Berkeley Engineering | BerkeleyHaas

PROFESSIONAL CERTIFICATE IN MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Module 21: Deep Neural Networks pt. 1

Office Hours with Francesca Vera
February 24, 2025
9:00pm UTC

ICE BREAKER

Welcome, Everyone! Hope you had a good break!

How is your capstone going?

Have you met with your LF?

Any findings?

AGENDA FOR TODAY

My OH aim is to provide you with helpful tips and an overview of the module content *(for you to dive deeper into individually)*

- **Capstone feedback + General Observations**
- **Concepts in Neural Networks**
  - **+ What to expect in pt. 2?**
- **Example Walkthrough**

CAPSTONE ADMIN

Looking forward

Holiday Break - Sunday, December 22, 2024- Sunday, January 05, 2025			
16	Classifying Nonlinear Features	16	Wednesday, January 8, 2025
17	Practical Application 3	17	Wednesday, January 15, 2025
	Natural Language Procession (NLP)	18	Wednesday, January 22, 2025
	Recommendation Systems	19	Wednesday, January 29, 2025
20	Ensemble Techniques	20	Wednesday, February 5, 2025
Break Week - Wednesday, February 12, 2025			
21	Deep Neural Networks 1	21	Wednesday, February 19, 2025
	Deep Neural Networks 2	22	Wednesday, February 26, 2025
23	RNNS and Gans	23	Wednesday, March 5, 2025
24	Capstone	24	Wednesday, March 12, 2025

Initial report due

We are here!

1:1 Meet with LF's

CAPSTONE ADMIN

Next Steps

1) **Submit your Initial Report if you have not done so already**
   a) If you have, review the feedback you received!

2) **Schedule 1:1 with your LF**
   a) Come with questions and comments
   b) Take note of feedback given

3) **Adjust project**
   a) More experiments
   b) Clean everything up
   c) Write your report

CAPSTONE FEEDBACK

Clean up your repo!

- If you want to upload images, create a folder
- Descriptive filename (**colabnotebook.ipynb** or **project.ipynb** vs. **loan_default_predictor.ipynb**)

Use formatting features to your advantage:

"The model features include age, zip code, income, number of previous defaults, amount of loan, years with credit score."

Model Features:

- Age
- Zip Code
- Income
- Number of Previous Defaults
- Loan Amount
- Years with Credit Score

CAPSTONE FEEDBACK

Don't assume anything!
Write as though the person encountering your repo has no idea what your project is about:

- Is my research question clear? Is what I'm predicting clear?
- Is the methodology clear? Do I justify my choices (e.g. model choices, feature selection, evaluation metric etc.)?
- Are my results clear and concise?

Display your results in a helpful format:

RF Accuracy 0.93
KNN Accuracy 0.89
LR Accuracy 0.96

vs...

Model	Train Accuracy	Test Accuracy
Random Forest	0.94	0.93
K Nearest Neighbors	0.88	0.89
Logistic Regression	0.96	0.96

AGENDA FOR TODAY

My OH aim is to provide you with helpful tips and an overview of the module content *(for you to dive deeper into individually)*

- **Capstone feedback + General Observations**
- **Concepts in Neural Networks**
  - **+ What to expect in pt. 2?**
- **Example Walkthrough**

NEURAL NETWORKS

Task:
Predict if a patient has diabetes; Given a dataset with features like:

- Glucose concentration
- Blood pressure
- Skin thickness
- Age
- Outcome

What algorithms could I use?

- Logistic Regression
- SVM
- KNN
- Decision Tree

NEURAL NETWORKS

Task:
Identify lifelike cat images from a dataset of images from an Internet search of “cat images”

Can I use the same algorithms?

How well would they do?

NEURAL NETWORKS

For complex tasks, like those in computer vision or with language, simple classification algorithms often won't work well.

- The data may be too complex
- There may be too many features
- The "patterns" in the data to learn from may be distributed in a way that is not captured by simple classification algorithms

For complicated problems, we use more complicated and powerful models like <u>Neural Networks.</u>

NEURAL NETWORK STRUCTURE

Recall from Linear Regression:

$$h(x) = A^T \phi(x) + b$$

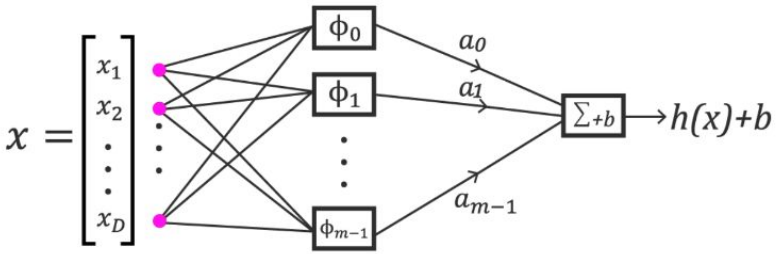
$$h(x) = [a_1 \; a_2 \; \ldots \; a_{m-1}] \begin{bmatrix} \phi_1(x) \\ \phi_2(x) \\ \ldots \\ \phi_{m-1}(x) \end{bmatrix} + b$$

Neural networks use specific feature functions called **activation functions** (Sigmoid, Tanh, ReLU etc.)

Neural networks **add coefficients** to all lines of the graph.

Neural networks allow for **multiple layers**.

Linear regression : $\phi(x)$
Neural network : $\phi(Ax + b)$

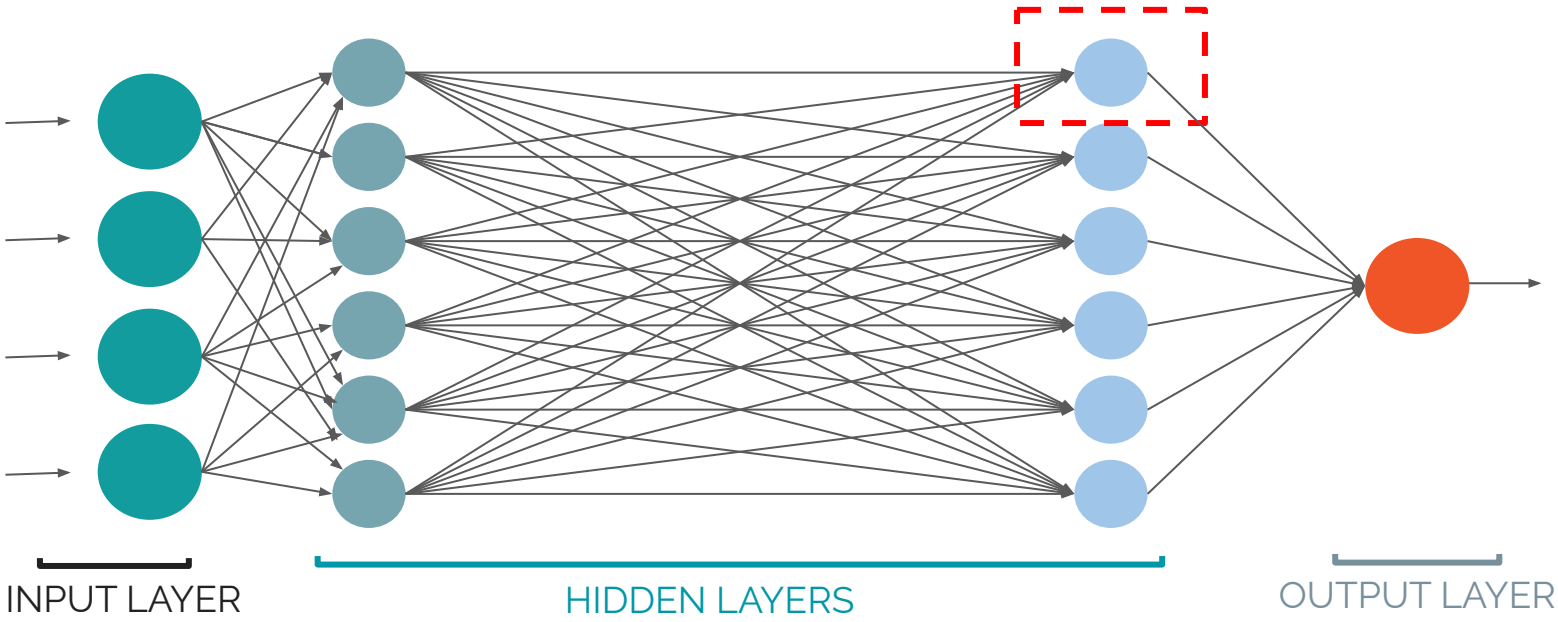


NEURAL NETWORK STRUCTURE

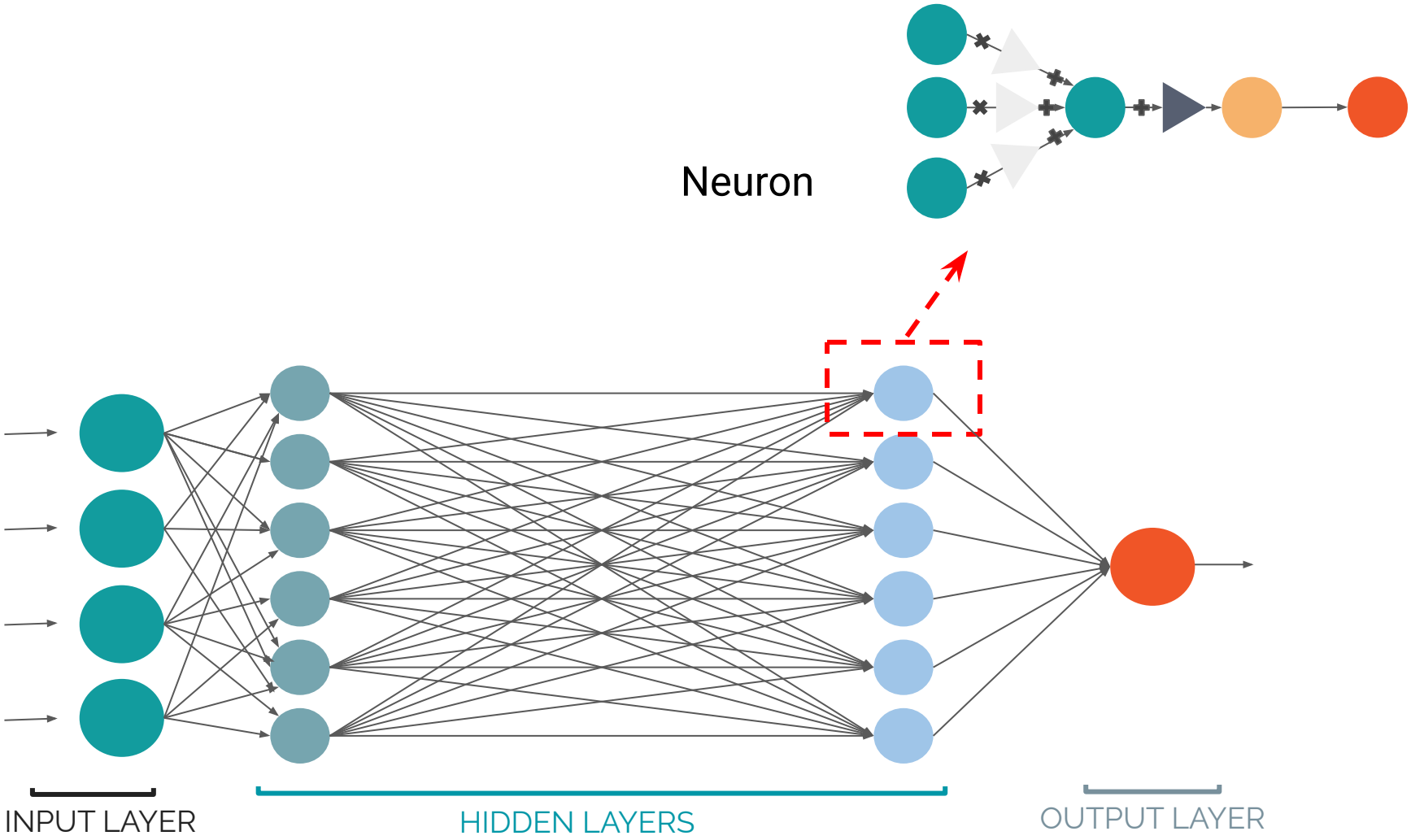
Neural networks use specific feature functions called **activation functions** (Sigmoid, Tanh, ReLU etc.)

Neural networks **add coefficients** to all lines of the graph.

Neural networks allow for **multiple layers**.

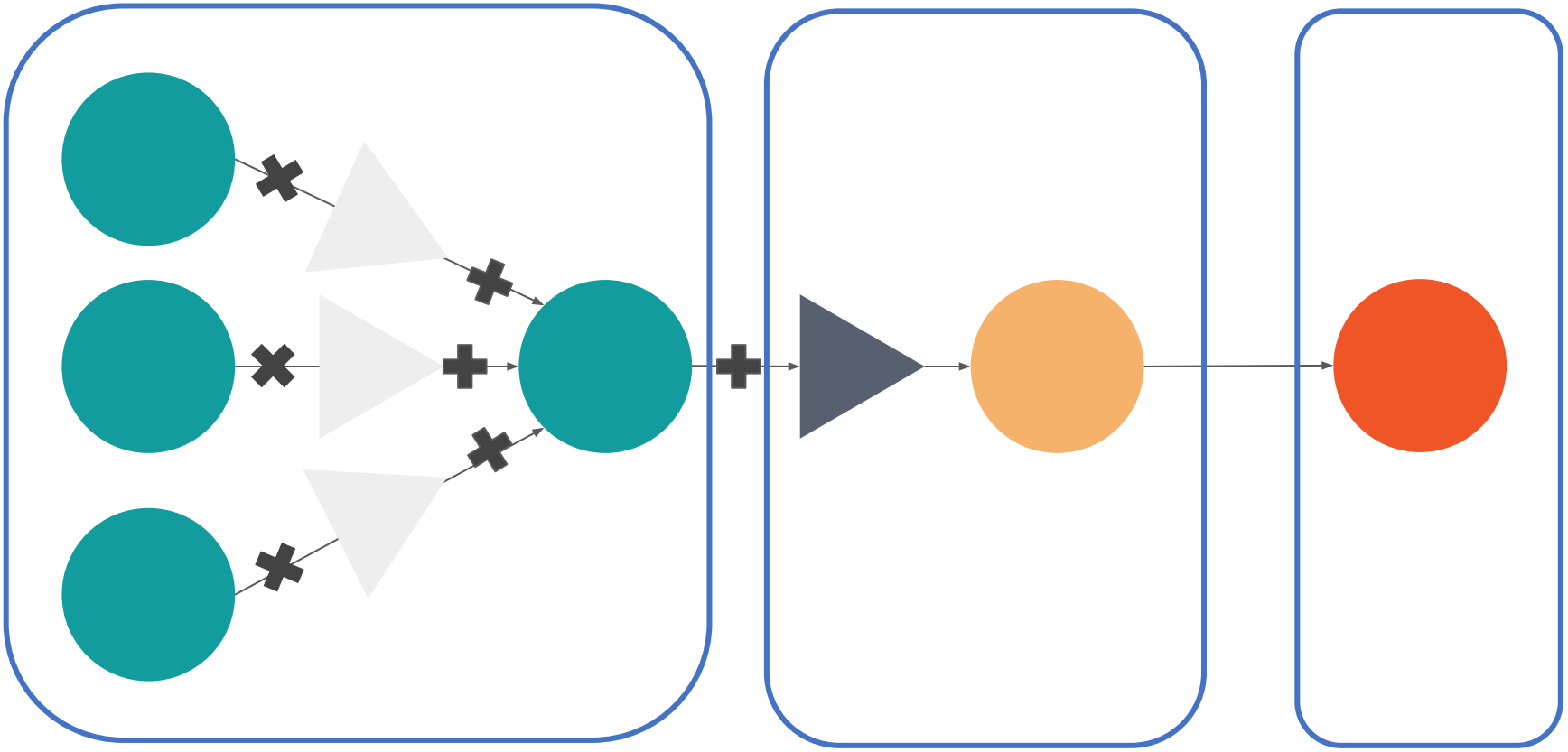


NEURAL NETWORK STRUCTURE



NEURAL NETWORK STRUCTURE

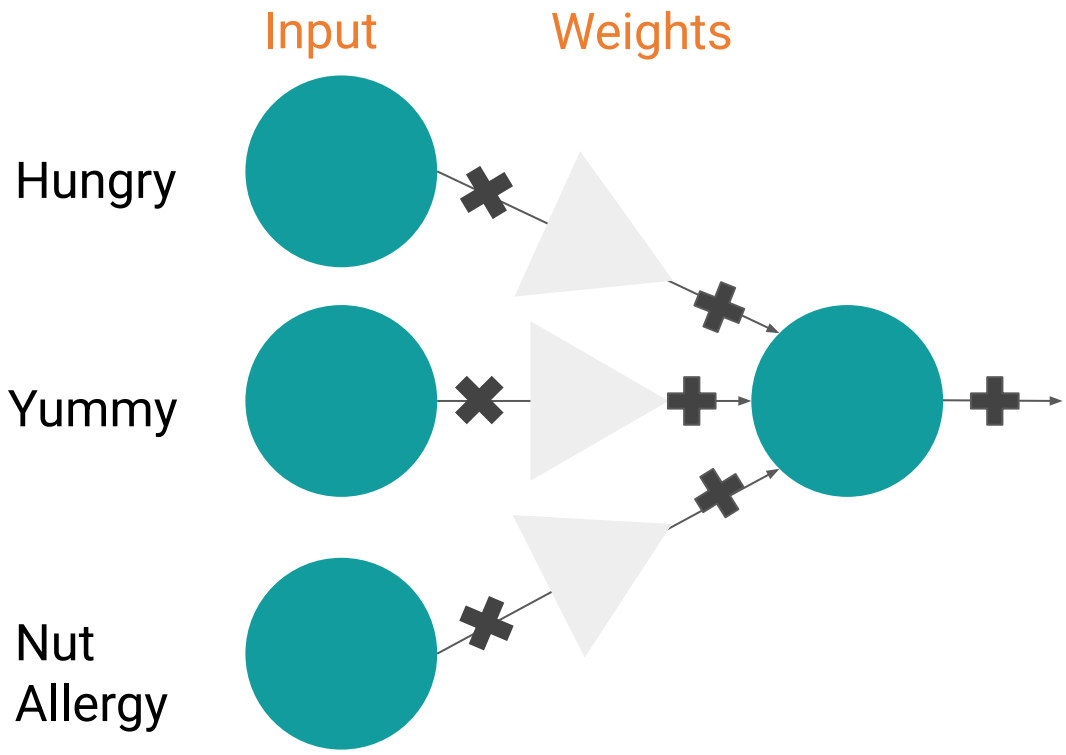
Neuron



NEURAL NETWORK STRUCTURE

Understanding a neuron

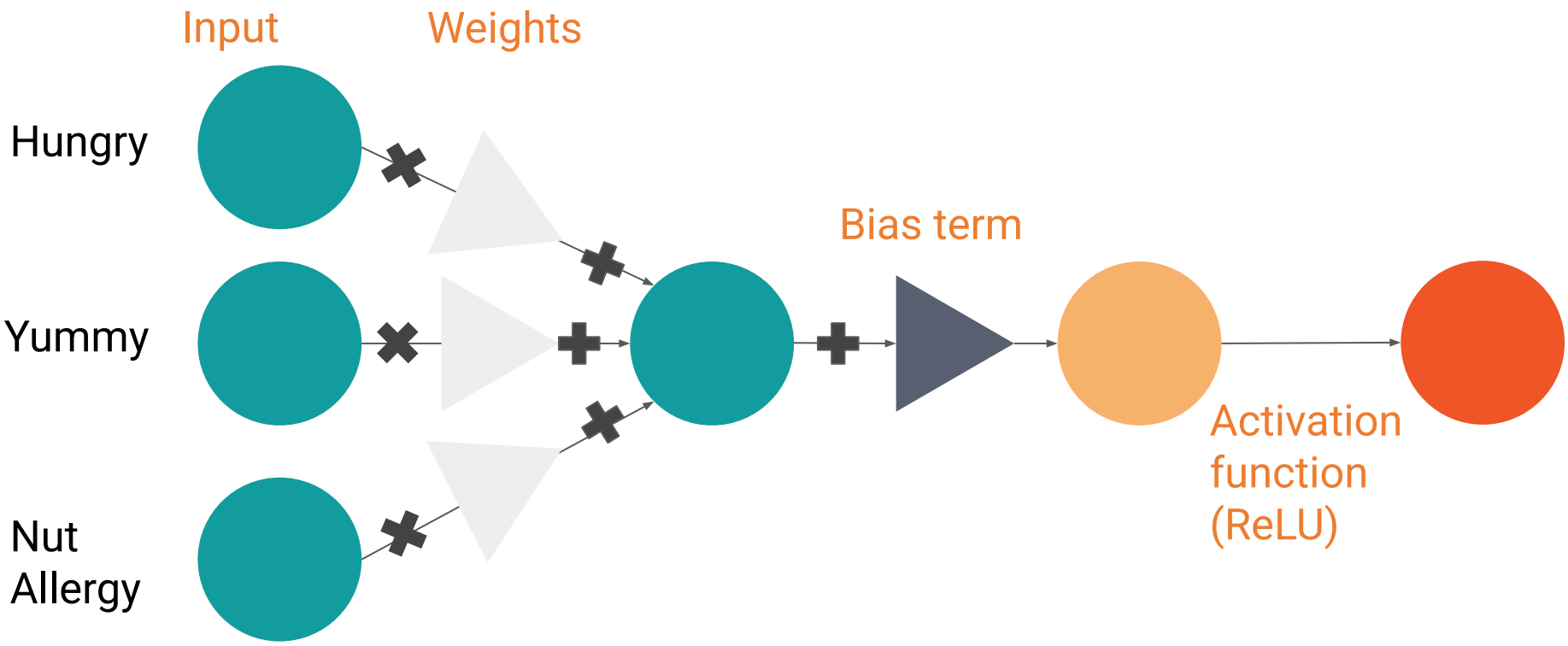
Should I eat this sandwich?



NEURAL NETWORK STRUCTURE

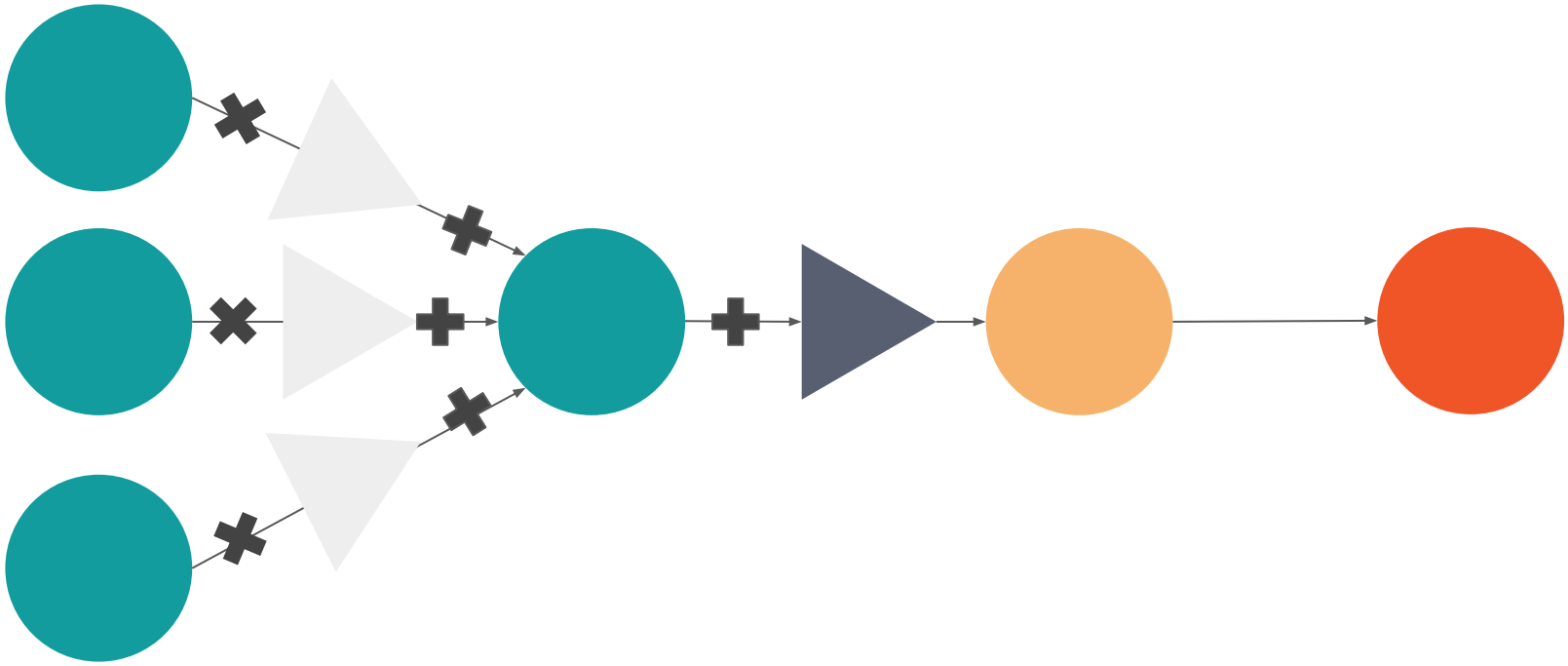
Understanding a neuron

Should I eat this sandwich?



NEURAL NETWORK STRUCTURE

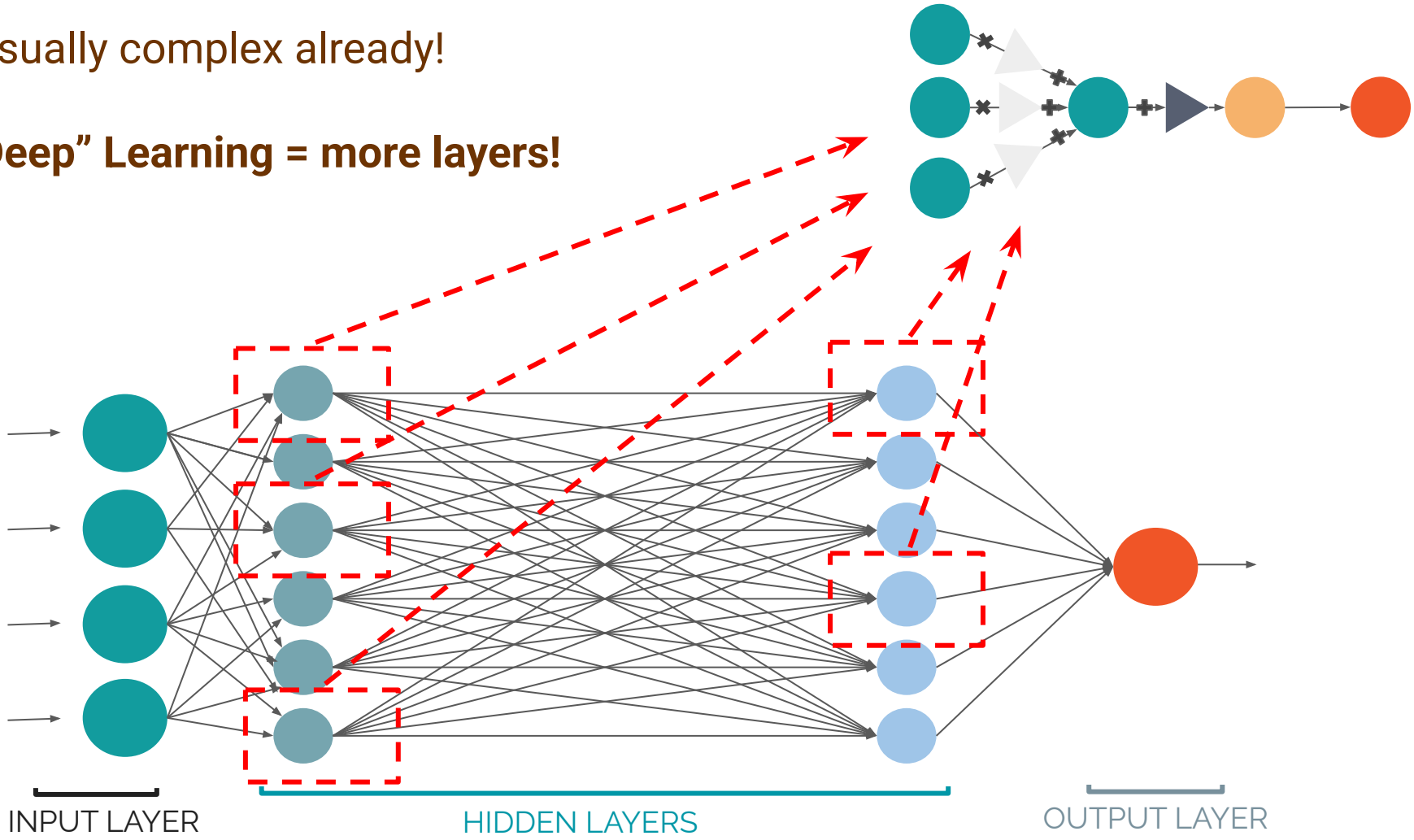
What adjustments (“design choices”) can I make?



NEURAL NETWORK STRUCTURE

Visually complex already!

“Deep” Learning = more layers!

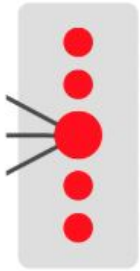


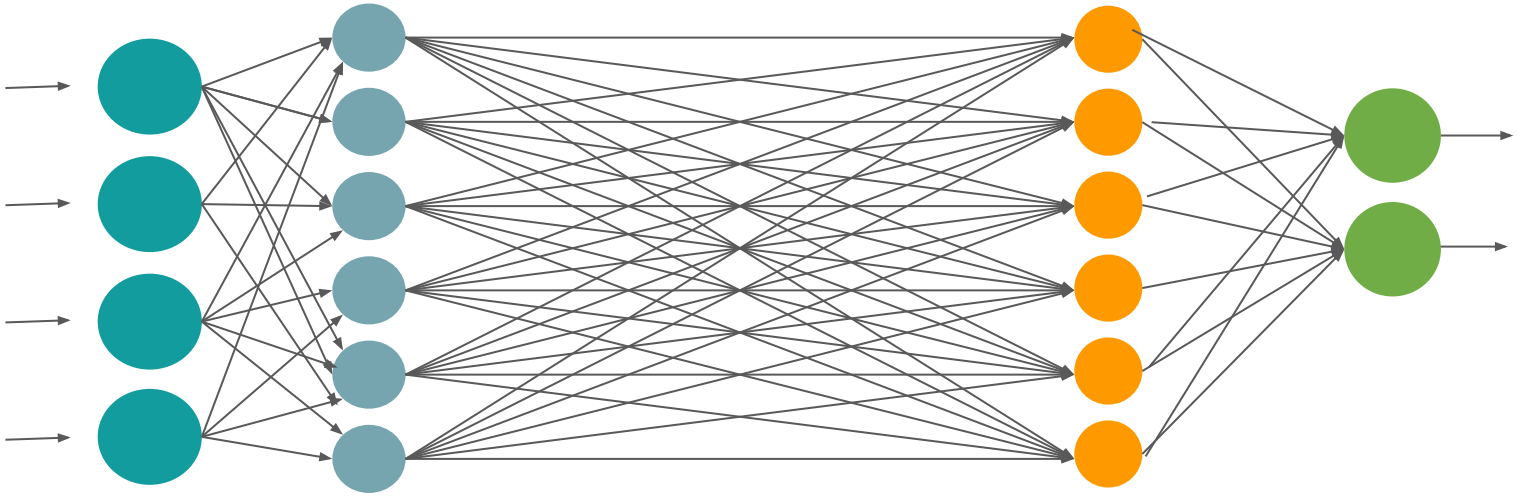
NEURAL NETWORK STRUCTURE

Neural Networks support multiclass classification

Visualizing the larger network…

The softmax layer has one output node for each class



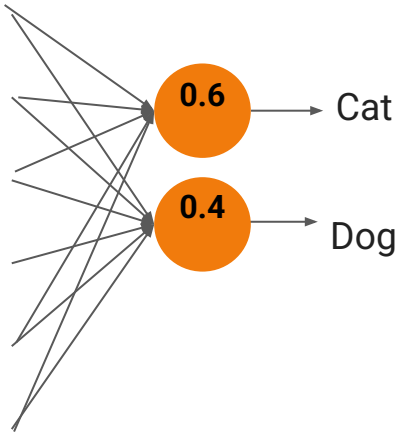


NEURAL NETWORK TRAINING

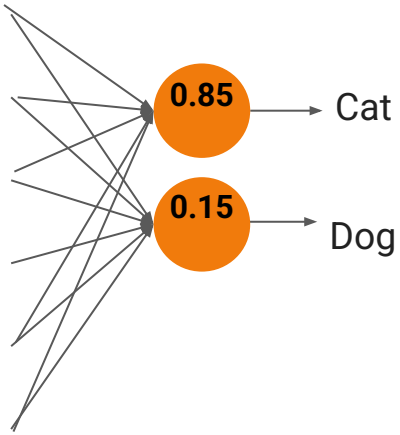
Loss - What is loss? (Think of it as akin to “error”)

We want to have as low a loss (ie. low “error”) as possible!

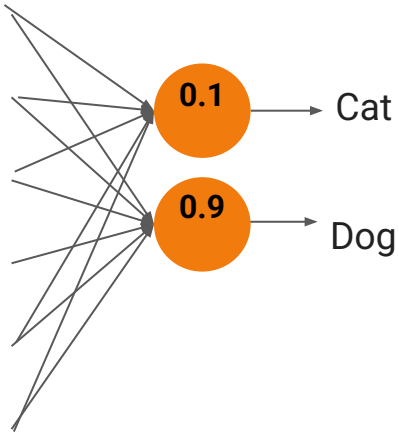
Network 1:

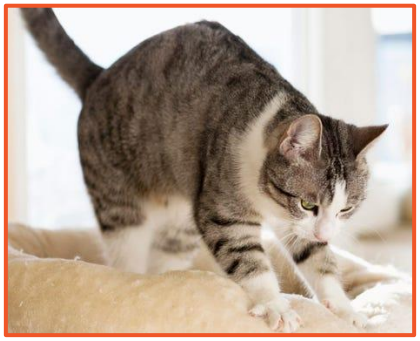


Network 2:



Network 3:





Network 1: **Medium** loss
Network 2: **Low** loss
Network 3: **High** loss