# Kalman Filter Theory and Applications
# Equation Drilldown

https://github.com/musicarroll/kalman_course

Michael L. Carroll

June 16, 2023

# Part I

## The Basic Kalman Equations

### Topics

- Understanding the Equations: Heuristic Introduction

- Understanding the Equations: Heuristic Introduction
- **Equation Drilldown: Taking the Equations Apart**

- Understanding the Equations: Heuristic Introduction
- **Equation Drilldown: Taking the Equations Apart**
- State Space Concepts

- Understanding the Equations: Heuristic Introduction
- **Equation Drilldown: Taking the Equations Apart**
- State Space Concepts

- **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

- Given two models:

- Given two models:
  - State Dynamics Model
    (sometimes called the System or Process Model)

- Given two models:
  - State Dynamics Model
    (sometimes called the System or Process Model)
  - Measurement Model
    (sometimes called the Observation Model)

- Given two models:
  - State Dynamics Model
    (sometimes called the System or Process Model)
  - Measurement Model
    (sometimes called the Observation Model)
- Find a way to **optimally** combine our knowledge of the **predicted state estimate** with our knowledge of the **actual measurement** to produce an **updated state estimate**

- Given two models:
  - State Dynamics Model
    (sometimes called the System or Process Model)
  - Measurement Model
    (sometimes called the Observation Model)
- Find a way to **optimally** combine our knowledge of the **predicted state estimate** with our knowledge of the **actual measurement** to produce an **updated state estimate**
  - I.e., find the optimal gain with which to weight the measurement residual (difference between actual and predicted measurement)

- Given two models:
  - State Dynamics Model
    (sometimes called the System or Process Model)
  - Measurement Model
    (sometimes called the Observation Model)
- Find a way to **optimally** combine our knowledge of the **predicted state estimate** with our knowledge of the **actual measurement** to produce an **updated state estimate**
  - I.e., find the optimal gain with which to weight the measurement residual (difference between actual and predicted measurement)
  - Use weighted residual to increment or update the state estimate

- Given two models:
  - State Dynamics Model
    (sometimes called the System or Process Model)
  - Measurement Model
    (sometimes called the Observation Model)
- Find a way to **optimally** combine our knowledge of the **predicted state estimate** with our knowledge of the **actual measurement** to produce an **updated state estimate**
  - I.e., find the optimal gain with which to weight the measurement residual (difference between actual and predicted measurement)
  - Use weighted residual to increment or update the state estimate
  - Use Kalman gain also to update the estimation uncertainty (estimation error covariance aka "covariance")

- Given two models:
  - State Dynamics Model
    (sometimes called the System or Process Model)
  - Measurement Model
    (sometimes called the Observation Model)
- Find a way to **optimally** combine our knowledge of the **predicted state estimate** with our knowledge of the **actual measurement** to produce an **updated state estimate**
  - I.e., find the optimal gain with which to weight the measurement residual (difference between actual and predicted measurement)
  - Use weighted residual to increment or update the state estimate
  - Use Kalman gain also to update the estimation uncertainty (estimation error covariance aka "covariance")

Given:

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
    - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
  - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$
  - $x(k)$=state vector, $f$=true dynamics function, $w(k)$=process noise

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
  - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$
  - $x(k)$=state vector, $f$=true dynamics function, $w(k)$=process noise
- Measurement Model: $z(k) = h[x(k)] + v(k)$

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
  - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$
  - $x(k)$=state vector, $f$=true dynamics function, $w(k)$=process noise
- Measurement Model: $z(k) = h[x(k)] + v(k)$
  - $z(k)$=measurement, $h$=measurement function, $v(k)$=measurement noise

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
  - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$
  - $x(k)$=state vector, $f$=true dynamics function, $w(k)$=process noise
- Measurement Model: $z(k) = h[x(k)] + v(k)$
  - $z(k)$=measurement, $h$=measurement function, $v(k)$=measurement noise

Find:

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
  - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$
  - $x(k)$=state vector, $f$=true dynamics function, $w(k)$=process noise
- Measurement Model: $z(k) = h[x(k)] + v(k)$
  - $z(k)$=measurement, $h$=measurement function, $v(k)$=measurement noise

Find:

- Optimal gain $K(k)$ that allows us to update our state estimate in the following way:
  $\hat{x}(k) = f[\hat{x}(k-1)] + K(k)\,[\,z(k) - h(f[\hat{x}(k-1)])\,]$

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
  - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$
  - $x(k)$=state vector, $f$=true dynamics function, $w(k)$=process noise
- Measurement Model: $z(k) = h[x(k)] + v(k)$
  - $z(k)$=measurement, $h$=measurement function, $v(k)$=measurement noise

Find:

- Optimal gain $K(k)$ that allows us to update our state estimate in the following way:
  $\hat{x}(k) = f[\hat{x}(k-1)] + K(k)\,[\,z(k) - h(f[\hat{x}(k-1)])\,]$
- (For now, we'll ignore the update of the covariance)

Given:

- State Dynamics Model: $x(k) = f[x(k-1)] + w(k)$
  - $k$ indicates the time step; the actual time at that time step is $t_k$, and the time increment is $\Delta t = t_k - t_{k-1}$
  - $x(k)$=state vector, $f$=true dynamics function, $w(k)$=process noise
- Measurement Model: $z(k) = h[x(k)] + v(k)$
  - $z(k)$=measurement, $h$=measurement function, $v(k)$=measurement noise

Find:

- Optimal gain $K(k)$ that allows us to update our state estimate in the following way:
  $\hat{x}(k) = f[\hat{x}(k-1)] + K(k)\,[\,z(k) - h(f[\hat{x}(k-1)])\,]$
- (For now, we'll ignore the update of the covariance)

- Original Kalman filter solution applies only to ...

- Original Kalman filter solution applies only to ...
  - linear systems with linear measurements driven by ...

- Original Kalman filter solution applies only to ...
    - linear systems with linear measurements driven by ...
    - zero-mean, Gaussian white noise processes (for both process and measurement noise)

- Original Kalman filter solution applies only to ...
  - linear systems with linear measurements driven by ...
  - zero-mean, Gaussian white noise processes (for both process and measurement noise)
- Assume from now on that ...

- Original Kalman filter solution applies only to ...
  - linear systems with linear measurements driven by ...
  - zero-mean, Gaussian white noise processes (for both process and measurement noise)
- Assume from now on that ...
  - Our vector-valued function $f$ is a linear, vector-valued function, i.e., a matrix $\Phi$

- Original Kalman filter solution applies only to ...
  - linear systems with linear measurements driven by ...
  - zero-mean, Gaussian white noise processes (for both process and measurement noise)
- Assume from now on that ...
  - Our vector-valued function $f$ is a linear, vector-valued function, i.e., a matrix $\Phi$
  - $h$ is also a linear, vector-valued function, i.e. a matrix $H$

- Original Kalman filter solution applies only to ...
  - linear systems with linear measurements driven by ...
  - zero-mean, Gaussian white noise processes (for both process and measurement noise)
- Assume from now on that ...
  - Our vector-valued function $f$ is a linear, vector-valued function, i.e., a matrix $\Phi$
  - $h$ is also a linear, vector-valued function, i.e. a matrix $H$
- Thus, the problem is to find a gain matrix $K(k)$ such that

- Original Kalman filter solution applies only to ...
  - linear systems with linear measurements driven by ...
  - zero-mean, Gaussian white noise processes (for both process and measurement noise)
- Assume from now on that ...
  - Our vector-valued function $f$ is a linear, vector-valued function, i.e., a matrix $\Phi$
  - $h$ is also a linear, vector-valued function, i.e. a matrix $H$
- Thus, the problem is to find a gain matrix $K(k)$ such that

$$\hat{x}(k) = \Phi\hat{x}(k-1) + K(k)\,[\,z(k) - H\Phi\hat{x}(k-1)\,]$$

- Original Kalman filter solution applies only to ...
  - linear systems with linear measurements driven by ...
  - zero-mean, Gaussian white noise processes (for both process and measurement noise)
- Assume from now on that ...
  - Our vector-valued function $f$ is a linear, vector-valued function, i.e., a matrix $\Phi$
  - $h$ is also a linear, vector-valued function, i.e. a matrix $H$
- Thus, the problem is to find a gain matrix $K(k)$ such that

$$\hat{x}(k) = \Phi\hat{x}(k-1) + K(k)\,[\,z(k) - H\Phi\hat{x}(k-1)\,]$$

Given:

Given:

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

Given:

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
  - $x(k)$=state vector, $\Phi$=state transition matrix, $w(k)$=process noise vector

Given:

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
  - $x(k)$=state vector, $\Phi$=state transition matrix, $w(k)$=process noise vector
- Measurement Model: $z(k) = Hx(k) + v(k)$

Given:

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
  - $x(k)$=state vector, $\Phi$=state transition matrix, $w(k)$=process noise vector
- Measurement Model: $z(k) = Hx(k) + v(k)$
  - $z(k)$=measurement vector, $H$=meas. matrix, $v(k)$=meas. noise vector

Given:

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
  - $x(k)$=state vector, $\Phi$=state transition matrix, $w(k)$=process noise vector
- Measurement Model: $z(k) = Hx(k) + v(k)$
  - $z(k)$=measurement vector, $H$=meas. matrix, $v(k)$=meas. noise vector

Find:

Given:

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
  - $x(k)$=state vector, $\Phi$=state transition matrix, $w(k)$=process noise vector
- Measurement Model: $z(k) = Hx(k) + v(k)$
  - $z(k)$=measurement vector, $H$=meas. matrix, $v(k)$=meas. noise vector

Find:

- State estimate in the form
  $\hat{x}(k) = \Phi\hat{x}(k-1) + K(k)[z(k) - H\Phi\hat{x}(k-1)]$

Given:

- State Dynamics Model: $x(k) = \Phi x(k-1) + w(k)$
  - $x(k)$=state vector, $\Phi$=state transition matrix, $w(k)$=process noise vector
- Measurement Model: $z(k) = Hx(k) + v(k)$
  - $z(k)$=measurement vector, $H$=meas. matrix, $v(k)$=meas. noise vector

Find:

- State estimate in the form
  $\hat{x}(k) = \Phi\hat{x}(k-1) + K(k)[z(k) - H\Phi\hat{x}(k-1)]$

- The term $\Phi\hat{x}(k-1)$ is the extrapolated state estimate prior to making the measurement update or correction

- The term $\Phi\hat{x}(k-1)$ is the extrapolated state estimate prior to making the measurement update or correction
- To distinguish prior (extrapolated) estimate from updated estimate, we use the following notation:

- The term $\Phi\hat{x}(k-1)$ is the extrapolated state estimate prior to making the measurement update or correction
- To distinguish prior (extrapolated) estimate from updated estimate, we use the following notation:

$$\hat{x}^{-}(k) = \Phi\hat{x}^{+}(k-1) = \textbf{predicted estimate} \text{ prior to update}$$

- The term $\Phi \hat{x}(k-1)$ is the extrapolated state estimate prior to making the measurement update or correction
- To distinguish prior (extrapolated) estimate from updated estimate, we use the following notation:

$$\hat{x}^-(k) = \Phi \hat{x}^+(k-1) = \textbf{predicted estimate} \text{ prior to update}$$

$$\hat{x}^+(k) = \textbf{updated state estimate}$$

- The term $\Phi\hat{x}(k-1)$ is the extrapolated state estimate prior to making the measurement update or correction
- To distinguish prior (extrapolated) estimate from updated estimate, we use the following notation:

$$\hat{x}^-(k) = \Phi\hat{x}^+(k-1) = \textbf{predicted estimate} \text{ prior to update}$$

$$\hat{x}^+(k) = \textbf{updated state estimate}$$

- Applying $H$ to $\hat{x}^-(k)$ yields the **predicted measurement**

- The term $\Phi\hat{x}(k-1)$ is the extrapolated state estimate prior to making the measurement update or correction
- To distinguish prior (extrapolated) estimate from updated estimate, we use the following notation:

$$\hat{x}^{-}(k) = \Phi\hat{x}^{+}(k-1) = \textbf{predicted estimate} \text{ prior to update}$$

$$\hat{x}^{+}(k) = \textbf{updated state estimate}$$

- Applying $H$ to $\hat{x}^{-}(k)$ yields the **predicted measurement**

- Thus, the solution we seek now looks like this:

- Thus, the solution we seek now looks like this:

$$\hat{x}^{+}(k) = \hat{x}^{-}(k) + K(k)[z(k) - H\hat{x}^{-}(k)] \tag{1}$$

- Thus, the solution we seek now looks like this:

$$\hat{x}^{+}(k) = \hat{x}^{-}(k) + K(k)[z(k) - H\hat{x}^{-}(k)] \tag{1}$$

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

- Thus, the solution we seek now looks like this:

$$\hat{x}^{+}(k) = \hat{x}^{-}(k) + K(k)[z(k) - H\hat{x}^{-}(k)] \tag{1}$$

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

- This now looks very much like the running average problem!

- Thus, the solution we seek now looks like this:

$$\hat{x}^{+}(k) = \hat{x}^{-}(k) + K(k)[z(k) - H\hat{x}^{-}(k)] \quad (1)$$

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

- This now looks very much like the running average problem!

- Where are we?

- Where are we?
  - We've formulated the linear discrete problem

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain
  - But we haven't yet presented the complete solution

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain
  - But we haven't yet presented the complete solution
    - For the complete solution, we'll need to compute the gain $K(k)$ and use it to update both the state estimate **and** the estimation uncertainty

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain
  - But we haven't yet presented the complete solution
    - For the complete solution, we'll need to compute the gain $K(k)$ and use it to update both the state estimate **and** the estimation uncertainty
- What's next?

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain
  - But we haven't yet presented the complete solution
    - For the complete solution, we'll need to compute the gain $K(k)$ and use it to update both the state estimate **and** the estimation uncertainty
- What's next?
  - Drill down on state dynamics and determine how estimation uncertainty changes over time and also how it is updated

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain
  - But we haven't yet presented the complete solution
    - For the complete solution, we'll need to compute the gain $K(k)$ and use it to update both the state estimate **and** the estimation uncertainty
- What's next?
  - Drill down on state dynamics and determine how estimation uncertainty changes over time and also how it is updated
  - Use both the predicted estimation uncertainty and measurement noise uncertainty to calculate the Kalman gain

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain
  - But we haven't yet presented the complete solution
    - For the complete solution, we'll need to compute the gain $K(k)$ and use it to update both the state estimate **and** the estimation uncertainty
- What's next?
  - Drill down on state dynamics and determine how estimation uncertainty changes over time and also how it is updated
  - Use both the predicted estimation uncertainty and measurement noise uncertainty to calculate the Kalman gain
  - Use Kalman gain to update estimation uncertainty

- Where are we?
  - We've formulated the linear discrete problem
  - We've indicated the form of the solution: the recursive estimating algorithm with the Kalman gain
  - But we haven't yet presented the complete solution
    - For the complete solution, we'll need to compute the gain $K(k)$ and use it to update both the state estimate **and** the estimation uncertainty
- What's next?
  - Drill down on state dynamics and determine how estimation uncertainty changes over time and also how it is updated
  - Use both the predicted estimation uncertainty and measurement noise uncertainty to calculate the Kalman gain
  - Use Kalman gain to update estimation uncertainty