NYU

NYU Summer Machine Learning Program

Presenter Name Here
Date Here

NYU

Introduction to Machine Learning

Day 1

Learning Objectives

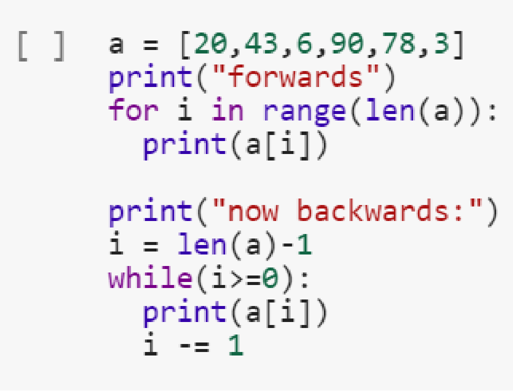
- ❑ Fundamentals of Python programming language
- ❑ Familiarity with NumPy and Pandas
- ❑ Provide examples of Machine Learning used today
- ❑ Given a new problem, qualitatively describe how a machine learning can be used
    - ❑ Formulate a potential machine learning task
    - ❑ Identify the data needed for the task
- ❑ Classify a machine learning task
    - ❑ Regression vs. Classification
- ❑ Identify the predictors and target variables
- ❑ Determine the role of expert knowledge in the task vs. data driven learning

Outline

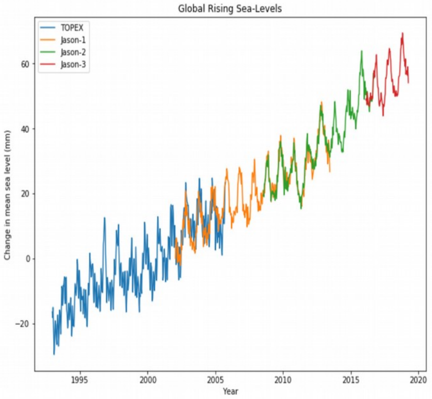
- Basics of Programming
  - Python loops and data structures
  - Scientific computational package – NumPy
  - Data visualization
- What is machine learning?
- Types of machine learning algorithms
  - Classification and regression
- Why the hype today?

Programming basics in Google Colab Notebook

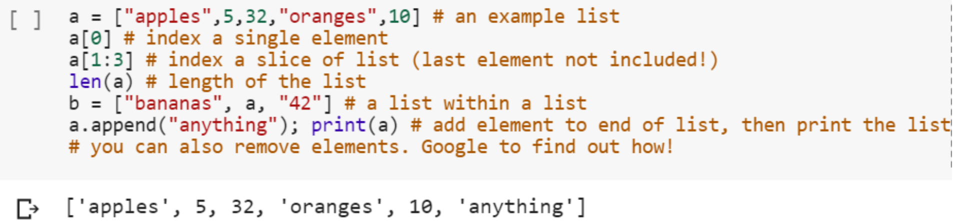
- ❑ Google Colab is a free cloud service
  - ❑ Machine Learning education and research tool
  - ❑ Free and requires no setup
  - ❑ Supports free GPU to perform fast computations
  - ❑ You can improve your python programming skills
- ❑ Python
  - ❑ Loops
  - ❑ Data Structures
- ❑ Data Visualization
  - ❑ Load data using Pandas
  - ❑ Visualize the data by plotting histograms, scatter plots, etc.



```
[ ]   a = [20,43,6,90,78,3]
      print("forwards")
      for i in range(len(a)):
        print(a[i])

      print("now backwards:")
      i = len(a)-1
      while(i>=0):
        print(a[i])
        i -= 1
```



Global Rising Sea-Levels



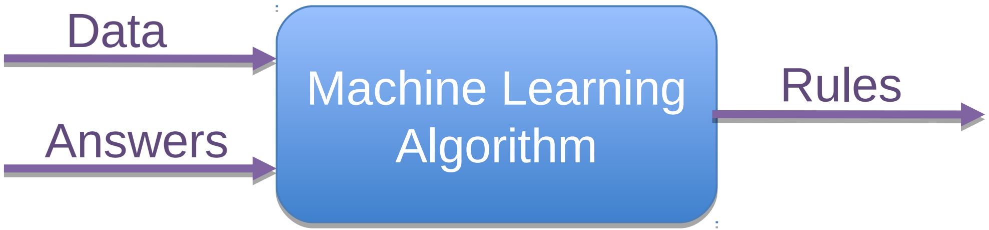
```
[ ]   a = ["apples",5,32,"oranges",10] # an example list
      a[0] # index a single element
      a[1:3] # index a slice of list (last element not included!)
      len(a) # length of the list
      b = ["bananas", a, "42"] # a list within a list
      a.append("anything"); print(a) # add element to end of list, then print the list
      # you can also remove elements. Google to find out how!

      ['apples', 5, 32, 'oranges', 10, 'anything']
```

Outline

- Basics of Programming
  - Python loops and data structures
  - Scientific computational package – NumPy
  - Data visualization
- What is machine learning?
- Types of machine learning algorithms
  - Classification and regression
- Why the hype today?

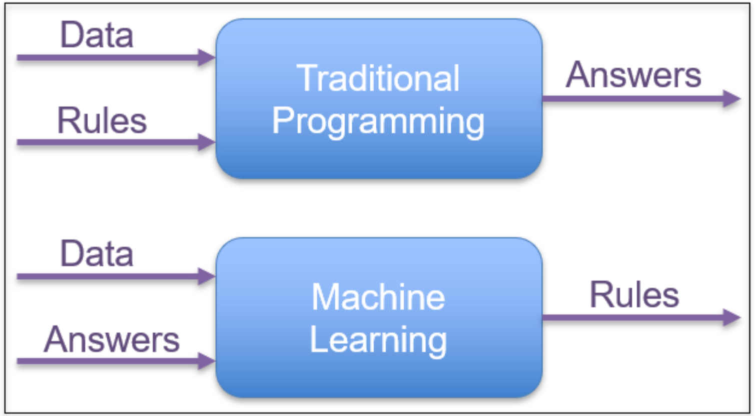
What is Machine Learning?

- ❑ Learn the algorithm from known data to generate the rules
- ❑ Make predictions on unknown data using these rules.



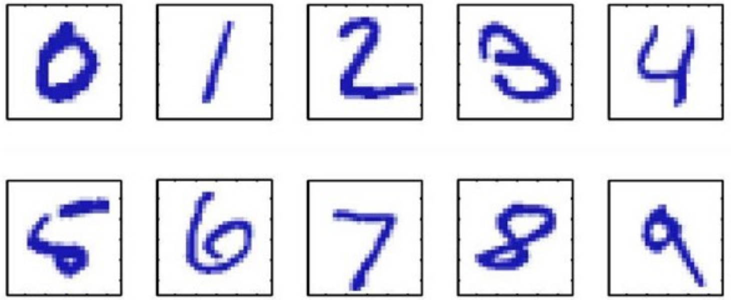
Why Machine Learning over Expert Approach?

- ❑ Human expertise does not exist (ex: complex medical processes we don’t fully understand)
- ❑ Humans are unable to explain their expertise (speech recognition)
- ❑ Solution changes in time (routing on a computer network)
- ❑ Solution needs to be adapted to specific cases (user biometrics)



Example 1: Digit Recognition

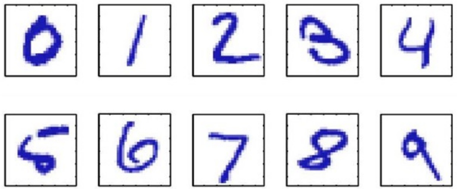
- Problem: Recognize a digit from the image
- MNIST dataset challenge
  - Dataset developed in 1990s to spur AI research on a challenging problem for the time
  - Data taken from census forms
  - Became a classic benchmark for machine vision problems
  - We will see this dataset extensively in this class



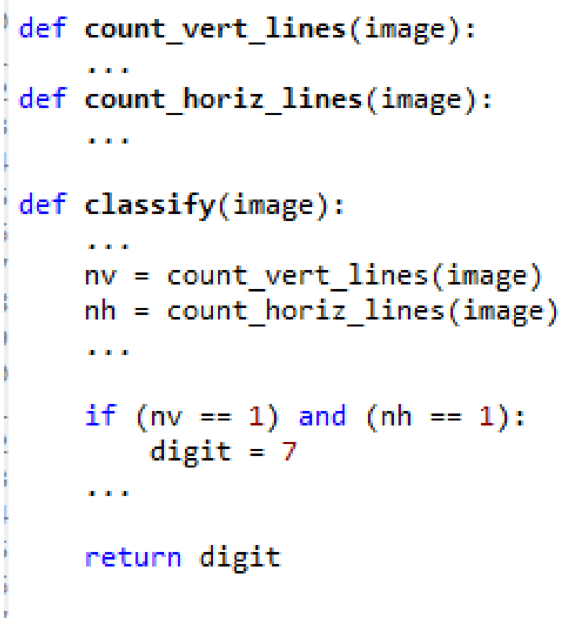
Images are 28 x 28 pixels

Example 1: Digit Recognition – Classical “Expert” Approach

- Idea: Use your knowledge about digits
  - You are an “expert” since you can do the task
  - So, you construct simple rules and code them
- Expert rule example: “Image is a digit 7 if…”:
  - There is a single horizontal line, and
  - There is a single vertical line
- Rule seems simple and reasonable
- But,…

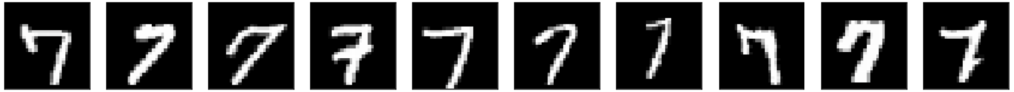


Images are 28 x 28 pixels

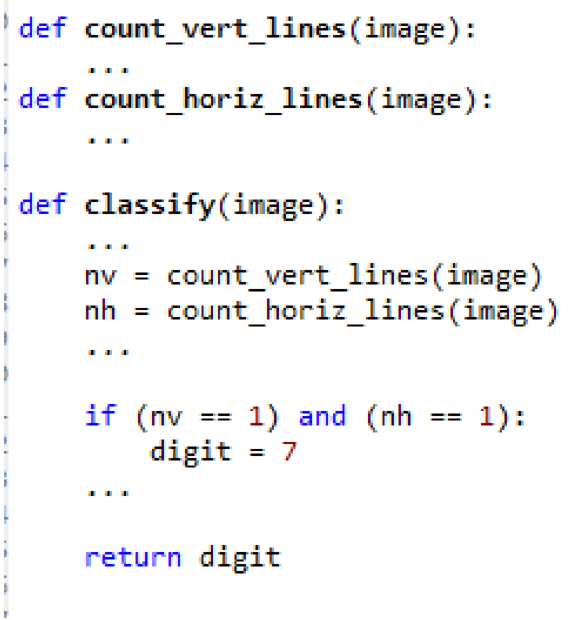


```
def count_vert_lines(image):
    ...
def count_horiz_lines(image):
    ...

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

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

    return digit
```

Example 1: Digit Recognition – Problems with Expert Rules



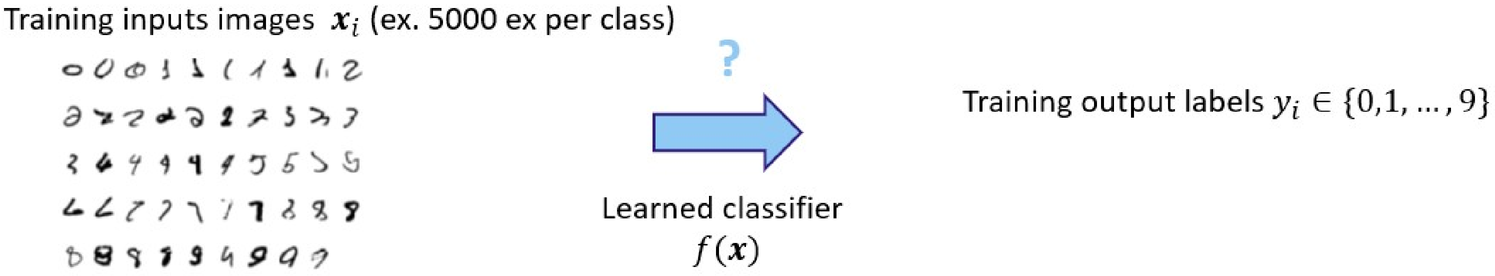
- □ Simple expert rule breaks down in practice
  - □ Hard to define a “line” precisely
  - □ Orientation, length, thickness, …
  - □ May be multiple lines…
- □ General problem: Difficult to code our knowledge
  - □ We can do the task
  - □ But it is hard to translate to simple mathematical formula



```
def count_vert_lines(image):
    ...
def count_horiz_lines(image):
    ...

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

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

    return digit
```

Example 1: Digit Recognition – Problems with Expert Rules

- ❑ Do not use your “expert” knowledge
- ❑ Learn the function from data!
- ❑ Supervised learning:
    - ❑ Get many labeled examples $(\boldsymbol{x}_i, y_i)$, i=1,…,N (Called the training data)
    - ❑ Each example has an input $\boldsymbol{x}_i$ and output $\boldsymbol{y}_i$
    - ❑ Learn a function $f(\boldsymbol{x})$ such that: $f(\boldsymbol{x})$ = $\boldsymbol{y}_i$ for “most” training examples



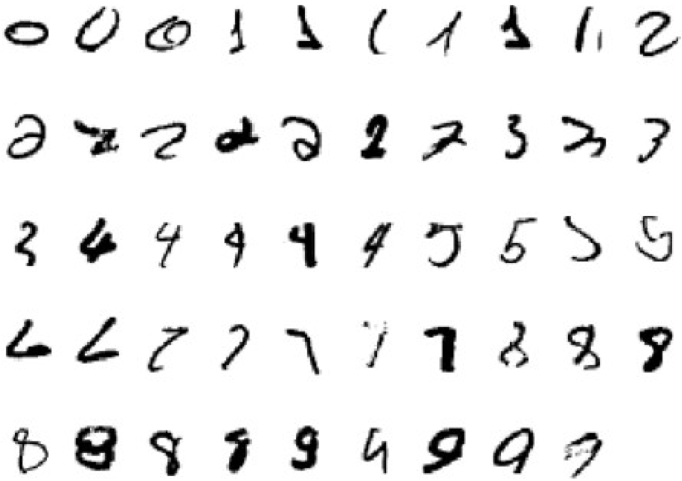
Training inputs images $\boldsymbol{x}_i$ (ex. 5000 ex per class)

Learned classifier $f(\boldsymbol{x})$

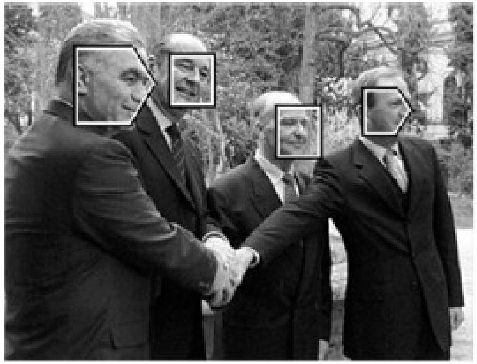
Training output labels $y_i \in \{0,1,\ldots,9\}$

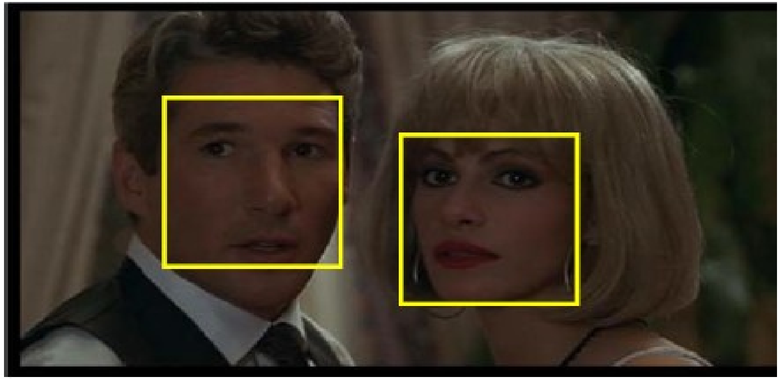
Example 1: Digit Recognition – ML Approach Benefits & Challenges

- ❑ Learned systems do very well on image recognition problems
  - ❑ On MNIST, current systems get <0.21% errors (as of 1/20/2018)
  - ❑ Used widely in commercial systems today (e.g. OCR)
  - ❑ Cannot match this performance with an expert system
- ❑ But there are challenges:
  - ❑ How do we acquire data? Someone must manually label examples.
  - ❑ How do we train an algorithm to learn from the data?
  - ❑ If a function works on training example, will it generalize on new data?
- ❑ This is what you will learn in this course



Example 2: Face Detection





- ❑ Problem: For each image region, determine if face or non-face
- ❑ More challenging than digit recognition
  - ❑ Even harder to describe a face via “rules” in a robust way

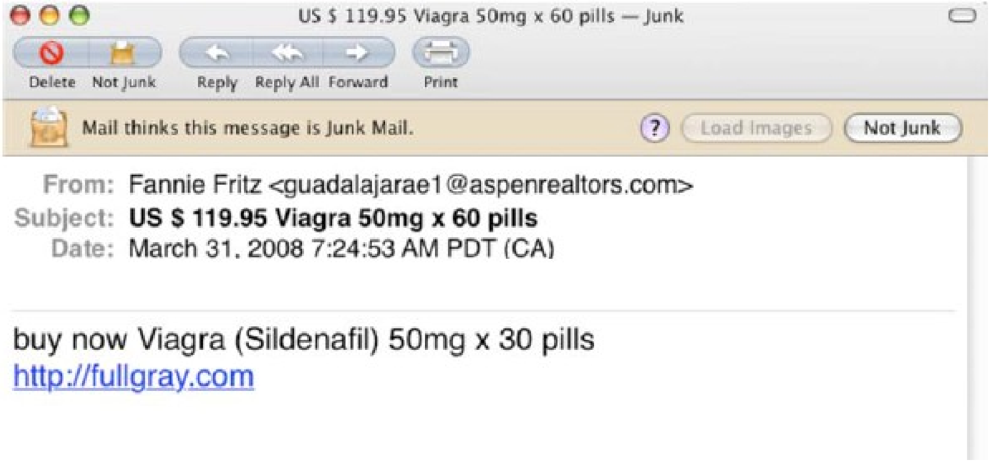
Example 2: Face Detection - Supervised Learning Approach

- ❑ Data: Get large number of face and non-face examples
- ❑ Typical early dataset
  - ❑ 5000 faces (all near frontal, vary age, race, gender, lighting
  - ❑ 10^8 non faces
- ❑ Train an algorithm to learn the classification rules/function
  - ❑ The function maps image to binary value “face” or “non-face”
  - ❑ For good performance, functions may be complex
  - ❑ Many parameters

Example 3: Spam Detection

- Classification problem:
  - Is email junk or not junk?
- For ML, must represent email numerically
  - Common model: bag of words
  - Enumerate all words, $i=1,\ldots,N$
  - Represent email via word count

    x_i = num instances of word i
- Challenge:
  - Very high-dimensional vector



Mail thinks this message is Junk Mail.

From: Fannie Fritz <guadalajarae1@aspenrealtors.com>
Subject: **US $ 119.95 Viagra 50mg x 60 pills**
Date: March 31, 2008 7:24:53 AM PDT (CA)

buy now Viagra (Sildenafil) 50mg x 30 pills
http://fullgray.com

Machine Learning in Many Fields

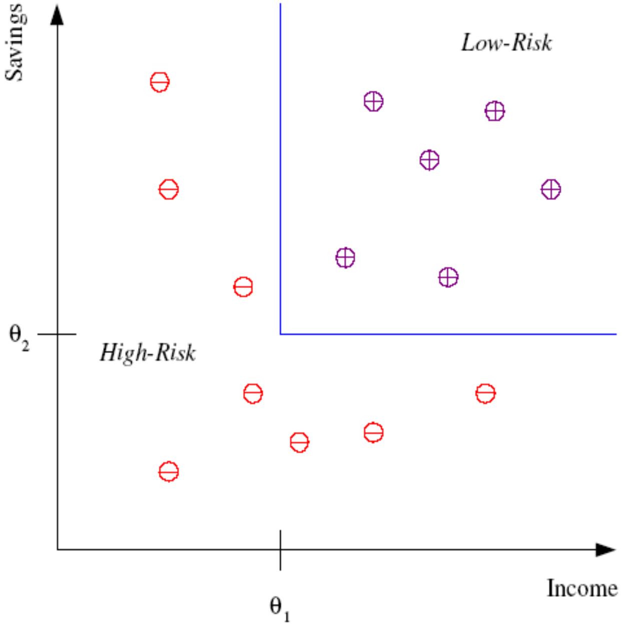
- ❑ Retail: Market basket analysis, Customer relationship management (CRM)
- ❑ Finance: Credit scoring, fraud detection
- ❑ Manufacturing: Control, robotics, troubleshooting
- ❑ Medicine: Medical diagnosis
- ❑ Telecommunications: Spam filters, intrusion detection
- ❑ Bioinformatics: Motifs, alignment
- ❑ Web mining: Search engines

Outline

- Basics of Programming
  - Python loops and data structures
  - Scientific computational package – NumPy
  - Data visualization
- What is machine learning?
- Types of machine learning algorithms
  - Classification and regression
- Why the hype today?

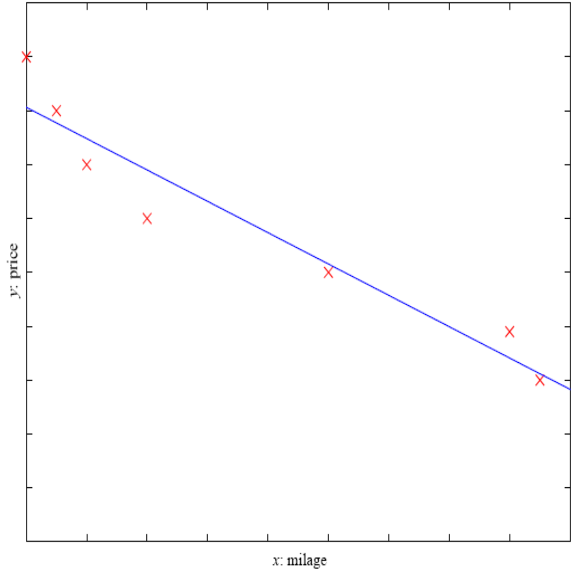
Classification

- ❑ Example: Credit score
- ❑ Determine if customer is high-risk or low-risk
- ❑ Select features:
  - ❑ Example: Income & Savings
  - ❑ Represent as a vector x=(x_1,x_2)
- ❑ Learn a function from features to target
  - ❑ Use past training data
- ❑ Need to get this data



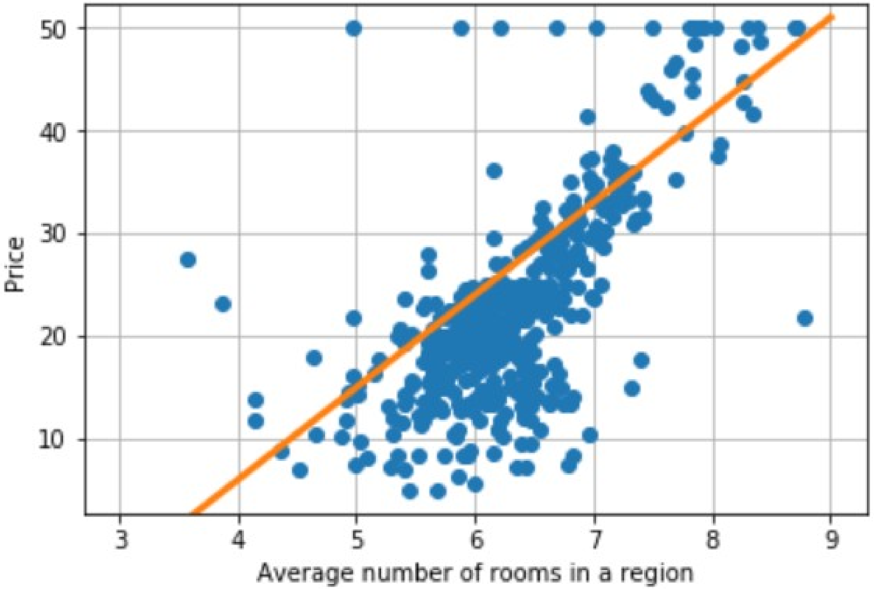
Regression

- ❑ Target variable y is continuous-valued
- ❑ Example:
  - ❑ Predict y = price of car
  - ❑ From x = mileage, size, horsepower, ..
  - ❑ Can use multiple predictors
- ❑ Assume some form of the mapping
  - ❑ Ex. Linear: $y = \beta_0 + \beta_1 * x$
  - ❑ Find parameters β_0, β_1 from data



Regression Example – In Google Colab Notebook

- ❑ The Boston housing data set was collected in the 1970s
- ❑ Predict housing prices
- ❑ Many possible predictors:
  - ❑ Crime
  - ❑ Areas of non-retail business in the town
  - ❑ Age of people who own the house



Outline

- ❑ Basics of Programming
  - ❑ Python loops and data structures
  - ❑ Scientific computational package – NumPy
  - ❑ Data visualization
- ❑ What is machine learning?
- ❑ Types of machine learning algorithms
  - ❑ Classification and regression
- ❑ Why the hype today?

What ML is Doing Today?

- Autonomous driving
- Jeopardy
- Very difficult games: Alpha Go
- Machine translation

- Many, many others…

Why Now?

- ❑ Machine learning is an old field
  - ❑ Much of the pioneering statistical work dates to the 1950s
- ❑ So what is new now?
- ❑ Big Data:
  - ❑ Massive storage. Large data centers
  - ❑ Massive connectivity
  - ❑ Sources of data from Internet and elsewhere
- ❑ Computational advances
  - ❑ Distributed machines, clusters
  - ❑ GPUs and hardware

Summary

- ❑ Fundamentals of Python programming language
- ❑ Familiarity with NumPy and Pandas
- ❑ Provide examples of Machine Learning used today
- ❑ Given a new problem, qualitatively describe how a machine learning can be used
  - ❑ Formulate a potential machine learning task
  - ❑ Identify the data needed for the task
- ❑ Classify a machine learning task
  - ❑ Regression vs. Classification
- ❑ Identify the predictors and target variables
- ❑ Determine the role of expert knowledge in the task vs. data driven learning

Thank You!