Statistics and Artificial Intelligence

Lecture 11: Getting Started with Neural Networks

Yixin Wang

Roadmap for Today

Last 2 lectures on shallow neural nets this week

- JiTT
- Open the blackbox: How do the calculation/vectorization translate into code?
- Getting Started with NNs: End-to-end examples

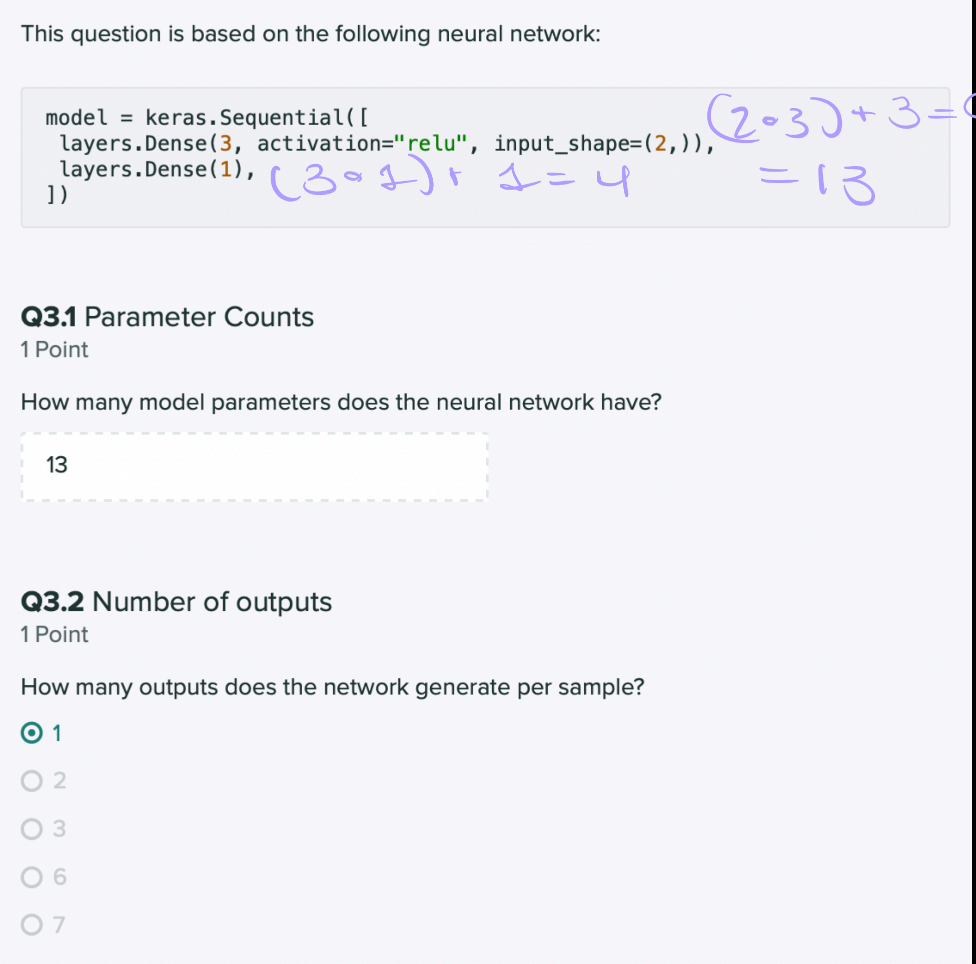
JiTT

JiTT

One layer

weight = input · output

bias = outputs



This question is based on the following neural network:

```
model = keras.Sequential([
 layers.Dense(3, activation="relu", input_shape=(2,)),
 layers.Dense(1),
])
```

$(2\cdot3)+3=9$

$(3\cdot1)+1=4$

$=13$

Q3.1 Parameter Counts

1 Point

How many model parameters does the neural network have?

13

Q3.2 Number of outputs

1 Point

How many outputs does the network generate per sample?

● 1

○ 2

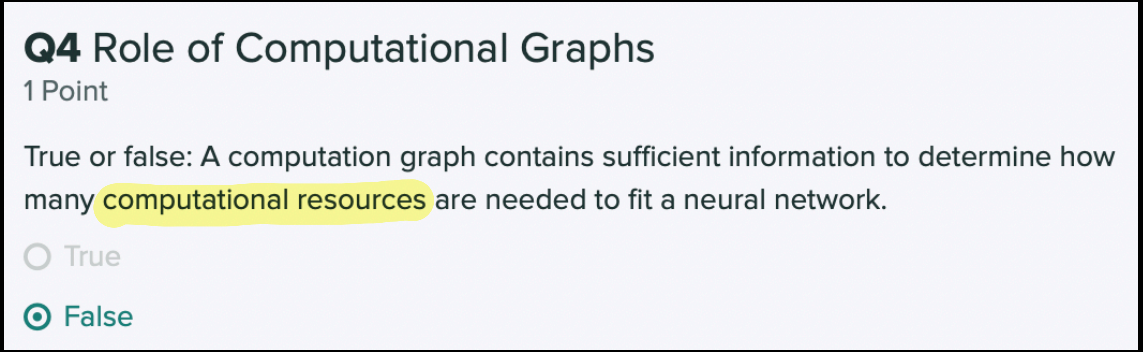
○ 3

○ 6

○ 7

JiTT

inputs
↳ intermediate calculations
↳ final output



Q4 Role of Computational Graphs

1 Point

True or false: A computation graph contains sufficient information to determine how many computational resources are needed to fit a neural network.

○ True

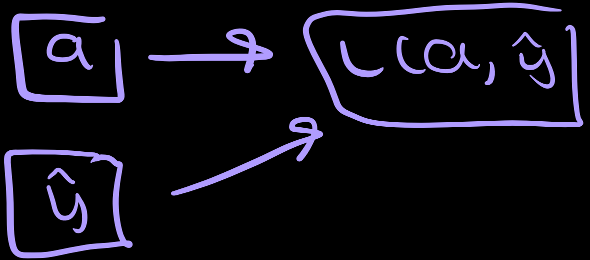
● False

Graph isn't about computational resource

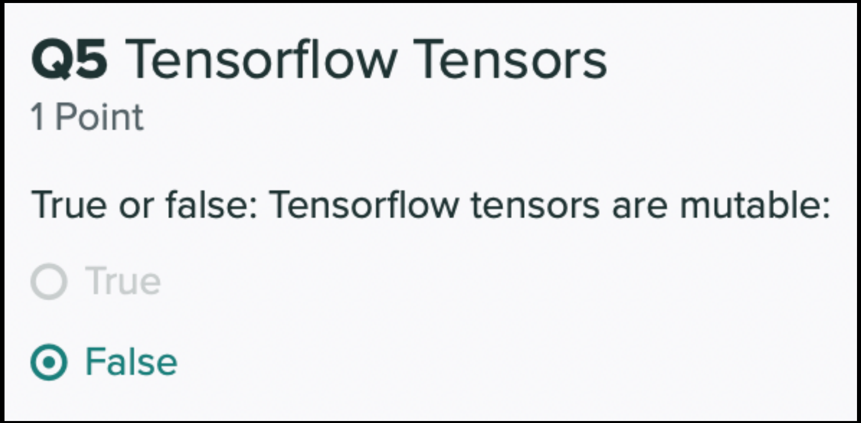
↓

same program can run faster or slower if its done via vectorization or a for loop

Graph shows us operations needed to get what we want



JiTT



Q5 Tensorflow Tensors

1 Point

True or false: Tensorflow tensors are mutable:

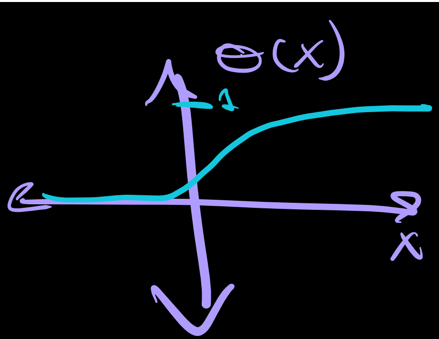
○ True

● False

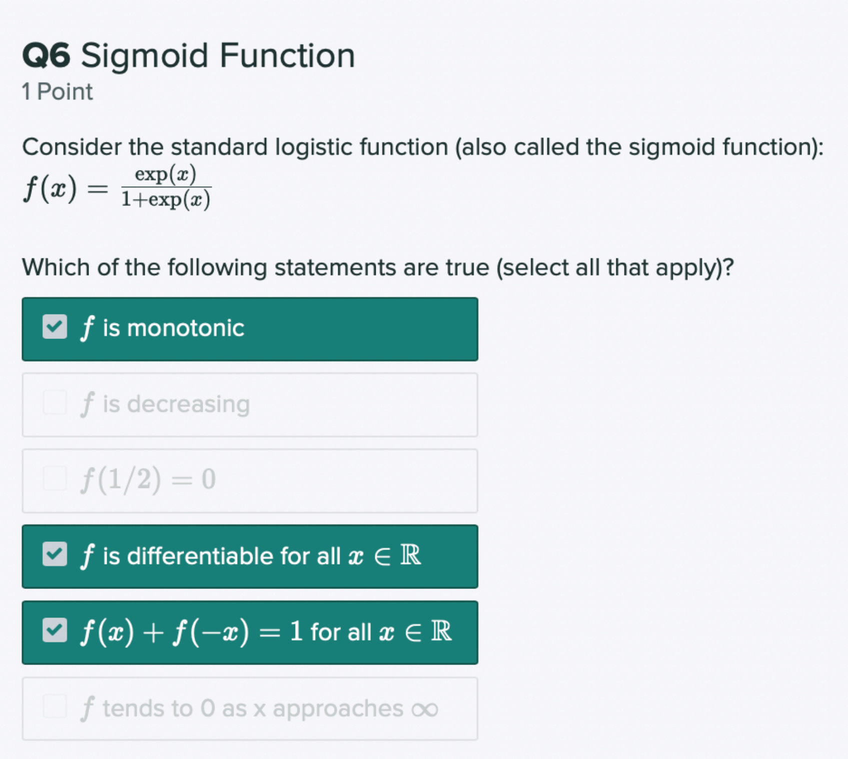
NP arrays are mutable

Tf variables are mutable, but tensors are immutable

JiTT



$$\sigma(x) = \frac{\exp(x)}{1+\exp(x)}$$



Q6 Sigmoid Function

1 Point

Consider the standard logistic function (also called the sigmoid function): $f(x) = \frac{\exp(x)}{1+\exp(x)}$

Which of the following statements are true (select all that apply)?

☑ f is monotonic

□ f is decreasing

□ $f(1/2) = 0$

☑ f is differentiable for all $x \in \mathbb{R}$

☑ $f(x) + f(-x) = 1$ for all $x \in \mathbb{R}$

□ f tends to 0 as x approaches ∞

monotonic: larger x → larger $\sigma(x)$
smaller (x) → smaller $\sigma(x)$

$f(x) + f(-x)$

$$\frac{\exp(x)}{1+\exp(x)} + \frac{\exp(-x)}{1+\exp(-x)} \cdot \frac{\exp(x)}{\exp(x)}$$

$$\frac{\exp(x)}{1+\exp(x)} + \frac{1}{\exp(x)+1}$$

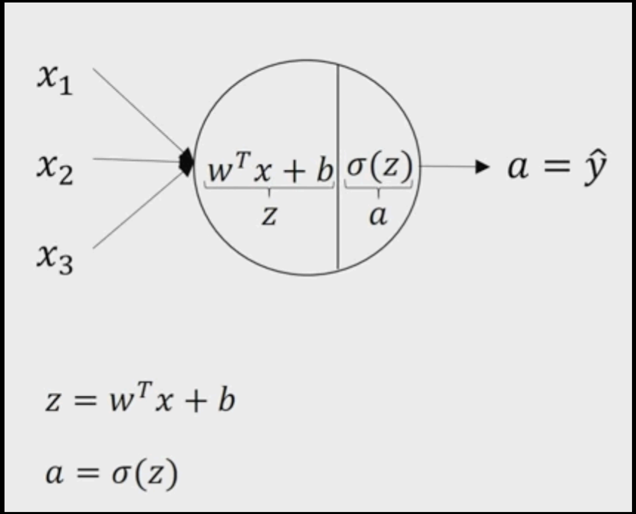
Function is symmetric

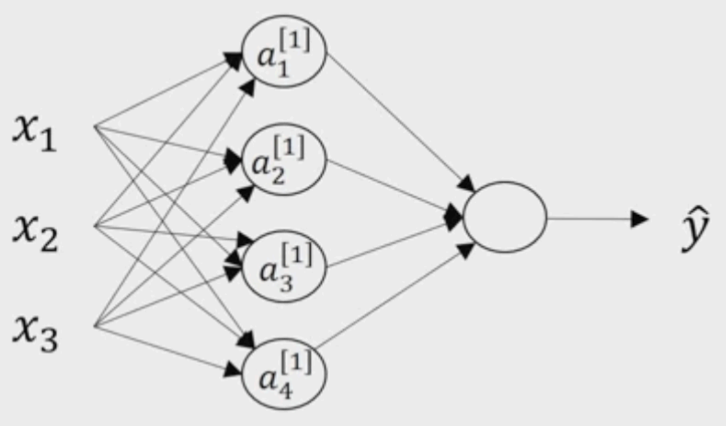
Open the black box of Sequential( )

Computing a Neural Network's Output

each neuron does
same computation
→ apply activation

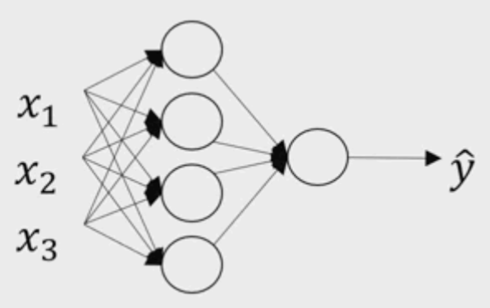
- Logistic Regression vs Neural Network





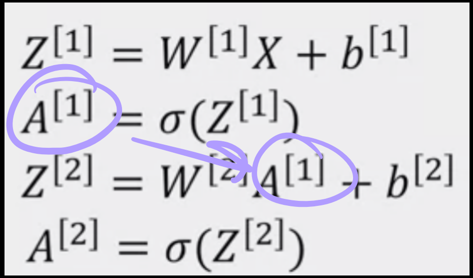
(Vectorized) Forward Pass and Backward Pass

Backward pass is also known as backpropagation.



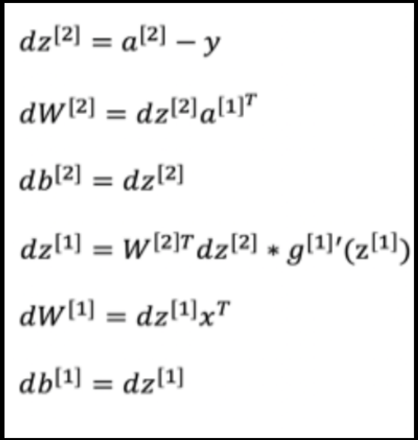
output is input to next layer

Forward Pass

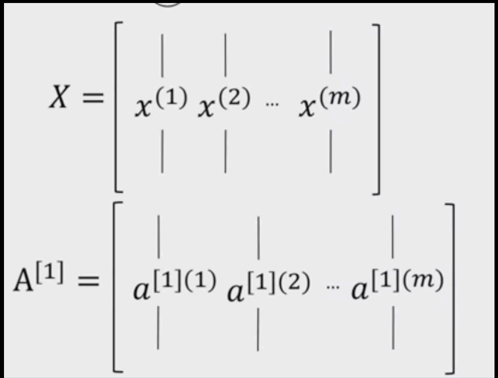


$$Z^{[1]} = W^{[1]}X + b^{[1]}$$
$$\boxed{A^{[1]}} = \sigma(Z^{[1]})$$
$$Z^{[2]} = W^{[2]}\boxed{A^{[1]}} + b^{[2]}$$
$$A^{[2]} = \sigma(Z^{[2]})$$

Backward Pass



$$dz^{[2]} = a^{[2]} - y$$
$$dW^{[2]} = dz^{[2]}a^{[1]^T}$$
$$db^{[2]} = dz^{[2]}$$
$$dz^{[1]} = W^{[2]T}dz^{[2]} * g^{[1]\prime}(z^{[1]})$$
$$dW^{[1]} = dz^{[1]}x^T$$
$$db^{[1]} = dz^{[1]}$$



$$X = \begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \cdots & x^{(m)} \\ | & | & & | \end{bmatrix}$$

$$A^{[1]} = \begin{bmatrix} | & | & & | \\ a^{[1](1)} & a^{[1](2)} & \cdots & a^{[1](m)} \\ | & | & & | \end{bmatrix}$$

Organize everything into matrices

How does it relate to the code we wrote?

- This is what happens under the hood of model.compile(), model.fit(), etc
- You will need to implement it if you need to work with a customized neural network with special activation or layer structures.
- https://colab.research.google.com/drive/1t2cobScOH7GUpSmglhQ9kg_cIpvl4IeM?usp=sharing

Code Demo

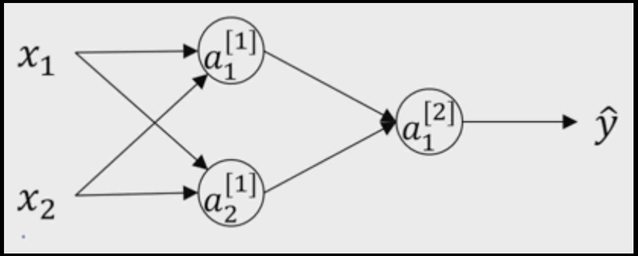
Homework will be going through custom neural nets

Look at batches of data points

Random Initialization

What happens if you initialize weights to zero

- All hidden units will be computing the same function due to symmetry.
- There will be no point to have multiple hidden units.
- Must perform random initialization.



All nodes calculated the same

→ all gradient calculations will be symmetric

→ as if we only have 1 neuron

Getting Started with NNs: End-to-end Examples

Getting Started with NN

end-to-end examples

- MNIST
  - https://colab.research.google.com/drive/1Z3K-49NoflTiyzK65fXHhFYb6dT3SID7?usp=sharing
- IMDB
  - https://colab.research.google.com/drive/1MC4ngpTT5ulpyTunCWr3c-huaGdwqhqd?usp=sharing
- Boston Housing
  - https://colab.research.google.com/drive/1CW9gKd-wCEFy_PWVg8wr82ChL6U5QdCC?usp=sharing

How does neural network training vary with the parameters?

- playground.tensorflow.org

The materials in this course are adapted from materials created by Alexander Amini, Alfredo Canziani, Justin Johnson, Andrew Ng, Bhiksha Raj, Grant Sanderson and the 3blue1brown channel, Rita Singh, Ava Soleimany, and Ambuj Tewari.