[Reminder again] Assessments:

- Tutorial participation: 10%
- Assignment (1, group): 20%(done)
- Essay/Report writing (1, individual): 30%
- Group Project: 40%

MC Quiz on lecture materials among 50%

Please submit your topic of group project by

26 Mar (Wed) 23:59pm!

Recall: Topic of the project:

- Each group can pick a topic of members' own choices.
- The topic should be related
  - FinTech (i.e., make use of FinTech technologies)
  - To solve a problem of the limitation of certain traditional finance services

Essay/Report writing (individual): 30%

This is an essay-type assignment, EACH student should submit an INDIVIDUAL article to address how ChatGPT/DeepSeek or other related LLMs can help FinTech or the financial industry? [Due date: 26 Mar Wed, 11:59pm]

What you need to hand in:
EACH student is required to submit a max 5-page essay (excluding the cove page and the references) by the deadline to elaborate your points.

Rubrics for scoring:
You can refer to this grade descriptors for reference:
https://ar.cetl.hku.hk/pdf/grade_descriptors/Grade%20Descriptors%20for%20Essays.pdf

CCST9080 Group Project
Peer Review Form

The peer review is to evaluate the contribution of each member throughout the project. Instructors and teaching assistants will look at the peer evaluation and then make the judgement accordingly.

Please list all the members of your group (including yourself) in the table below and indicate the percentage contribution of each member.

Do NOT share or discuss your evaluations with other group members.

The completed form should be submitted on Moodle individually by April 29, 2022.

****

Project title: XYZ

Your name: Member 1

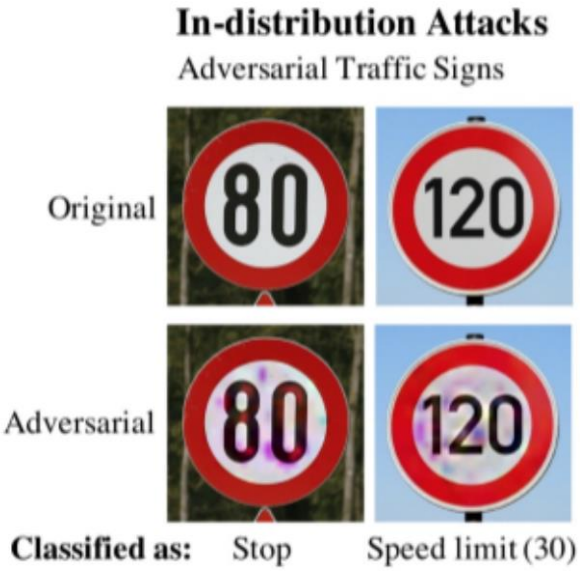
Group members *(Surname, First Name)*	**Contribution to Project** *(out of 100%)*	**Comments**
Member 1	**50%**	**My main responsibility is to research the issues of the problems with strong evidence**
Member 2	**50%**	**Member 2 is responsibility is to provide possible solutions to solve the problems**
Total	**100%**	

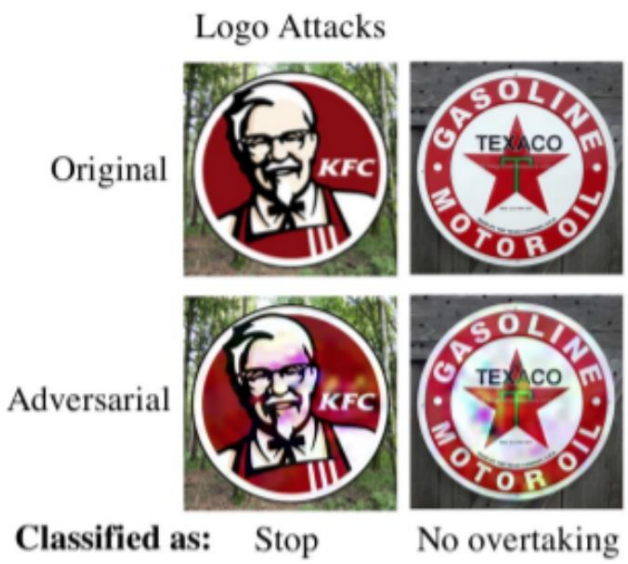
Recall the technologies for FinTech:

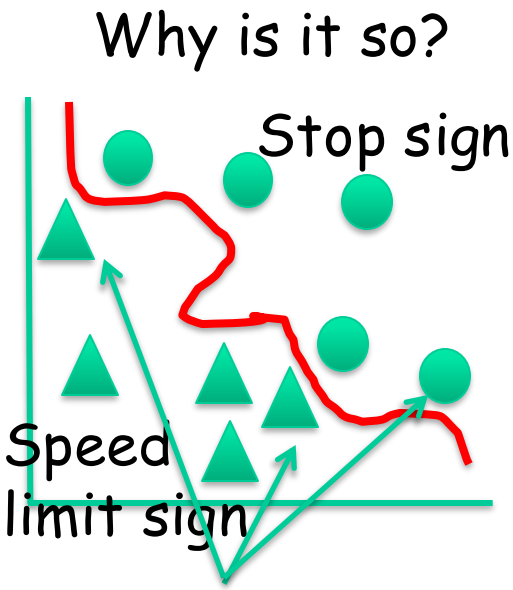
- Blockchain
- E-payment technology
- RegTech
- Cyber security and privacy
- Big data analytics
- AI/ML (Artificial Intelligence/Machine Learning)

  - A big R&D area together with lots of applications
  - Many tools & models
    e.g. decision tree, SVM (support vector machine), random forest, Bayesian network, ….

Recall these examples: Researchers from US demonstrated that only a little bit modification to a road sign (e.g. Speed limit of 80m/hr sign => STOP sign)







https://arxiv.org/pdf/1802.06430.pdf

A small modification in the image may make it cross the classification line!

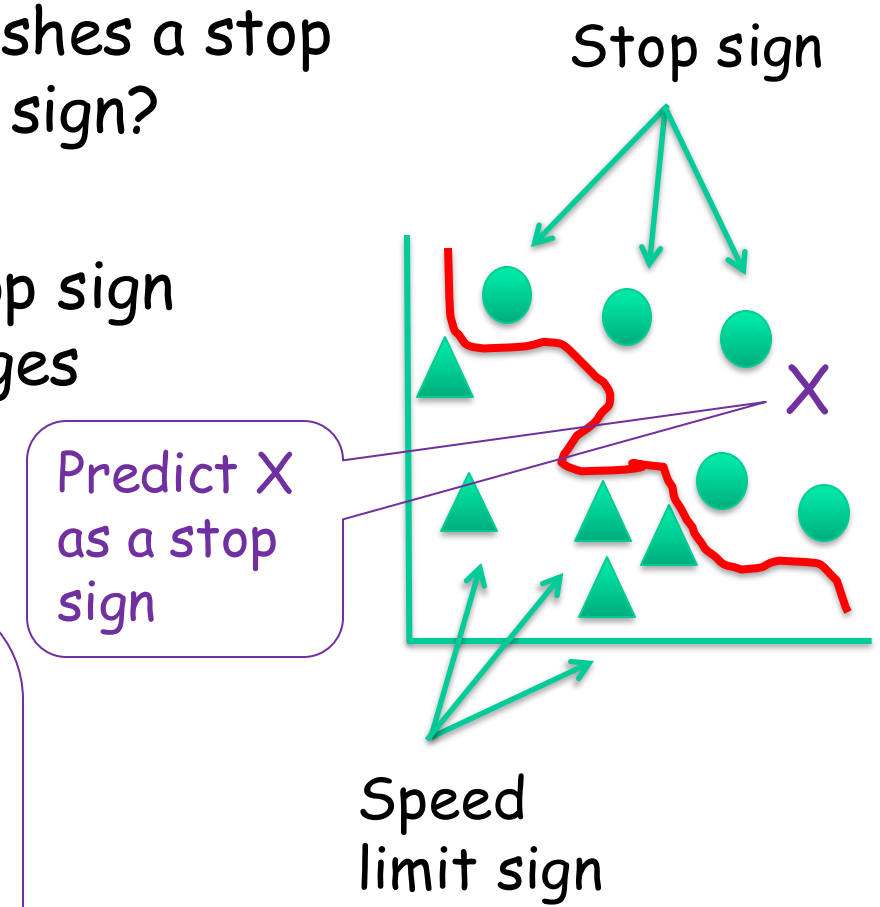
E.g. How an AI system distinguishes a stop sign from an 80mph speed limit sign?

<u>One approach is:</u>

(i) Collect a lot of different stop sign and 80mph speed limit sign images

(ii) Embed the images into a k-dimensional graph

E.g. Scale the images into the same size, each pixel carries a value (which represents the color), compute a set of representative values for the image (e.g. sum the pixels in each row)
=> $(s_1, s_2, \ldots, s_n)$



(iii) SVM tries to find a boundary to distinguish two sets of images

Remarks:

- More data => models/prediction more accurate
- Most of the these models/tools are generic. However, different models may fit better for different applications/datasets.

E.g. For SVM, we assume that there should be a big enough gap between the data points (after embedding) of two classes of objects (images)

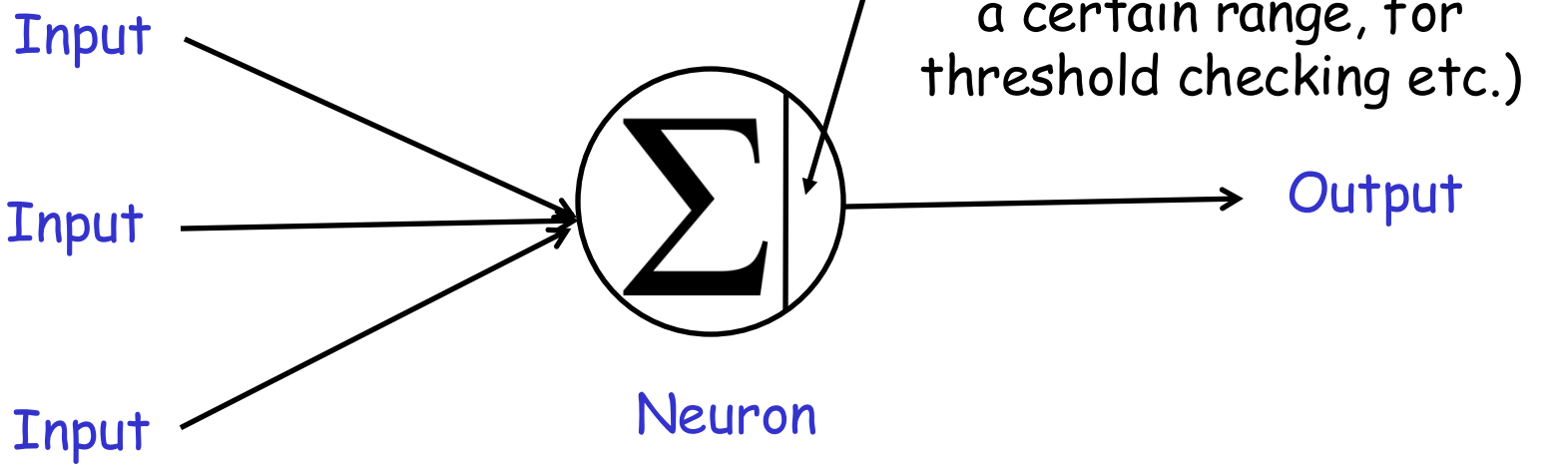
In this lecture, we briefly introduce Artificial Neural Network (ANN)* (the core for deep learning)

Also, a big area: in general, machine learning can be classified as (i) Reinforcement learning; (ii) unsupervised learning; and (iii) supervised learning

* Some slides are borrowed from Dr. Dirk Schnieders

The idea of Artificial Neural Network is motivated from the biological neuron in our body. Roughly speaking, signals are collected by different sensors, then after integration & processed, if the accumulated signal exceeds a certain threshold, an output signal will be generated if .
[Details omitted]

Artificial neuron



Example: Exam score prediction

Consider the problem of predicting the scores (Y) in your final exam based on

- The number of hours you study (x1); and
- The number of hours you slept the night before (x2)

We have collected some previous data already:

- You slept 3h, studied 5h and scored 75
- You slept 5h, studied 1h and scored 82
- You slept 10h, studied 2h and scored 93

The assumption behind is that somehow the values of x1 and x2 affect the final value Y.

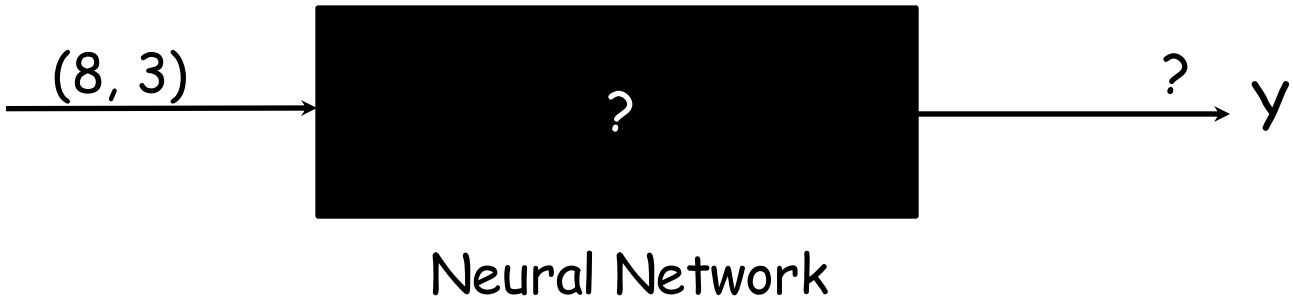
E.g. Maybe x1 (w1) + x2 (w2) => Y (+/-) small errors?

Training and Prediction

Training

x = (x1, x2) (hours sleep, hours study)	y (test score)
(3, 5)	75
(5, 1)	82
(10, 2)	93

Prediction (Testing)



Input, output, and hidden layers

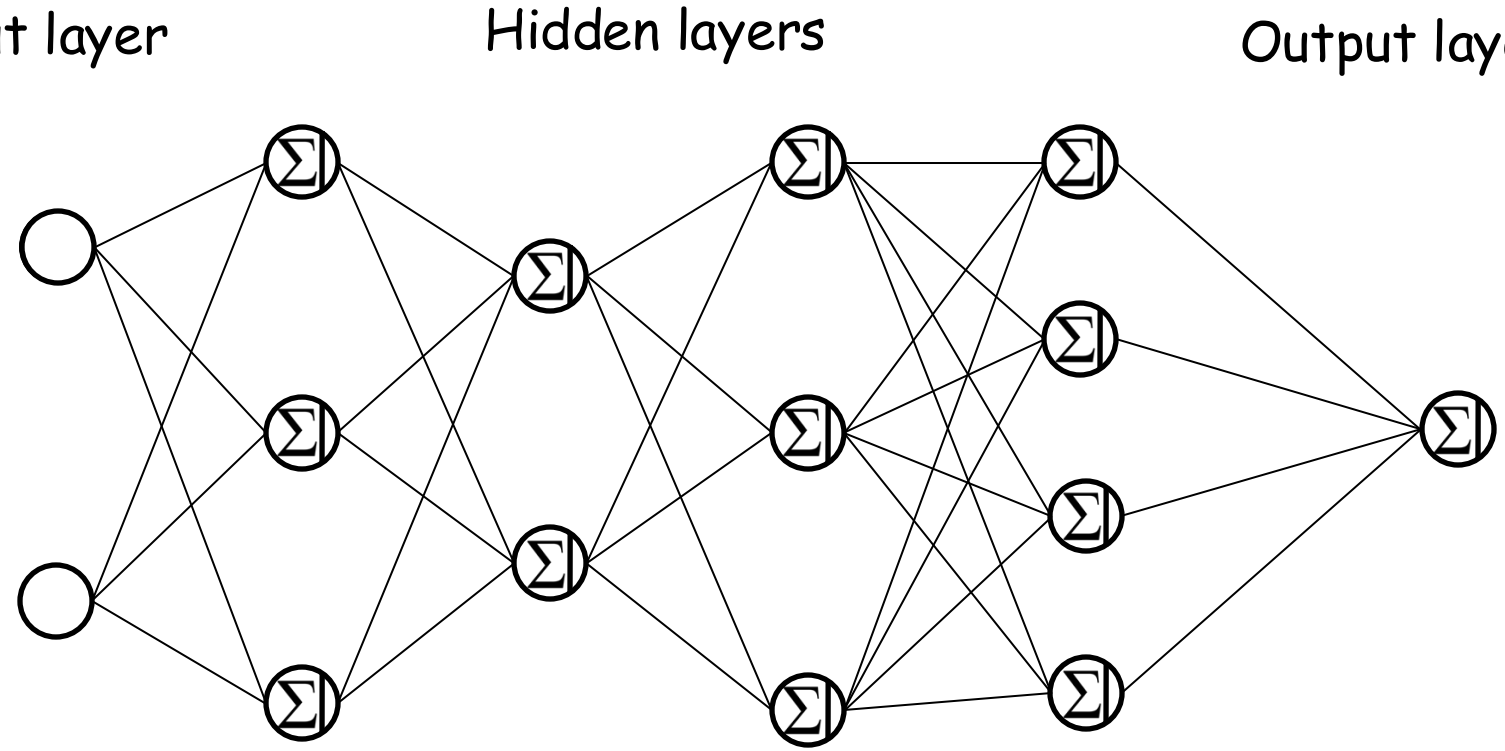
In our case, we have two inputs (x1, x2) and one output (Y)

Of course, we hope that we can train the network so that when we input (5, 1), Y = 82 +/- ε; or when we input (10, 2), Y = 93 +/- ε, where ε should be small enough for our application.

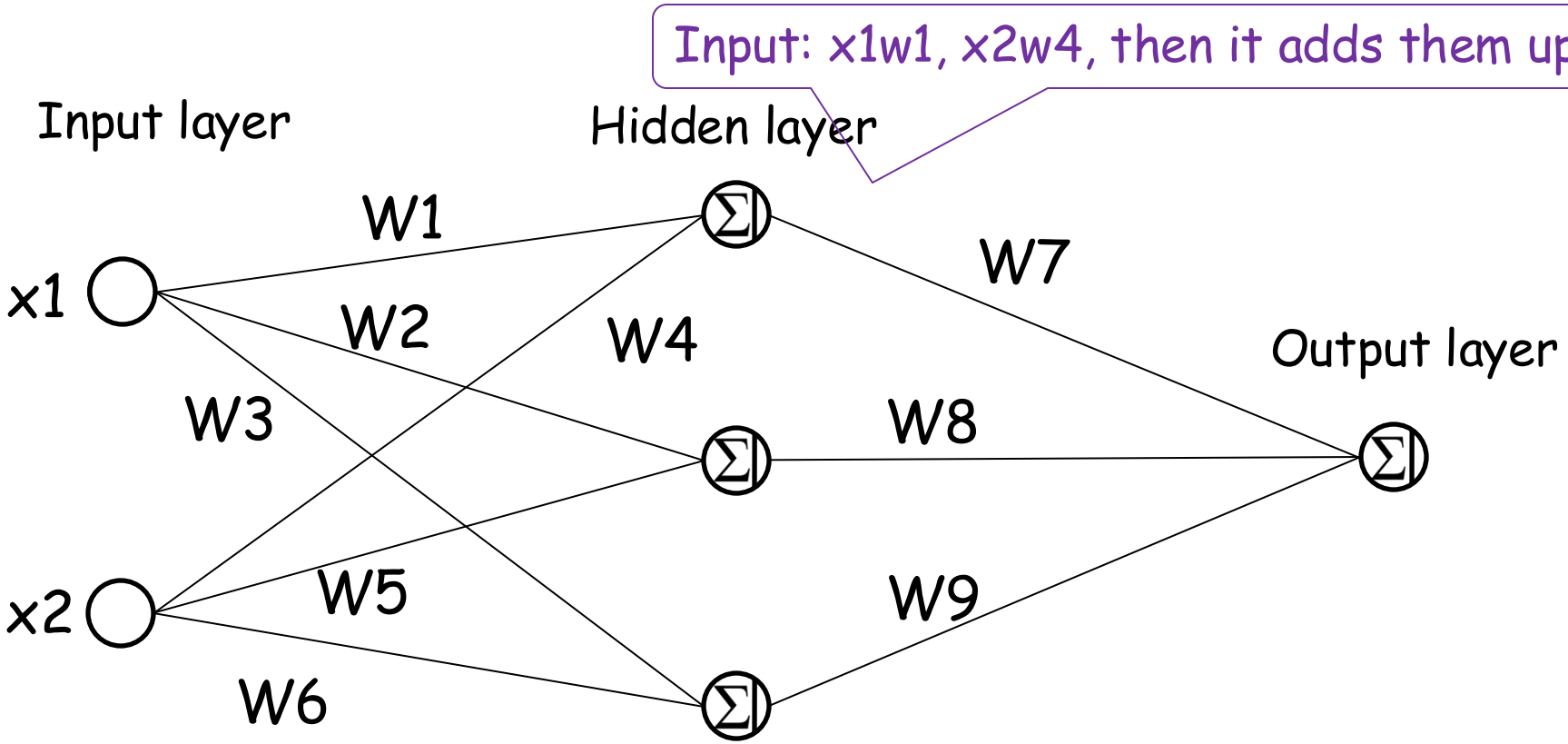
[Testing/Prediction] If the network has been trained successfully, we hope that if we input (8, 3), it will give us an accurate enough predicted score.

So, usually we divide the collected data into k groups, one used for testing and the others for training, repeat for each group (check out what is k-fold cross validation)

Any layer(s) between input layer and output layer is called a hidden layer!



In our problem, we only use one hidden layer for illustration:



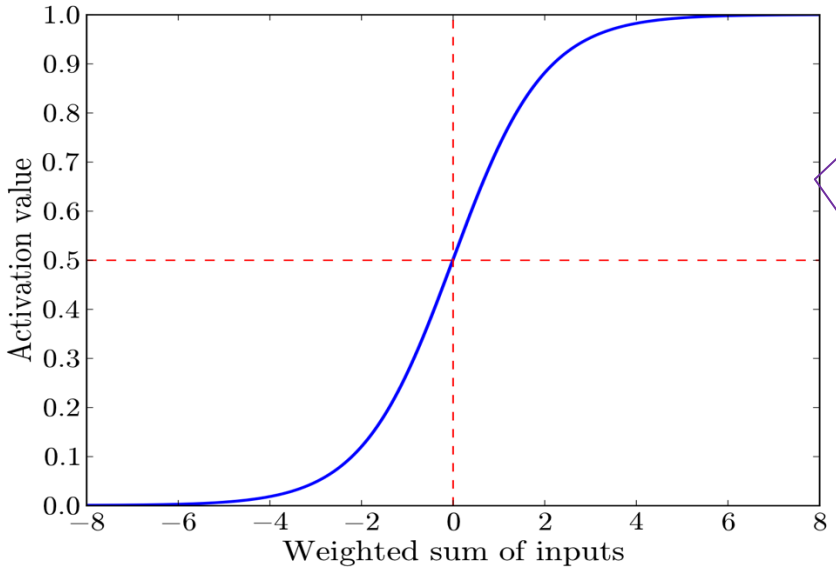
Recall that we assume x1(w1) + x2(w2) => Y +/- error

But we don't know how to set the weights, so we let the network to learn based on fitting enough examples to it. (how? See the later slides)

Remarks:

- We will see how we set the weights later
- Recall that each neuron has an activation function which will normalize the sum of inputs and produce an output to the next layer.
- In our example, 2 inputs, 1 hidden layer with 3 units, 1 output <= hyperparameters, to be determined before the training phase.

An example and frequently used activation function

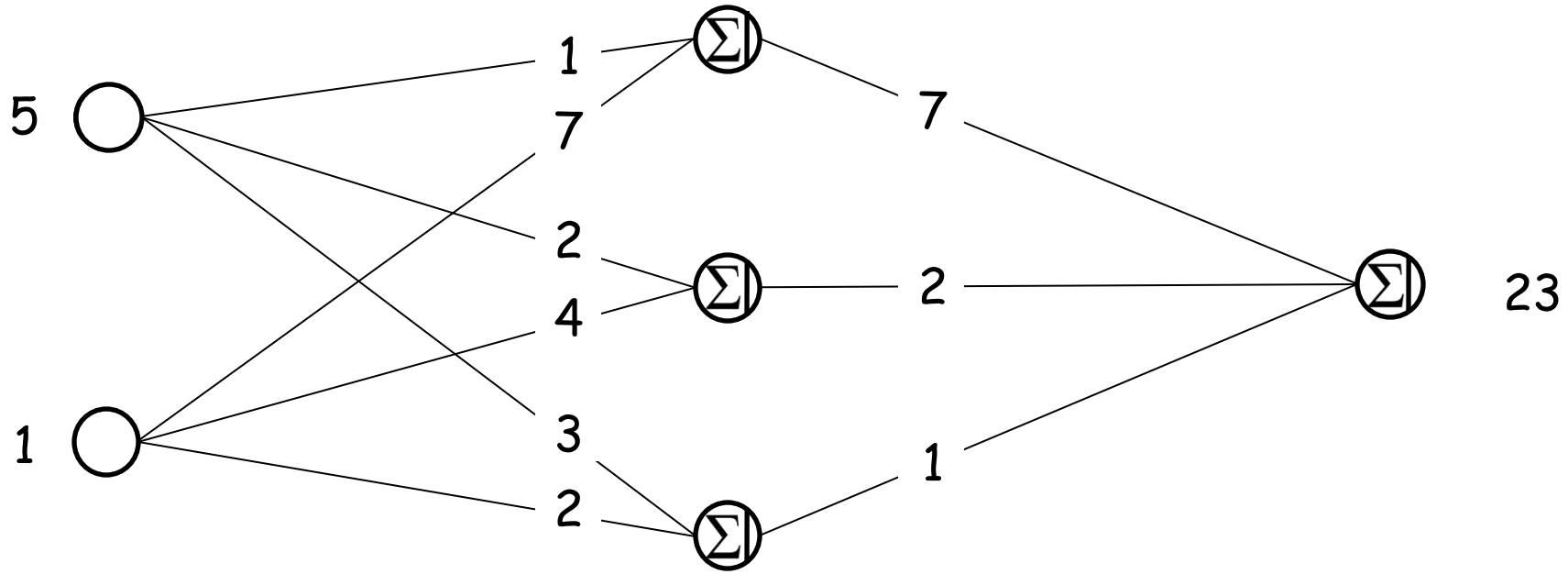


Here, we restrict the output to be [0..1], for instance, can later be used to do classification (either it is a stop sign or an 80mph limit sign)

How to train a network?

- Initialize the weights with random values
  - ❖ Of course, will make bad predictions

E.g.

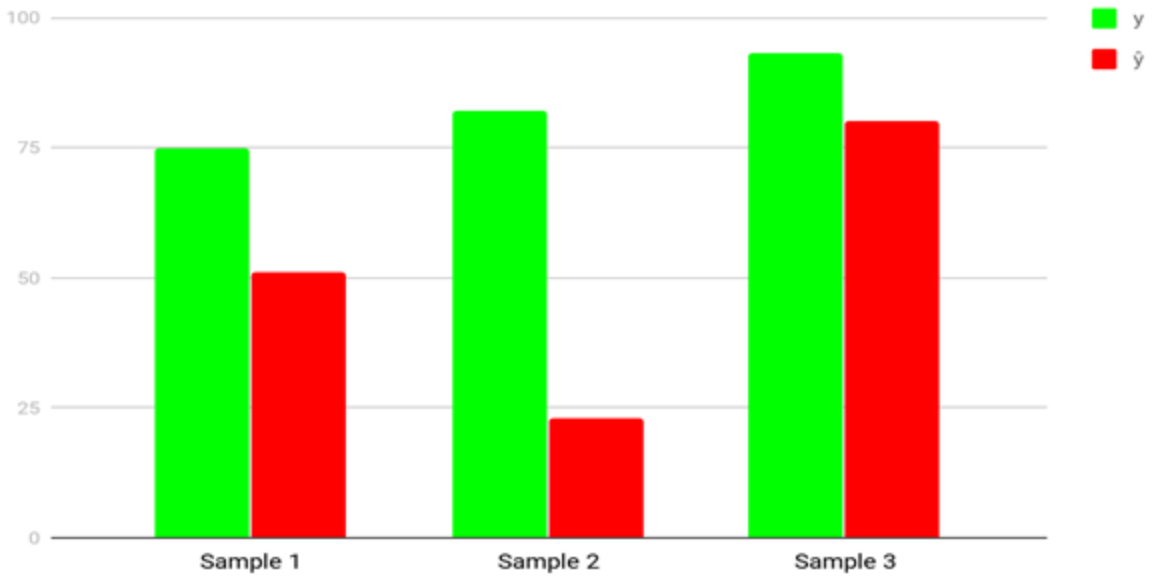


Recall that (5, 1) => Y = 82

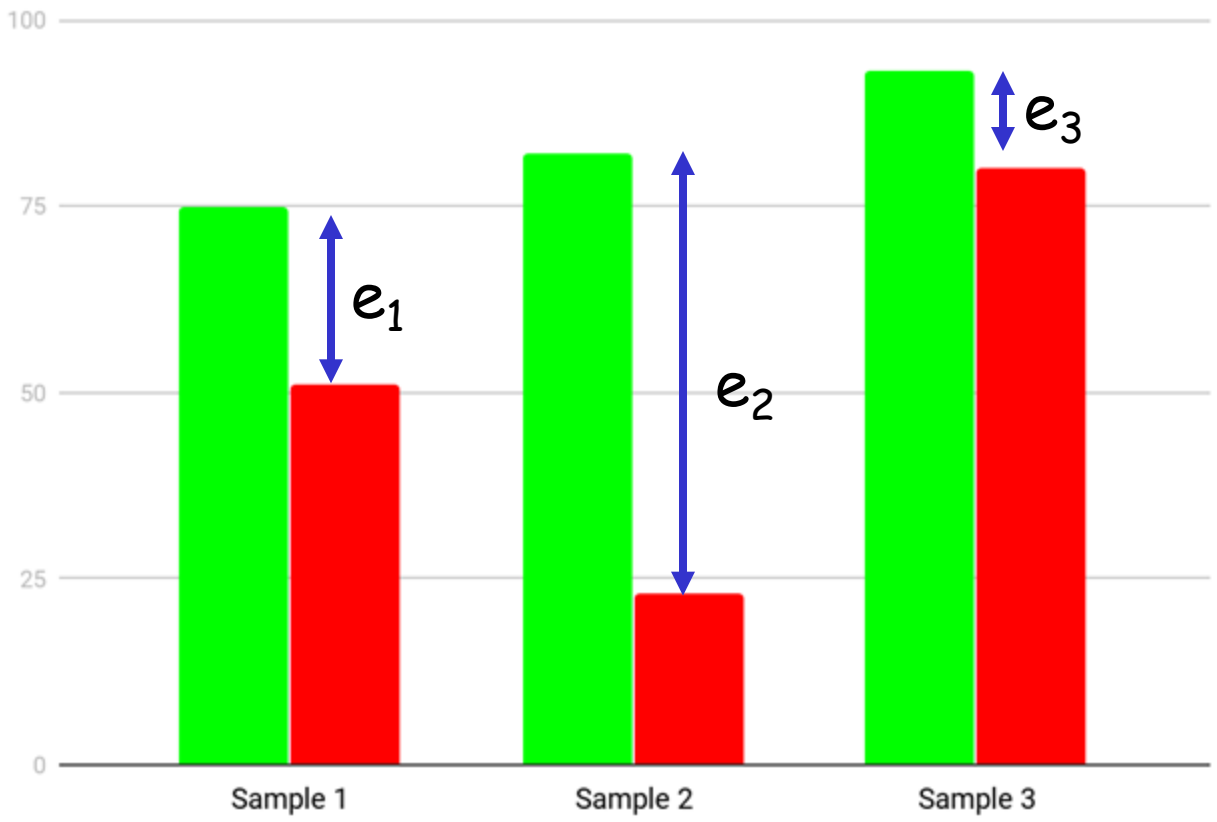
Error

- To improve the model, we first quantify the error
  - I.e., quantify how wrong the predictions are

x (hours sleep, hours study)	y (test score)	$\hat{y}$ (estimated test score)
(3, 5)	75	51
(5, 1)	82	23
(10, 2)	93	80



We use a cost function J to tell us how wrong our model is:



$$e_i = y_i - \hat{y}_i$$

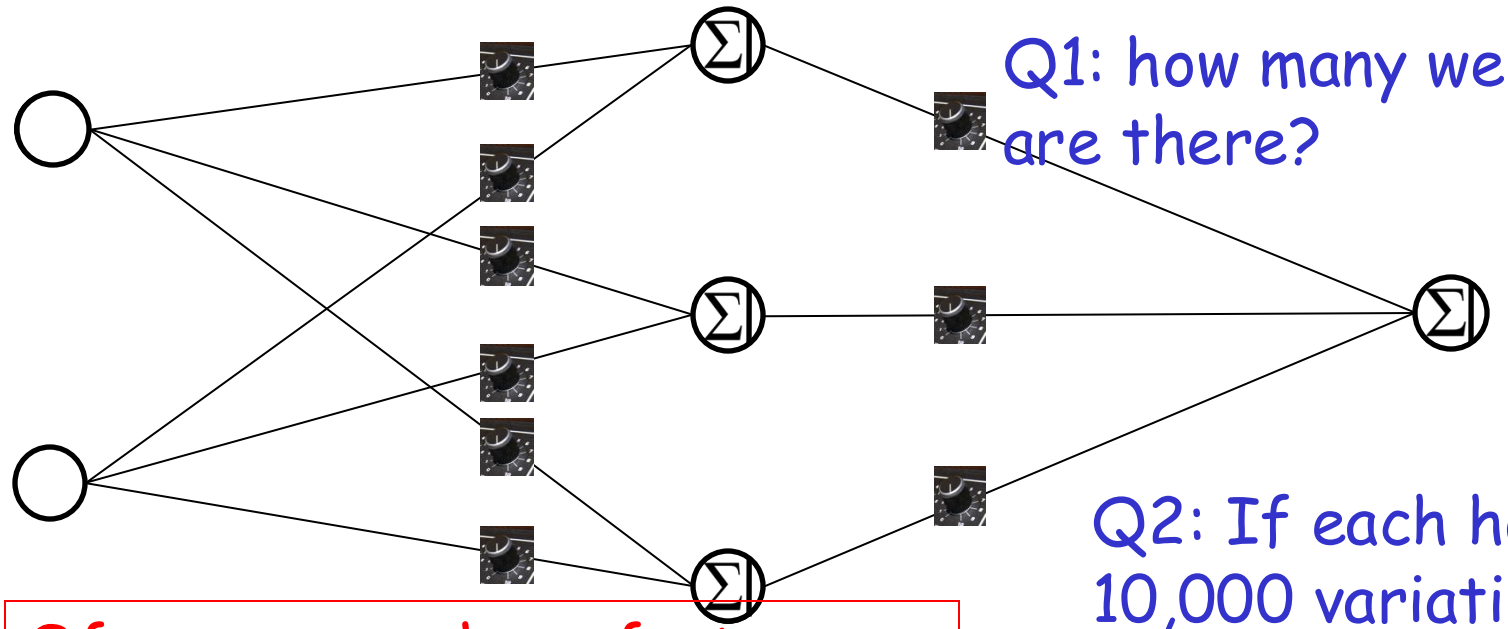
$$J = e_1^2 + e_2^2 + e_3^2$$

The whole idea is to tune the weights so that the error is minimized.

Training the network = minimizing the cost function J

One naïve approach: Try all possible values of weights

It will take too long ☹!



Q1: how many weights are there?

Q2: If each has 10,000 variations, how many combinations?

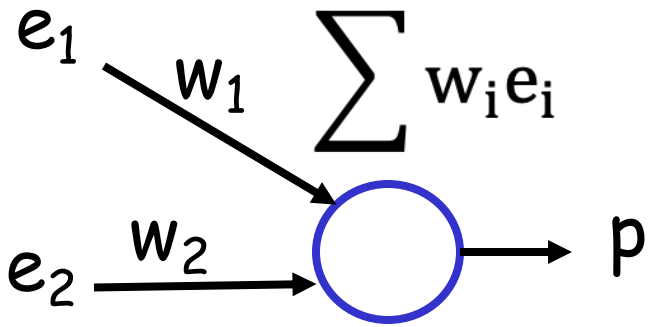
Of course, we have faster method to do it (e.g. Gradient Descent) [not covered!]

Last remarks

- If the network has more hidden layers, the number of parameters (weights) increases => we need more data in order to have an accurate estimation!

- How many layers are good enough?
  - ❖ More layers can capture more complicated relationship, but require more training data

E.g. # of parameters >> # of training data



If we only have one training record: (1, 2, 0)

We can have infinitely # of answers for weights:(w_1, w_2): (0, 0), (1, -0.5), (2, -1), (3, -1.5), (-1, 0.5), (-2, 1), …….

Many applications for AI in FinTech

- AI-based customer service
- AI-based robot advisor for investment
- AI-based quantitative trading
- AI in InsurTech

- AI in face recognition for virtual bank (opening an account)
- AI in risk assessment
- ……..

From: https://tech.pingan.com/en/news/392513296486.shtml

But do NOT forget that there are attacks on AI systems => (i) Is the AI system reliable; (ii) be careful if it is used for critical infrastructures/ large investment (decision not always explainable!!)