

Identificación de **SI**Stemas

Métodos de Estimación Recursivos

Métodos de Identificación Recursivos

□ Mínimos Cuadrados Recursivos

Para una estructura de modelo de **regresión lineal**

$$y(n) = \varphi^T(n)\theta + v(n)$$

la estima que minimiza un criterio cuadrático en los errores de predicción (*i.e.*, la **estima de mínimos cuadrados**)

$$\varepsilon(n, \theta) = y(n) - \varphi^T(n)\theta$$

$$V_N(\theta) = \frac{1}{N} \sum_{n=1}^N \text{Tr} \{ \varepsilon(n) \varepsilon(n)^T \} = \frac{1}{N} \underbrace{\sum_{n=1}^N \varepsilon(n)^2}_{\text{Caso Escalar}}$$

viene dada por

$$\hat{\theta}_N^{LS} = \arg \min_{\theta} V_N(\theta) = \left[\sum_{n=1}^N \varphi(n) \varphi^T(n) \right]^{-1} \left[\sum_{n=1}^N \varphi(n) y(n) \right]$$

donde el subíndice N indica que la estima se ha realizado usando N datos. Surge el interrogante de si se puede **actualizar** esta estima cuando arriba un nuevo dato, sin necesidad de recalcularla desde el inicio.

Definamos

$$P_N = \left[\sum_{n=1}^N \varphi(n) \varphi^T(n) \right]^{-1}$$

Es claro entonces que

$$P_N^{-1} = P_{N-1}^{-1} + \varphi(N) \varphi^T(N)$$

por lo que resulta

$$\begin{aligned} \hat{\theta}_N &= P_N \left[\sum_{n=1}^{N-1} \varphi(n) y(n) + \varphi(N) y(N) \right] \\ &= P_N \left[P_{N-1}^{-1} \hat{\theta}_{N-1} + \varphi(N) y(N) \right] + \hat{\theta}_{N-1} - \hat{\theta}_{N-1} \\ &= \hat{\theta}_{N-1} + P_N \varphi(N) \left[y(N) - \varphi^T(N) \hat{\theta}_{N-1} \right] \end{aligned}$$

Se tiene entonces el siguiente algoritmo recursivo

$$\hat{\theta}_N = \hat{\theta}_{N-1} + K(N)\varepsilon(N)$$

$$K(N) = P_N\varphi(N)$$

$$\varepsilon(N) = y(N) - \varphi^T(N)\hat{\theta}_{N-1}$$

En estas ecuaciones $\varepsilon(N)$ debe interpretarse como el error de predicción de la salida en el instante N basándose en la estima en el instante anterior ($N-1$). La primera ecuación indica que la estima en el instante N se obtiene a partir de la estima en el instante anterior más un término de corrección que depende del error de predicción. El factor $K(N)$ es una ganancia que indica cuanto debe modificarse la estima anterior cuando un nuevo dato está disponible.

El problema con el algoritmo recursivo anterior es que debe invertirse una matriz (para el cálculo de P_N) en cada iteración. Esto puede evitarse recurriendo al **Lema de Inversión de Matrices**, que establece

$$[A + BCD]^{-1} = A^{-1} - A^{-1}B[DA^{-1}B + C^{-1}]^{-1}DA^{-1}$$

para matrices A, B, C, D de dimensiones apropiadas.

Considerando que

$$P_N^{-1} = P_{N-1}^{-1} + \varphi(N)\varphi^T(N)$$

resulta

$$P_N = [P_{N-1}^{-1} + \varphi(N)\varphi^T(N)]^{-1}$$

y la aplicación del Lema de Inversión de Matrices resulta en

$$P_N = P_{N-1} - \frac{P_{N-1}\varphi(N)\varphi^T(N)P_{N-1}}{1 + \varphi^T(N)P_{N-1}\varphi(N)}$$

que es una ecuación recursiva para el cálculo de P_N , en donde sólo hay que invertir un escalar (cociente) en lugar de una matriz

El algoritmo de **Mínimos Cuadrados Recursivos (RLS)** resulta entonces

$$\hat{\theta}_N = \hat{\theta}_{N-1} + K(N)\varepsilon(N)$$

$$K(N) = P_N\varphi(N)$$

$$\varepsilon(N) = y(N) - \varphi^T(N)\hat{\theta}_{N-1}$$

$$P_N = P_{N-1} - \frac{P_{N-1}\varphi(N)\varphi^T(N)P_{N-1}}{1 + \varphi^T(N)P_{N-1}\varphi(N)}$$

Inicialización: El algoritmo necesita obviamente inicializarse. Sin ningún conocimiento previo del vector de parámetros es común tomar

$$\hat{\theta}_0 = 0$$

$$P_0 = \rho I$$

donde ρ es número grande. El algoritmo de RLS puede interpretarse como un filtro de Kalman lo que sugiere tomar como valores iniciales para $\hat{\theta}_0$ una estima previa del vector de parámetros, y para P_0 la covarianza de esa estima previa (es decir una medida del grado de confianza en esa estima). Otra cosa que suele hacerse es hacer una estima **off-line** y usar los resultados de la estima y su covarianza como condiciones iniciales del algoritmo.

□ Mínimos Cuadrados Recursivos Pesados

Se suele incluir un factor de peso en el criterio cuadrático de manera de darle distinto peso a los errores de predicción. Por ejemplo si se incluye un **factor de olvido (forgetting factor)** resulta

$$V_N(\theta) = \frac{1}{N} \sum_{n=1}^N \lambda^{N-n} \text{Tr} \left\{ \varepsilon(n) \varepsilon(n)^T \right\} = \underbrace{\frac{1}{N} \sum_{n=1}^N \lambda^{N-n} \varepsilon(n)^2}_{\text{Caso Escalar}}$$

donde λ es un escalar entre 0 y 1. De esta manera a las mediciones más recientes se les asigna un peso mayor que a las mediciones más viejas.

La estima de **Mínimos Cuadrados Recursivos Pesados** resulta en este caso

$$\hat{\theta}_N = \hat{\theta}_{N-1} + K(N)\varepsilon(N)$$

$$K(N) = P_N \varphi(N)$$

$$\varepsilon(N) = y(N) - \varphi^T(N)\hat{\theta}_{N-1}$$

$$P_N = \frac{1}{\lambda} \left(P_{N-1} - \frac{P_{N-1} \varphi(N) \varphi^T(N) P_{N-1}}{\lambda + \varphi^T(N) P_{N-1} \varphi(N)} \right)$$

□ Least Mean Squares (LMS)

$$\hat{\theta}_N = \hat{\theta}_{N-1} + K(N)\varepsilon(N)$$

$$K(N) = \gamma \varphi(N)$$

$$\varepsilon(N) = y(N) - \varphi^T(N)\hat{\theta}_{N-1}$$

Es un algoritmo del tipo de **gradiente descendiente**. La constante $0 < \gamma < 1$ gobierna la velocidad con la que se actualiza la estima cuando llega un nuevo dato. Este parámetro puede ser también una secuencia de valores $\gamma(N)$.

□ Filtro de Kalman

El **Filtro de Kalman** es un algoritmo recursivo de procesamiento de datos que es óptimo con respecto a casi cualquier criterio de optimalidad. Permite la estimación de estados pasados (**smoothing**), presentes (**filtering**) y futuros (**prediction**), a la vez que provee una solución recursiva eficiente de la estima de mínimos cuadrados.

El Filtro de Kalman da solución al problema de estimar el estado x , del sistema en tiempo discreto descrito por

$$x_{k+1} = A_k x_k + B u_k + w_k \quad (1)$$

a partir de mediciones ruidosas z , de la forma

$$z_k = H_k x_k + v_k \quad (2)$$

Se asume que w_k y v_k son procesos aleatorios independientes, blancos y con distribución de probabilidad Gaussiana, es decir

$$\begin{aligned} w_k &\in N(0, Q) \\ v_k &\in N(0, R) \end{aligned} \quad (3)$$

Orígenes computacionales del filtro

Designamos con \hat{x}_k^- a la estima **a priori** del vector de estados en el paso k basado en el conocimiento del proceso anterior al paso k , y con \hat{x}_k a la estima **a posteriori** del estado en el paso k dada la medición z_k . Podemos entonces definir los errores de las estimas **a priori** y **a posteriori** como

$$e_k^- = x_k - \hat{x}_k^- \quad (4)$$

$$e_k = x_k - \hat{x}_k \quad (5)$$

y las correspondientes matrices de covarianza de estos errores

$$P_k^- = E \left[e_k^- e_k^{-T} \right] \quad (6)$$

$$P_k = E \left[e_k e_k^T \right] \quad (7)$$

Para obtener las ecuaciones del filtro, partimos del objetivo de encontrar una ecuación que compute una estima *a posteriori* del estado \hat{x}_k , como una combinación lineal de una estima *a priori* y una diferencia “pesada” entre la medición presente z_k y una predicción de la medición $H_k \hat{x}_k^-$, es decir

$$\hat{x}_k = \hat{x}_k^- + K \left(z_k - H_k \hat{x}_k^- \right) \quad (8)$$

A la diferencia $\left(z_k - H_k \hat{x}_k^- \right)$ se la denomina **innovación** de la medición, o **residuo**. Refleja la discrepancia entre la salida predicha y la medición real.

La matriz de peso K debe ser elegida para minimizar la covarianza del error *a posteriori* P_k , y se denomina **ganancia óptima de Kalman**. Reemplazando (8) en la expresión del error e_k en (5), resulta:

$$\begin{aligned}
e_k &= \mathbf{x}_k - \widehat{\mathbf{x}}_k \\
&= \mathbf{x}_k - \widehat{\mathbf{x}}_k^- - K \left(z_k - H_k \widehat{\mathbf{x}}_k^- \right) \\
&= \mathbf{x}_k - \widehat{\mathbf{x}}_k^- - K \left(H_k \mathbf{x}_k + \mathbf{v}_k - H_k \widehat{\mathbf{x}}_k^- \right) \\
&= \left(I - KH_k \right) \mathbf{x}_k - \left(I - KH_k \right) \widehat{\mathbf{x}}_k^- - K \mathbf{v}_k \\
&= \left(I - KH_k \right) \left(\mathbf{x}_k - \widehat{\mathbf{x}}_k^- \right) - K \mathbf{v}_k \tag{9}
\end{aligned}$$

La covarianza del error *a posteriori* resulta entonces:

$$P_k = \text{cov} \left[e_k \right] = \text{cov} \left[\left(I - KH_k \right) \left(\mathbf{x}_k - \widehat{\mathbf{x}}_k^- \right) - K \mathbf{v}_k \right] \tag{10}$$

Como el ruido de medición \mathbf{v}_k no está correlacionado con los otros términos, resulta:

$$\begin{aligned} P_k &= \text{cov} \left[(I - KH_k) (x_k - \hat{x}_k^-) \right] + \text{cov} [Kv_k] \\ &= (I - KH_k) \text{cov} (x_k - \hat{x}_k^-) (I - KH_k)^T + K \text{cov} [v_k] K^T \end{aligned}$$

Es decir:

$$P_k = (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T \quad (11)$$

donde se ha agregado el subíndice k a la ganancia de Kalman, ya que se derivará una fórmula recursiva para la misma.

La ganancia óptima de Kalman es tal que minimiza la covarianza del error *a posteriori* P_k . Esto es equivalente a minimizar la traza de P_k , lo cual se logra igualando a cero la derivada de la traza, es decir:

$$\frac{\partial \text{Tr}[P_k]}{\partial K_k} = 0 \quad (12)$$

$$\frac{\partial}{\partial K_k} \text{Tr} \left[(I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T \right] = 0$$

$$\frac{\partial}{\partial K_k} \text{Tr} \left[P_k^- - K_k H_k P_k^- - P_k^- H_k^T K_k^T + K_k (H_k P_k^- H_k^T + R_k) K_k^T \right] = 0$$

$$-2P_k^- H_k^T + 2K_k (H_k P_k^- H_k^T + R_k) = 0 \quad (13)$$

En el paso a la última línea se han usado las siguientes identidades referidas a las derivadas de la traza de una matriz:

$$\frac{\partial \text{Tr}[X]}{\partial X} = I$$

$$\frac{\partial \text{Tr}[XA]}{\partial X} = \frac{\partial \text{Tr}[AX]}{\partial X} = A^T$$

$$\frac{\partial \text{Tr}[X^T A]}{\partial X} = \frac{\partial \text{Tr}[AX^T]}{\partial X} = A$$

$$\frac{\partial \text{Tr}[X^T AX]}{\partial X} = (A + A^T) X$$

De (13) resulta

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \quad (14)$$
$$= \frac{P_k^- H_k^T}{H_k P_k^- H_k^T + R_k} \quad (\text{caso escalar})$$

Puede verse que cuando la covarianza del ruido de medición R_k tiende a cero, la ganancia K_k pesa a los residuos más fuertemente. Específicamente

$$\lim_{R_k \rightarrow 0} K_k = H_k^{-1}$$

Por otra parte, cuando la covarianza P_k^- del error de la estima *a priori* tiende a cero, la ganancia K_k pesa a los residuos más levemente. Específicamente

$$\lim_{P_k^- \rightarrow 0} K_k = 0$$

En otras palabras, a medida que la covarianza del error de medición R_k se aproxima a cero, se confía más en la medición real z_k , mientras que se confía menos en la predicción $H_K \hat{x}_k^-$ de la medición. Por otra parte, a medida que la covarianza P_k^- del error de la estima *a priori* se aproxima a cero, se confía cada vez menos en la medición real z_k , y cada vez más en la predicción $H_K \hat{x}_k^-$ de la medición.

Simplificación de la fórmula de la covarianza *a posteriori*

Cuando la ganancia de Kalman iguala al valor óptimo en (14), la fórmula de la covarianza *a posteriori* (11) puede simplificarse considerablemente, lo que es ventajoso para su implementación recursiva. Teniendo en cuenta (14), la ecuación (11) puede escribirse como

$$\begin{aligned}
P_k &= P_k^- - P_k^- H_k^T K_k^T - K_k H_k P_k^- + K_k H_k P_k^- H_k^T K_k^T + K_k R_k K_k^T \\
&= P_k^- - P_k^- H_k^T K_k^T - K_k H_k P_k^- + K_k \left(H_k P_k^- H_k^T + R_k \right) K_k^T \\
&= P_k^- - P_k^- H_k^T K_k^T - K_k H_k P_k^- \\
&\quad + P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k \right)^{-1} \left(H_k P_k^- H_k^T + R_k \right) K_k^T \\
&= \left(I - K_k H_k \right) P_k^- - P_k^- H_k^T K_k^T + P_k^- H_k^T K_k^T \\
&= \left(I - K_k H_k \right) P_k^-
\end{aligned}$$

Es decir

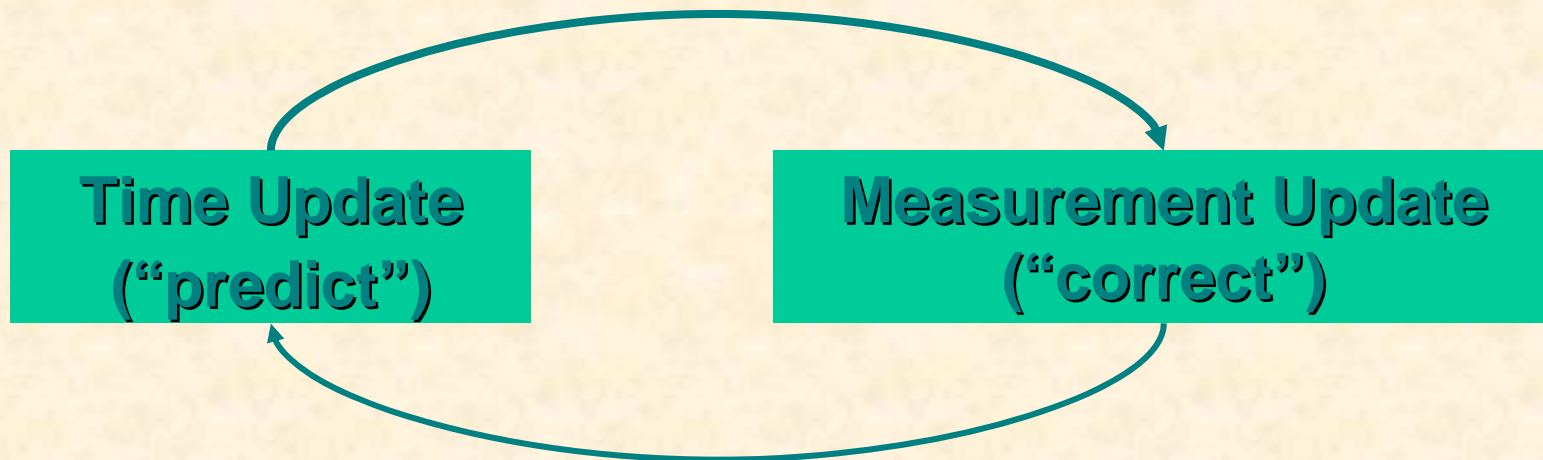
$$P_k = \left(I - K_k H_k \right) P_k^- \tag{15}$$

Algoritmo del filtro de Kalman

Las ecuaciones del filtro de Kalman pueden dividirse en dos grupos:

- **Time Update (Actualización Temporal):** proyectan hacia adelante (en el tiempo) la estima del estado presente y de la covarianza del error para obtener una estima *a priori* para la próxima iteración.

- **Measurement Update (Actualización de las mediciones):** proveen una retroalimentación (feedback), incorporando una nueva medición en la estima *a priori* para obtener una estima *a posteriori* mejorada.



Las ecuaciones específicas de cada grupo son las siguientes:

Estimas iniciales
de \hat{x}_k^- y P_k^-

Time update (“predict”)

(1) Proyecta estado adelante

$$\hat{x}_{k+1}^- = A_k \hat{x}_k + B u_k$$

(2) Proyecta covarianza error adelante

$$P_{k+1}^- = A_k P_k A_k^T + Q_k$$

Measurement update (“correct”)

(1) Computa ganancia de Kalman

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$$

(2) Actualiza Estima con medición z_k

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-)$$

(3) Actualiza covarianza de error

$$P_k = (I - K_k H_k) P_k^-$$

En las ecuaciones de **predicción** se propone actualizar la estima *a priori* mediante

$$\hat{x}_{k+1}^- = A_k \hat{x}_k + Bu_k \quad (16)$$

Por otra parte se verifica

$$x_{k+1} = A_k x_k + Bu_k + w_k$$

Por lo que el error *a priori* resulta

$$e_{k+1}^- = x_{k+1} - \hat{x}_{k+1}^- = A_k (x_k - \hat{x}_k) + w_k = A_k e_k + w_k$$

La covarianza de este error resulta entonces

$$P_{k+1}^- = A_k P_k A_k^T + Q_k \quad (17)$$

□ Estimación de parámetros

Consideremos una estructura de regresor lineal

$$y(n) = \varphi^T(n) \theta + e(n) \quad (18)$$

donde $e(n)$ es ruido blanco Gaussiano con media cero y varianza R . El modelo puede escribirse como ecuaciones de estado

$$\begin{aligned} x(n+1) &= x(n) \\ y(n) &= \varphi^T(n)x(n) + e(n) \end{aligned} \quad (19)$$

donde el vector de estados $x(n)$ es el vector de parámetros θ . La estima óptima de estados viene dada por el filtro de Kalman aplicado a (19), y coincide con la estima de mínimos cuadrados recursivos. Una forma de modificar el algoritmo de manera que pueda seguir parámetros que varíen en el tiempo es cambiar la ecuación de estado en (19) por

$$x(n+1) = x(n) + w(n) \quad E[w(n)w^T(s)] = Q\delta_{n,s}$$

Es decir que estamos modelando al vector de parámetros como una **random walk**. La matriz de covarianza Q puede usarse para describir cuan rápido se espera que varíen las componentes de θ .

En este caso resulta:

$$A = I \quad , \quad B = 0 \quad , \quad H(n) = \varphi^T(n)$$

y las ecuaciones del **Filtro de Kalman** para este caso resultan:

$$\hat{\theta}(n) = \hat{\theta}(n-1) + K(n)\varepsilon(n)$$

$$\varepsilon(n) = y(n) - \varphi^T(n)\hat{\theta}(n-1)$$

$$K(n) = P(n)\varphi(n) = P(n-1)\varphi(n) / \left[R + \varphi^T(n)P(n-1)\varphi(n) \right]$$

$$P(n) = P(n-1) - P(n-1)\varphi(n)\varphi^T(n)P(n-1) / \left[1 + \varphi^T(n)P(n-1)\varphi(n) \right] + Q$$