EESC 9945

Geodesy with the Global Positioning System

Classes 10: *Stochastic Filters I*

Sequential Least Squares

- A type of constrained least-squares solution in which the entire observation set is broken down into *independent* subsets ("batches") that are analyzed sequentially

- At each stage, results from the previous analysis are used as the a priori solution in the subsequent analysis

- Sequential least-squares can be used to process data in real time, and can be extended to the case where the parameter values chance in time

Sequential Least Squares

1. Starting parameter estimates $\widehat{x}_\circ$, $\Lambda_\circ$

2. For each data set y_k with corresponding error covariance G_k, $k = 1, \ldots, n$:

   (a) We use $\widehat{x}_{k-1}$, Λ_{k-1} as prior constraints (dropping hats)

   (b) Design matrix for batch A_k

   (c) Prefit residuals $\Delta y_k = y_k - A_k \widehat{x}_k$ (assume linear)

   (d) Adjustments $\Delta \widehat{x}_k = \widehat{x}_k - \widehat{x}_{k-1}$

Sequential Least Squares

3. Using constrained least-squares that we've used perviously, solutions are:

$$\widehat{x}_k = \widehat{x}_{k-1} + \left(A_k^T G_k^{-1} A_k + \Lambda_{k-1}^{-1}\right)^{-1} A_k G_k^{-1} \left(y_k - A_k \widehat{x}_{k-1}\right)$$

and

$$\Lambda_k = \left(A_k^T G_k^{-1} A_k + \Lambda_{k-1}^{-1}\right)^{-1}$$

Note: k does not have to index time, but it commonly does

Practical Considerations

- If new data are acquired, we don't have to reanalyze the entire batch, only add new batch

- A new batch can have as few as $N_k = 1$ observation

- With M parameters, for each batch we have to perform two inversions of $M \times M$ symmetric matrices, or store Λ_{k-1} and perform one inversion

Sequential Least Squares: Alternative Formulation

- Standard formulation:

$$\widehat{x}_k = \widehat{x}_{k-1} + \left(A_k^T G_k^{-1} A_k + \Lambda_{k-1}^{-1}\right)^{-1} A_k G_k^{-1} \left(y_k - A_k \widehat{x}_{k-1}\right)$$

$$\Lambda_k = \left(A_k^T G_k^{-1} A_k + \Lambda_{k-1}^{-1}\right)^{-1}$$

- Matrix identities (“inside out”):

$$\left(A^T G^{-1} A + \Lambda^{-1}\right)^{-1} A^T G^{-1} = \Lambda A^T \left(A \Lambda A^T + G\right)^{-1}$$

$$\left(A^T G^{-1} A + \Lambda^{-1}\right)^{-1} = \Lambda - \Lambda A^T \left(A \Lambda A^T + G\right)^{-1} A \Lambda$$

Sequential Least Squares: Alternative Formulation

- Therefore, we can write the sequential least-squares solution as

$$\widehat{x}_k = \widehat{x}_{k-1} + \Lambda_{k-1} A_k^T \left(A_k \Lambda_{k-1} A_k^T + G_k \right)^{-1} \left(y_k - A_k \widehat{x}_{k-1} \right)$$

$$\Lambda_k = \Lambda_{k-1} - \Lambda_{k-1} A_k^T \left(A_k \Lambda_{k-1} A_k^T + G_k \right)^{-1} A_k \Lambda_{k-1}$$

- The standard approach has a matrix inversion of order M (# params): $A^T G^{-1} A + \Lambda^{-1}$

- The alternative approach has a matrix inversion of order N (# obs this batch): $A \Lambda A^T + G$

Gain Matrix

- Define the gain matrix K as

$$K_k = \Lambda_{k-1} A_k^T \left(A_k \Lambda_{k-1} A_k^T + G_k \right)^{-1}$$

- Then the sequential least-squares solution can be written as

$$\widehat{x}_k = \widehat{x}_{k-1} + K_k \left(y_k - A_k \widehat{x}_{k-1} \right)$$

$$\Lambda_k = \Lambda_{k-1} - K_k A_k \Lambda_{k-1} = \left(I - K_k A_k \right) \Lambda_{k-1}$$

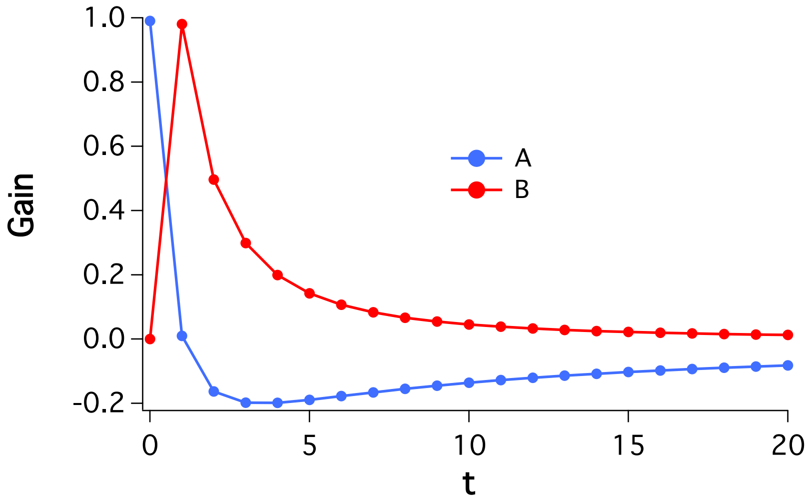
Gain Matrix

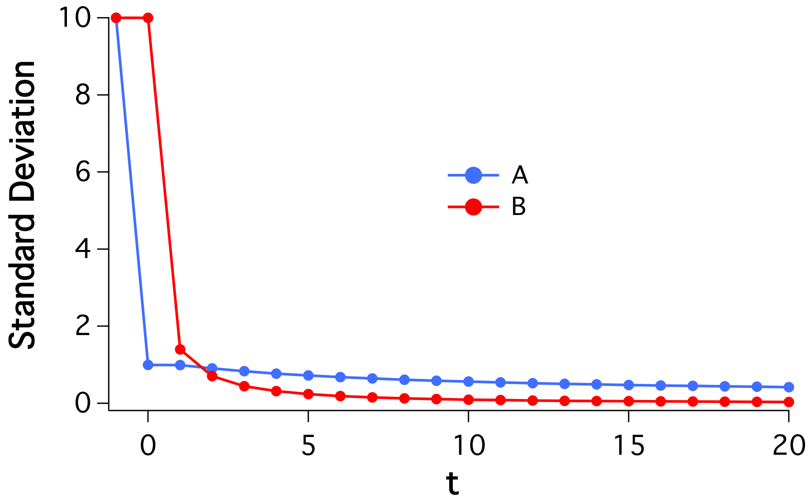
$$K_k = \Lambda_{k-1} A_k^T \left(A_k \Lambda_{k-1} A_k^T + G_k \right)^{-1} \quad \hat{x}_k = \hat{x}_{k-1} + K_k \Delta y_k \quad \Lambda_k = (I - K_k A_k) \Lambda_{k-1}$$

- What is meaning of K?
- Case 1: $K_k = 0$
  - $\hat{x}_k = \hat{x}_{k-1}$ and $\Lambda_{k-1} = \Lambda_k$
  - No information in *new* observations y_k
- Case 2: $K_k A_k = I$
  - Implies A_k^{-1} exists and $G = 0$
  - Then $\hat{x}_k = A_k^{-1} y_k$ and $\Lambda_k = 0$
  - No information in *old* observations
- Note that K_k does not depend on the observations

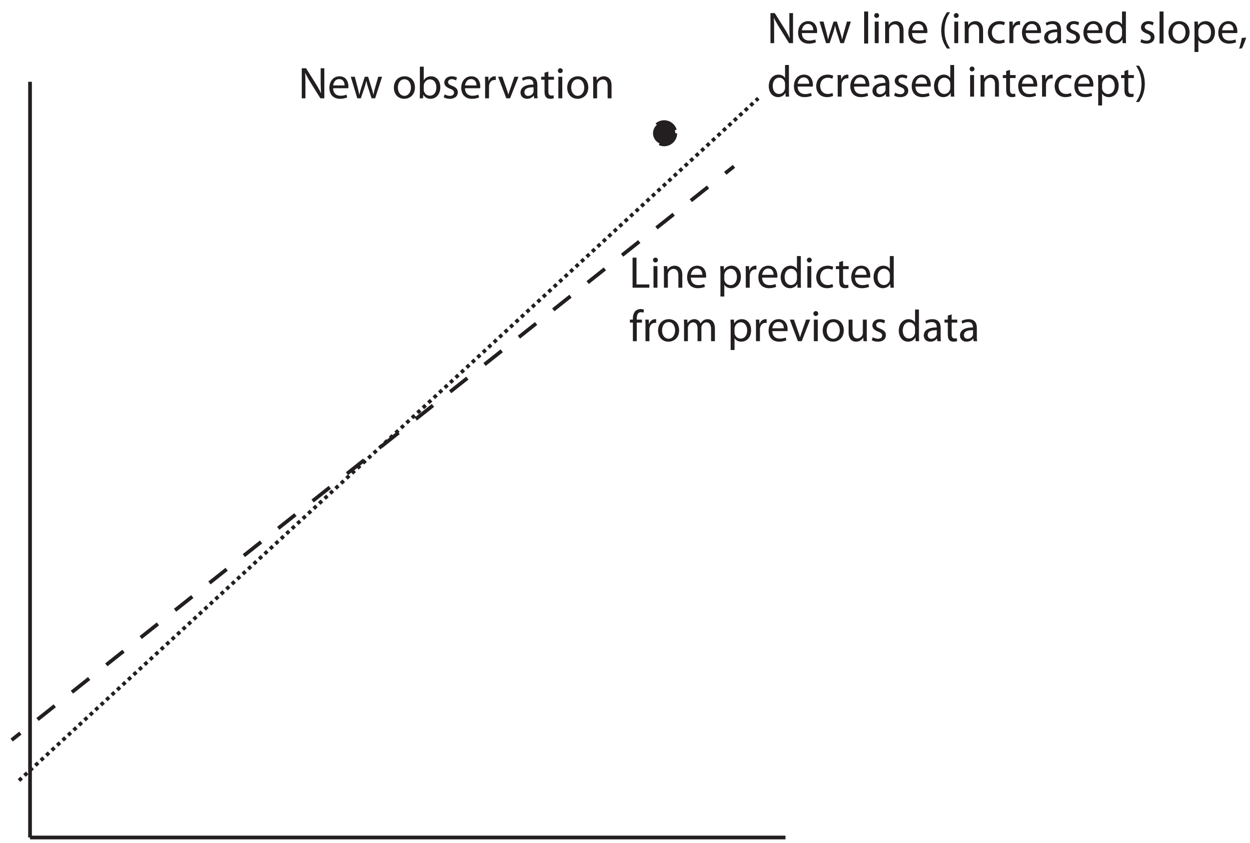
Example

Model: $y_k = a + bk$ with $\sigma = 1$, $\sigma_{a,\circ} = \sigma_{b,\circ} = 100$

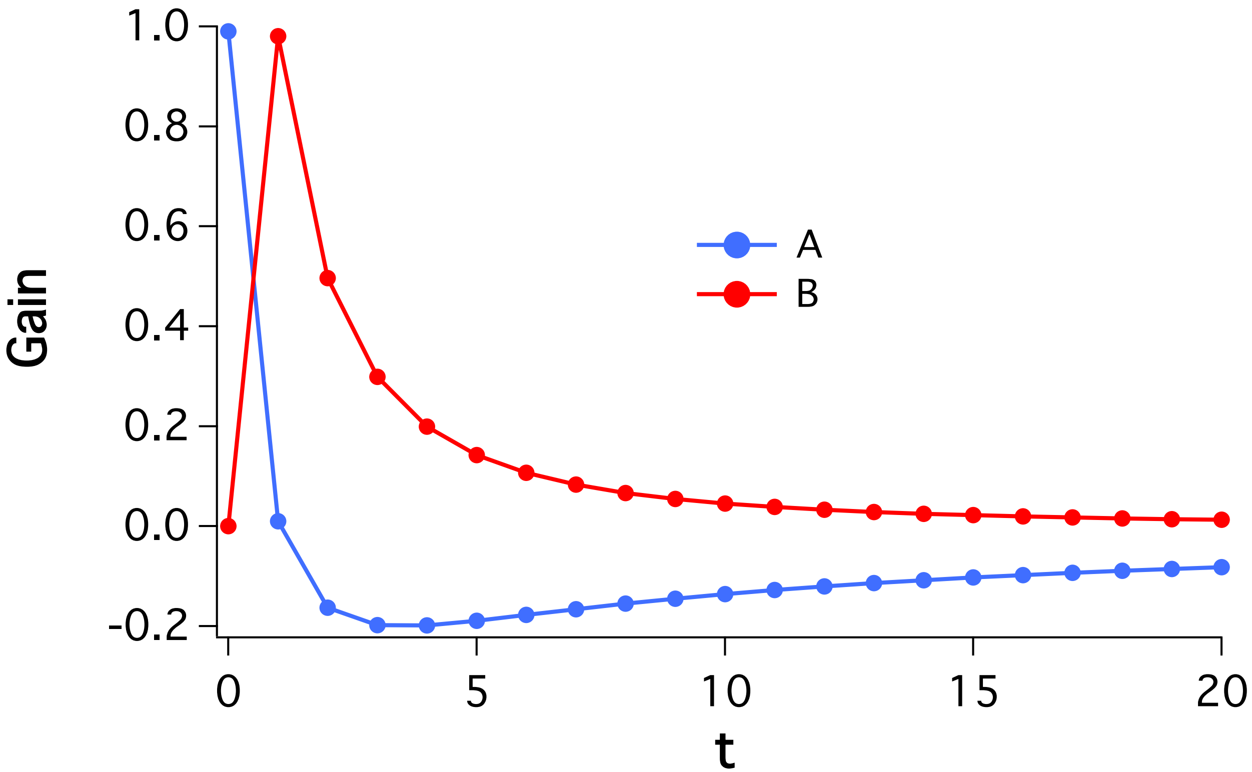




Why positive gain for intercept and negative for slope?



Example



Linear Dynamic Systems

- **State vector**: Set of quantities sufficient to completely describe "unforced" motion of dynamic system

- Transition matrix S relates the states at t_1 and t_2

$$x(t_2) = S(t_1, t_2)x(t_1)$$

- Properties of S:
  - $S(t, t) = I$
  - $S(t_1, t_3) = S(t_1, t_2)S(t_2, t_1)$

– $S(t_1, t_2)S(t_2, t_1) = I$

Example: 1-D velocity

If the state vector is

$$x(t) = \begin{bmatrix} z(t) \\ \dot{z}(t) \end{bmatrix}$$

Then we might have

$$x(t_2) = \begin{pmatrix} 1 & t_2 - t_1 \\ 0 & 1 \end{pmatrix} x(t_1)$$

Note that in this model $\dot{z}(t)$ is constant

State Estimates

- Suppose we have an estimate $\widehat{x}(t_1)$ of the state at time t_1 with covariance $\Lambda(t_1)$
- The estimate is a random variable, whether or not the state is
- We relate $\widehat{x}(t_1)$ to the expectation (mean, $\langle\cdot\rangle$) and Λ to the mean-square error:

$$\widehat{x}(t_1) = \langle x(t_1) \rangle$$

$$\Lambda = \left\langle [\widehat{x}(t_1) - x(t_1)] \, [\widehat{x}(t_1) - x(t_1)]^T \right\rangle$$

Covariance Propagation

- Given the estimate $\widehat{x}(t_1)$ and covariance $\Lambda(t_1)$, what are the estimate and covariance at $t_2 \geq t_1$?

$$\begin{aligned}
\widehat{x}(t_2) &= \langle x(t_2) \rangle = \langle S(t_1, t_2) x(t_1) \rangle \\
&= S(t_1, t_2) \, \langle x(t_1) \rangle \\
&= S(t_1, t_2) \, \widehat{x}(t_1)
\end{aligned}$$

$$\begin{aligned}
\Lambda(t_2) &= \left\langle [\widehat{x}(t_2) - x(t_2)] \, [\widehat{x}(t_2) - x(t_2)]^T \right\rangle \\
&= \left\langle S(t_1, t_2) \, [\widehat{x}(t_1) - x(t_1)] \, [\widehat{x}(t_1) - x(t_1)]^T \, S(t_1, t_2)^T \right\rangle \\
&= S(t_1, t_2) \left\langle [\widehat{x}(t_1) - x(t_1)] \, [\widehat{x}(t_1) - x(t_1)]^T \right\rangle S(t_1, t_2)^T \\
&= S(t_1, t_2) \Lambda(t_1) S(t_1, t_2)^T
\end{aligned}$$

Covariance Propagation: Example

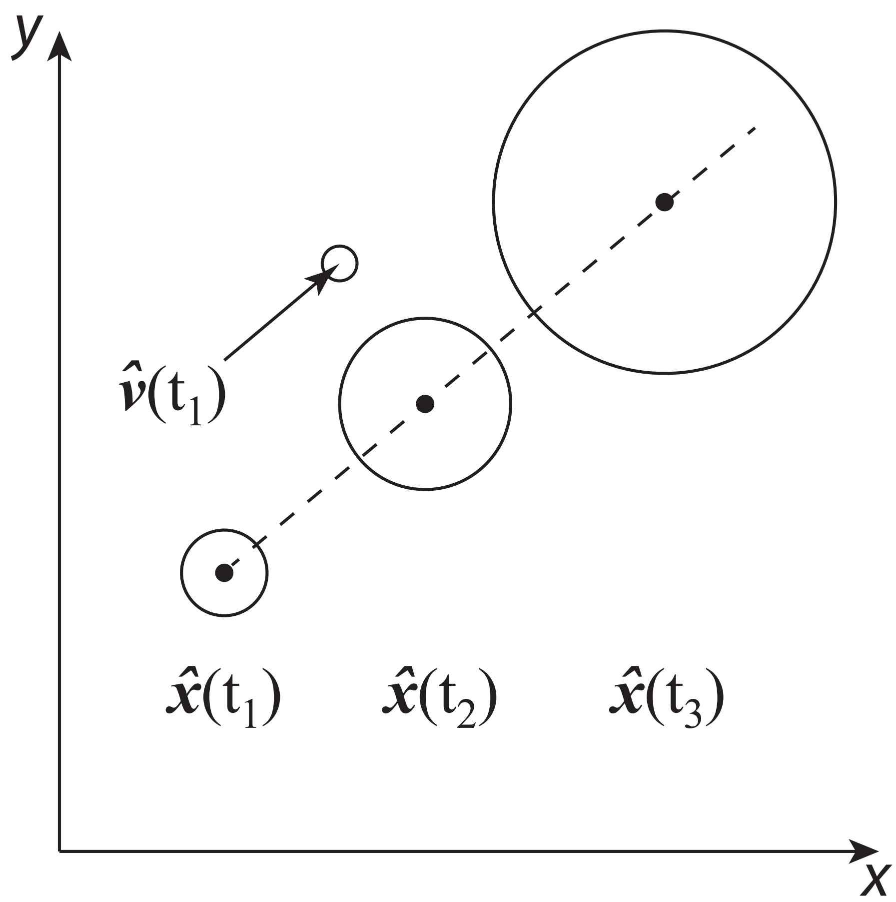
- Constant rate $S(t_1, t_2) = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix}$, $\Delta t = t_2 - t_1$

- Assume $\Lambda(t_1) = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_v^2 \end{pmatrix}$

- Then $\Lambda(t_2) = \Lambda(t_1 + \Delta t) = \begin{pmatrix} \sigma_x^2 + \sigma_v^2 (\Delta t)^2 & \sigma_v^2 \Delta t \\ \sigma_v^2 \Delta t & \sigma_v^2 \end{pmatrix}$

- Note correlations introduced between x and v state parameters

State vector with 2-D position and velocity



Estimation of state vector parameters

- In many problems, parameters of the state vector are unknown and targets of an estimation procedure

- The entire state vector must be included as parameters in the least-squares solution

- For example, in constant velocity example, if we wish to estimate position we also need to include velocity in our least-squares parameter vector

- How does having a dynamic system change the sequential least squares solution?

Sequential least squares (No dynamics)

1. Starting parameter estimates $\widehat{x}_\circ$, $\Lambda_\circ$

2. For each data set y_k with corresponding error co-variance G_k, $k = 1, \ldots, n$:

   (a) We use $\widehat{x}_{k-1}$, Λ_{k-1} as prior constraints

   (b) Design matrix for batch A_k

   (c) Prefit residuals $\Delta y_k = y_k - A_k \widehat{x}_k$ (assume linear)

   (d) Adjustments $\Delta \widehat{x}_k = \widehat{x}_k - \widehat{x}_{k-1}$

Sequential least squares (No dynamics)

1. Starting parameter estimates $\widehat{x}_\circ$, $\Lambda_\circ$

2. For each data set y_k with corresponding error co-variance G_k, $k = 1, \ldots, n$:

   (a) We use $\widehat{x}_{k-1}$, Λ_{k-1} as prior constraints

   (b) Design matrix for batch A_k

   (c) Prefit residuals $\Delta y_k = y_k - A_k \widehat{x}_k$ (assume linear)

   (d) Adjustments $\Delta \widehat{x}_k = \widehat{x}_k - \widehat{x}_{k-1}$

Prediction

1. Epochs $t_0, t_1, \ldots, t_n$ (not necessarily equally spaced)

2. Transition matrix $S(t_{k-1}, t_k) \rightarrow S_k$

3. Estimate of state determined using data $\forall t \leq t_k$

4. We will write this as $\widehat{x}_{k|k}$, $\Lambda_{k|k}$

5. **Prediction** is $\widehat{x}_{k+1|k} = S_k \, \widehat{x}_{k|k}$, $\Lambda_{k+1|k} = S_k \, \Lambda_{k|k} \, S_k^T$

Sequential least squares with dynamics

1. Starting parameter estimates $\hat{x}_{0|0}$, $\Lambda_{0|0}$

2. For each data set y_k & covariance G_k, $k = 1, \ldots, n$:

   (a) Prediction

   $$\hat{x}_{k|k-1} = S_k \, \hat{x}_{k-1|k-1} \qquad \Lambda_{k|k-1} = S_k \, \Lambda_{k-1|k-1} \, S_k^T$$

   (b) Gain $K_k = \Lambda_{k|k-1} A_k^T \left(A_k \Lambda_{k|k-1} A_k^T + G_k \right)^{-1}$

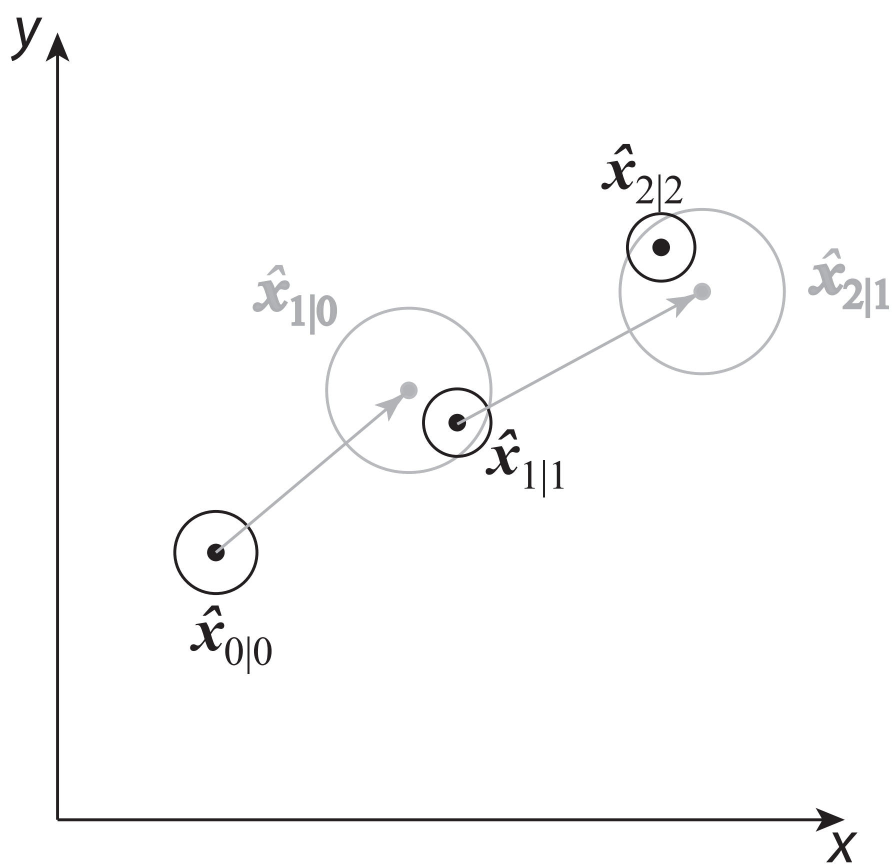
   (c) Update

   $$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \Delta y_k \qquad \Lambda_{k|k} = (I - K_k A_k) \, \Lambda_{k|k-1}$$

Sequential least squares with dynamics

	A priori solution	$\longleftarrow$	$\widehat{x}_{0\|0},\ \Lambda_{0\|0}$
	$\downarrow$		
$\rightarrow$	Prediction	$\longrightarrow$	$\widehat{x}_{k\|k-1},\ \Lambda_{k\|k-1}$
$\uparrow$	$\downarrow$		
	New Observation	$\longleftarrow$	y_k, G_k
$\uparrow$	$\downarrow$		
	Gain matrix	$\longrightarrow$	K_k
$\uparrow$	$\downarrow$		
$\leftarrow$	Update	$\longrightarrow$	$\widehat{x}_{k\|k},\ \Lambda_{k\|k}$

State vector with 2-D position and velocity with observations



Introduction

- Stochastic filters estimation of a **stochastic process**

- A stochastic process is a collection of random variables indexed by t

- Given the value of the process at time t_1 (and past values), it is not possible to predict with 100% certainty what its value will be at time $t_2 > t_1$

- Such processes are described by the familiar mathematics of probability and random variables

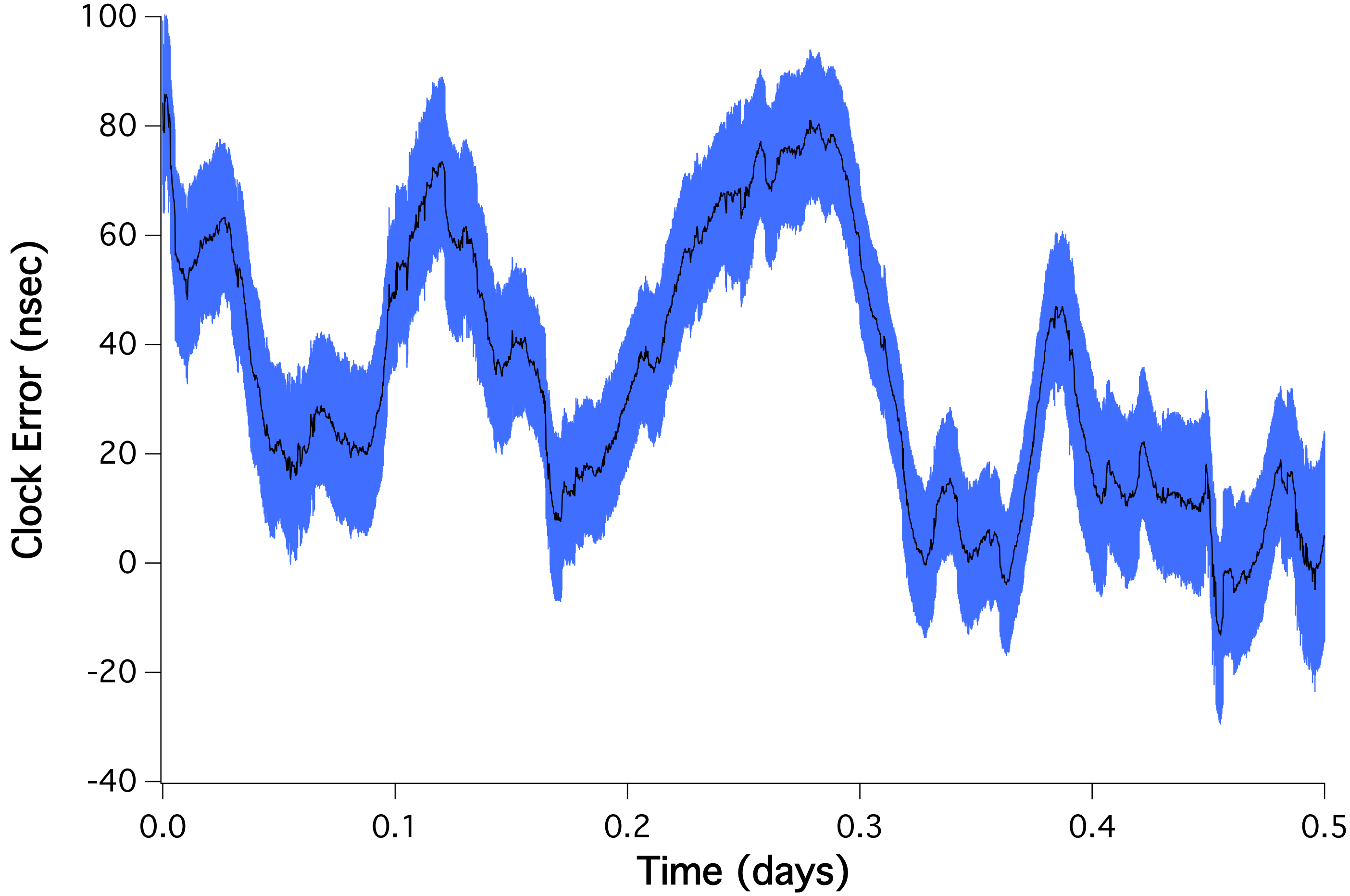
GPS-related examples of stochastic processes

- The wet delay, which is controlled over a range of timescales by turbulent transport of water vapor

- Site and satellite clocks errors

- GPS antenna positions under certain circumstances (vehicle tracking, glacier motion, earthquakes)

Estimated clock errors (Gonzak receiver)



Stochastic processes and dynamic systems

- A stochastic process acts to "force" the dynamic system

$$x_k = S_k\, x_{k-1} + R_k\, \xi_k$$

- ξ_k is a zero-mean stochastic-process vector

- $\left\langle \xi_j \xi_k^T \right\rangle = Q_k \delta_{jk}$

- R_k is a matrix

- $\left\langle R_j \xi_j x_k^T \right\rangle = 0$ for $j > k$

Update with stochastic process

- Since $\langle \xi_k \rangle = 0$ we have (using same math as before, and taking advantage of independence)

$$x_{k+1|k} = S_k \, x_{k-1|k-1}$$

$$\Lambda_{k+1|k} = S_k \, \Lambda_{k-1|k-1} \, S_k^T + R_k \, Q_k \, R_k^T$$

- How else would sequential least-squares change?

Sequential least squares with dynamics

	A priori solution	$\longleftarrow$	$\widehat{x}_{0\|0},\ \Lambda_{0\|0}$
	$\downarrow$		
$\rightarrow$	Prediction	$\longrightarrow$	$\widehat{x}_{k\|k-1},\ \Lambda_{k\|k-1}$
$\uparrow$	$\downarrow$		
	New Observation	$\longleftarrow$	y_k, G_k
$\uparrow$	$\downarrow$		
	Gain matrix	$\longrightarrow$	K_k
$\uparrow$	$\downarrow$		
$\leftarrow$	Update	$\longrightarrow$	$\widehat{x}_{k\|k},\ \Lambda_{k\|k}$

Kalman Filter equations

Prediction

$$\widehat{x}_{k|k-1} = S_k \, \widehat{x}_{k-1|k-1}$$

$$\Lambda_{k|k-1} = S_k \, \Lambda_{k-1|k-1} \, S_k^T + R_k \, Q_k \, R_k^T$$

Gain

$$K_k = \Lambda_{k|k-1} A_k^T \left(A_k \Lambda_{k|k-1} A_k^T + G_k \right)^{-1}$$

Update

$$\widehat{x}_{k|k} = \widehat{x}_{k|k-1} + K_k \Delta y_k$$

$$\Lambda_{k|k} = (I - K_k A_k) \, \Lambda_{k|k-1}$$

Point positioning example

- Let us take the example of point positioning using pseudorange with GPS

- Each epoch t_k we observe n_k satellites from a static (non-moving) receiver

- Our simplified observation equation for the jth satellite ($j = 1, \ldots, n_k$) is

$$R^j(t_k) = |\vec{x}_s^j - \vec{x}^r| + c(t_k)$$

  where R^j is the LC pseudorange corrected for the satellite clock error, $\vec{x}_s^j$ is the satellite position, $\vec{x}^r$ is receiver position, and $c(t_k)$ is the receiver clock

Point positioning example: State vector

- We will treat the clock error as a stochastic process

- The parameter (state) vector will be

$$x_k = \begin{bmatrix} x_r \\ y_r \\ z_r \\ c_k \end{bmatrix}$$

where $c_k = c(t_k)$

Point positioning example: Noise model

- We will assume, for this example, a zero-mean Gaussian white-noise model for the clock error

- This model implies that

$$\langle c_j \rangle = 0$$

$$\langle c_j c_k \rangle = \sigma_c^2 \delta_{jk}$$

Point positioning example: Dynamic model

- Putting these two equations together we can write our dynamic model as

$$x_{k+1} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{S_k} x_k + \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}}_{R_k} c_{k+1}$$

- We also have that Q_k is a scalar with $Q_k = \sigma_c^2$

Point positioning example: Prediction

- The prediction step gives us

$$\hat{x}_{k|k-1} = \begin{bmatrix} \hat{x}^r_{k-1|k-1} \\ \hat{y}^r_{k-1|k-1} \\ \hat{z}^r_{k-1|k-1} \\ 0 \end{bmatrix}$$

$$\Lambda_{k|k-1} = \left(\begin{array}{c|c} P_{k-1|k-1} & 0 \\ \hline 0 & \sigma_c^2 \end{array}\right)$$

where P is the position sub matrix of the covariance matrix