Biomedical Signal Processing

- Kalman Filter -

2021. 5. 25

Prof: Boreom Lee

INTRODUCTION

- Getting to understand a system – quite a challenge

1. To create a model
   - An abstraction of the system – captures the system's important features
2. To learn the system's behavior
   - By measuring its "spontaneous" output or its input-output relationship

➔ Pessimist view

- All models are wrong since they only represent part of the real thing
- All measurements are wrong too because they are also incomplete and noisy as well

INTRODUCTION

- Kalman filter approach
  - A more optimistic view
    - To optimally combine the not so perfect information obtained via both modeling and measurements → the best possible knowledge about a system's state
  - An example of a so-called least-square filter
    - Signal and noise separation using a minimum square error fit
      - This principle – first introduced by Norbert Wiener in the late 1940s → novel and appealing but not easy to apply in practical engineering problems
    - The basis for an algorithm using the state space method introduced by Rudolf E. Kalman in 1960 → a practical procedure to implement the novel least-squre filtering method although not initially recognized

INTRODUCTION

- Basic idea
  - Multiple (probably regular) measurements of a system's output → these measurements and knowledge of the system – used to reconstruct the state of the system
  - Steps in the Kalman filter process:
    1. A priori estimate of the state of the system before a measurement is made
    2. Subsequently, after the measurement, a new a posteriori estimate by *fusing the a priori prediction with this latest measurement*
  - In examples
    - Trivial dynamics assumed: prediction of the next state = the current state
    - More complex system dynamics
      - The prediction step 1 – reflect this complexity
      - Wouldn't change step 2 (i.e., the procedure of the assimilation/fusion of a prediction with a measurement)

INTRODUCTORY TERMINOLOGY

- Normal or Gaussian Distribution
  - The noise components in the Kalman filter approach in this chapter – Gaussian white noise terms with zero mean
    - Cf. Kalman filter versions with non-Gaussian noise
  - Probability density function (PDF) of the normal or Gaussian distribution:
    - μ: mean (zero in this case), σ: standard deviation

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Minimum Mean-Square Error
  - Several ways to achieve optimal fit of a series of predictions with a series of observations
  - Minimum mean-square error (MMSE) approach
    - The square of difference btw an estimate $\hat{y}$ and a target value y, $(\hat{y} - y)^2$ → impression of how well the two fit
    - A series of N estimates and target values
      - A quantification of the error E by determining the sum of the squares of their differences $E = \sum_{i=1}^{N} (\hat{y} - y)^2$ → minimize this w.r.t. the parameter of interest
      - Ex: $\hat{y}_i = ax_i$ → the minimum of E w.r.t. parameter a found by $\partial E / \partial a = 0$

INTRODUCTORY TERMINOLOGY

- Recursive Algorithm
  - Resting potential of a neuron based on a series of measurements: $z_1, z_2, \ldots, z_n$
  - The estimated mean of the n measurements – used as the estimate of the resting potential: $\widehat{m}_n = \frac{z_1 + z_2 + \cdots + z_n}{n}$
    - Not an optimal basis for the development of an algorithm
    - n memory locations needed for measurements $z_1 \sim z_n$
    - All measurements need to be completed before the estimate
  - Recursive approach
    - Used in the Kalman filter algorithm
    - Estimates the resting potential after each measurement using updates for each measurement:

$$\text{1st measurement: } \widehat{m}_1 = z_1, \text{ 2nd measurement: } \widehat{m}_2 = \frac{1}{2}\widehat{m}_1 + \frac{1}{2}z_2, \text{ 3rd measurement: } \widehat{m}_3 = \frac{2}{3}\widehat{m}_2 + \frac{1}{3}z_3$$

      - Update the estimate at each measurement
      - Only need memory for the previous estimate and the new measurement to make a new estimate

INTRODUCTORY TERMINOLOGY

- Data Assimilation
  - Ex: two measurements y_1 and y_2
    - Combine/fuse these observations into a single estimates y
    - The uncertainty of the two measurements – given by their variance (=standard deviation squared) s_1 and s_2, respectively
    - Fusion principle used in the Kalman approach based on MMSE: $$y = \frac{s_2}{s_1 + s_2} y_1 + \frac{s_1}{s_1 + s_2} y_2$$ for the new estimate
      - Its variance s obtained from: $$\frac{1}{s} = \frac{1}{s_1} + \frac{1}{s_2}$$

      → an improved estimate based on two separate ones (Fig. 19.1)
  - Fig. 19.1
    - Two distributions with mean values of 5 and 10 & s.d.'s of 1 and 3 → the best estimate – reflected by a distribution with a mean of 5.5. and s.d. of $\sqrt{0.9} \approx 0.95$
    - The measurement with less uncertainty – more weight in the estimate; the one with more uncertainty – only contributes a little
  - ❖ In the context of Kalman fiter
    - The terms <u>assimilation</u> and <u>blending</u> – sometimes used instead of data fusion → describe the <u>combination of estimate and measurement</u>

INTRODUCTORY TERMINOLOGY

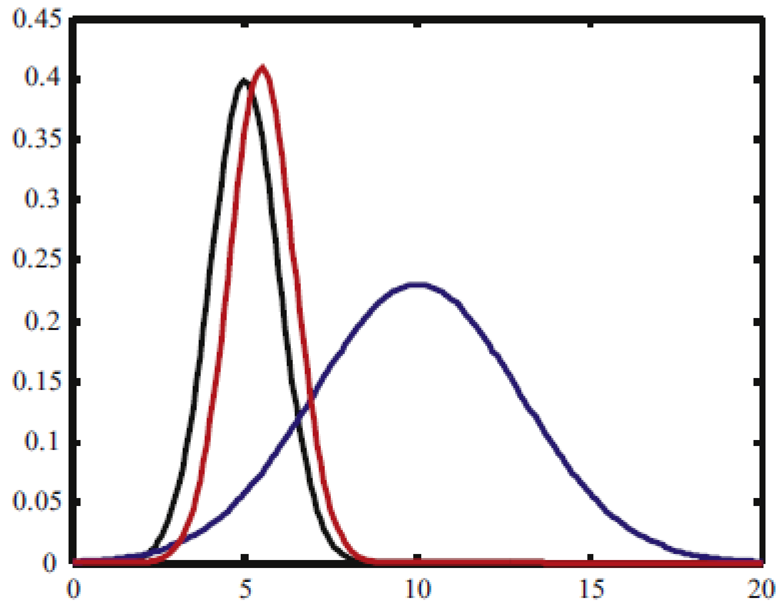


FIGURE 19.1 An example of the fusion procedure.

INTRODUCTORY TERMINOLOGY

- State Model
  - One way of characterizing the dynamics of a system = to describe how it evolves through a number of states
    - State – a vector of variables in a state space → the system's dynamics – described by its path through state space
  - Simple example: a moving object
    - If no acceleration → its state – described by the coordinates of its position $\vec{x}$ in three dimension (3D space = its state space)
    - Dynamics – seen by plotting position versus time → its velocity $d\vec{x}/dt$(the time derivative of its position)

  - Another example: $\ddot{x}_1 + b\dot{x}_1 + cx_1 = 0$ (a 2nd-order ODE)
    - Define $\underline{\dot{x}_1 = x_2}$ (a) & rewrite the ODE as $\underline{\dot{x}_2 + bx_2 + cx_1 = 0}$ (b) → the single 2nd-order ODE – rewritten as two 1st-order ODEs

INTRODUCTORY TERMINOLOGY

- State Model
  - Another example: $\ddot{x}_1 + b\dot{x}_1 + cx_1 = 0$ (a 2nd-order ODE)
    - Define a system state vector: $\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$
      - Not the position of a physical object but a vector that defines the state of the system
      - (a) & (b) – compactly written as: $\frac{d\vec{x}}{dt} = A\vec{x}$, with $A = \begin{pmatrix} 0 & 1 \\ -c & -b \end{pmatrix}$
        - The dynamics – expressed as an operation of matrix A on state vector $\vec{x}$(this notation – used as an alternative to any ODE)

- Bayesian Analysis
  - Kalman filter approach: update the pdf (i.e., the mean and variance) of the state x of a system based on measurements z, i.e., $p(x|z)$
  - Conditional probabilities and distributions: the topics of Bayesian statistics → the Kalman approach = a form of Bayesian analysis

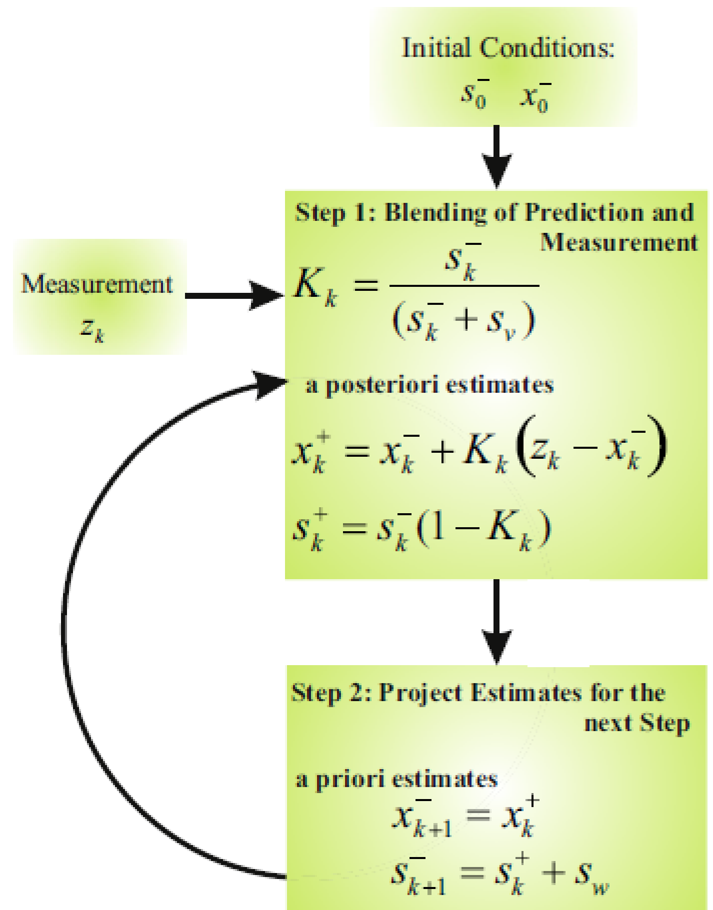
DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

- A simple version of a Kalman filter application
  - Brown and Hwang (1997): a derivation for a vector observation
  - Here: the equivalent simplification for a scalar
  - All equations with a border

  - A simple process x
    - A neuron's resting potential (i.e., its membrane potential in the absence of overt stimulation)
    - The update rule for the state of the process: $x_{k+1} = x_k + w$
  - The process measurement *z* defined (as simple as possible) as: $z_k = x_k + v$
  - *w* and *v*: Gaussian white noise variables
    - Model the process and measurement noise, respectively
    - Assumed stationary and ergodic (for the sake of this example)
    - Mean for both = zero; the associated variance values = s_w and s_v
    - The covariance of *w* and *v* = zero (∵they are independent)

DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

FIGURE 19.2 Flowchart of the Kalman filter procedure derived in this section and implemented in the MATLAB script pr19_1.m. In this MATLAB script, steps 1 – 2 are indicated. Step 1: Eqs. (19.19), (19.15), and (19.20); Step 2: Eqs. (19.21) and (19.22). The equations in the diagram are derived in the text.



DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

- A priori estimate x_k^-
  - A prior error $x_k - x_k^-$ → the variance s_k^- associated with this error:

$$s_k^- = E\left[\left(x_k - x_k^-\right)^2\right]$$

  - Assimilate a measurement z_k with a priori estimate x_k^- → a posteriori estimate x_k^+

  ❖ "hat" (i.e., $\hat{x}_k^-$) indicate an estimate → hats – omitted here to simplify
- A posteriori estimate x_k^+
  - Determined by the fusion of the a priori prediction and the new measurement: $\boxed{x_k^+ = x_k^- + K_k\left(z_k - x_k^-\right)}$ (c)
  - K_k: the blending factor → determine to what extent the measurement and a priori estimate affect the new a posteriori estimate
  - A posteriori error $x_k - x_k^+$ → the variance s_k^+ with this error:

$$s_k^+ = E\left[\left(x_k - x_k^+\right)^2\right]$$

DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

- Substitution of $z_k = x_k + v$ into (c): $x_k^+ = x_k^- + K_k(x_k + v - x_k^-)$

→ $$s_k^+ = E\left[(x_k - x_k^+)^2\right] = E\left[\left((x_k - x_k^-) - K_k(x_k + v - x_k^-)\right)^2\right]$$

$$s_k^+ = s_k^- - 2K_k s_k^- + K_k^2 s_k^- + K_k^2 s_v$$

$$\underline{s_k^+ = s_k^-(1 - K_k)^2 + K_k^2 s_v} \quad \text{or} \quad \underline{s_k^+ = s_k^- - 2K_k s_k^- + (s_k^- + s_v)K_k^2} \quad \text{(d)}$$

- $x_k - x_k^-$: the a priori estimation error & uncorrelated with measurement noise v
- A posteriori estimate – optimized using optimal blending in (c)
  - Minimize the variance s_k^+ (= square of the estimated error) w.r.t. K_k
  - Differentiate the expression for s_k^+ w.r.t K_k & set the result to zero:

$$-2s_k^- + 2(s_k^- + s_v)K_k = 0$$

→ the expression for an optimal K_k: $\boxed{K_k = \frac{s_k^-}{(s_k^- + s_v)}.}$

DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

- Substituting the expression for K_k in (c)
  - Meaning of the optimized result:

$$\boxed{K_k = \frac{s_k^-}{(s_k^- + s_v)}.} \rightarrow x_k^+ = x_k^- + \frac{s_k^-}{(s_k^- + s_v)}(z_k - x_k^-)$$

  - If a priori estimate – unreliable
    - Its large variance $s_k^- \rightarrow \infty$: $K_k \rightarrow 1$ (ignore the estimate)
    - Believe more of the measurement: $x_k^+ \rightarrow z_k$
  - If the measurement – completely unreliable
    - $s_v \rightarrow \infty$: $K_k \rightarrow 0$ (ignore the measurement)
    - Believe the a prior estimate: $x_k^+ \rightarrow x_k^-$

DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

- An expression for s_v:
  - $\boxed{K_k = \frac{s_k^-}{(s_k^- + s_v)}.}$ $\rightarrow$ $s_v = \frac{s_k^- - K_k s_k^-}{K_k}$: substituted in (d)
  - A posteriori error based on an optimized blending factor K_k:

$$s_k^+ = s_k^-(1-K_k)^2 + \frac{K_k^2 s_k^- - K_k^3 s_k^-}{K_k} \quad , \quad s_k^+ = s_k^- - 2K_k s_k^- + K_k^2 s_k^- + K_k s_k^- - K_k^2 s_k^-$$

$$\rightarrow \boxed{s_k^+ = s_k^-(1-K_k)}$$

    - Different expressions available depending on how one simplifies (d)
    - Use this here because of its simplicity

DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

- Projection towards the next time step at t_{k+1}
  - Depending on the model for the process to measure from
  - A simple model in this example ($x_{k+1} = x_k + w$)
    - A priori estimate of x = a posteriori estimate of the previous time step:

      $$\boxed{x_{k+1}^{-} = x_k^{+}} \quad \text{(e)}$$

      - Ignore the noise (w) term in $x_{k+1} = x_k + w$ ($\because$the expectation of w is zero)
      - In many applications, more complicated or even an extremely complex procedure depending on the dynamics btw t_k and t_{k+1}
    - A priori estimate of the variance at t_{k+1}: $e_{k+1}^{-} = x_{k+1} - x_{k+1}^{-}$
      - Based on the error of the prediction in (e)
      - Substitute $x_{k+1} = x_k + w$ and (e) into this expression: $e_{k+1}^{-} = x_k + w - x_k^{+}$

        → $e_{k+1}^{-} = e_k^{+} + w$ ($\because$a posteriori error at t_k: $e_k^{+} = x_k - x_k^{+}$)

DERIVATION OF A KALMAN FILTER FOR A SIMPLE CASE

- Projection towards the next time step at t_{k+1}
  - A priori variance at t_{k+1} associated with this expression for the a priori error ($e_{k+1}^- = e_k^+ + w$)
    - Depending on the a posteriori error at t_k and the variance of the noise process w: $s_{k+1}^- = E\left[(e_k^+ + w)^2\right]$
    - Final: $\boxed{s_{k+1}^- = s_k^+ + s_w}$ ($\because w$ and the error in the estimate – uncorrelated)

MATAB® EXAMPLE

- A recording of a membrane potential of a nonstimulated neuron
  - The resting potential Vm = -70 mV
  - s.d.'s of the process noise and the measurement noise = 1 and 2 mV, respectively

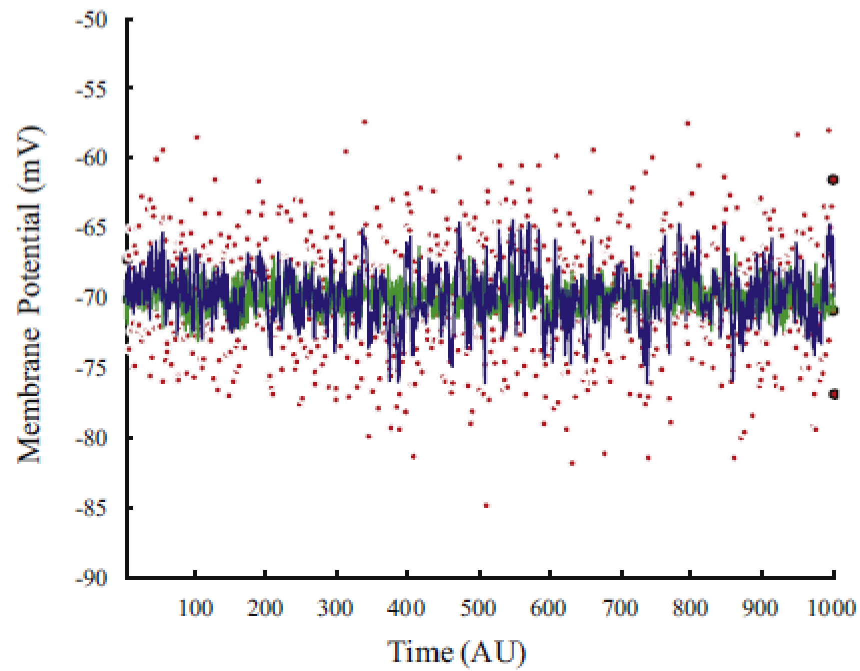


FIGURE 19.3 Example of the Kalman filter application created with the attached MATLAB script pr19_1. The true values are green, the measurements are represented by the red dots, and the Kalman filter output is the blue line. Although the result is certainly not perfect, it can be seen that the Kalman filter output is much closer to the real values than the measurements.

USE OF THE KALMAN FILTER TO ESTIMATE MODEL PARAMETERS

- The vector-matrix version of the Kalman filter
  - Can estimate a series of variables
  - Can also be used to estimate model parameters

- Constant parameter of the model or very slowly changing parameter (considered a constant)
  - Treats and estimates this parameter as a constant with noise as with x_k in $x_{k+1} = x_k + w$

- Noise component – allows the algorithm of the Kalman filter to find the best fit of the model parameter at hand