AMATH 352: Problem Set 5

Instructor: Brian de Silva

February 8, 2017

Due: Friday February 17, 2017

Instructions:

Complete the following problems. Submit a single Matlab script to Scorelator with your solutions to the problems in the Matlab section. You have five chances to submit your assignment. Your score for the Matlab portion of the assignment will be the highest score from your five submissions. The remaining problems (1-9) make up the written portion of this assignment. Turn in a write up of these problems digitally (via Canvas) by 5:00pm of the due date.

Matrices:

1. Consider a real matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ with $m \geq n$ which is full rank, i.e. rank($\boldsymbol{A}$)$= n$.
   (a) Find the nullspace of $\boldsymbol{A}$.
   (b) Show that for any vector $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{x}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} = \|\boldsymbol{A}\boldsymbol{x}\|_2^2$.
   (c) Show that the matrix $\boldsymbol{A}^T\boldsymbol{A}$ is invertible. Hint: show the nullspace of $\boldsymbol{A}^T\boldsymbol{A}$ is $\{\boldsymbol{0}\}$ by assuming $\boldsymbol{z} \in \text{null}(\boldsymbol{A}^T\boldsymbol{A})$ then using the previous parts of this question to show that $\boldsymbol{z} = \boldsymbol{0}$. (Second hint: try multiplying by $\boldsymbol{z}^T$. You will also need one of the defining properties of a norm).

2. Suppose $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ is invertible. Find the inverse of $\boldsymbol{A}^T\boldsymbol{A}$ and show that it is symmetric.

3. Suppose $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n \times n}$. Show that $\boldsymbol{AB}$ is invertible if and only if both $\boldsymbol{A}$ and $\boldsymbol{B}$ are invertible. This means you must show that if $\boldsymbol{AB}$ is invertible, so are $\boldsymbol{A}$ and $\boldsymbol{B}$ and you must also show that if $\boldsymbol{A}$ and $\boldsymbol{B}$ are invertible, so is $\boldsymbol{AB}$. Hint: use the determinant.

Nutrient	Food 1	Food 2	Food 3	Total nutrients required (mg)
Vitamin C	10	20	20	100
Calcium	50	40	10	300
Magnesium	30	10	40	200

Figure 1: Milligrams (mg) of nutrients per unit of food

4. Suppose $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C} \in \mathbb{R}^{n\times n}$ are all invertible. Show that $\boldsymbol{ABC}$ is invertible by finding a matrix $\boldsymbol{D}$ such that $(\boldsymbol{ABC})\boldsymbol{D} = \boldsymbol{D}(\boldsymbol{ABC}) = \boldsymbol{I}$.

5. Based on your answer to the previous problem, what do you think the inverse of $\boldsymbol{A}_1\boldsymbol{A}_2\cdots\boldsymbol{A}_k$ would be, assuming $\boldsymbol{A}_1, \boldsymbol{A}_2, \dots, \boldsymbol{A}_k \in \mathbb{R}^{n\times n}$ are all invertible? You do not have to provide a proof, but you should briefly explain your reasoning.

6. Let $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ be invertible and let $\boldsymbol{B} \in \mathbb{R}^{n\times r}$ for some positive integer r. Show that $\boldsymbol{AX} = \boldsymbol{B}$ has a unique solution. Note that $\boldsymbol{X} \in \mathbb{R}^{n\times r}$ is a matrix.

7. We saw in class that multiplication by an orthogonal matrix preserves lengths (with respect to the 2-norm). In this problem you will show that they also preserve the dot product and orthogonality, that is, the dot product between two vectors is the same as the dot product of any orthogonal matrix applied to the two vectors.

   Let $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ be an orthogonal matrix. Show that for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$

   (a) $(\boldsymbol{Ux}) \cdot (\boldsymbol{Uy}) = \boldsymbol{x} \cdot \boldsymbol{y}$,

   (b) $(\boldsymbol{Ux}) \cdot (\boldsymbol{Uy}) = 0$ if and only if $\boldsymbol{x} \cdot \boldsymbol{y} = 0$.
   (This means you must show that $(\boldsymbol{Ux}) \cdot (\boldsymbol{Uy}) = 0$ implies $\boldsymbol{x} \cdot \boldsymbol{y} = 0$ and that $\boldsymbol{x} \cdot \boldsymbol{y} = 0$ implies $(\boldsymbol{Ux}) \cdot (\boldsymbol{Uy}) = 0$).

Systems of equations

8. A dietician is planning a meal that supplies certain quantities of vitamin C, calcium, and magnesium. Three foods will be included in the diet, their quantities measured in appropriate units. The nutrients supplied by these foods and the dietary requirements are given in Figure 1. Write a linear system of equations which represents the problem of choosing the appropriate amounts of each food that should be consumed to get the desired nutrients. Write the linear system of equations as a matrix-vector equation.

9. For each of the following matrices, $\boldsymbol{A}$, and vectors, $\boldsymbol{b}$, determine the number of solutions to $\boldsymbol{Ax} = \boldsymbol{b}$.

   (a) $\boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$

(b) $\boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$

(c) $\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 1 \\ 1 & 1 & 0 \\ 1 & 2 & 0 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 4 \\ 2 \\ 3 \end{bmatrix}$

(d) $\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}$

(e) $\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 5 \\ -3 \\ 5 \end{bmatrix}$

Matlab:

Create and submit to Scorelator a single Matlab script (.m file) which performs the following tasks. Your code should generate the dat files `A1.dat`, `A2.dat`,..., and `A9.dat`.

Solving systems of equations in Matlab:
Suppose we wish to solve the linear system of equations

$$\boldsymbol{Ax} = \boldsymbol{b},$$

for $\boldsymbol{x}$, where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{b} \in \mathbb{R}^n$ are given. If `A` and `b` are defined in Matlab (and represent $\boldsymbol{A}$ and $\boldsymbol{b}$, respectively) and we wish to solve the above problem and store the solution in the variable `x_sol`, we would run the following command:

```
x_sol = A \ b;
```

The "backslash" command "\" is the primary command used in Matlab to solve linear systems. Given a matrix `M` (on the left of \) and a vector `v` (on the right of \), the backlash command `M \v` returns the approximate solution to `Mx=v`, if it exists (it will still do something if the solution doesn't exist, and we will discuss this later). Behind the scenes the command runs some checks on `M` and attempts to determine the most appropriate algorithm to use to solve the equation. We will learn more about some of these methods later in the course.

10. For each of the systems in problem 9 which had *exactly* one solution, find the solution by defining `A` and `b` in Matlab and using the backslash command. Save the solutions as column vectors in `A1.dat` (2×1 column vector) and `A2.dat` (3×1 column vector), respectively.

11. Suppose we wish to approximate the solution to the following ordinary

differential equation (ODE) on numerically

$$\begin{cases} u''(x) = f(x), & \text{on } (0,1) \\ u(0) = \alpha \\ u(1) = \beta \end{cases}.$$

One way we can obtain a numerical approximation to the solution at a set of 100 equally spaced points ($0 = x_0 < x_1 < x_2 < \cdots < x_{98} < x_{99} = 1$) is by solving the system of linear equations $\boldsymbol{AU} = \boldsymbol{F}$, where

$$\boldsymbol{A} = (99)^2 \begin{bmatrix} 1/99^2 & 0 & & & & & \\ 1 & -2 & 1 & & & & \\ & 1 & -2 & 1 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & 1 & -2 & 1 & \\ & & & & 1 & -2 & 1 \\ & & & & & 0 & 1/99^2 \end{bmatrix}, \quad \boldsymbol{U} = \begin{bmatrix} U_0 \\ U_1 \\ U_2 \\ \vdots \\ U_{97} \\ U_{98} \\ U_{99} \end{bmatrix}, \quad \boldsymbol{F} = \begin{bmatrix} \alpha \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_{97}) \\ f(x_{98}) \\ \beta \end{bmatrix}.$$

Note the factor of 99^2 factored out of $\boldsymbol{A}$ (the 99 comes from the fact that the spacing between the x-values is $1/99$–if we used m points it would become $1/(m-1)$). All entries of $\boldsymbol{A}$ not shown should be assumed to be 0 ($\boldsymbol{A}$ is tridiagonal). The components, U_i, of $\boldsymbol{U}$ are approximations of the actual solution to the ODE at x_i, i.e. $U_i \approx u(x_i)$.

(a) Define a (column) vector `x` of 100 equally spaced points between 0 and 1 (inclusive). Save this vector in `A3.dat`.

(b) Suppose we choose $\alpha = 0, \beta = 1$, and $f(x) = 50\sin(5x)$. Form the 100×1 column vector $\boldsymbol{F}$ given above. Output the result in `A4.dat`. Note that you can check your answer by solving the ODE (just integrate twice and solve for the constants of integration that satisfy the boundary conditions).

(c) Form the 100×100 matrix $\boldsymbol{A}$ given above and save the result in `A5.dat`. (You may wish to revisit your code from problem 8 of homework 4).

(d) Use the backslash command to solve the linear system $\boldsymbol{AU} = \boldsymbol{F}$. Save the result in `A6.dat`. Try plotting your approximate solution (if the solution to the system of equations is stored in `U`, the command `plot(x,U)` should generate the plot). Does it look like the boundary conditions $u(0) = \alpha$ and $u(b) = \beta$ are being satisfied? **Be sure to comment out or delete your plot command before submitting your assignment.**

(e) Now suppose we take $\alpha = 1/2, \beta = 0$, and $f(x) = 3x$ (this results in a new right-hand side vector $\boldsymbol{F}$). Again solve $\boldsymbol{AU} = \boldsymbol{F}$ for $\boldsymbol{U}$ using backslash. Save the result in `A7.dat`. Try plotting your new solution. Does it look qualitatively different from your previous one?

(f) Take $\alpha = 1, \beta = 2$, and $f(x) = -10/(1 + x^2)$, compute the new $\boldsymbol{F}$ and solve the system $\boldsymbol{AU} = \boldsymbol{F}$. Save the result in `A8.dat`. Try plotting the new solution. Does it look like the boundary conditions are satisfied? Does it look similar to your previous solutions?

(g) (Optional) Try playing around with different values for α and β and different choices of the right-hand side function $f(x)$. See what kinds of solutions you can get.

Be sure to comment out or delete your plot commands before submitting your assignment.