MATH 4931 - MSSC 5931 Homework 3

1. **Iteratively reweighted least squares for 1-norm approximation.** In an ordinary least squares problem, we are given $A \in \mathbb{R}^{m \times n}$ (skinny and full rank) and $y \in \mathbb{R}^m$, and we choose $x \in \mathbb{R}^n$ in order to minimize

$$\|Ax - y\|_2^2 = \sum_{i=1}^{m} (\tilde{a}_i^T x - y_i)^2.$$

Note that the penalty that we assign to a measurement error does not depend on the sensor from which the measurement was taken. However, this is not always the right thing to do: if we believe that one sensor is more accurate than another, we might want to assign a larger penalty to an error in the measurement from the more accurate sensor. We can account for differences in the accuracies of our sensors by assigning sensor i a weight $w_i > 0$, and then minimizing

$$\sum_{i=1}^{m} w_i (\tilde{a}_i^T x - y_i)^2.$$

By giving larger weights to more accurate sensors, we can account for differences in the precision of our sensors.

a) *Weighted least squares.* Explain how to choose x in order to minimize

$$\sum_{i=1}^{m} w_i (\tilde{a}_i^T x - y_i)^2,$$

where the weights $w_1, \ldots, w_n > 0$ are given.

b) *Iteratively reweighted least squares for ℓ_1-norm approximation.* Consider a cost function of the form

$$\sum_{i=1}^{m} w_i(x) (\tilde{a}_i^T x - y_i)^2. \tag{3}$$

One heuristic for minimizing a cost function of the form given in (3) is *iteratively reweighted least squares*, which works as follows. First, we choose an initial point $x^{(0)} \in \mathbb{R}^n$. Then, we generate a sequence of points $x^{(1)}, x^{(2)}, \ldots \in \mathbb{R}^n$ by choosing $x^{(k+1)}$ in order to minimize

$$\sum_{i=1}^{m} w_i(x^{(k)}) (\tilde{a}_i^T x^{(k+1)} - y_i)^2.$$

Each step of this algorithm involves updating our weights, and solving a weighted least squares problem. Suppose we want to use this method to solve minimize the ℓ_1-norm approximation error, which is defined to be

$$\|Ax - y\|_1 = \sum_{i=1}^{m} |\tilde{a}_i^T x - y_i|,$$

where the matrix $A \in \mathbb{R}^{m \times n}$ and the vector $y \in \mathbb{R}^m$ are given. How should we choose the weights $w_i(x)$ to make the cost function in (3) equal to the ℓ_1-norm approximation error?

Remark. Suppose we fit the least-squares line to some data. Then, a point that is very far from the least-squares line may be an *outlier*: that is, a point that does not seem to follow the same model as the rest of the data. Because such points may not follow the same model as the rest of data, it may make sense to give such points less weight. This idea is the intuition behind iteratively reweighted least squares for ℓ_1-norm approximation.

2. **Estimating a signal with interference.** This problem concerns three proposed methods for estimating a signal, based on a measurement that is corrupted by a small noise and also by an interference, that need not be small. We have

$$y = Ax + Bv + w,$$

where $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{m \times p}$ are known. Here $y \in \mathbb{R}^m$ is the measurement (which is known), $x \in \mathbb{R}^n$ is the signal that we want to estimate, $v \in \mathbb{R}^p$ is the interference, and w is a noise. The noise is unknown, and can be assumed to be small. The interference is unknown, but cannot be assumed to be small. You can assume that the matrices A and B are skinny and full rank (*i.e.*, $m > n$, $m > p$), and that the ranges of A and B intersect only at 0. (If this last condition does not hold, then there is no hope of finding x, even when $w = 0$, since a nonzero interference can masquerade as a signal.) Three data scientists proposes a method for estimating x. These methods, along with some informal justification from their proposers, are given below. Nikola proposes the **ignore and estimate method.** He describes it as follows:

> We don't know the interference, so we might as well treat it as noise, and just ignore it during the estimation process. We can use the usual least-squares method, for the model $y = Ax + z$ (with z a noise) to estimate x. (Here we have $z = Bv + w$, but that doesn't matter.)

Almir proposes the **estimate and ignore method.** He describes it as follows:

> We should simultaneously estimate both the signal x *and* the interference v, based on y, using a standard least-squares method to estimate $[x^\mathsf{T}\ v^\mathsf{T}]^\mathsf{T}$ given y. Once we've estimated x and v, we simply ignore our estimate of v, and use our estimate of x.

Miki proposes the **estimate and cancel method.** He describes it as follows:

> Almir's method makes sense to me, but I can improve it. We should simultaneously estimate both the signal x and the interference v, based on y, using a standard least-squares method, exactly as in Almir's method. In Almir's method, we then throw away $\hat{v}$, our estimate of the interference, but I think we should use it. We can form the "pseudo-measurement" $\tilde{y} = y - B\hat{v}$, which is our measurement, with the effect of the estimated interference subtracted off. Then, we use standard least-squares to estimate x from $\tilde{y}$, from the simple model $\tilde{y} = Ax + z$. (This is exactly as in Nikola's method, but here we have subtracted off or cancelled the effect of the estimated interference.)

These descriptions are a little vague; part of the problem is to translate their descriptions into more precise algorithms.

a) Give an explicit formula for each of the three estimates. (That is, for each method give a formula for the estimate $\hat{x}$ in terms of A, B, y, and the dimensions n, m, p.)

b) Are the methods really different? Identify any pairs of the methods that coincide (*i.e.*, always give exactly the same results). If they are all three the same, or all three different, say so. Justify your answer. To show two methods are the same, show that the formulas given in part (a) are equal (even if they don't appear to be at first). To show two methods are different, give a specific numerical example in which the estimates differ.

c) Which method or methods do you think work best? Give a very brief explanation. (If your answer to part (b) is "The methods are all the same" then you can simply repeat here, "The methods are all the same".)

3. Identifying a system from input/output data. Suppose $y = Ax + v$, where $x \in \mathbb{R}^n$ is the input, $y \in \mathbb{R}^m$ is the output, $A \in \mathbb{R}^{m\times n}$ is the sensor matrix, and $v \in \mathbb{R}^m$ is measurement noise. Suppose we are given N input/output pairs

$$(x^{(1)}, y^{(1)}), \dots, (x^{(N)}, y^{(N)}).$$

a) Explain how to choose A in order to minimize

$$J = \sum_{k=1}^{N} \|Ax^{(k)} - y^{(k)}\|^2.$$

State any assumptions that are needed for your method to work.

b) Apply your method to the data defined in `system_identification_data.Rdata`. Report the average relative approximation error:

$$\frac{1}{N}\sum_{k=1}^{N} \frac{\|Ax^{(k)} - y^{(k)}\|}{\|y^{(k)}\|}.$$

4. Robust input design. We are given a system, which we know follows $y = Ax$, with $A \in \mathbb{R}^{m \times n}$. Our goal is to choose the input $x \in \mathbb{R}^n$ so that $y \approx y^{\text{des}}$, where $y^{\text{des}} \in \mathbb{R}^m$ is a given target outcome. We'll assume that $m \leq n$, *i.e.*, we have more degrees of freedom in our choice of input than specifications for the outcome. If we knew A, we could use standard methods to choose x. The catch here, though, is that we don't know A exactly; it varies a bit, say, day to day. But we do have some possible values of A,

$$A^{(1)}, \ldots, A^{(K)},$$

which might, for example, be obtained by measurements of A taken on different days. We now define $y^{(i)} = A^{(i)}x$, for $i = 1, \ldots, K$. Our goal is to choose x so that $y^{(i)} \approx y^{\text{des}}$, for $i = 1, \ldots, K$.

We will consider two different methods to choose x.

- *Least norm method.* Define $\bar{A} = (1/K)\sum_{i=1}^{K} A^{(i)}$. Choose x^{ln} to be the least-norm solution of $\bar{A}x = y^{\text{des}}$. (You can assume that $\bar{A}$ is full rank.)
- *Mean-square error minimization method.* Choose x^{mmse} to minimize the mean-square error
$$\frac{1}{K}\sum_{i=1}^{K}\|y^{(i)} - y^{\text{des}}\|^2.$$

a) Give formulas for x^{ln} and x^{mmse}, in terms of y^{des} and $A^{(1)}, \ldots, A^{(K)}$. You can make any needed rank assumptions about matrices that come up, but please state them explicitly.

b) Find x^{ln} and x^{mmse} for the problem with data given in `rob_inp_des_data.Rdata`. Running this Rdata will define `ydes` and the matrices $A^{(i)}$ (given as a 3 dimensional array; for example, `A[,,13]` is $A^{(13)}$). Write down the values of x^{ln} and x^{mmse} you found. What would be the residual norms?

5. Householder reflections. A *Householder matrix* is defined as

$$Q = I - 2uu^{\mathsf{T}},$$

where $u \in \mathbb{R}^n$ is normalized, that is, $u^{\mathsf{T}}u = 1$.

a) Show that Q is orthogonal.

b) Show that $Qu = -u$. Show that $Qv = v$, for any v such that $u^{\mathsf{T}}v = 0$. Thus, multiplication by Q gives reflection through the plane with normal vector u.

c) Given a vector $x \in \mathbb{R}^n$, find a unit-length vector u for which Qx lies on the line through e_1. *Hint:* Try a u of the form $u = v/\|v\|$, with $v = x + \alpha e_1$ (find the appropriate α and show that such a u works ...) Compute such a u for $x = (3, 2, 4, 1, 5)$. Apply the corresponding Householder reflection to x to find Qx.

Note: Multiplication by an orthogonal matrix has very good numerical properties, in the sense that it does not accumulate much roundoff error. For this reason, Householder reflections are used as building blocks for fast, numerically sound algorithms.

6. True/false questions about linear algebra. Determine whether each of the following statements is true or false. In each case, give either a proof or a counterexample.

a) If Q has orthonormal columns, then $\|Q^{\mathsf{T}} w\| \leq \|w\|$ for all vectors w.

b) Suppose $A \in \mathbb{R}^{m\times p}$ and $B \in \mathbb{R}^{m\times q}$. If $\operatorname{null}(A) = \{0\}$ and $\operatorname{range}(A) \subset \operatorname{range}(B)$, then $p \leq q$.

c) If $V = \begin{bmatrix} V_1 & V_2 \end{bmatrix}$ is invertible and $\operatorname{range}(V_1) = \operatorname{null}(A)$, then $\operatorname{null}(AV_2) = \{0\}$.

d) If $\operatorname{rank}\left(\begin{bmatrix} A & B \end{bmatrix}\right) = \operatorname{rank}(A) = \operatorname{rank}(B)$, then $\operatorname{range}(A) = \operatorname{range}(B)$.

e) Suppose $A \in \mathbb{R}^{m\times n}$. Then, $x \in \operatorname{null}(A^{\mathsf{T}})$ if and only if $x \notin \operatorname{range}(A)$.

f) Suppose A is invertible. Then, AB is not full rank if and only if B is not full rank.

g) If A is not full rank, then there is a nonzero vector x such that $Ax = 0$.

7. Least-squares residuals. Suppose A is skinny and full-rank. Let x_{ls} be the least-squares approximate solution of $Ax = y$, and let $y_{\mathrm{ls}} = Ax_{\mathrm{ls}}$. Show that the residual vector $r = y - y_{\mathrm{ls}}$ satisfies

$$\|r\|^2 = \|y\|^2 - \|y_{\mathrm{ls}}\|^2.$$

Also, give a brief geometric interpretation of this equality (just a couple of sentences, and maybe a conceptual drawing).