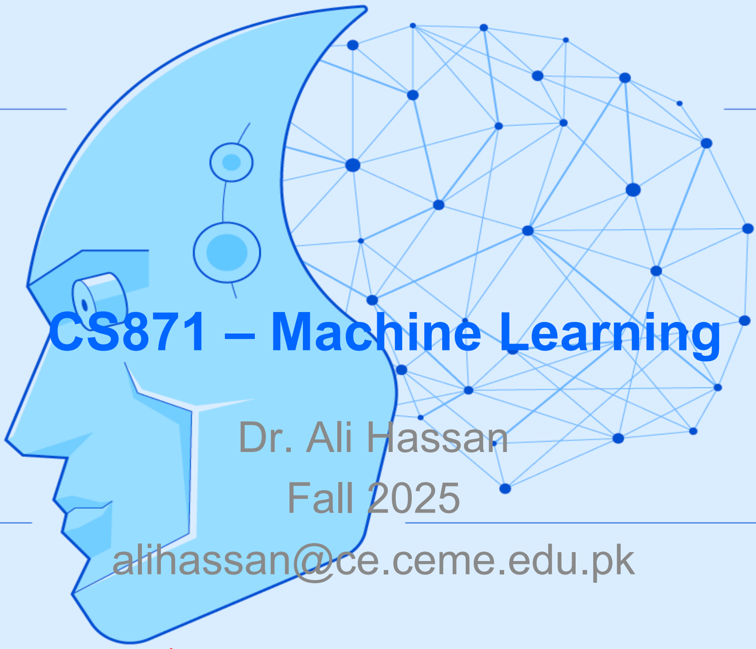


CS871 – Machine Learning

Dr. Ali Hassan
Fall 2025
alihassan@ce.ceme.edu.pk

1

Outline

- Previous Lecture
  - Working example – Pipeline of ML
  - **Difference between Supervised vs Unsupervised Learning**
  - ChatGPT

2

Outline

- This Lecture
  - Linear Algebra
  - Linear Regression
    - One Variable

Announcements

- Assignment # 1
  - Wednesday, 24 September 2024, 11:59 PM
  - 32/38 Submitted

Dr Ali Hassan

- Create an image of Pakistan’s football team lifting the FIFA World Cup trophy in a packed stadium. Players celebrate joyfully, surrounded by a euphoric crowd giving a standing ovation. Flags wave and confetti falls, capturing the moment of national pride and unity in a vibrant, electric atmosphere.
- Tool: ChatGPT

5

Dr Ali Hassan

- generate an AI image of tech hub in Muzaffarabad city , the time of the day should be late afternoon, with the sun casting a warm glow on the buildings. The tech hub should have a sleek, futuristic design and be surrounded by greenery and mountains. Multiple smaller buildings and offices representing various tech companies. These should be modern and well-maintained, with company logos e.g. Google, Meta, Facebook, Microsoft, Amazon and Yahoo prominently displayed. People including tech professionals, entrepreneurs, and students, walking around, entering and exiting buildings, and working in offices. The environment should be futuristic such as drones flying overhead, people using laptops and smartphones, and futuristic looking vehicles.
- Tool used: DALL-E 3

6

Dr Ali Hassan

- In 2035, a groundbreaking AI award ceremony takes place at the National University of Sciences and Technology (NUST) in Islamabad, where a young AI pioneer from Pakistan is honored with the prestigious Turing Award. His pioneering research in Explainable and Ethical Artificial Intelligence has transformed the interaction between humans and AI systems, emphasizing transparency and ethical governance. The ceremony features holographic display, merging advanced technology with classical architecture to highlight "NUST" and "TURING AWARD 2035" in backdrop. This event, attended by leading global AI researchers and scientists, marks a new era in AI, celebrated in a luxuriously decorated auditorium with attendees from around the world applauding this historic achievement.
- Credits: Dall-E (OpenAI)

Dr Ali Hassan

- Prompt: Draw an image of spiderman washing his clothes in pakistan sitting by Indus river while local kids are teasing him.
- sourse: Leonardo AI

- If Faisal Mosque was built in Saudi Arabia.
- Tool used: (ChatGPT)
- M. Ibtehaj Ahmad (MS23MTS)

9

- Description text: Make a picture of a boy sitting at high pastures of gilgit baltistan in Pakistan, surrounded by Yakk and a dog, eating local cherries along with his father and snowy mountains in the background
- Tool: Meta AI
- Regards Raheel Mumtaz

10

- Generate image of EME college where in classroom air-conditions are not working proper and due to high temperature students are sweaty and they are so frustrated from the management.

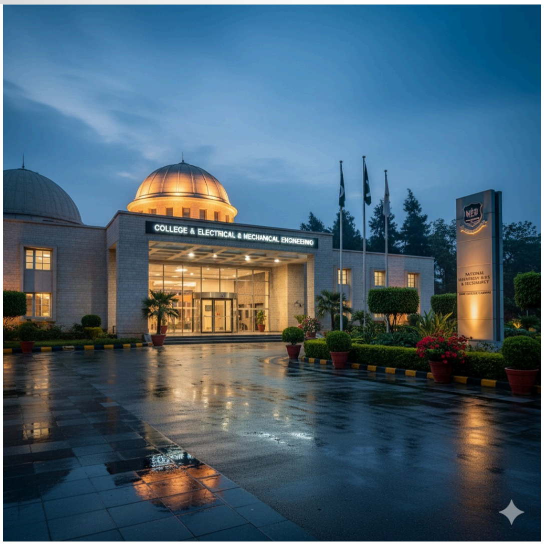
- Tool: ChatGPT

11

- Apple CEO Tim Cook selling iPhones from a street cart in a lively Pakistani local bazaar. The cart is filled with neatly arranged iPhones, with fresh red apples placed in between them. Two local Pakistani customers in traditional clothing are examining the iPhones with curiosity. Tim Cook is smiling and talking to them in a friendly manner. The background shows colorfull market stalls, fruits, and shops, creating an authentic South Asian bazaar atmosphere.”submitted by: Sheharyar Ahmed (REG NO: 500416

12

- { "scene": "main entrance of College of Electrical & Mechanical Engineering (CEME), NUST, Rawalpindi, Pakistan", "time": "midday with bright clear sunlight", "subjects": [{ "type": "entrance block", "facade": { "material": "light sandstone or pale brick", "details": "crisp mortar lines, subtle color variation, clean and smooth textures enhanced by sunlight", "lighting": "natural bright sunlight casting sharp, well-defined shadows and highlighting architectural details" },
- Name: Muhammad Soban Shaukat Tool: NANO BANANA - GEMINI AI IMAGE GENERATOR & PHOTO EDITOR BYGOOGLE DEEP MIND



Dr. Ali Hassan, DC&SE, College of E&ME, NUST 13

13

- Create a vibrant, colorful, bustling city market scene in a lively urban setting. Semi-realistic, hand-drawn, cinematic style inspired by graphic novels. Bright, warm, inviting colors with high saturation, avoiding dull tones. Lively but not overcrowded, with space between elements.Foreground:Diverse people interacting: women selling fresh fruits, baked goods, traditional foods; men and women shopping, chatting, strolling.Market stalls with woven baskets, ceramics, colorful spices, fresh fruits, artisan bread, musical instruments (drums, guitar).People in modern casual and traditional attire, representing cultural diversity.Hyper-detailed textures: basket weaves, shine on fruits, fabric patterns, metallic reflections.Middle-ground:
- Name: Zoha TazmenRegistration #: 538982AI: Gemini

Dr. Ali Hassan, DC&SE, College of E&ME, NUST 14

14

- Generate a photo-realistic, high-stakes diplomatic scene: Prime Minister of Pakistan Shehbaz Sharif and Crown Prince of Saudi Arabia Mohammed bin Salman signing the Strategic Mutual Defence Agreement at Al-Yamamah Palace, Riyadh, in front of national flags of Pakistan and Saudi Arabia. They are surrounded by senior military and diplomatic officials (including Pakistan's Army Chief), with media, cameras, and world leaders watching in the background. Include bold text on the document: 'Any aggression against either nation shall be considered aggression against both.'

- Name: Muhammad UzairTool: Gemini 2.5 flash

15

- A hyper-realistic, cinematic story-based illustration of a young woman with curly hair at the center, radiating calm strength and love. Around her, different scenes flo together like a tapestry of her busy yet happy life: helping her younger sibling wit studies, giving emotional support; teachin in front of a classroom of students; caring lovingly for her parents as a daughter; working on her startup in a professional setting with a laptop and colleagues in a meeting; enjoying personal pleasures like baking bread in the kitchen, planting and tending to her garden, and riding her scootie with joy. All elements are interconnected with glowing golden light and soft transitions, symbolizing her mind full of worlds and her heart full of care. Th atmosphere is warm, detailed, and cinematic, blending realism with dreamlike storytelling. Tool: Gemini ,Nano Banana Roll no: 537724

16

17

18

DEDICATION

19

Nergis Mavalvala

- **Nergis Mavalvala** (born 1968)
- a Pakistani-American astrophysicist
- Professor of Astrophysics at the MIT
- Dean of the university's school of science
- Mavalvala is best known for her work on the detection of gravitational waves in the Laser Interferometer Gravitational-Wave Observatory (LIGO) project

20

LINEAR ALGEBRA

Matrix from 10th, 11th Classes Text Books

version: 1.1

CHAPTER 3

Matrices and Determinants

Animation 3.1: Addition of matrix
Source & Credit: elearn.punjab

Chapter 1

Matrix: An arrangement of numbers in the form of rows and columns in a square bracket is called a matrix" and is denoted by A,B,C,...

Order of the Matrix: If A is a matrix, then

order of A = ord (A)= No: of Rows × No: of Columns.

order of A = ord (A)= No: of Rows –by- No: of Columns.

Note that order of matrix is also called dimension or size

Example1. Write the number of rows and columns of following matrices and hence mention their orders.

i). $A=\begin{bmatrix} p & q \\ r & s \end{bmatrix}$ ii). $B=\begin{bmatrix} 3 & 4 & 7 \\ 5 & 6 & 8 \end{bmatrix}$

i). solution; Given $A=\begin{bmatrix} p & q \\ r & s \end{bmatrix}$

number of rows = 2
number of colums = 2
Hence order (A) = 2-by- 2

ii). solution; Given $B=\begin{bmatrix} 3 & 4 & 7 \\ 5 & 6 & 8 \end{bmatrix}$

number of rows = 2
number of colums = 3
Hence order (A) = 2-by- 3

Equal matrix: Let A and B are two matrix of the same order, are equal if their corresponding elements are

System of Linear Equations

System of Linear Equation

$2.0x + 4.0y + 6.0z = 18$

$4.0x + 5.0y + 6.0z = 24$

$3.0x + 1y - 2.0z = 4$

Matrix representation

$$A = \begin{bmatrix} 2.0 & 4.0 & 6.0 \\ 4.0 & 5.0 & 6.0 \\ 3.0 & 1.0 & -2.0 \end{bmatrix} \quad X = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad b = \begin{bmatrix} 18.0 \\ 24.0 \\ 4.0 \end{bmatrix}$$

Solution

$$A\,x = b$$

$$x = A^{-1}b$$

24

24

Why Matrices in Machine Learning

- **Feature representation:**
  - Each row represents a sample and each column represents a feature or attribute
- **Mathematical operations:**
  - Machine learning algorithms involves mathematical operations like matrix multiplications and dot products.
- **Linear algebra operations:**
  - 2-D matrices are compatible with linear algebra operations such as addition, subtraction, and factorizations like SVD and Eigen value decomposition
- **Compatibility with libraries and frameworks:**
  - Many machine learning libraries and frameworks are optimized for 2-D matrix computations such as NumPy and TensorFlow
- **Image and text data:**
  - 2-D matrices are particularly useful for working with image and text data

Dr. Ali Hassan, DC&SE, College of E&ME, NUST 25

25

Data Representation

- AI Models cannot directly see, hear or sense
- Input is to be represented as numbers
- Data is stored in tables / matrices
- Each row represents a sample / instance
- Each column an attribute / feature of the samples

Example: Apartment Price

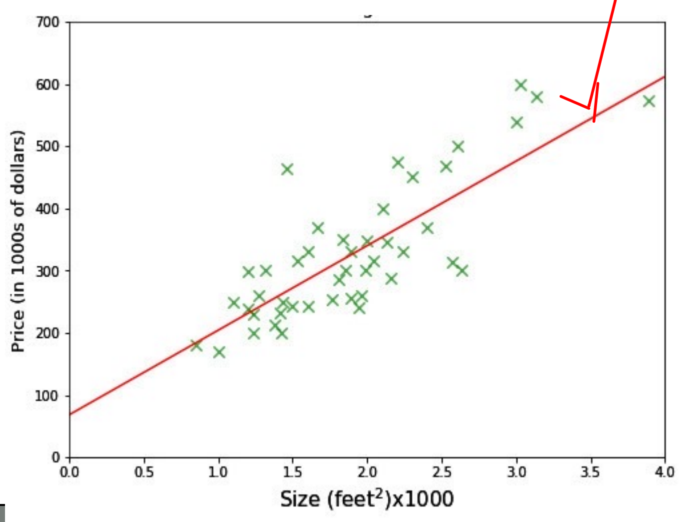
$y = mx + c$

1

$x \rightarrow y$

Sr No	Size	Price in $
0	2104	399900
1	1600	329900
2	2400	369000
3	1416	232000
4	3000	539900

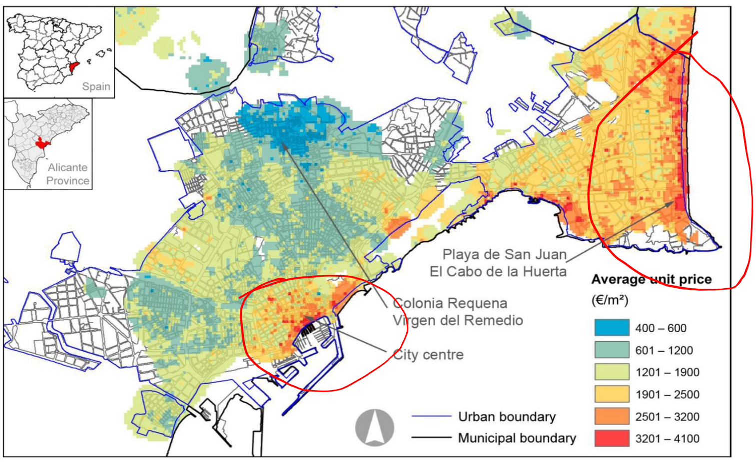
Data Fed to the Model



28

28

A Real Model is a Lot More Complex



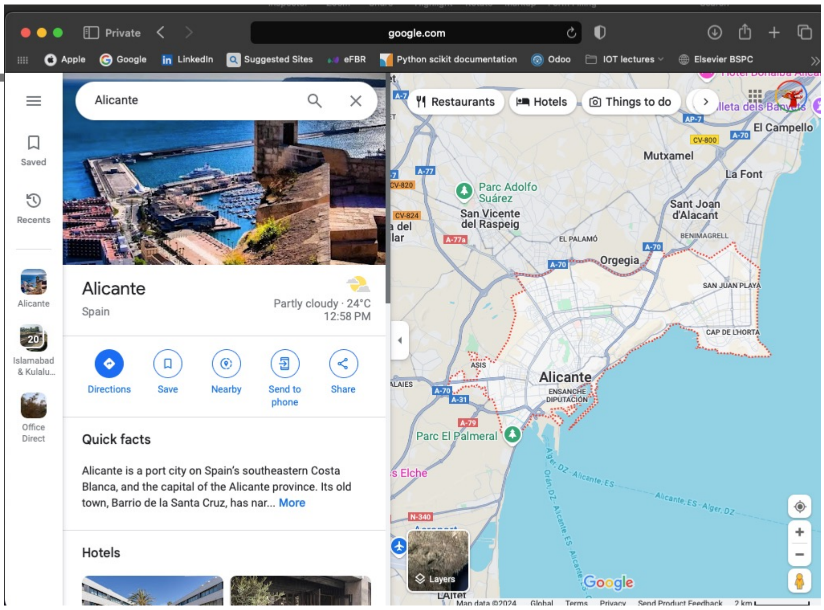
Housing Price Prediction Using Machine Learning Algorithms in COVID-19 Times

https://www.mdpi.com/2073-445X/11/11/2100

Spatial distribution of sales prices of multifamily housing in the city of Alicant

29

29



Dr. Ali Hassan, DC&SE, College of E&ME, NUST

30

30

Features

- Distances between properties and public services
  - Educational centers
  - Green areas
  - Coast
- The distances were calculated by network
  - Using a pre-established layout of streets and crossroads, the reality of the urban network
- Incorporating the presence of bridges or tunnels.
- Area of influence of 150 m (circle neighborhood)
- House features
  - Balcony availability
  - View
- Sports facilities

31

31

Example: Recommender System for Movies

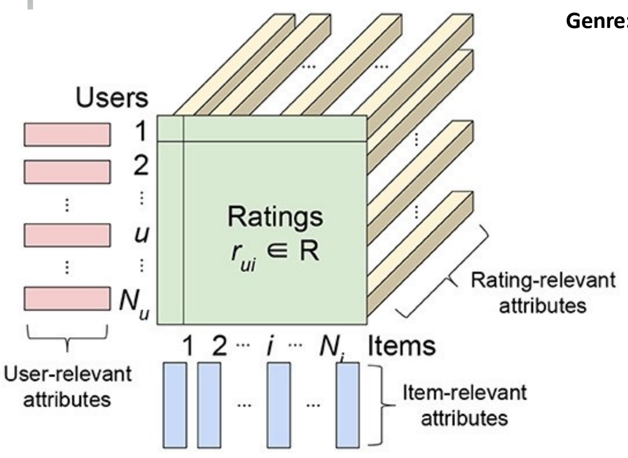
2

	Movie 1	Movie 2	Movie 3	…	Movie N
User 1	4	5	3	…	2
User 2	3	4	-	…	5
User 3	-	2	4	…	3
…	…	…	…	…	…
User M	5	-	2	…	4

32

32

Recommender System Data Representation



Genres, Starring, Directors, Year, Age Category, Gender, Rank

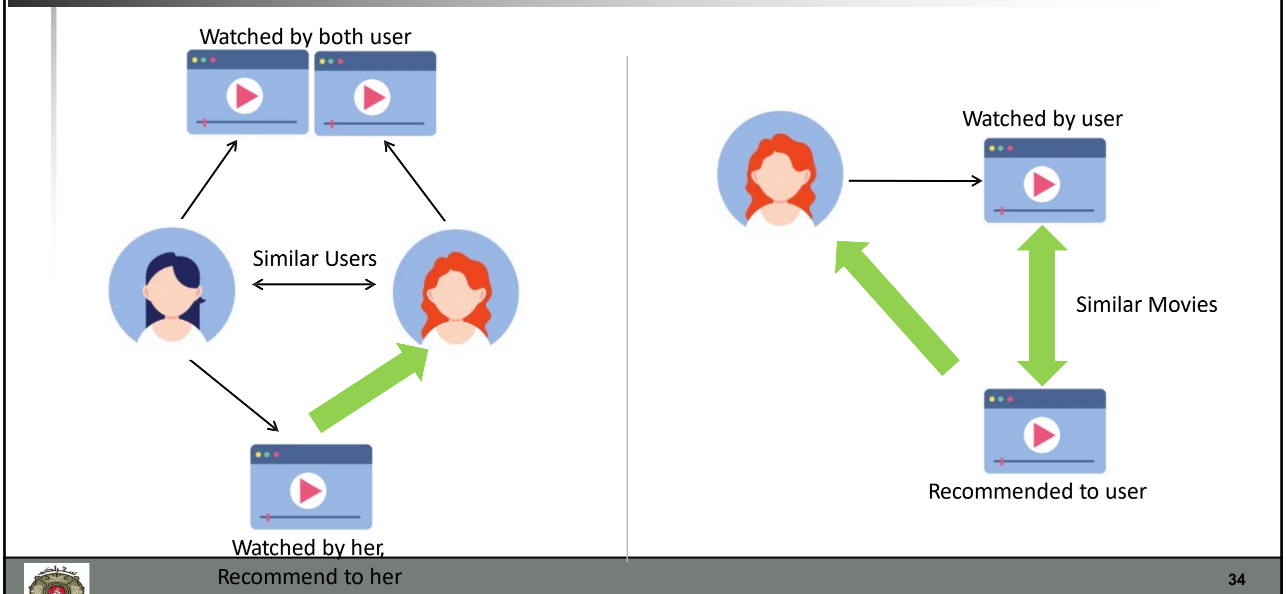
Genre: a style or category of art, music, or literature.

- **Action**
- Comedy
- Drama
- Fantasy
- Horror
- Mystery
- Romance
- **Thriller**
  - Crime thriller
  - Disaster thriller
  - Psychological thriller
  - Techno thriller
- War and military action
- Spy and espionage action
- Martial arts action
- Western shoot 'em up action
- Action hybrid genres

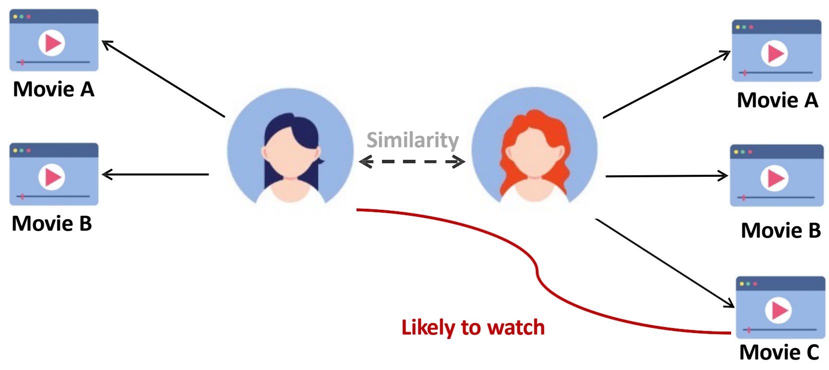
www.frontiersin.org/articles/10.3389/fdata.2019.00049/full

33

33



34



$$cosine(x, y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

35

Example: Patient Data for Treatment

3

	Age	Gender	Blood Pressure	Cholesterol	Diabetes	...	Treatment
Patient 1	52	Male	130/80	200	Yes	...	Medication A
Patient 2	38	Female	120/70	180	No	...	None
Patient 3	65	Male	140/90	220	Yes	...	Medication B
...	...	...	...	...	...	...	...
Patient N	45	Female	125/75	190	No	...	None

36

1

Image & Video
Representation

37

37

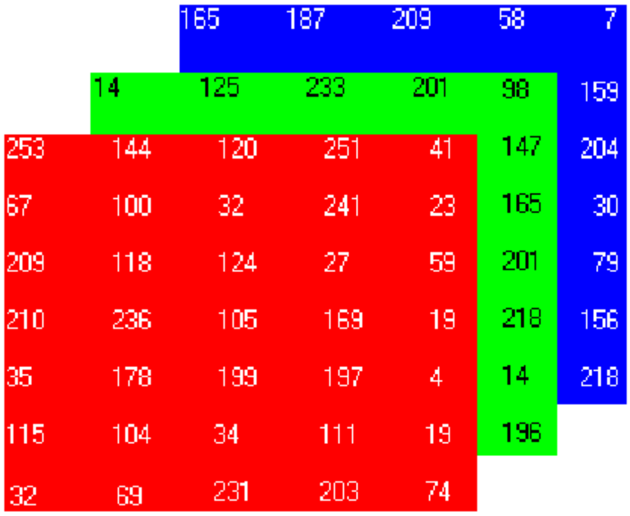
What we see

25 43 11 04 70 87 1 10 05 77 12 06 45 09 29 30 02
56 22 75 03 22 96 45 12 23 03 77 67 81 45 22 04 90 22 21
32 45 41 91 87 62 35 02 00 11 62 25 43 11 04 70 87 12 61
31 43 10 05 77 12 06 45 09 29 30 56 22 75 03 22 96 45 05
12 23 03 77 67 81 45 22 04 90 22 32 45 41 91 87 62 35 44
02 00 11 62 25 43 11 04 70 87 12 31 43 10 05 77 12 06 10
45 09 29 30 56 22 75 03 22 96 45 12 23 03 77 67 81 45 55
22 04 90 22 32 45 41 91 87 62 35 02 00 11 62 25 43 11 80
04 70 87 12 31 43 10 05 77 12 06 45 09 29 30 56 22 75 08
03 22 96 45 12 23 03 77 67 81 45 22 04 90 22 32 45 41
91 87 62 35 02 00 11 62 22 01 00 72 65 23 01 00 22 04 30
90 22 32 45 41 91 87 62 35 02 00 11 62 25 43 11 04 70 42
87 12 31 43 10 05 77 12 06 45 09 29 30 56 22 75 03 22 91
96 45 12 23 03 77 67 81 45 22 04 90 22 32 45 41 91 87 40
62 35 02 00 11 62 22 01 00 72 65 23 01 00 56 22 75 03 67
22 96 45 12 23 03 77 67 81 45 22 04 90 22 32 45 41

What we see

What computers see

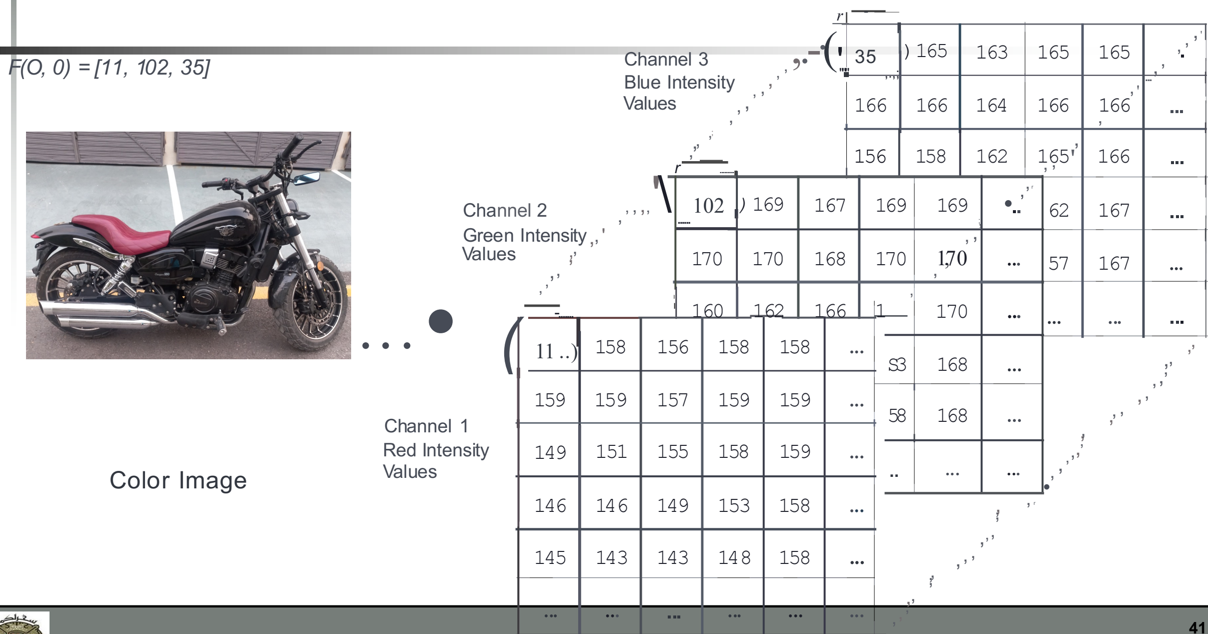
Color Image: RGB



40

40

RGB Channels



41

41

Go Run Terminal Window Help

import cv2.py — hw2_code

HW2_Spring2023_Student.ipynb | imageEdge | import cv2.py

Users > yourusername > Downloads > import cv2.py > ...

```python
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Load the color image with error handling
image_path = "/Users/yourusername/Downloads/TryThis.jpg"  # Replace with the path to your image

color_image = cv2.imread(image_path)

if color_image is None:
    print(f"Error: Unable to load the image at '{image_path}'. Please check the file path and image format."
else:
    # Split and display color channels (rest of the code remains the same)
    b, g, r = cv2.split(color_image)

    red_channel = np.zeros_like(color_image)
    red_channel[:, :, 2] = r  # Red channel

    green_channel = np.zeros_like(color_image)
    green_channel[:, :, 1] = g  # Green channel

    blue_channel = np.zeros_like(color_image)
    blue_channel[:, :, 0] = b  # Blue channel

    plt.figure(figsize=(10, 4))

    plt.subplot(131)
    plt.imshow(red_channel)
    plt.title("Red Channel")
    plt.axis("off")

    plt.subplot(132)
    plt.imshow(green_channel)
    plt.title("Green Channel")
    plt.axis("off")

    plt.subplot(133)
    plt.imshow(blue_channel)
    plt.title("Blue Channel")
    plt.axis("off")

    plt.tight_layout()
    plt.show()
```

Code Generation Using ChatGPT

write python code that takes a color image as input and shows the R, G and B color images

42

42

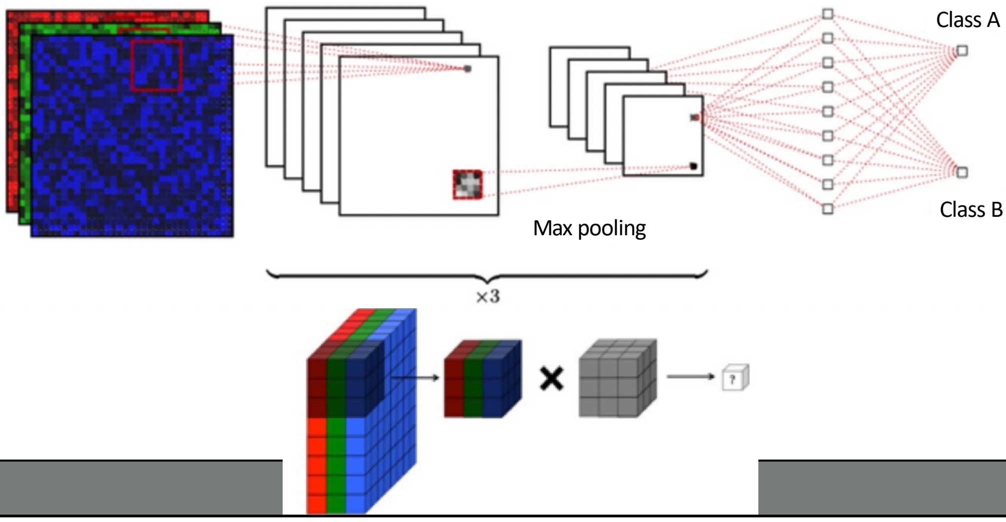
Original

Red Channel | Green Channel | Blue Channel

Dr. Ali Hassan, DC&SE, College of E&ME, NUST

43

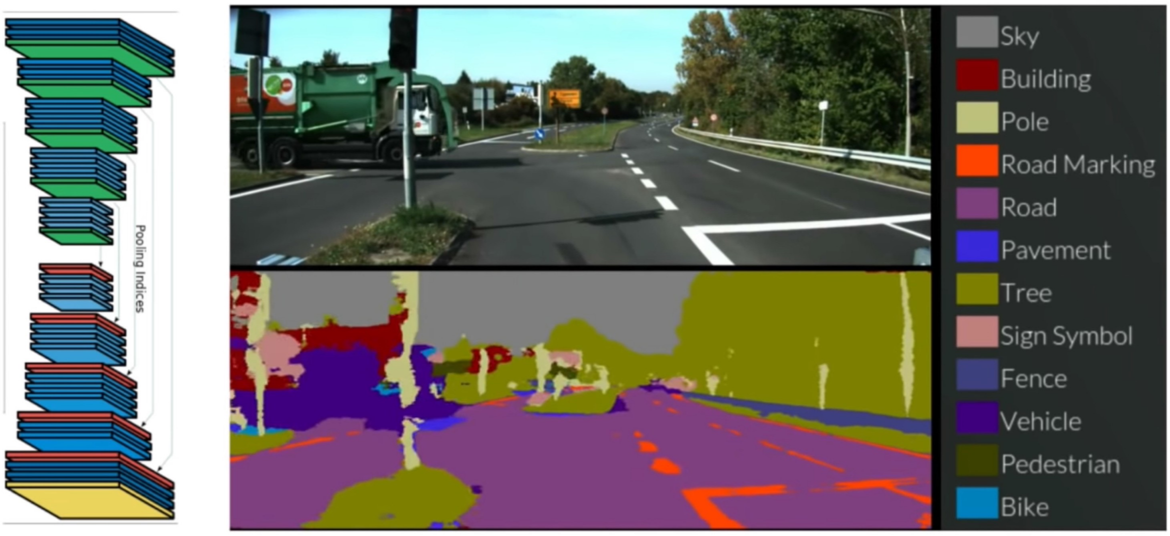
43



44

44

Scene Segmentation

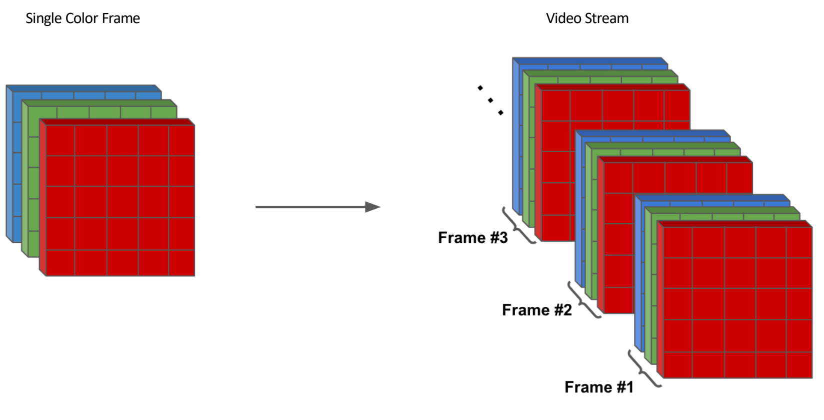


45

45

IMAGE & VIDEO REPRESENTATION

46



47

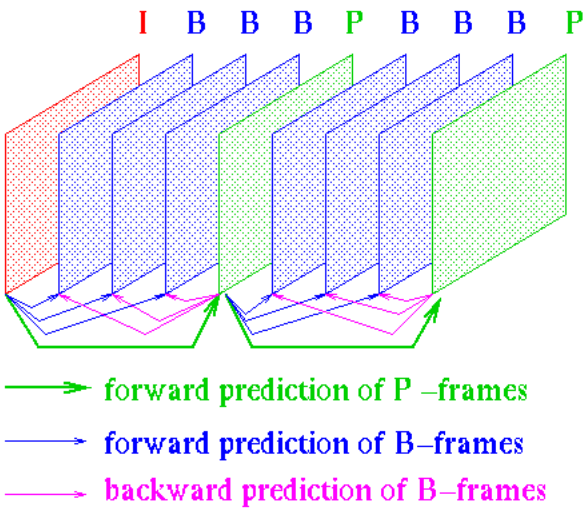
Video Compression

I-frames, P-frames, and B-frames are fundamental to video compression.

Intra-frame consists ONLY of macroblocks that use Intra-prediction / spatial redundancies

P-frame stands for Predicted Frame also uses temporal prediction with previously coded frame in addition to spatial prediction.

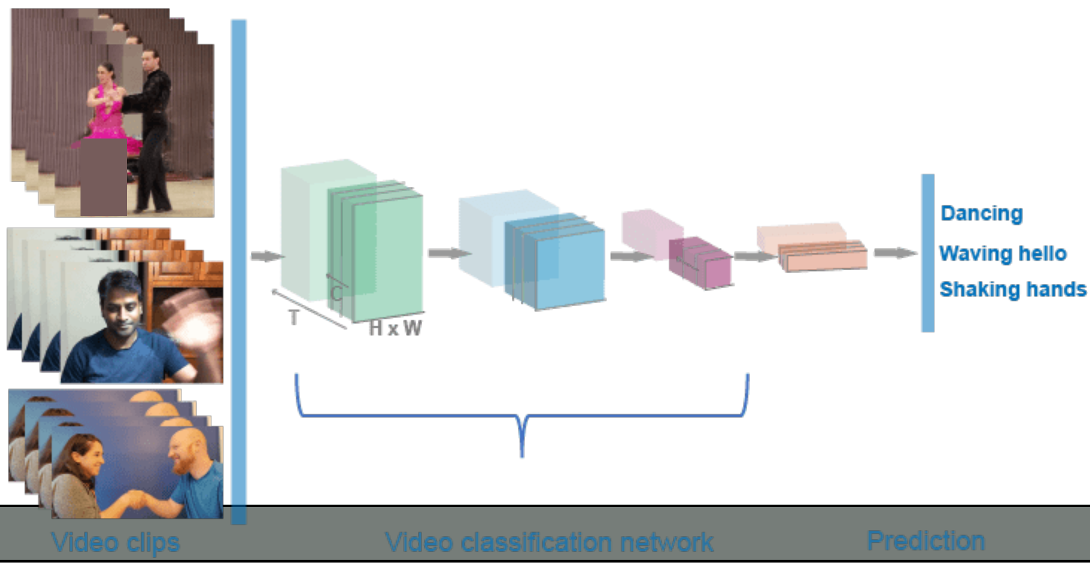
B-frame stands for Bidirectional Frame also uses forward frame in addition to previous coded frame for compression



48

48

Video Classification



49

49

DATA REPRESENTATION AND LINEAR ALGEBRA

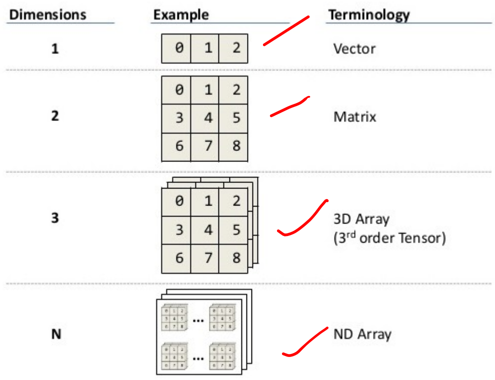
Scalar Vector Matrix

Scalar: 24

Vector: $\begin{bmatrix} 2 & -8 & 7 \end{bmatrix}$ row

or column $\begin{bmatrix} -6 \\ -4 \\ 27 \end{bmatrix}$

Matrix: $\begin{bmatrix} 6 & 4 & 24 \\ 1 & -9 & 8 \end{bmatrix}$ row(s) × column(s)



Dr. Ali Hassan, DCE, College of E&ME, NUST 52

52

Command Window

```
>> a = 24;
>> b = [2 -8 7]

b =

     2    -8     7

>> c = [-6;-4;27]

c =

    -6
    -4
    27

>> D = [6 4 24; 1 -9 8]

D =

     6     4    24
     1    -9     8

>> D(2,3)

ans =

     8

>> b(3)

ans =

     7

>> size(D)

ans =

     2     3
```

Workspace

Name	Value	Min	Max
a	24	24	24
ans	[2,3]	2	3
b	[2,-8,7]	-8	7
c	[-6;-4;27]	-6	27
D	[6,4,24;1,-9,8]	-9	24

53

53

```
leo@monsterpc2:~$ ipython
Python 2.7.6 (default, Jun 22 2015, 17:58:13)
Type "copyright", "credits" or "license" for more information.

IPython 2.3.0 -- An enhanced Interactive Python.
?         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.

In [1]: import numpy as np

In [2]: a = 24

In [3]: b = np.array([[2, -8, 7]])

In [4]: c = np.array([[-6],[-4],[27]])

In [5]: D = np.array([[6,4,24],[1,-9,8]])

In [6]: print(b)
[[ 2 -8  7]]

In [7]: print(c)
[[-6]
 [-4]
 [27]]

In [8]: print(D)
[[ 6  4 24]
 [ 1 -9  8]]

In [9]: D[1,2]
Out[9]: 8

In [10]: D.shape
Out[10]: (2, 3)
```

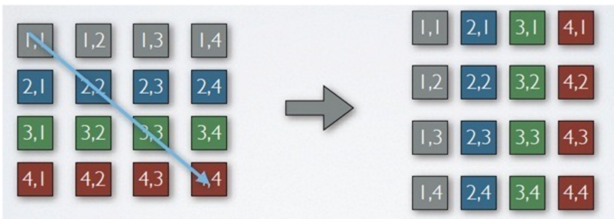
54

54

Matrix Operations

- Transpose

x^t



55

A

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

- $\begin{bmatrix} 1 & 2 \end{bmatrix}^T = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$
- $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^T = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$
- $\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}^T = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$

Addition/Subtraction

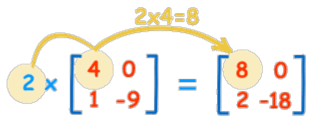
3+4=7

$$\begin{bmatrix} 3 & 8 \\ 4 & 6 \end{bmatrix} + \begin{bmatrix} 4 & 0 \\ 1 & -9 \end{bmatrix} = \begin{bmatrix} 7 & 8 \\ 5 & -3 \end{bmatrix}$$

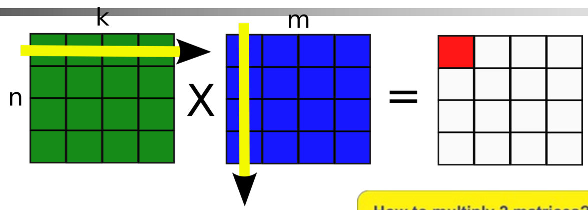
3-4=-1

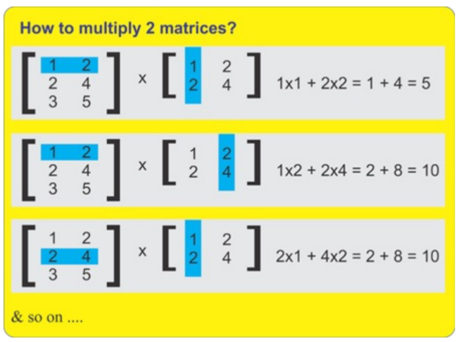
$$\begin{bmatrix} 3 & 8 \\ 4 & 6 \end{bmatrix} - \begin{bmatrix} 4 & 0 \\ 1 & -9 \end{bmatrix} = \begin{bmatrix} -1 & 8 \\ 3 & 15 \end{bmatrix}$$

Multiply by scalar



Matrix Multiplication





Types of Matrix

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

60

LINEAR REGRESSION

61

Housing Prices

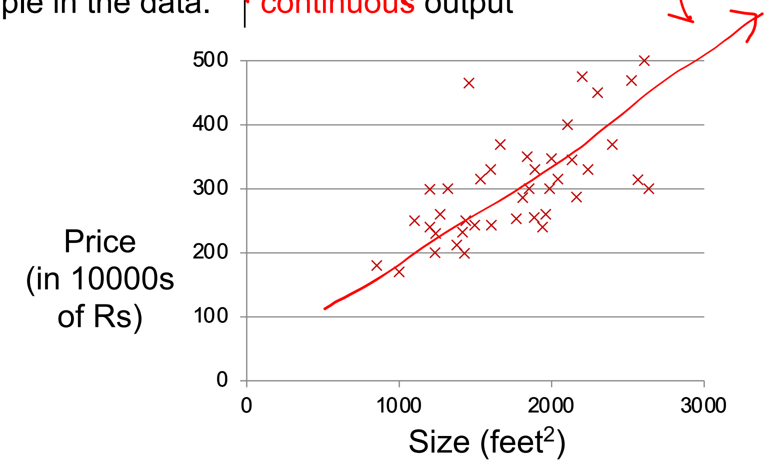
mx + c

<u>Supervised Learning</u>
Given the "right answer" for each example in the data.

w_0 w_1

<u>Regression Problem</u>
Predict real-valued or continuous output

θ



Training set of housing prices

$\theta_0 + \theta_1$

$X =$ θ

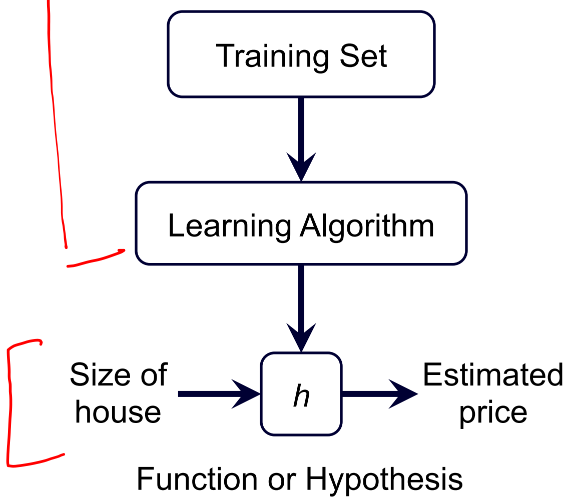
Size in feet² (x)	Price (PKR) in 10000's (y)
2104	460
1416	232
1534	315
852	178
…	…

Y

Notation:
m = Number of training examples
x's = "input" variable / features
y's = "output" variable / "target" variable
(x,y) = one training example
$(x^{(i)}, y^{(i)})$ = i^{th} training examples

x → y

Tr



64

Training Set

Size in feet^2 (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…

Hypothesis: $h_\theta(x) = \theta_0 + \theta_1 x = \hat{y}$ (prediction)

θ_i's: Parameters

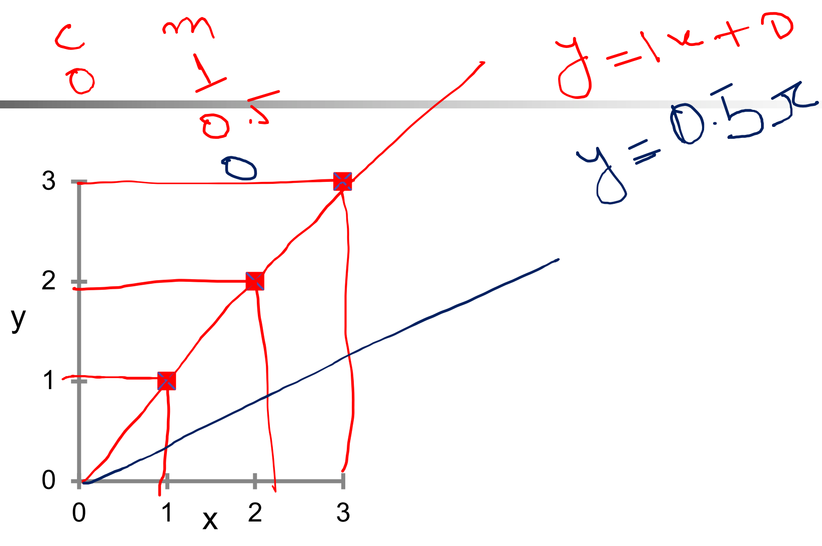
How to choose θ_i's ?

65

Idea: Choose θ_0, θ_1 so that $h_\theta(x)$ is close to y for our training examples (x, y)

- Question: What should be the cost / Loss function?

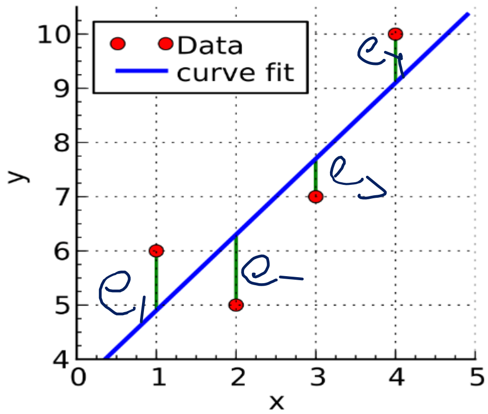
66



67

Finding out the best line

The line which minimizes the sum of square errors



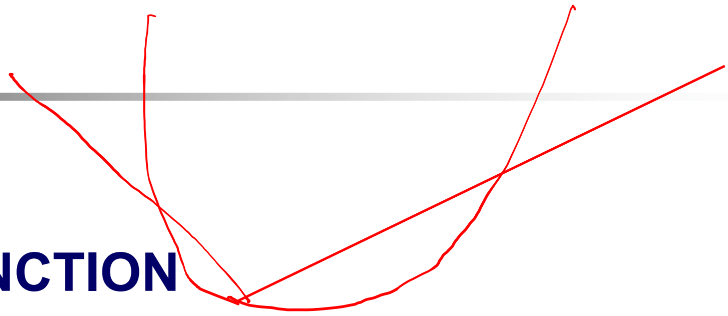
Finding out the best line

- $e1 = y1 - (\theta_1 x_1 + \theta_0)$
- $e2 = y2 - (\theta_1 x_2 + \theta_0)$
- $e3 = y3 - (\theta_1 x_3 + \theta_0)$
- $e4 = y4 - (\theta_1 x_4 + \theta_0)$

- Sum of Square Errors
- $SE = (y1 - (\theta_1 x_1 + \theta_0))^2 + (y2 - (\theta_1 x_2 + \theta_0))^2 + (y3 - (\theta_1 x_3 + \theta_0))^2 + (y4 - (\theta_1 x_4 + \theta_0))^2$

- Cost Function (J) = $\frac{1}{n}\sum_i^n \left(y^i - (\theta_1 x^i + \theta_0)\right)^2$

COST FUNCTION



$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

Mean Squared Error

70

COST/LOST FUNC – A STATISTICAL PERSPECTIVE

71

Cost / Loss Function

- Lets define non negative loss function
  - squared diff bw predicted value and actual value
- $L(\hat{y}, y) = L(h_\theta(x), y)$
- The risk associated with hypothesis *h*(*x*) is then defined as the expectation of the loss function:

$$R(h) = E[L(h(x), y)]$$
$$= \int L(h(x), y)dP(x, y)$$

- Discrete form of ERM
- $R_{emp}(h) = \frac{1}{m}\sum_i L(h(x), y)$
- Lets choose L as square loss …
- And you can start seeing similarities between *J* and R_{emp}

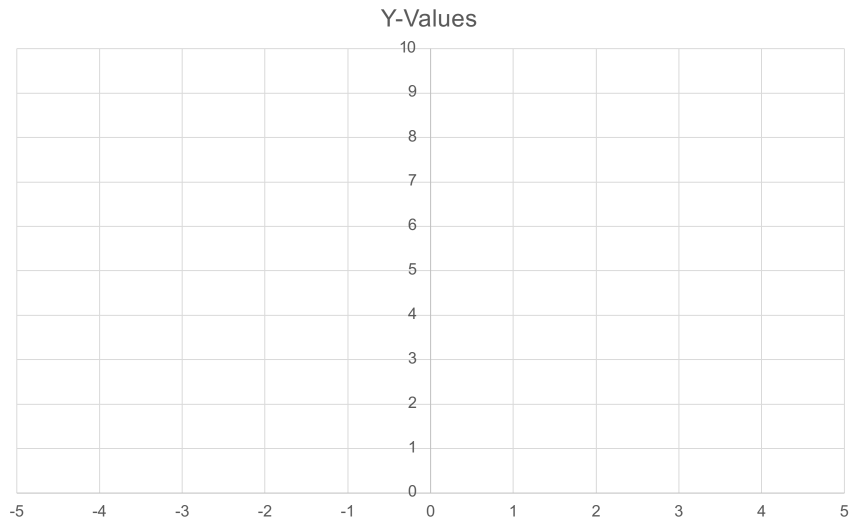
$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

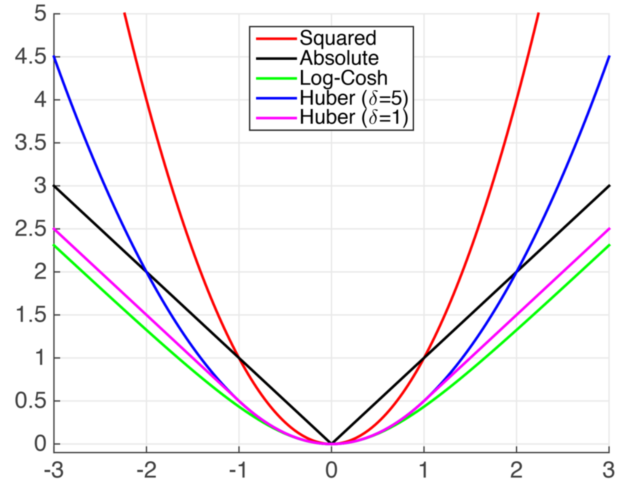
Ultimate Goal

- The ultimate goal of a learning algorithm is to find a hypothesis *h* among a fixed class of functions *H* for which the risk *R*(*h*) is minimal:

$$h^* = \arg\min_{h \in H} R(h)$$

Plots of Different Loss Functions





Dr. Ali Hassan, DC&SE, College of E&ME, NUST 76

76

More Details

- Lecture notes on Empirical Risk Minimization

Machine Learning – Lecture Notes

Instructor: Dr. Ali Hassan

1 Empirical Risk Minimization

Empirical risk minimization (ERM) is a principle in statistical learning theory that is used to give theoretical bounds on the performance of learning algorithms. Consider the following situation, which is a general setting of many supervised learning problems. We have two spaces of objects $\boldsymbol{X}$ and $\boldsymbol{Y}$ and we would like to learn a function $h : \boldsymbol{X} \rightarrow Y$ which outputs an object $y \in Y$, given $x \in X$. To do so, we have at our disposal a training set of a few examples $(x_1, y_1), \ldots, (x_m, y_m)$ where $x_i \in X$ is an input and $y_i \in Y$ is the corresponding response that we wish to get from $h(x_i)$.

To put it more formally, we assume that there is a joint probability distribution $P(x, y)$ over $\boldsymbol{X}$ and $\boldsymbol{Y}$, and that the training set consists of m instances $(x_1, y_1), \ldots, (x_m, y_m)$ drawn identically and independently drawn (i.i.d) from $P(x, y)$. Note that the assumption of a joint probability distribution allows us to model

77

77

COST FUNCTION

78

78

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

Uni-variate Cost Function

COST FUNCTION

79

79

Hypothesis:

$$h_\theta(x) = \theta_0 + \theta_1 x$$

Parameters:

$$\theta_0, \theta_1$$

Cost Function:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$

Simplified

$$h_\theta(x) = \theta_1 x$$

Limited function for which the solution has to pass through 0,0

$$\theta_1$$

$= (\theta_1 x^i - y)^2$

$$J(\theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

$$\underset{\theta_1}{\text{minimize}}\ J(\theta_1)$$

Uni-variate Linear Regression

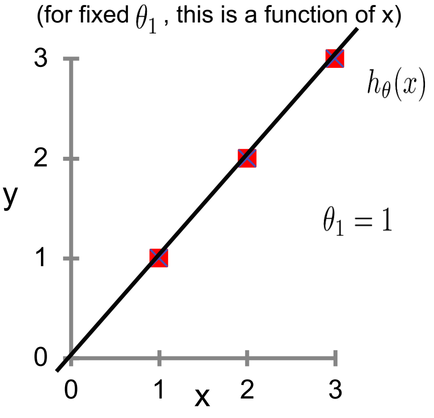
Dr. Ali Hassan, DC&SE, College of E&ME, NUST 80

80

Ali

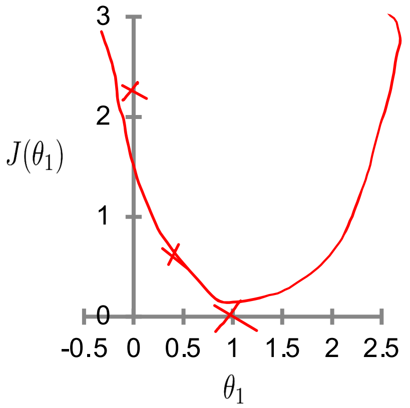
$h_\theta(x)$

(for fixed θ_1 , this is a function of x)



$J(\theta_1)$

(function of the parameter θ_1)



Dr. Ali Hassan, DC&SE, College of E&ME, NUST 81

81

Take Home Message

- Minimising $J(\theta)$ will find the best fitting $h(x)$ OR

Value of θ_1 that minimises the cost function will minimise the prediction error

Multi-Variate

COST FUNCTION

Hypothesis: $h_\theta(x) = \theta_0 + \theta_1 x$

Parameters: θ_0, θ_1

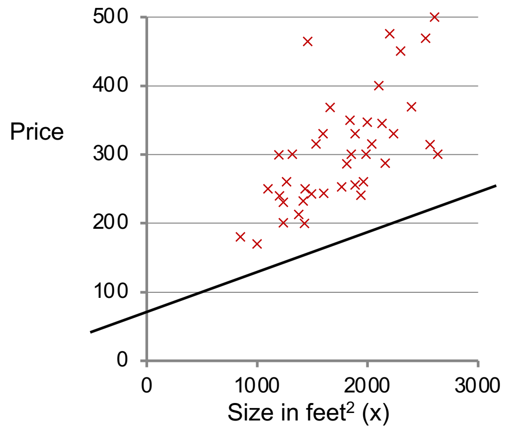
Cost Function: $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}} \; J(\theta_0, \theta_1)$

4

$h_\theta(x)$

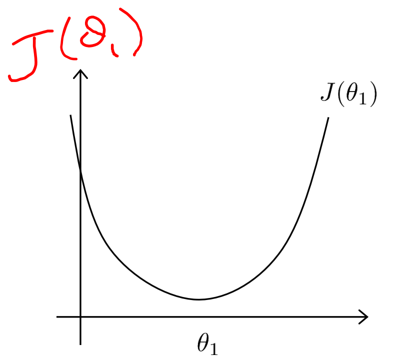
(for fixed θ_0, θ_1 , this is a function of x)



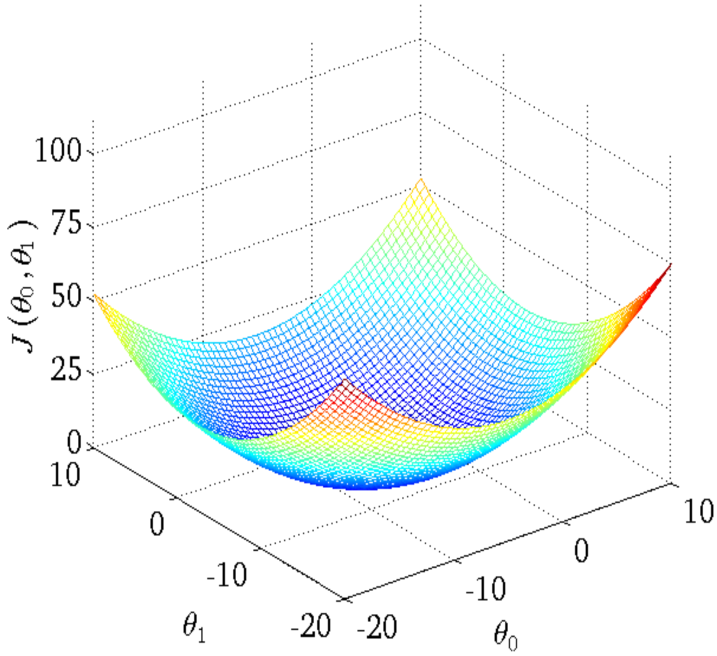
$h_\theta(x) = 50 + 0.06x$

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



The Bi-Variate cost function will look like a bowl or a cup



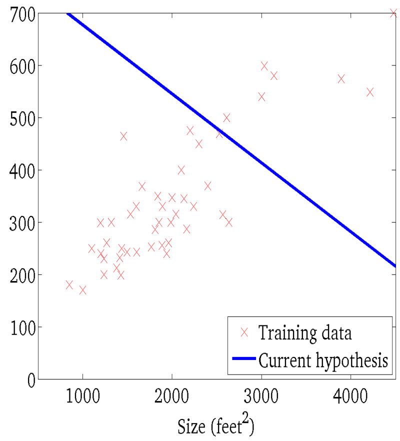
86

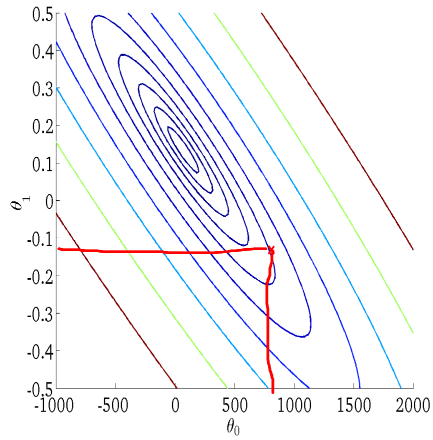
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)





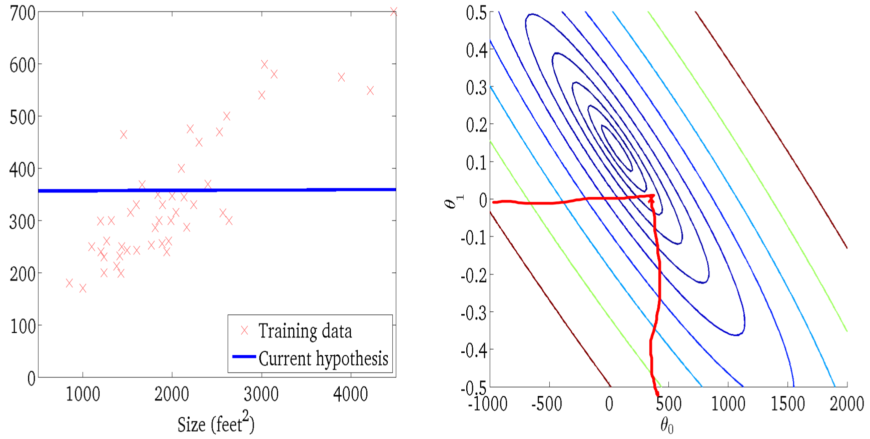
87

$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



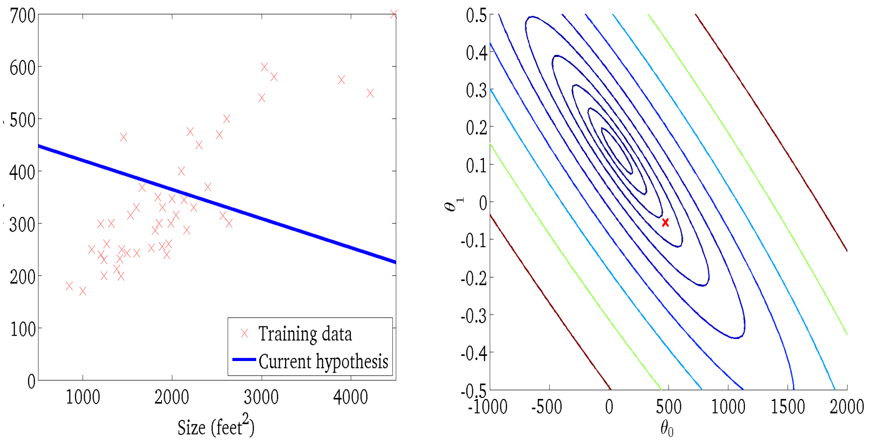
88

$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$

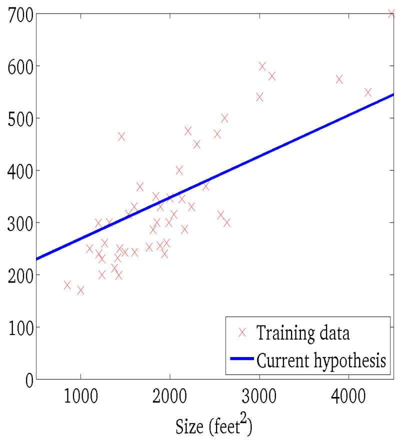
(function of the parameters θ_0, θ_1)



89

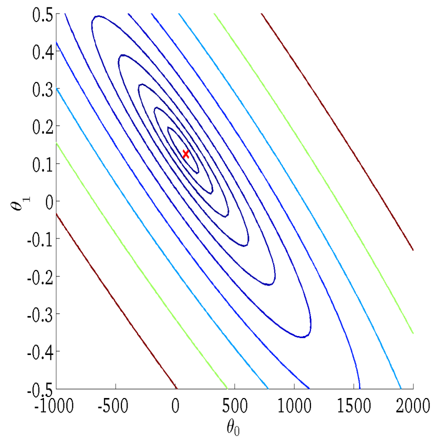
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



Minimising Cost Function

- We want to develop an automatic algorithm that finds the minimal solution of the cost function

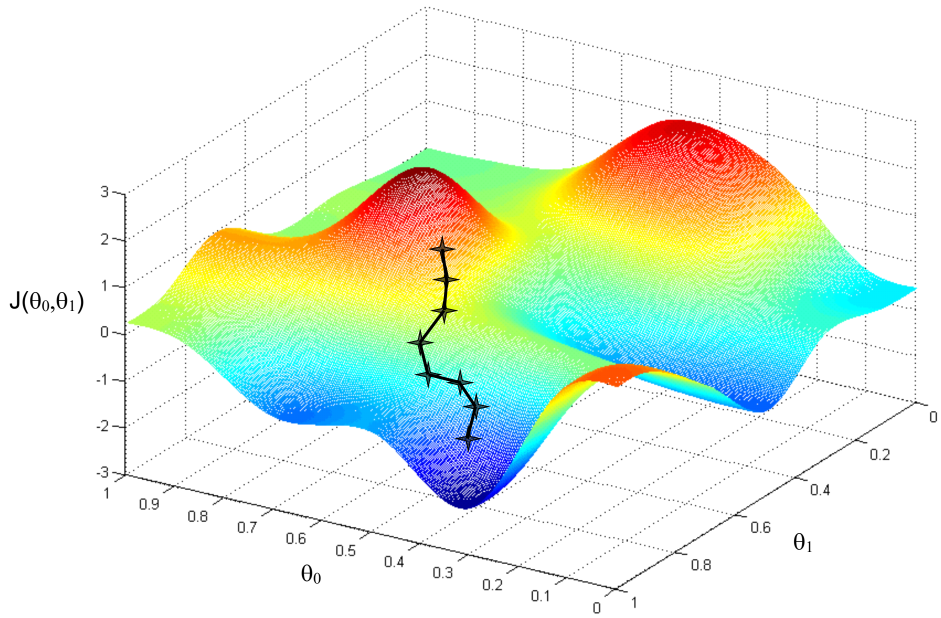
GRADIENT DESCENT

Have some function $J(\theta_0, \theta_1)$

Want $\min_{\theta_0, \theta_1} J(\theta_0, \theta_1)$

Outline:

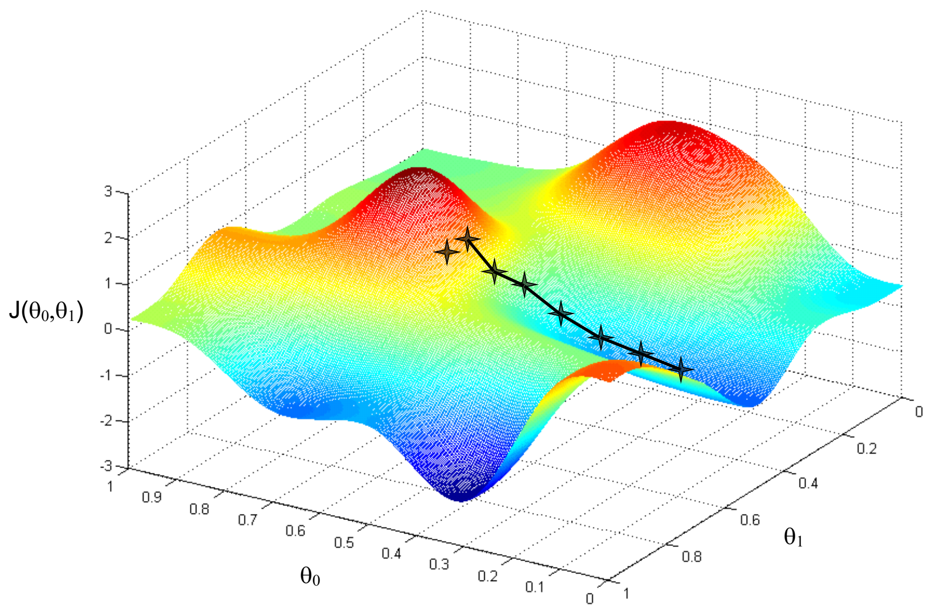
- Start with some θ_0, θ_1
- Keep changing θ_0, θ_1 to reduce $J(\theta_0, \theta_1)$ until we hopefully end up at a minimum



Dr. Ali Hassan, DC&SE, College of E&ME, NUST

94

94



Dr. Ali Hassan, DC&SE, College of E&ME, NUST

95

95

Gradient descent algorithm

$\alpha = 1$

repeat until convergence {

New updated $\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)}$ (for $j = 0$ and $j = 1$)

}

Prev

Correct: Simultaneous update	Incorrect:
$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$	$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$
$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$	$\theta_0 := \text{temp0}$
$\theta_0 := \text{temp0}$	$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$
$\theta_1 := \text{temp1}$	$\theta_1 := \text{temp1}$

96

Linear regression with one variable

GRADIENT DESCENT

97

Gradient descent algorithm

$$\text{repeat until convergence } \{$$
$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad \text{(simultaneously update } j = 0 \text{ and } j = 1)$$
$$\}$$

$\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$ Slope of the cost function

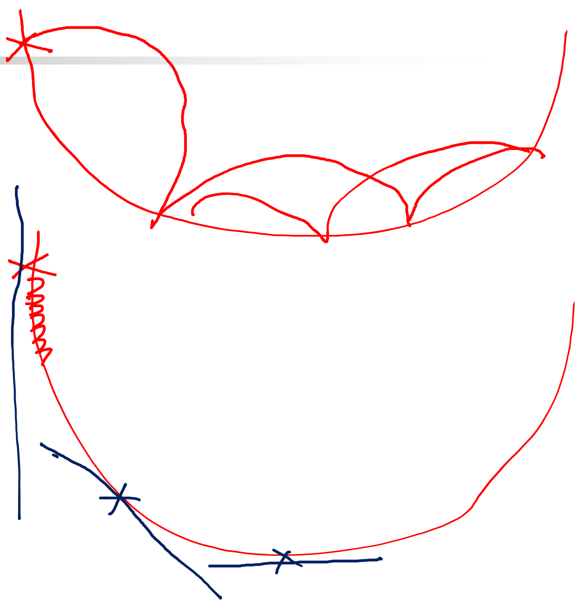
α Learning Rate

98

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$$

If α is too small, gradient descent can be slow.

If α is too large, gradient descent can overshoot the minimum. It may fail to converge, or even diverge.



99

What is the Intuition

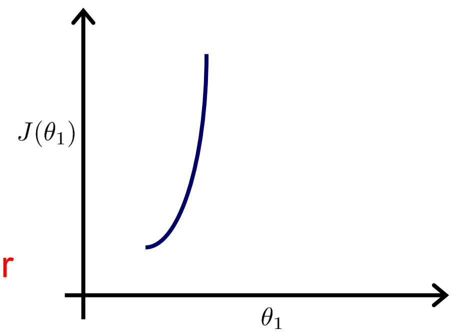
- Take larger steps when you start
- Take smaller steps when you are close to the solution
- This implies variable α

- Is it really necessary

Gradient descent can converge to a local minimum, even with fixed α.

$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

As we approach a local minimum, gradient descent will automatically take smaller steps. So, no need to decrease α over time.



Gradient descent algorithm

repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

(for $j = 1$ and $j = 0$)

}

Linear Regression Model

$$h_\theta(x) = \theta_0 + \theta_1 x$$

$$(\theta_0 + \theta_1 x^i - y^i)^2$$

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

$$x^{(i)}$$

$$\frac{\partial}{\partial \theta_1} J = \frac{\cancel{2}}{\cancel{2}m} \sum (h^i - y^i)\, x^{(i)}$$

$$\frac{\partial}{\partial \theta_0} J = \frac{1}{m} \sum (h^i - y^i) \cdot 1$$

$$h = \theta_0 \cdot 1 + \theta_1 x_1$$

$$\theta_0 x_0 + \theta_1 x_1$$

Gradient descent algorithm

repeat until convergence {

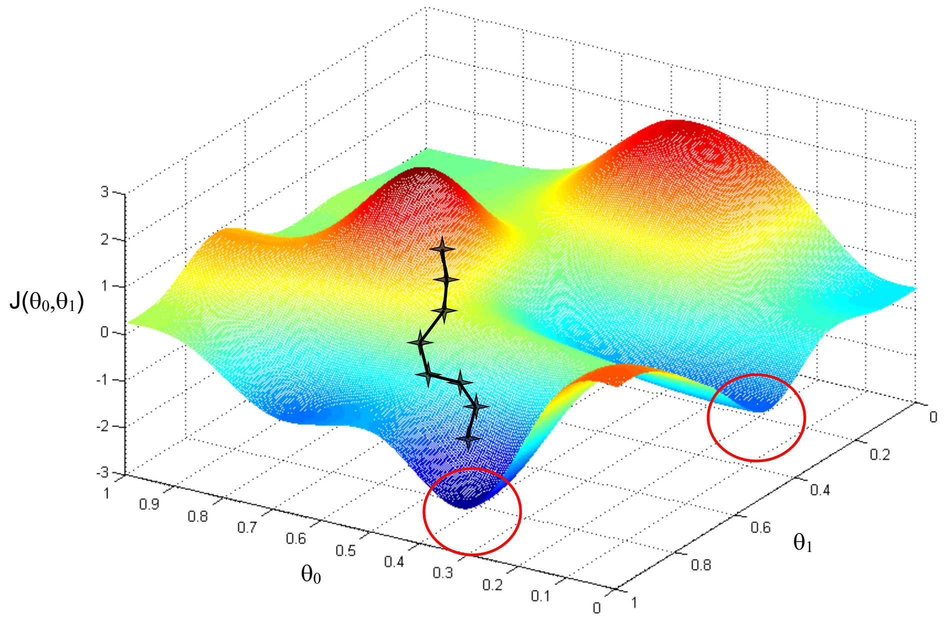
$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right) \cdot x_0^{(i)}$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right) \cdot x_1^{(i)}$$

}

update θ_0 and θ_1 simultaneously

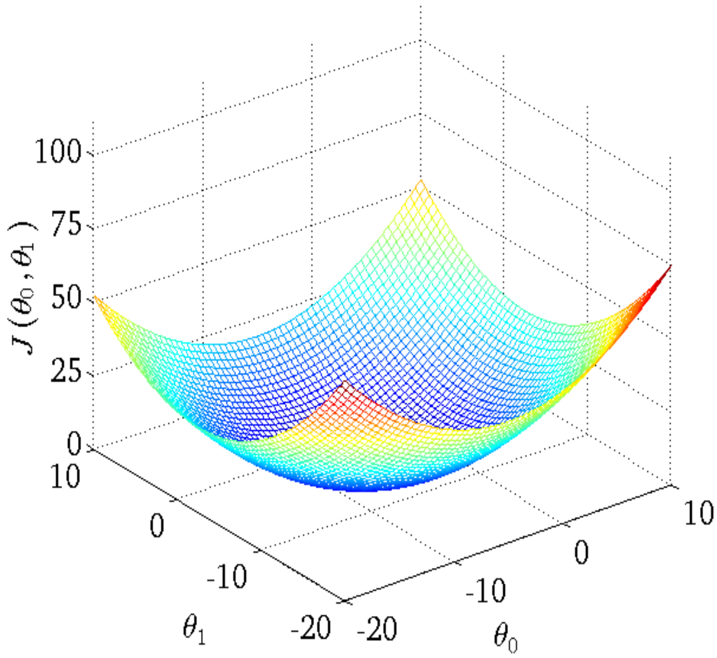
$$\boxed{\theta_j := \theta_j - \alpha \frac{1}{m} \sum_i (h^i - y^i) \cdot x_j^{(i)}}$$



Dr. Ali Hassan, DC&SE, College of E&ME, NUST

104

104



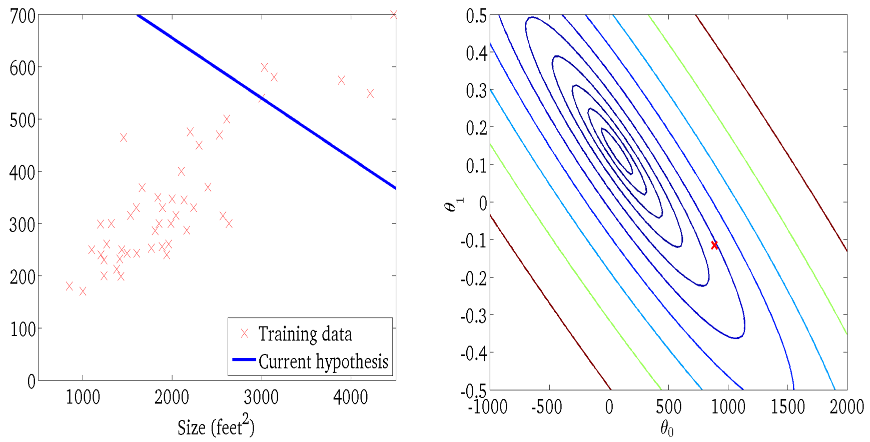
Dr. Ali Hassan, DC&SE, College of E&ME, NUST

105

105

$h_\theta(x)$
(for fixed θ_0, θ_1, this is a function of x)

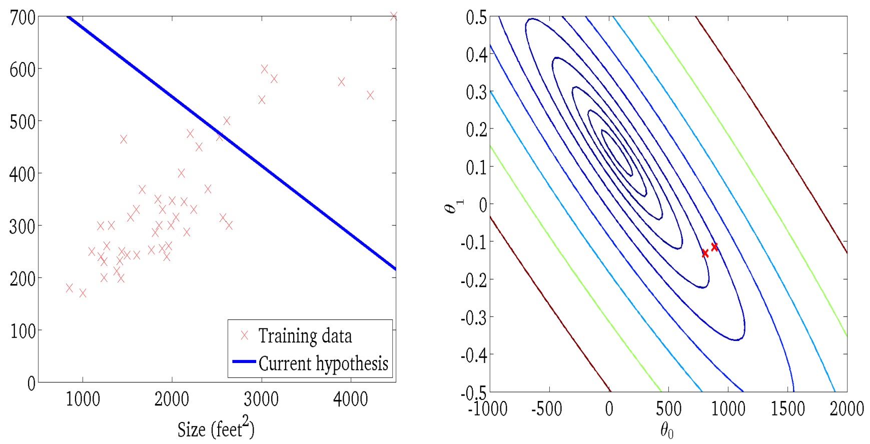
$J(\theta_0, \theta_1)$
(function of the parameters θ_0, θ_1)



106

$h_\theta(x)$
(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$
(function of the parameters θ_0, θ_1)



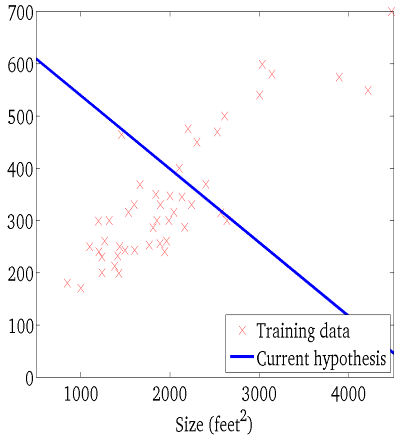
107

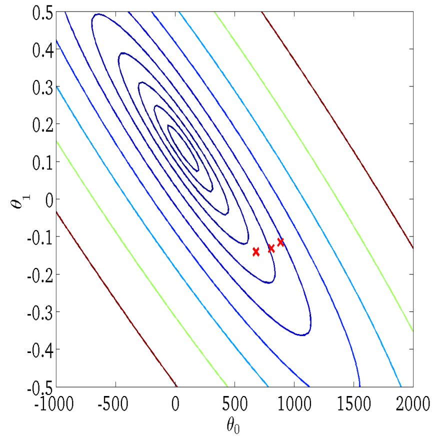
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)





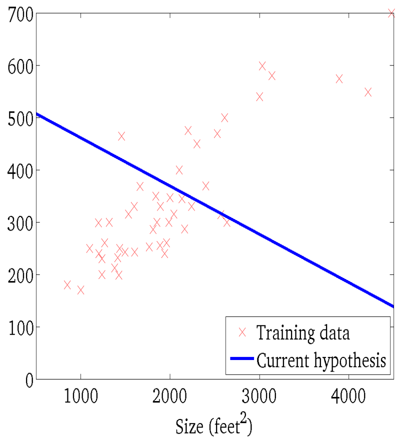
108

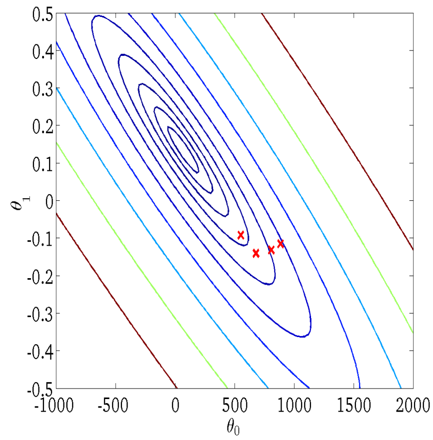
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)

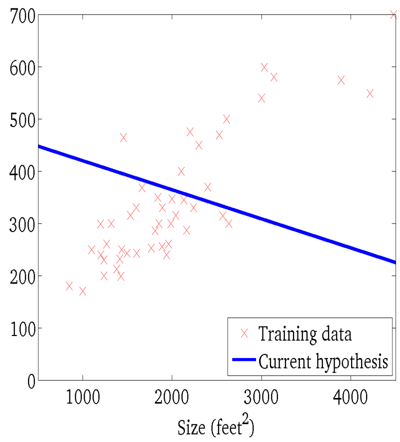


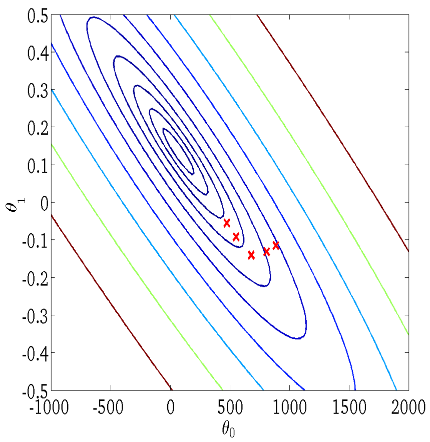


109

$h_\theta(x)$
(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$
(function of the parameters θ_0, θ_1)

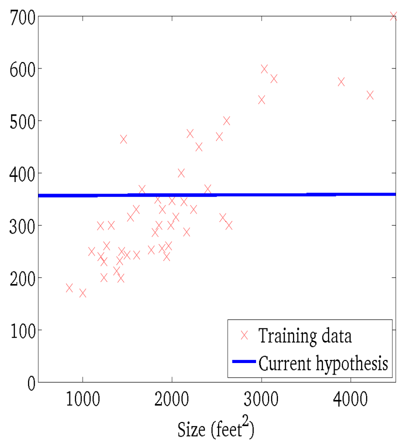


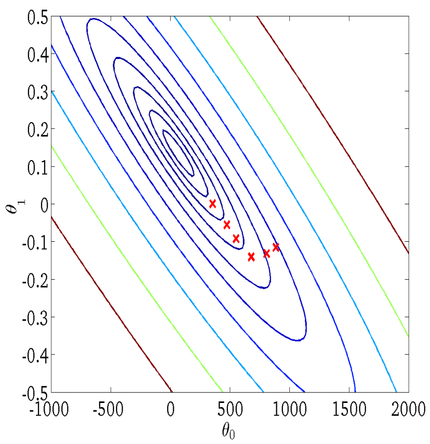


110

$h_\theta(x)$
(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$
(function of the parameters θ_0, θ_1)





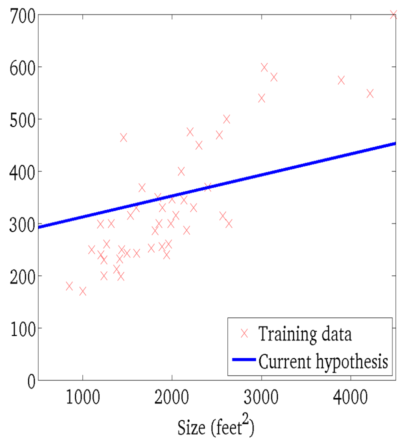
111

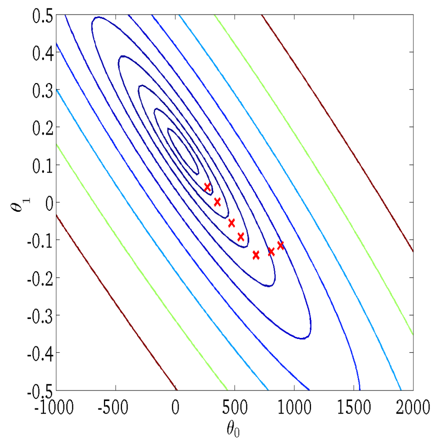
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)





Dr. Ali Hassan, DC&SE, College of E&ME, NUST 112

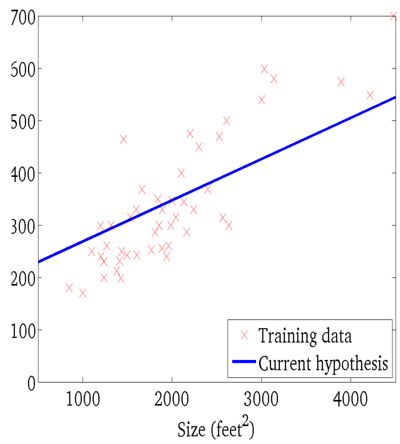
112

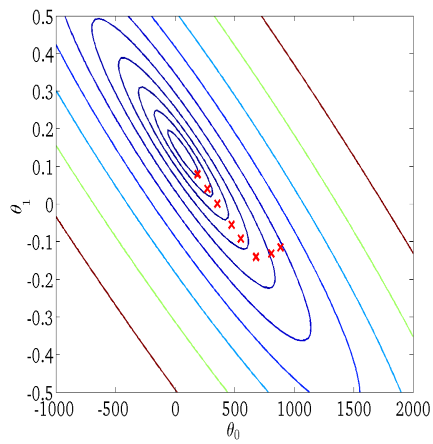
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



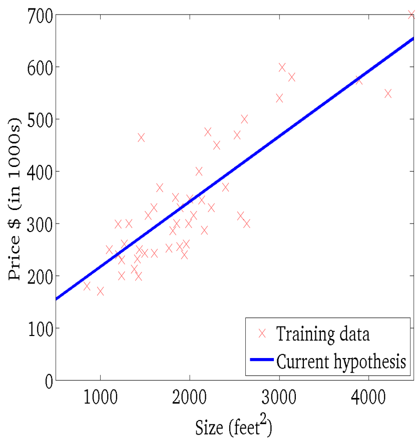


Dr. Ali Hassan, DC&SE, College of E&ME, NUST 113

113

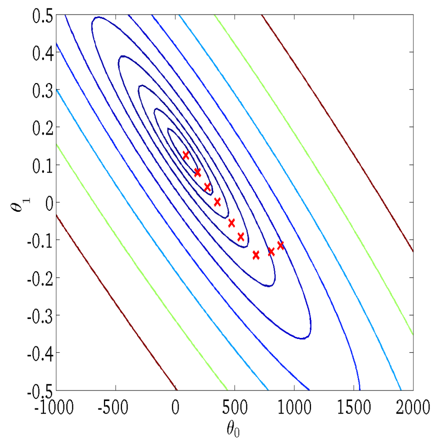
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



Linear Regression in Python

- m examples: $\{(x^i, y^i)\}_m$
- Each exp: $\boldsymbol{x} = < x_0, x_1 >$
- Hypothesis: $h_\theta(\boldsymbol{x}) = \theta_0 x_0 + \theta_1 x_1$
- Cost func: $J(\theta) = 1/2m \sum_{i=1}^{m} \left(h_\theta(x^i) - y^i\right)^2$
- $\frac{\partial J(\theta)}{\partial \theta_j} = 1/m \sum_{i=1}^{m} x_j^i \left(h_\theta(x^i) - y^i\right)$

Pseudo Code

- Initialize: $\theta \leftarrow zeros$
- Repeat:
  - $\theta = \theta - \frac{\alpha}{m} X^T (X\theta - y)$
- Output: θ

Linear Regression using GD

Initialization

```
self.cost_ = []
self.w_ = np.zeros((x.shape[1], 1))
```

Gradient Descent

```
for _ in range(self.n_iterations):
    y_pred = np.dot(x, self.w_)
    residuals = y_pred - y
    gradient_vector = np.dot(x.T, residuals)
    self.w_ -= (self.alpha / m) * gradient_vector
    cost = np.sum((residuals ** 2)) / (2 * m)
    self.cost_.append(cost)
return self
```

Linear Regression using GD

Prediction

```
def predict(self, x):
return np.dot(x, self.w_)
```


118

Linear Regression Using SciKit learn

Importing SciKitLearn Library

```
from sklearn.linear_model import
LinearRegression
```

Learning Model

```
# model initialization
regression_model = LinearRegression()
# fit the data(train the model)
regression_model.fit(x, y)
# predict
y_predicted = regression_model.predict(x)
```


119

Evaluating Model

```
from sklearn.metrics import mean_squared_error

# model evaluation
rmse = mean_squared_error(y, y_predicted)

# printing values
print('The coefficient is
{}'.format(regression_model.coef_))
print('The intercept is
{}'.format(regression_model.intercept_))
print('Root mean squared error of the model is
{}.'.format(rmse))
```


Uptill Now

- Gradient Descent—Feature Scaling
- Linear Regression with Multiple Variables
- Linear Regression -- Normal Equations

Next Topic

- Logistic Regression

Reading Work

- Chapter 7 Linear Regression from
  - *Machine Learning: A Probabilistic Perspective* by Kevin Murphy
  - Section 7-7.5.1 + 7.5.4
- A Good Visual Explanation of Linear regression
- https://mlu-explain.github.io/linear-regression/#:~:text=For%20example%3A%20predicting%20the%20price,x_2%20x2%3A%20age).

Next Lecture

- Linear Regression
  - Multiple variables
- Logistic Regression