Course 1: Neural Networks & Deep Learning

Week 1

price ↑ (graph vs size of house)
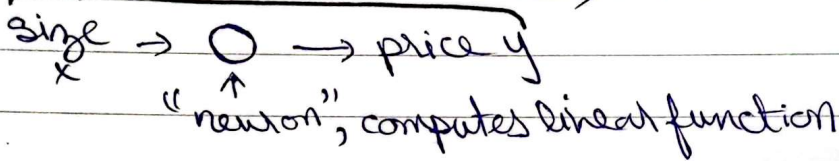
size of house

Q. Why not linear regression for predicting housing price?

— since price cannot be −ve

(graph)

Single neuron

size $\xrightarrow{x}$ O $\longrightarrow$ price y
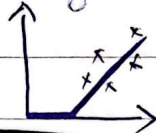
"neuron", computes linear function

→ Relu function (Rectified Linear Unit) → taking a max of 0

many neurons:

size
bedroom → family size

X {

zip code
(postal code) → tells walkability

wealth → school quality

→ price (y)

hidden unit (takes all 4 input features) → densely connected in hidden unit

x_1
x_2
x_3
x_4

(network diagram) $\longrightarrow y$
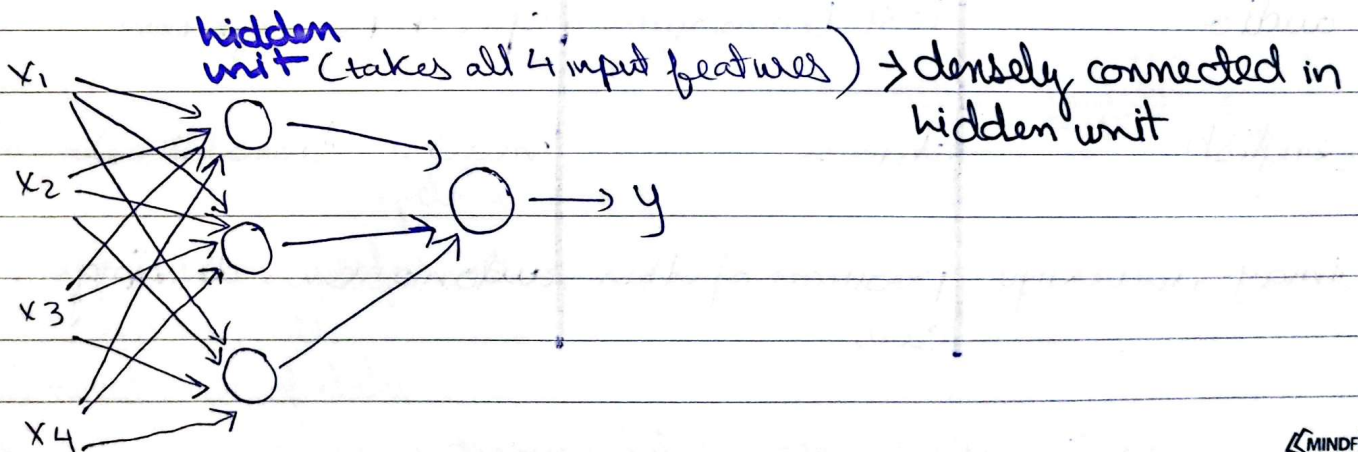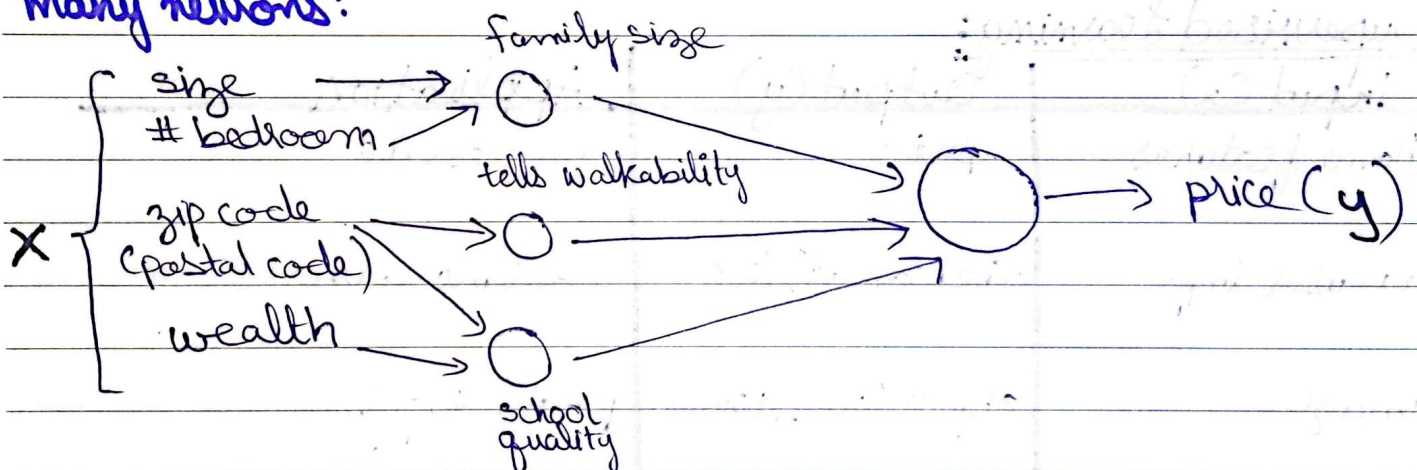
- Given enough data $(x, y) \rightarrow$ neural networks are good at figuring out functions.
- Used in supervised learning alot, but only in supervised learning? No, they work across all major learning paradigms.

  eg: supervised/unsupervised
  - autoencoders / VAEs
  - contrastive learning
  - masked prediction
  - diffusion learning
  - reinforcement learning
  - semi-supervised learning
  - clustering.

Supervised Learning:

| Input (x)	Output (y)	Application
home features	price	real estate
ad, user info	click on ad? (0/1)	online advertising
image	object (1,...100)	photo tagging
audio	text transcript	speech recognition
english	chinese	machine translation
image, radar info	position of other cars	autonomous driving.

real estate → could use standard NN

image ⟶ CNN

sequence data,
can have temporal ⟶ RNN
component
eg: audio, language
⟶ complex RNN

for autonomous ⎫⟶ Custom/Complex
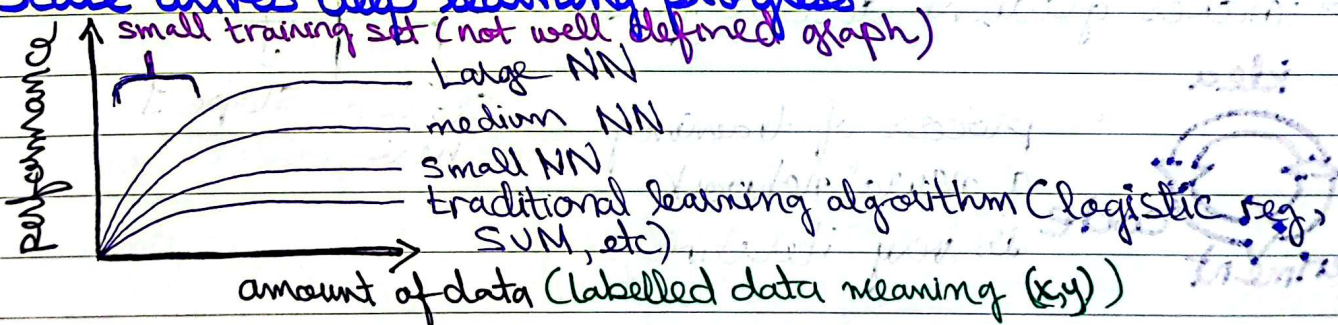driving ⎭ NN architecture
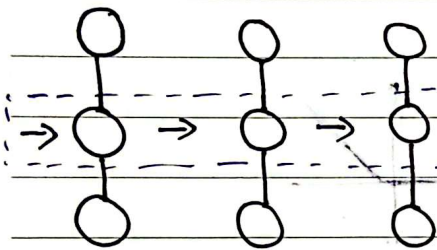
RNN looks something
like this.

Structured data	Unstructured data
database, table, column	audio, images, text

⇒ computers are now better at interpreting unstructured data.

Scale drives deep learning progress

small training set (not well defined graph)
Large NN
medium NN
Small NN
traditional learning algorithm (logistic reg,
SVM, etc)

Performance (y-axis)

amount of data (labelled data meaning (x,y))

For high level of performance, you need 2 things:
1) train big NN
2) need alot of data

⇒ So scale has been driving NN (scale meaning size
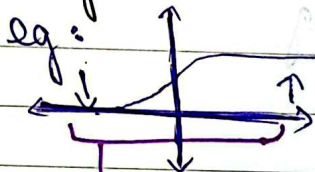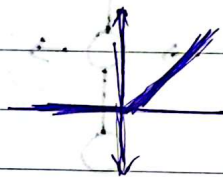of NN or data).

for small datasets:
relative ordering of algo is not well defined, so if you have a small dataset, it is often upto yourself skills & hand engineering features that determines its performance. Also dependent on lower details of that algorithm.

why deep learning is taking off now:
- data ✓
- computation ✓
- algo ✓

eg: Switch from sigmoid to relu.

why switching?
regions where gradient is nearly 0, learning becomes really slow

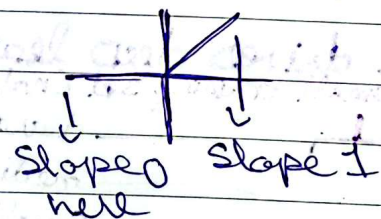changing activation function to it; gradient = 1 for all positive values of input
⇒ makes gradient descent faster

idea
process of training a neural network is very iterative

Slope 0 Slope 1
here

experiment code

→ idea for NN architecture ⇒ code ⇒ run experiment

result can take an hour, day, month