Kalman Filtering

EECS 126 at UC Berkeley

Spring 2022

1 Introduction

In this note, we will examine the **Kalman filter** (KF), an important application of LLSE used in fields such as control theory, signal processing, and econometrics. The Kalman filter is an algorithm that tracks an optimal estimate of the state of a stochastic dynamical system, given a sequence of noisy observations or measurements of the state over time.

As an algorithm, it is a *filter*, "filtering" out the effects of random noise; *recursive*, repeatedly calling itself in two phases called *prediction* and *update*; *online*, able to work with a stream of observations in real time; and *efficient*. The state estimate is linear, and optimal in the sense of minimizing the quadratic cost function of mean squared error, so the Kalman filter is also called the *linear-quadratic estimator* (LQE).

There are many generalizations of the Kalman filter, but we will start with a few simplifying assumptions. We will work in discrete time, and we will use time-homogeneous models (which do not change over time). We introduce the general vector formulation, though we will focus on the one-dimensional *scalar* case.

The Kalman filter can be quite notationally dense, so we will unravel and develop it in several sections.

a. State and observation: X_n, Y_n, A, C, V_n, W_n.

b. Control and feedback*: U_n, B, F.

c. Estimation and error: $\hat{X}_{n|n}$, $\hat{X}_{n|n-1}$, $\Sigma_{n|n}$, $\Sigma_{n|n-1}$, $\sigma^2_{n|n}$, $\sigma^2_{n|n-1}$.

d. Innovation and gain: $\mathbb{L}$, proj, span, $Y^{(1:n)}$, $\tilde{Y}_n$, K_n.

We will also find it useful to remind ourselves of a few key facts.

a. The variance of a matrix-vector product is $\mathsf{var}(AX) = \mathbb{E}((AX)(AX)^\mathsf{T}) = A\,\mathsf{var}(X)A^\mathsf{T}$.

b. The covariance of matrix-vector products is $\mathsf{cov}(AX, BY) = A\,\mathsf{cov}(X, Y)B^\mathsf{T}$, and covariance is bilinear.

c. The LLSE is an unbiased estimator: the *estimation residual* $X - \mathbb{L}(X \mid \cdot)$ is zero-mean.

d. The projection of X onto a zero-mean Y is $\mathsf{cov}(X, Y)\,\mathsf{var}(Y)^{-1}Y$, or $\mathbb{E}(XY^\mathsf{T})(\mathbb{E}(YY^\mathsf{T}))^{-1}Y$ in general.

2 State and observation

The **states** of the dynamical system are the sequence of random variables $(X_n)_{n\in\mathbb{N}}$, which are in general random vectors in $\mathbb{R}^d$, $d \geq 1$. The initial state X_0 is usually given, but the true values of every other state are not directly known to the algorithm (which is why we need estimation in the first place).

The **dynamics** or **transition model** is a scalar A, or in general a matrix $A \in \mathbb{R}^{d\times d}$, that describes how the states evolve over time. The transition model is very often the matrix form of some physical model, such as $x = x_0 + vt$ and $v = v_0 + at$, which is why it is also called the *dynamics*.

The **process noise** $(V_n)_{n\geq 1}$ are random variables of the same dimensions as X_n. The noise is assumed to be zero-mean, Gaussian, with common variance σ_V^2 (or covariance matrix $\Sigma_V \in \mathbb{R}^{d\times d}$). That is, $V_n \sim \mathcal{N}(0, \sigma_V^2)$. The noise is also independent of $(X_n)_{n\in\mathbb{N}}$, mutually independent, and unknown to the algorithm.

The state-transition equation is given below.

$$X_n = AX_{n-1} + V_n, \quad n \geq 1.$$

The **observations** or **measurements** $(Y_n)_{n\geq 1}$ are random variables that *are* available to the algorithm. They do *not* necessarily have the same dimensions as the X_n; they can be random scalars, or generally random vectors in $\mathbb{R}^e$.

The **observation model** C, like the transition model A, is a scalar or matrix $C \in \mathbb{R}^{e\times d}$. However, unlike A, C does not have to be square in general (so it may not be invertible). For example, if X_n are random vectors in $\mathbb{R}^2$ and $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$, then only the first entry of each X_n can be observed.

The **observation noise** or **measurement noise** $(W_n)_{n\geq 1}$ are defined analogously to the process noise. They are random scalars, or random vectors in $\mathbb{R}^e$, independent and identically distributed as $\mathcal{N}(0, \sigma_W^2)$. Additionally, the W_n are independent from the X_n, V_n, *and* Y_n.

The state-observation equation is given below.

$$Y_n = CX_n + W_n, \quad n \geq 1.$$

Graphically, we can summarize the states and observations by the following diagram. The underlying states and noise terms are hidden to the algorithm, and only the observations are directly accessible.

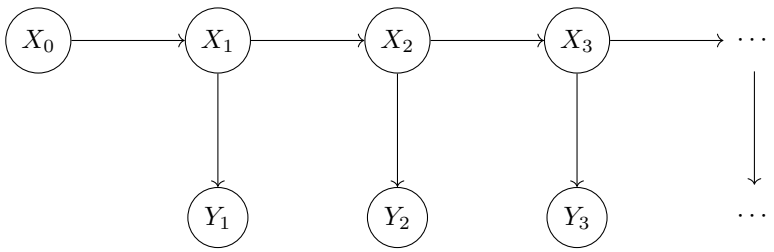

Figure 1: The states and observations may remind you of the *hidden Markov model*.

Some remarks on the modelling assumptions.

- The states very commonly describe position, velocity, and/or acceleration, for use in tracking. The entries of the observations, often taken from sensors, do not have to match those of the state — for example, we may be able to measure only position, or have multiple sensors measuring velocity.
- If we wanted to describe a temporally inhomogeneous system, we could use any of the terms A_n, C_n, Σ_{V_n}, and Σ_{W_n} as necessary.
- The filter assumes that the underlying system is a *linear* stochastic dynamical system, but in real systems, nonlinear dynamics that are incorporated into the random process noise instead of the model can greatly worsen the performance of the filter.
- The process noise and observation noise are both assumed to be zero-mean — the noise is not biased in any direction. But, we can also deal with any offset or *drift*, i.e. general noise distributed as $\mathcal{N}(\mu_V, \Sigma_V)$, by adding more components to the state vector. We will take zero-mean noise for simplicity.
- The specific distribution of the noise is a mostly arbitrary choice, because only the variance of the noise is used in the algorithm. (Though when X_0 is constant or Gaussian, the Gaussian distribution of the noise has the advantage that the Kalman filter is the same as the MMSE.)

We will make one final assumption without loss of generality: $C = 1$ in the scalar case. If $C = 0$, then the observation $Y_n = W_n$ is pure independent, random noise, so we do not consider this case. Otherwise, we can simply take the rescaled observations $Y'_n = Y_n/C = X_n + W'_n$, with noise $W'_n \sim \mathcal{N}(0, \sigma_W^2/C^2)$.

Exercise 1. Write X_n in closed form in terms of A, X_0, and $(V_k)_{k=1}^n$. Being able to use simple induction, and expand one-step recurrence relations, will be valuable in the following sections.

Exercise 2. Find the expectation of X_n in terms of A and $\mathbb{E}(X_0)$.

Exercise 3. Find the variance of X_n in terms of A, $\mathsf{var}(X_0)$, and Σ_V. If the scalar transition model satisfies $|A| < 1$, i.e. the system is *stable*, find $\lim_{n\to\infty} \mathsf{var}(X_n)$.

Exercise 4. Find Y_n in closed form in terms of X_0, A, $(V_k)_{k=1}^n$, C, and W_n.

3 Control and feedback*

We can also consider *controlling* a given system. The more general state-transition equation also includes the **control inputs** $(U_n)_{n\geq 1}$ and the **control-input model** B:

$$X_n = AX_{n-1} + BU_n + W_n, \quad n \geq 1.$$

For us, this more general model is out of scope. We will only hint at two types of control: *open-loop* control, in which the control input U_n is independent of the state X_{n-1}, and *closed-loop* (**feedback**) control, in which U_n is some function of X_{n-1}. When the input "incorporates feedback" from the previous state, the new transition model $X_n = (A + BF)X_{n-1} + W_n$ can be self-correcting or self-stabilizing.

4 Estimation and error

The goal of the Kalman filter is to find the optimal **estimate** $\hat{X}_n$ of the state X_n at every time step $n \geq 1$, given the *history* or *trajectory* of observations $Y^{(1:n)} = (Y_1, \ldots, Y_n)$ up to time n:

$$\hat{X}_n = \hat{X}_{n|n} := \underset{f(Y^{(1:n)}) \text{ affine}}{\operatorname{argmin}} \mathbb{E}\left(\left\| X_n - f(Y^{(1:n)}) \right\|^2 \right).$$

Remark. The norm of a scalar is itself. In general, $\mathbb{E}(\|X\|^2) = \mathbb{E}(X^\mathsf{T} X)$ is the objective function, because $\mathbb{E}(X^\mathsf{T} Y)$ is the usual inner product for the Hilbert space of random *vectors*. What does the affine function $f(Y^{(1:n)})$ look like for scalar $Y^{(1:n)}$? What about $Y^{(1:n)}$ with different dimensions than X_n?

We know that $\hat{X}_{n|n} = \mathbb{L}(X_n \mid Y^{(1:n)})$ is the LLSE. There is one small problem: the algorithm does not directly know X_n. Instead, we will need to find $\hat{X}_{n|n}$ from $\hat{X}_{n-1|n-1}$, A, C, and other available quantities. Before we do so, we introduce a few more relevant terms.

The Kalman filter can be used not only for estimation and tracking, but also prediction and forecasting. The **prediction** of the state X_n at time step n, given the history of observations up to time $k \leq n$, is

$$\hat{X}_{n|k} := \mathbb{L}(X_n \mid Y^{(1:k)}).$$

By the recurrence relation given by the transition model, we can find

$$\hat{X}_{n|k} = \mathbb{L}\left(A^{n-k} X_k + \sum_{i=k}^{n} A^{n-i} V_i \;\middle|\; Y^{(1:k)} \right) = A^{n-k}\, \mathbb{L}(X_k \mid Y^{(1:k)}) = A^{n-k} \hat{X}_{k|k}.$$

Notice that the process noise, which is independent and zero-mean, disappears from the prediction. In other words, to find the prediction of X_n given observations up to time k, we "advance the state model by $n-k$ steps" starting from $\hat{X}_{k|k}$, the best known estimator.

An important prediction is the one-step prediction, or simply *prediction* $\hat{X}_{n|n-1} = A\hat{X}_{n-1|n-1}$. We highlight it here, as it will be used quite often later.

The **estimation variance** is $\sigma^2_{n|n}$ or $\Sigma_{n|n} := \mathsf{var}(X_n - \hat{X}_{n|n})$, the variance of the estimation residual. Likewise, the *prediction variance* is $\sigma^2_{n|k}$ or $\Sigma_{n|k} := \mathsf{var}(X_n - \hat{X}_{n|k})$. Because the residual $X_n - \hat{X}_{n|k}$ is zero-mean,

$$\Sigma_{n|k} = \mathbb{E}\left((X_n - \hat{X}_{n|k})(X_n - \hat{X}_{n|k})^\mathsf{T} \right), \quad k \leq n.$$

Finally, the **estimation error** is $\mathbb{E}(\|X_n - \hat{X}_{n|n}\|^2)$, and the *prediction error* is $\mathbb{E}(\|X_n - \hat{X}_{n|k}\|^2)$. The estimation error is precisely the quadratic cost function, evaluated at its minimizer $\hat{X}_{n|n}$. In the scalar case, it is the *same* as the estimation variance; in general, the two are related by $\mathbb{E}(\|X_n - \hat{X}_{n|n}\|^2) = \operatorname{tr}(\Sigma_{n|n})$.

Exercise 5. Show that $\mathbb{E}(\|X_n - \hat{X}_{n|n}\|^2) = \operatorname{tr}(\Sigma_{n|n})$. *Hint*: trace is linear and has the *cyclic property*.

5 Innovation and gain

$\hat{X}_{n|n}$ is the LLSE of X_n given $Y^{(1:n)}$, which is also the projection of X_n onto the span of $\{1, Y_1, \ldots, Y_n\}$.

$$\hat{X}_{n|n} = \mathbb{L}(X_n \mid Y^{(1:n)}) = \text{proj}_{\text{span}\{1, Y_1, \ldots, Y_n\}}(X_n)$$

We might recall that projections onto an orthogonal basis are desirable, because they decompose nicely into a sum of projections onto the individual components. Moreover, every spanning set can be turned into an orthogonal basis via the Gram-Schmidt procedure.

$$= \text{proj}_{\text{span}\left\{1, \tilde{Y}_1, \ldots, \tilde{Y}_n\right\}}(X_n)$$

$$= \text{proj}_{\text{span}\left\{1, \tilde{Y}_1, \ldots, \tilde{Y}_{n-1}\right\}}(X_n) + \text{proj}_{\text{span}\left\{\tilde{Y}_n\right\}}(X_n)$$

Splitting the subspace into the two parts of $\text{span}\{1, Y_1, \ldots, Y_{n-1}\} \oplus \text{span}\{Y_n\}$ is connected to the Kalman filter being both recursive and online. We can find the first part recursively using the previous estimate $\hat{X}_{n-1|n-1}$, and we can find the second part online, as the new observation Y_n arrives.

$$= \mathbb{L}(X_n \mid Y^{(1:n-1)}) + \text{proj}_{\text{span}\left\{\tilde{Y}_n\right\}}(X_n)$$

$$= A\hat{X}_{n-1|n-1} + K_n\tilde{Y}_n.$$

The projection of X_n onto the span of $\tilde{Y}_n$ must be some linear function of $\tilde{Y}_n$, and we denote the linear transformation as K_n. The random variable or random vector $\tilde{Y}_n$ is the **innovation** at time n, and the scalar or matrix K_n is the Kalman **gain** at time n.

The innovation can be found from Gram-Schmidt, and will be orthogonal to $\hat{X}_{n|n-1} \in \text{span}\left\{Y^{(1:n-1)}\right\}$:

$$\begin{aligned}
\tilde{Y}_n &= Y_n - \mathbb{L}(Y_n \mid Y_1, \ldots, Y_{n-1}) \\
&= Y_n - \mathbb{L}(CX_n + W_n \mid Y^{(1:n-1)}) \\
&= Y_n - CA\hat{X}_{n-1|n-1}.
\end{aligned}$$

The Kalman gain can be found from the zero-mean projection formula, $\text{proj}_Y(X) = \mathbb{E}(XY^\mathsf{T})\left[\mathbb{E}(YY^\mathsf{T})^{-1}\right]Y$:

$$\begin{aligned}
K_n &= \mathsf{cov}(X_n, \tilde{Y}_n)\,\mathsf{var}(\tilde{Y}_n)^{-1} \\
&= \mathsf{cov}(X_n - \hat{X}_{n|n-1}, C(X_n - \hat{X}_{n|n-1}) + W_n)\,\mathsf{var}(\tilde{Y}_n)^{-1} \\
&= \Sigma_{n|n-1}C^\mathsf{T}\left[(C\Sigma_{n|n-1}C^\mathsf{T} + \Sigma_W)^{-1}\right]
\end{aligned}$$

In summary, the estimate of the true state at time n is a weighted average of the prediction from the previous estimate at time $n-1$ and the new observation at time n.

"The optimal estimate of X_n lies between the past prediction and the present observation."

$$\boxed{\hat{X}_{n|n} = A\hat{X}_{n-1|n-1} + K_n\tilde{Y}_n = (I - K_nC)A\hat{X}_{n-1|n-1} + K_nY_n}.$$

Exercise 6. Write $\hat{X}_{n|n}$ in closed form in terms of $\hat{X}_{0|0} := X_0$, Y_k, K_k, and $(I - K_k C)A$, for $k = 1, \ldots, n$.

Exercise 7. Prove the **orthogonal update** for LLSE: if X_n is zero-mean, then $\hat{X}_{n|n} = \hat{X}_{n|n-1} + \mathbb{L}(X_n \mid \tilde{Y}_n)$.

Exercise 8. Find $\mathbb{E}(\hat{X}_{n|n})$ in terms of A and $\mathbb{E}(X_0)$. *Hint*: what is the expectation of any $\mathbb{L}(X \mid \cdot)$?

6 Prediction and update

Now we can describe the implementation of the actual algorithm. The Kalman filter is often carried out in two phases: **prediction** and **update**, also called *propagation* and *correction*.

Throughout, we keep track of the state estimate $\hat{X}_{n|n}$ and the estimation variance $\Sigma_{n|n}$. We begin by initializing $\hat{X}_{0|0} \leftarrow X_0$ and $\Sigma_{0|0} \leftarrow \mathsf{var}(X_0)$ at time 0.

In the prediction phase after time step $n-1$, we have access to $(\hat{X}_{n-1|n-1}, \Sigma_{n-1|n-1})$. We can find the *predicted* or *a priori* state estimate and estimation variance:

$$\begin{aligned}
\hat{X}_{n|n-1} &\leftarrow A\hat{X}_{n-1|n-1} \\
\Sigma_{n|n-1} &= \mathsf{var}(X_n - A\hat{X}_{n-1|n-1}) \\
&= \mathsf{var}(A(X_{n-1} - \hat{X}_{n-1|n-1}) + V_n) \\
&\leftarrow A\Sigma_{n-1|n-1}A^{\mathsf{T}} + \Sigma_V.
\end{aligned}$$

Interestingly, we can already find the Kalman gain here:

$$K_n \leftarrow \Sigma_{n|n-1}C^{\mathsf{T}}\left[(C\Sigma_{n|n-1}C^{\mathsf{T}} + \Sigma_W)^{-1}\right].$$

In the update phase at time step n, we know $(\hat{X}_{n|n-1}, \Sigma_{n|n-1})$, and the new observation $Y_n = CX_n + W_n$ becomes available. We can then find the innovation, and the *a posteriori* state estimate and estimate variance:

$$\begin{aligned}
\tilde{Y}_n &\leftarrow Y_n - C\hat{X}_{n|n-1} \\
\hat{X}_{n|n} &\leftarrow \hat{X}_{n|n-1} + K_n\tilde{Y}_n \\
\Sigma_{n|n} &= \mathsf{var}(X_n - [(I - K_nC)\hat{X}_{n|n-1} + K_nY_n]) \\
&= \mathsf{var}((I - K_nC)(X_n - \hat{X}_{n|n-1}) + K_nW_n) \\
&= (I - K_nC)\Sigma_{n|n-1}(I - K_nC)^{\mathsf{T}} + K_n\Sigma_W K_n^{\mathsf{T}} \\
&\overset{\cdots}{\leftarrow} (I - K_nC)\Sigma_{n|n-1}.
\end{aligned}$$

The prediction and update phases typically alternate, but they do not have to: we can predict several steps in advance without incorporating any new observations, or we can update several times in sequence to account for multiple newly available observations.

7 Remarks and interpretations*

Innovation. $\tilde{Y}_n$ represents the "new information" given by the observation at time n, found by removing any redundant information (some linear combination of the known $Y_1, \ldots, Y_{n-1}$) from Y_n. The innovation, possibly zero, is orthogonal to all of the previous observations.

Kalman gain. K_n is "how much we can gain from the new observation"; "how much we distrust the previous estimate"; the "learning rate," which satisfies $0 \leq K_n \leq 1$ in the scalar case:

$$K_n = \frac{\sigma_{n|n-1}^2}{\sigma_{n|n-1}^2 + \sigma_W^2}.$$

The scalar gain can be thought of as the proportion of total variability ($\mathsf{var}(\tilde{Y}_n) = \sigma_{n|n-1}^2 + \sigma_W^2$) that is due to the prediction ($\sigma_{n|n-1}^2$), not due to the observation noise (σ_W^2).

In $\hat{X}_{n|n} = (1 - K_n)\hat{X}_{n|n-1} + K_n Y_n$, if $K_n \sim 1$, then $\hat{X}_{n|n} \sim Y_n$, the new observation; if $K_n \sim 0$, then $\hat{X}_{n|n} \sim \hat{X}_{n|n-1}$, the previous prediction. We can also *tune* the value of K_n manually, often having a higher gain earlier to "learn more quickly," and gradually lowering the gain to "converge" to a model.

The Kalman gain derived above is *optimal*, i.e. minimizes estimation error. The final simplification of $\Sigma_{n|n}$ is correct only for the optimal gain; the unsimplified version is the more generally correct *Joseph form*.

Derivation. The intermediate quantities of X_{n-1} and X_n only appear to justify the final Kalman filter equations, because the algorithm only knows A, Σ_V, $\hat{X}_{n-1|n-1}$, and $\Sigma_{n-1|n-1}$ before the prediction phase, or C, Σ_W, $\hat{X}_{n|n-1}$, and $\Sigma_{n|n-1}$ before the update phase.

Efficiency. Interestingly, the Kalman gain K_n, estimation variance $\Sigma_{n|n}$, and prediction variance $\Sigma_{n|n-1}$ can all be recursively computed *offline* and stored. The computation of $\hat{X}_{n|n}$ is the only online operation! Also, the Kalman filter is space-efficient — it does not need to store any past estimates or observations!

8 A geometric derivation

We can also derive the scalar Kalman filter equations by leveraging the geometry of the Hilbert space. Keep in mind these are only visualizations of elements and subspaces in an infinite-dimensional space.

Make sure you know which random variables are orthogonal to each other — a "triangle" might have more than one right angle! First, let us see what the length of the only term involving K_n is equal to:

$$K_n \left\| \tilde{Y}_n \right\| = \left\| \hat{X}_{n|n} - \hat{X}_{n|n-1} \right\| = K_n \left\| Y_n - \hat{X}_{n|n-1} \right\|.$$

Now, we can leverage the similarity of the two triangles formed by $(\hat{X}_{n|n-1}, \hat{X}_{n|n}, X_n)$ [the smaller one] and $(\hat{X}_{n|n-1}, X_n, Y_n)$ [the larger one]:

$$K_n = \frac{\left\| \hat{X}_{n|n} - \hat{X}_{n|n-1} \right\|}{\left\| X_n - \hat{X}_{n|n-1} \right\|} \frac{\left\| X_n - \hat{X}_{n|n-1} \right\|}{\left\| Y_n - \hat{X}_{n|n-1} \right\|} = \left(\frac{\left\| X_n - \hat{X}_{n|n-1} \right\|}{\left\| Y_n - \hat{X}_{n|n-1} \right\|} \right)^2 = \frac{\sigma_{n|n-1}^2}{\sigma_{n|n-1}^2 + \sigma_W^2}.$$

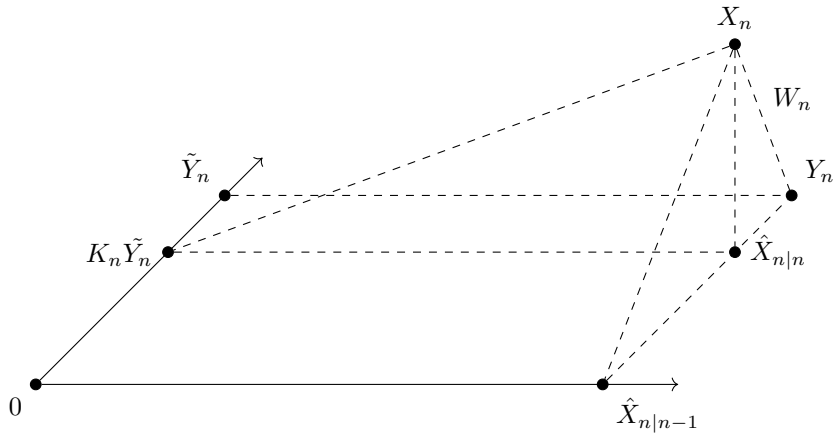

Figure 2: $\hat{X}_{n|n}$ is the orthogonal projection of X_n onto $\operatorname{span}\{1, Y_1, \ldots, Y_{n-1}\} \oplus \operatorname{span}\{\tilde{Y}_n\}$.

We can also find $\sigma^2_{n|n} = \|X_n - \hat{X}_{n|n}\|^2$ by applying the Pythagorean theorem to the smaller triangle:

$$\sigma^2_{n|n} = \left\|X_n - \hat{X}_{n|n-1}\right\|^2 \left(1 - \frac{\left\|\hat{X}_{n|n} - \hat{X}_{n|n-1}\right\|^2}{\left\|X_n - \hat{X}_{n|n-1}\right\|^2}\right) = \sigma^2_{n|n-1}(1 - K_n).$$

To find $\sigma^2_{n|n-1}$, we need to draw X_{n-1}, which possibly introduces a new dimension. The following diagram is slightly "rotated" from above, if we pay attention to the "vertical" projection in both.

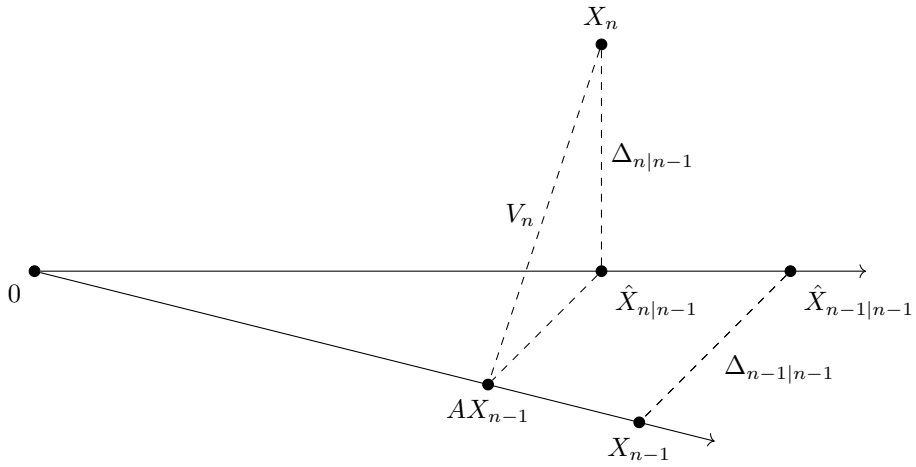

Figure 3: The difference between AX_{n-1} and the prediction $\hat{X}_{n|n-1}$ is orthogonal to the noise V_n.

Applying the Pythagorean theorem to the triangle formed by $(AX_{n-1}, \hat{X}_{n|n-1}, X_n)$,

$$\sigma^2_{n|n-1} = \left\|\Delta_{n|n-1}\right\|^2 = \left\|AX_{n-1} - A\hat{X}_{n-1|n-1}\right\|^2 + \|V_n\|^2.$$

Finally, we can leverage the similarity of triangles once more:

$$\sigma^2_{n|n-1} = A^2\sigma^2_{n-1|n-1} + \sigma^2_V.$$

9 Summary

For time steps $n \geq 1$, the states and observations are given by the following equations.

$$X_n = AX_{n-1} + V_n$$

$$Y_n = CX_n + W_n.$$

At time $n = 0$, we initialize the estimate and estimation variance as $(\hat{X}_{0|0}, \Sigma_{0|0}) \leftarrow (X_0, \mathsf{var}(X_0))$. We can compute the Kalman gain and estimation variances offline using the following equations.

$$\Sigma_{n|n-1} = A\Sigma_{n-1|n-1}A^\mathsf{T} + \Sigma_V \qquad \text{(prediction)}$$

$$K_n = \Sigma_{n|n-1}C^\mathsf{T}\left[(C\Sigma_{n|n-1}C^\mathsf{T} + \Sigma_W)^{-1}\right] \qquad \text{(gain)}$$

$$\Sigma_{n|n} = (I - K_nC)\Sigma_{n|n-1} \qquad \text{(update)}$$

We can compute the state estimates online as new observations arrive.

$$\hat{X}_{n|n-1} = A\hat{X}_{n-1|n-1} \qquad \text{(prediction)}$$

$$\tilde{Y}_n = Y_n - C\hat{X}_{n|n-1} \qquad \text{(innovation)}$$

$$\hat{X}_{n|n} = \hat{X}_{n|n-1} + K_n\tilde{Y}_n \qquad \text{(update)}$$

In the scalar case, the observation model is $C = 1$, the prediction variance is $\sigma^2_{n|n-1} = A^2\sigma^2_{n-1|n-1} + \sigma^2_V$, the Kalman gain is $K_n = \sigma^2_{n|n-1}/(\sigma^2_{n|n-1} + \sigma^2_W)$, etc.

We hope you had a fun semester in EECS 126! Good luck as you prepare for finals. :)

■