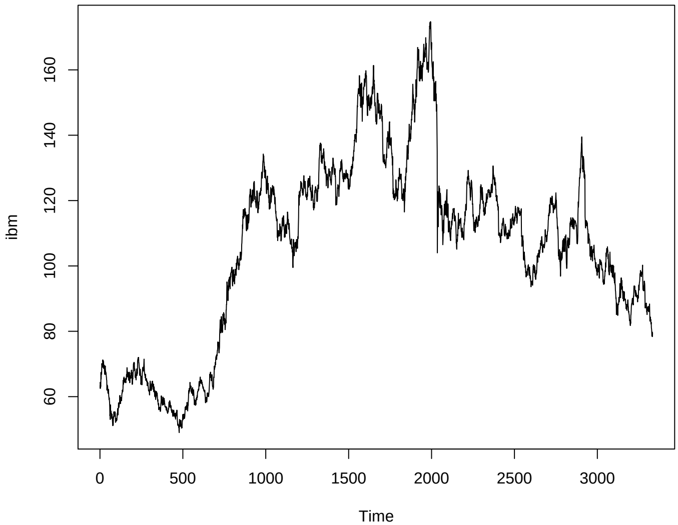
A Horizon Effect?

- There appears to be a difference between these two plots.
- My conjecture is that there is wiring in the brain which means that we notice peaks rather than troughs in plots.
- This is especially true for plots which are divided horizontally by a colour horizon.
- Because of this effect, my recommendation is that you avoid this kind of plot.

Example: Stock Prices

- In this example we will look at the closing price for IBM stock, daily from Jan 1, 1980 to Oct. 8, 1992.
- This is a typical pattern for any stock.

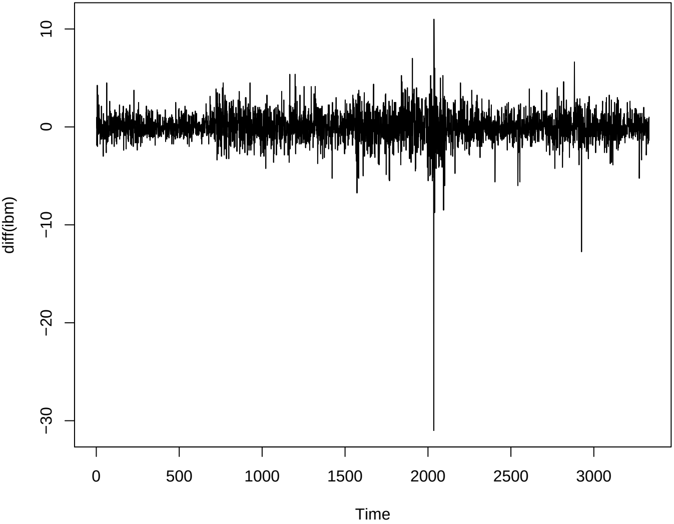
```
> plot(ibm)
```


ibm
160
140
120
100
80
60
0
500
1000
1500
2000
2500
3000
Time

Stock Prices and Efficient Markets

- Theory says that in an efficient market stock prices should behave as random walks.
- This means that on any given day the price of a stock will go up or down with equal probability.
- There are a variety of reasons why the New Zealand market cannot be considered efficient.
- We can check the theory with the IBM stock by examining the first differences in the series – i.e. each day's value minus the day before.

```
> plot(diff(ibm))
```


10
0
−10
−20
−30
diff(ibm)
0
500
1000
1500
2000
2500
3000
Time

Time Series Decomposition

- It can useful to regard many real world series as being composed of several independent components.
- A particularly useful model is the trend plus seasonal plus irregular component model.

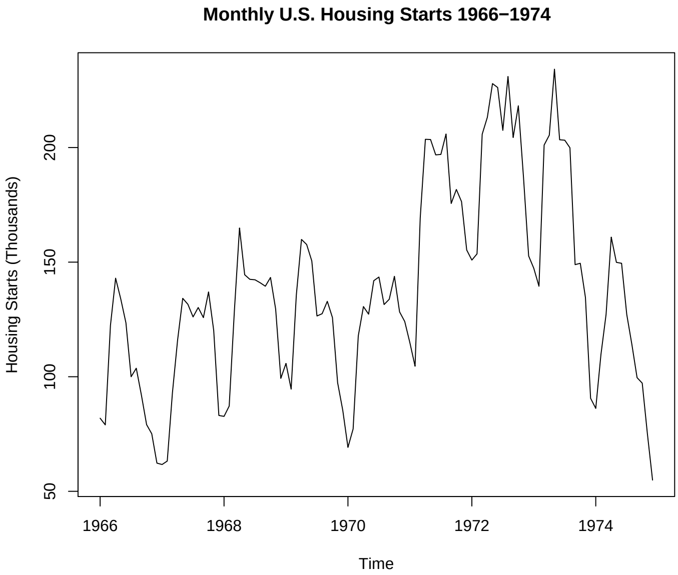
$$x_t = T_t + S_t + I_t$$

where

$$\begin{aligned} T_t &= \text{a slowly varying trend model} \\ S_t &= \text{a periodic seasonal component} \\ I_t &= \text{a set of random irregular “shocks”} \end{aligned}$$

Example – U.S. Housing Starts

- The number of housing starts in any given month is an important leading economic indicator.
- Houses are only built when there are clearly economic “good times” ahead.
- This example shows the United States housing start series from 1966 to 1974.



Monthly U.S. Housing Starts 1966−1974
Housing Starts (Thousands)
50
100
150
200
1966
1968
1970
1972
1974
Time

Interpretation

- The series clearly shows:
  - a regular seasonal variation with a peak in housing starts in summer and a trough in winter.
  - a long term (cyclical) trend.
  - short term irregularities which are not explained by the other two components.
- This is typical of monthly or quarterly economic series.

Seasonal Decomposition

- There are statistical techniques which can be used to decompose a time series into trend plus seasonal plus irregular components.
- We will use a technique called STL which uses the lowess smoother as follows.
  - A long term trend is estimated using a lowess smooth and then subtracted from the series.
  - Each month (or quarter) is smoothed separately and this seasonal effect is subtracted.
  - The remainder of the series is taken to be the irregular component.

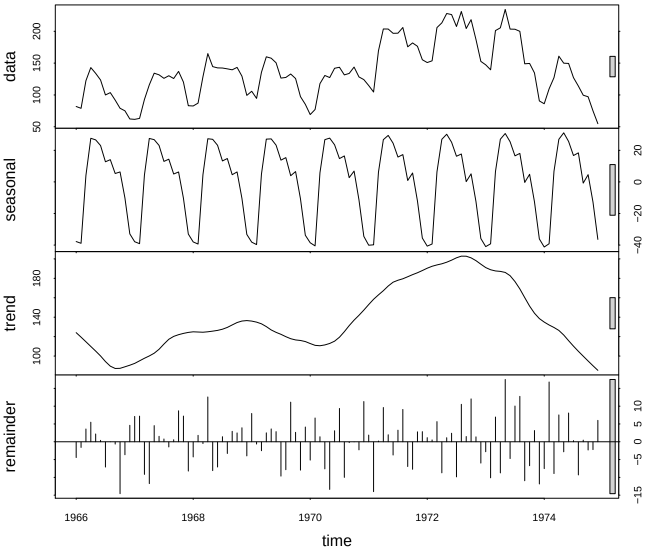
Seasonal Decomposition in R

- Assuming that we have the housing start series stored in `hstart`, here is how we carry out a seasonal decomposition and display the result graphically.

```
> library(ts)
> sd = stl(hstart, s.window=10, t.window=10)
> plot(sd)
```

- The STL procedure can be tuned in a variety of ways.

- The values of `s.window` and `t.window` determine the amount of smoothing used to determine the seasonal pattern and the trend (larger values produce more smoothing).


data
seasonal
trend
remainder
time
50
100
150
200
20
0
−20
−40
100
140
180
10
5
0
−5
−15
1966
1968
1970
1972
1974

Interpretation

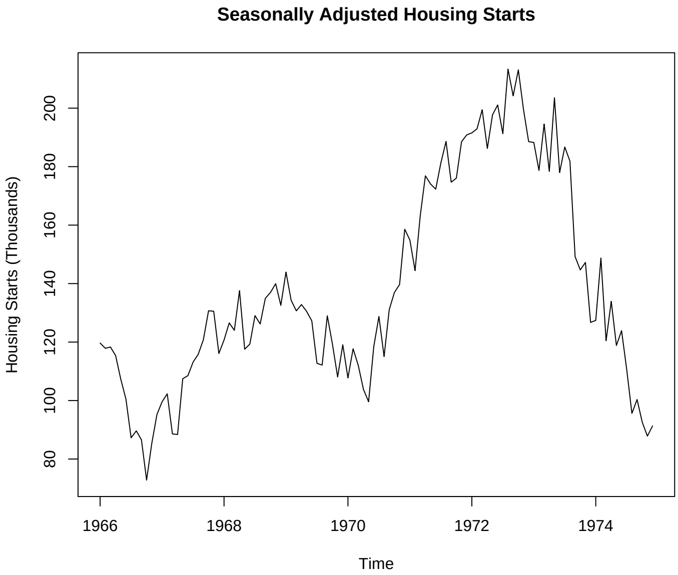
- The procedure has done a god job of decomposing the original series into interpretable subseries.
- The seasonal subseries is remarkably stable over time.
- The irregular component is far from being a “random” series.

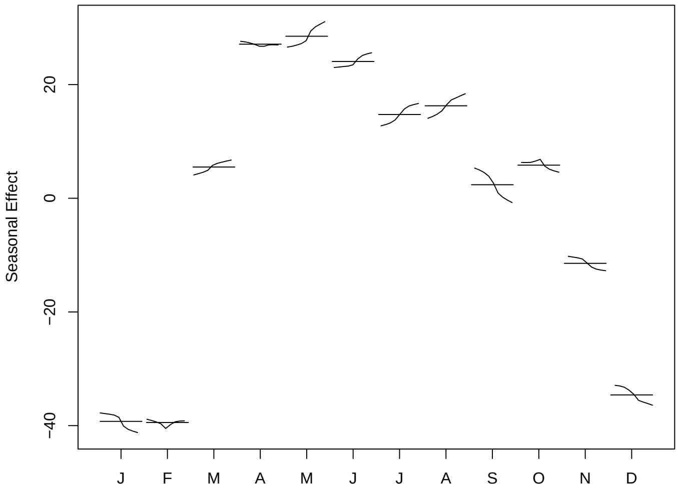
Seasonal Adjustment

- Seasonal effects ten to obscure the trends and short term variation present in a time series.
- A technique called seasonal adjustment is used to remove seasonal variation from a time series.
- A seasonally adjusted series can be obtained from the results produced by `stl`, by adding the trend and irregular components.

```
> plot(sd$time.series[,2]+sd$time.series[,3]
       main="Seasonally Adjusted Housing Starts",
       ylab="Housing Starts (Thousands)")

> monthplot(sd$time.series[,1],
            ylab="Seasonal Effect")
```


Seasonally Adjusted Housing Starts
Housing Starts (Thousands)
80
100
120
140
160
180
200
1966
1968
1970
1972
1974
Time


Seasonal Effect
20
0
−20
−40
J
F
M
A
M
J
J
A
S
O
N
D