Rob 501 - Mathematics for Robotics
HW #8

Prof. Grizzle

Due Nov. 15, 2018
3PM via Gradescope

Remarks: Start Problem 4 early enough that you can seek MATLAB help in time! Problems 1 through 3 use the minimum variance estimator (MVE); do all the calculations in MATLAB (you do not need to turn in your code). In Problem 4, we start playing with the Kalman filter. Problems 5 and 6 develop the theory behind the "magic" formula for under determined equations. Problem 5 is hard, but the hints essentially solve it for you. I suggest that you save Problem 5 for last. When working Problem 6, it is important to realize that the Lemmas stated in Problem 5 give you everything except the Normal Equations. Problem 7 tries to unify a few concepts.

1. You are given the data
$$y = Cx + \epsilon$$

   and

$$C = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 0 \\ 0 & 6 \end{bmatrix} \quad y = \begin{bmatrix} 1.5377 \\ 3.6948 \\ -7.7193 \\ 7.3621 \end{bmatrix} \quad E\{\epsilon\epsilon^\top\} = Q = \begin{bmatrix} 1.00 & 0.50 & 0.50 & 0.25 \\ 0.50 & 2.00 & 0.25 & 1.00 \\ 0.50 & 0.25 & 2.00 & 1.00 \\ 0.25 & 1.00 & 1.00 & 4.00 \end{bmatrix} \quad E\{xx^\top\} = P = \begin{bmatrix} 0.5 & 0.25 \\ 0.25 & 0.5 \end{bmatrix}$$

   As in class, $E\{x\} = 0$ and $E\{\epsilon\} = 0$.

   (a) Find the Minimum Variance Estimate for x, using only the first value of y and the upper left entry of Q. Also compute the covariance of the estimate.

   (b) Find the Minimum Variance Estimate for x, using only the first two values of y and the upper left 2×2 part of Q. Also compute the covariance of the estimate.

   (c) Find the Minimum Variance Estimate for x, using only the first three values of y and the upper left 3×3 part of Q. Also compute the covariance of the estimate.

   (d) Find the Minimum Variance Estimate for x, using all the values of y and Q. Also compute the covariance of the estimate.

2. This problem reuses some of the data in Problem 1, namely the FULL vector $y \in \mathbb{R}^4$ and the 4×2 matrix C.

   (a) Ignore all of the stochastic data, and do a standard least squares approximation of x, using the inner product $< x, y > = x^\top y$. Yes, the problem is then our usual over determined system of equations.

   (b) Find a BLUE of x assuming $Q = E\{\epsilon\epsilon^\top\} = I$.

(c) Find the Minimum Variance Estimate for x, assuming $E\{\epsilon\epsilon^\top\} = Q = I$ and $P = E\{xx^\top\} = 100I$ (identity matrix times 100). Repeat for $P = 10^6 I$. (Conceptually, you are taking $P \to \infty I$.)

(d) Compare all of your estimates[1].

3. This problem reuses the covariance data in Problem 1, but this time, the means are no longer zero. The minimum variance estimator (MVE) becomes[2]

$$\widehat{x} = \bar{x} + PC^\top (CPC^\top + Q)^{-1}(y - \bar{y}) \ \text{ and } \ E\{(x - \hat{x})(x - \hat{x})^\top\} = P - PC^\top (CPC^\top + Q)^{-1}CP,$$

where $\bar{x} = E\{x\}$, $\bar{\epsilon} = E\{\epsilon\}$ and $\bar{y} = C\bar{x} + \bar{\epsilon}$. Assuming the data in Problem 1, plus

$$\bar{x} = \left[\begin{array}{c} 1 \\ -1 \end{array} \right] \ \text{ and } \ \bar{\epsilon} = 0,$$

determine the MVE for x using the full vector y. Turn in $\widehat{x}$; you do not have to provide the covariance.

4. Download the file HW08KFdata.zip from the MATLAB folder on CANVAS, and unzip it. The file contains a discrete-time planar model of a Segway, some data, and a test file. In MATLAB, run the command

```
>> SegwayTest
```

You should first see a low-budget animation of a Segway, just to convince you that you are working with a physical system. If you want to know more about the model, read the file Segway560.pdf (it is contained in the zip file); this is optional. The state vector in the model consists of the angle of the Segway support bar with respect to the vertical, φ, the angle of the wheel with respect to the base, θ, and the corresponding velocities. Hence,

$$x = \left[\begin{array}{c} \varphi \\ \theta \\ \dot{\varphi} \\ \dot{\theta} \end{array} \right].$$

(a) Download the file KalmanFilterDerivationUsingConditionalRVs from the Handout folder on CANVAS. Using the data in the file SegwayData4KF.mat, implement the one-step Kalman filter on page 9 of the handout, for the model

$$x_{k+1} = Ax_k + Bu_k + Gw_k$$
$$y_k = Cx_k + v_k.$$

The model data is given to you

```
>> clear all
>> load  SegwayData4KF.mat
>> whos
```

[1]How would your comparisons have changed if we had assumed $Q \succ 0$ not equal to the identity matrix, as long as in (a) we used $< x, y > = x^\top Q^{-1}y$? Almost no change at all. BLUE equals weighted deterministic least squares with the weighting equal to the inverse of the covariance matrix of the noise. And MVE reduces to BLUE when the covariance of the unknown x becomes very large. In this case, reading the fine print gave you the answer to the question. :) !

[2]Notice the form of the estimate: best estimate of x given no measurements, plus a gain times the measurement minus its best estimate given the prior knowledge. Compare to RLS, compare to Kalman filter, compare to result in HW #7.

In this example, the model matrices are constant: for all $k \geq 0$, $A_k = A$, $B_k = B$, $G_k = G$, $C_k = C$, and the noise covariance matrices are constant as well $R_k = R$, and $Q_k = Q$. The model comes from a linear approximation about the origin of the nonlinear Segway model. You can learn how to compute such approximations in EECS 562 (Nonlinear Control). A deterministic input sequence u_k is provided to excite the Segway and cause it to roll around. The measurement sequence y_k corresponds to the horizontal position of the base of the Segway. x0 and P0 are the mean and covariance of the initial condition x_0. The number of measurements is N.

(b) Run your Kalman filter using the data in `SegwayData4KF.mat`. Turn in the following plots:

- On one plot, $\widehat{\varphi}$ and $\widehat{\theta}$ versus time, t, or versus the time index k (either is fine).

- On a second plot, $\dot{\widehat{\varphi}}$ and $\dot{\widehat{\theta}}$ versus time, t or versus the time index k (either is fine).

- On a third plot, the four components of your Kalman gain K versus time, t, or versus the time index k (either is fine).
  Remark: $t(k) = kT_s$, where T_s is the sampling period.

(c) You should see the components of your Kalman gain K_k converging to constant values. Report these steady-state values. Then, execute the command below and compare Kss to your steady-state value of K.

```
[Kss,Pss] = dlqe(A,G,C,R,Q)
```

Using K_{ss} in place of K_k is called the steady-state Kalman filter. When the model matrices are time invariant, it is quite common to use the steady-state Kalman filter.

5. Let $(\mathcal{X}, \mathbb{R}, < \cdot, \cdot >)$ be a finite-dimensional inner product space. Let $\{y_1, \cdots, y_p\}$ be a linearly independent set in $\mathcal{X}$ and let $c_1, \cdots, c_p$ be real constants. Define

$$V = \{x \in \mathcal{X} \mid \; <x, y_i> = c_i, \; 1 \leq i \leq p\}.$$

Prove the following:

(a) **Lemma 1:** There exists a unique $x_0 \in \mathrm{span}\{y_1, \cdots, y_p\}$ such that $<x_0, y_i> = c_i$, $1 \leq i \leq p$.

Remark: Another way of stating your result in (a) is that there exists a unique $x_0 \in \mathcal{X}$ such that

$$V \cap \mathrm{span}\{y_1, \cdots, y_p\} = \{x_0\}.$$

(b) **Lemma 2:** Let $M = (\mathrm{span}\{y_1, \cdots, y_p\})^\perp$. Then $V = x_0 + M$; in other words, $x \in V$ if, and only if, $(x - x_0) \perp \mathrm{span}\{y_1, \cdots, y_p\}$.

(c) **Lemma 3:** There exists a unique $v^* \in V$ having minimum norm, and v^* is characterized by $v^* \perp M$ (just for emphasis, we note that the result does not say that $v^* \perp V$).

Remark: We note that $v^* \perp M \Leftrightarrow v^* \in \mathrm{span}\{y_1, \cdots, y_p\}$ because $\mathcal{X} = M \oplus M^\perp$ implies that

$$M^\perp := \left(\mathrm{span}\{y_1, \cdots, y_p\}^\perp\right)^\perp = \mathrm{span}\{y_1, \cdots, y_p\}$$

Remark: We are using the standard induced norm, $||x|| = <x, x>^{1/2}$.

Remark: There exists v^* having minimum norm means $||v^*|| = \inf_{v \in V} ||v||$, and thus

$$v^* = \arg \min_{v \in V} ||v||.$$

3

6. Using Lemmas 1 through 3, prove the following: **Theorem:** Let $(\mathcal{X}, \mathbb{R}, <\cdot, \cdot>)$ be a finite-dimensional inner product space. Let $\{y_1, \cdots, y_p\}$ be a linearly independent set in $\mathcal{X}$ and let $c_1, \cdots, c_p$ be real constants. Define $V = \{x \in \mathcal{X} \mid <x, y_i> = c_i, \ 1 \le i \le p\}$. Then there exists a unique $v^* \in V$ such that

$$v^* = \arg \min_{v \in V} ||v||.$$

Moreover, $v^* = \sum_{i=1}^{p} \beta_i y_i$, where the β_i's satisfy the normal equations

$$
\begin{bmatrix}
<y_1, y_1> & <y_2, y_1> & \cdots & <y_p, y_1> \\
<y_1, y_2> & <y_2, y_2> & \cdots & <y_p, y_2> \\
\vdots & \vdots & \ddots & \vdots \\
<y_1, y_p> & <y_2, y_p> & \cdots & <y_p, y_p>
\end{bmatrix}
\begin{bmatrix}
\beta_1 \\
\beta_2 \\
\vdots \\
\beta_p
\end{bmatrix}
=
\begin{bmatrix}
c_1 \\
c_2 \\
\vdots \\
c_p
\end{bmatrix}
$$

Remark: What you have just solved is a special case of a Quadratic Program, typically called a **QP** for short. We'll talk more about this in next week's HW set!

7. This problems seeks to relate several concepts we have seen in the course. Suppose that that X and Y are jointly distributed normal random variables with

$$
\mu =: \begin{bmatrix} \mathcal{E}\{X\} \\ \mathcal{E}\{Y\} \end{bmatrix} = \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix} \quad \text{and} \quad \Sigma = \text{cov}(\begin{bmatrix} X \\ Y \end{bmatrix}, \begin{bmatrix} X \\ Y \end{bmatrix}) = \begin{bmatrix} P & PC^\top \\ CP & CPC^\top + Q \end{bmatrix}
$$

   (a) Compute the mean and covariance of X conditioned on $Y = y$, using Fact 1 in the handout on Gaussian Random Vectors, and compare to the formula for the MVE given in Problem 3.

   (b) Compute the Schur complement of $CPC^\top + Q$ in Σ and compare to the covariance of X conditioned on Y.

   Remark: Why is this interesting? The Minimum Variance Estimator (MVE) was derived using the Projection Theorem. We computed $\hat{x}$ as the orthogonal projection of x onto the measurement, y. From this problem, we get the hint that when working with Gaussian random vectors, conditional expectations are orthogonal projections. If you take EECS 564 (Estimation and Detection), this fact is actually proven! I hope this helps to bring together the various estimation schemes.

Hints

Hints: Prob. 1 Recall our formulas

$$\widehat{K} = PC^\top(CPC^\top + Q)^{-1} = (C^\top Q^{-1}C + P^{-1})^{-1}C^\top Q^{-1}$$

and

$$E\{(\hat{x} - x)(\hat{x} - x)^\top\} = P - PC^\top(CPC^\top + Q)^{-1}CP$$

Hints: Prob. 3 The problem is as simple as it looks. Its purpose is to make you aware of the MVE when the means are non-zero. If you compare this formula to the measurement update step of the Kalman filter, you will see that they are basically the same. This is because computing a conditional expectation with Gaussian random vectors is really an orthogonal projection. You can learn more about this in EECS 564, Estimation, Filtering, and Detection.

Hints: Prob. 4

(a) The file `testSegway` illustrates how to simulate a deterministic discrete-time model using a `for loop`. While the physical model is assumed to be subjected to random perturbations, the noise terms themselves are <u>not</u> part of the Kalman filter: it uses the noise statistics, such as the covariance matrices. Hence, your implementation of the Kalman filter is a deterministic system and can be done in a manner similar to the `for loop` in the file `testSegway`.

(b) If you get stuck, it is OK to post MATLAB questions on Piazza.

Hints: Prob. 5 The set V is similar to looking at the set of solutions of a linear equation $\{x \in \mathbb{R}^n \mid Ax = b\}$. When $b = 0$, you have a subspace. When $b \neq 0$, you have the same subspace translated by x_0 where $Ax_0 = b$. If this confuses you, then ignore it. I am just trying to give you some intuition! The real hints follow:

(a) Express $x_0 = \sum_{i=1}^{p} \alpha_i y_i$, a linear combination of the y_i's, and then plug into the conditions for $x_0 \in V$. You will get something that looks exactly like the normal equations! You will see a Gram matrix. You will recall that a Gram matrix is invertible if, and only if, the set $\{y_1, \cdots, y_p\}$ is linearly independent.

(b) We have that $x_0 \in V$. Suppose that $x \in V$. Then $< x, y_i > = c_i = < x_0, y_i >$ for $1 \leq i \leq p$. Now ask yourself about $< x - x_0, y_i >$ and see what that tells you about the relation of $x - x_0$ to $\text{span}\{y_1, \cdots, y_p\}$.

(c) This one is more conceptual than the previous two parts. We now know what elements in V look like. In particular, $v \in V$ if, and only if, $v = x_0 + m$ for $m \in M$. Because M is a subspace, we can also write this as $v \in V$ if, and only if, $v = x_0 - m$ for $m \in M$ (because $m \in M \Leftrightarrow -m \in M$). Hence, we have

$$\inf_{v \in V} ||v|| = \inf_{m \in M} ||x_0 + m|| = \inf_{m \in M} ||x_0 - m|| = d(x_0, M).$$

From the Projection Theorem, you know a lot about the right side of the above string of equalities. Use this knowledge to characterize the optimal $m^* \in M$, and then apply that knowledge to $v^* = x_0 - m^*$.

Hints: Prob. 6 Almost everything has been proved in Prob. 5. Here, it is mainly a matter of assembling the pieces into the final "beautiful" form of the answer. Note that in Prob. 5, part (a), you derived equations that look remarkably like those in the Theorem of Prob. 6. However, when you were working (a), you had no idea that the particular solution you were computing was in fact the element in V of minimum norm. Part (c) of Prob. 5 is what allows you to make that connection. Because we are working with $\mathcal{F} = \mathbb{R}$, $< y_i, y_j > = < y_j, y_i >$, and thus the Gram matrix is symmetric. Hence, if you end up with the ij component as $< y_i, y_j >$ instead of $< y_j, y_i >$, you are not wrong!