PyTorch and Neural Nets

CS285 Deep RL

Instructor: Marwa Abdulhai

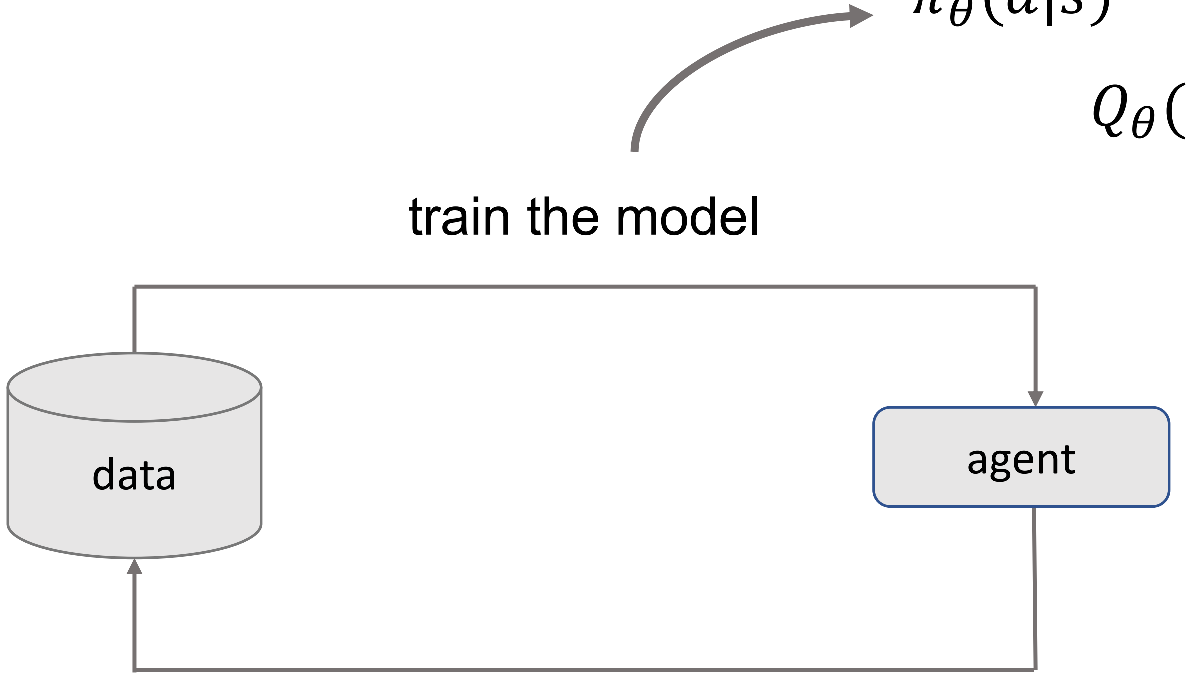
[Adapted from Kevin Li’s CS285 Fa21 Slides]

Goal of this course

Train an agent to perform useful tasks

$\pi_\theta(a|s)$ $\hat{\Delta}_{t+1} = f_\theta(s_t, a_t)$

$Q_\theta(s,a)$



train the model

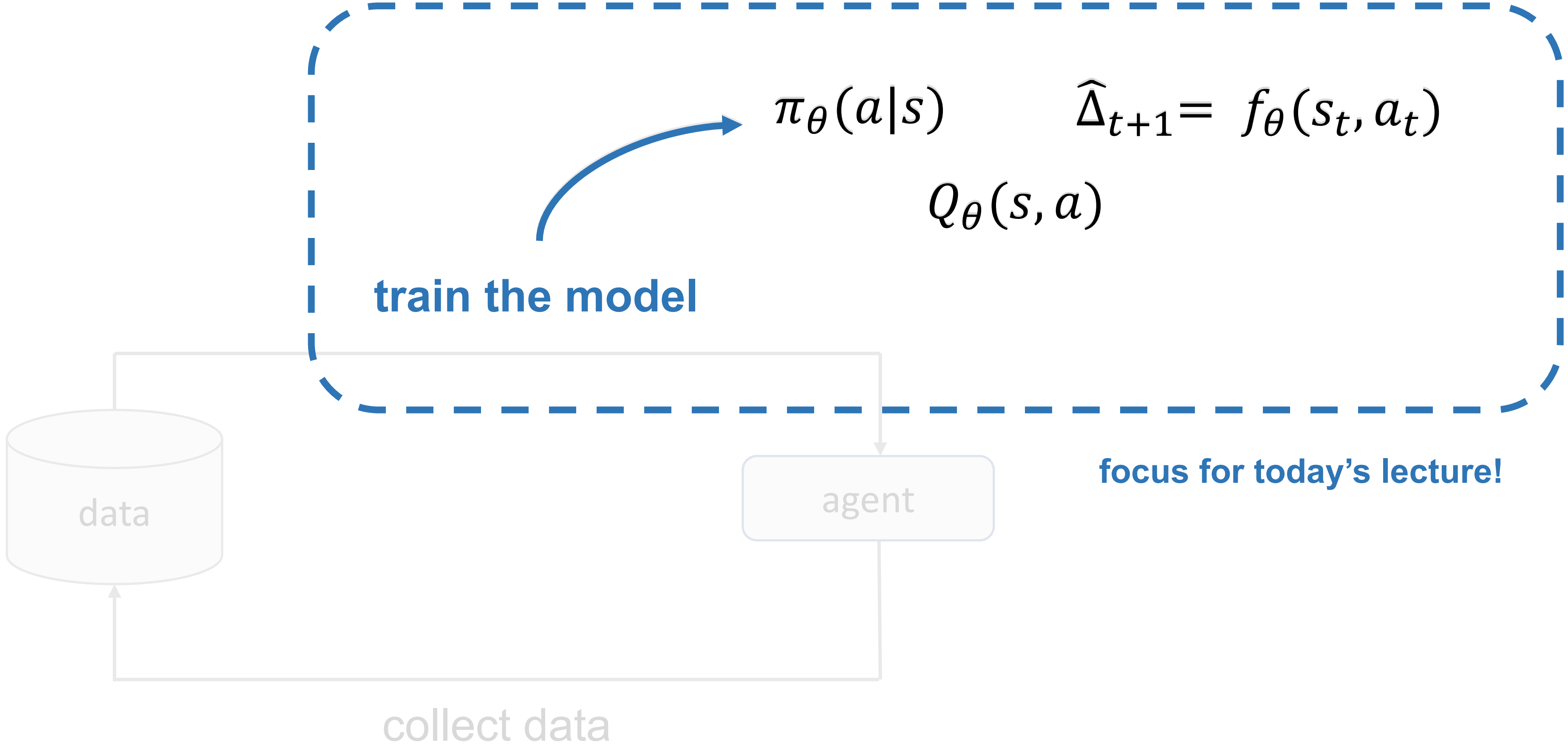
data

agent

collect data

Goal of this course

Train an agent to perform useful tasks



$\pi_\theta(a|s)$ $\hat{\Delta}_{t+1} = f_\theta(s_t, a_t)$

$Q_\theta(s, a)$

train the model

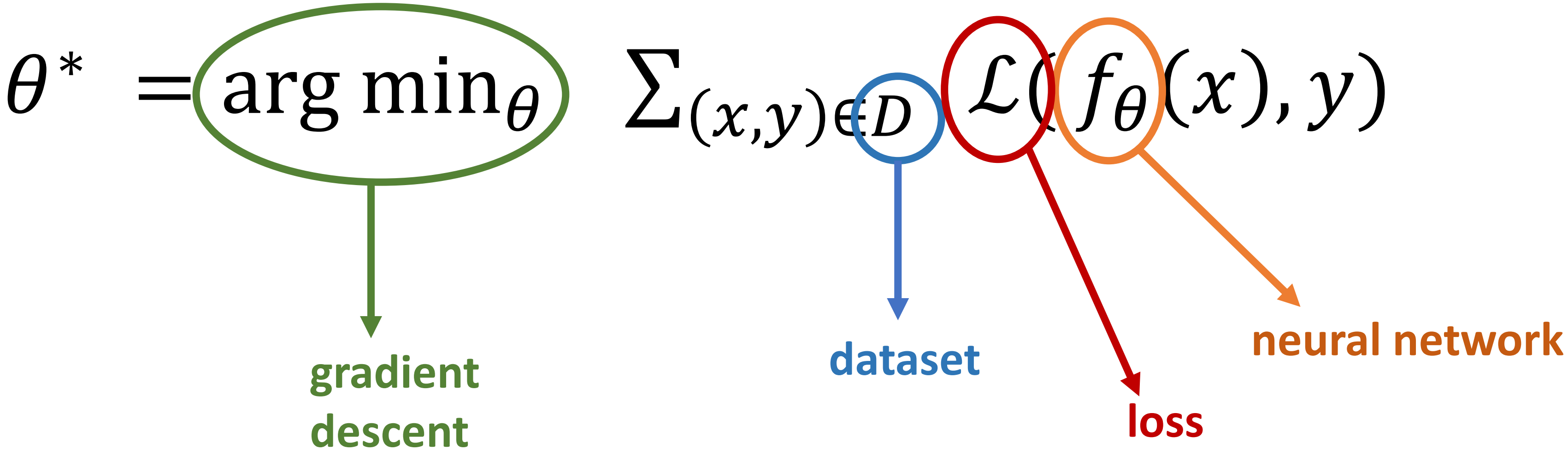
data

agent

collect data

focus for today's lecture!

How do train a model?



$$\theta^* = \arg\min_{\theta} \sum_{(x,y)\in D} \mathcal{L}(f_{\theta}(x), y)$$

gradient descent

dataset

loss

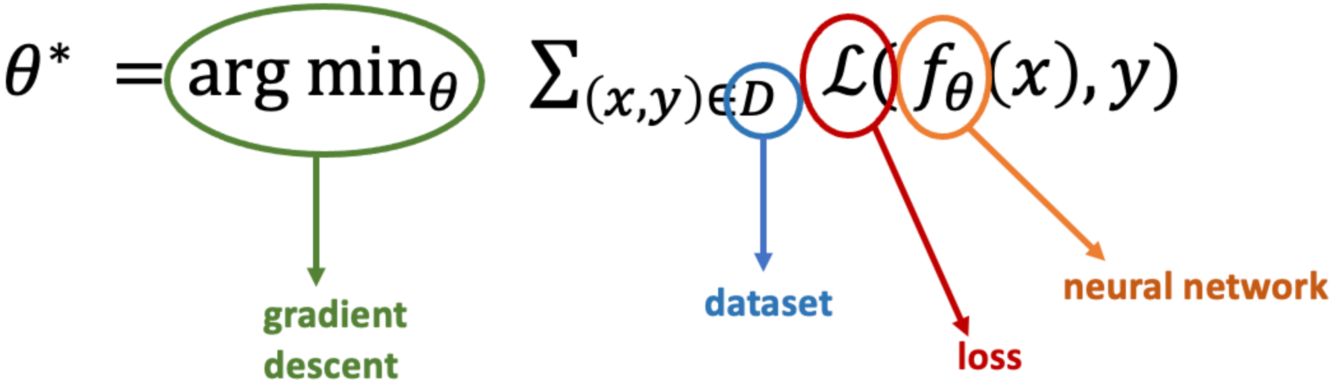
neural network

PyTorch does all of these!

What is PyTorch?

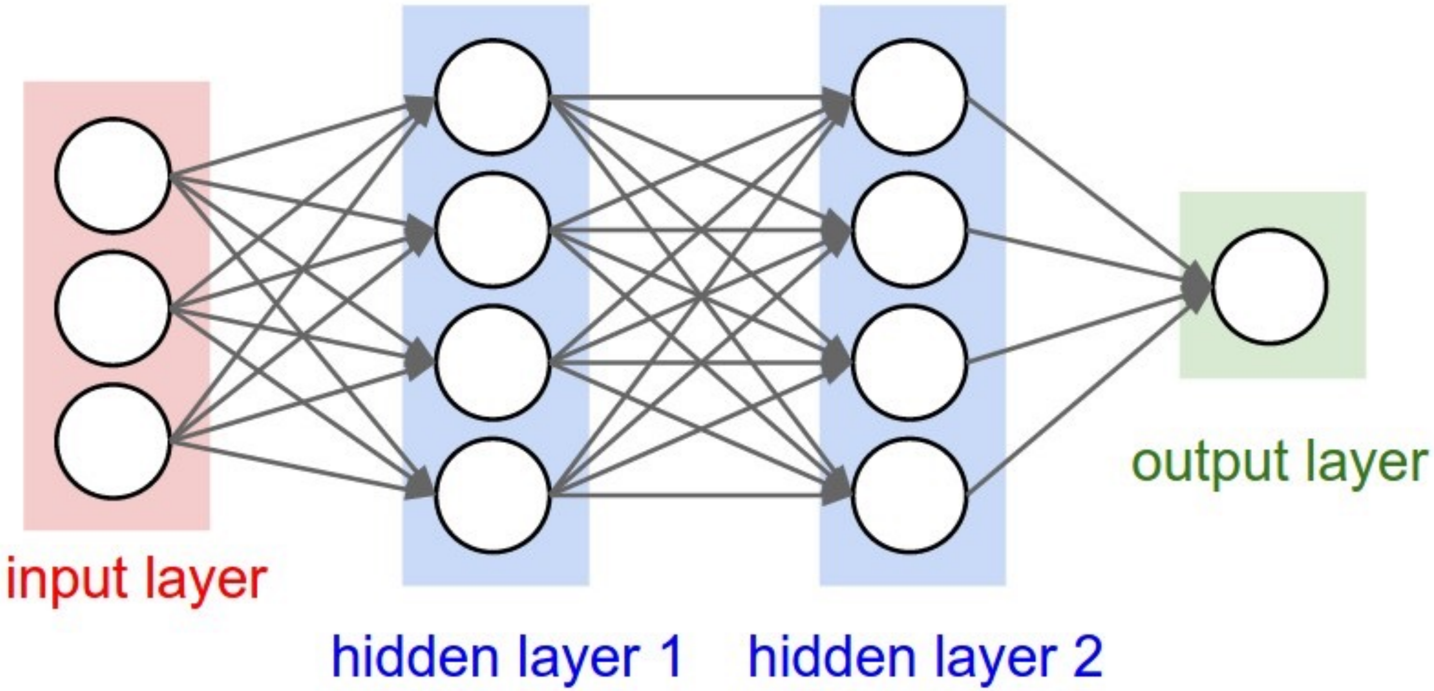
Python library for:

- Defining neural networks
- Automating computing gradients
- And more! (datasets, optimizers, GPUs, etc.)



$$\theta^* = \arg\min_\theta \sum_{(x,y)\in D} \mathcal{L}(f_\theta(x), y)$$

How does PyTorch work?



You define: $h_1 = \sigma(W_1 x)$ $h_2 = \sigma(W_2 h_1)$ $y = \sigma(W_3 h_2)$

PyTorch computes: $\frac{\partial y}{\partial W_1} = \frac{\partial y}{\partial h_2}\frac{\partial h_2}{\partial h_1}\frac{\partial h_1}{\partial W_1}$ $\frac{\partial y}{\partial W_2} = \frac{\partial y}{\partial h_2}\frac{\partial h_2}{\partial W_1}$ $\frac{\partial y}{\partial W_3}$

[picture from Stanford's CS231n]

PyTorch Tutorial (Colab)

https://colab.research.google.com/drive/1XQu1mUbGtvkQY-D7_YCOZlRzSnjp4u9f?usp=sharing

https://bit.ly/3CM6lcf