MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Department of Electrical Engineering and Computer Science

6.241: Dynamic Systems—Fall 2007

Homework 2 Solutions

Exercise 1.4 a) First define all the spaces:

$$\begin{aligned}
\mathcal{R}(A) &= \{y \in \mathbb{C}^m \mid \exists x \in \mathbb{C}^n \text{ such that } y = Ax\} \\
\mathcal{R}^{\perp}(A) &= \{z \in \mathbb{C}^m \mid y'z = z'y = 0, \forall y \in \mathcal{R}(A)\} \\
\mathcal{R}(A') &= \{p \in \mathbb{C}^n \mid \exists v \in \mathbb{C}^m \text{ such that } p = A'v\} \\
\mathcal{N}(A) &= \{x \in \mathbb{C}^n \mid Ax = 0\} \\
\mathcal{N}(A') &= \{q \in \mathbb{C}^m \mid A'q = 0\}
\end{aligned}$$

i) Prove that $\mathcal{R}^{\perp}(A) = \mathcal{N}(A')$.
Proof: Let

$$\begin{aligned}
z \in \mathcal{R}^{\perp}(A) &\rightarrow y'z = 0 \ \forall y \in \mathcal{R}(A) \\
&\rightarrow x'A'z = 0 \ \forall x \in \mathbb{C}^n \\
&\rightarrow A'z = 0 \rightarrow z \in \mathcal{N}(A') \\
&\rightarrow \mathcal{R}^{\perp}(A) \subset \mathcal{N}(A').
\end{aligned}$$

Now let

$$\begin{aligned}
q \in \mathcal{N}(A') &\rightarrow A'q = 0 \\
&\rightarrow x'A'q = 0 \ \forall x \in \mathbb{C}^n \\
&\rightarrow y'q = 0 \ \forall y \in \mathcal{R}(A) \\
&\rightarrow q \in \mathcal{R}^{\perp}(A) \\
&\rightarrow \mathcal{N}(A') \subset \mathcal{R}^{\perp}(A).
\end{aligned}$$

Therefore

$$\mathcal{R}^{\perp}(A) = \mathcal{N}(A').$$

ii) Prove that $\mathcal{N}^{\perp}(A) = \mathcal{R}(A')$.
Proof: From i) we know that $\mathcal{N}(A) = \mathcal{R}^{\perp}(A')$ by switching A with A'. That implies that

$$\mathcal{N}^{\perp}(A) = \{\mathcal{R}^{\perp}(A')\}^{\perp} = \mathcal{R}(A').$$

b) Show that $rank(A) + rank(B) - n \leq rank(AB) \leq \min\{rank(A), rank(B)\}$.
Proof: i) Show that $rank(AB) \leq min\{rank(A), rank(B)\}$. It can be proved as follows:
Each column of AB is a combination of the columns of A, which implies that $\mathcal{R}(AB) \subseteq \mathcal{R}(A)$. Hence, $dim(\mathcal{R}(AB)) \leq dim(\mathcal{R}(A))$, or equivalently, $rank(AB) \leq rank(A)$.
Each row of AB is a combination of the rows of $B \rightarrow$ rowspace $(AB) \subseteq$ rowspace (B), but the dimension of rowspace = dimension of column space = rank, so that $rank(AB) \leq rank(B)$.
Therefore,

$$rank(AB) \leq min\{rank(A), rank(B)\}.$$

ii) Show that $rank(A) + rank(B) - n \leq rank(AB)$.
Let

$$\begin{aligned} r_B &= rank(B) \\ r_A &= rank(A) \end{aligned}$$

where $A \in \mathbb{C}^{m\times n}$, $B \in \mathbb{C}^{n\times p}$.
Now, let $\{v_1, \cdots, v_{r_B}\}$ be a basis set of $\mathcal{R}(B)$, and add $n - r_B$ linearly independent vectors $\{w_1, \cdots, w_{n-r_B}\}$ to this basis to span all of $\mathbb{C}^n$, $\{v_1, v_2, \cdots, v_n, w_1, \cdots, w_{n-r_B}\}$.
Let

$$M = \begin{pmatrix} v_1| & v_2 & \cdots & v_{r_B}| & w_1| & \cdots & w_{n-r_B} \end{pmatrix} = \begin{pmatrix} V & W \end{pmatrix}.$$

Suppose $x \in \mathbb{C}^n$, then $x = M\alpha$ for some $\alpha \in \mathbb{C}^n$.

1. $\mathcal{R}(A) = \mathcal{R}(AM) = \mathcal{R}([AV|AW])$.
   Proof: i) Let $x \in \mathcal{R}(A)$. Then $Ay = x$ for some $y \in \mathbb{C}^n$. But y can be written as a linear combination of the basis vectors of $\mathbb{C}^n$, so $y = M\alpha$ for some $\alpha \in \mathbb{C}^n$.
   Then, $Ay = AM\alpha = x \to x \in \mathcal{R}(AM) \to \mathcal{R}(A) \subset \mathcal{R}(AM)$.
   ii) Let $x \in \mathcal{R}(AM)$. Then $AMy = x$ for some $y \in \mathbb{C}^n$. But $My = z \in \mathbb{C}^n \to Az = x \to x \in \mathcal{R}(A) \to \mathcal{R}(AM) \subset \mathcal{R}(A)$.
   Therefore, $\mathcal{R}(A) = \mathcal{R}(AM) = \mathcal{R}([AV|AW])$.

2. $\mathcal{R}(AB) = \mathcal{R}(AV)$.
   Proof: i) Let $x \in \mathcal{R}(AV)$. Then $AVy = x$ for some $y \in \mathbb{C}^{r_B}$. Yet, $Vy = B\alpha$ for some $\alpha \in \mathbb{C}^p$ since the columns of V and B span the same space. That implies that $AVy = AB\alpha = x \to x \in \mathcal{R}(AB) \to \mathcal{R}(AV) \subset \mathcal{R}(AB)$.
   ii) Let $x \in \mathcal{R}(AB)$. Then $(AB)y = x$ for some $y \in \mathbb{C}^p$. Yet, again $By = V\theta$ for some $\theta \in \mathbb{C}^{r_B} \to ABy = AV\theta = x \to x \in \mathcal{R}(AV) \to \mathcal{R}(AB) \subset \mathcal{R}(AV)$.
   Therefore, $\mathcal{R}(AV) = \mathcal{R}(AB)$.

Using fact 1, we see that the number of linearly independent columns of A is less than or equal to the number of linearly independent columns of AV + the number of linearly independent columns of AW, which means that

$$rank(A) \leq rank(AV) + rank(AW).$$

Using fact 2, we see that

$$rank(AV) = rank(AB) \to rank(A) \leq rank(AB) + rank(AW),$$

yet, there re only $n - r_B$ columns in AW. Thus,

$$\begin{aligned} &\to \quad rank(AW) \leq n - r_B \\ &\to \quad rank(A) - rank(AB) \leq rank(AW) \leq n - r_B \\ &\to \quad r_A - (n - r_B) \leq r_{AB}. \end{aligned}$$

This completes the proof.

Exercise 2.2 (a) For the 2nd order polynomial $p_2(t) = a_0 + a_1 t + a_2 t^2$, we have $f(t_i) = p_2(t_i) + e_i \quad i = 1, \ldots, 16$, and $t_i \in T$. We can express the relationship between y_i and the polynomial as follows;

$$\begin{bmatrix} y_1 \\ \vdots \\ y_{16} \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 \\ \vdots & \vdots & \vdots \\ 1 & t_{16} & t_{16}^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ \vdots \\ e_{16} \end{bmatrix}$$

The coefficients a_0, a_1, and a_2 are determined by the least squares solution to this (overconstrained) problem, $\underline{a} = (A'A)^{-1}A'y$, where $\underline{a}_{LS} = \begin{bmatrix} a_0 & a_1 & a_2 \end{bmatrix}'$.
Numerically, the values of the coefficients are:

$$\underline{a}_{LS} = \begin{bmatrix} 0.5296 \\ 0.2061 \\ 0.375 \end{bmatrix}$$

For the 15th order polynomial, by a similar reasoning we can express the relation between data points y_i and the polynomial as follows:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_{16} \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{15} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_{16} & t_{16}^2 & \cdots & t_{16}^{15} \end{bmatrix} \begin{bmatrix} a_0 \\ \vdots \\ a_{15} \end{bmatrix} + \begin{bmatrix} e_1 \\ \vdots \\ e_{16} \end{bmatrix}$$

This can be rewritten as $\underline{y} = A\underline{a} + \underline{e}$. Observe that matrix A is invertible for distinct $t_i's$. So the coefficients a_i of the polynomial are $\underline{a}_{exact} = A^{-1}y$, where $\underline{a}_{exact} = \begin{bmatrix} a_0 & a_1 & \cdots & a_{15} \end{bmatrix}'$. The resulting error in fitting the data is $\underline{e} = 0$, thus we have a perfect fit at these particular time instants.
Numerically, the values of the coefficients of are:

$$\underline{a}_{exact} = \begin{bmatrix} 0.49999998876521 \\ 0.39999826604650 \\ 0.16013119161635 \\ 0.04457531385982 \\ 0.00699544100513 \\ -0.00976690595462 \\ -0.02110628552919 \\ 0.02986537283027 \\ -0.03799813521505 \\ 0.00337725219202 \\ -0.00252507772183 \\ 0.00072658523695 \\ -0.00021752221402 \\ -0.00009045014791 \\ -0.00015170733465 \\ -0.00001343734075 \end{bmatrix}$$

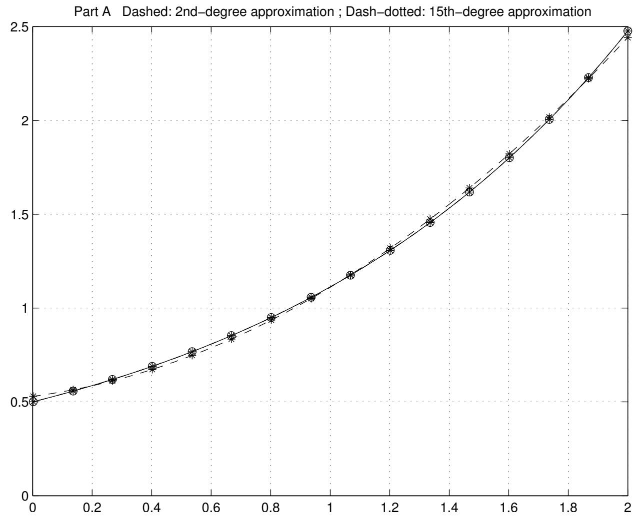

Figure 2.2a

The function $f(t)$ as well as the approximating polynomials $p_{15}(t)$ and $p_2(t)$ are plotted in Figure 2.2b. Note that while both polynomials are a good fit, the fifteenth order polynomial is a better approximation, as expected.

b) Now we have measurements affected by some noise. The corrupted data is

$$\tilde{y}_i = f(t_i) + e(t_i) \quad i = 1, \ldots, 16 \quad t_i \in T$$

where the noise $e(t_i)$ is generated by a command "randn" in Matlab.

Following the reasoning in part (a), we can express the relation between the noisy data points $\tilde{y}_i$ and the polynomial as follows:

$$\underline{\tilde{y}} = A\underline{a} + \underline{\tilde{e}}$$

The solution procedure is the same as in part (a), with y replaced by $\tilde{y}$.

Numerically, the values of the coefficients are:

$$\underline{a}_{exact} = \begin{bmatrix} 0.00001497214861 \\ 0.00089442543781 \\ -0.01844588716755 \\ 0.14764397515270 \\ -0.63231582484352 \\ 1.62190727992829 \\ -2.61484909708492 \\ 2.67459894145774 \\ -1.67594757924772 \\ 0.56666848864500 \\ -0.06211921500456 \\ 0.00219622725954 \\ -0.01911248745682 \\ 0.01085690854235 \\ -0.00207893294346 \\ 0.00010788458590 \end{bmatrix} * 10^5$$

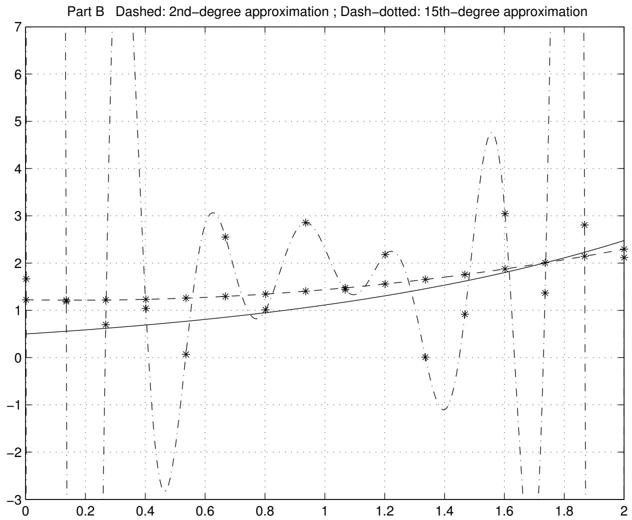

Figure 2.2b

and

$$\underline{a}_{LS} = \begin{bmatrix} 1.2239 \\ -0.1089 \\ 0.3219 \end{bmatrix}$$

The function $f(t)$ as well as the approximating polynomials $p_{15}(t)$ and $p_2(t)$ are plotted in Figure 2.2b. The second order polynomial does much better in this case as the fifteenth order polynomial ends up fitting the noise. Overfitting is a common problem encountered when trying to fit a finite data set corrupted by noise using a class of models that is too rich.

Additional Comments A stochastic derivation shows that the "minimum variance unbiased estimator" for $\underline{a}$ is $\hat{\underline{a}} = argmin\|\tilde{\underline{y}} - A\underline{a}\|_W^2$ where $W = R_n^{-1}$, and R_n is the covariance matrix of the random variable $\underline{e}$. So,

$$\hat{\underline{a}} = (A'WA)^{-1}A'W\tilde{\underline{y}}.$$

Roughly speaking, this is saying that measurements with more noise are given less weight in the estimate of $\underline{a}$. In our problem, $R_n = I$ because the $e_i's$ are independent, zero mean and have unit variance. That is, each of the measurments is "equally noisy" or treated as equally reliable.

c) $p_2(t)$ can be written as

$$p_2(t) = a_0 + a_1 t + a_2 t^2.$$

In order to minimize the approximation error in least square sense, the optimal $\hat{p}_2(t)$ must be such that the error, $f - \hat{p}_2$, is orthogonal to the span of $\{1, t, t^2\}$:

$$< f - \hat{p}_2, 1 >= 0 \rightarrow < f, 1 >=< \hat{p}_2, 1 >$$

$$< f - \hat{p}_2, t >= 0 \rightarrow < f, t >=< \hat{p}_2, t >$$

$$< f - \hat{p}_2, t^2 >= 0 \rightarrow < f, t^2 >=< \hat{p}_2, t^2 > .$$

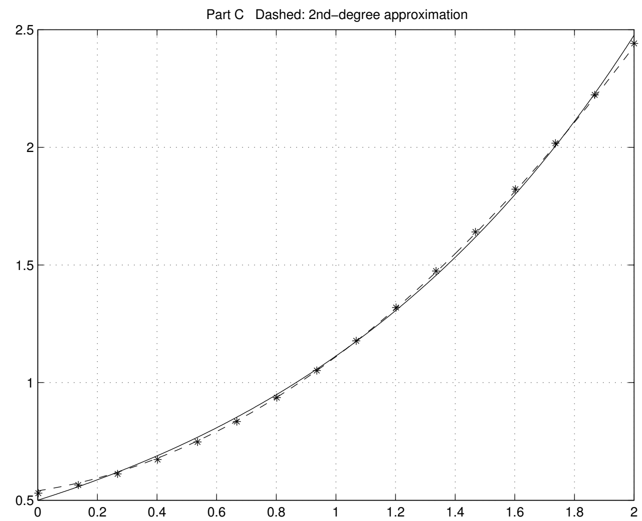

Figure 2.2c

We have that $f = \frac{1}{2}e^{0.8t}$ for $t \in [0, 2]$, So,

$$< f, 1 >= \int_0^2 \frac{1}{2}e^{0.8t}dt = \frac{5}{8}e^{\frac{8}{5}} - \frac{5}{8}$$

$$< f, t >= \int_0^2 \frac{t}{2}e^{0.8t}dt = \frac{15}{32}e^{\frac{8}{5}} + \frac{25}{32}$$

$$< f, t^2 >= \int_0^2 \frac{t^2}{2}e^{0.8t}dt = \frac{85}{64}e^{\frac{8}{5}} - \frac{125}{64}.$$

And,

$$< \hat{p_2}, 1 >= 2a_0 + 2a_1 + \frac{8}{3}a_2$$

$$< \hat{p_2}, t >= 2a_0 + \frac{8}{3}a_1 + 4a_2$$

$$< \hat{p_2}, t^2 >= \frac{8}{3}a_0 + 4a_1 + \frac{32}{5}a_2$$

Therefore the problem reduces to solving another set of linear equations:

$$\begin{bmatrix} 2 & 2 & \frac{8}{3} \\ 2 & \frac{8}{3} & 4 \\ \frac{8}{3} & 4 & \frac{32}{5} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} < f, 1 > \\ < f, t > \\ < f, t^2 > \end{bmatrix}.$$

Numerically, the values of the coefficients are:

$$\underline{a} = \begin{bmatrix} 0.5353 \\ 0.2032 \\ 0.3727 \end{bmatrix}$$

The function $f(t)$ and the approximating polynomial $p_2(t)$ are plotted in Figure 2.2c. Here we use a different notion for the closeness of the approximating polynomial, $\hat{p}_2(t)$, to the original function, f. Roughly speaking, in parts (a) and (b), the optimal polynomial will be the one for which there is smallest discrepancy between $f(t_i)$ and $p_2(t_i)$ for all t_i, i.e., the polynomial that will come closest to passing through all the sample points, $f(t_i)$. All that matters is the 16 sample points, $f(t_i)$. In this part however, all the points of f matter.

Exercise 2.3 Let $y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$, $A = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}$, $e = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}$ and $S = \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix}$.

Note that A has full column rank because C_1 has full column rank. Also note that S is symmetric positive definite since both S_1 and S_2 are symmetric positive definite. Therefore, we know that $\hat{x} =$ argmin $e'Se$ exists and is unique and is given by $\hat{x} = (A'SA)^{-1}A'Sy$.

Thus by direct substitution of terms, we have:

$$\hat{x} = (C_1'S_1C_1 + C_2'S_2C_2)^{-1}(C_1'S_1y_1 + C_2'S_2y_2)$$

Recall that $\hat{x}_1 = (C_1'S_1C_1)^{-1}C_1'S_1y_1$ and that $\hat{x}_2 = (C_2'S_2C_2)^{-1}C_2'S_2y_2$. Hence $\hat{x}$ can be re-written as:

$$\hat{x} = (Q_1 + Q_2)^{-1}(Q_1\hat{x}_1 + Q_2\hat{x}_2)$$

Exercise 2.8 We can think of the two data sets as sequentially available data sets. $\hat{x}$ is the least squares solution to $y \approx Ax$ corresponding to minimizing the euclidean norm of $e_1 = y - Ax$. $\bar{x}$ is the least squares solution to $\begin{bmatrix} y \\ z \end{bmatrix} \approx \begin{bmatrix} A \\ D \end{bmatrix} x$ corresponding to minimizing $e_1'e_1 + e_2'Se_2$ where $e_2 = z - Dx$ and S is a symmetric (hermitian) positive definite matrix of weights.

By the recursion formula, we have:

$$\bar{x} = \hat{x} + (A'A + D'SD)^{-1}D'S(z - D\hat{x})$$

This can be re-written as:

$$\begin{aligned}\bar{x} &= \hat{x} + (I + (A'A)^{-1}D'SD)^{-1}(A'A)^{-1}D'S(z - D\hat{x}) \\ &= \hat{x} + (A'A)^{-1}D'(I + SD(A'A)^{-1}D')^{-1}S(z - D\hat{x})\end{aligned}$$

This step follows from the result in Problem 1.3 (b). Hence

$$\begin{aligned}\bar{x} &= \hat{x} + (A'A)^{-1}D'(SS^{-1} + SD(A'A)^{-1}D')^{-1}S(z - D\hat{x}) \\ &= \hat{x} + (A'A)^{-1}D'(S^{-1} + D(A'A)^{-1}D')^{-1}S^{-1}S(z - D\hat{x}) \\ &= \hat{x} + (A'A)^{-1}D'(S^{-1} + D(A'A)^{-1}D')^{-1}(z - D\hat{x})\end{aligned}$$

In order to ensure that the *constraint* $z = Dx$ is satisfied *exactly*, we need to penalize the corresponding error term heavily ($S \to \infty$). Since D has full row rank, we know there exists at least one value of x that satisfies equation $z = Dx$ exactly. Hence the optimization problem we are setting up does indeed have a solution. Taking the limiting case as $S \to \infty$, hence as $S^{-1} \to 0$, we get the desired expression:

$$\bar{x} = \hat{x} + (A'A)^{-1}D'(D(A'A)^{-1}D')^{-1}(z - D\hat{x})$$

In the 'trivial' case where D is a square (hence non-singular) matrix, the set of values of x over which we seek to minimize the cost function consists of a single element, $D^{-1}z$. Thus, $\bar{x}$ in this case is simply $\bar{x} = D^{-1}z$. It is easy to verify that the expression we obtained does in fact reduce to this when D is invertible.

Exercise 3.1 The first and the third facts given in the problem are the keys to solve this problem, in addition to the fact that:

$$UA = \begin{pmatrix} R \\ 0 \end{pmatrix}.$$

Here note that R is a nonsingular, upper-triangular matrix so that it can be inverted. Now the problem reduces to show that

$$\hat{x} = arg \min_x \|y - Ax\|_2^2 = arg \min_x (y - Ax)'(y - Ax)$$

is indeed equal to

$$\hat{x} = R^{-1}y_1.$$

Let's transform the problem into the familiar form. We introduce an error e such that

$$y = Ax + e,$$

and we would like to minimize $\|e\|_2$ which is equivalent to minimizing $\|y - Ax\|_2$. Using the property of an orthogonal matrix, we have that

$$\|e\|_2 = \|Ue\|_2.$$

Thus with $e = y - Ax$, we have

$$\begin{aligned} \|e\|_2^2 &= \|Ue\|_2^2 = e'U'Ue = (U(y - Ax))'(U(y - Ax)) = \|Uy - UAx\|_2^2 \\ &= \left\| \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - \begin{pmatrix} R \\ 0 \end{pmatrix} x \right\|_2^2 = (y_1 - Rx)'(y_1 - Rx) + y_2'y_2. \end{aligned}$$

Since $\|y_2\|_2^2 = y_2'y_2$ is just a constant, it does not play any role in this minimization.
Thus we would lik to have

$$y_1 - R\hat{x} = 0$$

and because R is an invertible matrix, $\hat{x} = R^{-1}y_1$.

6.241J / 16.338J Dynamic Systems and Control
Spring 2011