

Day 1: Introduction to Machine Learning

Summer STEM: Machine Learning

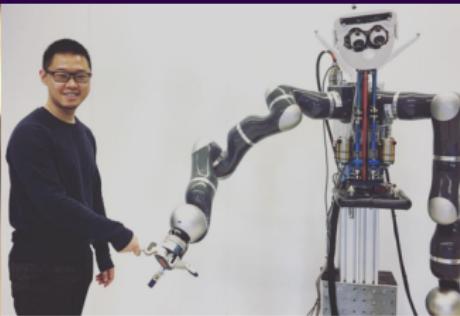
Department of Electrical and Computer Engineering
NYU Tandon School of Engineering
Brooklyn, New York

August 2, 2020

Outline

- 1** Teacher and Student Introductions
- 2** What is Machine Learning?
- 3** Course Outline
- 4** Matrices and Vectors
- 5** Setting Up Python
- 6** Lab: Python Basics
- 7** Demo and Exercises: NumPy

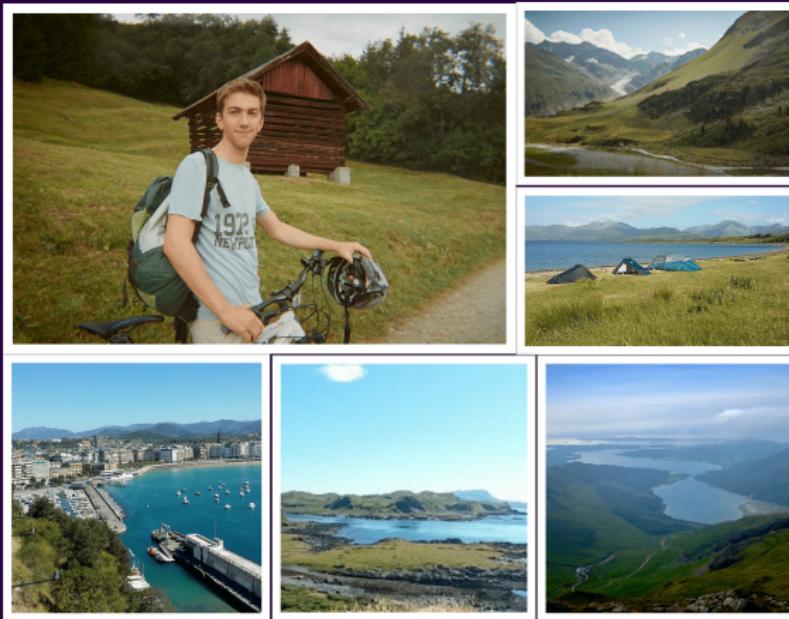
Huaijiang



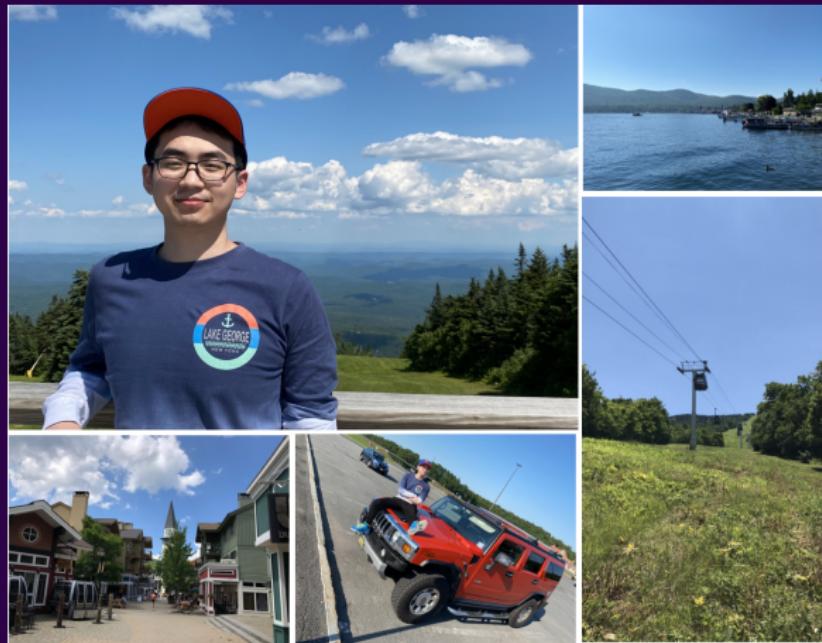
Alireza



Armand



Hao



Tell the class about yourself

- Write down the following information:
 - Name
 - Grade
 - In which city/town are you currently living?
 - What do you want to get out of this class?
 - What is your favourite movie?
 - What is the IMDB score of this movie!
 - What is the category of this movie? (thriller/drama/action, etc)
 - Rate your coding experience from 1 (no experience) to 5 (plenty of experience)!
- Share your answers with the class!
- We'll visualize this dataset using Python tomorrow!
 - Link to excel sheet here

Outline

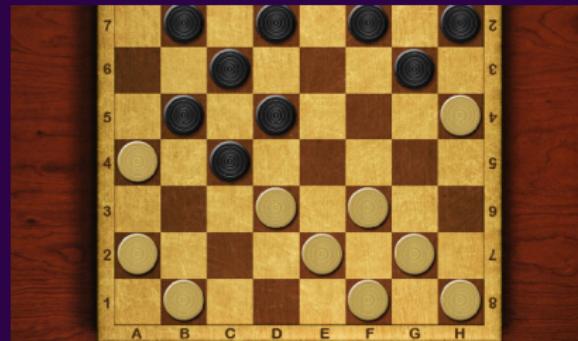
- 1** Teacher and Student Introductions
- 2** What is Machine Learning?
- 3** Course Outline
- 4** Matrices and Vectors
- 5** Setting Up Python
- 6** Lab: Python Basics
- 7** Demo and Exercises: NumPy

Machine Learning

- Most recent exciting technology
- We use these algorithms dozens of times a day
 - Search Engine
 - Recommendations
- Machine Learning is an important component to achieve artificial general intelligence
- Practice is the key to learn machine learning

Definition

- Machine Learning is a field of study that gives computers the ability to learn without being explicitly programmed.



Example: Digit Recognition



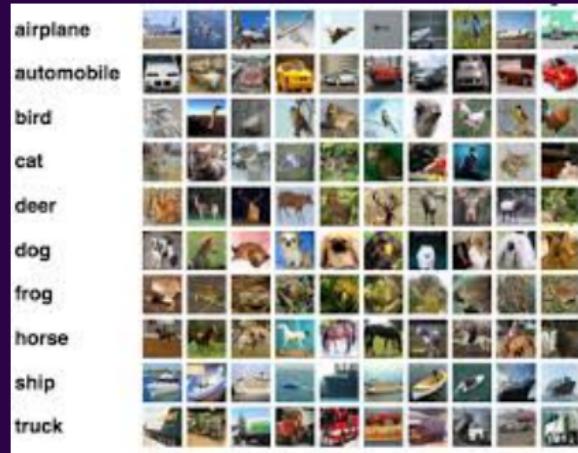
- Challenges with expert approach
 - Simple expert rule breaks down in practice
 - Difficult to translate our knowledge into code
- Machine Learning approach
 - Learned systems do very well on image recognition problems

```
def classify(image):
    ...
    nv = count_vert_lines(image)
    nh = count_horiz_lines(image)
    ...

    if (nv == 1) and (nh == 1):
        digit = 7
    ...

    return digit
```

Example: CIFAR 10



Machine Learning Problem Pipeline

- 1** Formulate the problem: regression, classification, or others?
- 2** Gather and visualize the data
- 3** Design the model and the loss function
- 4** Train your model
 - (a) Perform feature engineering
 - (b) Construct the design matrix
 - (c) Choose regularization techniques
 - (d) Tune hyper-parameters using a validation set
 - (e) If the performance is not satisfactory, go back to step (a)
- 5** Evaluate the model on a test set

Outline

- 1** Teacher and Student Introductions
- 2** What is Machine Learning?
- 3** Course Outline
- 4** Matrices and Vectors
- 5** Setting Up Python
- 6** Lab: Python Basics
- 7** Demo and Exercises: NumPy

Course Outline

- Day 1: Introduction to ML
- Day 2: Linear Regression
- Day 3: Overfitting and Generalization
- Day 4: Classification and Logistic Regression
- Day 5: Mini Project
- Day 6: Neural Networks
- Day 7: Convolutional Neural Networks
- Day 8: Deep Generative Models
- Day 9: Final Project
- Day 10: Social Impacts of ML and Final Project Presentations

Course Format, Website, Resources

- Course Website:

https://github.com/asarmadi/summer_ml_s3

- Github: share collections of documents, repositories of code
- Contains lecture slides, code notebooks, and datasets
- Slides and demo code posted before lecture, solutions to the lab posted after

- After-class discussion: Piazza. Links to class recordings will also be posted there.

- We strongly encourage programming in Python via Google Colab.

- We'll give additional resources at the end of each day based on student interest

Outline

- 1** Teacher and Student Introductions
- 2** What is Machine Learning?
- 3** Course Outline
- 4** Matrices and Vectors
- 5** Setting Up Python
- 6** Lab: Python Basics
- 7** Demo and Exercises: NumPy

Vectors

- A **vector** is an ordered list of numbers or symbols
 - Ex:

$$\mathbf{v} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} 2 \\ -3 \\ 8 \\ 6 \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} 3 \\ 0 \\ 2 \end{bmatrix}$$

Vectors

- Vectors of the same size may be added together, element-wise

- Ex: $\mathbf{u} = \begin{bmatrix} 3 \\ -1 \end{bmatrix}$ and $\mathbf{v} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$

$$\mathbf{u} + \mathbf{v} = \begin{bmatrix} 3 + 1 \\ (-1) + 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix}$$

- Vectors may be scaled by a number, element-wise

- Ex: $3\mathbf{v} = \begin{bmatrix} 3 \times 1 \\ 3 \times 2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix}$

Vectors

- Norm of a vector (L₂ Norm)

- Ex: If $\mathbf{v} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $\|\mathbf{v}\| = \sqrt{1^2 + 2^2} = \sqrt{5}$

- Inner product: sum of element-wise products of two vectors

- Ex: $\mathbf{u} \cdot \mathbf{v} = \begin{bmatrix} 3 \\ -1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 3 \times 1 + (-1) \times 2 = 3 - 2 = 1$

- Gives the angle between two vectors $\cos \theta = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}$

Vectors

■ If $\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}$ and $\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$

$$\mathbf{u} + \mathbf{v} = \begin{bmatrix} u_1 + v_1 \\ u_2 + v_2 \\ \vdots \\ u_n + v_n \end{bmatrix}$$

■ For any real number α , $\alpha\mathbf{u} = \begin{bmatrix} \alpha u_1 \\ \alpha u_2 \\ \vdots \\ \alpha u_n \end{bmatrix}$

Vectors

- inner product :

$$\mathbf{u} \cdot \mathbf{v} = u_1 \times v_1 + u_2 \times v_2 + \cdots + u_n \times v_n = \sum_{i=1}^n u_i \times v_i$$

- norm :

$$\|\mathbf{u}\| = \sqrt{u_1^2 + u_2^2 + \cdots + u_n^2} = \sqrt{\sum_{i=1}^n u_i^2}$$

- Squared norm :

$$\|\mathbf{u}\|^2 = u_1^2 + u_2^2 + \cdots + u_n^2 = \sum_{i=1}^n u_i^2$$

Exercise: Vectors

Let $\mathbf{p} = \begin{bmatrix} 3 \\ 2 \\ 9 \\ 4 \end{bmatrix}$ and $\mathbf{q} = \begin{bmatrix} 1 \\ 9 \\ 0 \\ 3 \end{bmatrix}$, calculate

- $3\mathbf{q} + 2\mathbf{p}$
- $\mathbf{q} \cdot \mathbf{q}$ and $\|\mathbf{q}\|^2$
- $\mathbf{p} \cdot \mathbf{q}$ and $\|\mathbf{p}\| \|\mathbf{q}\|$
- $\left\| \frac{\mathbf{p}}{\|\mathbf{p}\|} \right\|$

Matrices

- A **matrix** is a rectangular array of numbers or symbols arranged in rows and columns. We can conceptualize it as a collection of vectors.
 - Ex: 2 by 2 matrix, $M = \begin{bmatrix} 1 & 3 \\ 2 & -1 \end{bmatrix}$
- Matrices of the same shape may be added together, element-wise
 - Ex: $A = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 & 8 \\ 7 & 11 \end{bmatrix}, A + B = \begin{bmatrix} 1 & 9 \\ 9 & 12 \end{bmatrix}$
- Matrices may be scaled, element-wise
 - Ex: $\alpha B = \begin{bmatrix} 0 & 8\alpha \\ 7\alpha & 11\alpha \end{bmatrix}$, where α is a scalar

Exercise: Matrices

- $\begin{bmatrix} 1 & 3 \\ 2 & -1 \\ 4 & 7 \end{bmatrix} + \begin{bmatrix} 9 & 2 \\ -7 & 6 \\ 3 & 1 \end{bmatrix} = ?$
- $2 \begin{bmatrix} 1 & 9 \\ 3 & -2 \end{bmatrix} = ?$

Vectors and Matrices

- We may consider a vector as a matrix
 - **Row Vector:** shape $(1 \times N)$
Ex: $\mathbf{v} = \begin{bmatrix} 1 & 2 \end{bmatrix}$
 - **Column Vector:** shape $(N \times 1)$
Ex: $\mathbf{v} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$
- We'll consider vectors as column vectors by default

Matrices

Example :

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 & 7 \end{bmatrix}$$

- What is the shape of A ?

Matrices

Example :

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 & 7 \end{bmatrix}$$

- What is the shape of A ?

(2 × 5)

Matrices

General case :

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1m} \\ A_{21} & A_{22} & \dots & A_{2m} \\ \vdots & & & \vdots \\ A_{n1} & A_{n2} & \dots & A_{nm} \end{bmatrix}$$

- What is the shape of A ?

Matrices

General case :

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1m} \\ A_{21} & A_{22} & \dots & A_{2m} \\ \vdots & & & \vdots \\ A_{n1} & A_{n2} & \dots & A_{nm} \end{bmatrix}$$

- What is the shape of A ?

$n \times m$

Matrices

General case :

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1m} \\ A_{21} & A_{22} & \dots & A_{2m} \\ \vdots & & & \vdots \\ A_{n1} & A_{n2} & \dots & A_{nm} \end{bmatrix}$$

- What is the shape of A ?

$n \times m$

- A_{ij} is the element at the i^{th} row and j^{th} column

Matrices

Example :

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 & 7 \end{bmatrix}$$

- $A_{13} = ?$
- $A_{21} = ?$
- $A_{24} = ?$

Matrices

- Two matrices, A and B, can be multiplied together provided their shapes meet the criteria :
- **Criteria:** # cols of A must equal the # rows of B
 - Ex : $A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$ $B = \begin{bmatrix} -3 & 1 \\ 0 & -1 \\ 3 & 1 \end{bmatrix}$
 - Shape of A : (2×3) , Shape of B : (3×2)

Matrices

- Two matrices, A and B, can be multiplied together provided their shapes meet the criteria :
- **Criteria:** # cols of A must equal the # rows of B
 - Ex : $A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$ $B = \begin{bmatrix} -3 & 1 \\ 0 & -1 \\ 3 & 1 \end{bmatrix}$
 - Shape of A : (2×3) , Shape of B : (3×2)
- Result is a matrix with shape (# rows A \times # cols B)
 - Ex : If $C = AB$ then, C is of shape (2×2) :

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

Matrices

■ To sum up :

If A is of shape $(M \times K)$ and B of shape $(K \times N)$,
We can define $C = AB$, and C will be of shape $(M \times N)$

Matrices

- If $C = AB$ then, $(C)_{ij} = \sum_{k=1}^N A_{ik}B_{kj}$
- Inner product of the i -th row of A and the j -th column of B

Matrices

- If $C = AB$ then, $(C)_{ij} = \sum_{k=1}^N A_{ik}B_{kj}$
- Inner product of the i -th row of A and the j -th column of B

$$\text{■ Ex : } C = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -3 & 1 \\ 0 & -1 \\ 3 & 1 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

$$C_{11} = \begin{bmatrix} 2 \\ -1 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} -3 \\ 0 \\ 3 \end{bmatrix} = -6 \quad C_{12} = \begin{bmatrix} 2 \\ -1 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = 3$$

$$C_{21} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} -3 \\ 0 \\ 3 \end{bmatrix} = 6 \quad C_{22} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = 0$$

Matrices

- $\begin{bmatrix} 1 & 3 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} = ?$
- $\begin{bmatrix} 1 & 3 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} 2 & 4 \\ 3 & 2 \end{bmatrix} = ?$
- $\begin{bmatrix} 2 & 4 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 2 & -1 \end{bmatrix} = ?$
- In general, $AB \neq BA$

Matrices

- **Transpose:** A^T swaps the rows and columns of matrix A

- Ex: $\begin{bmatrix} 2 \\ 3 \end{bmatrix}^T = [2 \ 3]$ and $\begin{bmatrix} 2 & 4 \\ 3 & 2 \end{bmatrix}^T = \begin{bmatrix} 2 & 3 \\ 4 & 2 \end{bmatrix}$

- $[2 \ 3] \begin{bmatrix} 1 & 2 \\ 3 & -1 \end{bmatrix} = ?$

- $(AB)^T = B^T A^T$

Exercises: Matrix Multiplication

- $X = \begin{bmatrix} 1 & 2 & -1 \\ 1 & 0 & 1 \end{bmatrix}$ $Y = \begin{bmatrix} 3 & 1 \\ 0 & -1 \\ -2 & 3 \end{bmatrix}$ $Z = \begin{bmatrix} 1 \\ 4 \\ 6 \end{bmatrix}$
- Calculate XY , YX , $Z^T Y$

Matrices

- $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ?$
- $\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = ?$

Matrices

- $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ?$
- $\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = ?$
- $I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- $I_n = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$

- For any matrix of shape $n \times n$, $AI_n = I_nA = A$

Matrix Inverse

- Analogy: Reciprocal of a number $\frac{1}{a}a = 1$
- Matrix inverse only defined for square matrix ($\# \text{ rows} = \# \text{ cols}$)

$$A^{-1}A = AA^{-1} = I = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Matrix Inverse

- Hard to compute by hand, but for 2 by 2 matrix, it is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Matrix Inverse

- Hard to compute by hand, but for 2 by 2 matrix, it is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

- The matrix inverse does not always exist. Can you tell when that is the case for 2 by 2 matrices based on the formula given above?

Matrix Inverse

When is matrix inverse useful? We can use it to solve systems of linear equations!

- Consider the following equations

$$\begin{cases} x + 2y = 5 \\ 3x + 5y = 13 \end{cases}$$

- In matrix-vector form

$$\begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 5 \\ 13 \end{bmatrix}$$

Matrix inverse

$$\begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 5 \\ 13 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix}^{-1} \begin{bmatrix} 5 \\ 13 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix}^{-1} \begin{bmatrix} 5 \\ 13 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix}^{-1} \begin{bmatrix} 5 \\ 13 \end{bmatrix}$$

Outline

- 1 Teacher and Student Introductions
- 2 What is Machine Learning?
- 3 Course Outline
- 4 Matrices and Vectors
- 5 Setting Up Python
- 6 Lab: Python Basics
- 7 Demo and Exercises: NumPy

Setting Up Python

- Google Colab
 - Interactive programming online
 - No installation
 - Free GPU for 12 hours
- Your task:
 - Register a Google account and set up Google Colab
 - Run `print('hello world!')`
 - Save the notebook on your computer

Outline

- 1** Teacher and Student Introductions
- 2** What is Machine Learning?
- 3** Course Outline
- 4** Matrices and Vectors
- 5** Setting Up Python
- 6** Lab: Python Basics
- 7** Demo and Exercises: NumPy

Python Basics

- Program
 - We write operations to be executed on variables
- Variables
 - Referencing and interacting with items in the program
- If-Statements
 - Conditionally execute lines of code
- Functions
 - Reuse lines of code at any time

Python Basics

- Lists
 - Store an ordered collection of data
- Loops
 - Conditionally re-execute code
- Strings
 - Words and sentences are treated as lists of characters
- Classes (advanced)
 - Making your own data-type. Functions and variables made to be associated with it too.

Lab: Python Basics

- Write a function to find the second largest number in a list
(Hint: use `sort()`)
- Define a class which has at least two methods:
 - `getString`: to get a string from console input (Hint: use `input()`)
 - `printString`: to print the string in upper case (Hint: use `upper()`)
- Write a test function to test the class methods.

Outline

- 1** Teacher and Student Introductions
- 2** What is Machine Learning?
- 3** Course Outline
- 4** Matrices and Vectors
- 5** Setting Up Python
- 6** Lab: Python Basics
- 7** Demo and Exercises: NumPy

Demo and Exercises: NumPy

Open `demo_vectors_matrices.ipynb`

- Your task: use NumPy functions to compute the exercises we did earlier this morning. Compare the results.

Homework

- Matrices exercises on Piazza
- Tutorial Video on Matplotlib

These slides have been modified from the original slides provided through the courtesy of Nikola, Akshaj, Aishwarya, and Jack.