Beyond CS106A
Chris Piech
CS106A, Stanford University
Life after ACM
Life After The ACM Libraries

• All quarter we have relied on the ACM Java libraries.
  – Karel, ConsoleProgram, RandomGenerator
  – GraphicsProgram, GOval, GRect, GOval, GLine, GImage, ...

• Today we will see how standard Java programs are made.
Using the ACM Libraries

import acm.program.*;

public class MyProgram extends ConsoleProgram {
    public void run() {
        println("Hello, world!");
    }
}

• This is a console program written using the ACM libraries.
  – It uses the ConsoleProgram class to represent a console.
  – The run method contains the program code.
  – The println method prints output to the graphical console.
public class Hello {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}

• The method **main** is the true entry point for a Java program.
  – It must have the exact heading shown above.
  – The String[] **args** are "command line arguments" (ignored).
  – The **println** command's true name is **System.out.println**.
  – Standard Java methods are **static** unless part of a class of objects.
Lets make Hello World!
Console Programs

• What does the **ConsoleProgram** library class do?
  – Creates a new graphical **window**
  – Puts a scrollable **text area** into it
  – Provides print and println commands to send text **output** to that window
  – contains a **main method** that calls your program class's run method
    • ConsoleProgram's run is empty, but you extend and override it

```
public class Hello extends ConsoleProgram {
    public void run() {
        println("Hello, world!");
    }
}
```
public class Age extends ConsoleProgram {
    public void run() {
        String name = readLine("What's your name? ");
        int age = readInt("How old are you? ");
        int years = 65 - age;
        println(name + " has " + years
                     + " years until retirement!");
    }
}

• The ACM library has simple console input commands like readLine, readInt, readDouble, and so on.
• These methods display a 'prompt' message, wait for input, re-prompt if the user types a bad value, and return the input.
public class Age {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);
        System.out.print("What's your name? ");
        String name = console.nextLine();
        System.out.print("How old are you? ");
        int age = console.nextInt();
        int years = 65 - age;
        System.out.println(name + " has " + years + " years until retirement!");
    }
}

- In standard Java, you must create a Scanner or similar object to read input from the console, which is also called System.in.
  - It does not automatically re-prompt and can crash on bad input.
The ACM library does several things to make graphics easier:

• Automatically creates and displays a **window** on the screen.
  – In standard Java, we must do this ourselves; it is called a **JFrame**.

• **Sets up a drawing canvas** in the center of the window
  In standard Java, we must create our own drawing canvas.

• Provides convenient methods to listen for mouse events.
  – In standard Java, event handling takes a bit more code to set up.
public class ColorFun extends Program {
    public void init() {
        JButton button1 = new JButton("Red!");
        JButton button2 = new JButton("Blue!");
        add(button1, SOUTH);
        add(button2, SOUTH);
        addActionListeners();
    }

    public void actionPerformed(ActionEvent event) {
        if (event.getActionCommand().equals("Red!")) {
            setBackground(Color.BLUE);
        } else {
            setBackground(Color.RED);
        }
    }
}
public class ColorFun implements ActionListener {
    public static void main(String[] args) {
        new ColorFun().init();
    }

    private JFrame frame;
    public void init() {
        frame = new JFrame("ColorFun");
        frame.setSize(500, 300);
        JButton button1 = new JButton("Red!");
        JButton button2 = new JButton("Blue!");
        button1.addActionListener(this);
        button2.addActionListener(this);
        frame.add(button1, "South");
        frame.add(button2, "South");
        frame.setVisible(true);
    }

    public void actionPerformed(ActionEvent event) {
        if (event.getActionCommand().equals("Red!")) {
            frame.setBackground(Color.BLUE);
        } else {
            frame.setBackground(Color.RED);
        }
    }
}
Summary

• **Benefits of libraries:**
  – simplify syntax/rough edges of language/API
  – avoid re-writing the same code over and over
  – possible to make advanced programs quickly
  – leverage work of others

• **Drawbacks of libraries:**
  – limitations on usage; e.g. ACM library cannot be re-distributed for commercial purposes
ArrayList<Double> evens = new ArrayList<>();
for(int i = 0; i < 100; i++) {
    if(i % 2 == 0) {
        evens.add(i);
    }
}
println(evens);

prints [2, 4, 6, 8, 10, 12, ... ]
Vector<double> evens;
for(int i = 0; i < 100; i++) {
    if(i % 2 == 0) {
        evens.add(i);
    }
}
cout << evens << endl;

prints [2, 4, 6, 8, 10, 12, ... ]
evens = []
for i in range(100):
    if i % 2 == 0:
        evens.append(i)
print evens

prints [2, 4, 6, 8, 10, 12, ...]
```javascript
var evens = []
for(var i = 0; i < 100; i++) {
    if(i % 2 == 0) {
        evens.push(i)
    }
}
console.log(evens)
```

prints `[2, 4, 6, 8, 10, 12, ... ]`
Future in the CS curriculum
The CS Curriculum

CS106A → CS106B → CS109 + CS103 → Further study

Piech, CS106A, Stanford University
The CS Curriculum

CS106A

CS106B

CS109 + CS103

Further study

Computational Biology
The CS Curriculum

CS106A
CS106B
CS109 + CS103

Further study

The Internet

Piech, CS106A, Stanford University
The CS Curriculum

CS106A → CS106B

Further study

AI and Robotics

CS109 + CS103

Piech, CS106A, Stanford University
The CS Curriculum

CS106A → CS106B → CS109 + CS103 → Datascience → Further study

Piech, CS106A, Stanford University
The CS Curriculum

CS106A

CS106B

CS109 + CS103

Further study

Human Computer Interaction

Piech, CS106A, Stanford University
Machine Learning
Machine Learning
or, How we learned to decompose
There is something going on in the world of AI
Something big (for us)...
[suspense]
How can we develop intelligent agents?
How can we develop intelligent agents?

- Computer programs
- Better than chance
- As well as humans
Big Milestones Starting in 1997

1997 Deep Blue

2005 Stanley

2011 Watson
Self Driving Cars
Computers Making Art
The Last Remaining Board Game
Early Optimism 1950

1952

1955

Axioms \[ \vdash C \]

ATP System (theorem prover)

Yes (proof/answer)

No

Timeout
Early Optimism 1950

“Machines will be capable, within twenty years, of doing any work a man can do.”
–Herbert Simon, 1952
The spirit is willing but the flesh is weak.

(Russian)

The vodka is good but the meat is rotten.

The world is too complex
BRACE YOURSELVES

WINTER IS COMING
Simple Example: Identifying Cats

- We have a picture and we want to know if it’s a cat or not.

- True

- False

- True

- False

Piech, CS106A, Stanford University
private void isCat(GImage animal) {
    int[][][] pixels = animal.getPixelArray();
    if (containsTwoEyes(pixels)){
        if (hasWhiskers(pixels)){
            if (hasPointyEars(pixels)){
                return true;
            }
        }
    }
    return false;
}
Some Tricky Cases
Pros / Cons

• Pros
  – Matches our human intuition about what a cat is
  – Easy to understand the code

• Cons
  – Requires us to explicitly enumerate every feature that’s important, and know how important it is
  – Need to write code to detect eyes, and whiskers, and the pointiness of ears
  – Will never improve... cannot learn from its mistakes
Hard problems seemed impossible.
Great idea #1 learn from experience
Great idea #2 inspired by biology
Neuron
Neuron
Neuron

- Dendrites
- Soma
- Axon
- Myelin sheath
- Terminal button
Some Inputs are More Important
Artificial Neurons
Java Demo

Artificial Neuron

input1
2.5

input2
1.0

input3
7.0

input4
-9.0

3.5

0.97
// calculate the activation of a neuron
private double activate(double[] weights, double[] inputs) {
    double weightedSum = 0;
    for(int i = 0; i < inputs.length; i++) {
        weightedSum += weights[i] * inputs[i];
    }

    return squash(weightedSum);
}

// the sigmoid function forces a value to be between 0 and 1
private double squash(double value) {
    return 1 / (1 + Math.exp(-value));
}
Digit Recognition Example

Let’s make feature vectors from pictures of numbers

input = [0, 0, 0, 0, ..., 1, 0, 0, 1, ..., 0, 0, 1, 0]
label = 0

input = [0, 0, 1, 1, ..., 0, 1, 1, 0, ..., 0, 1, 0, 0]
label = 1
Computer Vision

Piech, CS106A, Stanford University
Logistic Regression is like the Harry Potter Sorting Hat

Classification

That is a picture of a one
Logistic Regression is like the Harry Potter Sorting Hat

Classification

That is a picture of a zero
Classification

That is a picture of an zero

* It doesn’t have to be correct all of the time
Can you do it?
Single Neuron

This means it predicts a 0
Single Neuron

Indicates fully connected

This means it predicts a 0
Single Neuron

This means it predicts a 1
This means it predicts a 1.
Biological Basis for Neural Networks

• A neuron

• Your brain

Actually, it’s probably someone else’s brain
We Can Put Neurons Together

This means it predicts a 0
Demonstration

http://scs.ryerson.ca/~aharley/vis/conv/

Piech, CS106A, Stanford University
What Does This Look Like in Code?
Aside: decomposition
How do we get those weights?
Chain Rule Down the Network

\[
\frac{\text{d}y}{\text{d}o} = \frac{\text{d}z}{\text{d}y} \times \frac{\text{d}y}{\text{d}x}
\]
Learning Weights

\[ LL(\theta) = y \log \sigma(\theta^T x) + (1 - y) \log[1 - \sigma(\theta^T x)] \]

\[
\frac{\partial LL(\theta)}{\partial \theta_j} = \frac{\partial}{\partial \theta_j} y \log \sigma(\theta^T x) + \frac{\partial}{\partial \theta_j} (1 - y) \log[1 - \sigma(\theta^T x)] = \left[ \frac{y}{\sigma(\theta^T x)} - \frac{1 - y}{1 - \sigma(\theta^T x)} \right] \frac{\partial}{\partial \theta_j} \sigma(\theta^T x) = \left[ \frac{y - \sigma(\theta^T x)}{\sigma(\theta^T x)[1 - \sigma(\theta^T x)]} \right] \sigma(\theta^T x)[1 - \sigma(\theta^T x)] x_j = \left[ y - \sigma(\theta^T x) \right] x_j
\]
Artificial Neurons: One of the greatest decompositions of our lifetimes
model.calculatePartialDerivative(data)
model.update(data)
Let’s Train!

http://cs.stanford.edu/people/karpathy/convnetjs/demo/classify2d.html

Piech, CS106A, Stanford University
Like lego pieces
Training set: Aligned images of faces.
GoogLeNet Brain

1 Trillion Artificial Neurons
GoogLeNet Brain Graph

22 layers deep

Multiple, Multi class output
The Face Neuron

Le, et al., *Building high-level features using large-scale unsupervised learning*. ICML 2012
Hire the smartest people in the world

Invent cat detector
<table>
<thead>
<tr>
<th>Neuron 1</th>
<th>Neuron 2</th>
<th>Neuron 3</th>
<th>Neuron 4</th>
<th>Neuron 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
</tr>
</tbody>
</table>

Le, et al., *Building high-level features using large-scale unsupervised learning*. ICML 2012
Le, et al., *Building high-level features using large-scale unsupervised learning*. ICML 2012
ImageNet Classification

22,000 categories

14,000,000 images

Hand-engineered features (SIFT, HOG, LBP), Spatial pyramid, SparseCoding/Compression

Le, et al., *Building high-level features using large-scale unsupervised learning*. ICML 2012
... smoothhound, smoothhound shark, Mustelus mustelus
American smooth dogfish, Mustelus canis
Florida smoothhound, Mustelus norrisi
whitetip shark, reef whitetip shark, Triaenodon obesus
Atlantic spiny dogfish, Squalus acanthias
Pacific spiny dogfish, Squalus suckleyi
hammerhead, hammerhead shark
smooth hammerhead, Sphyrna zygaena
smalleye hammerhead, Sphyrna tudes
shovelhead, bonnethead, bonnet shark, Sphyrna tiburo
angel shark, angelfish, Squatina squatina, monkfish
electric ray, crampfish, numbfish, torpedo
guitarfish
smooth hammerhead, Sphyrna zygaena
smalleye hammerhead, Sphyrna tudes
shovelhead, bonnethead, bonnet shark, Sphyrna tiburo
angel shark, angelfish, Squatina squatina, monkfish
guitarfish
...
Le, et al., *Building high-level features using large-scale unsupervised learning*. ICML 2012
0.005%  1.5%  43.9%

Random guess  Pre Neural Networks  GoogLeNet

Szegedy et al, Going Deeper With Convolutions, CVPR 2015
0.005%  1.5%  82.7%
Random guess  Pre Neural Networks  NASNet

Vision has Social Implications

Apoptotic

Viable tumor region

Necrosis

Neural network
Where is this useful?

A machine learning algorithm performs **better than** the best dermatologists.

Developed this year, at Stanford.

Estimated daily per capita expenditure, 2012-2015

http://sustain.stanford.edu/
Understanding Students

Huge improvement in ability to predict for real students

Piech, CS106A, Stanford University
Tl;dr our brain is constantly decomposing
Told Vision Was 30 Years Out
Almost perfect…
What a time to be alive
Ethics in AI
The end
• **JAR**: Java Archive. A compressed binary of a Java program.
  – The typical way to **distribute a Java app as a single file**.
  – Essentially just a ZIP file with Java .class files.

• Making a JAR of your project in Eclipse:
  – File → Export ... → Java → **Runnable JAR File**

• **see handout on course web site**