CSci 5304 Fall 2018
Computational Aspects of Matrix Theory

General Information

This course introduces the basic numerical techniques of linear algebra. The emphasis of the course is on the design of these algorithms and their analysis, with some coverage of related applications to motivate and support the material. Students taking this class should have a good background in linear algebra (prerequisite is csci 2033 or equivalent) and be familiar with Matlab.

- **Class Schedule:** MW 08:15 AM - 09:30 AM — Keller Hall 3-115.
- **Instructor:** Yousef Saad [≪ saad@cs.umn.edu ≫] `http://www.cs.umn.edu/~saad`
  Office: Keller H. 5 -225B – Office Phone: (612) 624 – 7804.
- **Teaching Assistant:** – Sijie He ≪ hexxx893@umn.edu ≫
- **Office hours:** posted on the class web-site
- **Class Website** The main class web-site is
  `www-users.cselabs.umn.edu/classes/Fall-2018/csci5304/`
  We will use canvas **only for posting grades**.

Textbook

There is no *required* textbook for this class. However, you will need a good reference for an in-depth coverage of the material that will be seen. Here are a few listed in order of preference.

- Main reference: *Numerical linear algebra*, Lloyd N. Trefethen and David Bau, III. SIAM, 1997 (pbk). Very well written, easy to understand and insightful presentation of most topic to be covered.
- *Matrix Computations* 4th edition, G. Golub and C. Van Loan. John Hopkins, 2015. This is a much more comprehensive book than Trefethen and Bau. It is recommended as a reference for those of you who will do research involving numerical linear algebra.
- *Numerical methods in Matrix Computations* Åke Björk, Springer Verlag, 2015. ISBN: 978 3 319 05088-1; ISBN: 978 3 319 05088-8 (eBook).
- *Numerical Analysis, a second course*, J. M. Ortega, SIAM (Classics in applied mathematics), 1990.

Matlab references: Matlab will often be used for writing short programs (in particular for homeworks). Matlab has extensive online documentation and there are many resources posted on the web, so a manual is not really needed unless you have never used matlab before in which case it is recommended to get a reference manual. Here are a couple that I found quite good:

- *" Matlab, Third Edition: A Practical Introduction to Programming and Problem Solving 3rd Edition"* by Stormy Attaway. (2013) Publisher: Elsevier, ISBN-13: 978-0124058767 ISBN-10: 0124058760
- *"Mastering Matlab"* by Duane Hanselman and Bruce Littlefield. Prentice Hall (2011) ISBN-13: 978-0136013303 ISBN-10: 0136013309.

Lecture Notes

Lecture notes will be posted regularly on the class web-site (see above – not on canvas). Click on the "Lect. Notes" icon in the menu. These may include material not available in the text-book. These notes will be posted by topic rather than lecture by lecture, and they are usually posted prior to the lectures. Also note that this class is on Unite so you will have access to streaming video archives of class meetings on a TEN-DAY delay (7-DAY prior to exams) for the length of the semester. See details in the 'announcements' page of the class web-site.

Evaluation

Your evaluation for this class will be based on 6 homeworks, 3 mid-terms, and a final exam. The final score will consist of

- Homework total: 30 % for a total of 6 homeworks (5% each).
- Exam total: 70%. This is $3 \times 23\frac{1}{3}\%$, from 3 best scores out of Midterm Exam 1, Midterm Exam 2, Midterm Exam 3, and Final exam.

A few details follow.
Exams: only your best 3 exam grades out of the 4 exams (3 midterms and a final) will be used to compute your final score and these will contribute 70% to this score (23.33% each).
Homeworks: The remaining 30% of the score will come from the 6 homeworks, at 5% each.
The final is not mandatory: Based on the 3 mid-terms (and all 6 homeworks) you will be given a provisional letter grade at the very last day of class (earlier if possible). If you are satisfied with this grade as a final grade for the class you need not take the final. If not, you may take the final in which case, the best 3 of the 4 exam grades you received will be used (weight: 23.33% each). There will be no make-up exams: if you miss any one of the 4 exams then you still have 3 other exams that can be used to calculate the full exam part (70%) of the score.
I may include a few bonus points (max of 3 points to be added to the final score) for a few students who show very active participation in class. There may be a few pop-quizzes with the goal of improving class participation and discussions. These will not be graded. Final grades will be decided based on the following scale, where T is the total score (out of 100) you achieved in the class.

A : $100 \geq T \geq 93$	**A-** : $93 > T \geq 87$	**B+** : $87 > T \geq 82$
B : $82 > T \geq 77$	**B-** : $77 > T \geq 72$	**C+** : $72 > T \geq 65$
C : $65 > T \geq 60$	**C-** : $60 > T \geq 55$	**D+** : $55 > T \geq 50$
D : $50 > T \geq 40$	**F** : $40 > T$	

If you are taking the class on an S-N basis your total score must be at least 60% in order to get an S grade.

Grading

Grades will be posted immediatly after each homework or exam is graded. This will usually take about one week. It is important that you check your grades regularly. If you see a discrepancy between your grades and the grades posted, you need to alert the TA immediatly. You have one week after the homework/ exam is returned for requesting a change. Details on this can be found in the general **policy on homeworks and exams** which is posted in the class web-site.

Cheating

All homeworks labs, and exams, must represent your own individual effort. Cheating cases will be dealt with in a very strict manner. At a minimum, violators of this policy will fail the course and will have their names recorded. For additional information please consult the student code of conduct which can be found here: `https://regents.umn.edu/policies/index`

Overview of topics to be covered (tentative)

- Background: Subspaces, Bases, Orthogonality, Matrices, Projectors, Norms. Floating point arithmetic. Introduction to Matlab.
- Systems of linear equations. Solution of Systems of Linear Equations: matrix LU factorization. Special matrices: symmetric positive definite, banded.
- Error analysis, condition numbers, operation counts, estimating accuracy.
- Orthogonality, the Gram-Schmidt process. Classical and modified Gram-Schmidt. Householder QR factorization. Givens rotations. Least-squares systems. Rank deficient LS problem.
- More on least squares problems. Regularization, Least squares problems with constraints.
- Eigenvalues, singular values. The Singular Value Decomposition. Applications of the SVD.
- Eigenvalue problems: Background, Schur decomposition, perturbation analysis, power and inverse power methods, subspace iteration; the QR algorithm.
- The Symmetric Eigenvalue Problem: special properties and perturbation theory, Law of inertia, Min-Max theorem, symmetric QR algorithm, Jacobi method. Applications.
- Sparse matrix techniques. The Lanczos algorithm. Lanczos bidiagonalization. Sparse direct solution methods (overview). Krylov subspace methods for linear systems (overview).