13. SARIMA models (Seasonal ARIMA) ([BD] §6.5)

SARIMA is a modification of ARIMA to account for seasonal and non-stationary behavior: if the data has seasonal behavior at lag s, the dependence on the past occurs most strongly at multiples of s. Thus, SARIMA introduces terms that identify with the seasonal lags.

Definition. Let d and D be non-negative integers.

$\{X_t\}$ is a seasonal ARIMA (SARIMA) $(p, d, q) \times (P, D, Q)$ process with period s if the differenced process $Y_t := (1-B)^d(1-B^s)^D X_t$ is a causal ARMA process defined by

$$\phi(B)\Phi(B^s)Y_t = \theta(B)\Theta(B^s)Z_t, \;\; Z_t \sim WN(0, \sigma_Z^2),$$

where $\phi(z) = 1 - \phi_1 z - \ldots - \phi_p z^p$, $\Phi(z) = 1 - \Phi_1 z - \ldots - \Phi_P z^P$, $\theta(z) = 1 + \theta_1 z + \ldots + \theta_q z^q$, $\Theta(z) = 1 + \Theta_1 z + \ldots + \Theta_Q z^Q$. Recall that Y_t is causal if and only if $\phi(z) \neq 0, \Phi(z) \neq 0$ for $|z| \leq 1$.

How does SARIMA process arise?

Suppose that time series X_t has a period s, for example, $s = 12$ for monthly data, $s = 4$ for quarterly data, $s = 24$ for hourly data, etc.

Thus , the time series $X_1, X_2, \ldots, X_n$ can be viewed as s series:
$X_j, X_{j+s}, X_{j+2s}, \ldots, X_{j+(r-1)s}$, $j = 1, 2, \ldots, s$.

For example, $j = 1$ may correspond to January data for r years, $j = 2$ may correspond to February data for r years, etc.:

January 1980 X_1	February 1980 X_2	March 1980 X_3	...	December 1980 X_{12}
January 1981 X_{13}	February 1981 X_{14}	March 1981 X_{15}	...	December 1981 X_{24}
...	...	...	...	...
January 2016 X_{433}	February 2016 X_{434}	March 2016 X_{435}	...	December 2016 X_{444}
$\downarrow$ X_1, X_{13}, X_{25},...	$\downarrow$ X_2, X_{14}, X_{26},...	$\downarrow$ X_3, X_{15}, X_{27},...	$\downarrow$ ...	$\downarrow$ ...

We thus have a total of s=12 series (only January or only February, etc), each has r=37 entries.

Model Assumption 1: Each of these s series (that is, series taken for each month j) is generated by **the same ARMA(P, Q)** process:

$$X_{j+st} - \Phi_1 X_{j+s(t-1)} - \ldots - \Phi_P X_{j+s(t-P)} = U_{j+st} + \Theta_1 U_{j+s(t-1)} + \ldots + \Theta_Q U_{j+s(t-Q)}, (U_{j+st})_{t=0,1,2,\ldots} \sim WN(0, \sigma_U^2).$$

Using shift operator B, rewrite the model as

$$(1 - \Phi_1 B^s - \ldots - \Phi_P B^{sP})X_t = (1 + \Theta_1 B^s + \ldots + \Theta_Q B^{sQ})U_t,$$

(because for $t = j + s\tau$, $B^{sP}X_t = X_{(j+s\tau)-sP} = X_{j+s(\tau-P)}$)

Between-Year Model Summary: $\Phi(B^s)X_t = \Theta(B^s)U_t$,

Example: $s = 12, P = 1, Q = 0$. $(1 - \Phi_1 B^{12})X_t = U_t$ or $X_t - \Phi_1 X_{t-12} = U_t$.

If for different j (for different months) U_t's are uncorrelated (i.e. the s series are uncorrelated–not a very realistic assumption!), the sample ACF and PACF are as follows:

P=1, Q=0, s=12, $X_t - \Phi_1 X_{t-12} = U_t$: $\rho_{12k} = \Phi_1^k, k = 1, 2, \ldots$.
P=0, Q=1, s=12, $X_t = U_t + \Theta_1 U_{t-12}$: $\rho_{12} > 0$, the rest are zeros.

.....

Explanation: For $X_t = U_t + \Theta_1 U_{t-12}$,

$\rho(h) = E[(U_t + \Theta_1 U_{t-12})(U_{t+h} + \Theta_1 U_{t-12+h})] = \rho_U(h) + 2\Theta_1 \rho_U(h-12) + \Theta_1^2 \rho_U(h) \neq 0$ *only for* $h = 0$ *and* $h = 12$.

.....

Incorporate dependence between monthly series.

Model Assumption 2: For different j's (months) U_t's are dependent and follow ARMA (p, q) process

$$\phi(B) U_t = \theta(B) Z_t,\ Z_t \sim WN(0, \sigma_Z^2).$$

Then the original time series X_t follows model (recall: $\Phi(B^s) X_t = \Theta(B^s) U_t$):

$$\phi(B)\Phi(B^s) X_t = \phi(B)\Theta(B^s) U_t = \Theta(B^s)\phi(B) U_t = \Theta(B^s)\theta(B) Z_t,\ \ Z_t \sim WN(0, \sigma_Z^2).$$

If we incorporate trend, we get $\mathbf{SARIMA}(p, d, q) \times (P, D, Q)$**:**

$$\phi(B)\Phi(B^s) Y_t = \theta(B)\Theta(B^s) Z_t,\ \ Z_t \sim WN(0, \sigma_Z^2),\ for\, Y_t := (1-B)^d (1-B^s)^D X_t.$$

Procedure to identify SARIMA:

1. Find d, D to make $Y_t = (1-B)^d (1-B^s)^D X_t$ stationary. In practice usually use $d = 1, 2$ and $D = 1$.

2. Find P and Q: look at $\hat{\rho}(ks)$, $k = 1, 2, \ldots$, i.e. look at ACF and PACF at lags which are multiples of s. Identify ARMA(P, Q).

3. Find p, q: $\hat{\rho}(1), \ldots, \hat{\rho}(s-1)$ should look as ACF of ARMA (p, q).

Note: Y_t constitutes ARMA$(p + sP, q + sQ)$ process in which some of the coefficients are zeros and the rest of the coefficients are functions of $\underline{\beta}' = (\underline{\phi}', \underline{\Phi}', \underline{\theta}', \underline{\Theta}')$.

4. Use ML Estimation for $(\underline{\beta}, \sigma_Z^2)$ and use AICC and diagnostic checking to identify the best model.

Example: Let $Q = 1, q = 1, P = p = 0, s = 12$. SARIMA$(0, 0, 1) \times (0, 0, 1)$ model for Y_t:

$$Y_t = (1 + \theta_1 B)(1 + \Theta_1 B^{12}) Z_t = Z_t + \theta_1 Z_{t-1} + \Theta_1 Z_{t-12} + \theta_1 \Theta_1 Z_{t-13}$$

This is a MA(13) model so that ACF $\rho(h) = 0$ for $h > 13$. ACF is nonzero at lags

– lag 1: $\rho(1) = EY_t Y_{t-1} = E(\ldots + \theta_1 Z_{t-1} \ldots)(Z_{t-1} + \ldots) \neq 0$,
– lag 12: $\rho(12) = EY_t Y_{t-12} = E(\ldots + \Theta_1 Z_{t-12} + \ldots)(Z_{t-12} + \ldots) \neq 0$
– lag 13: $\rho(13) = EY_t Y_{t-13} = E(\ldots + \theta_1 \Theta_1 Z_{t-13})(Z_{t-13} + \ldots) \neq 0$.

However, also at lag 11 the correlation is not zero:
– $\rho(11) = EY_t Y_{t-11} = E(Z_t + \theta_1 Z_{t-1} + \Theta_1 Z_{t-12} + \theta_1 \Theta_1 Z_{t-13})(Z_{t-11} + \theta_1 Z_{t-12} + \ldots) \neq 0$.

Example: Let $Q = q = 0, P = p = 1, s = 12$. SARIMA$(1, 0, 0) \times (1, 0, 0)$ model for Y_t:

$$(1 - \phi_1 B)(1 - \Phi_1 B^{12}) Y_t = Z_t, \text{ or}$$

$$(1 - \phi_1 B - \Phi_1 B^{12} + \phi_1 \Phi_1 B^{13}) Y_t = Z_t \text{ or } Y_t - \phi_1 Y_{t-1} - \Phi Y_{t-12} + \phi_1 \Phi_1 Y_{t-13} = Z_t$$

For example, if $\phi = .6$ and $\Phi = .5$ we have: $Y_t - .6 Y_{t-1} - .5 Y_{t-12} - (-.3) Y_{t-13} = Z_t$.

This is an AR(13) model so that PACF $\alpha(h) = 0$ for $h > 12$.
PACF has distinct spikes at lags 1, 12, 13 with a bit of action coming before lag 12.

14. Forecasting ([BD] §2.5, 3.3, 5.4, and 6.4)

Times series objective is to predict future observations, using past data. One would like to predict future interest rates, future sales, earnings, unemployment rate, GNP, etc.

Major assumption when applying time series model is that

– the future behaves as a past, i.e. that it is described by the same model.

– Conclude: check model frequently and be careful about forecasts with long lead time.

Prior to forecasting:

• Model identification, • parameter estimation, and • diagnostic checking precede forecasting.

• Thus, we assume that an adequate time series model was obtained and the maximum likelihood estimates of coefficients were obtained.

Forecasting Problem:

• Assume that we observe values $X_1 = x_1, \ldots, X_n = x_n$.
• We are interested in forecasting the future values X_{n+h}, $h = 1, 2, \ldots$.
• When $h = 1$ we get a one-step ahead predictor $\hat{X}_n$, defined in §11.1.2 (i), (iii) of LNs.

Definitions and Terminology.

The best forecast $P_n X_{n+h}$ of unobserved value X_{n+h} in terms of observed values $\{X_1, \ldots, X_n\}$ is defined as a value which minimizes the expected mean square error (m.s.e.) called **the prediction error**:

$$E[(X_{n+h} - P_n X_{n+h})^2 | X_1, \ldots, X_n] = E_n[(X_{n+h} - P_n X_{n+h})^2],$$

where
$E_n(\cdot) = E[\cdot | X_1, \ldots, X_n]$ denotes the conditional expectation, given $X_1 = x_1, \ldots, X_n = x_n$.

The best h-step ahead forecast = m.s.e. predictor $P_n X_{n+h} = E_n(X_{n+h})$.

One-step ahead predictor $P_{n-1} X_n \equiv \hat{X}_n$ (§11.1.2 of week 5 LNs) is given by the **innovation algorithm** that works as follows:

(i) write the one-step predictor in terms of innovations:

$$\hat{X}_{n+1} = \sum_{j=1}^{n} \theta_{nj} (X_{n+1-j} - \hat{X}_{n+1-j}), \ n \geq 1.$$

(ii) innovation algorithm finds coefficients θ_{nj} for covariance $\kappa(t, s)$ of X. It also gives prediction error $v_n = E(X_{n+1} - \hat{X}_{n+1})^2$.

Sometimes, we can calculate predictors directly – see examples with AR models.

Example 14.1. Forecasting AR(1). $X_t = \phi_1 X_{t-1} + Z_t$.

One-step ahead predictor:
$P_n X_{n+1} = E_n(X_{n+1}) = E_n(\phi_1 X_{n+1-1} + Z_{n+1}) = \phi_1 E_n(X_n) = \phi_1 X_n$.

h-step ahead predictor:
$P_n X_{n+h} = E_n(X_{n+h}) = \phi_1 E_n(X_{n+h-1}) + E_n(Z_{n+h}) = \phi_1 P_n X_{n+h-1}$.
Thus, $P_n X_{n+2} = \phi_1 P_n X_{n+1} \equiv \phi_1^2 X_n, \ldots, P_n X_{n+h} = \phi_1^h X_n$.

Conclude: for stationary AR(1) process ($|\phi_1| < 1$) and large lead time h the best forecast is eventually converge to zero = the mean of the process.

Prediction error for AR(1): First, consider the differences
$e_n(1) = X_{n+1} - P_n X_{n+1} = (\phi_1 X_n + Z_{n+1}) - (\phi_1 X_n) = Z_{n+1}$;

$e_n(2) = X_{n+2} - P_n X_{n+2} = (\phi_1 X_{n+1} + Z_{n+2}) - (\phi_1^2 X_n)$
$= \phi_1(X_{n+1} - \phi_1 X_n) + Z_{n+2} = \phi_1(X_{n+1} - P_n X_{n+1}) + Z_{n+2}$
$= \phi_1 e_n(1) + Z_{n+2} = \phi_1 Z_{n+1} + Z_{n+2}$

.....
$e_n(h) = \phi_1^{h-1} Z_{n+1} + \phi_1^{h-2} Z_{n+2} + \ldots + \phi_1 Z_{n+h-1} + Z_{n+h}$.

Then **the prediction error is the variance** (shocks are uncorrelated!)
$Var(e_n(h)) \equiv Var[\phi_1^{h-1} Z_{n+1} + \phi_1^{h-2} Z_{n+2} + \ldots + \phi_1 Z_{n+h-1} + Z_{n+h}]$
$= \sigma_Z^2(\phi_1^{2(h-1)} + \ldots + \phi_1^2 + 1) = \sigma_Z^2(1 - \phi_1^{2h})/(1 - \phi_1^2)$.

Example 14.2. Forecasting AR(p) $X_t = \phi_1 X_{t-1} + \ldots + \phi_p X_{t-p} + Z_t$.

One-step ahead predictor:
.....
[Recall: $E_n(X_{n-k}) \equiv E(X_{n-k}|X_n, X_{n-1}, \ldots, X_{n-k}, \ldots, X_1) = X_{n-k}$ when $0 \le k < n$]
[Recall: $E_n(Z_{n+k}) \equiv E(Z_{n+k}|X_n, X_{n-1}, \ldots, X_1) = E(Z_{n+k}) = 0$ when $k > 0$]
.....
$P_n X_{n+1} = E_n(X_{n+1}) = \phi_1 E_n(X_{n+1-1}) + \ldots + \phi_p E_n(X_{n+1-p}) + E_n(Z_{n+1})$
$= \phi_1 X_n + \ldots + \phi_p X_{n+1-p}$.

Two-steps ahead predictor:
$P_n X_{n+2} = E_n(X_{n+2}) = \phi_1 E_n(X_{n+1}) + \ldots + \phi_p E_n(X_{n+2-p}) + E_n(Z_{n+2})$
$= \phi_1 P_n(X_{n+1}) + \ldots + \phi_p X_{n+2-p}$, a recursive formula.

Example 14.3. Forecasting MA(q) $X_t = Z_t + \theta_1 Z_{t-1} + \ldots + \theta_q Z_{t-q}$ and $n > q$.

– **Innovation Algorithm** applied to MA(q) (see section 11.1.2) gives a one-step ahead predictor
$P_n X_{n+1} = \hat{X}_{n+1} = \theta_{n,1}(X_n - \hat{X}_n) + \ldots + \theta_{n,q}(X_{n+1-q} - \hat{X}_{n+1-q})$
$\equiv \sum_{j=1}^q \theta_{n,j}(X_{n+1-j} - \hat{X}_{n+1-j})$. (*)

– Based on innovation algorithm, we now proceed with

Recursive calculation of the h-step predictor.

Goal: find $P_n X_{n+h}$ = the h-step predictor of X_{n+h} given $X_1, \ldots, X_n$.

Method:

Step1: Use *Tower property of the conditional expectation* to write $P_n X_{n+h} = P_n P_{n+h-1} X_{n+h}$.
.....
Explanation:
– *By definition,* $P_n X_{n+h} = E(X_{n+h}|X_n, \ldots, X_1) \equiv E_n(X_{n+h})$.
– *By Tower property,* $E_n(X_{n+h}) = E_n\{E(X_{n+h}|X_{n+h-1}, \ldots, X_1)\}$.
– *In our notations,* $E(X_{n+h}|X_{n+h-1}, \ldots, X_1) = P_{n+h-1} X_{n+h}$, *a one step ahead predictor of* X_{n+h}.
.....

Step 2: (Recursive!) Use formula (*) with n replaced by $n + h - 1$ to write the one-step predictor
$P_{n+h-1} X_{n+h} = \sum_{j=1}^q \theta_{n+h-1,j}(X_{n+h-j} - \hat{X}_{n+h-j})$.
(coefficients $\theta_{n+h-1,j}$ are given by the innovation algorithm.)

Step 3: Put formulas of Step 1 and Step 2 together, use linearity of conditional expectations:
$P_n X_{n+h} = P_n\left(\sum_{j=1}^q \theta_{n+h-1,j}(X_{n+h-j} - \hat{X}_{n+h-j})\right) = \sum_{j=1}^q \theta_{n+h-1,j} P_n(X_{n+h-j} - \hat{X}_{n+h-j})$. (**)

Step 4. Observe that in (**), for $k = h - j$ we have:
– if $k = h - j > 0$, that is, $j < h$, holds: $P_n(X_{n+k} - \hat{X}_{n+k}) \equiv P_n(X_{n+h-j} - \hat{X}_{n+h-j}) = 0$
[*because* $P_n \hat{X}_{n+k} \stackrel{def}{=} E_n\{E(X_{n+k}|X_{n+k-1}, \ldots X_1)\} = E_n X_{n+k} \stackrel{def}{=} P_n X_{n+k}$, $k \ge 1$.]
– if $k = h - j \le 0$, that is, $j \ge h$, holds: $P_n(X_{n-k} - \hat{X}_{n-k}) = X_{n-k} - \hat{X}_{n-k}$
[*because* $P_n \hat{X}_{n-k} \stackrel{def}{=} E\{E(X_{n-k}|X_{n-k-1}, \ldots, X_1)|X_n, \ldots, X_1\}$ (*usetower property*)
$= E(X_{n-k}|X_{n-k-1}, \ldots, X_1) = \hat{X}_{n-k}$, $k \ge 0$.
and $P_n X_{n-k} \stackrel{def}{=} E_n(X_{n-k}|X_n, \ldots, X_{n-k}, \ldots, X_1) = X_{n-k}$, $k \ge 0$.]

Step 5. Substituting observations from Step 4. into (**) of Step 3. gives us final result:

$$\boxed{P_n X_{n+h} = \sum_{j=h}^q \theta_{n+h-1,j}(X_{n+h-j} - \hat{X}_{n+h-j}).}$$

Example 14.3.3: Assume fitted model MA(1) model $X_t + 4.1 = Z_t - 0.8 Z_{t-1}$. Then,

$$P_{50} X_{50+h} = -4.1 + \sum_{j=h}^{1} \theta_{50+h-1,j}(X_{50+h-j} - \hat{X}_{50+h-j}) = \begin{cases} -4.1 + \theta_{50,1}(X_{50} - \hat{X}_{50}), & \text{for } h = 1, \\ -4.1, & \text{for } h > 1. \end{cases}$$

14.4 Forecasting ARMA: $\phi(B)X_t = \theta(B)Z_t$, $Z_t \sim WN(0, \sigma_Z^2)$

In general, we could simply apply innovation algorithm to X using cov function of X. However, the application of the innovation algorithm is simplified significantly if it is applied to transformed X.

14.4.1 Innovation method applied to ARMA(p,q), one-step ahead predictor.
Recall the model: $\phi(B)X_t = \theta(B)Z_t$.

Step 1: Let $m = \max(p, q)$. **Transform** X: $W_t = \begin{cases} \sigma_Z^{-1} X_t & t = 1, 2, \ldots, m \\ \sigma_Z^{-1}\phi(B)X_t & t > m. \end{cases}$

Motivation: One may calculate acf of W to see that $\kappa(i,j) := EW_iW_j = 0$ for $i > m, |i-j| > q$. (This is similar to acf of MA(q)! Exact reference: §3.3 of [BD]).

Step 2: Write a one-step ahead predictor for W (acf $\kappa(i,j)$) using innovation algorithm:

$$\hat{W}_{n+1} = \begin{cases} \sum_{j=1}^{n} \theta_{n,j}(W_{n+1-j} - \hat{W}_{n+1-j}) & \text{if } 1 \le n \le m \\ \sum_{j=1}^{q} \theta_{n,j}(W_{n+1-j} - \hat{W}_{n+1-j}) & \text{if } n \ge m, \end{cases}$$

the coefficients $\theta_{n,j}$ and the m.s.e. $r_n = E[W_{n+1} - \hat{W}_{n+1}]^2$ are found by innovation algorithm.

Step 3: Return to the original time series X:

– $t \le m$: $X_t = \sigma_Z W_t$, and $\boldsymbol{\hat{X}_t = \sigma_Z \hat{W}_t}$ because of linearity of conditional expectation;

– $t > m$: $W_t = \sigma_Z^{-1}(X_t - \phi_1 X_{t-1} - \ldots - \phi_p X_{t-p})$ or $X_t = \sigma_Z W_t + \phi_1 X_{t-1} + \ldots + \phi_p X_{t-p}$.

Because $X_{t-1}, \ldots, X_{t-p}$ are in the past, and therefore, are known, we get:
$\boldsymbol{\hat{X}_t = \sigma_Z \hat{W}_t + \phi_1 X_{t-1} + \ldots + \phi_p X_{t-p}}$.

Step 4: Conclude: **for all t, $\boldsymbol{X_t - \hat{X}_t = \sigma_Z(W_t - \hat{W}_t)}$.**

Step 5: Substitute into formula from Step 2 to get final result for one-step ahead predictor:

$$\hat{X}_{n+1} = \begin{cases} \sum_{j=1}^{n} \theta_{n,j}(X_{n+1-j} - \hat{X}_{n+1-j}) & \text{if } 1 \le n < m \\ \phi_1 X_n + \ldots + \phi_p X_{n+1-p} + \sum_{j=1}^{q} \theta_{n,j}(X_{n+1-j} - \hat{X}_{n+1-j}) & \text{if } n \ge m \end{cases}$$

$E[X_{n+1} - \hat{X}_{n+1}]^2 = \sigma_Z^2 r_n$.

Note that for pure AR and MA models the predictor remain the same as in 14.1 and 14.3:

Example: AR(p) model. $m = p$; for $n \ge p$

$$\hat{X}_{n+1} = \phi_1 X_n + \ldots + \phi_p X_{n+1-p} \ (q = 0.)$$

Example: MA(q) model. $m = q$; for $n \ge q$

$$\hat{X}_{n+1} = \sum_{j=1}^{q} \theta_{n,j}(X_{n+1-j} - \hat{X}_{n+1-j}) \ (p = 0).$$

14.4.2 h-step ahead predictor for ARMA(p,q). Let $m = \max\{p, q\}$ and $n > m$.

Using formulas from 14.3 for prediction operator P_n ($P_n = E_n$, so that these are properties of conditional expectations), we obtain the following recursion:

$$\begin{aligned} P_n X_{n+h} &= P_n P_{n+h-1} X_{n+h} = P_n\Big\{\sum_{i=1}^{p} \phi_i X_{n+h-i} + \sum_{j=1}^{q} \theta_{n+h-1,j}(X_{n+h-j} - \hat{X}_{n+h-j})\Big\} \\ &= \sum_{i=1}^{p} \phi_i P_n X_{n+h-i} + \sum_{j=h}^{q} \theta_{n+h-1,j}(X_{n+h-j} - \hat{X}_{n+h-j}), \ \textit{recursion!} \end{aligned}$$

14.4.3 Confidence intervals for ARMA forecasts.

If X_t is Gaussian, then **the h-step predictor and prediction errors are Gaussian** and we can write a $(1-\alpha)100\%$ confidence interval for the future value X_{n+h} as follows:

With the probability $(1-\alpha)100\%$, $|X_{n+h} - P_n X_{n+h}| \le \Phi_{1-\alpha/2}\sigma_n(h)$, where for large n,
$\sigma_n^2(h) \approx \sigma_Z^2 \sum_{j=0}^{h-1} \psi_j^2$, $\psi(z) = \theta(z)/\phi(z)$.

14.5 Forecasts of the transformed series.

Let say, V_t is an original data and we performed log transformation: $Y_t = \ln V_t$. Assume that Y_t was modeled as (S)ARIMA process and predicted at lag l:

forecast for Y_{n+l} is $\hat{Y}_n(l) +$ st. error (Notation: $\hat{Y}_n(l) \stackrel{def}{=} P_n V_{n+l}$).

The **naive forecast** of V_{n+l} on basis of $V_n, \ldots, V_1$ is $\hat{V}_n(l) = e^{\hat{Y}_n(l)}$. One can show that naive forecast is **not minimum mean square error forecast.**

Naive forecasts are used because of its simplicity and for the following reason:

Often transformed series Y_t is approximately normal. This means that $V_t = e^{Y_t}$ is approximately lognormal, i.e., asymmetric and with long right tail.

Thus, for this distribution a criterion based on mean absolute value, which estimates median of the distribution of V_{n+l} conditional on $V_n, \ldots, V_1$ makes more sense.

Since the log transformation preserves median and since for normal distribution of Y_{n+l} mean and median coincide, the naive forecast $\hat{V}_n(l) = e^{\hat{Y}_n(l)}$ is the optimal forecast of V_{n+l} in sense that it minimizes the mean absolute forecast error.

Problem with transformation approach: usually we get Y_t Gaussian for each t, but NOT a Gaussian process. This is why non-Gaussian and non-linear models are important.

R has several routines performing power Box-Cox transformations (see 8. of LNs, lecture 7):

$$f_\lambda(V_t) = \begin{cases} \ln V_t, & \text{if } V_t > 0, \ \lambda = 0; \\ \lambda^{-1}(V_t^\lambda - 1), & \text{if } V_t \ge 0 \end{cases}$$

Most often: $\lambda = 0$, $\lambda = 0.5$ which correspond to $Y_t = \ln V_t$ and $Y_t = \sqrt{V_t}$.

In general, for a family of transformations f_λ we try to find a value λ_0 of λ, such that $Y_t(\lambda_0) = f_{\lambda_0}(V_t)$

is a Gaussian process which can be modeled by ARIMA process. There are maximum likelihood techniques to find the value of λ_0.

In general, λ can be negative or positive. [BD] take it between 0&1.5, but R routine boxcox {MASS} has it in (-2,2) interval. Other routines: *boxcox.fit in geoR package*, etc.

Remember: the goal is give the forecasts of the original data!

To use *boxcox* routine in R (discussed in lab 5) for finding "the best" λ follow steps:

1. *Look up boxcox() in the package "MASS".*
2. *Assume "bc" is an object returned by boxcox(...), you can do*
   with(bc, x[which.max(y)])