Revised The discrete Kalman Filter ( Ch.3.4 )

%%% Kim’s comment

In the textbook there is a sign error as

%%%

1. Problem

Given the measurement data as

Find the conditional estimator:

1. Solution
   1. **Solution in the text –two states**
2. Prediction (a priori)

* The propagating conditional mean
* The variance of the propagating

1. Correction ( a posteriori)

* The updating conditional mean

* The updating the covariance

%%% **In the textbook (3.44) should be changed in sign** %%%

* 1. **The solution without the inverse of (wiki): Three States**

1. Prediction ( a priori)

* The propagating conditional mean
* The variance of the propagating

1. Innovation

\*\*Kalman gain (at the top of page 98)

1. Correction (a posteriori)

The updating conditional mean

The updating the covariance

%%% Kim’s comment

* Here the kalman gain is more robust numerically, i.e,

In the left hand side, if is small, or is not definite, then the inverse is unstable numerically. However the right hand, even if is small, is more stable.

How to verify it? In the exercise 7, in chapter 3, using the matrix identity as

We may get

* The variance of the conditional expectation in (3.49) in the text book

**should be equivalent to**

Is it strange? I think so. The proof is in Chapter 4 or wiki. %%%

1. To initiate the algorithm, we need the initial conditions. Let us assume are constant, i.e., time-invariant system

To proceed the kalman filter, there are two arbitrary initial values. One is the estimator and the covariance of the estimator,

Let us consider if the measurement is missed, i.e., no measurement date is obtained, in this case, we may consider the intensity of the noise variance to be infinite, so that

In fact, the initial estimator, in general may be assumed to be zero, or as you want any value.

1. Some comments
2. There are several algorithms to get Kalman gain{see the bottom} So which one is good? (“Kalman and Bayesian Filters in Python, 2020”)

One of the criteria, which is most important, is the numerical stable. Up to now, in my knowledge, in wiki it is the best.

1. In the problem, we may only consider the Gaussian cases, so **the matters to be important are to find mean and variance.**
2. In Kalman filter, at every time step, the kalman gain is needed. In fact you may see the algorithm**, the kalman gain may be calculated in advance**, which mean, if the noise intensities constant, to get **the kalman gain you do not need the measurement**!
3. In the reality, we may not know the intensity of the noise in the process and measurement. In general, it depends on the your heuristic experiences. Or in the scalar case, you may guess the effects on the estimator considering the relative magnitude of two noise intensity
4. Here, we may have a question. Is the kalman gain calculated at any case? One of the necessary condition is the invertibility of the noise intensity,i.e., **every sensor has to be noise!**
5. If the measurement is missed at , how does the kalman procedure continue? As in the first measurement, we may assume so that

%%% Kalman and Bayesian Filters in Python, 2020 %%%%%%%%%





