Introduction to Time Series

Hyoeun Lee

Module 1 - Week 1

Time Series Elements (Lecture 1.1.1)

General Learning outcome of this course

Understand and model the stochastic mechanism that generates an observed time series, and predicts or forecast its future values.

Time Series Definition

Time Series: a Discrete Stochastic Process

Stochastic Process: collection of Random Variables. Denote: $\{X_t\}_{t\geq 0}$, t: time index

- E.g. Discrete-time stochastic Process
  - Humidity in Illini Union at 3pm on day t
  - Amazon stock price at closing on day t
- E.g. Continuous-time stochastic Process
  - Humidity in Illini Union at time instant t
  - Amazon stock price at time instant t

Time Series Features

- Data collected over time.
- Data cannot be assumed independent.
- Time series models incorporate **time dependence**

Many applications

- Economics (monthly unemployment figures, weekly interest rates, daily stock market values);
- Meteorology (annual precipitation and drought index, global annual temperature series);
- Agriculture (annual crop production, food consumption, cultivated area);
- Biological Sciences (cell tissue hourly growth rate, daily seed germination rate);
- Ecology (annual species abundance, monthly deforestation rate, annual carbon stock rate);
- Medicine (daily blood pressure measurements, brain MRI time series, number of annual breast cancer cases)

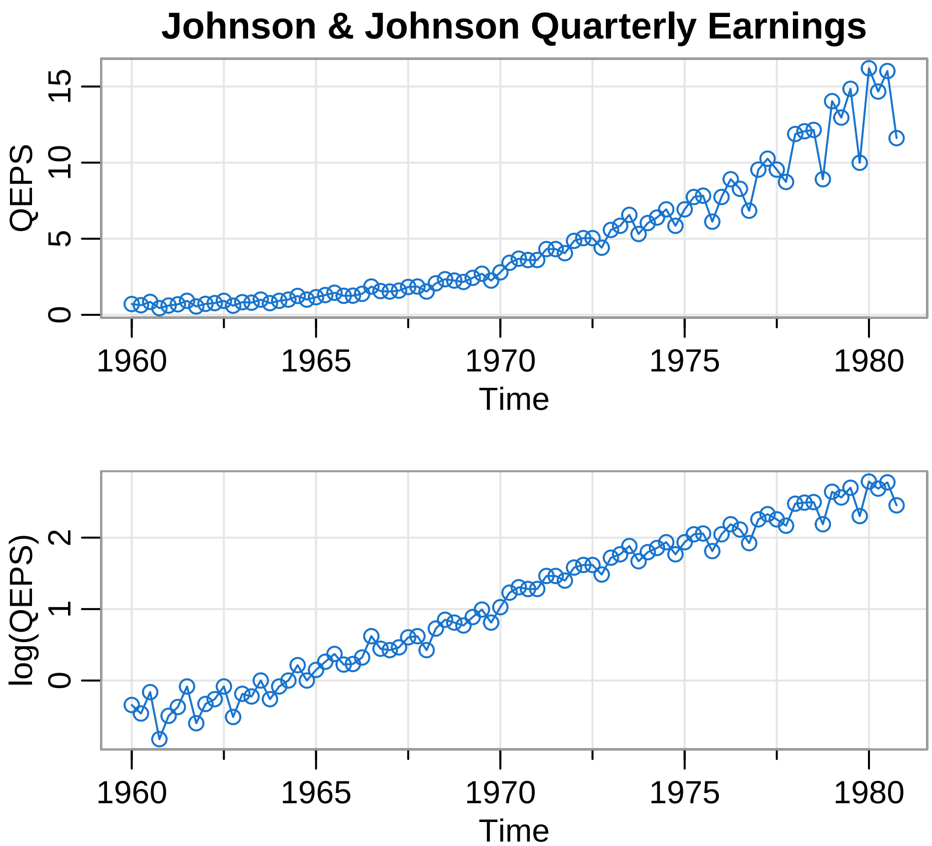
Two approaches to time series analysis:

- Time domain: Modeling a future value of a time series as a parametric function of the current and past values (Box and Jenkins, 1970 and more recently Box, Jenkins and Reinsel, 2013)
- Frequency domain: Assumes the primary feature of a time series is its periodical or systematic sinusoidal variations. These variations are naturally found in most data sets.
- The best path to analyze most data sets is to use both approaches in a complementary fashion.

Time Series Data (Lecture 1.1.2)

Example 1.1

```
par(mfrow=2:1)
tsplot(jj, ylab="QEPS", type="o", col=4, main="Johnson & Johnson Quarterly Earnings")
tsplot(log(jj), ylab="log(QEPS)", type="o", col=4)
```



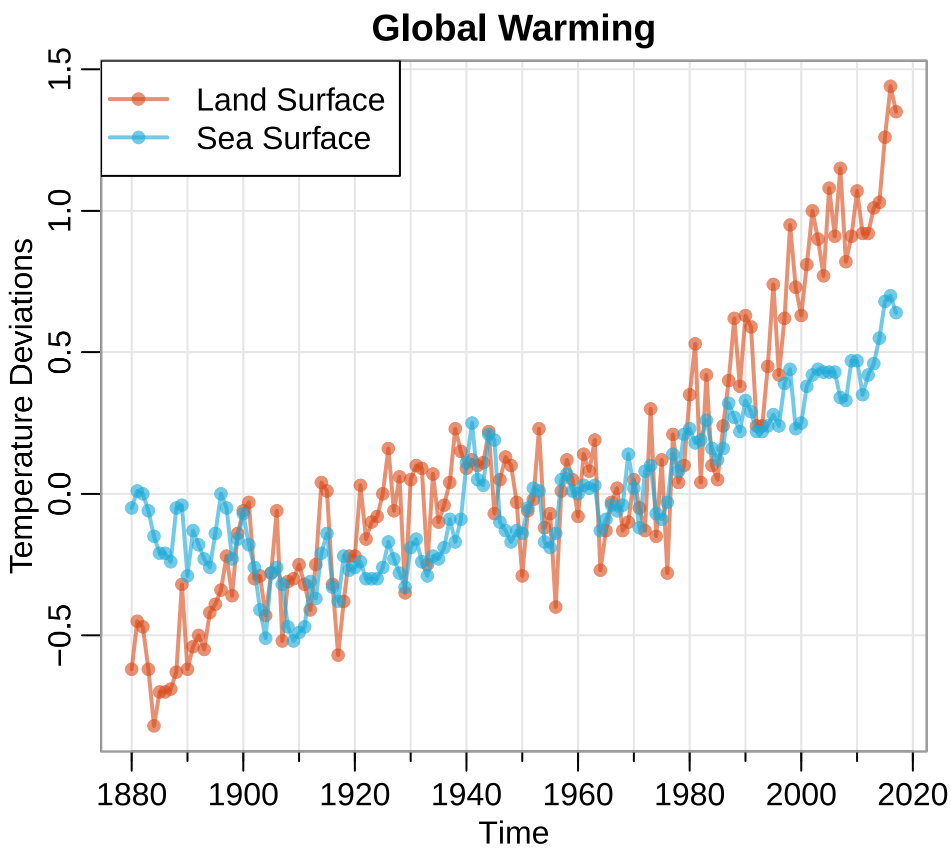

```
# ?jj #R code to load information
```

primary patterns?

- increasing underlying trend and variability
- regular oscillation

Example 1.2

```
culer = c(rgb(217,77,30,160,max=255), rgb(30,170,217,160,max=255))
tsplot(gtemp_land, col = culer[1], lwd=2, type="o", pch=20,
       ylab="Temperature Deviations", main="Global Warming")
lines(gtemp_ocean, col=culer[2], lwd=2, type="o", pch=20)
legend("topleft", col=culer, lty=1, lwd=2, pch=20, legend=c("Land Surface", "Sea Surface"), bg="white")
```

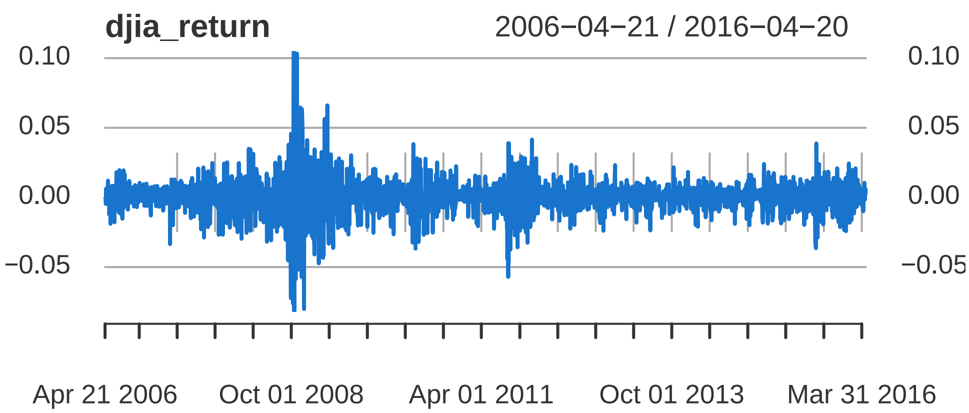


primary patterns?

- upward trend in both series from 1960-2017 (climate change)
- trend is not linear (sharp upward trend during 1960-2017)

Example 1.3

```r
djia_return = diff(log(djia$Close))[-1] # R_t = log(X_t) - log(X_{t-1})
par(mfrow=2:1)
plot(djia$Close, col=4)
plot(djia_return, col=4)
```



- X_t: DJIA closing value on day t
- R_t, return, percent of change

$$
\begin{aligned}
R_t &= (X_t - X_{t-1})/X_{t-1} && (1)\\
X_{t-1}(1+R_t) &= X_t, \quad 1+R_t = X_t/X_{t-1} && (2)\\
& && (3)
\end{aligned}
$$

We approximate R_T with $\log(1+R_t)$, so we say

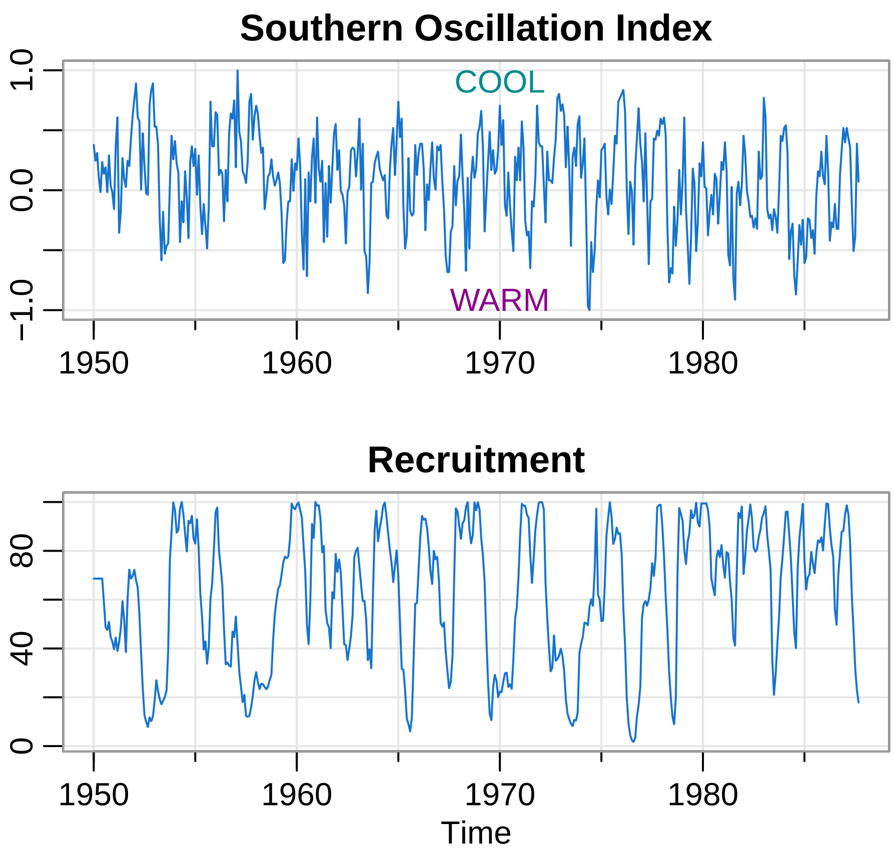
$$R_t \approx log(X_t) - log(X_{t-1})$$

primary patterns?

- Around 2008 (financial crisis), return data volatility cluster
- how to forecast volatility of future return: common problem for financial data

Example 1.4

```
par(mfrow = c(2,1))
tsplot(soi, ylab="", xlab="", main="Southern Oscillation Index", col=4)
text(1970, .91, "COOL", col="cyan4")
text(1970,-.91, "WARM", col="darkmagenta")
tsplot(rec, ylab="", main="Recruitment", col=4)
```



primary patterns?

- annual cycle
- slower frequency of three to seven years (maybe hard to see here)
- Related to seasonal models

Time Series Models

- Why do we model? To reduce uncertainty.
- There is a probability law that governs each stochastic process.
- A model is an approximation of such a probability law.
- A model is estimated (usually) using past data.
- Three statistics impotant for modeling: Mean, Variance, Dependence

Model Building Strategy

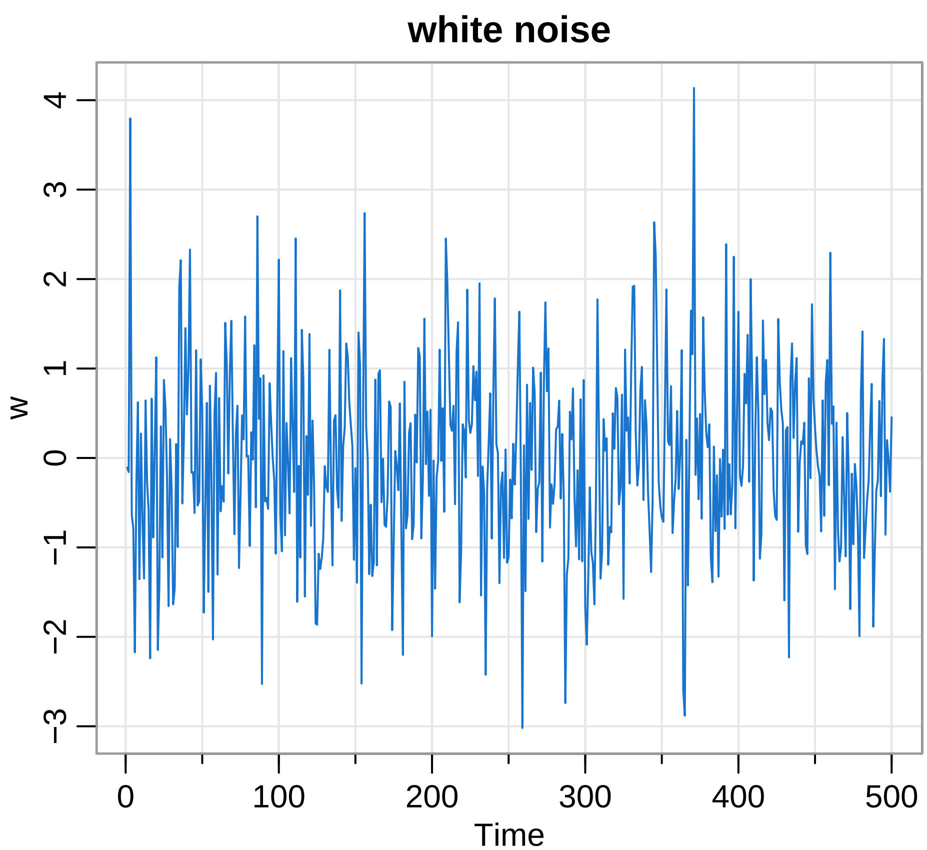
1. Model Specification: Select the appropriate model for the observed data set
2. Model Fitting: Finding the best estimates of model parameters
3. Model Diagnostics: How well does the model fit the Data?

Some Basic Models

White Noise Model

- Collection of uncorrelated random variables, W_t.
- W_t: Random variable with mean 0 and finite variance σ_W^2.
- Notation: $W_t \sim wn(0, \sigma_W^2)$
- Used as a model for noise in engineering applications

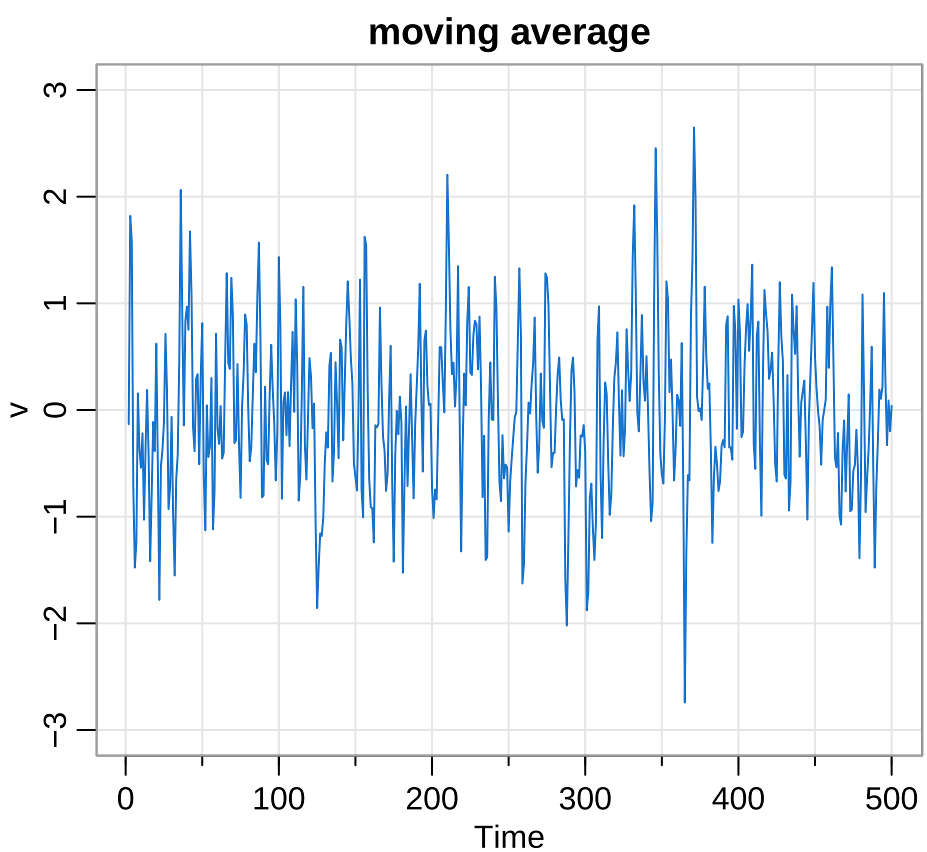
```r
w = rnorm(500) # 500 N(0,1) variates
tsplot(w, col=4, main="white noise")
```



Moving Average (MA) Model

- Moving average of white noise
- $X_t = \frac{1}{2}(W_{t-1} + W_t)$
- Interpret the time series as linear combination of white noise
- could be used for smoothing (in Section 3.3)

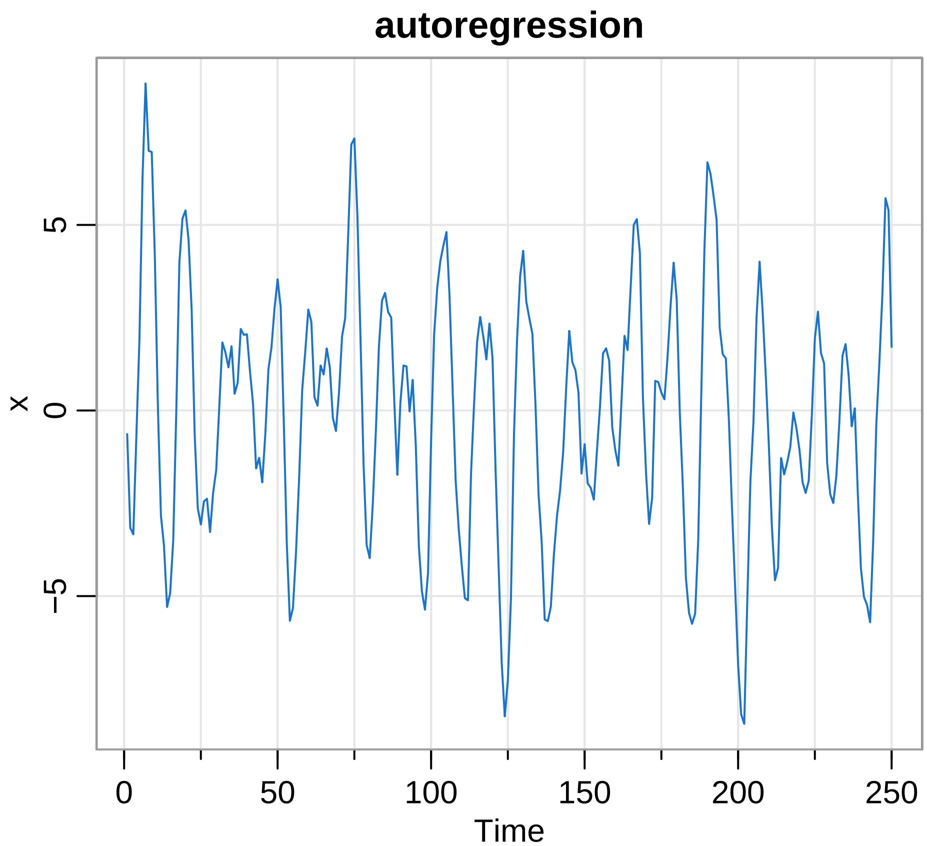
```
v = filter(w, sides=1, filter=rep(1/2,2)) # moving average
tsplot(v, ylim=c(-3,3), col=4, main="moving average")
```



Autoregression (AR) Model

- $X_t = 1.5X_{t-1} - 0.75X_{t-2} + W_t$
- Regression or prediction of the current value (X_t) as a function of the past two values.
- periodic behavior

```
set.seed(90210)
w = rnorm(250 + 50) # 50 extra to avoid startup problems
x = filter(w, filter=c(1.5,-.75), method="recursive")[-(1:50)]
tsplot(x, main="autoregression", col=4)
```

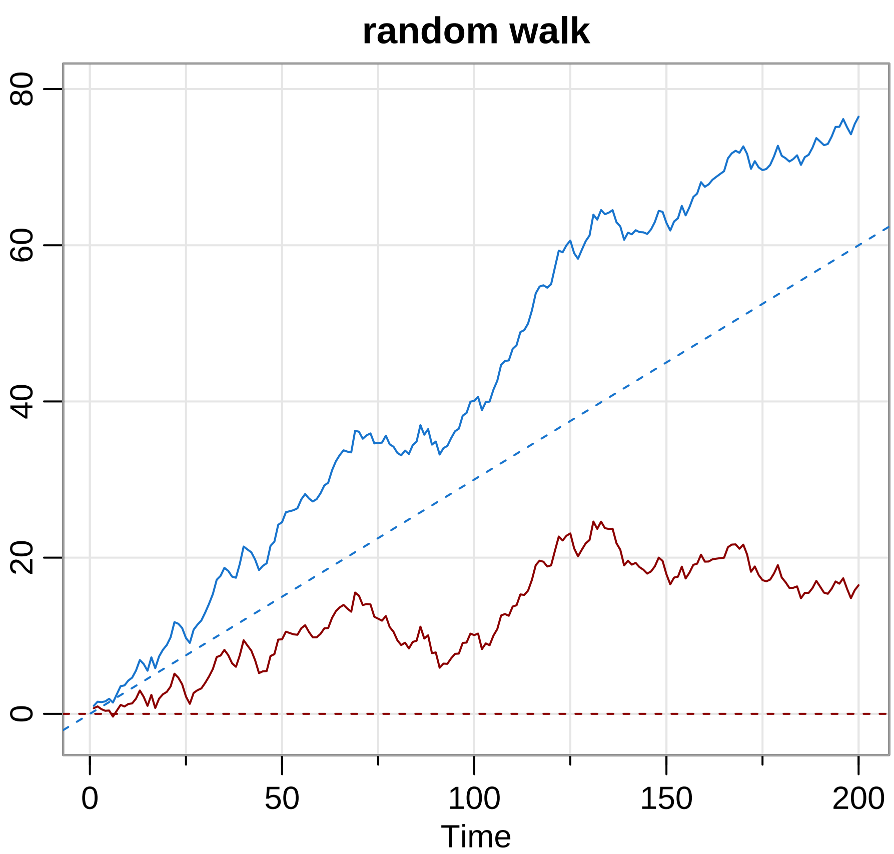


Random Walk with drift

- $X_t = \delta + X_{t-1} + W_t$
- Random walk: value at time t is the value of the series at time $t-1$ plus a completely random movement determined by W_t.
- δ: drift
- With initial condition $x_0 = 0$,

$$X_t = \delta t + \sum_{j=1}^{t} W_j$$

```
set.seed(314159265) # so you can reproduce the results
w  = rnorm(200)
x  = cumsum(w)
wd = w +.3
xd = cumsum(wd)
tsplot(xd, ylim=c(-2,80), main="random walk", ylab="", col=4)
abline(a=0, b=.3, lty=2, col=4) # drift
lines(x, col="darkred")
abline(h=0, col="darkred", lty=2)
```



Signal Plus noise

- $X_t = 2\cos(2\pi\frac{t+15}{50}) + W_t$
- $X_t = A\cos(2\pi\omega t + \phi) + W_t$
- A: amplitude, ω: frequency, ϕ: phase shift.

```
cs = 2*cos(2*pi*(1:500)/50 + .6*pi)
w  = rnorm(500,0,1)
par(mfrow=c(3,1), mar=c(3,2,2,1), cex.main=1.5)    # help(par) for info
tsplot(cs, col=4, ylab="", main = expression(x[t]==2*cos(2*pi*t/50+.6*pi)))
tsplot(cs + w, col=4,ylab="", main = expression(x[t]==2*cos(2*pi*t/50+.6*pi)+N(0,1)))
tsplot(cs + 5*w, col=4,ylab="", main = expression(x[t]==2*cos(2*pi*t/50+.6*pi)+N(0,25)))
```