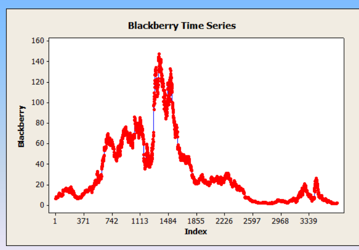
ARIMA MODELING OF TIME SERIES IN R

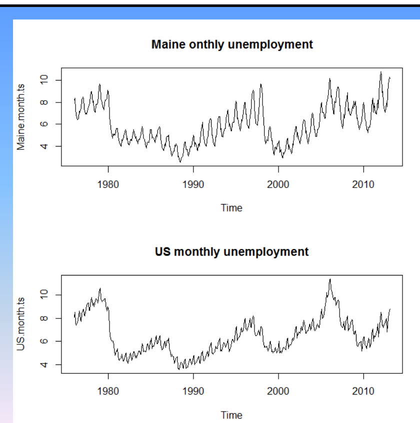


Time Series = an ordered sequence of values of a quantitative random variable at equally spaced time points (e.g., monthly time series of coin-in)

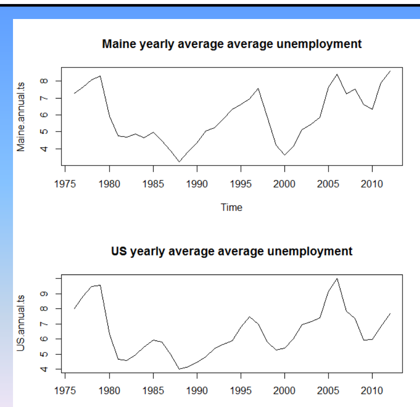
Applications:
Economic forecasting
Demand forecasting
Stock prices forecasting
Sales projections

Example 1: The data file Maine.csv has monthly unemployment figures for Maine and U.S.
(Use R-code time series Ex1.txt file)

```
maine <-read.csv("Stats24x7/R/Maine.csv")

# create a time series object
Maine.month.ts <- ts(maine$Maine,start=c(1976,1),freq=12)
US.month.ts <- ts(maine$US,start=c(1976,1),freq=12)

# plot the two monthly series on one page
layout(1:2)
plot(Maine.month.ts,main = "Maine onthly unemployment")
plot(US.month.ts,main = "US monthly unemployment")
```


```
# aggregate over a year
Maine.annual.ts <- aggregate(Maine.month.ts)/12
US.annual.ts <- aggregate(US.month.ts)/12

# plot the two annual time series
plot(Maine.annual.ts, main = "Maine yearly average
average unemployment")
plot(US.annual.ts, main = "US yearly average average
unemployment")
```

Nonseasonal ARIMA Modeling

- A time series is stationary if: $E(Y_t) = \mu,\ Var(Y_t) = \sigma^2$ for all t

In other words, if $y_1, y_2, \ldots, y_n$ values of the time series fluctuate around a constant mean with constant variation, the time series is stationary (Figure 1(b), next slide)

If the y - values do not seem to fluctuate around a constant mean or do not fluctuate with constant variation around a constant mean, then it is non-stationary (Figure 1(a), next slide)

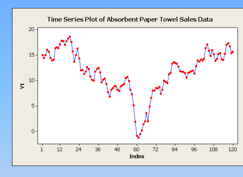


Figure 1(a): non-stationary time series

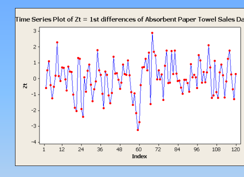
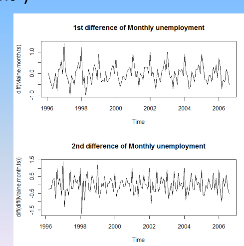


Figure 1(b): stationary time series

If a time series is non-stationary, 1st order difference of the time series is calculated. If the original time series exhibits a linear trend, and var(Y_t) is constant, 1st difference will yield a stationary time series.

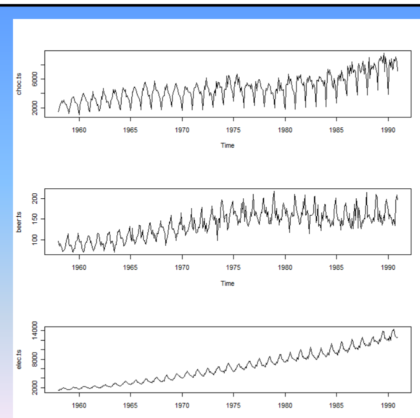
If var(Y_t) is non-constant, a log-transform or a square-root transform may yield a time series with constant variance.

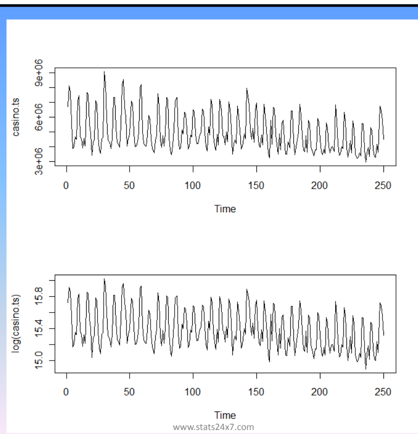

```
# plot 1-st and 2-nd order differences of the monthly time series
plot(diff(Maine.month.ts), main = "1st difference of Monthly unemployment")

plot(diff(diff(Maine.month.ts)), main = "2nd difference of Monthly unemployment")
```



1st difference appears to be stationary.

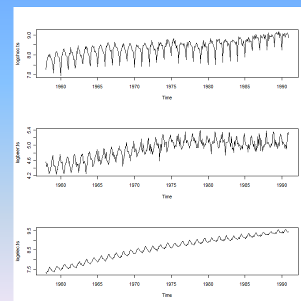

```
# Example 2
www <- "http://www.massey.ac.nz/~pscowper/ts/cbe.dat"
CBE <- read.table(www, header=T)
names(CBE)
#[1] "choc" "beer" "elec"

choc.ts <- ts(CBE[,1], start=1958, freq=12)
beer.ts <- ts(CBE[,2], start=1958, freq=12)
elec.ts <- ts(CBE[,3], start=1958, freq=12)
plot(choc.ts)
plot(beer.ts)
plot(elec.ts)
```



NOTE: each of the three time series is non-stationary (non-constant mean, non-constant variance)

```
logchoc.ts <-
log(choc.ts)
logbeer.ts <- log(beer.ts)
logelec.ts <- log(elec.ts)
plot(logchoc.ts)
plot(logbeer.ts)
plot(logelec.ts)
```



ARIMA MODEL IDENTIFICATION

$\{y_1, y_2, ..., y_n\} \xrightarrow{differencing} \{z_b, z_{b+1}, ..., z_n\}$, working series

We look at the SAC and SPAC of the working series to identify a Box-Jenkins model.

Two commonly used Box-Jenkins models are

1) Non-seasonal autoregressive (AR) model of order 1

$z_t = \phi_1 z_{t-1} + a_t$ where $a_t =$ random shock~$N(0, \sigma^2)$, independent

2) Non-seasonal moving average (MA) model of order 1

$z_t = a_t - \theta_1 a_{t-1}$

Sample Autocorrelation Function (ACF) at lag K is defined as:

$$r_k = Corr\left(z_b, z_{b+k}\right) = \frac{\sum_{t=b}^{n-k}\left(z_t - \bar{z}\right)\left(z_{t+k} - \bar{z}\right)}{\sum_{t=b}^{n}\left(z_t - \bar{z}\right)^2}$$

$$where\ \bar{z} = \frac{\sum_{t=b}^{n} z_t}{n-b+1}$$

The standard error of r_k is

$$s_{r_k} = \begin{cases} \sqrt{\dfrac{1}{n-b+1}} & \text{if } k = 1 \\ \sqrt{\dfrac{1 + 2\sum_{j=1}^{k-1} r_j^2}{n-b+1}} & \text{if } k = 2, 3, ... \end{cases}$$

The $t-value$ is given by $t_{r_k} = \dfrac{r_k}{s_{r_k}}$

A spike exists at lag k if $|t_{r_k}| > 2$.

SAC cuts off after lag k if there are no spikes at lags $> k$.

The Sample Partial Autocorrelation (PACF) Function is defined as:

$$r_{KK} = \begin{cases} r_1 & K = 1 \\ \dfrac{r_K - \sum_{j=1}^{K-1} r_{K-1,j}\, r_{K-j}}{1 - \sum_{j=1}^{K-1} r_{K-1,j}\, r_j} & \end{cases}$$

$$where\ r_{Kj} = r_{K-1,j} - r_{KK} r_{K-1,K-j},\ j = 1, 2, .., K-1.$$

It's standard error is: $s_{r_{KK}} = \dfrac{1}{\sqrt{n-b+1}}$

and the t-statistic is: $t_{r_{KK}} = \dfrac{r_{KK}}{s_{r_{KK}}}$

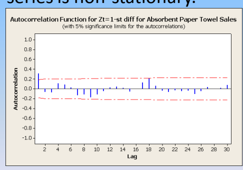
The SPAC function is a graph of r_{KK} vs. K.

r_{KK} = SAC at lag K with the effect of intervening observations eliminated.

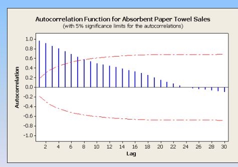
The SAC function is a graph of r_k vs. $k(lag)$

NOTE: If SAC of $z_b, z_{b+1}, \cdots z_n$ either cuts off or dies down very quickly, then the time series is stationary.

If the SAC dies down extremely slowly, then the time series is non-stationary.



SAC cuts off quickly (after lag k = 1) as r_k is not significantly different from 0 for k 1; working series is stationary.



SAC cuts off very slowly (after lag k 7) as r_k is significantly different from 0 for $k \le 7$; time series is non-stationary.

If $|t_{r_{KK}}| > 2$ we say that a spike at lag K exists in the time series.

If there are no spikes in the time series at lagK in SPAC, then we say that SPAC cuts off after lag K.

If SPAC does not cut off but decreases steadily, we say that SPAC dies down.

NOTE:1. For nonseasonal data, if SPAC cuts off it will do so (typically) for $K \le 2$

2. The behavior of SPAC helps us to identify Box-Jenkins models.

Identification of ARIMA(p,d,q) Model

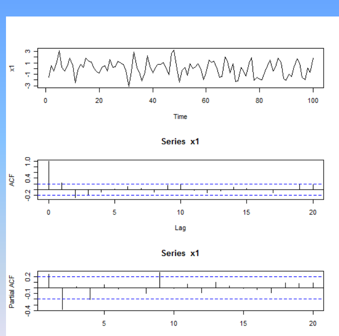
- If ACF dies down and PACF has spike at lag 2, cuts off past lag 2, then try to fit ARIMA(2,0,0)

  if there is no trend, and try ARIMA(2,1,0) if there is a trend.

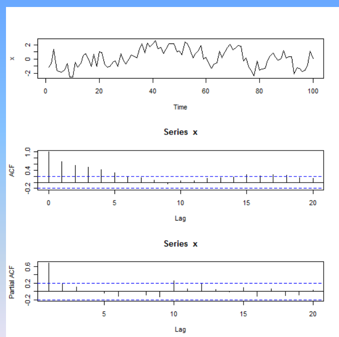
- If PACF dies down and ACF has spike at lag 2, cuts off past lag 2, then try to fit ARIMA(0,0,2)

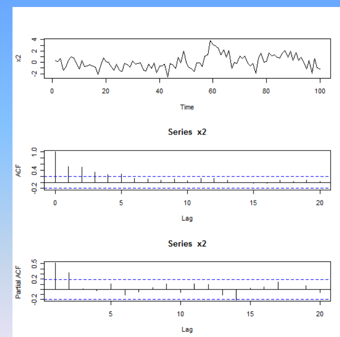
  if there is no trend, and try ARIMA(0,1,2) if there is a trend.


```
# simulate ARMA(0,0,2) time series
 x1 <- arima.sim(list(ma=c(0.6,-.4)), n=100)
 layout(1:3)
 plot(x1)
 acf(x1)
 pacf(x1)

# simulate ARMA(2,0,0) time series
 x2 <- arima.sim(list(ar=c(0.4,.4)), n=100)

 layout(1:3)
 plot(x2)
 acf(x2)
 pacf(x2)
```

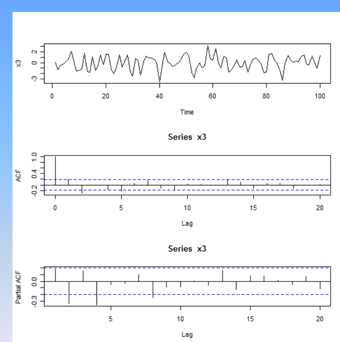



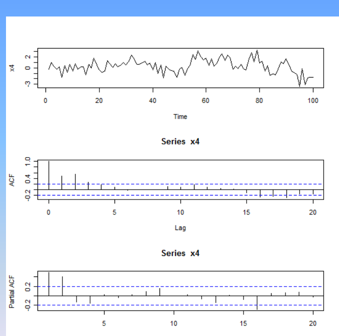

```
# simulate ARMA(1,0,1) time series
 x = arima.sim(list(order=c(1,0,1), ar=.9, ma=-.5), n=100)
 layout(1:3)
 plot(x)
 acf(x)
 pacf(x)
```






```
# simulate ARMA(0,0,2) time series
 x3 <- arima.sim(list(ma=c(0.8,-.4)), n=100)
 layout(1:3)
 plot(x3)
 acf(x3)
 pacf(x3)

# simulate ARMA(2,0,0) time series
 x4 <- arima.sim(list(ar=c(0.3,.5)), n=100)

 layout(1:3)
 plot(x4)
 acf(x4)
 pacf(x4)
```



Example 3 (ARIMA MODELING OF Brett's Slot handle data)

```
casino <- read.csv("K:/Brett Abarbanel/Brett_Nov3_2009.csv",
header=TRUE)
 names(casino)
 [1] "date"            "month"           "Bingo.Win"
 [4] "Bingo.Write"     "Keno.Win"        "Keno.Write"
 [7] "Race.Win"        "Race.Write"      "Slot.Handle"
[10] "Slot.Win"        "Sports.Win"      "Sports.Write"
[13] "Table.Games.Drop" "Table.Games.Win" "Poker.Rake"
[16] "daytext"         "day"             "New.Years"
[19] "MLK"             "Presidents"      "St.Patricks"
[22] "Memorial.Day"    "Indep.Day"       "Labor.Day"
[25] "Superbowl"       "NBA.All.Star.Game" "March.Madness"
[28] "Golf.Masters"    "Kentucky.Derby"  "Preakness"
[31] "Indy.500"        "NBA.Finals"      "Belmont"
[34] "US.Open"         "Wimbledon.Finals" "MLB.All.Star.Game"
```

```
attach(casino)
DFeb<-as.numeric(month==2)
DMar<-as.numeric(month==3)
DApr<-as.numeric(month==4)
DMay<-as.numeric(month==5)
DJun<-as.numeric(month==6)
DJul<-as.numeric(month==7)
DAug<-as.numeric(month==8)
DSep<-as.numeric(month==9)
DTues<-as.numeric(day==2)
DWeds<-as.numeric(day==3)
DThurs<-as.numeric(day==4)
DFri<-as.numeric(day==5)
DSat<-as.numeric(day==6)
DSun<-as.numeric(day==7)
```



```
casino.out1<-
lm(Slot.Handle~Sports.Write+DFeb+DMar+DApr+DMay+DJun
+DJul+DAug+DSep+DTues+DWeds+DThurs+DFri+DSat+DSun+
trend+New.Years+MLK+Presidents+St.Patricks+Memorial.Day
+Indep.Day+Labor.Day+Superbowl+NBA.All.Star.Game+March
.Madness+Golf.Masters+Kentucky.Derby+Preakness+Indy.500
+NBA.Finals+Belmont+US.Open+Wimbledon.Finals+
MLB.All.Star.Game+trend)
```


	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	4.220e+06	1.423e+05	29.660	< 2e-16 ***
Sports.Write	7.191e-01	1.139e+00	0.631	0.528634
DFeb	2.380e+05	1.630e+05	1.460	0.145780
DMar	-5.693e+04	2.353e+05	-0.242	0.809080
DApr	1.961e+05	3.423e+05	0.573	0.567305
DMay	3.114e+05	4.351e+05	0.716	0.474959
DJun	1.693e+05	5.512e+05	0.307	0.759006
DJul	-1.649e+05	6.522e+05	-0.253	0.800684
DAug	-3.380e+05	7.545e+05	-0.448	0.654663
DSep	3.532e+04	8.423e+05	0.042	0.966594
DTues	-7.106e+04	1.094e+05	-0.649	0.516772
DWeds	6.077e+05	1.083e+05	5.613	6.13e-08 ***
DThurs	4.838e+05	1.094e+05	4.422	1.55e-05 ***
DFri	2.892e+06	1.090e+05	26.526	< 2e-16 ***
DSat	2.560e+06	1.164e+05	21.987	< 2e-16 ***
DSun	7.538e+05	1.148e+05	6.567	3.82e-10 ***
trend	-3.341e+03	3.546e+03	-0.942	0.347120

New.Years	1.037e+06	2.772e+05	3.741	0.000235 ***
MLK	6.186e+05	2.844e+05	2.175	0.030715 *
Presidents	1.667e+06	3.316e+05	5.027	1.05e-06 ***
St.Patricks	8.706e+05	4.585e+05	1.899	0.058967 .
Memorial.Day	9.129e+05	2.766e+05	3.300	0.001133 **
Indep.Day	5.063e+05	4.616e+05	1.097	0.273946
Labor.Day	9.732e+05	3.472e+05	2.803	0.005520 **
Superbowl	3.557e+05	8.665e+05	0.410	0.681884
NBA.All.Star.Game	-3.817e+04	5.535e+05	-0.069	0.945089
March.Madness	2.012e+05	1.973e+05	1.020	0.308899
Golf.Masters	-1.866e+05	1.958e+05	-0.953	0.341714
Kentucky.Derby	-2.731e+05	4.642e+05	-0.588	0.556956
Preakness	-6.606e+05	4.607e+05	-1.434	0.153075
Indy.500	1.049e+06	5.229e+05	2.006	0.046160 *
NBA.Finals	-2.025e+05	2.381e+05	-0.851	0.395834
Belmont	-3.822e+05	4.661e+05	-0.820	0.413162
US.Open	-2.747e+05	2.009e+05	-1.368	0.172901
Wimbledon.Finals	-2.122e+05	5.531e+05	-0.384	0.701686
MLB.All.Star.Game	1.505e+04	4.598e+05	0.033	0.97392


```
casino.out2 <-
lm(Slot.Handle~Sports.Write+DWeds+DThurs+DFri+DSat+DSun+t
rend+New.Years+Presidents+Memorial.Day+Labor.Day+Indy.500+
trend)
 summary(casino.out)

Call:
lm(formula = Slot.Handle ~ Sports.Write + DWeds + DThurs + DFri
+
   DSat + DSun + trend + New.Years + Presidents + Memorial.Day
+
   Labor.Day + Indy.500 + trend)

Residuals:
    Min      1Q        Median       3Q       Max
-1063415  -280747   -7819       260579  1841288
```


	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	4.399e+06	8.657e+04	50.813	< 2e-16 ***
Sports.Write	1.346e+00	6.181e-01	2.177	0.03046 *
DWeds	6.279e+05	9.596e+04	6.543	3.68e-10 ***
DThurs	4.952e+05	9.564e+04	5.177	4.80e-07 ***
DFri	2.925e+06	9.557e+04	30.604	< 2e-16 ***
DSat	2.549e+06	9.781e+04	26.058	< 2e-16 ***
DSun	7.635e+05	1.008e+05	7.572	8.18e-13 ***
trend	-4.936e+03	4.410e+02	-11.192	< 2e-16 ***
New.Years	7.367e+05	2.473e+05	2.979	0.00319 **
Presidents	1.757e+06	2.738e+05	6.418	7.46e-10 ***
Memorial.Day	1.215e+06	2.711e+05	4.481	1.16e-05 ***
Labor.Day	1.174e+06	2.765e+05	4.246	3.13e-05 ***
Indy.500	1.061e+06	5.429e+05	1.954	0.05193 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 463000 on 237 degrees of freedom
Multiple R-squared: 0.8836, Adjusted R-squared: 0.8777

```
# fit ARIMA models with predictors

# ARIMA (0,0,1) + linear trend + Sports.Write
x1 <-
cbind(Sports.Write,DWeds,DThurs,DFri,DSat,DSun,trend,New.Years,Presidents,Memorial.Day,Labor.Day,Indy.500)
fit1 <- arima(Slot.Handle, order=c(0,0,1), xreg=x1)
print(fit1)

# ARIMA (0,0,1) + standardized linear and quadratic trend + Sports.Write
x1a <-
cbind(Sports.Write,DWeds,DThurs,DFri,DSat,DSun,t,t2,New.Years,Presidents,Memorial.Day,Labor.Day,Indy.500)
fit1a <- arima(Slot.Handle, order=c(0,0,1), xreg=x1a)
print(fit1a)

# ARIMA (0,0,1) + standardized linearand quadratic trend - Sports.Write
x1b <-
cbind(DWeds,DThurs,DFri,DSat,DSun,t,t2,New.Years,Presidents,Memorial.Day,Labor.Day,Indy.500)
fit1b <- arima(Slot.Handle, order=c(0,0,1), xreg=x1b)
print(fit1b)
```


ARIMA (0,0,1)
With
All significant terms
+
Sports.Write
+
linear trend

Term	Beta	se
ma1	0.2826	0.0571
intercept	4439595	94984
Sports.Write	0.6988	0.5689
DWeds	646142	87269
DThurs	499265	96643
DFri	2929980	96619
DSat	2574468	98427
DSun	784965	91813
trend	-5011.39	518.41
New.Years	859259	275635
Presidents	1669062	295976
Memorial.Day	1170025	275329
Labor.Day	1026819	308186
Indy.500	1112333	445231
sigma^2	1.87E+11	
log-likelihood	-3598.72	
aic	7227.45	

ARIMA (0,0,1)
With
All significant terms
+
Sports.Write
+
Standardized linear and quadratic trend

Term	Beta	se
ma1	0.2442	0.0601
intercept	3936643	73019
Sports.Write	0.8087	0.5663
DWeds	642300	85566
DThurs	497532	93338
DFri	2931036	93253
DSat	2564167	95127
DSun	775154	90061
t	-358687	35709
t2	-135689	40295
New.Years	1096579	273904
Presidents	1714554	285956
Memorial.Day	1068027	268841
Labor.Day	1299666	304998
Indy.500	1.10E+06	443738
sigma^2	1.79E+11	
log-likelihood	-3593.39	
aic	7218.78	

ARIMA (0,0,1)
With
All significant terms
-
Sports.Write
+
Standardized linear and quadratic trend

Term	Beta	se
ma1	0.2578	0.0569
intercept	3968382	70425
DWeds	648521	85739
DThurs	509855	93743
DFri	2942516	93737
DSat	2593873	93659
DSun	815316	85904
t	-369097	35473
t2	-133567	40912
New.Years	1162314	273478
Presidents	1685484	287936
Memorial.Day	1086857	270935
Labor.Day	1318221	309206
Indy.500	1054593	441887
sigma^2	1.80E+11	
log-likelihood	-3.59E+03	
aic	7218.83	

ARIMA (0,1,2)
With
All significant terms
+
Sports.Write
-
trend

Term	Beta	se
ma1	-0.6807	0.0604
ma2	-0.2286	0.0583
Sports.Write	0.7787	0.5677
DWeds	640809	85335
DThurs	494538	93224
DFri	2929273	93228
DSat	2568907	95091
DSun	778732	89898
New.Years	1112150	291885
Presidents	1674176	295669
Memorial.Day	994610	279706
Labor.Day	1071935	320842
Indy.500	1082424	441810
sigma^2	1.91E+11	
log-likelihood	-3587.91	
aic	7203.81	

ARIMA (0,1,2)
With
All significant terms
-
Sports.Write
-
trend

Term	Beta	se
ma1	-0.664	0.0572
ma2	-0.2441	0.0554
DWeds	646768	85474
DThurs	506057	93750
DFri	2940076	93785
DSat	2597441	93692
DSun	817287	85647
New.Years	1174806	291913
Presidents	1651287	298002
Memorial.Day	1007697	282176
Labor.Day	1087848	323979
Indy.500	1043512	438759
sigma^2	1.92E+11	
log-likelihood	-3588.86	
aic	7203.71	