ECE531 Screencast 7.6: Kalman Filter Simulation Example

D. Richard Brown III

Worcester Polytechnic Institute

Problem Setup

The first step in working with a Kalman filter is to correctly set up the dynamical model. For purposes of illustration, we will specify a simple linear time-invariant model here.

Suppose we have one-dimensional motion with $T = 0.1$

$$X[n+1] = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix} X[n] + \begin{bmatrix} 0 \\ 0.1 \end{bmatrix} U[n]$$

$$Y[n] = \begin{bmatrix} 1 & 0 \end{bmatrix} X[n] + V[n]$$

and with noise covariances $\mathrm{E}[V^2[n]] = R = 0.01$ and $\mathrm{E}[U[n]U^\top[n]] = Q = 1$.

Generate States and Observations

To simulate a Kalman filter, we can generate states and observations from the state-update and observation equations. For our example, in Matlab, this might look something like

```
% declare variables F, G, H, Q, R, N, MU0, SIGMA0
% (omitted)

% generate states and observations
X = zeros(2,N);
X(:,1) = mvnrnd(MU0,SIGMA0)'; % initial state X[0]
for n=1:N-1,
    U = sqrt(Q)*randn;
    X(:,n+1) = F*X(:,n)+G*U; % update state
end
V = sqrt(R)*randn(1,N);     % observation noise
Y = H*X+V;                  % generate observations
```

Initialize the Kalman Filter

Using the known prior on $X[0]$, you should initialize $\hat{X}[0|-1]$ and $\Sigma[0|-1]$. You should also pre-allocate space for any variables you wish to save.

Some suitable Matlab code:

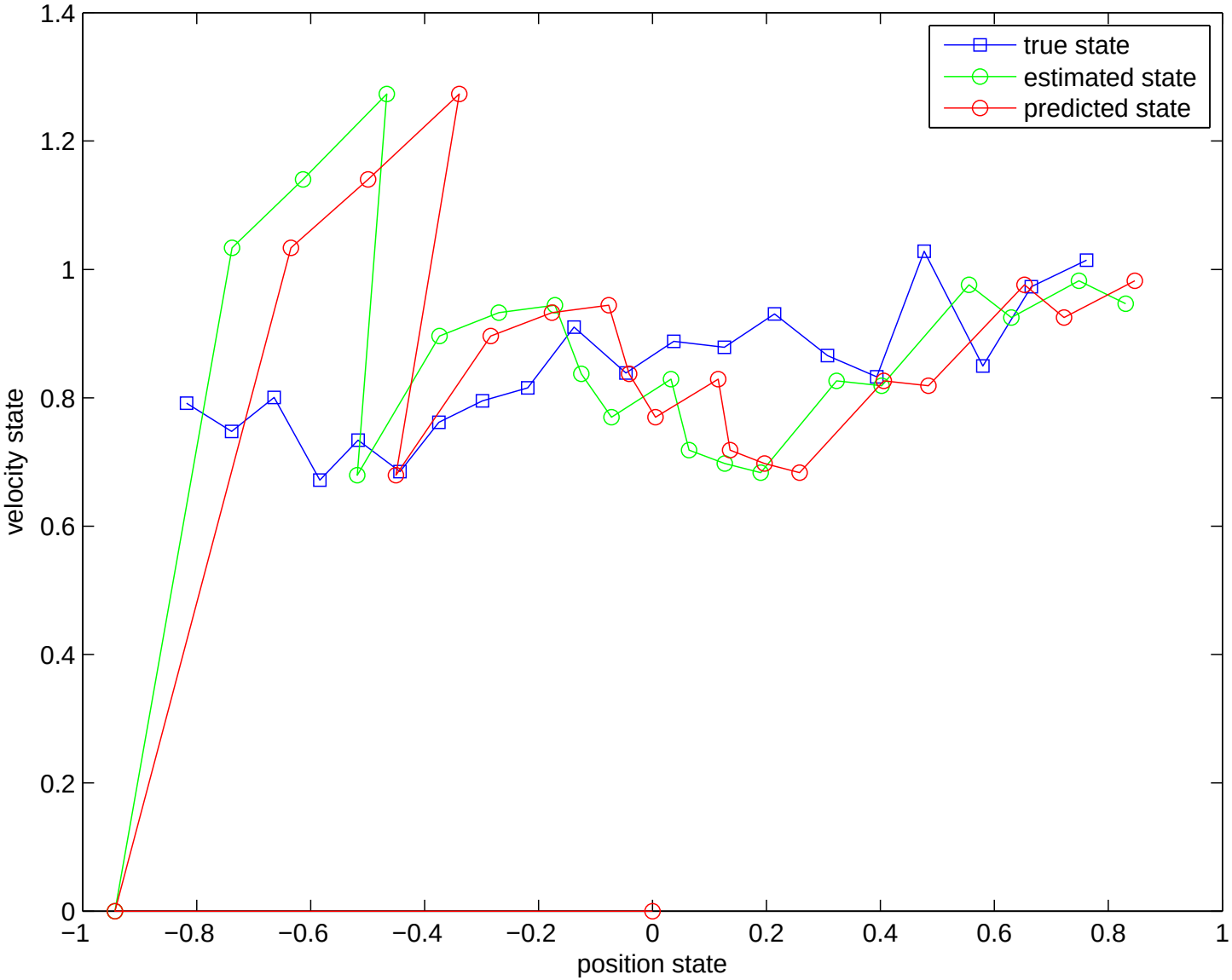
```
% initialize (state prediction prior to observations)
Xhat_kplus1_k(:,1) = MU0;
Sigma_kplus1_k = SIGMA0;

% allocate space for variables
Xhat_k_k = zeros(2,N);
Xhat_kplus1_k = zeros(2,N);
```

Run the Kalman Filter

This code just executes the Kalman filter recursion as derived.

```
for n=1:N,
    K = Sigma_kplus1_k*H'*inv(H*Sigma_kplus1_k*H'+R);
    Xhat_k_k(:,n) = Xhat_kplus1_k(:,n) + K*(Y(n)-H*
        Xhat_kplus1_k(:,n));
    Sigma_k_k = Sigma_kplus1_k-K*H*Sigma_kplus1_k;
    Sigma_kplus1_k = F*Sigma_k_k*F'+G*Q*G';
    if n<N,
        Xhat_kplus1_k(:,n+1) = F*Xhat_k_k(:,n);
    end
end
```

Sample Results for One Run



MSE Averaged Over 5000 Runs