# ENGR 695 Advanced Topics in Engineering Mathematics Fall 2024

## Review Sheet for Midterm Exam

The midterm exam is scheduled to take place **4:00–5:00 pm on Monday, October 21 in Dana 221**. The exam will be designed to be completed in less than 1 hour, but the exam period might be extended to 1.5 or 2 hours.

The following policies will be in effect for the exam:

1. You will be allowed to use a non-wireless enabled calculator, such as a TI-99. You must put away smart watches, cell phones, tablets, laptops, and other devices that can connect to the internet wirelessly. I will record your calculator make and model during the exam.
2. You will be allowed to use an unlimited number of paper help sheets. These could include the notes that you recorded in class (either loose or collected in a notebook), homework solutions, supplemental notes from the course Moodle site, copied or printed pages from the textbook, or any other printed or handwritten material. Hard copies of the textbook will not be allowed since not every student has access to one. A copy of the linear system solvability flowchart (Fig. 8.3.1 of Zill, 6th ed.) will be provided. If there is anything else that you feel would be helpful, please notify me.
3. If you begin the exam after the start time, you must complete it in the remaining allotted time. However, you will not be allowed to take the exam if you arrive after the first student has completed it and left the room. The latter case is equivalent to missing the exam.
4. **You may not leave the exam room without prior permission except in an emergency or for an urgent medical condition. Please use the restroom before the exam.**

The following is a list of topics that could appear in one form or another on the exam. Not all of these topics will be covered, and it is possible that an exam problem could cover a detail not specifically listed here. However, this list has been made as comprehensive as possible.

Although significant effort has been made to ensure that there are no errors in this review sheet, some might nevertheless appear. The textbook and the course lecture notes and supplemental readings are the final authority in all factual matters, unless errors have been specifically identified there. You are ultimately responsible for obtaining accurate information when preparing for the exam.

---

Solution of systems of linear algebraic equations in matrix form

- rank of matrix = max. no. of linearly independent row or column vectors
- max. rank = $\min\{M, N\}$, $M$ = no. rows and $N$ = no. of columns
- rules of linear algebra; interactions between matrices, vectors, and scalars; dimensions
- matrix transpose and its properties
- solution of $A\mathbf{x} = \mathbf{b}$; for an $N \times N$ system, the following statements are equivalent:
    - $A\mathbf{x} = \mathbf{b}$ has a unique solution
    - $A$ has a unique inverse ($A^{-1}$)
    - $A$ is non-singular
    - $A$ has full rank (i.e., rank($A$) = $N$)
    - $\det(A) = |\mathrm{A}| \neq 0$
- solution using element row operations (EROs) to obtain row echelon form or reduced row echelon form

- consistency
    - consistent = one solution or infinitely many solutions (2-D analogy: solution space is a point or line)
    - inconsistent = no solution (2-D analogy: parallel lines – no intersecting point)
- solvability of $M \times N$ systems (see Fig. 8.3.1 of Zill, 6th ed.)
    - $M$ = no. of equations; $N$ = no. of unknowns
    - if $r = \text{rank}(A) = \text{rank}(A|\mathbf{b})$, system is consistent & a solution is possible
        - $r = N$: unique solution; typically, a square system
        - $r < N$: infinitely many solutions; typically, an underdetermined system
    - if $r = \text{rank}(A) < \text{rank}(A|\mathbf{b})$, system is inconsistent & no solution is possible (but a "best" fit is possible); typically, an overdetermined system

Curve-fitting and the method of least squares

- data set: $(x_i, y_i)$, $i = 1$ to $M$ → data vectors $\mathbf{x}$ and $\mathbf{y}$
- model: a set of functions $\{f_j(x)\}_{j = 1 \text{ to } N}$ and coefficients $\{c_j\}_{j = 1 \text{ to } N}$ that yield the best approximation to $\{y_i\}_{i = 1 \text{ to } M}$ (data set) or $y(x)$ (continuous function):

  $y(x_i) \approx \hat{y}(x_i) = \sum_{j=1}^{N} c_j f_j(x_i) \quad \rightarrow \quad \hat{\mathbf{y}} = F\mathbf{c}$, where $F_{ij} = f_j(x_i)$; $\hat{\mathbf{y}}$ contains best fit; vector $\mathbf{c}$ contains coefficients
- basis functions $\{f_j(x)\}$ are usually elementary functions like polynomials (including constant, linear, quadratic, and cubic), sin/cos, exponentials, logarithms
- seemingly nonlinear models can sometimes be transformed into linear models via changes of variables and/or algebraic manipulation
- least squares approach
    - minimize residual vector: $\mathbf{r} = \mathbf{y} - \hat{\mathbf{y}}$ → minimize $E = |F\mathbf{c} - \mathbf{y}|^2$ by setting $\partial E/\partial c_k = 0$ for all $k = 1$ to $N$
    - minimize $|\mathbf{r}|^2 = \mathbf{r}^T\mathbf{r}$ or make residual orthogonal to approximation ($\mathbf{r}^T\hat{\mathbf{y}} = 0$)
- basic normal equation

  $F^T F\mathbf{c} = F^T\mathbf{y} \quad \rightarrow \quad \mathbf{c} = \left(F^T F\right)^{-1} F^T\mathbf{y}$,
- practical considerations:
    - In *Matlab*, can write `c = F\y`; automatically forms solution using normal equation (or its equivalent)
    - $F^T F$ is symmetric and nonsingular if there are no repeated data points
    - $F$ is $M \times N$, so $F^T F$ is $N \times N$
- constrained LS optimization
    - many constraints are possible; need to express them in linear algebraic form
    - one approach: suppress magnitudes of coefficients
        - cost function: $E = |F\mathbf{c} - \mathbf{y}|^2 + \gamma\mathbf{c}^T\mathbf{c}$
        - modified normal equation: $\mathbf{c} = \left(F^T F + \gamma I_N\right)^{-1} F^T\mathbf{y}$
    - another approach: suppress magnitudes of second finite differences of coefficients
        - cost function: $E = |F\mathbf{c} - \mathbf{y}|^2 + \gamma\mathbf{c}^T H\mathbf{c}$

        - $H = K^T K$, where $K = \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & \cdots & \\ 0 & 1 & -2 & 1 & & \vdots \\ \vdots & & & \ddots & & \\ 0 & \cdots & 0 & 1 & -2 & 1 \\ 0 & \cdots & 0 & 0 & 0 & 0 \end{bmatrix}$
        - modified normal equation: $\mathbf{c} = \left(F^T F + \gamma H\right)^{-1} F^T \mathbf{y}$
    - adjust value of Lagrange multiplier $\gamma$ until coefficients vary smoothly and have reasonable magnitudes

Eigenvalues and eigenvectors for an $N \times N$ matrix $A$

- basic form: $A\mathbf{x} = \lambda\mathbf{x}$
- geometric interpretation: For $A\mathbf{x} = \mathbf{b}$, if $\mathbf{x}$ is an eigenvector, then $\mathbf{b}$ is a stretched or shrunk and possibly flipped but not rotated version of $\mathbf{x}$
- computing eigenvalues: $\det\left(A - \lambda I\right) = 0$
    - eigenvalues are roots of $N$th-order polynomial
    - special case: If $N = 2$, then $\lambda^2 - \mathrm{tr}\left(A\right) + \det\left(A\right) = 0$, where tr($A$) = trace of $A$, the sum of the entries on the main diagonal
    - eigenvalues can be repeated, and they can be complex
    - eigenvalues of upper and lower-triangular matrices are the diagonal values
    - eigenvalues of diagonal matrices are the diagonal values
- eigensystem properties
    - if $A$ is square with real entries, then any complex eigenvalues/eigenvectors come in conjugate pairs
    - if $A$ is square, then 0 is an eigenvalue iff (if and only if) $A$ is singular
    - if $A$ is nonsingular and $\lambda$ is an eigenvalue, then $1/\lambda$ is an eigenvalue of $A^{-1}$; both eigenvalues have the same corresponding eigenvectors
    - simple eigensystem has $N$ distinct eigenvalues, $N$ linearly independent eigenvectors
    - $A\mathbf{x}_i = \lambda\mathbf{x}_i \quad \rightarrow \quad AX = X\Lambda$ (horizontal augmentation of column vectors)
    - symmetric matrices behave well
        - eigenvalues of nonsingular symmetric matrices are nonzero and real
        - eigenvectors for distinct eigenvalues are orthogonal
        - repeated eigenvalues also have linearly independent (LI) eigenvectors (LI ≠ orthogonal); however, the eigenvectors *could* be orthogonal; it is always possible to find orthogonal eigenvectors with enough effort (see end of Sec. 8.10)
        - symmetric matrices can be singular and therefore have at least one zero eigenvalue; even so, the properties above still apply
    - orthogonal matrices ( $A^{-1} = A^T$ , which implies that $A^T A = I$ )
        - $A$ is orthogonal iff its columns form an orthonormal set (orthonormal is orthogonal with each vector having a length of 1; i.e., $|\mathbf{x}| = \mathbf{x}^T\mathbf{x} = 1$)
        - orthogonal matrices are not usually symmetric (The only orthogonal *and* symmetric matrix is $I$ because a matrix that is both satisfies $A^{-1} = A^T = A$.)
        - if eigenvectors are orthogonal, a matrix formed with eigenvectors as columns is orthogonal and has the property $X^{-1} = X^T$
    - linearly independent ≠ orthogonal

- diagonalization
    - o express $N \times N$ matrix $A$ as $A = PDP^{-1}$, where $D$ is a diagonal matrix
    - o $A$ does not have to be symmetric or orthogonal to be diagonalizable
    - o Theorem: An $N \times N$ matrix $A$ is diagonalizable iff $A$ has $N$ LI eigenvectors
    - o Theorem: If $N \times N$ matrix $A$ has $N$ distinct eigenvalues then it is diagonalizable (but it might be diagonalizable even if the eigenvalues are not distinct, that is, if some are repeated)
    - o Theorem: An $N \times N$ matrix $A$ can be orthogonally diagonalized iff $A$ is symmetric (*orthogonal diagonalization* means that $P$ is orthogonal)
    - o many ways to diagonalize a matrix
    - o one important example: Given an $N \times N$ matrix $A$ with $N$ LI eigenvectors (i.e., $A$ is full rank), since $A\mathbf{x}_1 = \lambda_1\mathbf{x}_1$, $A\mathbf{x}_1 = \lambda_1\mathbf{x}_1$, etc., then, using the definitions

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_N \end{bmatrix} \quad \text{and} \quad X = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_N \\ \downarrow & \downarrow & & \downarrow \end{bmatrix},$$

where $X$ is formed by arranging the eigenvectors in adjacent columns, we obtain $AX = X\Lambda \rightarrow \ A = X\Lambda X^{-1}$. Thus, an $N \times N$ matrix $A$ with LI eigenvectors can be diagonalized into its eigenvalues and eigenvectors.

Some basic factorizations

- LU factorization
    - o express $N \times N$ matrix $A$ as $A = LU$ , where $L$ is a lower triangular matrix and $U$ is upper triangular
    - o One application is solution of $A\mathbf{x} = \mathbf{b}$:  
      $A\mathbf{x} = \mathbf{b} \rightarrow LU\mathbf{x} = \mathbf{b}$. Let $U\mathbf{x} = \mathbf{y} \rightarrow L\mathbf{y} = \mathbf{b}$. Solve $L\mathbf{y} = \mathbf{b}$ for $\mathbf{y}$ and then $U\mathbf{x} = \mathbf{y}$ for $\mathbf{x}$. Both solutions are straightforward using forward and backward substitution.
    - o LU factorizations are not typically unique; results depend on method used
    - o *Matlab* command `[L, U] = lu(A)`
    - o determinants: $\det(A) = \det(L)\det(U)$
- *QR* factorization
    - o *LU* factorization is for square systems; *QR* factorization is for over- and underdetermined systems ($M > N$ or $M < N$). (It also works for square systems, but it is computational overkill.)
    - o for overdetermined ($M > N$) systems, express $M \times N$ matrix $A$ in the product form $A = QR$, where $Q$ is an $M \times M$ orthogonal matrix and $R$ is an $M \times N$ upper triangular matrix
    - o only the first $N$ columns of $Q$ are actually needed
    - o to solve a system $A\mathbf{x} = \mathbf{b}$:  
      $A\mathbf{x} = \mathbf{b} \rightarrow QR\mathbf{x} = \mathbf{b}$. Let $R\mathbf{x} = \mathbf{z} \rightarrow Q\mathbf{z} = \mathbf{b}$. Solve $Q\mathbf{z} = \mathbf{b}$ for $\mathbf{z}$ and then $R\mathbf{x} = \mathbf{z}$ for $\mathbf{x}$ using backward substitution. Matrix $Q$ is orthogonal, so $Q^{-1} = Q^T$. Thus, $\mathbf{z} = Q^T\mathbf{b}$.
    - o *Matlab* command `[Q, R] = qr(A)`

Singular value decomposition (SVD)

- using “economy-sized” or “thin” decomposition, an $M \times N$ matrix $A$ can be expressed in the product form $A = U\Sigma V^T$, where $U$ is an $M \times N$ column-orthogonal matrix, $\Sigma$ (sometimes labeled $S$) is an $N \times N$ diagonal matrix, and $V$ is an $N \times N$ orthogonal matrix:

$$U = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_N \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}_{M\times N} \qquad \Sigma = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \sigma_N \end{bmatrix}_{N\times N} \qquad V = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_N \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}_{N\times N}$$

  where $\mathbf{u}_i^T\mathbf{u}_j = \delta_{ij}$ (orthogonal column vectors) and $\mathbf{v}_i^T\mathbf{v}_j = \delta_{ij}$ (orthogonal column vectors), and where $\delta_{ij} = 1$ if $i = j$ and 0 if $i \neq j$
- the above assumes $M > N$ or $M = N$; the $M < N$ case requires a little more care
- in full SVD, $U$ is $M \times M$ and $\Sigma$ is $M \times N$, but parts of $U$ and $\Sigma$ are not necessary for non-square systems, hence the “economy-sized” decomposition
- product form $A = U\Sigma V^T$ still valid in full SVD
- *Matlab* command (full SVD unless ‘econ’ option is added): `[U, S, V] = svd(A)`
- diagonal elements of $\Sigma$ are called *singular values*; always real and either positive or zero, even if $A$ has complex entries, and can repeat
  $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \ldots \geq \sigma_N$
- zero singular values, if any, occupy the highest index numbers (i.e., up to and including $\sigma_N$)
- if $A$ is symmetric, then $A^TA = AA^T = A^2$, so $\lambda_i^2 = \sigma_i^2 \quad \rightarrow \quad |\lambda_i| = |\sigma_i|$ (sign ambiguity)
- implications and properties:
  - condition number of a matrix $A$: cond($A$) = $\sigma_1/\sigma_N$; figure of merit that indicates how close the matrix is to being singular; it could be “numerically singular” because of finite precision of stored numbers in computer
  - $A = U\Sigma V^T = \sum_{j=1}^{N} \sigma_j \mathbf{u}_j \mathbf{v}_j^T$, where $\mathbf{u}_j\mathbf{v}_j^T$ is an $M \times N$ outer product; weights are singular values ($\sigma_j$) and grow smaller with increasing $j$. Also, rank($\mathbf{u}_j\mathbf{v}_j^T$) = 1.
  - $U$ and $V$ are both orthogonal ($U^{-1} = U^T$ and $V^{-1} = V^T$); thus,
    - $A = U\Sigma V^T \quad \rightarrow \quad AV = U\Sigma \quad \rightarrow \quad A\mathbf{v}_i = \sigma_i \mathbf{u}_i$
    - $A = U\Sigma V^T \quad \rightarrow \quad A^T = V\Sigma^T U^T \quad \rightarrow \quad A^TU = V\Sigma \quad \rightarrow \quad A^T\mathbf{u}_i = \sigma_i \mathbf{v}_i$
    - $A^TA = \left(U\Sigma V^T\right)^T \left(U\Sigma V^T\right) = V\Sigma^T U^T U\Sigma V^T = V\Sigma^T\Sigma V^T$

      $\rightarrow \quad \left(A^TA\right)V = V\Sigma^T\Sigma \quad \rightarrow \quad \left(A^TA\right)\mathbf{v}_i = \sigma_i^2\mathbf{v}_i$

      $\sigma_i^2$ are the eigenvalues of $A^TA$, and $\{\mathbf{v}_i\}_{i\,=\,1\text{ to }N}$ are the eigenvectors; one way to find singular values and **v** eigenvectors
    - $AA^T = \left(U\Sigma V^T\right)\left(U\Sigma V^T\right)^T = U\Sigma V^T V\Sigma^T U^T = U\Sigma\Sigma^T U^T$

      $\rightarrow \quad \left(AA^T\right)U = U\Sigma^T\Sigma \quad \rightarrow \quad \left(AA^T\right)\mathbf{u}_i = \sigma_i^2\mathbf{u}_i$

      $\sigma_i^2$ are also the eigenvalues of $AA^T$, and $\{\mathbf{u}_i\}_{i\,=\,1\text{ to }N}$ are the eigenvectors; one way to find singular values and **u** eigenvectors

- to solve a system $A\mathbf{x} = \mathbf{b}$ (for overdetermined and square systems; underdetermined requires interpretation):
  $$U\Sigma V^T\mathbf{x} = \mathbf{b} \quad \rightarrow \quad \Sigma V^T\mathbf{x} = U^T\mathbf{b} \quad \rightarrow \quad V^T\mathbf{x} = \Sigma^{-1}U^T\mathbf{b} \quad \rightarrow \quad \mathbf{x} = V\Sigma^{-1}U^T\mathbf{b}$$
- for overdetermined curve-fitting
  - $\mathbf{c} = V\Sigma^{-1}U^T\mathbf{y}$ yields same coefficients as unconstrained LS if all singular values are used
  - equivalent to
    $$\mathbf{c} = \sum_{j=1}^{N}\left(\frac{\mathbf{u}_j^T\mathbf{y}}{\sigma_j}\right)\mathbf{v}_j$$
  - to smooth oscillating coefficient values, set entries in $\Sigma^{-1}$ matrix corresponding to small singular values (i.e., below a certain threshold) to zero; reduces impact of eigenvectors that do not model data vector **y** very well

Ordinary differential equations (ODEs) – general properties and theorems

- linear differential equations have the form
  $$a_n(x)\frac{d^n y}{dx^n}(x) + a_{n-1}(x)\frac{d^{n-1}y}{dx^{n-1}} + \cdots + a_0(x)y = g(x)$$
  coefficients can be constant or variable; if variable, they are functions of indep. variable
- important special case is the 2nd order linear equation
- distinction between homogeneous and inhomogeneous ODEs; does $g(x) = 0$?
- distinction between initial value problems (IVPs) and boundary value problems (BVPs)
  - IVP: dependent variable and/or its derivatives defined at one point
  - IVP: always have unique solutions
  - BVP: dependent variable and/or its derivatives defined at two or more points
  - BVP: can have no solution, a unique solution, or many solutions
  - BVPs are often eigenvalue problems (EVPs) as well
- superposition principle; a linear combination of solutions to a homogeneous ODE over an interval is also a solution over the same interval
- corollaries:
  - a constant multiple of a solution is also a solution
  - a homogeneous ODE always possess the trivial solution $y = 0$
- linearly independent vs. linearly dependent solutions (analogy to vectors)
- $N$th order homogeneous linear ODE has a fundamental set of $N$ linear indep. solutions
- nonhomogeneous ODEs: general solution = complementary solution + particular solution (complementary solution is the full solution set of the corresponding homogenous ODE)
- superposition also applies to particular solutions: If $y_{p1}$ is a solution of the ODE with $g_1(x)$, $y_{p2}$ is a solution with $g_2(x)$, etc., then $y_{p1} + y_{p2} + \ldots$ is a solution to ODE with $g(x) = g_1(x) + g_1(x) + \ldots$

Fourier and modified Fourier equations and their suggested solutions

- Fourier equation: $y'' + a^2y = 0$
  suggested solution for closed boundaries: $y(x) = c_1\cos(ax) + c_2\sin(ax)$
- modified Fourier equation: $y'' - a^2y = 0$
  suggested solution for closed boundaries: $y(x) = c_1\cosh(ax) + c_2\sinh(ax)$
- Fourier equation w/homogeneous boundary conditions (BCs) are eigenvalue problems (EVPs)
- modified Fourier equation w/homogeneous BCs has only the trivial solution
- Fourier equation and modified Fourier equation w/nonhomogeneous BCs can have nontrivial solutions

Relevant course material:

| | |
|---|---|
| Homework: | #1, #2, and #3 |
| Labs: | #1 through #5 |
| Reading: | Assignments from Aug. 26 through Oct. 4, including the supplemental readings:<br>“Constrained Least-Squares Optimization Using Minimized Coefficient Magnitudes”<br>Mathworks Help Center documentation on factorizations<br>SVD readings from *Numerical Recipes in C*, 2nd ed.<br>“The Fundamental Theorem of Linear Algebra” (optional) |

This exam will focus primarily on the course outcomes listed below, although only the parts of the third outcome relevant to linear algebra will be covered.

- Review basic concepts of linear algebra
- Apply linear-algebraic methods to the solution of applied problems
- Develop the concept of an eigenvalue problem in both linear algebra and differential equations

The course outcomes are listed on the course syllabus, which was distributed at the beginning of the semester and is available on the Syllabus and Policies page at the course web site. The outcomes are also listed on the Course Description page at the web site. Note, however, that some topics not directly related to the course outcomes could be covered on the exam as well.