CS/ECE/ME 532

Homework 3: least squares and quadratic forms

due: Sunday October 2, 2016

1. Products of PSDs. Suppose $P \succeq 0$ and $Q \succeq 0$ are (symmetric) positive semidefinite $n \times n$ matrices.

a) Prove that $PQP \succeq 0$.

b) Prove that $P^k \succeq 0$ for any $k = 1, 2, \ldots$

SOLUTION:

a) Showing that $A \succeq 0$ is equivalent to showing that $x^\mathsf{T} Ax \geq 0$ for all x. In this case, we have:

$$\begin{aligned} x^\mathsf{T}(PQP)x &= x^\mathsf{T} P^\mathsf{T} QPx \\ &= (Px)^\mathsf{T} Q(Px) \\ &\geq 0 \end{aligned}$$

The first equality follows because P is symmetric and the final inequality follows because $Q \succeq 0$. Notice that we never used the fact that $P \succeq 0$, the statement is true so long as P is symmetric, and it need not be positive semidefinite!

b)
- The case $k = 1$ is true by assumption; we already know that $P \succeq 0$.
- The case $k = 2$ is a special case of the result we proved in part (a); simply set $Q = I$.
- The case $k = 3$ is a special case of (a); set $Q = P$.
- The case $k = 4$ is a special case of (a); set $Q = P^2$. This is ok because we already proved that $P^2 \succeq 0$, this was the case $k = 2$.
- We can continue in this manner. The case for any other k is a special case of (a) because it follows from setting $Q = P^{k-2}$

This proof technique is known as *induction*. We start by proving that the result holds for the first few values of k, then we show that subsequent values of k can be proved by appealing to the results for previous k values already proved.

2. Simple least squares. Consider the following matrix and vector:

$$A = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & 1 \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

a) Find the solution $\widehat{x}$ to $\min_x \|Ax - b\|^2$.

b) Make a sketch of the geometry of this particular problem in $\mathbb{R}^3$, showing the columns of A, the plane they span, the target vector b, the residual vector and the solution $\widehat{b} = A\widehat{x}$.

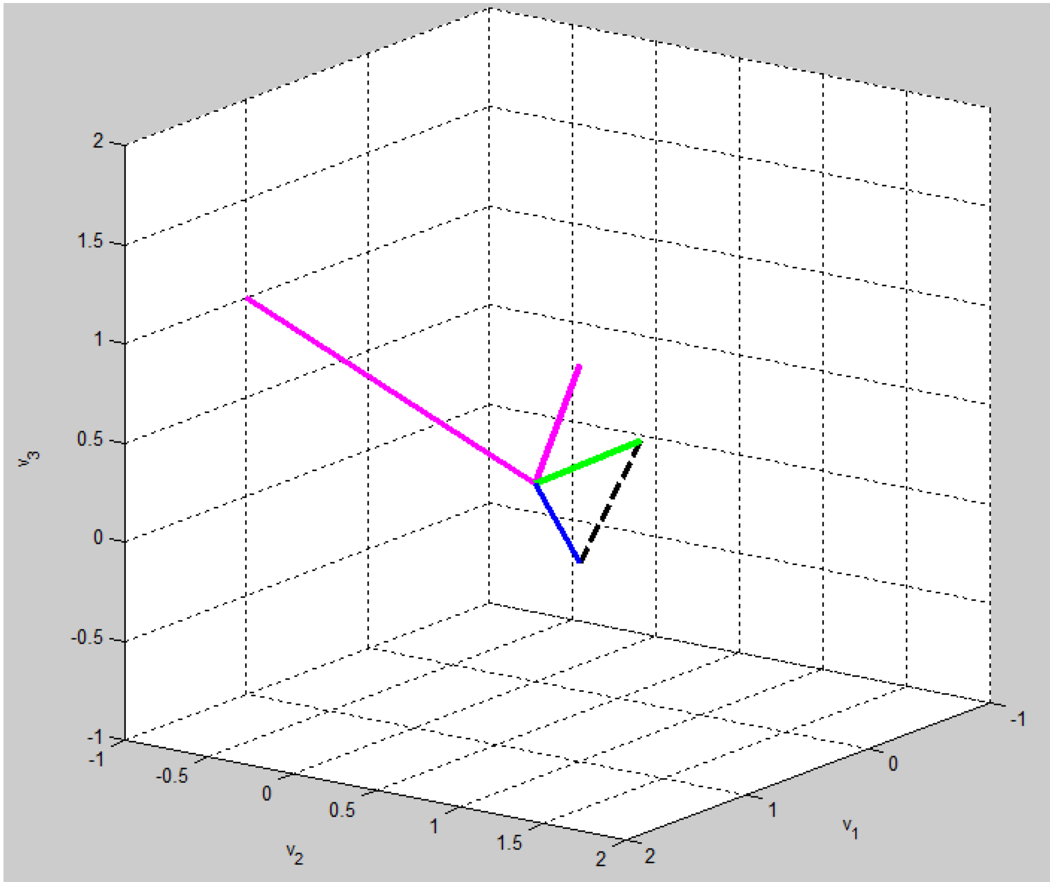
SOLUTION:

(a) The normal equations are: $A^\mathsf{T} Ax = A^\mathsf{T} b$. Substituting A and b, they become:

$$\begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

From the second equation, $x_1 = -3x_2$. Substituting into the first equation, $-8x_2 = 2$. Therefore, $x_1 = \frac{3}{4}$ and $x_2 = -\frac{1}{4}$.

(b) Here is a Matlab plot. We'll accept anything remotely close as a valid sketch!



The magenta lines are the columns of A, the blue line is b, and the green line is the least-squares solution (which lies in the plane spanned by the magenta vectors). The dashed line is the path of the projection (also the residual).

3. Tikhonov regularization. Sometimes we have competing objectives. For example, we want to find an x that minimizes $\|b - Ax\|^2$ (least-squares), but we also want the solution x to have a small norm. One way to achieve a compromise is to solve the following problem:

$$\underset{x}{\text{minimize}} \quad \|b - Ax\|^2 + \lambda \|x\|^2 \tag{1}$$

where $\lambda > 0$ is a parameter we choose that determines the relative weight we want to assign to each objective. This is called *Tikhonov regularization* (also known as L_2 regularization).

a) The optimization problem (1) has its own "normal equations" similar to those we derived for the standard least squares problem. Find them.

Hint: one approach is to reformulate (1) as a modified least squares problem with different "A" and "b" matrices. Another approach is to use the vector derivative method seen in class.

b) Suppose that $A \in \mathbb{R}^{m \times n}$ with $m < n$. Is there a unique least squares solution? Is there a unique solution to (1) ? Explain your answers.

SOLUTION:

a) The augmented cost function is an ordinary least-squares problem in disguise. To see why, notice that

$$\|b - Ax\|^2 + \lambda \|x\|^2 = \left\| \begin{bmatrix} b - Ax \\ \sqrt{\lambda} x \end{bmatrix} \right\|^2 = \left\| \begin{bmatrix} b \\ 0 \end{bmatrix} - \begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix} \right\|^2$$

The normal equations for this new least squares problem are:

$$\left(\begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix}^T \begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix} \right) \widehat{x} = \begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix}^T \begin{bmatrix} b \\ \mathbf{0} \end{bmatrix}$$

which simplifies to:

$$\left(A^T A + \lambda I\right) \widehat{x} = A^T b$$

Alternatively, we can find the modified normal equations directly by differentiating the cost function. Doing so, we obtain:

$$\begin{aligned} \frac{d}{dx}\left(\|b - Ax\|^2 + \lambda \|x\|^2\right) &= \frac{d}{dx}\left((b - Ax)^T (b - Ax) + \lambda x^T x\right) \\ &= \frac{d}{dx}\left(x^T (A^T A + \lambda I) x - 2 b^T A x\right) \\ &= 2(A^T A + \lambda I) - 2 A^T b \end{aligned}$$

Setting the derivative equal to zero, we obtain $\left(A^T A + \lambda I\right) \widehat{x} = A^T b$ as before.

b) There is always a unique solution to this problem, regardless of the dimensions of A. There are many ways to prove this. One way is to examine the modified least-square problem from part (a). The solution will be unique as long as the modified A matrix has full column rank. In other words:

$$\text{does} \quad \widehat{A} = \begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix} \quad \text{have full column rank?}$$

Again, many ways to see why the answer is yes. If $\widehat{A}x = 0$, then we have $Ax = 0$ (first block) and $\sqrt{\lambda}x = 0$ (second block). Therefore $x = 0$, so $\widehat{A}$ has linearly independent columns.

Another way to prove this is to recall from class that $A^T A \succeq 0$ (positive semidefinite). Also, we have $\lambda I \succ 0$ (positive definite). Therefore, $(A^T A + \lambda I) \succ 0$. Positive-definite matrices are always invertible, and so we are done.

4. Polynomial fitting. Suppose we observe pairs of points (a_i, b_i), $i = 1, \ldots, m$. Imagine these points are measurements from a scientific experiment. The variables a_i are the experimental conditions and the b_i correspond to the measured response in each condition. Suppose we wish to fit a degree $d < m$ polynomial to these data. In other words, we want to find the coefficients of a degree d polynomial p so that $p(a_i) \approx b_i$ for $i = 1, 2, \ldots, m$. We will set this up as a least-squares problem.

a) Suppose p is a degree d polynomial. Write the general expression for $p(a_i) = b$. Then, express the $i = 1, \ldots, m$ equations as a system in matrix form $Ax = b$. Specifically, what is the form/structure of b in terms of the given a_i.

SOLUTION: A polynomial of degree d has the form: $p(z) = x_0 + x_1 z + x_2 z^2 + \cdots + x_d z^d$. Where $x_0, \ldots, x_d$ are the coefficients. Note that there are $d + 1$ coefficients. Therefore, the equation $p(a_i) = b_i$ is:

$$x_0 + x_1 a_i + x_2 a_i^2 + \cdots + x_d a_i^d = b_i$$

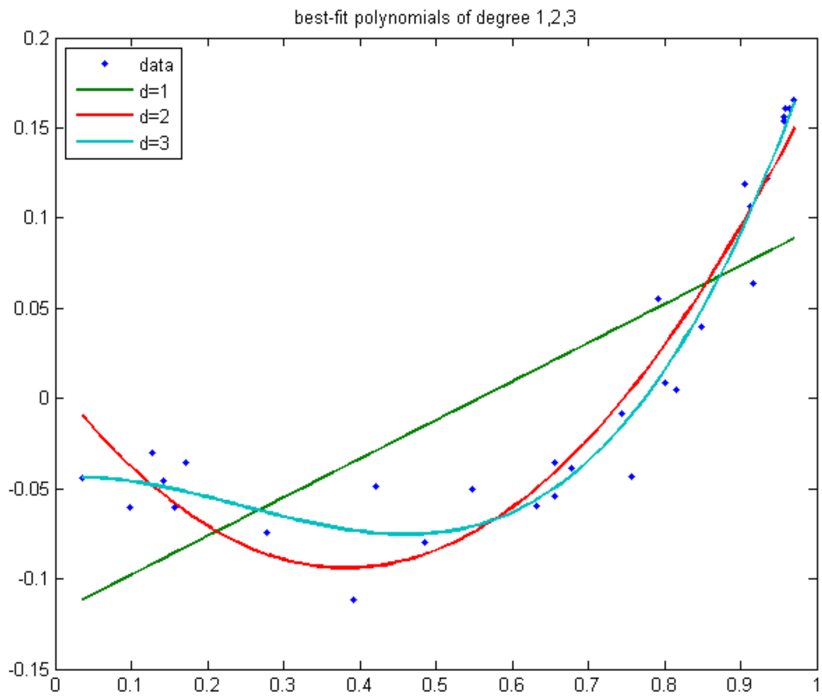
Writing out the equations from the previous part for $i = 1, \dots, m$ and stacking them into a matrix, we obtain the equation:

$$\begin{bmatrix} 1 & a_1 & a_1^2 & \dots & a_1^p \\ 1 & a_2 & a_2^2 & \dots & a_2^p \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & a_m & a_m^2 & \dots & a_m^p \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_m \end{bmatrix}$$

Interesting bit of trivia: matrices of this form are known as *Vandermonde* matrices.

b) Write a Matlab or Python script to find the least-squares fit to the $m = 30$ data points in `polydata.csv`. Plot the points and the polynomial fits for $d = 1, 2, 3$.

SOLUTION: Here is the output of the code:



And here is the Matlab code that produced the plot.

```
% load a and b vectors
load polydata.mat

m = numel(a);                         % number of data points
N = 100;                              % num points to use for interpolation
z = linspace(min(a),max(a),N); % pts where interpolant is evaluated
y = zeros(3,N);                       % where we'll store polynomial values

for d = 1:3

   % generate A-matrix for this choice of d
   A = zeros(m,d+1);
   for i = 1:m
      for j = 1:d+1
```

```
        A(i,j) = a(i)^(j-1);
      end
   end

   % solve least-squares problem. x is the list of coefficients.
   % NOTE: a shortcut in matlab is to just type: x = A\b;
   x = inv(A'*A)*(A'*b);

   % evaluate best-fit polynomial at all points z. store result in y.
   % NOTE: you can do this in one line with the polyval command!
   for i = 1:N
      for j = 1:d+1
         y(d,i) = y(d,i) + x(j)*z(i)^(j-1);
      end
   end

end

% plot the data and the best-fit polynomials
figure(1)
plot(a,b,'.', z,y(1,:), z,y(2,:), z,y(3,:),'LineWidth',2)
legend('data','d=1','d=2','d=3','Location','NorthWest')
title('best-fit polynomials of degree 1,2,3')
```

5. **Calorie prediction for cereal, revisited.** Recall the cereal calorie prediction problem discussed in class. The data matrix for this problem is

$$A = \begin{bmatrix} 25 & 0 & 1 \\ 20 & 1 & 2 \\ 40 & 1 & 6 \end{bmatrix}$$

Each row contains the grams/serving of carbohydrates, fat, and protein, and each row corresponds to a different cereal (*Frosted Flakes, Froot Loops, Grape-Nuts*). The total calories for each cereal are

$$b = \begin{bmatrix} 110 \\ 110 \\ 210 \end{bmatrix}$$

a) Write a short program (e.g., in Matlab or Python) that solves the system of equations $Ax = b$. Recall the solution b gives the calories/gram of carbohydrate, fat, or protein. Verify that the solution you find is the same as the solution we found in class.

SOLUTION: The solution is: $x = \begin{bmatrix} 4.2500 \\ 17.5000 \\ 3.7500 \end{bmatrix}$. Matlab code that produces it is:

```
A = [ 25 0 1
      20 1 2
      40 1 6 ];
b = [ 110 110 210 ]';
x = inv(A)*b
```

b) The solution does not agree with the known calories/gram, which are 4 for carbs, 9 for fat and 4 for protein. We suspect this may be due to rounding the grams to integers, especially for the grams of fat. Assuming the true value for calories/gram is

$$x^\star = \begin{bmatrix} 4 \\ 9 \\ 4 \end{bmatrix}$$

and that the total calories, grams of carbs, and grams of protein *are correctly reported above*, determine the "correct" grams of fat in each cereal.

SOLUTION: Let the unknown grams of fat be f_1, f_2, f_3. The new equations are:

$$\begin{bmatrix} 25 & f_1 & 1 \\ 20 & f_2 & 2 \\ 40 & f_3 & 6 \end{bmatrix} \begin{bmatrix} 4 \\ 9 \\ 4 \end{bmatrix} = \begin{bmatrix} 110 \\ 110 \\ 210 \end{bmatrix}$$

Rearranging and solving for the f's, we obtain:

$$f_1 = \frac{6}{9} \approx 0.667 \qquad f_2 = \frac{22}{9} \approx 2.444 \qquad f_3 = \frac{26}{9} \approx 2.889$$

c) Now suppose that we predict total calories using a more refined breakdown of carbohydrates, into total carbohydrates, complex carbohydrates and sugars (simple carbs). So now we will have 5 features to predict calories (the three carb features + fat and protein). So let's suppose we measure the grams of these features in 5 different cereals to obtain this data matrix

$$A = \begin{bmatrix} 25 & 15 & 10 & 0 & 1 \\ 20 & 12 & 8 & 1 & 2 \\ 40 & 30 & 10 & 1 & 6 \\ 30 & 15 & 15 & 0 & 3 \\ 35 & 20 & 15 & 2 & 4 \end{bmatrix}$$

and the total calories in each cereal

$$b = \begin{bmatrix} 104 \\ 97 \\ 193 \\ 132 \\ 174 \end{bmatrix}$$

Can you solve $Ax = b$? Carefully examine the situation in this case. Is there a solution that agrees with the true calories/gram?

SOLUTION: Since the total carbs are equal to the complex carbs plus the sugars, the first column of the A matrix is equal to the sum of the second and third columns. So the A matrix is *not* full rank. Therefore, we can't simply invert it and solve for x. To see whether the system of equations still has a solution, we can use Matlab to check that:

$$\operatorname{rank}(A) = 4 \qquad \text{and} \qquad \operatorname{rank}\begin{bmatrix} A & b \end{bmatrix} = 4$$

since adding the column b to the matrix A does not increase its rank, b must lie in the span of the columns of A (i.e. there is a solution to $Ax = b$). But how do we find it? One way to do this is to remove the first column of A, which is redundant. We then have:

$$\begin{bmatrix} 15 & 10 & 0 & 1 \\ 12 & 8 & 1 & 2 \\ 30 & 10 & 1 & 6 \\ 15 & 15 & 0 & 3 \\ 20 & 15 & 2 & 4 \end{bmatrix} z = \begin{bmatrix} 104 \\ 97 \\ 193 \\ 132 \\ 174 \end{bmatrix}$$

Solving the associated least-squares problem (which is now full-rank), we obtain the solution:

$$z = \begin{bmatrix} 4 \\ 4 \\ 9 \\ 4 \end{bmatrix}$$

But this isn't just the least-squares solution... it's an exact solution to the problem! (the residual is zero). Moreover, it agrees with the true calories per gram! Due to the rank deficiency of A, the solution to the original problem is not unique. The most general solution to the original problem is:

$$x = \begin{bmatrix} \alpha \\ 4-\alpha \\ 4-\alpha \\ 9 \\ 4 \end{bmatrix} \qquad \text{for any } \alpha \in \mathbb{R}$$

and the solution that agrees with the true carbs/gram is when $\alpha = 0$. Note that this problem is very special. Usually in such situations (e.g. if b is chosen randomly but we keep the same A matrix), there won't exist a solution to $Ax = b$ and we would have to resort to solving the least squares problem (i.e. finding an x that is "close" to being a solution).