KANAZAWA UNIVERSITY | Emerging Media Initiative

GSCI1801A

Information Science

Lecture 4: Artificial Intelligence and Applications in Daily Life

Asst. Prof. Chawanat NAKASAN | 2020-10-26

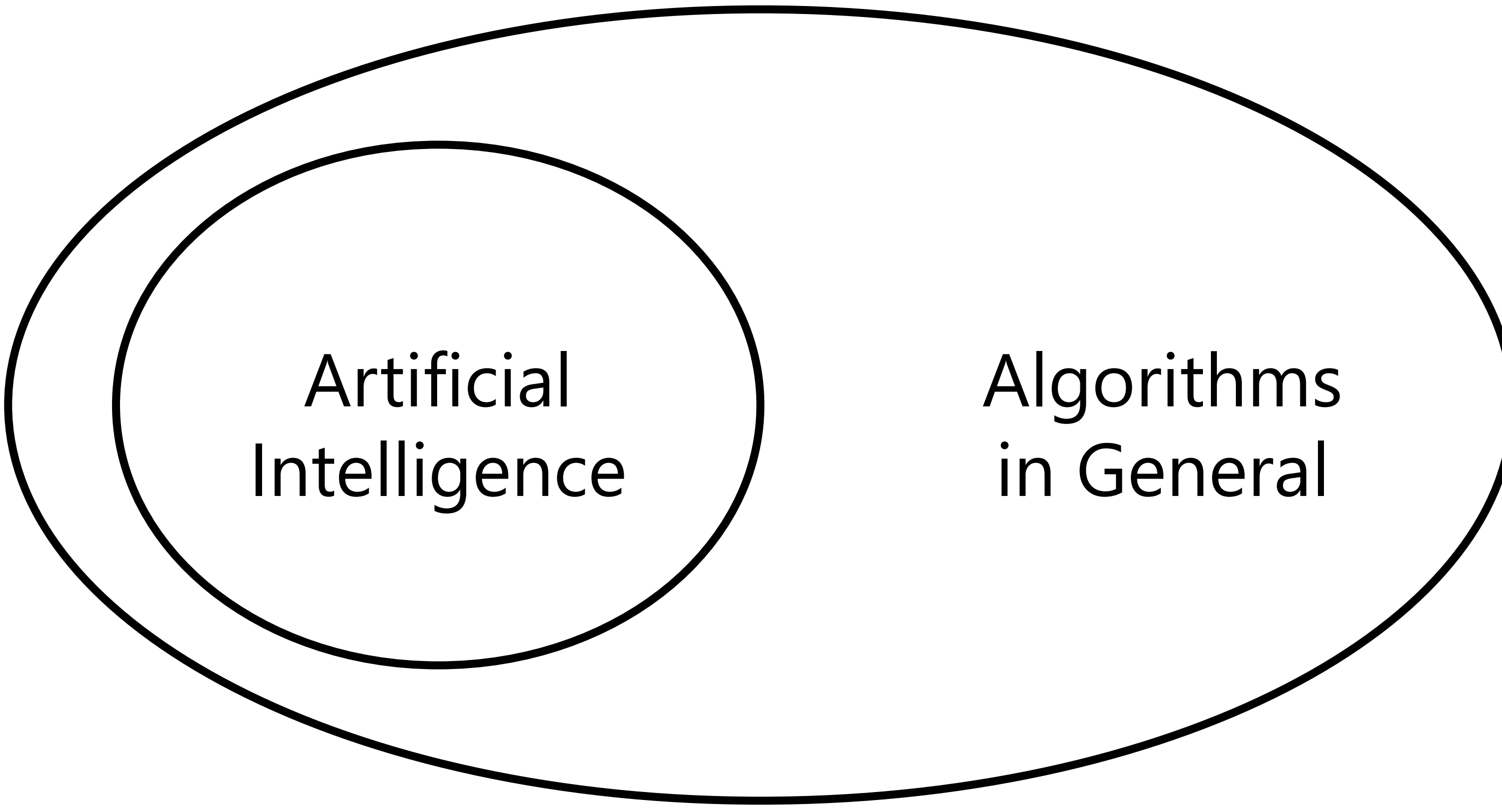
Agenda

- What is Artificial Intelligence?
- Optimization Problems
- Genetic Algorithm
- Machine Learning Concepts
- Linear Regression
- Neural Networks
- Deep Learning

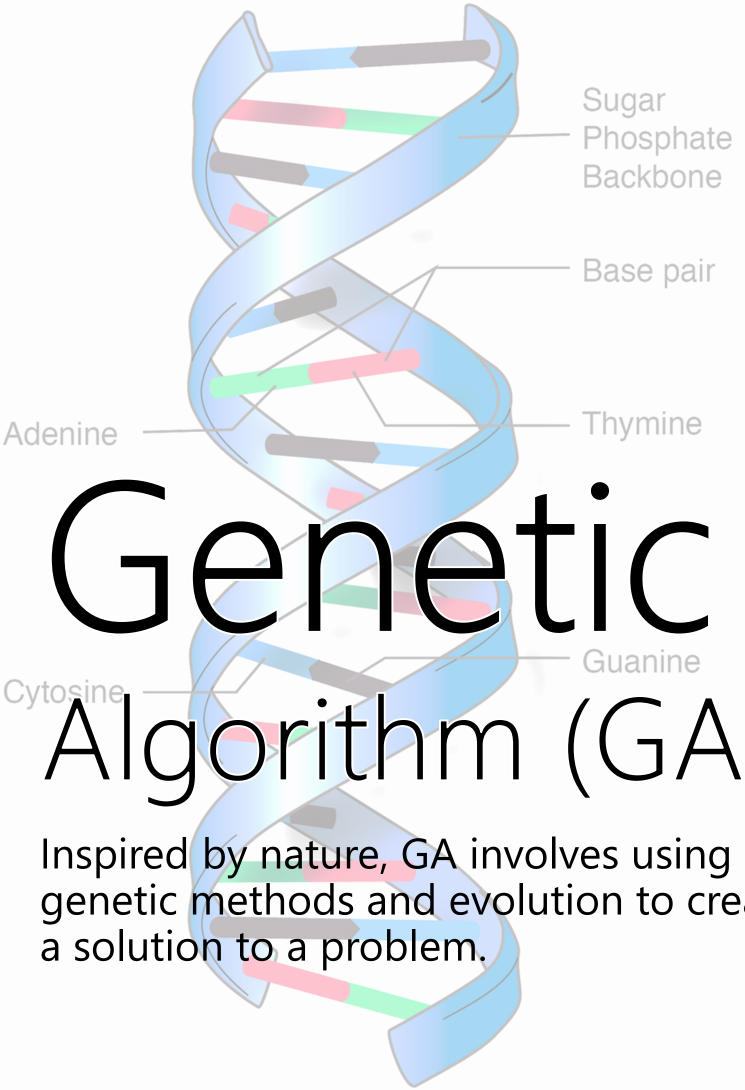
What is Artificial Intelligence?

- Up until now, all we have is a glorified calculator.
- It has more functions, it’s faster, but still a calculator.
- How do we make computers actually think and help us decide?

Context of AI and Algorithms

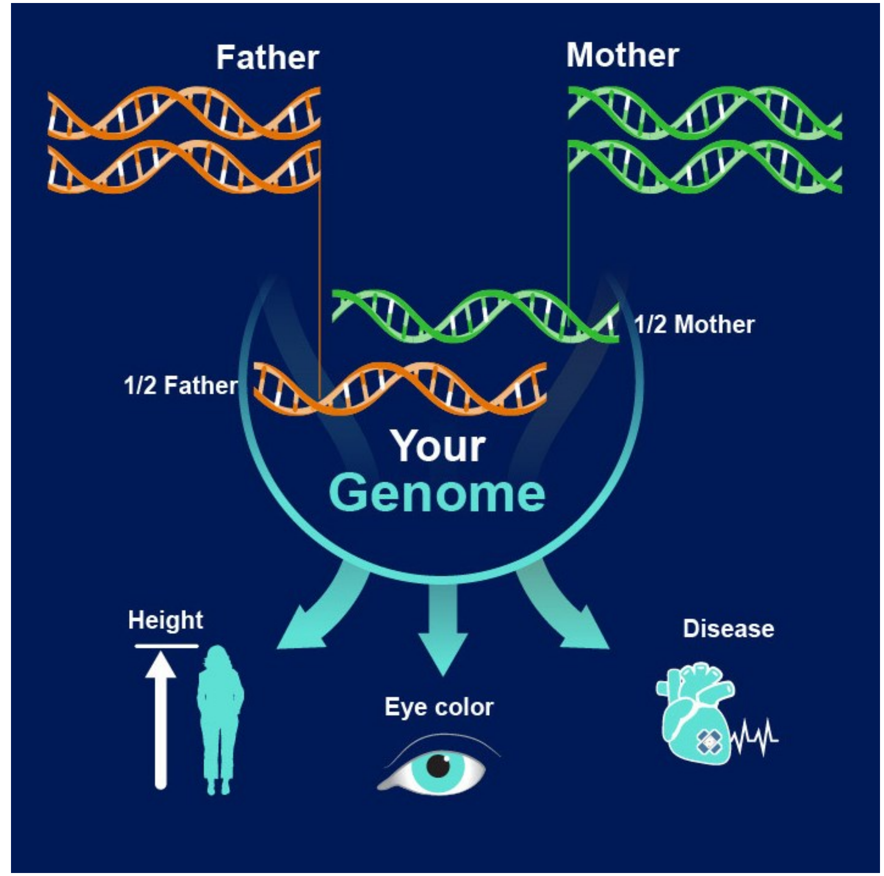


Some algorithms are not used to build an AI system.
All AI systems require some sort of algorithm.



Genetic Algorithm (GA)

Inspired by nature, GA involves using genetic methods and evolution to create a solution to a problem.



Graphics: National Human Genome Research Institute [Left/BG] [Right]

Genetic Algorithm (GA)

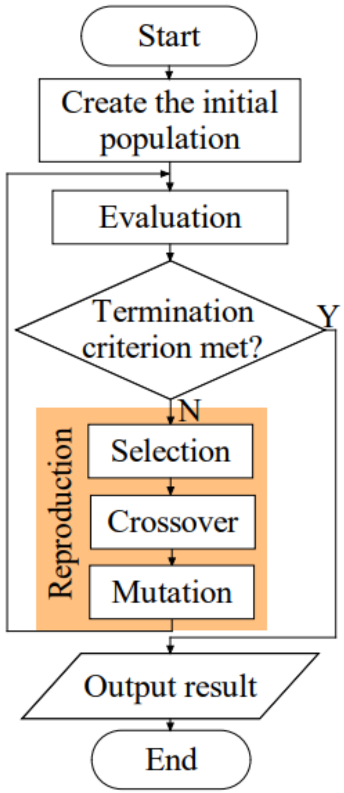


Figure 5-1: Flowchart of genetic algorithm

(Weng, 2021)

Initial Population: Random set of possible solutions (“candidates”).

Selection: Pick the best candidates.

Crossover: Use genetic concepts to “breed” pairs of selected candidates.

Mutation: Mutate the new candidates to introduce random-walking variation.

Back to the 'sack: Not everything is solved by greed!

(And not everything is solved *quickly enough* by brute force)

Weight	Value	V/W
15	12	0.800
90	15	0.167
*20	120	6.000
*120	720	6.000
25	30	1.200
*40	240	6.000
*20	120	6.000
50	20	0.400
30	20	0.667
20	18	0.900
35	55	1.571
#191	1147	6.005

Max weight = 200

*** Brute-forced solution: $1200, 120 kg**

$O(2^n)$. 12 objects isn't too tough for brute-forcing, but it won't work in the real world.

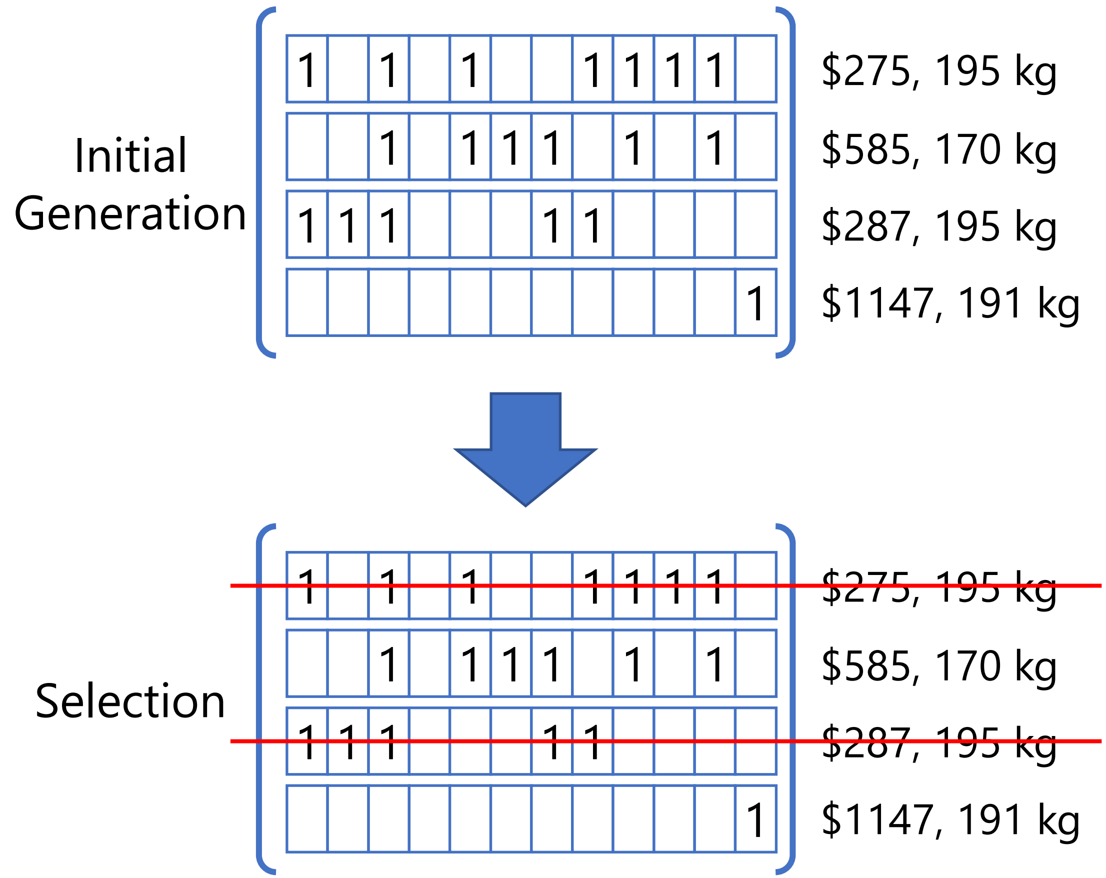
 # Greedy solution: $1147, 191 kg

$O(n \log n)$. However, blindingly choosing highest V/W produces suboptimal results.

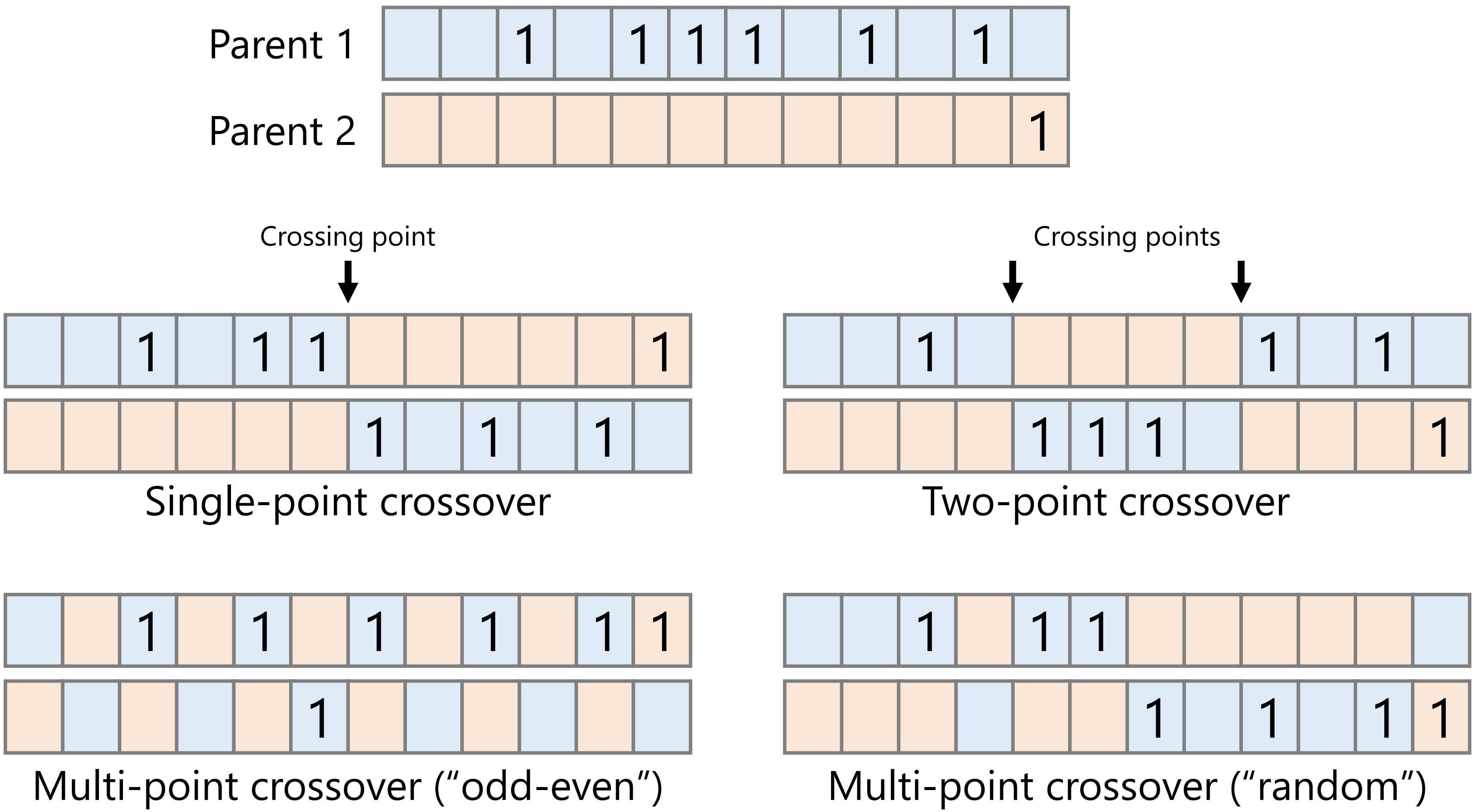
GA: Selection

Weight	Value	V/W
15	12	0.800
90	15	0.167
20	120	6.000
120	720	6.000
25	30	1.200
40	240	6.000
20	120	6.000
50	20	0.400
30	20	0.667
20	18	0.900
35	55	1.571
191	1147	6.005

Max weight = 200

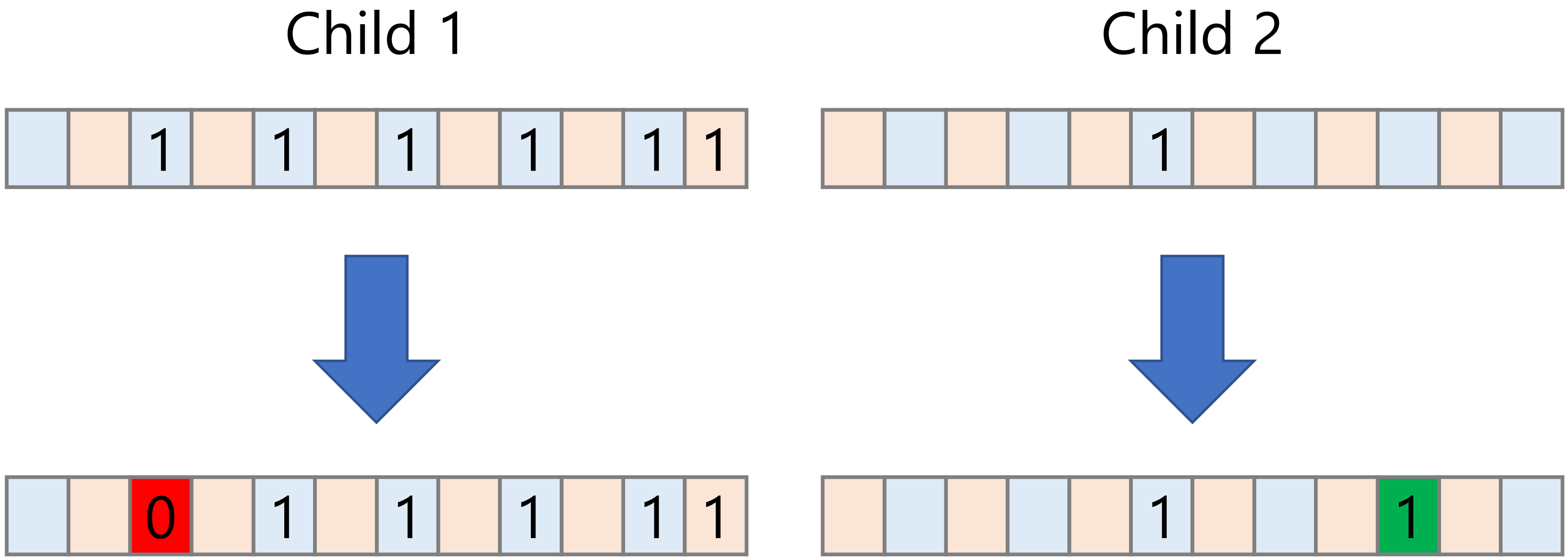


GA: Crossover



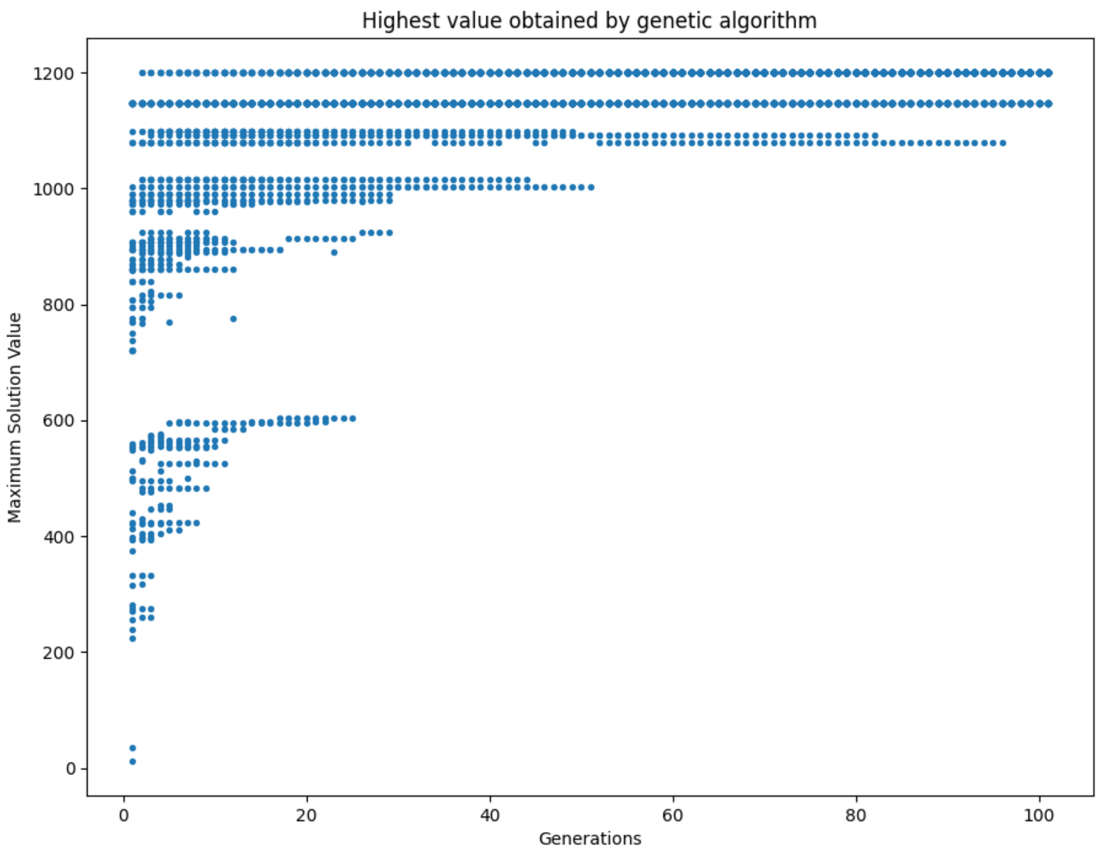
There are many kinds of crossovers invented, reviewed and described in many papers (Poon & Carter, 1995; Kaya, 2011)

GA: Mutation



Mutation can be done by changing random bit(s) or other methods.

Results of GA:
A system that improves itself



I wrote this in Python and charted its performance.

Each dot represents one (or more) repeated executions of the GA program.

You can see that there is an upward trend of the dots the more generations we run.

Machine Learning

How can computers learn?

- I still maintain the idea that computers understand only 0 and 1.

Supervised & Unsupervised Learning

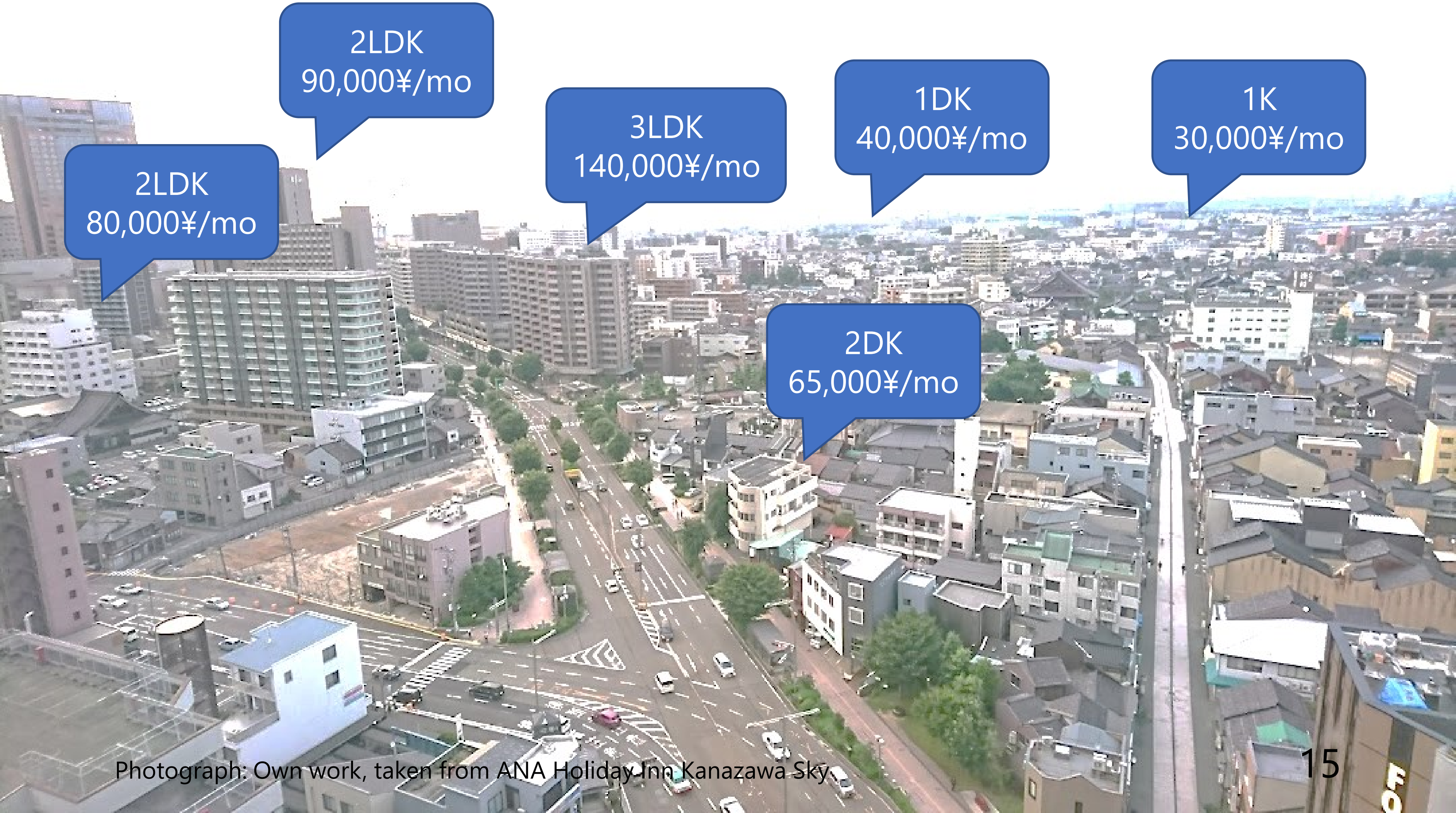
Supervised Learning

- Teaching with pre-trained data or samples.
- Examples:
  - Regression
  - Biometric Recognition

Unsupervised Learning

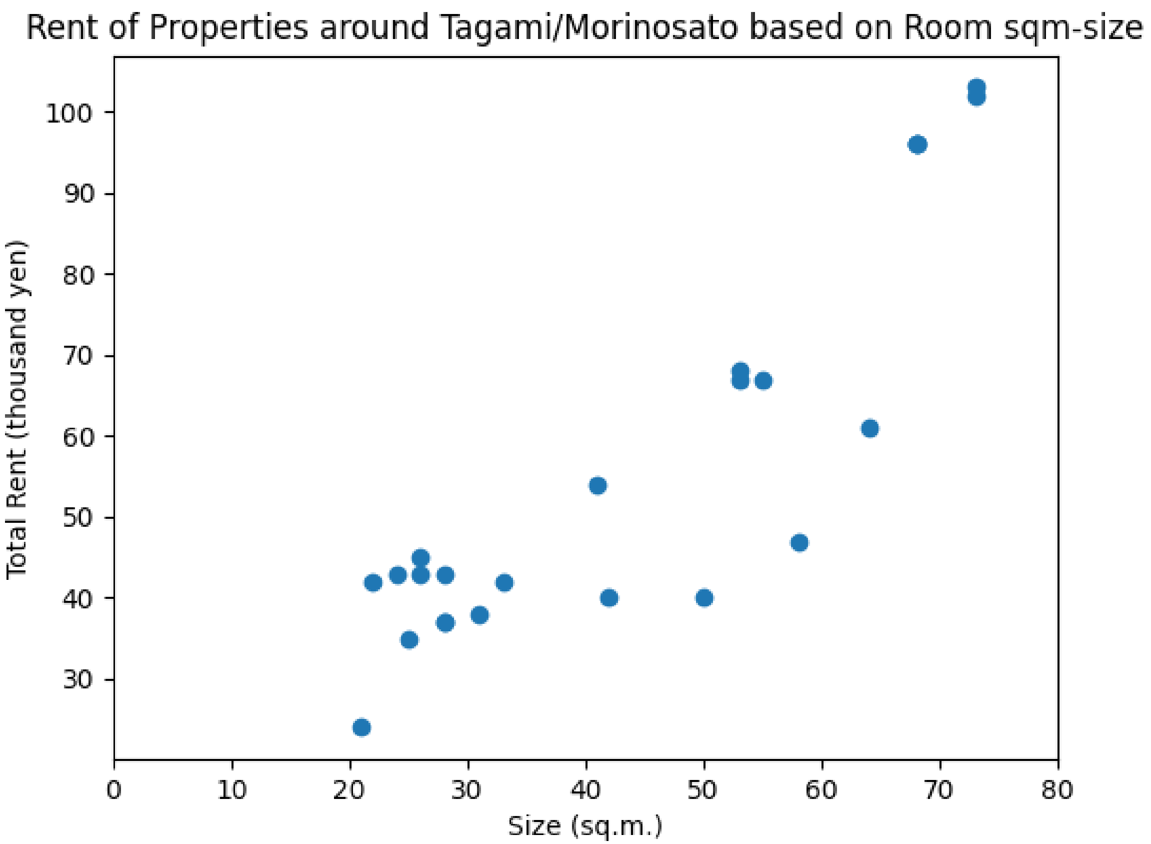
- Find new relationships without pre-trained data.
- Examples:
  - Clustering
  - Outlier Detection

Regression:
What influences housing prices?

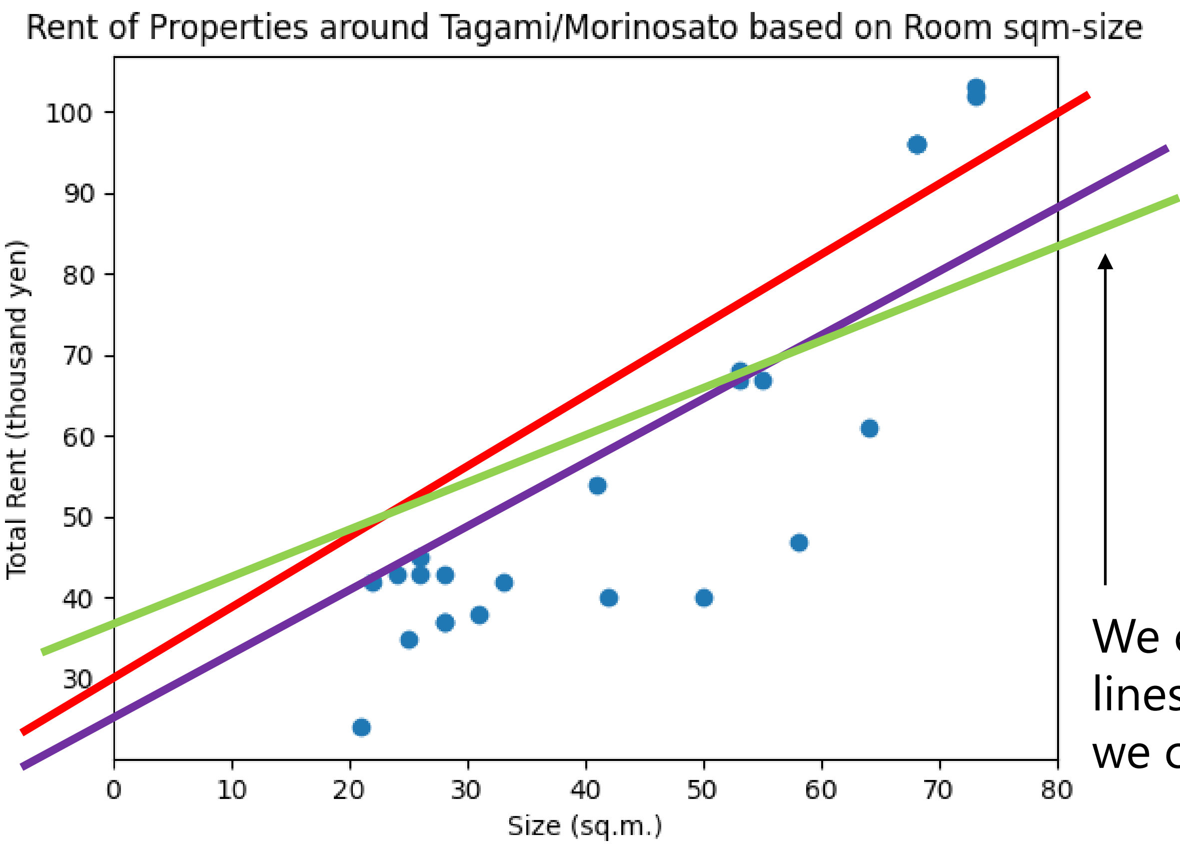


Photograph: Own work, taken from ANA Holiday Inn Kanazawa Sky

Converting Data Points to Insights: Establishing a Trend



There is a trend here, but how do we explain the relationship?



$y = mx + c$

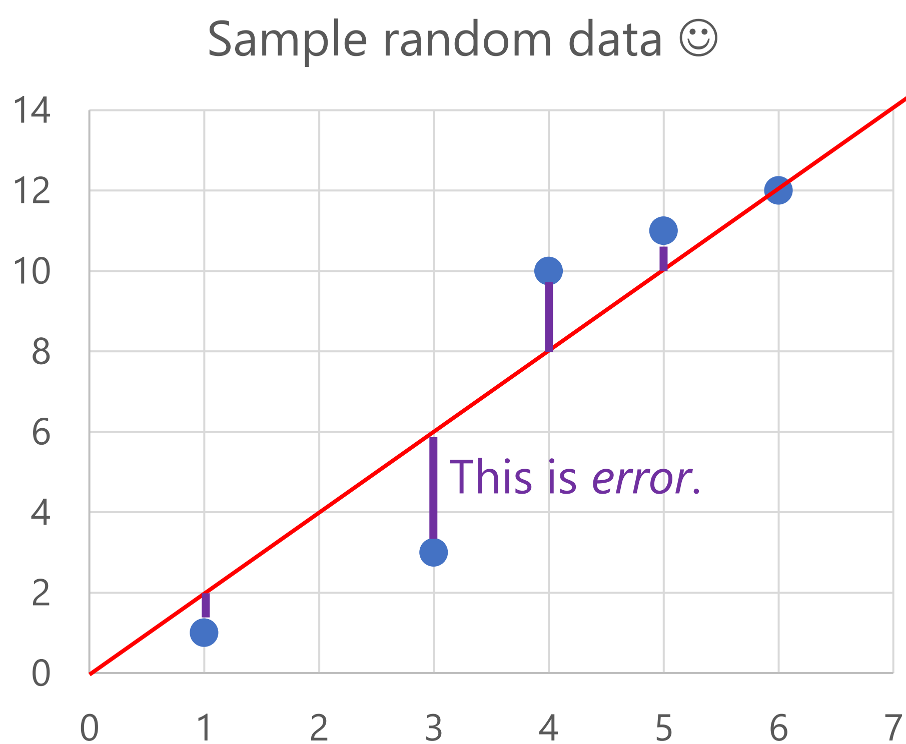
$m = ? \rightarrow \theta_1$

$c = ? \rightarrow \theta_0$

We choose to draw straight lines (linear relationship), so we call this **Linear Regression**.

Cost Function

Cost function is a key component to linear regression.
It shows how good our choice of m and c is.



$\hat{y}$ = 2x + 0

x	**y_i (actual)**	**$\hat{y}_i$ (predicted)**	**Error $(\hat{y}_i - y_i)$**	**Error^2 $(\hat{y}_i - y_i)^2$**
1	1	2	1	1
3	3	6	3	9
4	10	8	2	4
5	11	10	1	1
6	12	12	0	0
			SUM》	15

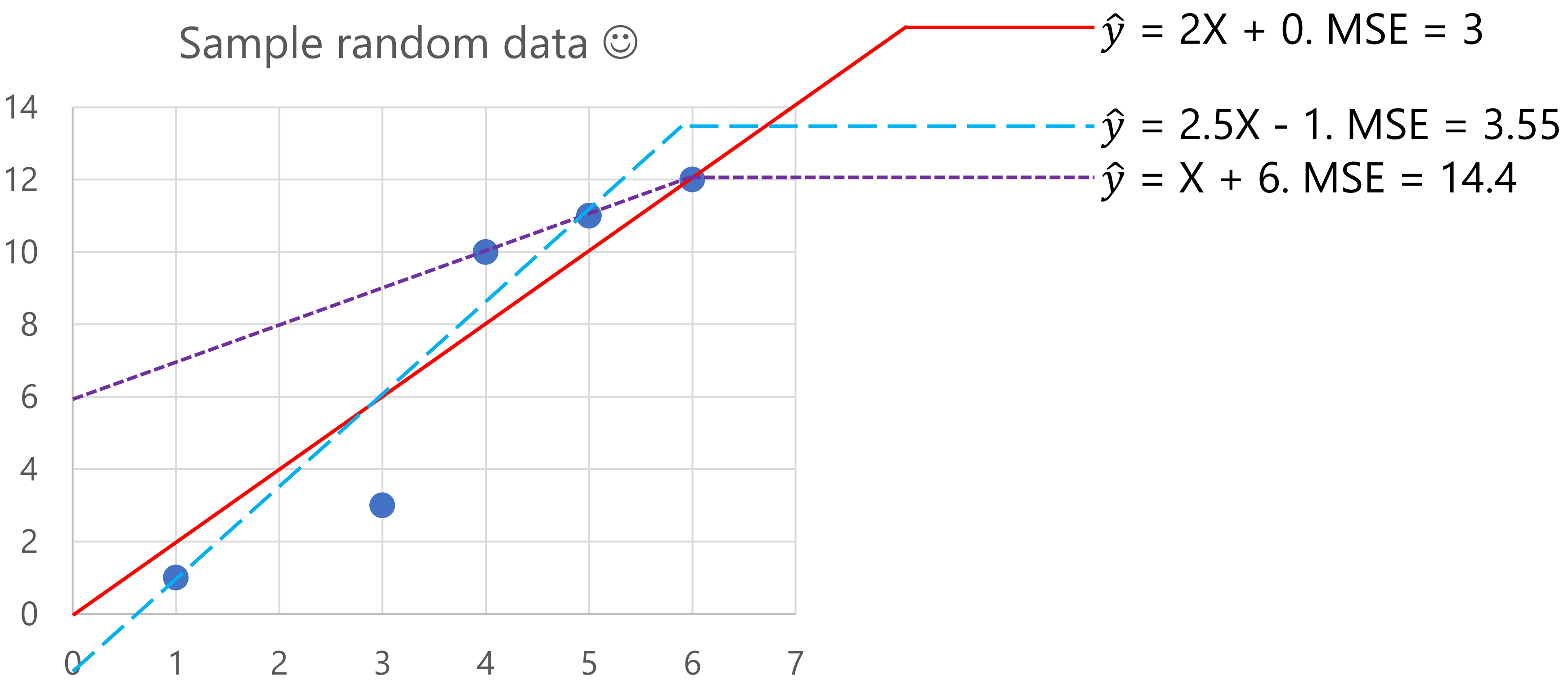
$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 = \frac{15}{5} = 3$$

(This is a separate example not related to the housing data.)

That's all the characters introduced.

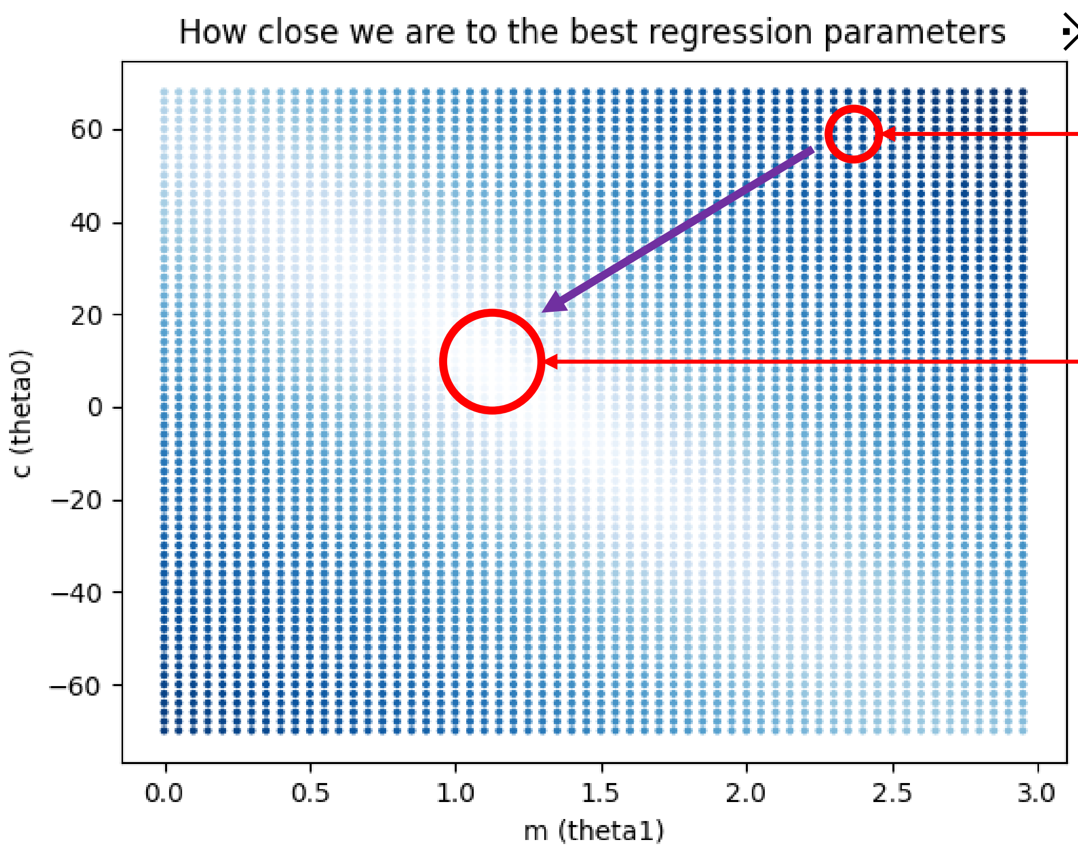
- Predictor function:
  - $\hat{Y} = m\mathrm{x} + c$
  - $h_\theta(x) = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n + \theta_0$
- Cost function:
  - $MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)^2$

How to adjust *m* and *c*?



Gradient Descent Concept

(No, we're not going to brute force again.)



※ apartments dataset

Suppose we're here

And we know this is where the best parameters are.

We should move as indicated by the purple arrow, i.e. "downwards".

Gradient Descent ~~Magic~~ Math

(This is the hardest math today.)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

For the purpose of this class, we'll formally define the cost function as ➔

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{n}(h_\theta(x_i) - y_i)^2$$

In case of only one data point:

$$J(\theta) = \frac{1}{2}(h_\theta(x) - y)^2$$

Gradient Descent Math
(Continued)

Derivative of the inside

Continue derivation

$$J(\theta) = \frac{1}{2}(h_\theta(x) - y)^2$$

Chain Rule:

$$\frac{\partial}{\partial \theta_j} J(\theta) = \underbrace{2 \times \frac{1}{2}(h_\theta(x) - y)}_{} \times \frac{\partial}{\partial \theta_j}(h_\theta(x) - y)$$

$$= (h_\theta(x) - y) \times \frac{\partial}{\partial \theta_j}\left(\theta_j x_j + \cancelto{0}{\sum_{i=0..n}^{not\ j} \theta_i x_i} - \cancelto{0}{y}\right)$$

$$= (h_\theta(x) - y) x_j$$

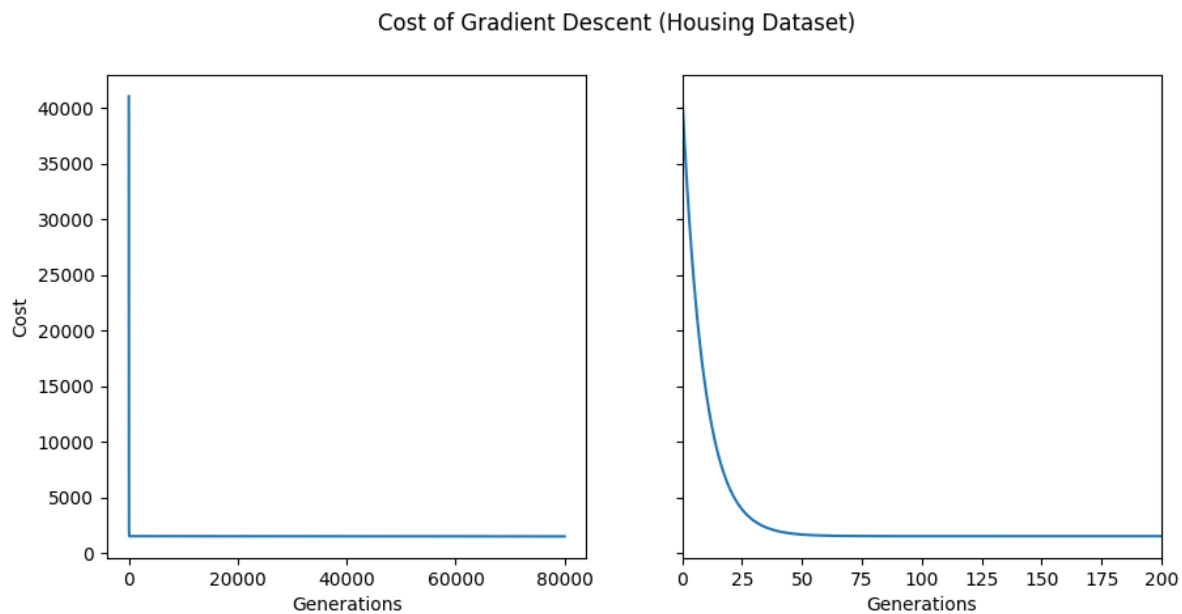
Gradient Descent Math

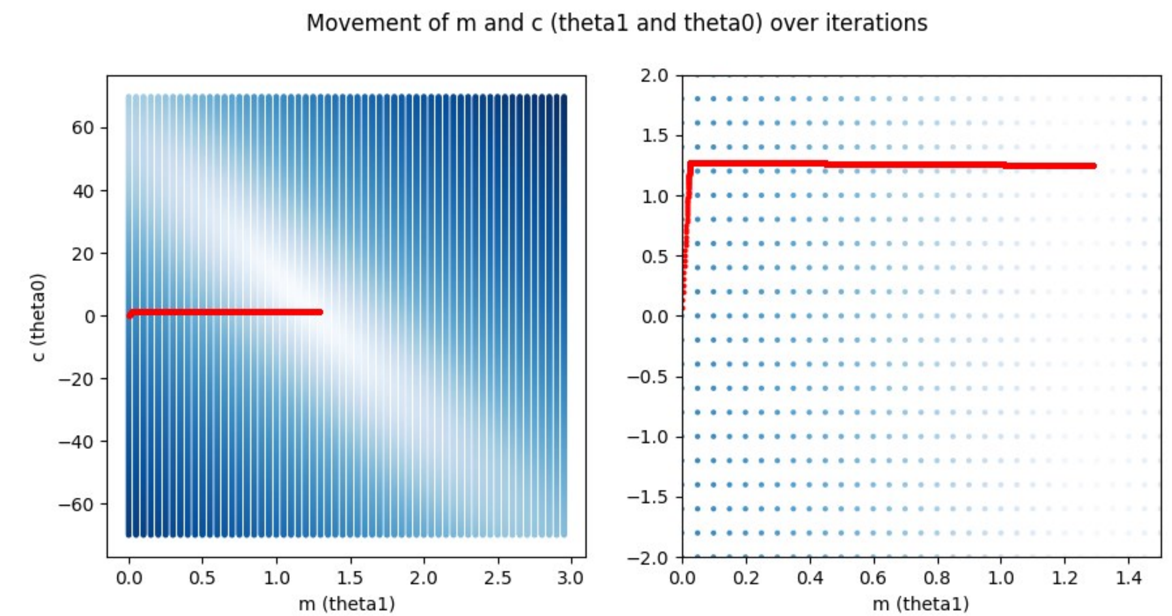
(Final math slide, I promise)

- So our slope is: $\frac{\partial}{\partial \theta_j} J(\theta) = (h_\theta(x) - y)x_j$

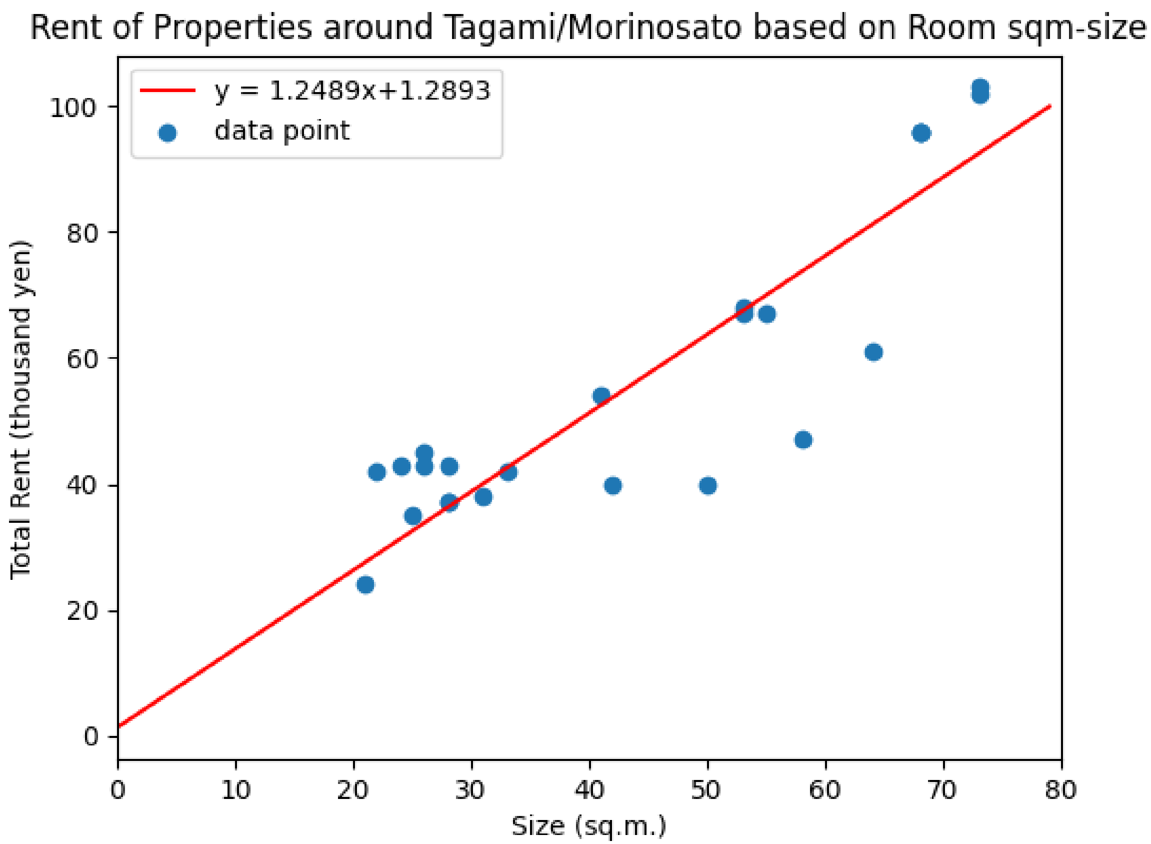
- This means while we're standing at a specific point x, we should adjust our θ_j such that …

$$\theta_j \leftarrow \theta_j + \alpha\big(y_i - h_\theta(x_i)\big){x_i}_j$$


Cost of Gradient Descent (Housing Dataset)
Cost
Generations
Generations
40000
35000
30000
25000
20000
15000
10000
5000
0
0
20000
40000
60000
80000
0
25
50
75
100
125
150
175
200


Movement of m and c (theta1 and theta0) over iterations
c (theta0)
m (theta1)
m (theta1)
60
40
20
0
−20
−40
−60
0.0
0.5
1.0
1.5
2.0
2.5
3.0
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4

Finally!



Rent of Properties around Tagami/Morinosato based on Room sqm-size
y = 1.2489x+1.2893
data point
Total Rent (thousand yen)
Size (sq.m.)
0
20
40
60
80
100
0
10
20
30
40
50
60
70
80

Other forms of regression exist!

- Logistic regression
  - Answer yes/no using value inputs
- Polynomial regression
  - Predict more complex systems

- Others exist, but linear + logistic + polynomial is usually good enough for general production use.

Uses of Regression

- Explain factors behind phenomena
  - Why housing is expensive?
  - What makes this animal big?
  - What contributes to the success of this product?
- Predict future values
  - Note: Predicting “out of range” isn’t a very good idea.

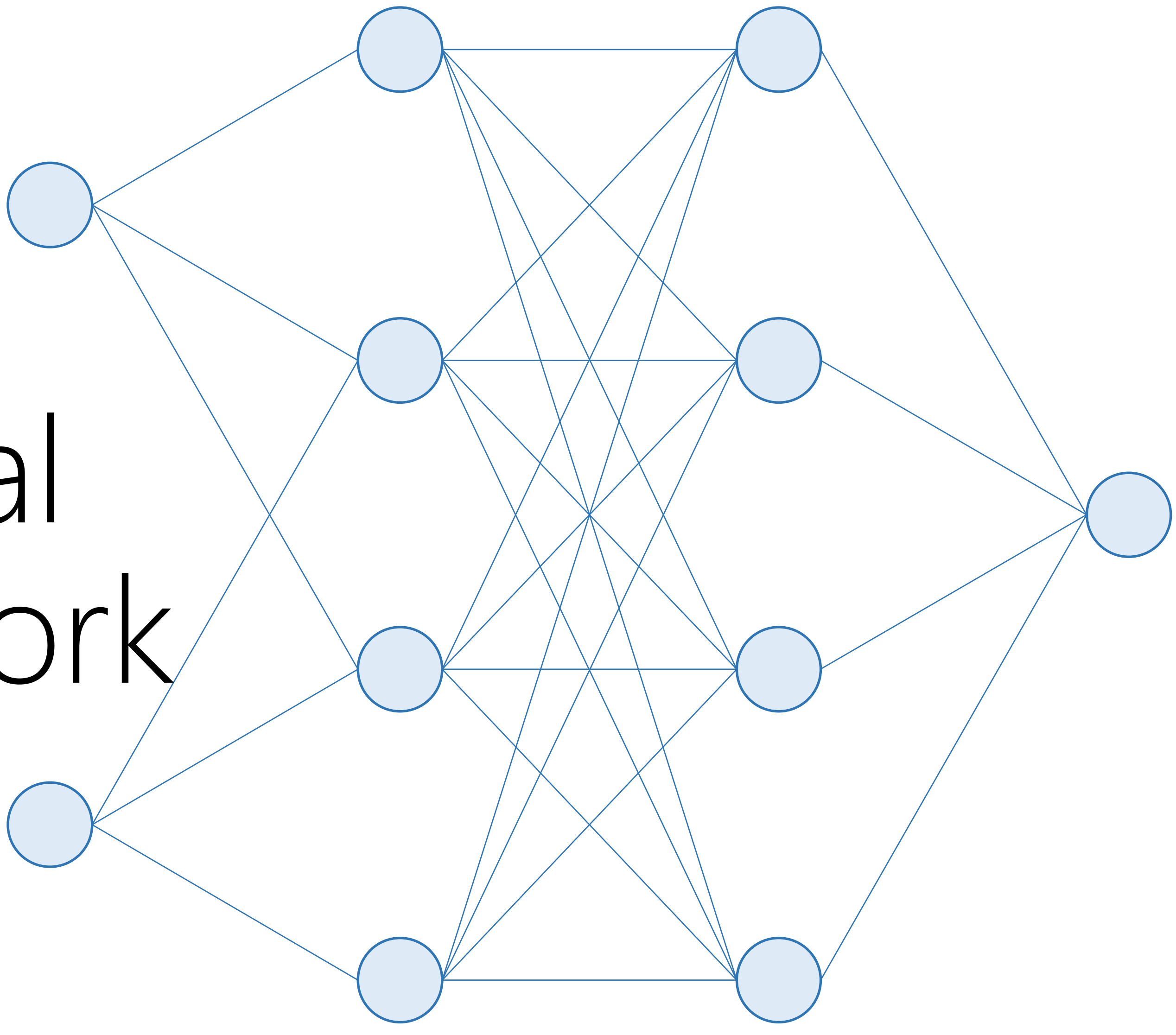
But in real life, we can do more.

- Size (m^2)
- Type of Internet
- Rooms+LDK
- City or LP Gas
- Building Age
- Security
- Floor of Room
- Distance from Work/School
- Transportation
- Qualitative Factors

They all affect → Price!

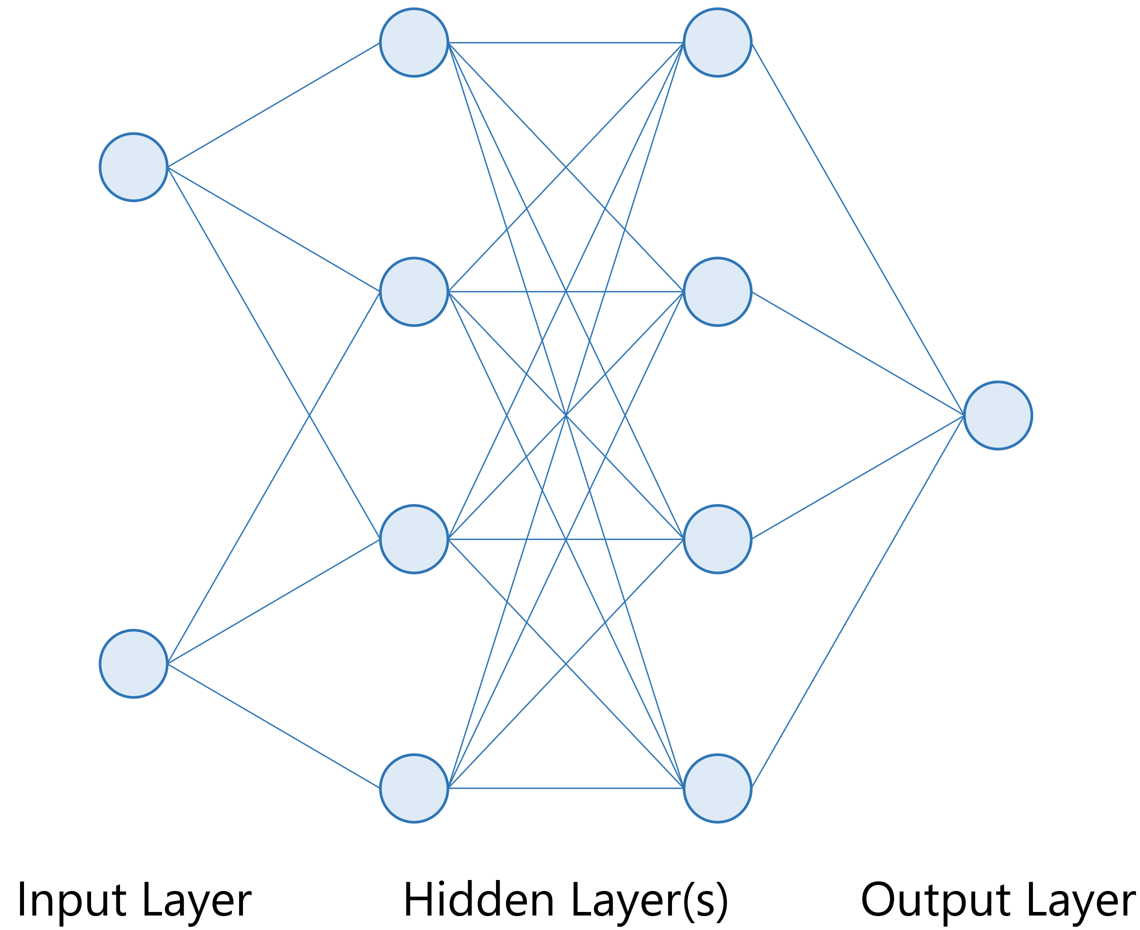
Multivariate regression is also possible, but it's just additional complexity so let's just go for now.

Neural Network



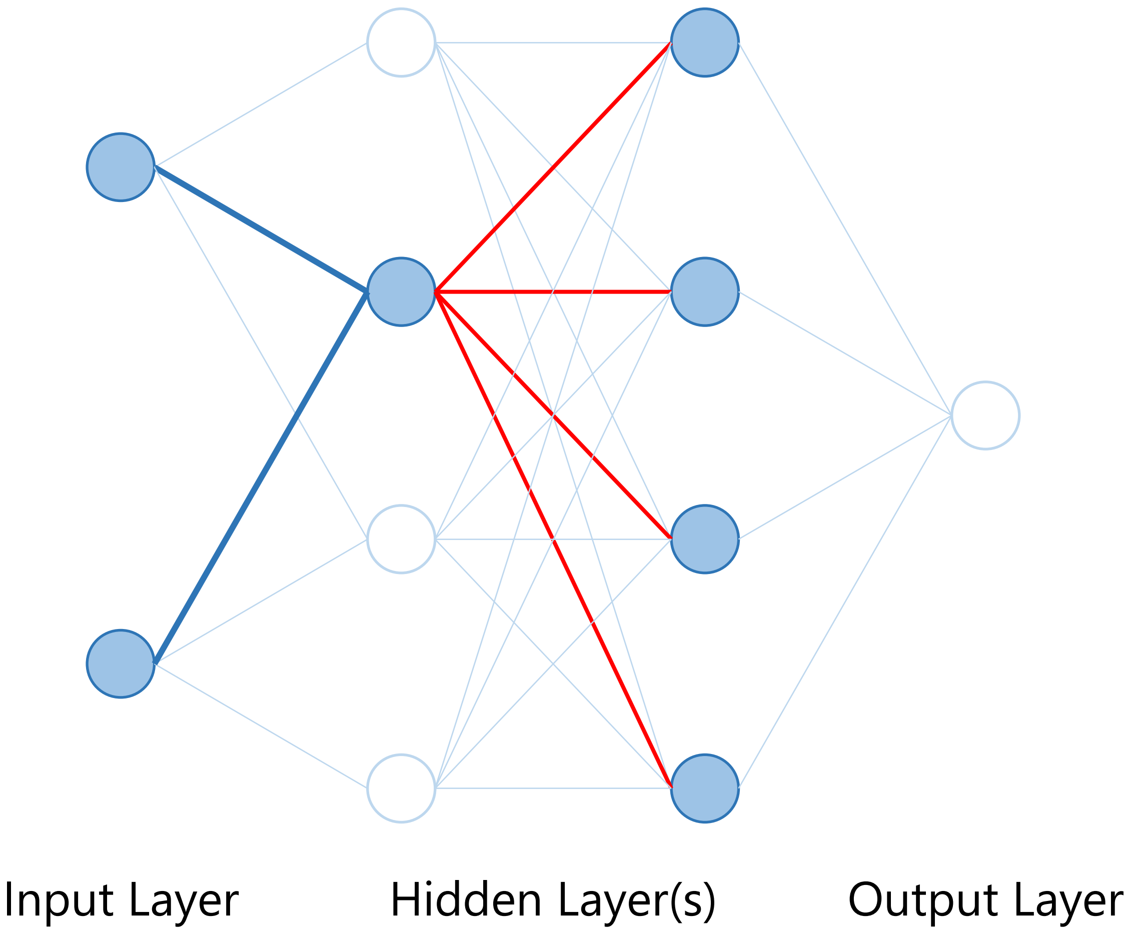
What is Neural Network (NN)?

- Simulation and approximation of neural systems.
- Since we're making an artificial one, sometimes we say, "artificial neural network" (ANN) to be specific.

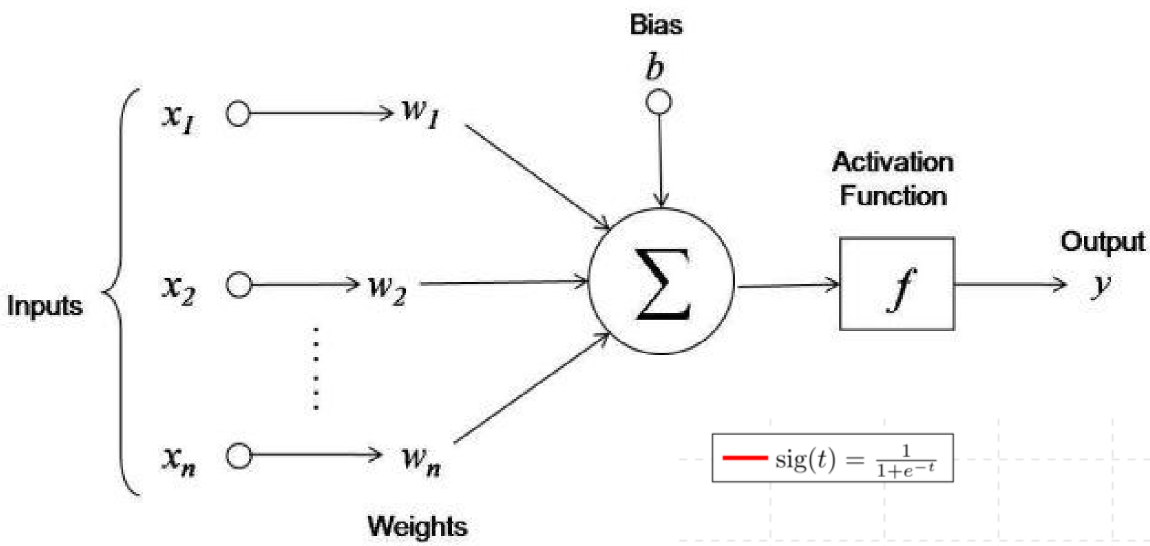


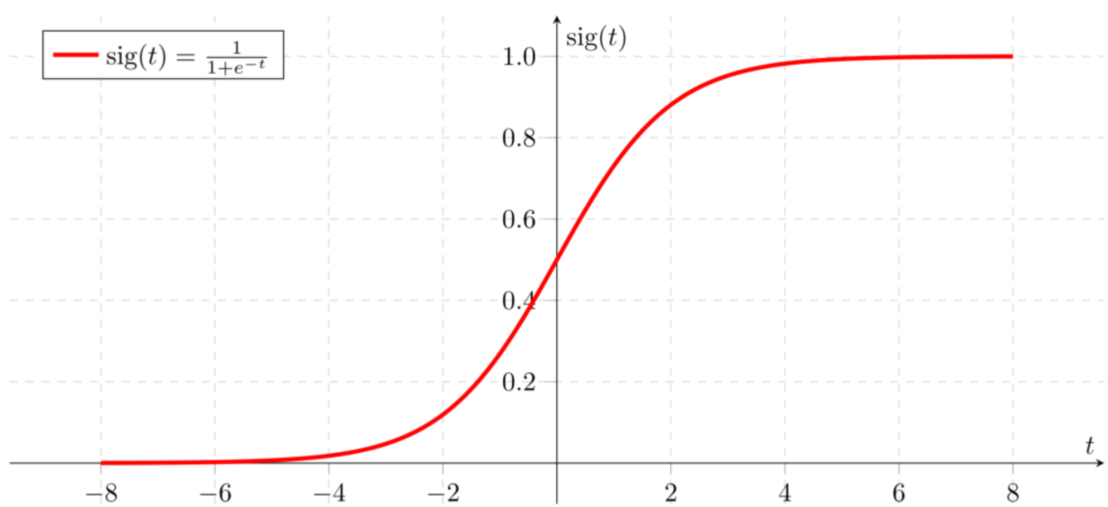
What makes NNs work?

- Each neuron *activates* (fires) based on inputs leading to it.
- This is based on ~~*magic*~~ *math*.



Specifically, this kind of math.



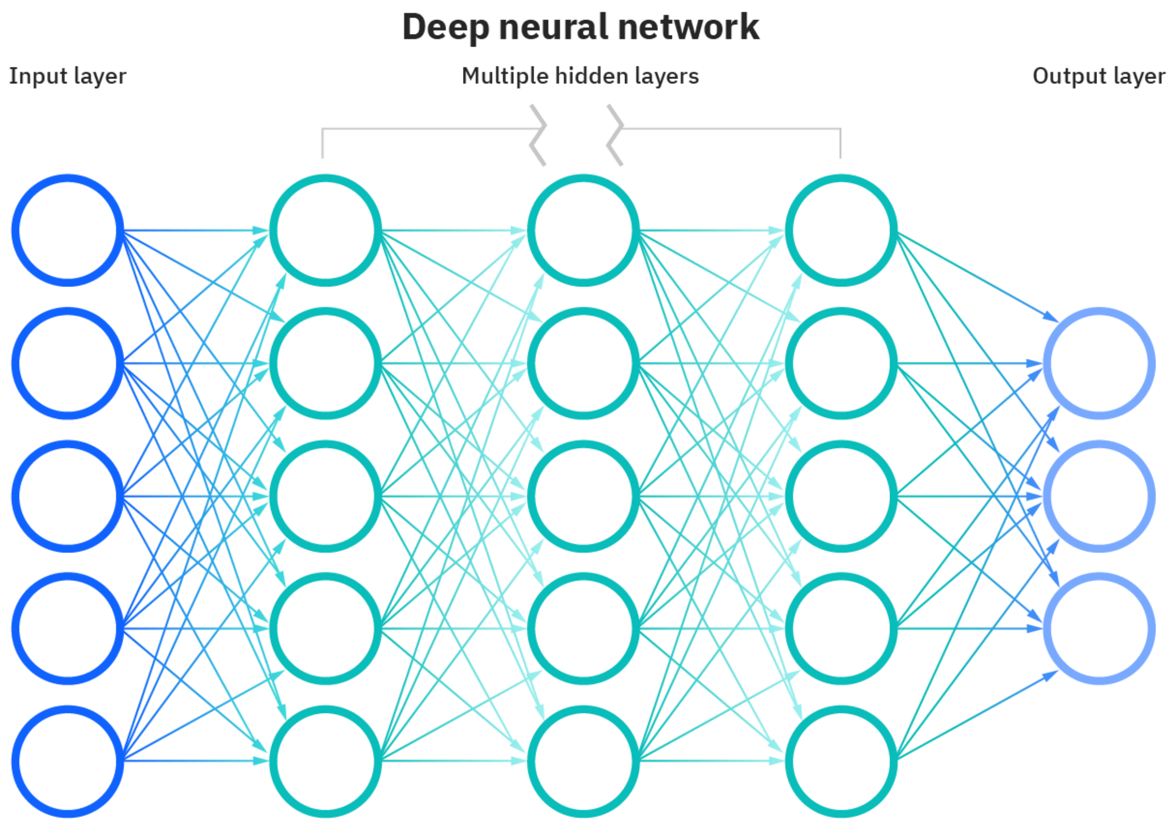


Graphics: https://towardsdatascience.com/first-neural-network-for-beginners-explained-with-code-4cfd37e06eaf

Simulation & Playground

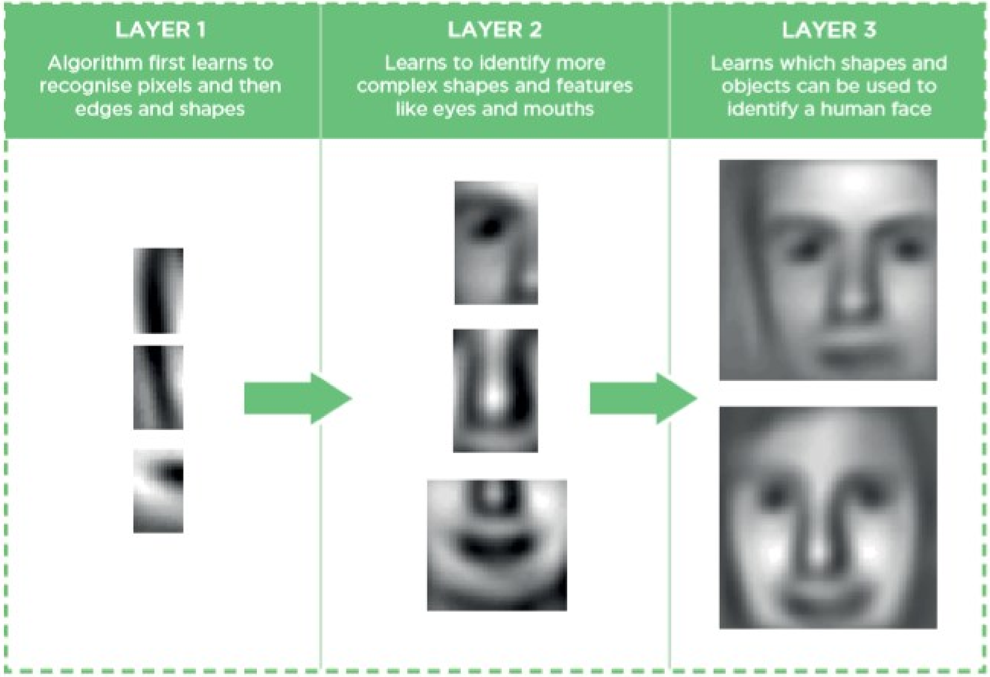
- https://playground.tensorflow.org/

Wait, what about *deep learning*?



Graphics: https://www.ibm.com/cloud/learn/neural-networks

Why add more layers?



Graphics: https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063

References

- Part of the content on this lecture (linear regression) is based on Prof. Andrew Ng's lecture notes.
  - https://see.stanford.edu/materials/aimlcs229/cs229-notes1.pdf