Kalman Filter Theory and Applications
Equation Drilldown

Michael L. Carroll

November 1, 2020

- Mathematical Formulation of the Problem

- Mathematical Formulation of the Problem
- **Drilldown on State Dynamics and Covariance Extrapolation Equations**

- Mathematical Formulation of the Problem
- **Drilldown on State Dynamics and Covariance Extrapolation Equations**
- The Five Kalman Filter Equations

- Mathematical Formulation of the Problem
- **Drilldown on State Dynamics and Covariance Extrapolation Equations**
- The Five Kalman Filter Equations
- Examples

- Mathematical Formulation of the Problem
- **Drilldown on State Dynamics and Covariance Extrapolation Equations**
- The Five Kalman Filter Equations
- Examples
- Exercises

- Mathematical Formulation of the Problem
- **Drilldown on State Dynamics and Covariance Extrapolation Equations**
- The Five Kalman Filter Equations
- Examples
- Exercises

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
- $x(k)$ is the **state vector** with n elements, written as a column vector:

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
- $x(k)$ is the **state vector** with n elements, written as a column vector:
- $x(k) = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} (k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix}$

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
- $x(k)$ is the **state vector** with n elements, written as a column vector:
- $x(k) = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} (k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix}$

- Discrete State Dynamics

- Discrete State Dynamics
- KF doesn't work for nonlinear dynamics function $f[x(k-1)]$

- Discrete State Dynamics
- KF doesn't work for nonlinear dynamics function $f[x(k-1)]$
- Restriction to matrices: $x(k) = \Phi(k)x(k-1) + w(k)$

- Discrete State Dynamics
- KF doesn't work for nonlinear dynamics function $f[x(k-1)]$
- Restriction to matrices: $x(k) = \Phi(k)x(k-1) + w(k)$
- $\Phi(k)$ is called the **state transition matrix**. It is $n \times n$ in size.

- Discrete State Dynamics
- KF doesn't work for nonlinear dynamics function $f[x(k-1)]$
- Restriction to matrices: $x(k) = \Phi(k)x(k-1) + w(k)$
- $\Phi(k)$ is called the **state transition matrix**. It is $n \times n$ in size.

Understanding Kalman Filter Notation

Drilldown on State Dynamics

- State Transition Matrix: $\Phi(k-1) = \Phi(k, k-1)$

- State Transition Matrix: $\Phi(k-1) = \Phi(k, k-1)$
- Extrapolates state vector from time step $k-1$ to time step k (pulling from past to present)

- State Transition Matrix: $\Phi(k-1) = \Phi(k, k-1)$
- Extrapolates state vector from time step $k-1$ to time step k (pulling from past to present)
- Tells you how the state would evolve in the absence of forcing functions

- State Transition Matrix: $\Phi(k-1) = \Phi(k, k-1)$
- Extrapolates state vector from time step $k-1$ to time step k (pulling from past to present)
- Tells you how the state would evolve in the absence of forcing functions
- Remember: this is still just a model. We don't really know what the true state is, because the random forcing function (process noise) generates disturbances!

- State Transition Matrix: $\Phi(k-1) = \Phi(k, k-1)$
- Extrapolates state vector from time step $k-1$ to time step k (pulling from past to present)
- Tells you how the state would evolve in the absence of forcing functions
- Remember: this is still just a model. We don't really know what the true state is, because the random forcing function (process noise) generates disturbances!

Understanding Kalman Filter Notation

Drilldown on State Dynamics

- We could throw in other factors and terms

- We could throw in other factors and terms
  - A matrix $\Gamma(k)$ to distribute the Gaussian white noise to the states

- We could throw in other factors and terms
  - A matrix $\Gamma(k)$ to distribute the Gaussian white noise to the states
  - An additional, deterministic control variable $u(k)$, and associated distribution matrix $\Lambda(k)$

- We could throw in other factors and terms
  - A matrix $\Gamma(k)$ to distribute the Gaussian white noise to the states
  - An additional, deterministic control variable $u(k)$, and associated distribution matrix $\Lambda(k)$
- For simplicity we'll generally leave the control variable out of the picture

- We could throw in other factors and terms
  - A matrix $\Gamma(k)$ to distribute the Gaussian white noise to the states
  - An additional, deterministic control variable $u(k)$, and associated distribution matrix $\Lambda(k)$
- For simplicity we'll generally leave the control variable out of the picture
- We are deliberately trying to keep the notation as simple as possible

- We could throw in other factors and terms
  - A matrix $\Gamma(k)$ to distribute the Gaussian white noise to the states
  - An additional, deterministic control variable $u(k)$, and associated distribution matrix $\Lambda(k)$
- For simplicity we'll generally leave the control variable out of the picture
- We are deliberately trying to keep the notation as simple as possible

The Kalman Filter Solution:
State Estimate Extrapolation

The Kalman Filter Solution:
State Estimate Extrapolation

- Given an estimate $\hat{x}^{+}(k-1)$ at time $k-1$, we extrapolate it forward in time by the state transition matrix:

$$\hat{x}^{-}(k) = \Phi(k)\hat{x}^{+}(k-1) \tag{1}$$

The Kalman Filter Solution:
State Estimate Extrapolation

- Given an estimate $\hat{x}^+(k-1)$ at time $k-1$, we extrapolate it forward in time by the state transition matrix:

$$\hat{x}^-(k) = \Phi(k)\hat{x}^+(k-1) \tag{1}$$

- Q: Why isn't the white noise sequence w included in this extrapolation?

- Given an estimate $\hat{x}^+(k-1)$ at time $k-1$, we extrapolate it forward in time by the state transition matrix:

$$\hat{x}^-(k) = \Phi(k)\hat{x}^+(k-1) \tag{1}$$

- Q: Why isn't the white noise sequence w included in this extrapolation?
  A: Because the white noise forcing function is a **zero-mean** process. Thus, on average, the equation is homogeneous and we should therefore use the homogeneous solution, i.e., the state transition matrix.

- Given an estimate $\hat{x}^{+}(k-1)$ at time $k-1$, we extrapolate it forward in time by the state transition matrix:

$$\hat{x}^{-}(k) = \Phi(k)\hat{x}^{+}(k-1) \tag{1}$$

- Q: Why isn't the white noise sequence w included in this extrapolation?
  A: Because the white noise forcing function is a **zero-mean** process. Thus, on average, the equation is homogeneous and we should therefore use the homogeneous solution, i.e., the state transition matrix.

- Q: How certain are we that our solution is right?

The Kalman Filter Solution:
Covariance Extrapolation

- Q: How certain are we that our solution is right?
  A: The answer is given by the **estimation error covariance matrix** or often just called the covariance matrix $P(k)$

- Q: How certain are we that our solution is right?
  A: The answer is given by the **estimation error covariance matrix** or often just called the covariance matrix $P(k)$
- Just like the state estimate, the covariance matrix is propagated forward in time before being corrected:

The Kalman Filter Solution:
Covariance Extrapolation

- Q: How certain are we that our solution is right?
  A: The answer is given by the **estimation error covariance matrix** or often just called the covariance matrix $P(k)$
- Just like the state estimate, the covariance matrix is propagated forward in time before being corrected:

$$P^{-}(k) = \Phi(k) P^{+}(k-1) \Phi(k)^{\top} + Q(k) \tag{2}$$

where $Q(k)$ is a matrix called the **process noise covariance matrix**

- Q: How certain are we that our solution is right?
  A: The answer is given by the **estimation error covariance matrix** or often just called the covariance matrix $P(k)$
- Just like the state estimate, the covariance matrix is propagated forward in time before being corrected:

$$P^-(k) = \Phi(k)P^+(k-1)\,\Phi(k)^\top + Q(k) \tag{2}$$

  where $Q(k)$ is a matrix called the **process noise covariance matrix**
- $Q(k)$ measures the amount of dispersion in the white noise forcing function $w(k)$

The Kalman Filter Solution:
Covariance Extrapolation

- Q: How certain are we that our solution is right?
  A: The answer is given by the **estimation error covariance matrix** or often just called the covariance matrix $P(k)$
- Just like the state estimate, the covariance matrix is propagated forward in time before being corrected:

$$P^{-}(k) = \Phi(k) P^{+}(k-1) \Phi(k)^{\top} + Q(k) \tag{2}$$

  where $Q(k)$ is a matrix called the **process noise covariance matrix**
- $Q(k)$ measures the amount of dispersion in the white noise forcing function $w(k)$
- Later we'll define $P^{-}(k)$ in terms of the expected error in the estimation $\tilde{x}^{-}(k) = \hat{x}^{-}(k) - x(k)$

- Q: How certain are we that our solution is right?
  A: The answer is given by the **estimation error covariance matrix** or often just called the covariance matrix $P(k)$
- Just like the state estimate, the covariance matrix is propagated forward in time before being corrected:

$$P^{-}(k) = \Phi(k)P^{+}(k-1)\,\Phi(k)^{\top} + Q(k) \tag{2}$$

  where $Q(k)$ is a matrix called the **process noise covariance matrix**
- $Q(k)$ measures the amount of dispersion in the white noise forcing function $w(k)$
- Later we'll define $P^{-}(k)$ in terms of the expected error in the estimation $\tilde{x}^{-}(k) = \hat{x}^{-}(k) - x(k)$

- What does Eq. (2) really mean?

- What does Eq. (2) really mean?
- The term $\Phi(k)P^{+}(k-1)\,\Phi(k)^{\top}$ represents the effect that the state dynamics has on the estimation error covariance

- What does Eq. (2) really mean?
- The term $\Phi(k)P^{+}(k-1)\,\Phi(k)^{\top}$ represents the effect that the state dynamics has on the estimation error covariance
  - P(k) is positive definite matrix and Φ tends to increase its norm

- What does Eq. (2) really mean?
- The term $\Phi(k)P^{+}(k-1)\,\Phi(k)^{\top}$ represents the effect that the state dynamics has on the estimation error covariance
  - P(k) is positive definite matrix and Φ tends to increase its norm
- The other term $Q(k)$ represents the increased uncertainty added at each step due to the process noise inherent in the system

- What does Eq. (2) really mean?
- The term $\Phi(k)P^{+}(k-1)\,\Phi(k)^{\top}$ represents the effect that the state dynamics has on the estimation error covariance
    - P(k) is positive definite matrix and Φ tends to increase its norm
- The other term $Q(k)$ represents the increased uncertainty added at each step due to the process noise inherent in the system

- Where are we?

- Where are we?
  - Drilled down on state dynamics

- Where are we?
  - Drilled down on state dynamics
  - Introduced Φ and Q and discussed time extrapolation of P

- Where are we?
  - Drilled down on state dynamics
  - Introduced Φ and Q and discussed time extrapolation of P
- What's next?

- Where are we?
  - Drilled down on state dynamics
  - Introduced Φ and Q and discussed time extrapolation of P
- What's next?
  - Drill down on the Kalman gain equation as a function of P, Q, R

- Where are we?
    - Drilled down on state dynamics
    - Introduced Φ and Q and discussed time extrapolation of P
- What's next?
    - Drill down on the Kalman gain equation as a function of P, Q, R
    - Present all five Kalman filter equations

- Where are we?
  - Drilled down on state dynamics
  - Introduced Φ and Q and discussed time extrapolation of P
- What's next?
  - Drill down on the Kalman gain equation as a function of P, Q, R
  - Present all five Kalman filter equations