CS 357 – Numerical Methods 1

Prof. Mariana Silva

SPRING 2021

CampusWire

- All communication will happen via CampusWire. NO EMAILS!

- Check it daily!

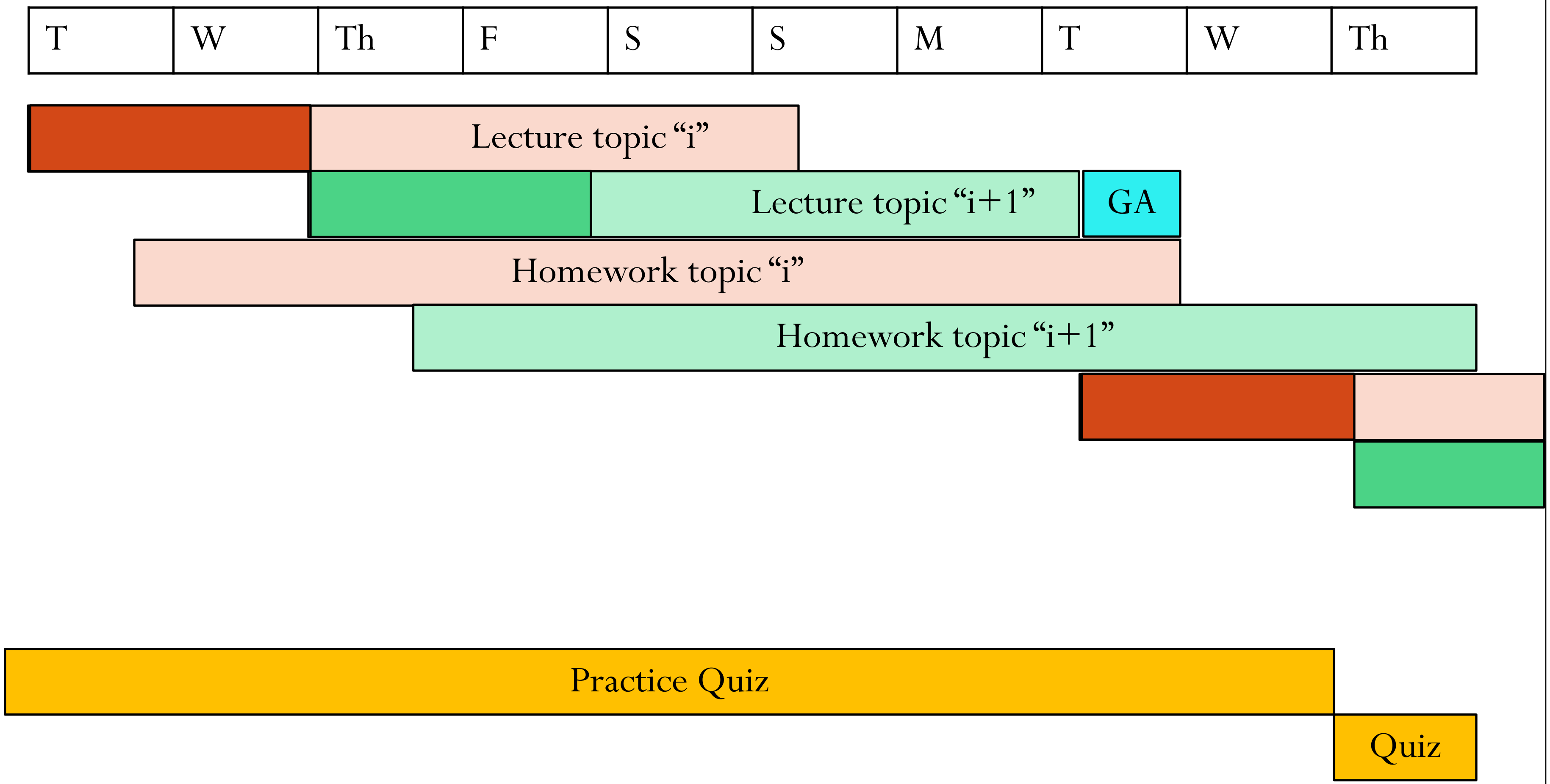
- Important course announcements will be pinned.

Course Website - Syllabus

PrairieLearn Learning Resources

- Lecture
- Workspaces
- Group Activity:
- Homework
- Machine Problem
- Practice Quiz

How should I study?



Collaborative Learning

- Complete weekly activity in groups (select one of these options)

  Tuesdays 2pm, 9am or 8pm

- Week 1 and 2: randomly assigned groups (only 2pm, attendance not required)
- Week 3-8: fixed groups
- Week 9-14: fixed groups
- Next class we will talk about group assignment.

Group Activity Grade

1) Attendance + Completion (9% of the grade):

GAPLScore (0-100%): this is the ”completion” grade attained by your group

Attendance: this is your individual attendance score (0% or 100% if you join the zoom link for at least 45 minutes)

Your final GA Score for each week is computed as:

GA# Score = Attendance * (40% + 0.6*GAPLScore)

Group Activity Grade

2) Participation (2% of the grade):

Manager: Create groups assessment, record roles, keep team on track

Recorder: Share screen and enter (most) answers

Reflector: Complete survey at the end of each activity

Each student will need to participate in each role twice in the semester (one in each fixed group cycle). Each participation counts as 100%.

Participation Score: Sum of all participation scores / 6

PrairieLearn Tour

Next class

- Course topics
- Use PL for group activity: trial run

Introduction and “Big Idea”

What are…

Numerical Methods ?

Numbers in a computer
(and how computer understands these numbers)

- Mathematical model
  - "algorithms" derived from math ideas to solve equations numerically
- Complexity of the problem
  - Slow vs fast
- Accuracy
  - Accurate vs inaccurate

Method = Math + Complexity + Accuracy

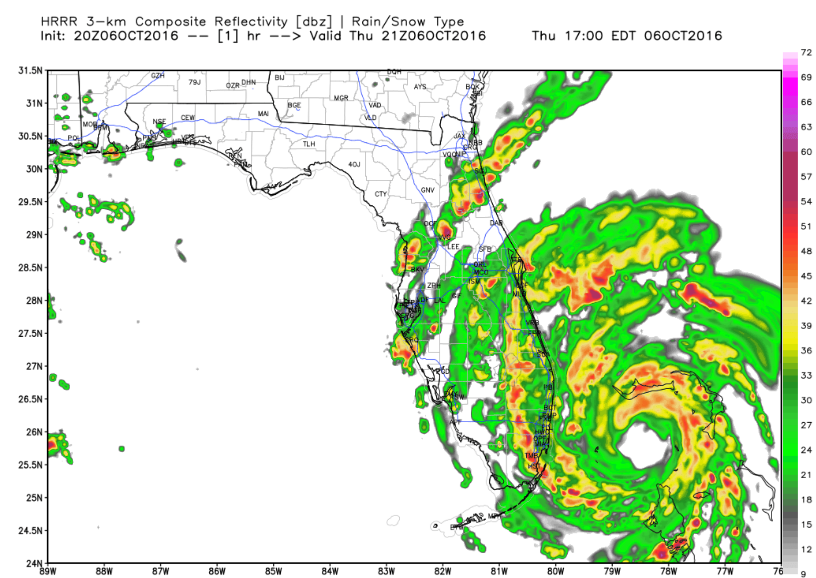
Why is this course important?

1. Understanding and reconstruction of known problems
   - Natural disasters
   - Catastrophic failures

2. Prediction of unknown situations
   - Weather conditions
   - Behavior of new materials

3. Optimization of existing problems
   - Image recognition
   - Reduce fabrication costs

Explosion of Ariane 5 in 1996



Goals for this course

- Understand how numbers are represented in the computer.
- When developing code, you will likely run into numerical errors. What are the sources of these errors?
- How can you avoid numerical errors?
- How can you choose a suitable algorithm for a given application?
- Use existing libraries to solve real applications.

(Numerical) **Method** = **Math** + Complexity + Accuracy

Mathematical model:

What equations can we use to represent our problem?

Accuracy:

Are we getting accurate results?

Why is the method not giving me the correct solution?

Complexity:

How long does it take to solve this problem?

Is it cost-effective?

Your entire CS 357 semester in a few slides!

Are you ready?

Accuracy

- Why a numerical method might not give the right answer?
  - Computers have finite representation of numbers
  - Sometimes the "right answer" cannot be represented in a finite way
  - Example:

$$\pi = 3.14159265358979323846264338327950288419 71\ldots$$

Demo: Waiting for the number 1

```
from time import sleep

x = 0.0

while x != 1.0:
    x += 0.1
    print(repr(x))

    sleep(0.1)
```

What is going to happen when we run this code?

A. Code will stop after printing 11 values for x
B. Code will stop after printing 10 values for x
C. Code will not stop
D. Code will not start

Monte Carlo Methods

Texas Holdem Game: we would like to determine the probability of winning of a given starting hand

Physical experiment
vs
Numerical experiment

Numerical Experiments

- What do we want to know about a numerical experiment?
  1. What questions are we attempting to answer?
  2. What is the outcome of the experiment?
  3. Is it repeatable?
  4. Is the answer accurate?
  5. How long will it take?

Time vs accuracy trade-off

Question: Is running this method (with a certain accuracy) a good use of our time and/or computer resources?

Complexity

How long does it take to solve a problem?

Given A, B matrices of size $m \times m$, the matrix-matrix multiplication $A \cdot B$ takes τ seconds.

How long does it take to perform $C \cdot D$, matrices of size $2m \times 2m$?

```
from time import process_time
import numpy as np
from time import process_time
```

```
n = 2000
A = np.random.randn(n,n)
B = np.random.randn(n,n)

t = process_time()  # store the time
C = A @ B
t = process_time() - t
print(t)
```

```
A = np.random.randn(2*n,2*n)
B = np.random.randn(2*n,2*n)

t2 = process_time()  # store the time
C = A @ B
t2 = process_time() - t2
print(t2)
```

Linear system of equations: Image processing

How can we use linear operators to create blurred images? How can we do the inverse process?

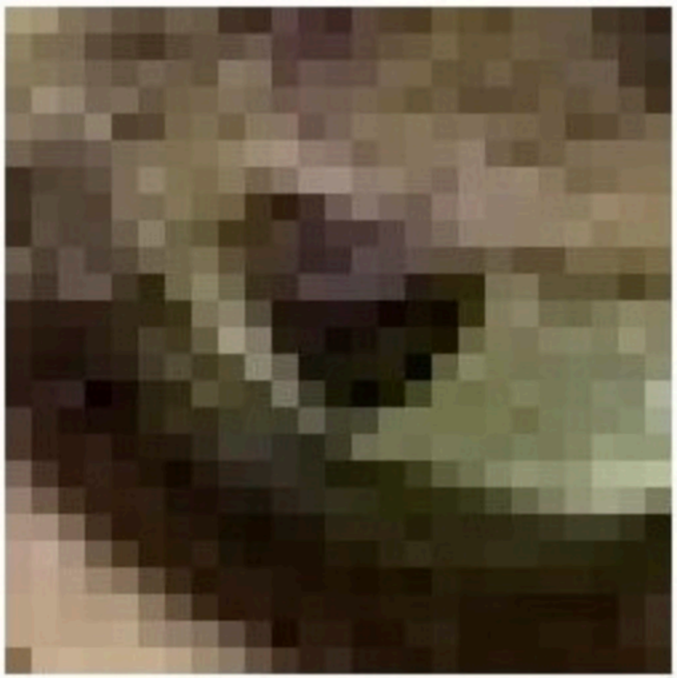
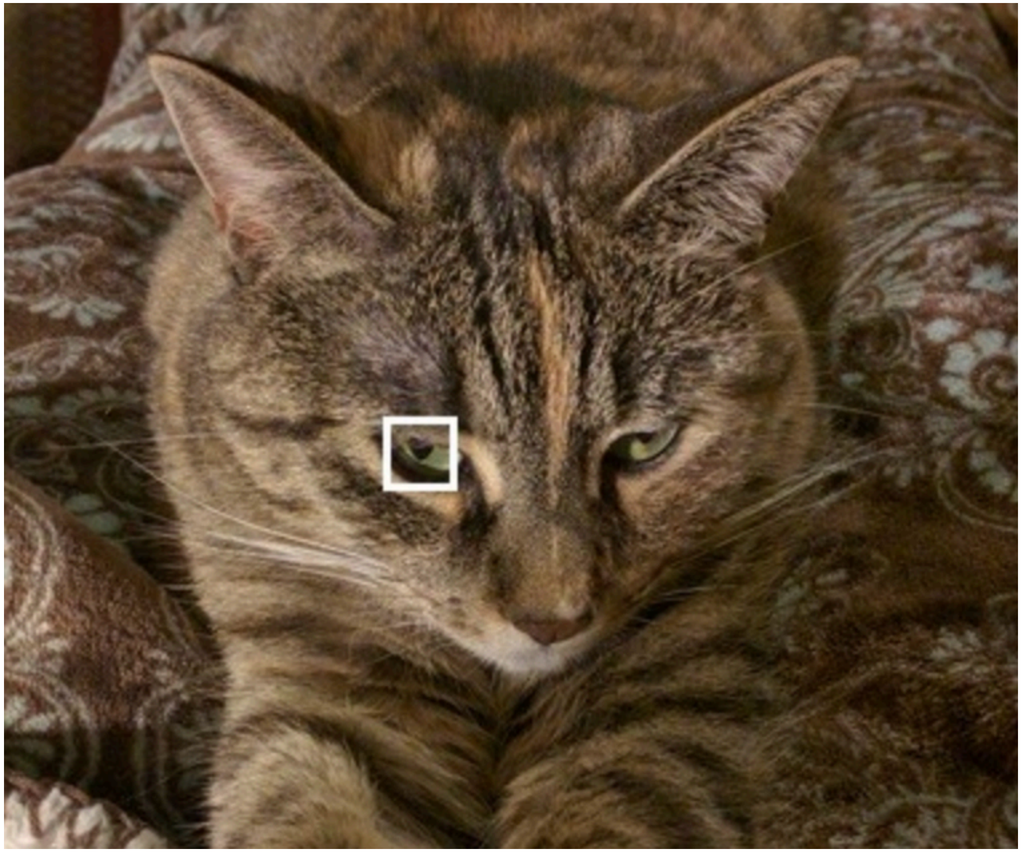
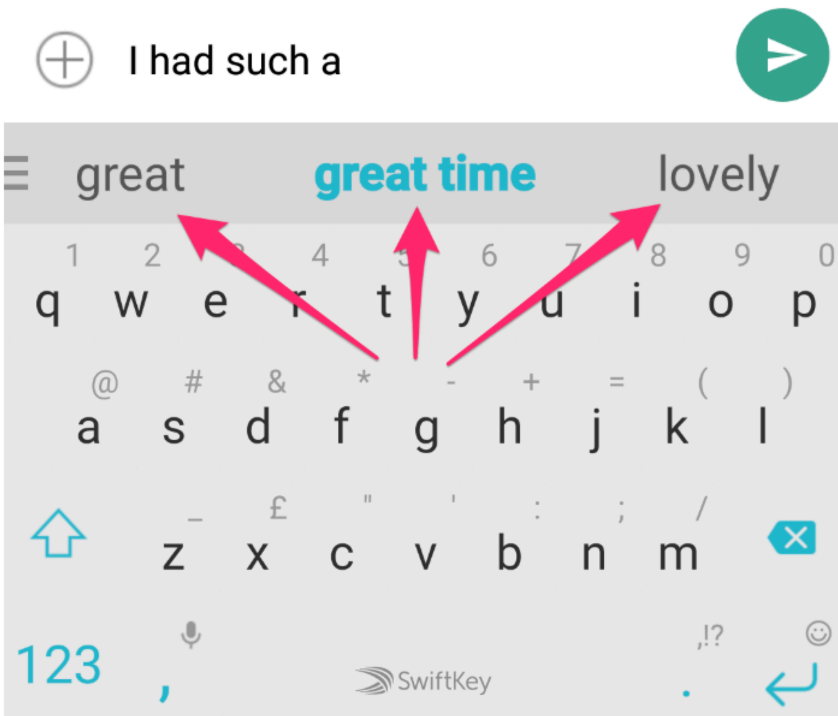


Image credit: https://datacarpentry.org/image-processing/

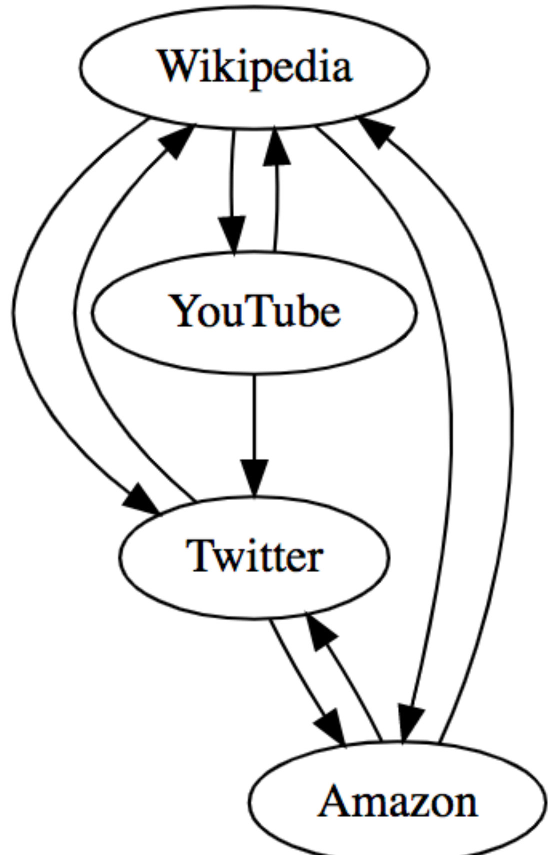


Markov chain

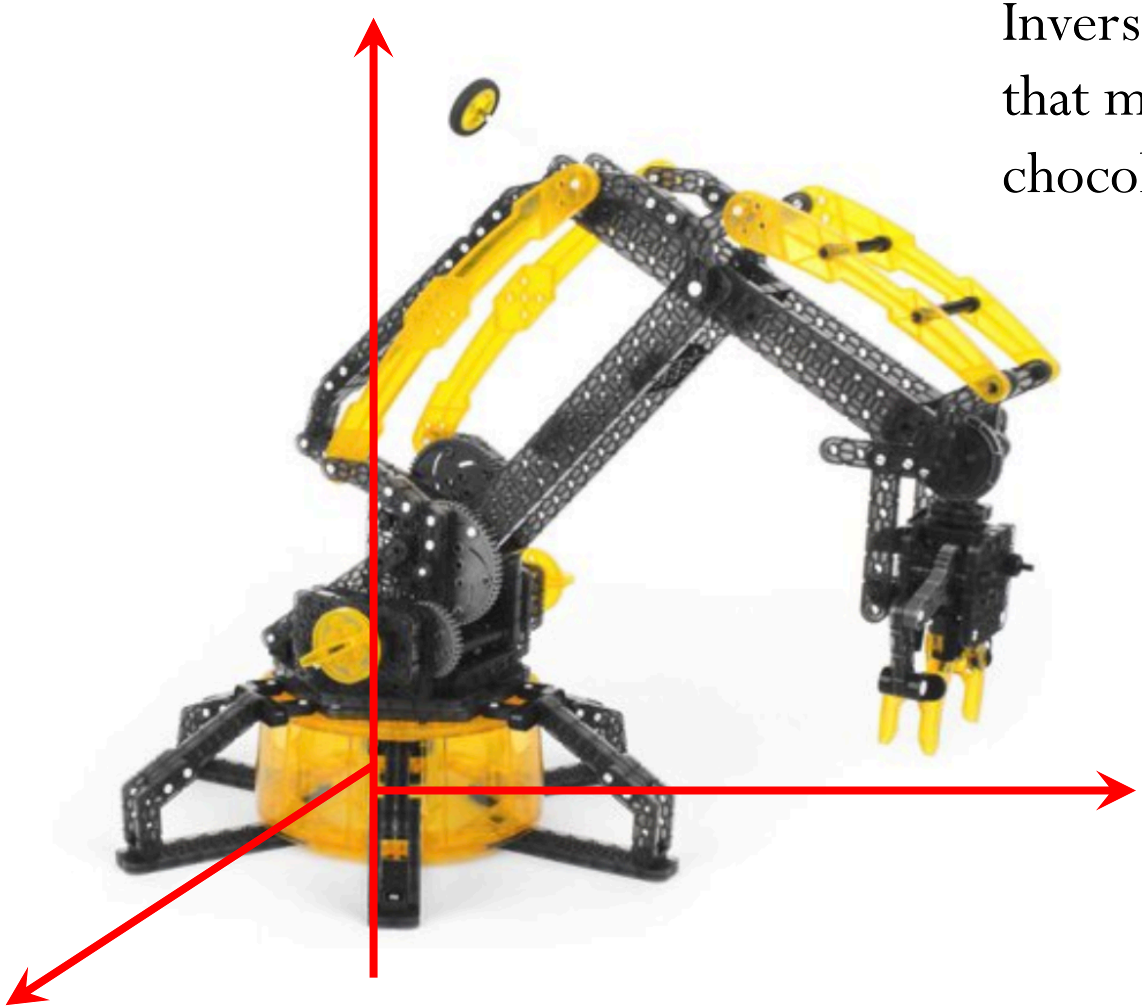
Word prediction



Page Rank



Nonlinear system of equations

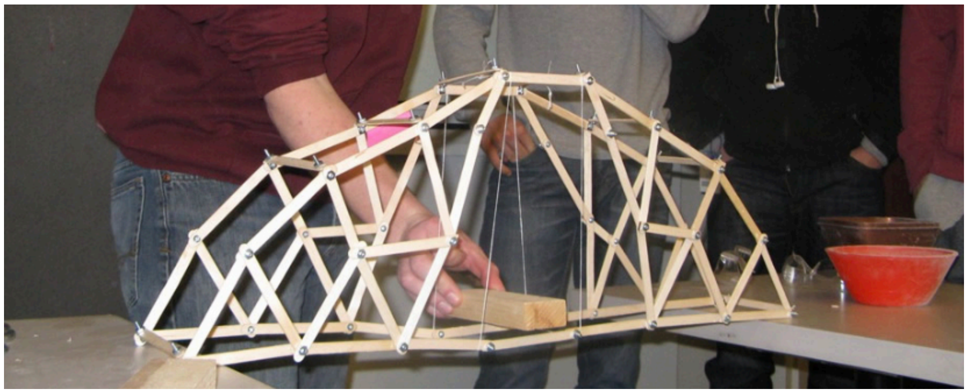


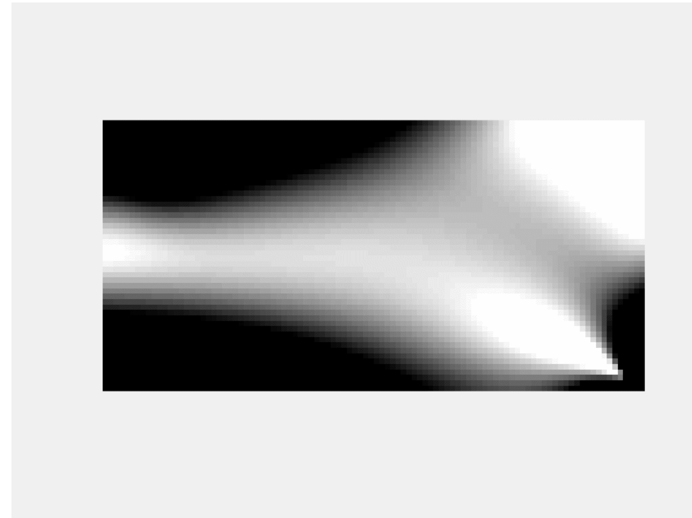
Inverse kinematics: find the angles that make the robotic hand grab a chocolate candy!

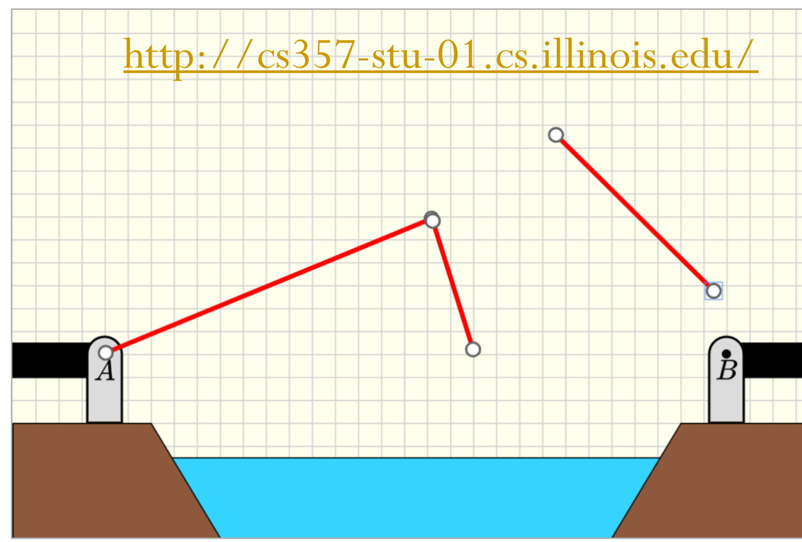
Optimization

Numerical simulations to find optimized bridge designs

Bridge design (high school projects)



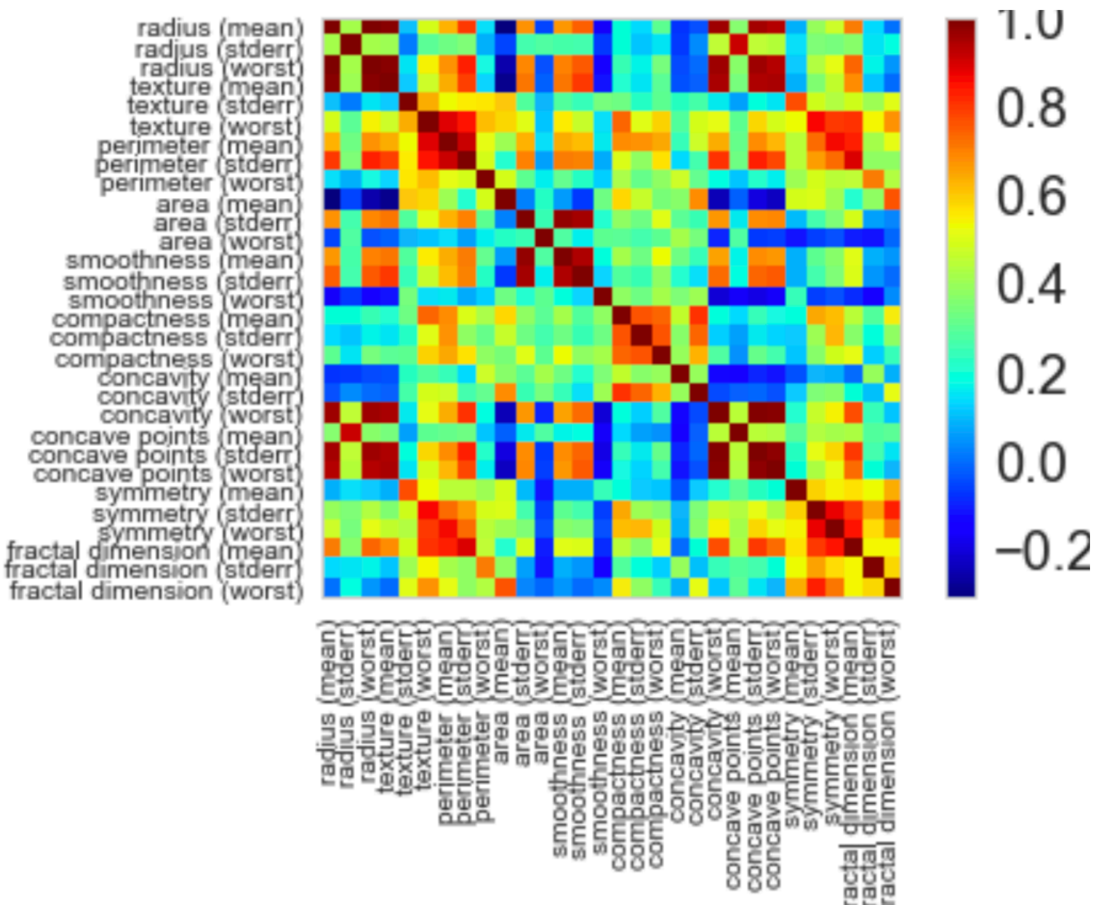


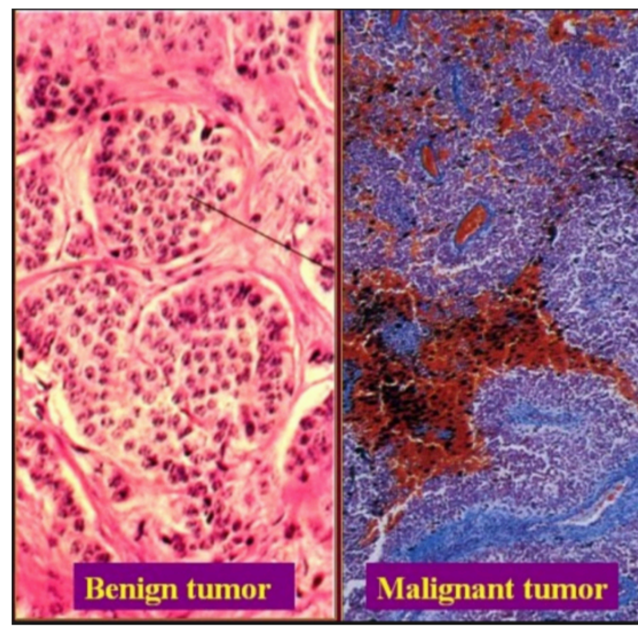


(Tolerance for members to be considered connected is 1/2 square grid.)

Linear Least Squares

Dataset containing the characteristics of cells for several patients. Can we make predictions if cells are benign or malignant?





Principal component analysis

Sometimes our dataset has too many features? How can we reduce the feature space and still keep the most important information?