# Data filtering and optimal recursive prediction of a discrete, stochastic alien invasion

Will Rafey

December 11, 2012

Our natural and social worlds are cluttered with stochastic processes – like stages in evolutionary, economic, environmental or social transformation – which might be advantageous to predict [1]. It is mathematically convenient to assume these processes are discrete, linear and entirely dependent on prior conditions: analytical solutions are extremely difficult with any less restrictive assumptions [2]. This paper reviews the optimal data filtering and recursive estimation of these linear processes in a Hilbert space of random vectors (section 1), summarizing Luenberger's resulting solution to minimum-variance estimation (section 2), explicating Kalman's canonical random-process model (section 3), and applying the results to filter military data and estimate a UFO's future location during an alien invasion (section 4).

## 1 Probability in a Hilbert space (Luenberger)

First, we will introduce the notation that will allow us to define discrete stochastic processes with precision. From introductory probability, a random variable $x_i$ is defined by the cumulative probability distribution $F(k) = \Pr(x_i \leq k)$ with the usual properties:

- $0 \leq F(k) \leq 1$ for all $k$
- $\int_{-\infty}^{\infty} dF(k)dk = 1$
- $E[g(x)] = \int_{-\infty}^{\infty} g(k)dF(k)$ denotes the expected value of $g(x)$.

We can extend this concept of a random variable to $n$ dimensions by defining a random vector $x = \{x_1, \dots x_n\}$ composed of $n$ random variables.

Next, it is especially useful to introduce notation consistent with Luenberger's *Optimization by Vector Space Methods* [4], in order to apply his general geometric approach to $n$-dimensional optimization to stochastic problems. This allows us to define the corresponding Hilbert space (§4.2, p. 81), which is a normed space with an inner product defined analogously to the dot product.

**Definition 1.** *(Luenberger, p. 81) A* ***Hilbert space of random vectors*** *contains all linear combinations of all random vectors* $\{y_1, \ldots y_m\}$ *with bounded variance (i.e.,* $E(y_{ij}^2) < \infty$ *for all* $i = 1, \ldots m$ *and* $j = 1, \ldots n$*). Formally, any element in this space may be written as*

$$y = \sum_{k=1}^{m} K_i y_i$$

*where* $K_i \in mat(n, n, \mathbb{R})$*, which means that the space is at most dimension* $m$ *(less where one or more of the* $y_i$ *are linearly independent).*

Again, following Luenberger, the inner product of this Hilbert space is the expectation of the dot product of any two random vectors in the space (§4.2, p. 81); i.e.,

$$(x|z) = E\left(\sum_{i=1}^{n} x_i z_i\right).$$

Note that the expectation operator satisfies the four axioms given by Luenberger (§3.2, p. 46-47): scalar multiplication is commutative, so $(x|z) = (z|x)$; addition is linear, so $(x+y|z) = (x|z)+(y|z)$; scalar multiplication is associative, so $(\lambda x|z) = \lambda(x|z)$; and clearly $(x|x) \geq 0$ and $(x|x) = 0$ if and only if $x = \theta$. This lets us construct a normed vector space, where, the norm generates an $n$-dimensional vector.

**Definition 2.** *(Luenberger, p. 81) The squared* ***norm*** *of any random vector* $x = (x_1, \ldots x_n)$ *in a Hilbert space of random vectors is*

$$||x||^2 = (x|x) = Tr\{E(xx^T)\}$$

*the trace of the* $n \times n$ *matrix of the expected values of the product of* $x$ *and its transpose* $x^T$*.*

In sum, our notation gives us three increasingly reduced representations of the random vector $x$:

1. The $n \times n$ covariance matrix,

$$E(xx^T) = \begin{bmatrix} E(x_1 x_1^T) & \ldots & E(x_n x_1^T) \\ \vdots & \ddots & \vdots \\ E(x_1 x_n^T) & \ldots & E(x_n x_n^T) \end{bmatrix}$$

2. The $n-$dimensional trace of the covariance matrix, the squared norm, $||x||^2$.
3. The scalar expectation of the squared norm, $E[||x||^2] = \sum_{i=1}^{n} E[x_i^2]$.

## 2 General minimum-variance estimation (Luenberger)

With notation from section one in hand, we can study the first part of the discrete stochastic process model, which assumes that our vector $y$ can be expressed completely with a "true" linear combination of the orthogonal random vectors $\beta = \{\beta_1, \dots \beta_n\}$, where $n < m$ (where $n = m$, the system of equations is completely determined and there is no need for a best guess). If we collect a $n \times m$ matrix of data $W$ (Luenberger, §4.5, p. 86), then our imperfect data observations are given in the form

$$y = W\beta + \epsilon \tag{1}$$

with noise from an additive error term $\epsilon$ with mean zero, i.e., $E(\epsilon) = \theta$, and positive-definite covariance matrix $E(\epsilon\epsilon^T) = Q$. Our optimal estimate for $\beta$, which we denote $\hat{\beta}$, is the solution to the minimization problem,

$$\min_{\hat{\beta}} E[||\hat{\beta} - \beta||^2], \tag{2}$$

which minimizes the expected value of the $n$-dimensional norm of the difference between our guess $\hat{\beta}$ and the "true" $\beta$ (which, again, exists only due to our construction of a linear model). This is the Hilbert normed space representation of a minimum-variance problem. Luenberger presents an elegant solution as follows (§4.5, Theorem 1, p. 87):

**Theorem 1.** *For $y$ and $\beta$ random vectors, with invertible $E(yy^T)$, the linear (in terms of $y$) estimate $\hat{\beta}$ of $\beta$ that minimizes $E[||\hat{\beta} - \beta||^2]$ is*

$$\hat{\beta} = E(\beta y^T)[E(yy^T)]^{-1}y \tag{3}$$

*with an error covariance matrix*

$$E[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^T] = E(\beta\beta^T) - E(\hat{\beta}\hat{\beta}^T) = E(\beta\beta^T) - E(\beta y')[E(yy^T)]^{-1}E(y\beta^T). \tag{4}$$

*Proof.* (Luenberger, p. 87-88). Finding the best $\hat{\beta}$ in the subspace $[y]$ is equivalent to solving the linear system of equations

$$\hat{\beta} = Ky \tag{5}$$

where $\hat{\beta} \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, and $K \in \mathrm{mat}(n, m)$. We can decompose this problem as follows: for each $i = 1, 2, \dots, n$, we find the $i^{\text{th}}$ row of $K$ that creates the optimal linear combination of $y_j$, which will equal $\hat{\beta}_i$. Because the $y_i$'s are orthogonal to one another by definition 1, we can apply the modified projection theorem (Luenberger §3.10, Theorem 1, p. 64) to construct a solution to equation (5) with the Gram-Schmidt matrix $E(yy^T)$ and the constraint equivalent $E(y\beta)$. Writing these solutions simultaneously (p. 88), we obtain

$$[E(yy^T)]K^T = E(y\beta)$$

and since the covariance matrix of $y$ is invertible by assumption, we obtain $K = \hat{\beta}$ as

$$K = E(\beta y^T)[E(yy^T)]^{-1}. \tag{6}$$

Existence and uniqueness follows from Luenberger's Theorem 2 (§3.10, p. 65).

□

## 3 Discrete random processes (Kalman)

The discrete random process model assumes that variables are linear combinations of past values, so that the behavior $x \in \mathbb{R}^n$ of the $n-$dimensional system at time $k+1$ is a linear combination $\Phi_k$ of its state at time $k$ plus an error or noise term $\mu(k)$:

$$x(k+1) = \Phi_k x(k) + \mu(k) \tag{7}$$

where, as in Luenberger (p. 95), $\mu(k)$ has mean zero, and a covariance matrix equivalent to the covariance matrix of the $m$-dimensional error term defined in equation (1), $\epsilon$, with its off-diagonal terms set to zero:

$$E[u(k)u(l)^T] = E(\epsilon\epsilon^T)\delta_{kl} = Q\delta_{kl},$$

where $\delta_{kl}$ is equal to 1 for $k = l$ and 0 otherwise (the "Kronecker delta"). The intuitive explanation for this is that each distinct error is drawn independently from the same distribution, so their correlations will be zero. Naturally, the "initial random vector $x(0)$ together with an initial estimate $\hat{x}(0)$ having covariance $E[(\hat{x}(0) - x(0))(\hat{x}(0) - x(0))^T] = P(0)$" (p. 95) is given.

Finally, we collect data (p. 95) for $t = 0, 1, \ldots$ that looks like

$$v(t) = Mx(t) + \eta(t) \tag{8}$$

where $M$ is a matrix of $\dim(\Phi)$ and $\eta$ is another random process with mean zero and positive covariance matrix $R_t$.

Kalman [3] formulates the solution below.

**Theorem 2.** *(Kalman, p. 41) In the discrete random process model with information collected as in equation (8) from time $t = 0, \ldots, k$, the optimal estimate for the position of an $n-$dimensional object at time $k+1$ is*

$$\hat{x}(k+1|k) = \Phi_{k+1|k}\hat{x}(k|k-1) + \Delta^*(k)[v(k) - M_k\hat{x}(k|k-1)] \tag{9}$$

*with recursive matrices for the "updating," $\Delta^*(k)$, and for the covariance matrix of the estimate, $P_{k+1}$, given by*

1. $\Delta^*(k) = \Phi_{k+1|k}P_kM_t^T[M_kP_kM_k^T + R_k]^{-1}$
2. $P_{k+1} = [\Phi_{k+1|k} - \Delta^*(k)M_t]P_k\Phi_{k+1|k}^T + Q_t$

*Proof.* (Luenberger, p. 96-97). This algorithm is iterative, so we will assume that $\hat{x}(k|k-1)$ and its covariance matrix $P_k$ already have been calculated on the basis of data from $t = 0, \ldots k-1$. First, we "update" our optimal guess for $x(k)$ at $k$, which is the old guess added to an adjustment from the familiar minimum-variance result proved above in theorem one, $E(\beta y^T)E(yy^T)^{-1}y$, where

- $y = v(k) - M_k\hat{x}(k|k-1)$,
- $E(\beta y^T) = P_kM_k^T$, and
- $E(yy^T) = M_kP_kM_k^T + R_k$ because $w$ and $x$ are orthogonal.

So,

$$\hat{x}(k|k) = \hat{x}(k|k-1) + P_kM_k^T[M_kP_kM_k^T + R_k]^{-1}[v(k) - M_k\hat{x}(k|k-1)] \tag{10}$$

with the error given by

$$\begin{aligned} P_{k|k} &= P_k - P_kM_k^T[M_kP_kM_k^T + R_k]^{-1}M_kP_k \\ &= P_k - \Phi_{k+1|k}^{-1}\Delta^*(k)M_kP_k \end{aligned} \tag{11}$$

again symmetric to our result in theorem 1 (equation (4)). And our best guess, with the fact that the linear predictor minimizing $||\beta - \hat{\beta}||^2$ will also minimize $E(\beta_i - \hat{\beta}_i)$ for all $i$ (Luenberger §4.6, theorem 1, pp. 90-91), coupled with the orthogonality of $\mu(k)$, will be

$$\hat{x}(k+1|k) = \Phi_k\hat{x}(k|k) \tag{12}$$

And substituting (10) into (12) gives the best guess, (9). As in our derivation of (12), the error covariance of (12) is

$$P_{k+1} = \Phi_kP_{k|k}\Phi_k^T + Q_k \tag{13}$$

which we can rewrite as the second recursive matrix in Theorem 2 by substituting equation (11) for $P_{k|k}$. □

## 4 Alien invasion: an application

### 4.1 Formulation of the UFO problem

The U.S. Department of Defense has observed an unidentified foreign object (UFO) moving in the atmosphere (the $z$-axis traces through the North and South poles, and the $xy$ plane intercepts the equator) and collected a matrix of data vectors in $\mathbb{R}^3$ indexing its location each hour from $t = 0, \ldots, 10$. Unfortunately, due to chronic budget cuts after Republicans pushed the country over the fiscal cliff, the military's locating equipment is no long as precise as it was during the Cold War. It collects measurements of the form

$$v(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x(t) + \epsilon(t) \tag{14}$$

where $\epsilon^T = \begin{bmatrix} \epsilon_1 & \epsilon_2 & \epsilon_3 \end{bmatrix}$ is a three-dimensional vector with each of its components independently distributed along the standard normal, such that $\epsilon_i \sim \mathrm{N}(0, 1)$ for $i = 1, 2, 3$.

Table 1: The military's data, $t = 0, \ldots 10$.

| v(1) | v(2) | v(3) | v(4) | v(5) | v(6) | v(7) | v(8) | v(9) | v(10) |
|---|---|---|---|---|---|---|---|---|---|
| 0.80 | 0.42 | -0.08 | 1.92 | -1.31 | 7.08 | -13.68 | 34.82 | -72.4586 | 152.18 |
| 0.08 | 1.12 | 2.21 | -1.34 | 5.30 | -8.11 | 13.38 | -30.31 | 66.4670 | -137.35 |
| -0.66 | 2.27 | -0.24 | 2.69 | -4.88 | 11.46 | -27.71 | 60.17 | -124.3533 | 259.90 |

On the basis of intercepted alien communications, the CIA claims that the UFO's true path is given by the linear differential equation

$$x(k+1|k) = \begin{bmatrix} 1 & \frac{1}{2} & -\frac{3}{2} \\ 1 & -1 & 0 \\ -\frac{1}{2} & \frac{3}{2} & -1 \end{bmatrix} x(k) + \eta(t) \tag{15}$$

The military, fearing that China might decide to strike the UFO, plans to launch their own nuclear missile to knock out the UFO. Concerned that their location estimate will become less and less accurate as the UFO continues to move (since each additional estimate adds noise to the location), they need us to predict the location of the UFO in the eleventh hour.

### 4.2 The solution to the UFO problem

The solution follows almost immediately from equation (9). Note that

$$\Phi = \begin{bmatrix} 1 & \frac{1}{2} & -\frac{3}{2} \\ 1 & -1 & 0 \\ -\frac{1}{2} & \frac{3}{2} & -1 \end{bmatrix},$$

that $M = [I]_3$, $Q = [I]_3$, and $R = [I]_3$ as well. $P(0)$ is the identity matrix, and $P$ is generated recursively. Our optimal vector is given by the equation

$$\hat{x}(k+1|k) = \Phi\hat{x}(k|k-1) + \Delta^*(k)[v(k) - M\hat{x}(k|k-1)] \tag{16}$$

with

$$\Delta^*(k) = \Phi P_k M^T[MP_kM_k^T + R_k]^{-1} = \Phi P_k[P_k + I_3]^{-1}. \tag{17}$$

We compute this explicitly (with MATLAB code in the appendix detailing the arithmetic) to be

$$\hat{x}(k+1|k) = \begin{pmatrix} -310.06 \\ 288.45 \\ -544.41 \end{pmatrix} \tag{18}$$

which turns out to be very close to the true location of the UFO,

$$x(k+1) = \begin{pmatrix} -304.34 \\ 284.35 \\ -534.63 \end{pmatrix}.$$

So, in this case, a very large-tipped missile might successfully avert an alien invasion! It is possible, of course, to rerun the MATLAB code for $N$ invasions and generate a posterior distribution of accuracies of eleventh-hour nuclear strikes. In this application, however, the elegance and simplicity of the Kalman solution to the filtering problem – as well as its possible military applications (think: summer internship) – should be evident.

## References

[1] Anderson, Brian D. and John B. Moore (1979). *Optimal Filtering.* Englewood Cliffs, NJ: Prentice-Hall.

[2] Doucet, Arnaud, Simon Godsill and Christophe Andrieu (2000). "On sequential Monte Carlo sampling methods for Bayesian filtering." *Statistics and Computing*, Vol. 10, pp. 197-208.

[3] Kalman, R.E. (1960). "A New Approach to Linear Filtering and Prediction Problems." *Transactions of the AME–Journal of Basic Engineering*, Vol. 82 (series D), pp. 35-45.

[4] Luenberger, David G. (1969). *Optimization by Vector Space Methods.* New York, NY: Wiley.

**APPENDIX: MATLAB Code.**

```matlab
%Construct the vector difference equation
%bigX is a matrix of x's: X = [x(1), x(2), ... x(k), x(k+1)]
%totaldraws
K=11;
x=1;
d=x;
n=3;
m=n;
bigQ=[zeros(n,K*n)];
for i=1:3
bigQ(i,i)=i/i;
end

%dimension of the state vector
x0=ones(n,1);
bigX=[x0'; zeros(K-1,n)]'
%Phi = eye(n);
Phi = [1 .5 -1.5; 1 -1 0; -.5 1.5 -1];
%Phi = ones(3,3);
for k = 2:K
    u = randn(n,1);
    bigX(:,k)=Phi*bigX(:,k-1)+u;
    for j=1:m
    bigQ(j,3*k-3+j)=u(j)^2;
    end
end

%Generate a vector of observations, call it bigV = [v(1) | v(2) | ..]
%dimofmeasurements
m=3;
bigV=[zeros(n,K)];
bigM=[zeros(n,K*n)];
for i=1:3
bigM(i,i)=i/i;
end

bigR=[zeros(n,K*n)];
for i=1:3
bigR(i,i)=i/i;
end

%Eachmeasurement
for k=1:K
M = eye(n);
w = randn(m,1);
bigV(:,1)=M*x0+w;
for i=1:n
bigM(:,n*k+i)=M(:,i);
end

bigV(:,k) = M*bigX(:,k)+w;

%Covariance matrix
for j=1:m
bigR(j,3*k-3+j)=w(j)^2;
```

```
end
end

%assume x(0)=v(0)
xhat0=bigV(:,1);
bigP=[zeros(n,K*n)];
for i=1:3
bigP(i,i)=i/i;
end

bighatX=[xhat0';zeros(K-1,n)]';
for k = 2:K
    R=eye(n);
    Q=eye(n);
    %R=[bigR(:,3*k) bigR(:,3*k-1) bigR(:,3*k-2)];
    %Q =[bigQ(:,3*k-2) bigQ(:,3*k-1) bigQ(:,3*k) ];
    M=[bigM(:,3*k-2) bigM(:,3*k-1) bigM(:,3*k)];
    %Q = eye(n);
    Pprior=[bigP(:,3*k) bigP(:,3*k-1) bigP(:,3*k-2)];
    P=Phi*Pprior*(eye(n)-M'*(M*Pprior*M'+R)^(-1)*M*Pprior)*Phi'+Q;
    v=bigV(:,k-1);
    hatx=bighatX(:,k-1);
    bighatX(:,k)=Phi*P*M'*[M*P*M'+R]^(-1)*[v-M*hatx]+Phi*hatx;
    %bighatX(:,k)=v-hatx+hatx;
end

bigbighatX=[zeros(n,1) bighatX];

errorfromprediction=(sum(abs(bigX-bighatX)))
avgerror=sum(errorfromprediction)/(K*3)
firsterrorfromprediction=(sum(abs(bigX(:,1:K/2)-bighatX(:,1:K/2))));
secondhalferrorfrompred=(sum(abs(bigX(:,K/2+1:K)-bighatX(:,K/2+1:K))));
avgerror1=sum(firsterrorfromprediction)/(K*1.5)
avgerror2=sum(secondhalferrorfrompred)/(K*1.5)
```