ECE 586: Vector Space Methods
Course Introduction Video

Henry D. Pfister
Duke University
Overview

- Engineering uses science and mathematics to **invent, design, and build** things that solve problems and achieve practical goals.

- From a mathematical perspective, engineering models the world using **vectors** and analyzes it using logic, linear algebra, and optimization.

- This class is essentially an applied math class designed for graduate engineers in **data science, signal processing, and quantum computing**.

- Its goal is to **refresh and extend your undergraduate knowledge** of linear systems, vector spaces, matrices, and optimization.
Major Topics

- Logic and Set Theory
- Metric Spaces and Topology
  - Practical applications include the analysis of iterative algorithms
- Linear Algebra: Normed and Inner-Product Spaces
  - Practical applications include spaces of functions and Markov chains
- Approximation and Projection
  - Practical applications include function approximation and machine learning
- The Four Fundamental Subspaces and Singular Value Decomposition
  - Practical applications include dimensionality reduction
- Optimization in Vector Spaces
  - Practical applications duality bounds in convex optimization
Example Problem 1

- Approximation Error in Linear Systems
  - Suppose you know a vector $b$ satisfies $Ax = b$ for an invertible matrix $A$.
  - To compute $x$, you could multiply by $A^{-1}$ to get $A^{-1}b = A^{-1}(Ax) = x$.
  - But, what if $A$ is not known perfectly?
  - Let’s assume we know $\hat{A} = A + E$ where the error matrix $E$ is “small”.
  - Now, we can compute $\hat{x} = \hat{A}^{-1}b$. But, what can we say about $x - \hat{x}$?

- The mathematical study of “closeness” is called Topology.
  - To answer the above question, one can define distances between objects.
  - Using these distances, we can define a constant $c(\hat{A})$ such that
    \[
    \text{distance}(x, \hat{x}) \leq c(\hat{A}) \text{ distance}(A, \hat{A}) = c(\hat{A}) \text{ length}(E)
    \]
Example Problem 2

Approximation of Functions by Simpler Functions

- Suppose you want to find a 3rd-order polynomial that gives a “good” approximation of a function $f(x)$ on the interval $[0, 1]$.

- A key question is, “What do we mean by good?” max error? avg error?

- A standard approach is to minimize the mean squared error and this gives

$$\min_{a_0, a_1, a_2, a_3} \int_0^1 \left[ f(x) - (a_0 + a_1 x + a_2 x^2 + a_3 x^3) \right]^2 dx$$

- Given $f(x)$ and the ability to integrate, one can solve this via brute force.

- This course will show there is also an simple intuitive approach.

Solution via Orthogonality and Linear Algebra

- Compute $a_0, a_1, a_2, a_3$ using 4 linear equations ($k = 0, 1, 2, 3$) given by

$$\int_0^1 (a_0 + a_1 x + a_2 x^2 + a_3 x^3)x^k dx = \int_0^1 f(x)x^k dx.$$
Example Problem 3

- Convex Optimization
  - Consider a real function \( f(x) \) on the real interval \([a, b]\)
  - A chord of \( f \) is a line connecting \((x, f(x))\) and \((y, f(y))\)
  - \( f(x) \) is convex if all chords lie above the function, i.e. for \( \lambda \in [0, 1] \),
    \[
    f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)
    \]
  - Convexity implies all local minima have the same minimum value

Q: Can we compute a simple lower bound on the minimum value?

- Solution via Tangent Line
  - Yes, a convex function also lies above any tangent line and
    \[
    \min_{x \in [a, b]} f(x) \geq \min_{x \in [a, b]} \left[ f(x_0) + (x - x_0)f'(x_0) \right]
    \]
Course Overview and Philosophy

- **First Theme: Standard definitions and vocabulary**
  - Engineering has many subfields where new ideas are described using abstract mathematics.
  - A key goal of this course is to teach the vocabulary necessary to read and understand papers and textbooks in these subfields.

- **Second Theme: Standard theorems and their proof**
  - Our focus is on mathematical foundations of engineering analysis.
  - Proofs illustrate mathematical reasoning and show why results are true.
  - Emphasis is on general techniques that extend to advanced problems.

- **Assignments**
  - Written homework builds knowledge of definitions and techniques.
  - Computer assignments demonstrate practical applications.
  - These are like training for a sport; each workout provides a small gain but consistent effort leads to significant improvement over time.
Learning Strategies

Course Elements

- Syllabus: Lists lecture/video topics by date along with HW
- Course Notes: Detailed description of the course content
- Flip Videos: Slide-based introduction to course notes 😊
- Lectures: Q/A about videos/notes + group work on example problems

My Recommendations

- Watch the flip video first and then read the matching course notes
- Afterward, try the homework while referring to the course notes as needed
- If you’re stuck >30min on a question, ask for a hint (via piazza or e-mail)
- Visit office hours to get additional help as needed
- Make sure to watch the video before the class meeting!