Kalman Filter

Theory and Applications

Prof. Dr. Hellinton H. Takada
Insper
hellinton.takada@blitz-trading.com

Insper

November 11, 2012

Contents

- State Space Models

Contents

- State Space Models
- Kalman Filter

Contents

- State Space Models
- Kalman Filter
- Pairs Trading

Introduction

The linear multivariate time series model can be represented by the state space form:

$$\begin{aligned} y_t &= c_t + Z_t \alpha_t + G_t \varepsilon_t \\ \alpha_{t+1} &= d_t + T_t \alpha_t + H_t \varepsilon_t \end{aligned}$$

where $\varepsilon_t \sim NID(0, I)$, $\alpha_1 \sim NID(a, P)$ and $t = 1, ..., n$. The first equation is the observation or mearsurement equation. The second equation is the state equation.
The following dimensions are fixed according to the model being represented:

y_t :	$N \times 1$	space variable dimension
α_t :	$m \times 1$	state variable dimension
ε_t :	$r \times 1$	error dimension

Introduction

All the remaining dimensions are given in the following:

$$\boxed{\begin{array}{cc} \alpha_{t+1}, d_t, a: & m \times 1 \\ T_t, P: & m \times m \\ H_t: & m \times r \\ c_t: & N \times 1 \\ Z_t: & N \times m \\ G_t: & N \times r \end{array}}$$

Introduction

It is possible to write the previous model using a compact notation:

$$\begin{pmatrix} \alpha_{t+1} \\ y_t \end{pmatrix} = \delta_t + \Phi_t \alpha_t + u_t$$

where $t = 1, ..., n$, $\alpha_1 \sim NID\,(a, P)$, $u_t \sim NID\,(0, \Omega_t)$, $\delta_t = \begin{pmatrix} d_t \\ c_t \end{pmatrix}$, $\Phi_t = \begin{pmatrix} T_t \\ Z_t \end{pmatrix}$, $u_t = \begin{pmatrix} H_t \\ G_t \end{pmatrix} \varepsilon_t$ and $\Omega_t = \begin{pmatrix} H_t H_t' & H_t G_t' \\ G_t H_t' & G_t G_t' \end{pmatrix}$. It is also necessary to define $\Sigma = \begin{pmatrix} P \\ a' \end{pmatrix}$.

Local Linear Trend Model

The local linear trend model is given by:

$$\begin{aligned} \mu_{t+1} &= \mu_t + \beta_t + \eta_t, & \eta_t &\sim NID\left(0, \sigma^2_\eta\right), \\ \beta_{t+1} &= \beta_t + \zeta_t, & \zeta_t &\sim NID\left(0, \sigma^2_\zeta\right), \\ y_t &= \mu_t + \varepsilon_t, & \varepsilon_t &\sim NID\left(0, \sigma^2_\varepsilon\right). \end{aligned}$$

Local Linear Trend Model

Considering the case when $\sigma_\eta^2 = 0$, $\sigma_\zeta^2 = 0.1$, $\sigma_\varepsilon^2 = 1$, $\mu_1 = 0$ and $\beta_1 = 0$. Then,

$$\Phi = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}, \Omega = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \Sigma = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Introduction

The ideia of Kalman filter (1961) is an algorithm for sequentially updating a linear projection for a system. Among other benefits, this algorithm provides a way to calculate exact finite-sample forecasts and the exact likelihood function for Gaussian ARMA processes, to factor matrix auto-covariance-generating functions or spectral densities, and to estimate vector autoregressions with coefficients that change over time.

Procedure

Define $Y_t = \{y_1, ..., y_t\}$. Given $a_{t+1} = E\left(\alpha_{t+1} \middle| Y_t\right)$ and $P_{t+1} = cov\left(\alpha_{t+1} \middle| Y_t\right)$, the iterative process of Kalman filter is:

$$
\begin{aligned}
v_t &= y_t - c_t - Z_t a_t, \\
F_t &= Z_t P_t Z_t' + G_t G_t', \\
K_t &= \left(T_t P_t Z_t' + H_t G_t'\right) F_t^{-1}, \\
a_{t+1} &= d_t + T_t a_t + K_t v_t, \\
P_{t+1} &= T_t P_t T_t' + H_t H_t' - K_t F_t K_t'.
\end{aligned}
$$

Log-likelihood

The log-likelihood function is given by:

$$l = -\frac{nN}{2}\log\left(2\pi\right) - \frac{1}{2}\sum_{i=1}^{n}\left(\log\left|F_t\right| + v_t' F_t^{-1} v_t\right)$$

where n is the number of observations.

Local Linear Trend Model

```
%% Local Linear Trend Model
ss.mPhi = [1 1; 0 1; 1 0];
ss.mOmega = [0 0 0; 0 0.1 0; 0 0 1];
ss.mSigma = [0 0; 0 0; 0 0];
[phi, omega, delta, sigma, kappa, ss] = ssCheck1(length(y),ss);
[v, K, F_1, yHat, a, P] = ssFilter1(y, phi, omega, delta, sigma);
plot([yHat y]);
plot([a(1:end-1,1) y]);
%% How do you verify the quality of projection?
```

Local Linear Trend Model

```
function  lik = lik_LocalLevelTrend(par,y)
% sigma_eta
% sigma_zeta
% sigma_epsilon
log_sigma_eta = par(1);
log_sigma_zeta = par(2);
log_sigma_epsilon = par(3);
sigma_eta = exp(log_sigma_eta);
sigma_zeta = exp(log_sigma_zeta);
sigma_epsilon = exp(log_sigma_epsilon);
.
.
.
```

Local Linear Trend Model

.
.
.

```
T = [1 1; 0 1];
Z = [1 0];
H = [sigma_eta 0 0; 0 sigma_zeta 0];
G = [0 0 sigma_epsilon];
ss.mPhi = [T; Z];
ss.mOmega = [H*(H') H*(G'); G*(H') G*(G')];
ss.mSigma = [0 0; 0 0; 0 0];
lik = - ssLik1(y,ss);
```

Local Linear Trend Model

```
%% Optimization
% sigma_eta
% sigma_zeta
% sigma_epsilon
init = randn(3,1);
nr = round(length(y)/2);
options = optimset('Display','iter','MaxIter',500,'MaxFunEvals',20000,...
'TolFun',1e-6,'TolX',1e-8,'DiffMaxChange',1e-2,'DiffMinChange',1e-15);
optimum = fminunc(@lik_LocalLevelTrend, init, options, y(1:nr));
sigma_eta = abs(optimum(1));
sigma_zeta = abs(optimum(2));
sigma_epsilon = abs(optimum(3));
```

Local Linear Trend Model

```
%% Local Linear Trend Model using some optimized parameters
H = [sigma_eta 0 0; 0 sigma_zeta 0];
G = [0 0 sigma_epsilon];
ss.mPhi = [1 1; 0 1; 1 0];
ss.mOmega = [H*(H') H*(G'); G*(H') G*(G')];
ss.mSigma = [0 0; 0 0; 0 0];
[phi, omega, delta, sigma, kappa, ss] = ssCheck1(length(y),ss);
[v, K, F_1, yHat, a, P] = ssFilter1(y, phi, omega, delta, sigma);
plot([y yHat]);
plot([a(1:end-1,1) y]);
```

Activity

1. Is it possible to include more parameters in the previous optimization? Describe the idea.
2. Develop a strategy using the Kalman filter for the FX time series performing a backtest. Repeat the procedure for at least 4 different FX time series.

Case study approach

The following model (in level) is proposed:

$$\beta_{t+1} = \beta_t + \eta_t, \quad \eta_t \sim NID\left(0, \sigma_\eta^2\right),$$
$$y_t = x_t\beta_t + \zeta_t, \quad \zeta_t \sim NID\left(0, \sigma_\zeta^2\right).$$

State Space Model

Considering the case when $\sigma_\eta^2 = 0.1$, $\sigma_\zeta^2 = 0.1$ and $\beta_1 = 0$. Then,

$$\Phi_t = \begin{pmatrix} 1 \\ x_t \end{pmatrix}, \Omega = \begin{pmatrix} \sigma_\eta^2 & 0 \\ 0 & \sigma_\zeta^2 \end{pmatrix}, \Sigma = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

State Space Model

```
%% Pairs Trading - Beta
ss.mPhi = [1; 0];
ss.mOmega = [0.1 0; 0 0.1];
ss.mSigma = [0; 0];
ss.mJPhi = [-1; 1];
ss.mXt = x;
[phi, omega, delta, sigma, kappa, ss] = ssCheck1(length(y),ss);
[v, K, F_1, yHat, a, P] = ssFilter1(y, phi, omega, delta, sigma);
plot([y yHat]);
plot(a);
plot(y-x.*a(1:end-1))
```

Activity

1. Develop a strategy using the model in level given (don't forget to present formally the strategy, optimize the parameters and perform a backtest).
2. Develop a new strategy using a model in difference based on CAPM.