最小二乘法辨识

改进算法

邢超

<LS.1>

主要内容

Contents

<LS.2>

1 辅助变量法

辅助变量法

- 辨识精度高于基本最小二乘估计法；
- 计算简单；
- 渐近无偏估计；
- 需构造辅助变量矩阵。

<LS.3>

辅助变量法原理

$$
\begin{aligned}
Y &= \Phi\theta+\xi \\
\Phi^T Y &= \Phi^T\Phi\theta+\Phi^T\xi \\
(\Phi^T\Phi)^{-1}\Phi^T Y &= (\Phi^T\Phi)^{-1}\Phi^T\Phi\theta+(\Phi^T\Phi)^{-1}\Phi^T\xi \\
(\Phi^T\Phi)^{-1}\Phi^T Y &= \theta+(\Phi^T\Phi)^{-1}\Phi^T\xi \\
Z^T Y &= Z^T\Phi\theta+Z^T\xi \\
(Z^T\Phi)^{-1}\Phi^T Y &= (Z^T\Phi)^{-1}\Phi^T\Phi\theta+(Z^T\Phi)^{-1}Z^T\xi \\
(Z^T\Phi)^{-1}\Phi^T Y &= \theta+(Z^T\Phi)^{-1}Z^T\xi
\end{aligned}
$$

其中：

$$
\begin{aligned}
E(Z^T\xi) &= 0 \\
E(Z^T\Phi) &= Q
\end{aligned}
$$

其中 Q 非奇异。

<LS.4>

渐近无偏性

$$\begin{aligned} E[\hat{\theta}_{IV}] &= E[(Z^T\Phi)^{-1}Z^TY] \\ &= E[(Z^T\Phi)^{-1}Z^T(\Phi\theta+\xi)] \\ &= \theta + E[(Z^T\Phi)^{-1}Z^T\xi] \\ \lim_{N\to\infty} E[\hat{\theta}_{IV}] &= \theta + E[(Z^T\Phi)^{-1}]E[Z^T\xi] \\ &= \theta \end{aligned}$$

<LS.5>

辅助变量法的构造方法

- 递推辅助变量参数估计法
- 自适应滤波法
- 纯滞后法
- 塔利原理法

<LS.6>

递推辅助变量参数估计法:Z

$$\begin{aligned} \hat{Y} &= Z\hat{\theta} \\ Z &= \begin{bmatrix} -\hat{y}_n & \cdots & -\hat{y}_1 & u_{n+1} & \cdots & u_1 \\ \vdots & & \vdots & \vdots & & \vdots \\ -\hat{y}_{n+N-1} & \cdots & -\hat{y}_N & u_{n+N} & \cdots & u_N \end{bmatrix} \end{aligned}$$

<LS.7>

递推辅助变量参数估计法: 具体步骤

- 初始化：应用基本最小二乘法估计 $\hat{\theta}$, 取 $Z=\Phi$,
- 递推：
  - 更新 Z
  $$\hat{Y} = Z\hat{\theta}$$
  - 计算 $\hat{\theta}$
  $$\hat{\theta} = (Z^T\Phi)^{-1}Z^TY$$
  - 重复迭代，直至 $\hat{\theta}$ 收敛。

<LS.8>

自适应滤波法

在递推辅助变量参数估计法基础上取:

$$\hat{\theta}_k = (1-\alpha)\hat{\theta}_{k-1} + \alpha\hat{\theta}_{k-d}$$

α: $\in [0.01, 0.1]$
d: $\in [0, 10]$,

<LS.9>

纯滞后法

$$\hat{y}_k = u_{k-m}$$

$d = n$ 时，有：

$$Z = \begin{bmatrix} -u_0 & \cdots & -u_{1-n} & u_{n+1} & \cdots & u_1 \\ -u_1 & \cdots & -u_{2-n} & u_{n+2} & \cdots & u_2 \\ \vdots & & \vdots & \vdots & \vdots & \\ -u_{N-1} & \cdots & -u_{N-n} & u_{n+N} & \cdots & u_2 \end{bmatrix}$$

<LS.10>

塔利（Tally）原理

若噪声 ξ_k 可看作模型：

$$\xi_k = c(z^{-1})n_k$$

的输出，其中 n_k 是 0 均值不相关随机噪声。且：

$$c(z^{-1}) = 1 + c_1 z^{-1} + \cdots + c_m z^{-m}$$

则可取：

$$\begin{aligned} \hat{y}_k &= y_{k-m} \\ Z &= \begin{bmatrix} -y_{n-m} & \cdots & -y_{1-m} & u_{n+1} & \cdots & u_1 \\ -y_{n+1-m} & \cdots & -y_{2-m} & u_{n+2} & \cdots & u_2 \\ \vdots & & \vdots & \vdots & \vdots & \\ -y_{n+N-1-m} & \cdots & -y_{N-m} & u_{n+N} & \cdots & u_2 \end{bmatrix} \end{aligned}$$

<LS.11>

递推辅助变量法

$$\begin{aligned} \hat{\theta}_N &= P_N Z_N^T Y_N \\ P_N &= (Z_N^T \Phi_N)^{-1} \\ \hat{\theta}_{N+1} &= P_{N+1} Z_{N+1}^T Y_{N+1} \\ P_{N+1} &= \left(\begin{bmatrix} Z_N^T & Z_{N+1} \end{bmatrix} \begin{bmatrix} \Phi_N \\ \Psi_{N+1}^T \end{bmatrix} \right)^{-1} \\ &= (P_N^{-1} + Z_{N+1}\Psi_{N+1}^T)^{-1} \\ \Psi_{N+1} &= \begin{bmatrix} -y_{n+N} & \cdots & -y_{N+1} & u_{n+N+1} & \cdots & u_{N+1} \end{bmatrix}^T \\ z_{N+1} &= \begin{bmatrix} -\hat{y}_{n+N} & \cdots & -\hat{y}_{N+1} & u_{n+N+1} & \cdots & u_{N+1} \end{bmatrix}^T \end{aligned}$$

<LS.12>

递推辅助变量法

利用矩阵求逆引理可推导出递推计算公式：

$$\begin{aligned} \hat{\theta}_{N+1} &= \hat{\theta}_N + K_{N+1}(y_{N+1} - \psi_{N+1}^T \hat{\theta}_N) \\ P_{N+1} &= P_N - K_{N+1}\Psi_{N+1}^T P_N \\ K_{N+1} &= P_N z_{N+1}(1 + \Psi_{N+1}^T P_N z_{N+1})^{-1} \end{aligned}$$

- 初始参数参照递推最小二乘法选取
- 对初始值 P_0 的选取比较敏感，最好在前 50~100 个点采用递推最小二乘，然后转换到辅助变量法

<LS.13>

2 广义最小二乘法

广义最小二乘法

- 建立滤波模型，对数据进行白化处理
- 方法较复杂，计算量较大
- 迭代算法收敛性未证明

<LS.14>

广义最小二乘法: 系统模型

$$
\begin{aligned}
a(z^{-1})y_k &= b(z^{-1})u_k + \xi_k \\
f(z^{-1}) &= 1 + f_1 z^{-1} + \cdots + f_m z^{-m} \\
\xi_k &= \frac{1}{f(z^{-1})}\varepsilon_k \\
f(z^{-1})\xi_k &= \varepsilon_k \\
\xi_k &= -f_1\xi_{k-1} - \cdots - f_m\xi_{k-m} + \varepsilon_k \\
a(z^{-1})f(z^{-1})y_k &= b(z^{-1})f(z^{-1})u_k + \varepsilon_k \\
a(z^{-1})\bar{y}_k &= b(z^{-1})\bar{u}_k + \varepsilon_k \\
\bar{y}_k &= f(z^{-1})y_k \\
&= y_k + f_1 y_{k-1} + \cdots + f_m y_{k-m} \\
\bar{u}_k &= f(z^{-1})u_k \\
&= u_k + f_1 u_{k-1} + \cdots + f_m u_{k-m}
\end{aligned}
$$

<LS.15>

广义最小二乘法: 噪声模型参数估计

$$
\begin{aligned}
\xi &= \Omega f + \varepsilon \\
\xi &= \begin{bmatrix} \xi_{n+1} & \xi_{n+2} & \cdots & \xi_{n+N} \end{bmatrix}^T \\
f &= \begin{bmatrix} f_1 & f_2 & \cdots & f_m \end{bmatrix}^T \\
\varepsilon &= \begin{bmatrix} \varepsilon_{n+1} & \varepsilon_{n+2} & \cdots & \varepsilon_{n+N} \end{bmatrix}^T \\
\Omega &= \begin{bmatrix} -\xi_n & \cdots & -\xi_{n+1-m} \\ -\xi_{n+1} & \cdots & -\xi_{n+2-m} \\ \vdots & & \vdots \\ -\xi_{n+N-1} & \cdots & -\xi_{n+N-m} \end{bmatrix} \\
\hat{f} &= (\Omega^T\Omega)^{-1}\Omega^T\xi
\end{aligned}
$$

<LS.16>

广义最小二乘法: 步骤

- 初始化，取

$$\hat{f}(z^{-1}) = 1$$

- 迭代
  - 滤波：

$$
\begin{aligned}
\bar{y}_k &= \hat{f}(z^{-1})y_k \\
\bar{u}_k &= \hat{f}(z^{-1})u_k
\end{aligned}
$$

– 最小二乘估计：

$$\hat{\theta} = (\bar{\Phi}^T\bar{\Phi})^{-1}\bar{\Phi}^T\bar{Y}$$

– 残差：

$$\hat{\xi} = Y - \Phi\hat{\theta}$$

– 用残差 $\hat{\xi}$ 代替 ξ 计算 $\hat{f}$：

$$\hat{f} = (\hat{\Omega}^T\hat{\Omega})^{-1}\hat{\Omega}^T e$$

<LS.17>

递推广义最小二乘法

- 包括参数 $\hat{\theta}$ 的递推及噪声模型参数 $\hat{f}$ 的递推
- 离线与递推计算结果不完全一样
- 主要步聚：
  – 初始化，参照递推最小二乘选取初始值
  – 滤波，计算新的 $\bar{y}_k,\bar{u}_k$
  – 按递推最小二乘算法计算 $\hat{\theta}$ 与 $\hat{f}$

<LS.18>

递推广义最小二乘法

- 初始化：

$$\begin{aligned}
\hat{\theta}_0 &= 0\\
P_0^{(\theta)} &= c_1^2 I\\
\hat{f}_{(0)} &= 0\\
P_0^{(f)} &= c_2^2 I
\end{aligned}$$

- 滤波

$$\begin{aligned}
\bar{y}_{N+1} &= \hat{f}_{(N)}(z^{-1})y_{N+1}\\
&= \hat{f}_{(N)}(z^{-1})y_{(n+N+1)}\\
\bar{u}_{N+1} &= \hat{f}_{(N)}(z^{-1})u_{N+1}\\
&= \hat{f}_{(N)}(z^{-1})u_{(n+N+1)}
\end{aligned}$$

<LS.19>

递推广义最小二乘法

- 计算 $\hat{\theta}$

$$\begin{aligned}
\hat{\theta}_{N+1} &= \hat{\theta}_N + K_{N+1}^{(\theta)}(\bar{y}_{N+1} - \bar{\Psi}_{N+1}^T\hat{\theta}_N)\\
K_{N+1}^{(\theta)} &= P_N^{(\theta)}\bar{\Psi}_{N+1}(1+\bar{\Psi}_{N+1}^T P_N^{(\theta)}\bar{\Psi}_{N+1})^{-1}\\
P_{N+1}^{(\theta)} &= P_N^{(\theta)} - K_{N+1}^{(\theta)}\bar{\Psi}_{N+1}^T P_N^{(\theta)}\\
\bar{\Psi}_{N+1} &= \begin{bmatrix}-\bar{y}_{n+N} & \cdots & -\bar{y}_{N+1} & \bar{u}_{n+N+1} & \cdots & \bar{u}_{N+1}\end{bmatrix}^T
\end{aligned}$$

<LS.20>

递推广义最小二乘法

- 计算残差 $\hat{\xi}_{N+1}$

$$\hat{\xi}_{N+1} = y_{N+1} - \Psi_{N+1}\hat{\theta}_{N+1}$$

- 计算 $\hat{f}$

$$\begin{aligned}
\hat{f}_{N+1} &= \hat{f}_N + K_{N+1}^{(f)}(\hat{\xi}_{N+1} - \hat{\omega}_{N+1}^T \hat{f}_N) \\
K_{N+1}^{(f)} &= P_N^{(f)}\hat{\omega}_{N+1}(1 + \hat{\omega}_{N+1}^T P_N^{(f)}\hat{\omega}_{N+1})^{-1} \\
P_{N+1}^{(f)} &= P_N^{(f)} - K_{N+1}^{(f)}\hat{\omega}_{N+1}^T P_N^{(f)} \\
\hat{\omega}_{N+1} &= \begin{bmatrix} -\hat{\xi}_{n+N} & \cdots & -\hat{\xi}_{n+N+1-m} \end{bmatrix}
\end{aligned}$$

<LS.21>

3 夏氏法

夏氏法

- 交替求解系统模型与噪声模型参数
- 分为夏式修正法和夏式改良法两种
- 递推算法可推广至 MIMO 系统
- 不需对数据进行反复过滤，计算效率较高
- 估计结果较好

<LS.22>

夏氏法: 系统模型

$$\begin{aligned}
a(z^{-1})y(k) &= b(z^{-1})u(k) + \xi_k \\
\xi_k &= \frac{\varepsilon(k)}{f(z^{-1})} \\
a(z^{-1}) &= 1 + a_1 z^{-1} + \cdots + a_n z^{-n} \\
b(z^{-1}) &= b_0 + b_1 z^{-1} + \cdots + b_n z^{-n} \\
f(z^{-1}) &= 1 + f_1 z^{-1} + \cdots + f_m z^{-m} \\
\xi_k &= (1 - f(z^{-1})\xi_k + \varepsilon_k \\
a(z^{-1})y(k) &= b(z^{-1})u(k) + (1 - f(z^{-1}))\xi_k + \varepsilon_k
\end{aligned}$$

<LS.23>

夏氏法：系统模型向量表示

$$\begin{aligned}
y_N &= y_{(n+N)} \\
&= \Psi_N^T \theta + \omega_N^T f + \varepsilon_N \\
f &= \begin{bmatrix} f_1 & \cdots & f_m \end{bmatrix}^T \\
\Psi_N &= \begin{bmatrix} -y_{(n+N-1)} & \cdots & -y_{(N)} & u_{(n+N)} & \cdots & u_{(N)} \end{bmatrix}^T \\
\omega_N &= \begin{bmatrix} -\xi_{(n+N-1)} & \cdots & -\xi_{(n+N-m)} \end{bmatrix}^T
\end{aligned}$$

<LS.24>

夏氏法：参数求解

$$\begin{aligned}
\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} &= \begin{bmatrix} \Psi_1^T & \omega_1^T \\ \vdots & \vdots \\ \Psi_N^T & \omega_N^T \end{bmatrix} \begin{bmatrix} \theta \\ f \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{bmatrix} \\
Y &= \begin{bmatrix} \Phi & \Omega \end{bmatrix} \begin{bmatrix} \theta \\ f \end{bmatrix} + \varepsilon \\
\begin{bmatrix} \hat{\theta} \\ \hat{f} \end{bmatrix} &= \begin{bmatrix} \Phi^T\Phi & \Phi^T\Omega \\ \Omega^T\Phi & \Omega^T\Omega \end{bmatrix}^{-1} \begin{bmatrix} \Phi^T Y \\ \Omega^T Y \end{bmatrix}
\end{aligned}$$

<LS.25>

夏氏法：夏式偏差修正法

由分块矩阵求逆：

$$\begin{aligned}
\begin{bmatrix} \hat{\theta} \\ \hat{f} \end{bmatrix} &= \begin{bmatrix} \Phi^T\Phi & \Phi^T\Omega \\ \Omega^T\Phi & \Omega^T\Omega \end{bmatrix}^{-1} \begin{bmatrix} \Phi^T Y \\ \Omega^T Y \end{bmatrix} \\
&= \begin{bmatrix} P_N\Phi^T Y - P_N\Phi^T\Omega D^{-1}\Omega^T MY \\ D^{-1}\Omega^T MY \end{bmatrix} \\
&= \begin{bmatrix} \hat{\theta}_{LS} - P_N\Phi^T\Omega\hat{f} \\ D^{-1}\Omega^T MY \end{bmatrix} \\
P_N &= (\Phi^T\Phi)^{-1} \\
M &= I - \Phi(\Phi^T\Phi)^{-1}\Phi^T \\
D &= \Omega^T M\Omega
\end{aligned}$$

<LS.26>

夏氏法：夏式偏差修正法迭代步骤

- 初始化：计算基本最小二乘估计

$$\hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^T Y$$

- 迭代
  - 计算残差 $\hat{\xi}$ 构造 $\hat{\Omega}$

$$\hat{\xi} = Y - \Phi\hat{\theta}$$

  - 计算 $\hat{f}$, 对 $\hat{\theta}$ 进行修正

$$\begin{aligned}
\hat{f} &= D^{-1}\hat{\Omega}^T MY \\
\hat{\theta} &= \hat{\theta} - (\Phi^T\Phi)^{-1}\Phi^T\hat{\Omega}\hat{f}
\end{aligned}$$

<LS.27>

夏氏法：夏氏改良法

用 $\hat{\theta}$ 代替 θ：

$$\begin{aligned}
Y &= \begin{bmatrix} \Phi & \Omega \end{bmatrix} \begin{bmatrix} \hat{\theta} \\ f \end{bmatrix} + \varepsilon \\
&= \Phi\hat{\theta} + \Omega f + \varepsilon \\
Y - \Phi\hat{\theta} &= \Omega f + \varepsilon
\end{aligned}$$

可得 f 的最小二乘估计：

$$\begin{aligned}
\hat{f} &= (\hat{\Omega}^T\hat{\Omega})^{-1}\hat{\Omega}^T(Y - \Phi\hat{\theta}) \\
\hat{\theta} &= \hat{\theta} - (\Phi^T\Phi)^{-1}\Phi^T\Omega\hat{f}
\end{aligned}$$

<LS.28>

夏氏法：夏式改良法迭代步骤

- 初始化：计算基本最小二乘估计

$$\hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^T Y$$

- 迭代
  - 计算残差 $\hat{\xi}$ 构造 $\hat{\Omega}$

$$\hat{\xi} = Y - \Phi\hat{\theta}$$

  - 计算 $\hat{f}$, 对 $\hat{\theta}$ 进行修正

$$\begin{aligned} \hat{f} &= (\hat{\Omega}^T\hat{\Omega})^{-1}\hat{\Omega}^T(Y - \Phi\hat{\theta}) \\ \hat{\theta} &= \hat{\theta} - (\Phi^T\Phi)^{-1}\Phi^T\hat{\Omega}\hat{f} \end{aligned}$$

<LS.29>

夏氏法：递推夏氏法

$$\begin{aligned} \tilde{\Phi} &= \begin{bmatrix}\Phi & \hat{\Omega}\end{bmatrix} \\ \tilde{\theta} &= \begin{bmatrix}\hat{\theta} \\ f\end{bmatrix} \\ \tilde{\theta}^T_{N+1} &= \tilde{\theta}_N + K_{N+1}(y_{N+1} - \tilde{\Psi}^T_{N+1}\tilde{\theta}_N) \\ P_{N+1} &= P_N - K_{N+1}\tilde{\Psi}^T_{N+1}P_N \\ K_{N+1} &= P_N\tilde{\Psi}^T_{N+1}(1 + \tilde{\Psi}^T_{N+1}P_N\tilde{\Psi}_{N+1})^{-1} \end{aligned}$$

其中：

$$\begin{aligned} y_N &= \tilde{\Psi}^T_N\tilde{\theta} + \hat{\varepsilon}_{(n+N)} \\ \tilde{\Psi}_N &= \begin{bmatrix}\Psi^T_N & \hat{\omega}^T_N\end{bmatrix}^T \\ \Psi_N &= \begin{bmatrix}-y_{(n+N-1)} & \cdots & -y_{(N)} & u_{(n+N)} & \cdots & u_{(N)}\end{bmatrix}^T \\ \hat{\omega}_N &= \begin{bmatrix}\hat{\xi}_{(n+N-1)} & \cdots & \hat{\xi}_{(n+N-m)}\end{bmatrix}^T \\ \hat{\xi}_k &= y_k - \Psi_k\hat{\theta} \end{aligned}$$

<LS.30>

4 增广矩阵法

增广矩阵法

- 将噪声模型参数扩充到被辨识参数向量中
- 系统参数与噪声参数同时辨识
- 应用广泛，算法收敛性好
- 实际算法中常采用递推方法

<LS.31>

增广矩阵法: 系统模型

$$\begin{aligned} a(z^{-1})y(k) &= b(z^{-1})u(k) + c(z^{-1})\varepsilon(k) \\ a(z^{-1}) &= 1 + a_1z^{-1} + \cdots + a_nz^{-n} \\ b(z^{-1}) &= b_0 + b_1z^{-1} + \cdots + b_nz^{-n} \\ c(z^{-1}) &= 1 + c_1z^{-1} + \cdots + c_nz^{-n} \end{aligned}$$

<LS.32>

增广矩阵法：系统模型向量表示

$$\begin{aligned}
y_N &= y_{(n+N)} \\
&= \Psi_N^T\theta + \varepsilon_{(n+N)} \\
&= \begin{bmatrix}\Psi_{N,(y,u)}^T & \Psi_{N,\xi}^T\end{bmatrix}\begin{bmatrix}\theta_{(y,u)} \\ \theta_\xi\end{bmatrix} + \varepsilon_N \\
\theta &= \begin{bmatrix}\theta_{(y,u)} & \theta_\xi\end{bmatrix}^T \\
\theta_{(y,u)} &= \begin{bmatrix}a_1 & \cdots & a_n & b_0 & \cdots & b_n\end{bmatrix}^T \\
\theta_\xi &= \begin{bmatrix}c_1 & \cdots & c_n\end{bmatrix}^T \\
\Psi_N &= \begin{bmatrix}\Psi_{N,(y,u)} & \Psi_{N,\xi}\end{bmatrix}^T \\
\Psi_{N,(y,u)} &= \begin{bmatrix}-y_{(n+N-1)} & \cdots & -y_{(N)} & u_{(n+N)} & \cdots & u_{(N)}\end{bmatrix}^T \\
\Psi_{N,\xi} &= \begin{bmatrix}\varepsilon_{(n+N-1)} & \cdots & \varepsilon_{(N)}\end{bmatrix}^T
\end{aligned}$$

<LS.33>

增广矩阵法：参数求解

$$\begin{aligned}
\begin{bmatrix}y_1 \\ \vdots \\ y_N\end{bmatrix} &= \begin{bmatrix}\Psi_{1,(y,u)}^T & \Psi_{1,\xi}^T \\ \vdots & \vdots \\ \Psi_{N,(y,u)}^T & \Psi_{N,\xi}^T\end{bmatrix}\begin{bmatrix}\theta_{(y,u)} \\ \theta_\xi\end{bmatrix} + \begin{bmatrix}\varepsilon_1 \\ \vdots \\ \varepsilon_N\end{bmatrix} \\
Y &= \begin{bmatrix}\Phi_{(y,u)} & \Phi_\xi\end{bmatrix}\begin{bmatrix}\theta_{(y,u)} \\ \theta_\xi\end{bmatrix} + \varepsilon \\
\begin{bmatrix}\hat\theta_{(y,u)} \\ \hat\theta_\xi\end{bmatrix} &= \begin{bmatrix}\Phi_{(y,u)}^T\Phi_{(y,u)} & \Phi_{(y,u)}^T\Phi_\xi \\ \Phi_\xi^T\Phi_{(y,u)} & \Phi_\xi^T\Phi_\xi\end{bmatrix}^{-1}\begin{bmatrix}\Phi_{(y,u)}^T Y \\ \Phi_\xi^T Y\end{bmatrix}
\end{aligned}$$

<LS.34>

增广矩阵法：递推方程

以 $\hat\varepsilon$ 代替 ε：

$$\begin{aligned}
y_N &= \hat\Psi_N^T\hat\theta + \hat\varepsilon_{(n+N)} \\
\hat\Psi_N &= \begin{bmatrix}-y_{(n+N-1)} & \cdots & -y_{(N)} & u_{(n+N)} & \cdots & u_{(N)} & \hat\varepsilon_N^T\end{bmatrix}^T \\
\hat\varepsilon_N &= \begin{bmatrix}\hat\varepsilon_{(n+N-1)} & \cdots & \hat\varepsilon_{(N)}\end{bmatrix}^T
\end{aligned}$$

可得递推公式：

$$\begin{aligned}
\hat\theta_{N+1}^T &= \hat\theta_N + K_{N+1}(y_{N+1} - \hat\Psi_{N+1}^T\hat\theta_N) \\
P_{N+1} &= P_N - K_{N+1}\hat\Psi_{N+1}^T P_N \\
K_{N+1} &= P_N\hat\Psi_{N+1}^T(1 + \hat\Psi_{N+1}^T P_N\hat\Psi_{N+1})^{-1}
\end{aligned}$$

<LS.35>