

Course Introduction

MATH-151: Mathematical Algorithms in Matlab

August 21, 2023



ALGORITHMS

- Who here would say they think computers are smart?

ALGORITHMS

- Who here would say they think computers are smart?
- At base level, they are actually pretty limited.
 - Receive, store, and output data
 - Add or multiply two numbers

ALGORITHMS

- Who here would say they think computers are smart?
- At base level, they are actually pretty limited.
 - Receive, store, and output data
 - Add or multiply two numbers
- But they are very good at following directions! By being clever, we can tell the computer to do these operations in very specific ways to achieve bigger goals! These are called algorithms.

ALGORITHMS

- Who here would say they think computers are smart?
- At base level, they are actually pretty limited.
 - Receive, store, and output data
 - Add or multiply two numbers
- But they are very good at following directions! By being clever, we can tell the computer to do these operations in very specific ways to achieve bigger goals! These are called algorithms.
- We will use algorithms to make computers do the work for us.



MATHEMATICAL ALGORITHMS IN MATLAB

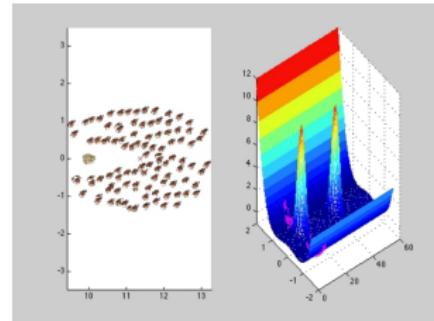
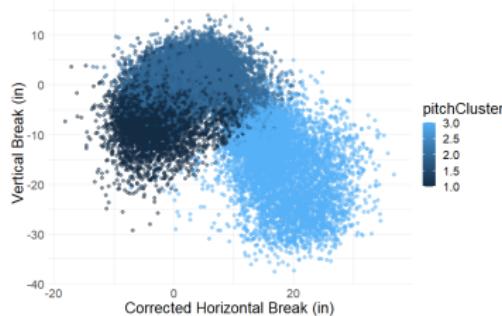
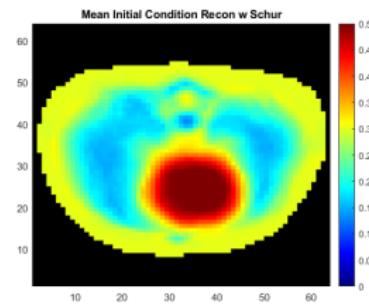
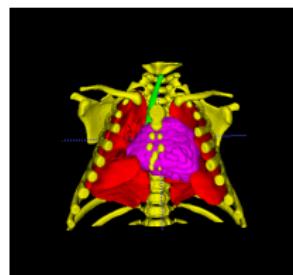
In this course students will be introduced to common algorithms used to numerically solve problems as well as learn how to implement them in the Matlab programming language.

[Click here to install Matlab](#)

Upon completion of this course, the student is expected to have the following skills:

- Ability to write and run common algorithms in Matlab, with coding familiarity transferable to other languages
- Effective code commenting and formatting, to ensure code is readable to others
- Understanding of algorithm design and how a computer “thinks”
- Debug code to identify and fix mistakes.

APPLICATIONS



WHO AM I?

- Education
 - PhD Student. Mathematics, CSU, Current
 - M.S. Applied Mathematics, UMass-Amherst, 2014
 - B.S. Applied Mathematics, RPI, 2011
- Work and Teaching
 - Previously worked as a Sr. Systems Engineer
 - Taught on the side for 4 years

WHO AM I?

- Education
 - PhD Student. Mathematics, CSU, Current
 - M.S. Applied Mathematics, UMass-Amherst, 2014
 - B.S. Applied Mathematics, RPI, 2011
- Work and Teaching
 - Previously worked as a Sr. Systems Engineer
 - Taught on the side for 4 years
- Interests and Hobbies
 - Baseball
 - Tabletop and video games



WHO AM I?

- Education
 - PhD Student. Mathematics, CSU, Current
 - M.S. Applied Mathematics, UMass-Amherst, 2014
 - B.S. Applied Mathematics, RPI, 2011
- Work and Teaching
 - Previously worked as a Sr. Systems Engineer
 - Taught on the side for 4 years
- Interests and Hobbies
 - Baseball
 - Tabletop and video games
 - I can go pretty fast on rollerblades ...



MY TEACHING PHILOSOPHY

- The best I can do as an instructor is to expose the class to course materials and help students gain familiarity with the topics.
- Learning occurs when students **engage with the material.**

MY TEACHING PHILOSOPHY

- The best I can do as an instructor is to expose the class to course materials and help students gain familiarity with the topics.
- Learning occurs when students **engage with the material.**
 - Actively following along with lectures.
 - Taking notes.
 - Thinking about material.
 - Asking questions.

MY TEACHING PHILOSOPHY

- The best I can do as an instructor is to expose the class to course materials and help students gain familiarity with the topics.
- Learning occurs when students **engage with the material.**
 - Actively following along with lectures.
 - Taking notes.
 - Thinking about material.
 - Asking questions.
 - Working through problems.
 - Doing labs with intent. Working in groups is highly recommended.
 - Understand the purpose of each step.
 - Discussing approaches and solutions to your peers.
 - Learn from your mistakes!

MY TEACHING PHILOSOPHY

- The best I can do as an instructor is to expose the class to course materials and help students gain familiarity with the topics.
- Learning occurs when students **engage with the material.**
 - Actively following along with lectures.
 - Taking notes.
 - Thinking about material.
 - Asking questions.
 - Working through problems.
 - Doing labs with intent. Working in groups is highly recommended.
 - Understand the purpose of each step.
 - Discussing approaches and solutions to your peers.
 - Learn from your mistakes!
- Don't be afraid to talk to me if you are struggling, I am very willing to help!

COURSE GRADING SCHEME

- **Lecture Journals (5%)**

- Short writing prompt assigned about each lecture
- Graded based on completion

COURSE GRADING SCHEME

- **Lecture Journals (5%)**

- Short writing prompt assigned about each lecture
- Graded based on completion

- **Labs (80%)**

- Hands-on coding problems assigned each week
- Due before Monday class the following week.

COURSE GRADING SCHEME

- **Lecture Journals (5%)**

- Short writing prompt assigned about each lecture
- Graded based on completion

- **Labs (80%)**

- Hands-on coding problems assigned each week
- Due before Monday class the following week.

- **Final Lab (15%)**

- Cumulative lab assignment connecting various algorithms seen in class

COURSE GRADING SCHEME

- **Lecture Journals (5%)**

- Short writing prompt assigned about each lecture
- Graded based on completion

- **Labs (80%)**

- Hands-on coding problems assigned each week
- Due before Monday class the following week.

- **Final Lab (15%)**

- Cumulative lab assignment connecting various algorithms seen in class
- Lab grading will be based on the “three C’s”,
 - **Completion:** Is the lab completed?
 - **Correctness:** Does the code perform as desired?
 - **Clarity:** Is the code well presented? Is the code commented?

TENTATIVE SCHEDULE

Dates	Monday	Wednesday	Friday
8/21 - 8/25	Course Introduction	General Coding Concepts	Matlab Introduction
8/28 - 9/1	Logic and Loops	Lab 1	Extra Lab Time
9/4 - 9/8	No Class!	Vectors and Plotting	Lab 2
9/11 - 9/15	Functions and Recursion	Lab 3	Extra Lab Time
9/18 - 9/22	Interpolation	Lab 4	Extra Lab Time
9/25 - 9/29	Numerical Integration	Lab 5	Extra Lab Time
10/2 - 10/6	Numerical Differentiation	Lab 6	Extra Lab Time
10/9 - 10/13	Nonlinear Solvers	Lab 7	Extra Lab Time
10/16 - 10/20	Differential Equation Solvers	Lab 8	Extra Lab Time
10/23 - 10/27	Final Exam	Final Exam	Final Exam

* Schedule subject to change as necessary.