Part 1

Remarks: Problems 1 through 5 involve calculations. Problem 5 is very important. Please spend extra time on it as it is very helpful for understanding the Kalman Filter.

1. Classify each matrix as positive definite, positive semi-definite, or neither. In addition, if the matrix is either positive definite or positive semi-definite, find a square root. You may use MATLAB to factor a symmetric matrix as $\Lambda = O^\top P O$ or as $P = O \Lambda O^\top$.

   (a) 🎃 $= \begin{bmatrix} 1 & 3 \\ 3 & 9 \end{bmatrix}$.

   (b) 💀 $= \begin{bmatrix} 6 & 10 & 11 \\ 10 & 19 & 19 \\ 11 & 19 & 21 \end{bmatrix}$.

   (c) 👻 $= \begin{bmatrix} 2 & 6 & 10 \\ 6 & 10 & 14 \\ 10 & 14 & 18 \end{bmatrix}$.

2. Use the results on Schur Complements to solve the following problems BY HAND:

   (a) Determine if 🎃 $= \begin{bmatrix} 1 & 3 \\ 3 & 8 \end{bmatrix}$ is positive definite or not.

   (b) Determine if 💀 $= \begin{bmatrix} 1 & 0 & 6 \\ 0 & 4 & 7 \\ 6 & 7 & 10 \end{bmatrix}$ is positive definite or not.

   (c) Find the range of a such that the following matrix is positive definite: 👻 $= \begin{bmatrix} 1 & 2 & 6 \\ 2 & 5 & 7 \\ 6 & 7 & a \end{bmatrix}$

3. Find x of minimum norm that satisfies the equation

$$\begin{bmatrix} 1 & 3 & 2 \\ 3 & 8 & 4 \end{bmatrix} x = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

   (a) Use the standard inner product on $\mathbb{R}^3$

(b) Use the inner product $< x, y >= x^\top \begin{bmatrix} 5 & 1 & 9 \\ 1 & 2 & 1 \\ 9 & 1 & 17 \end{bmatrix} y$

4. x has been bewitched to give the following data:

$y =$  $x + \epsilon$

with

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

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

(a) Find the Best Linear Unbiased Estimate (BLUE) for x, using only the first two values of y. Also compute the covariance of the estimate.

(b) Find the Best Linear Unbiased Estimate (BLUE) for x, using only the first three values of y. Also compute the covariance of the estimate.

(c) Find the Best Linear Unbiased Estimate (BLUE) for x, using all the values of y. Also compute the covariance of the estimate.

Note: For (a), you use the first 2 rows of y and  and the upper 2×2 part of Q. For (b), you use the first 3 rows of y and C, as well as the upper 3×3 part of Q. You see the pattern, I hope. Do all the calculations in MATLAB. You do not have to turn in your code.

5. We consider three jointly normal random variables (X, Y, Z), with

$$\text{mean} \quad \mu = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \quad \text{and covariance} \quad \Sigma = \begin{bmatrix} 2 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 2 \end{bmatrix}$$

(a) Compute the conditional distribution of $\begin{bmatrix} X \\ Y \end{bmatrix} |\{Z = z\}$, the conditional distribution of the vector $[X, Y]^\top$ given $Z = z$, which is the same as the joint distribution of the normal random variables $X|\{Z = z\}$ and $Y|\{Z = z\}$. To be extra clear, give the mean vector and covariance matrix for $[X, Y]^\top$ given $Z = z$.

(b) Compute the distribution of $X|_{\{Z=z\}}$ conditioned on $Y|_{\{Z=z\}} = y$.

(c) Compute the conditional distribution of $X|_{\begin{bmatrix} Y = y \\ Z = z \end{bmatrix}}$, or more compactly, $X_{|Y=y,Z=z}$, the conditional distribution of X given the vector $[Y = y, Z = z]^\top$.

(d) Compare your answers for (b) and (c).

Hints

Hints: Prob. 1 Use the `help eig` command in MATLAB.

Hints: Prob. 2 Recall that for a symmetric matrix $M = \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}$ the following are equivalent:

(a) $M \succ 0$

(b) $A \succ 0$ and $C - B^\top A^{-1} B \succ 0$

(c) $C \succ 0$ and $A - BC^{-1} B^\top \succ 0$

Hints: Prob. 3 This is an under determined system of equations and not an over determined system of equations.

Hints: Prob. 4 Recall our formulas (using $C :=$  for convenience):

$$\widehat{K} = (C^\top Q^{-1} C)^{-1} C^\top Q^{-1} \quad \text{and} \quad E\{(\widehat{x} - x)(\widehat{x} - x)^\top\} = (C^\top Q^{-1} C)^{-1}$$

For (a), you use the first 2 rows of y and C and the upper 2×2 part of Q. For (b), you use the first 3 rows of y and C, as well as the upper 3×3 part of Q. You see the pattern, I hope. Do all the calculations in MATLAB. You do not have to turn in your code.

Hints: Note **Fact 1:** Conditional Distributions of Gaussian Random Vectors

(a) Identify $X_1 = \begin{bmatrix} X \\ Y \end{bmatrix}$ and $X_2 = Z$. Based on this, identify and write down Σ_{11}, Σ_{12}, Σ_{21}, and Σ_{22}, and then μ_1 and μ_2, and finally, note that $x_2 = z$. Now apply the formulas for $\mu_{1|2}$ and $\Sigma_{1|2}$. These are the mean and covariance of the jointly normally distributed random variables $X_{|Z=z}$ and $Y_{|Z=z}$.

(b) From (a), we know $X_{|Z=z}$ and $Y_{|Z=z}$ are jointly distributed normal random variables, and we know their mean and covariance. Rename the mean μ and the covariance Σ (i.e., $\mu_{1|2} \to \mu$ and $\Sigma_{1|2} \to \Sigma$). Using Fact 1, identify $X_1 = X_{|Z=z}$ and $X_2 = Y_{|Z=z}$, and then identify and write down Σ_{11}, Σ_{12}, Σ_{21}, and Σ_{22}, and then μ_1 and μ_2, and finally, note that $x_2 = y$. Now apply the formulas for $\mu_{1|2}$ and $\Sigma_{1|2}$. These are the mean and covariance of a normally distributed random variable. Which one? If you do not know, work part (c) and then return here. If you do know, still work part (c).

(c) Go back to the very beginning with our three jointly normal random variables, and this time identify $X_1 = X$ and $X_2 = \begin{bmatrix} Y \\ Z \end{bmatrix}$. Based on this, identify and write down Σ_{11}, Σ_{12}, Σ_{21}, and Σ_{22}, and then μ_1 and μ_2, and finally, note that $x_2 = \begin{bmatrix} y \\ z \end{bmatrix}$. Now apply the formulas for $\mu_{1|2}$ and $\Sigma_{1|2}$. These are the mean and covariance of the normally distributed random variable $X_{|Y=y, Z=z}$.

(d) Read again the handout on Jointly Gaussian Random Vectors. Go back to the beginning and repeat as necessary: **FACT 4:** *If we have jointly distributed normal random vectors, when we condition one* *block of vectors on another, we always obtain either a jointly distributed normal random vector or, if only a scalar quantity is left, a normally distributed random variable.* This is an amazingly useful property of Gaussian (i.e., normal) random variables.

Part 2

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]

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

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

$$\bar{x} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad \text{and} \quad \bar{\epsilon} = 0,$$

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 HW06.zip. 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,

$$x = \begin{bmatrix} \varphi \\ \theta \\ \dot{\varphi} \\ \dot{\theta} \end{bmatrix}.$$

(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

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + Gw_k \\ y_k &= Cx_k + v_k. \end{aligned}$$

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, $\widehat{\dot{\varphi}}$ and $\widehat{\dot{\theta}}$ 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. 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 $\widehat{x}$ 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`.