

# UC Berkeley·CS285 | Deep Reinforcement Learning (2020)

## CS285 (2020)· 课程资料包 @ShowMeAI



视频

中英双语字幕



课件

一键打包下载



笔记

官方笔记翻译



代码

作业项目解析



视频·B 站 [ 扫码或点击链接 ]

<https://www.bilibili.com/video/BV12341167kL>



课件 & 代码·博客 [ 扫码或点击链接 ]

<http://blog.showmeai.tech/cs285>

Berkeley

actor-critic

reinforcement learning

exploration

元学习

exploitation

梯度策略

探索与利用

DQN

迁移学习

逆强化学习

Q-learning / Q 学习

deep q network

meta learning

Awesome AI Courses Notes Cheatsheets 是 [ShowMeAI](#) 资料库的分支系列，覆盖最具知名度的 **TOP20+** 门 AI 课程，旨在为读者和学习者提供一整套高品质中文学习笔记和速查表。

点击课程名称，跳转至课程**资料包**页面，**一键下载**课程全部资料！

机器学习	深度学习	自然语言处理	计算机视觉
Stanford · CS229	Stanford · CS230	Stanford · CS224n	Stanford · CS231n
# Awesome AI Courses Notes Cheatsheets· 持续更新中			
知识图谱	图机器学习	深度强化学习	自动驾驶
Stanford · CS520	Stanford · CS224W	UCBerkeley · CS285	MIT · 6.S094



微信公众号

资料下载方式 2: 扫码点击**底部菜单栏**

称为 **AI 内容创作者**? 回复 [ 添砖加瓦 ]

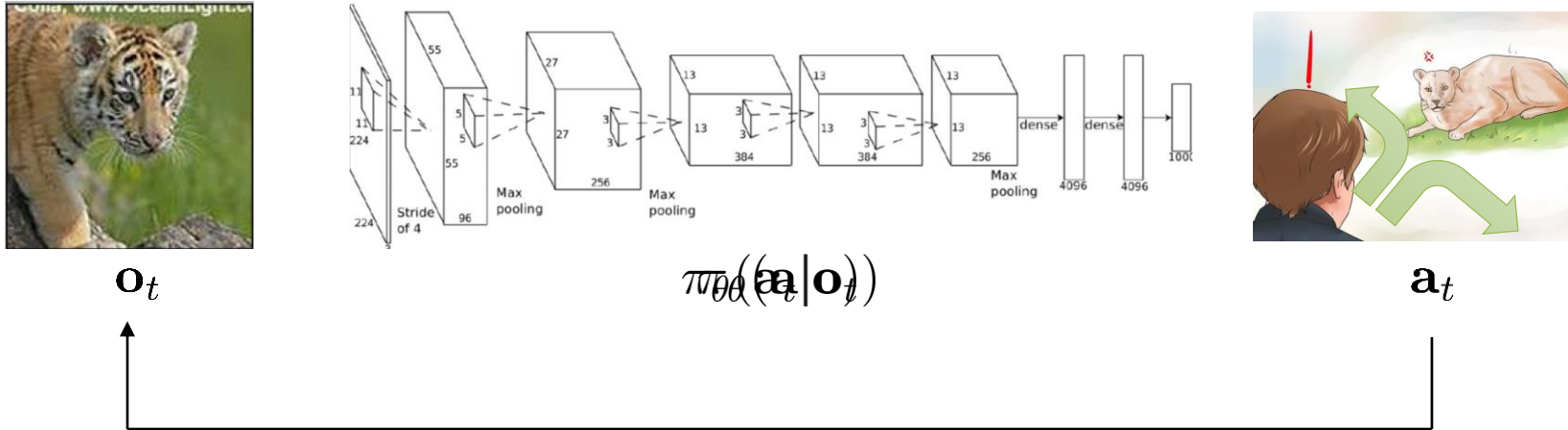
# Supervised Learning of Behaviors

CS 285

Instructor: Sergey Levine  
UC Berkeley



# Terminology & notation



$\mathbf{s}_t$  – state

$\mathbf{o}_t$  – observation

$\mathbf{a}_t$  – action

$\pi_{\theta}(\mathbf{a}_t | \mathbf{o}_t)$  – policy

$\pi_{\theta}(\mathbf{a}_t | \mathbf{s}_t)$  – policy (fully observed)

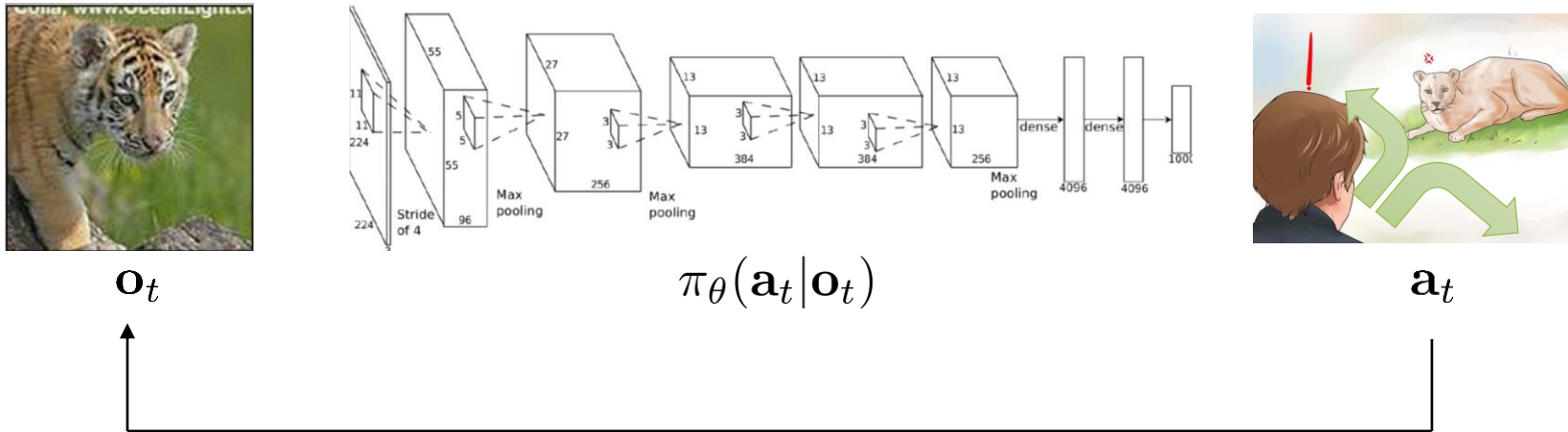


$\mathbf{o}_t$  – observation



$\mathbf{s}_t$  – state

# Terminology & notation



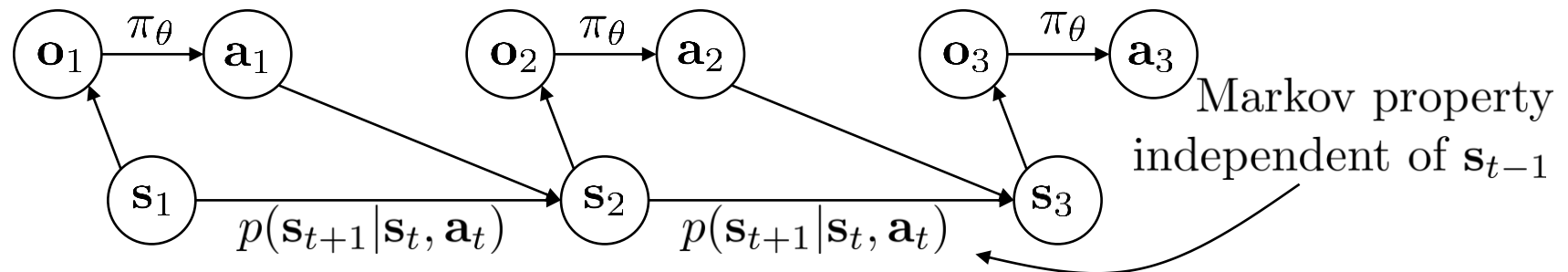
$\mathbf{s}_t$  – state

$\mathbf{o}_t$  – observation

$\mathbf{a}_t$  – action

$\pi_{\theta}(\mathbf{a}_t | \mathbf{o}_t)$  – policy

$\pi_{\theta}(\mathbf{a}_t | \mathbf{s}_t)$  – policy (fully observed)



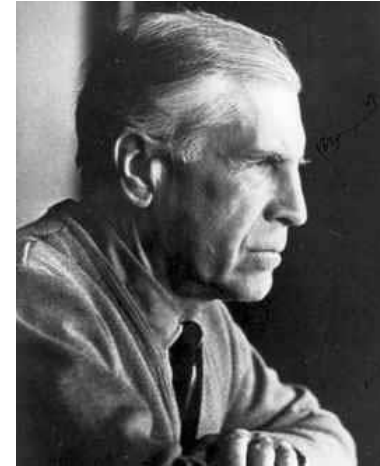
# Aside: notation

$\mathbf{s}_t$  – state  
 $\mathbf{a}_t$  – action



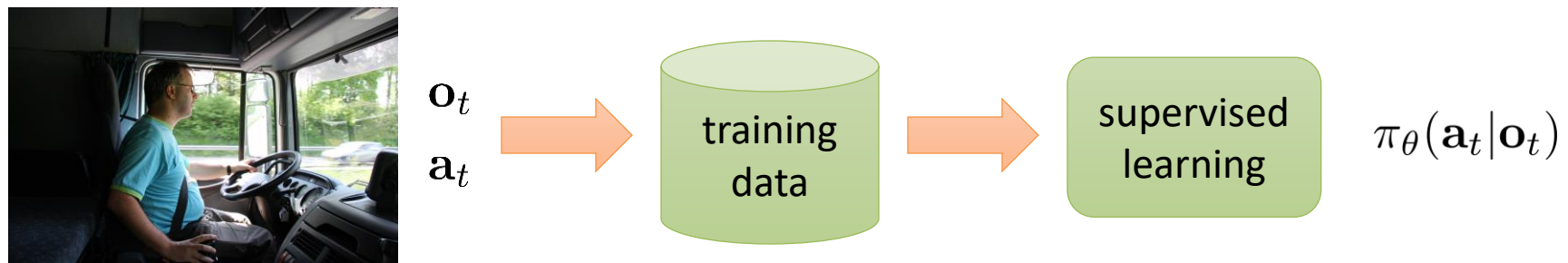
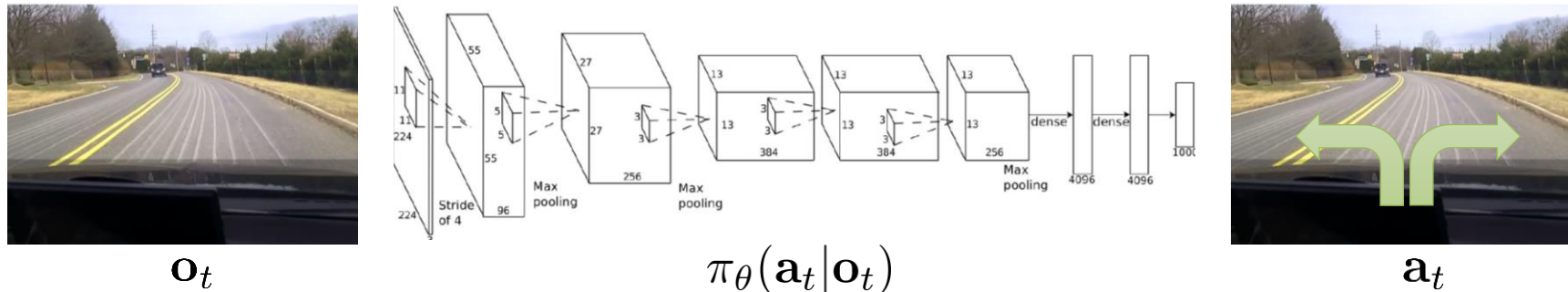
Richard Bellman

$\mathbf{x}_t$  – state  
 $\mathbf{u}_t$  – action     управление



Lev Pontryagin

# Imitation Learning

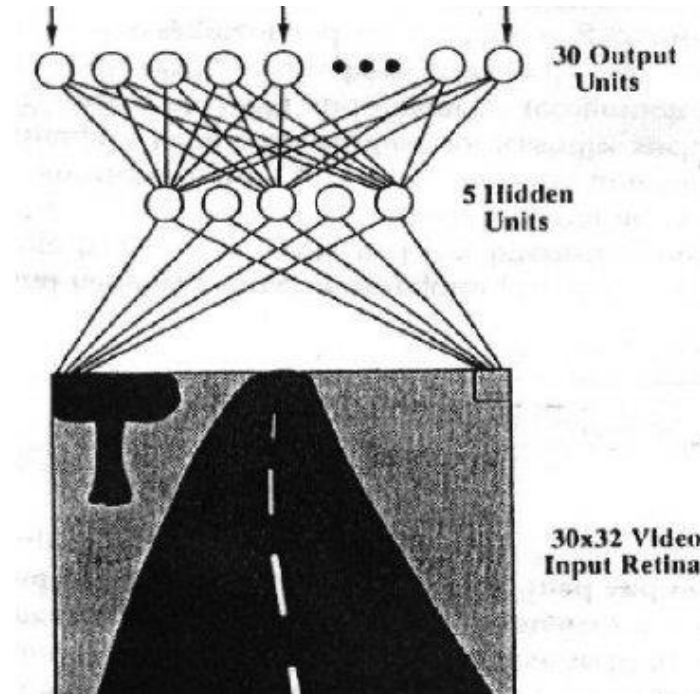


behavioral cloning



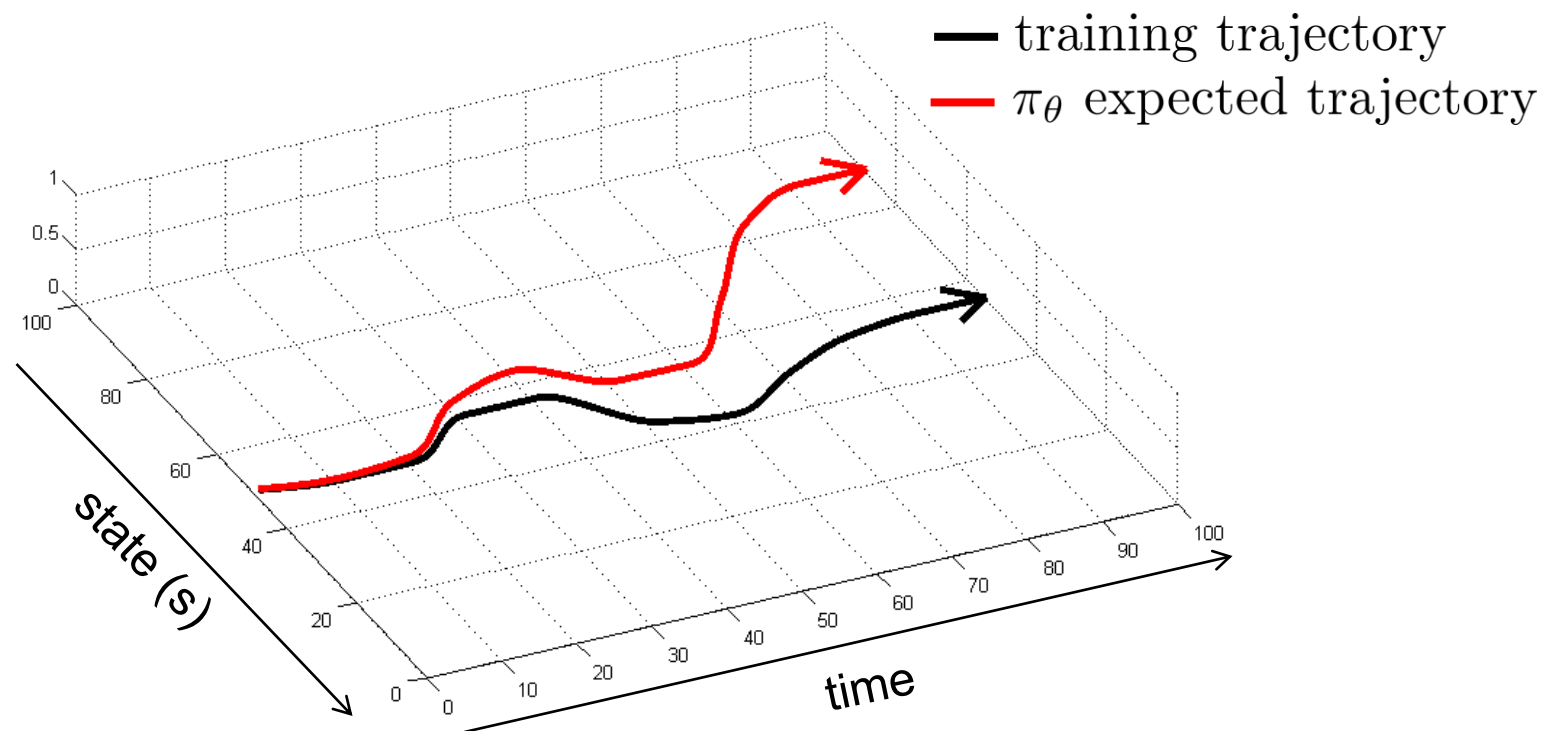
# The original deep imitation learning system

ALVINN: **A**utonomous **L**and **V**ehicle **I**n a **N**eural **N**etwork  
1989



Does it work?

No!



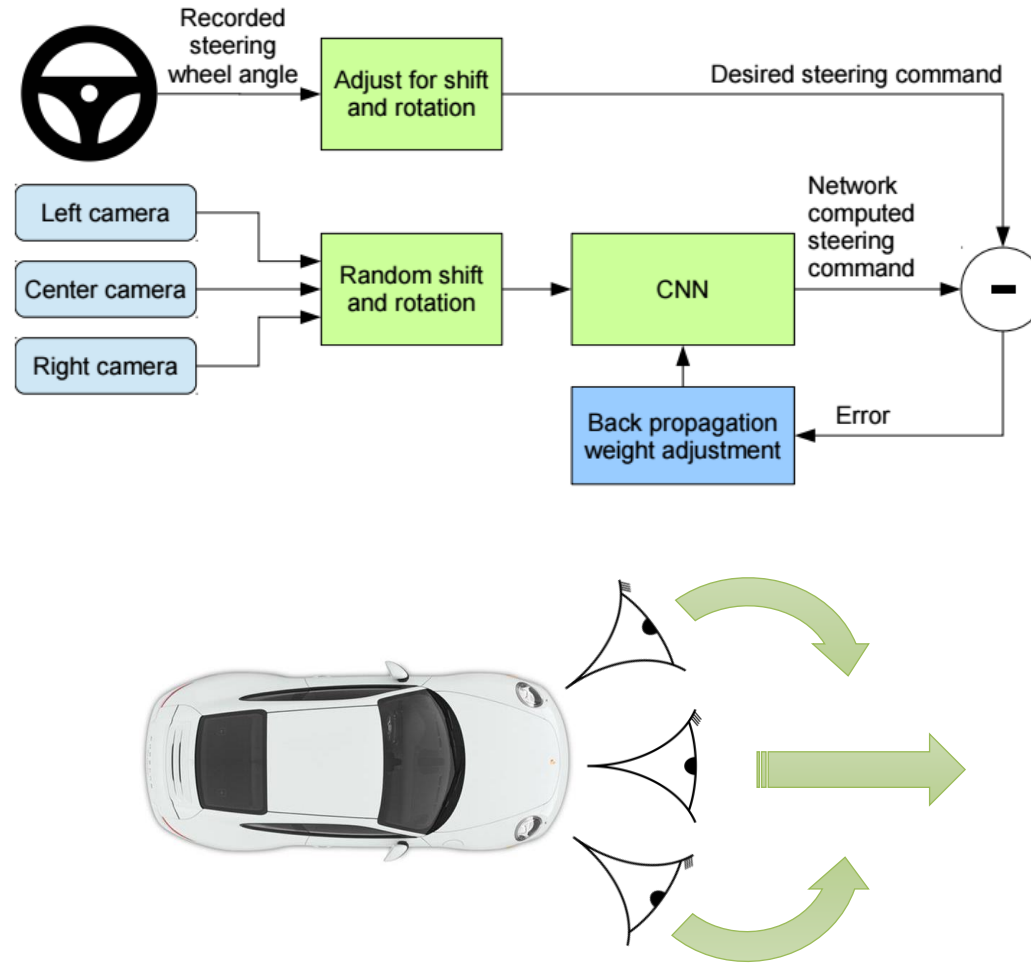


Does it work?

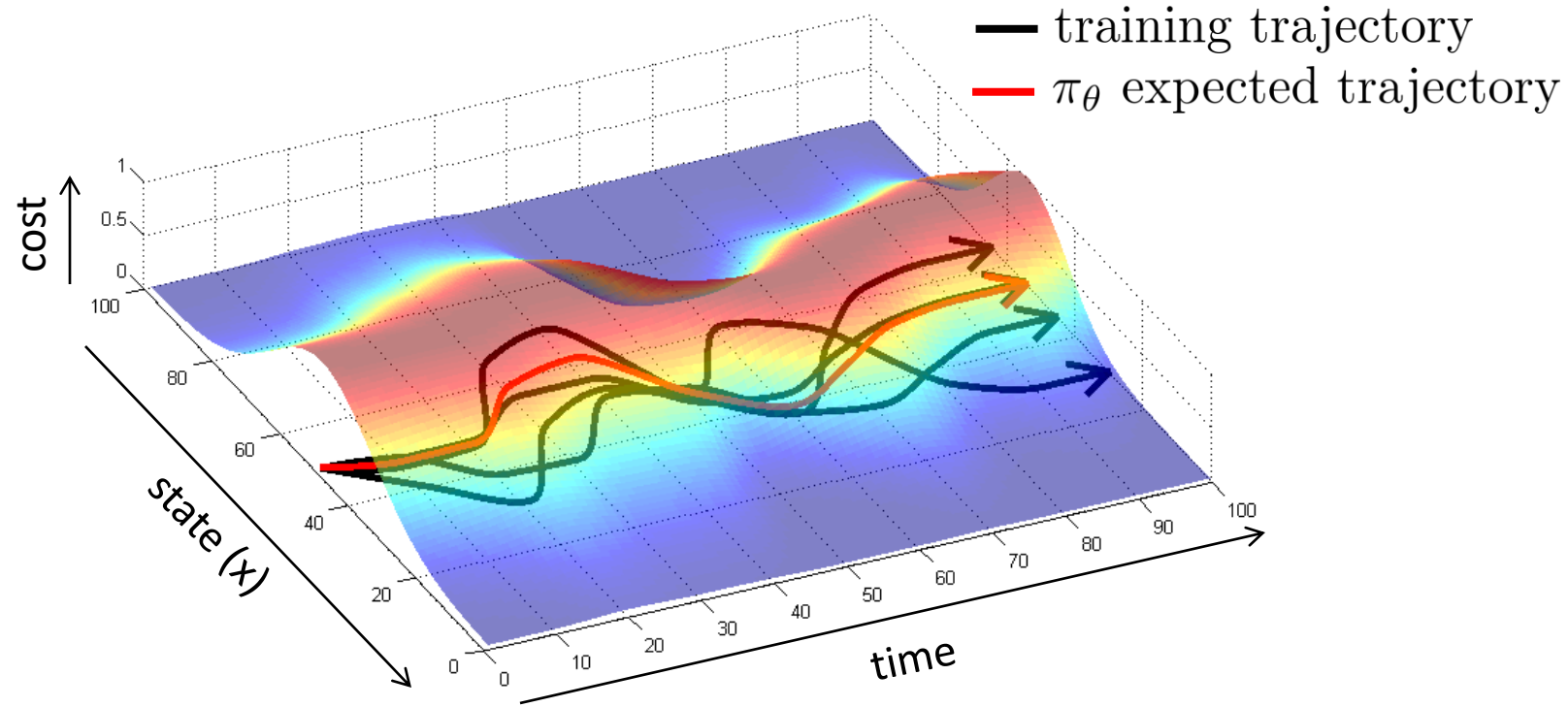
Yes!



# Why did that work?



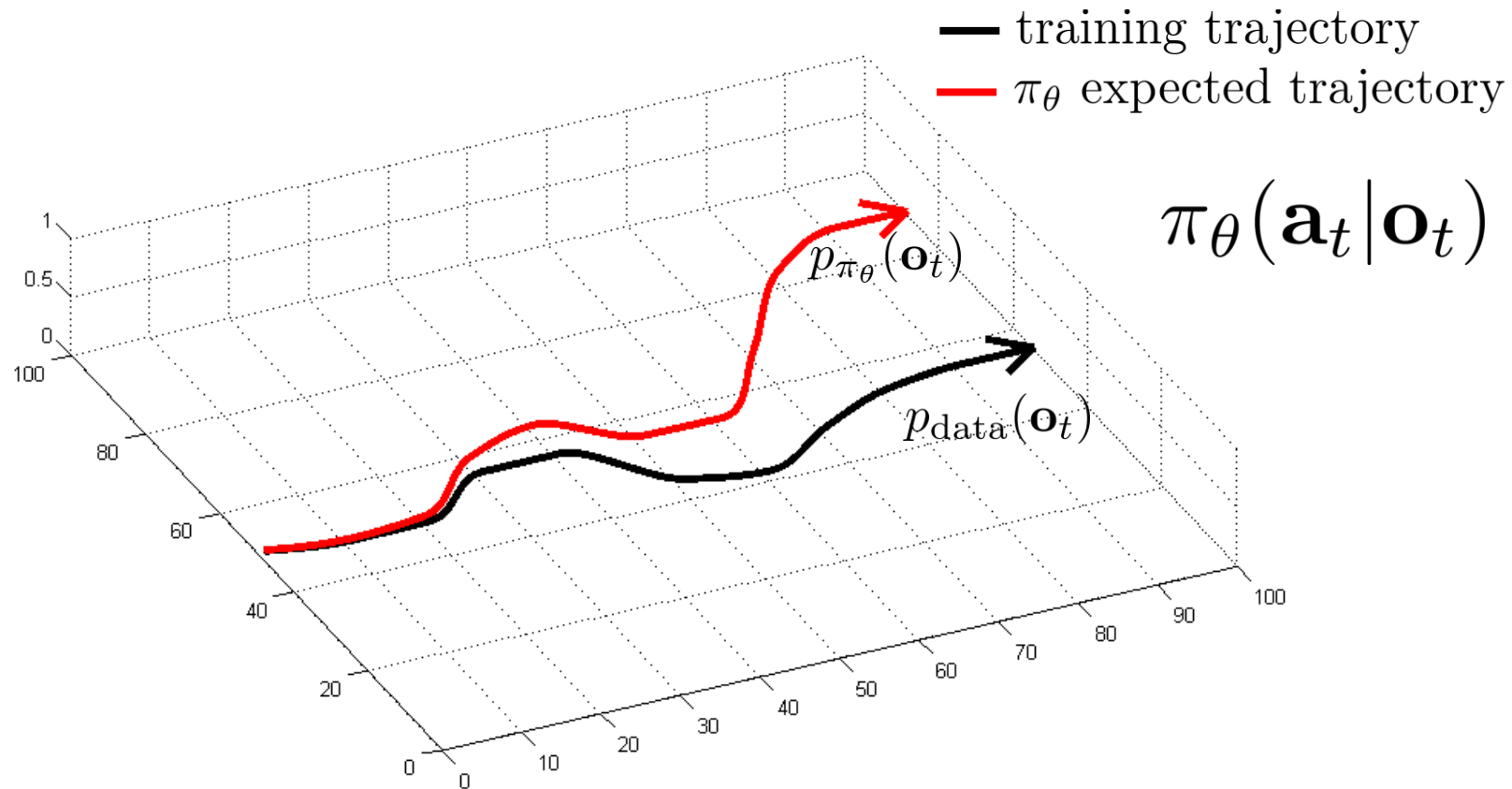
# Can we make it work more often?



stability

(more on this later)

# Can we make it work more often?



can we make  $p_{\text{data}}(\mathbf{o}_t) = p_{\pi_\theta}(\mathbf{o}_t)$ ?

# Can we make it work more often?

can we make  $p_{\text{data}}(\mathbf{o}_t) = p_{\pi_\theta}(\mathbf{o}_t)$ ?

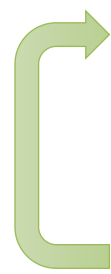
idea: instead of being clever about  $p_{\pi_\theta}(\mathbf{o}_t)$ , be clever about  $p_{\text{data}}(\mathbf{o}_t)$ !

## **D**Agger: Dataset Aggregation

goal: collect training data from  $p_{\pi_\theta}(\mathbf{o}_t)$  instead of  $p_{\text{data}}(\mathbf{o}_t)$

how? just run  $\pi_\theta(\mathbf{a}_t|\mathbf{o}_t)$

but need labels  $\mathbf{a}_t$ !

- 
1. train  $\pi_\theta(\mathbf{a}_t|\mathbf{o}_t)$  from human data  $\mathcal{D} = \{\mathbf{o}_1, \mathbf{a}_1, \dots, \mathbf{o}_N, \mathbf{a}_N\}$
  2. run  $\pi_\theta(\mathbf{a}_t|\mathbf{o}_t)$  to get dataset  $\mathcal{D}_\pi = \{\mathbf{o}_1, \dots, \mathbf{o}_M\}$
  3. Ask human to label  $\mathcal{D}_\pi$  with actions  $\mathbf{a}_t$
  4. Aggregate:  $\mathcal{D} \leftarrow \mathcal{D} \cup \mathcal{D}_\pi$

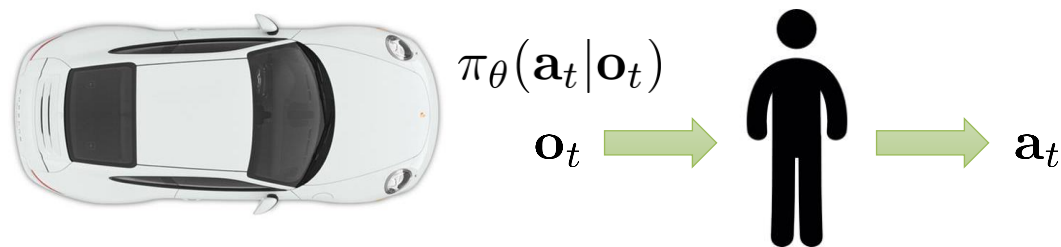
# Dagger Example





# What's the problem?

1. train  $\pi_{\theta}(\mathbf{a}_t|\mathbf{o}_t)$  from human data  $\mathcal{D} = \{\mathbf{o}_1, \mathbf{a}_1, \dots, \mathbf{o}_N, \mathbf{a}_N\}$
2. run  $\pi_{\theta}(\mathbf{a}_t|\mathbf{o}_t)$  to get dataset  $\mathcal{D}_{\pi} = \{\mathbf{o}_1, \dots, \mathbf{o}_M\}$
3. Ask human to label  $\mathcal{D}_{\pi}$  with actions  $\mathbf{a}_t$
4. Aggregate:  $\mathcal{D} \leftarrow \mathcal{D} \cup \mathcal{D}_{\pi}$

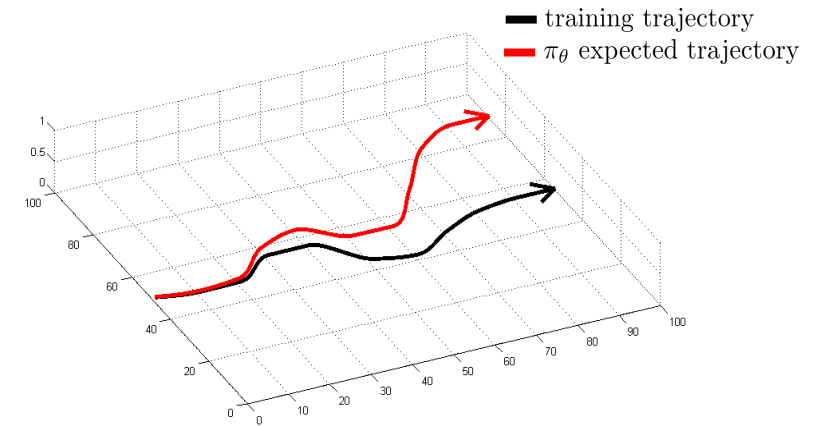




Deep imitation learning in practice

# Can we make it work without more data?

- DAgger addresses the problem of distributional “drift”
- What if our model is so good that it doesn’t drift?
- Need to mimic expert behavior very accurately
- But don’t overfit!



# Why might we fail to fit the expert?

- 
1. Non-Markovian behavior
  2. Multimodal behavior

$$\pi_{\theta}(\mathbf{a}_t | \mathbf{o}_t)$$

behavior depends only  
on current observation

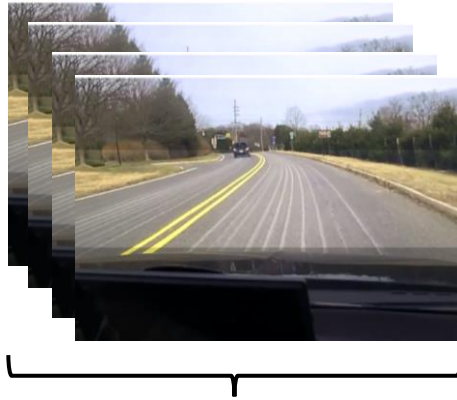
$$\pi_{\theta}(\mathbf{a}_t | \mathbf{o}_1, \dots, \mathbf{o}_t)$$

behavior depends on  
all past observations

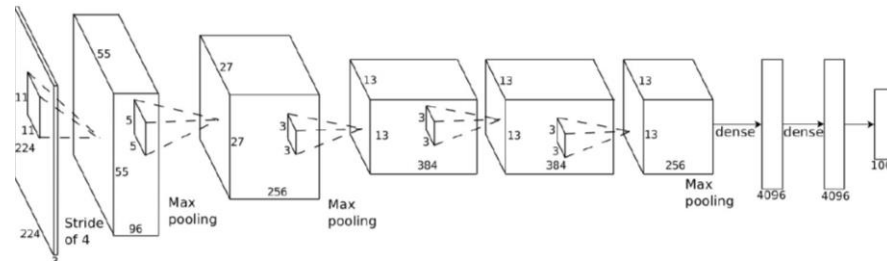
If we see the same thing  
twice, we do the same thing  
twice, regardless of what  
happened before

Often very unnatural for  
human demonstrators

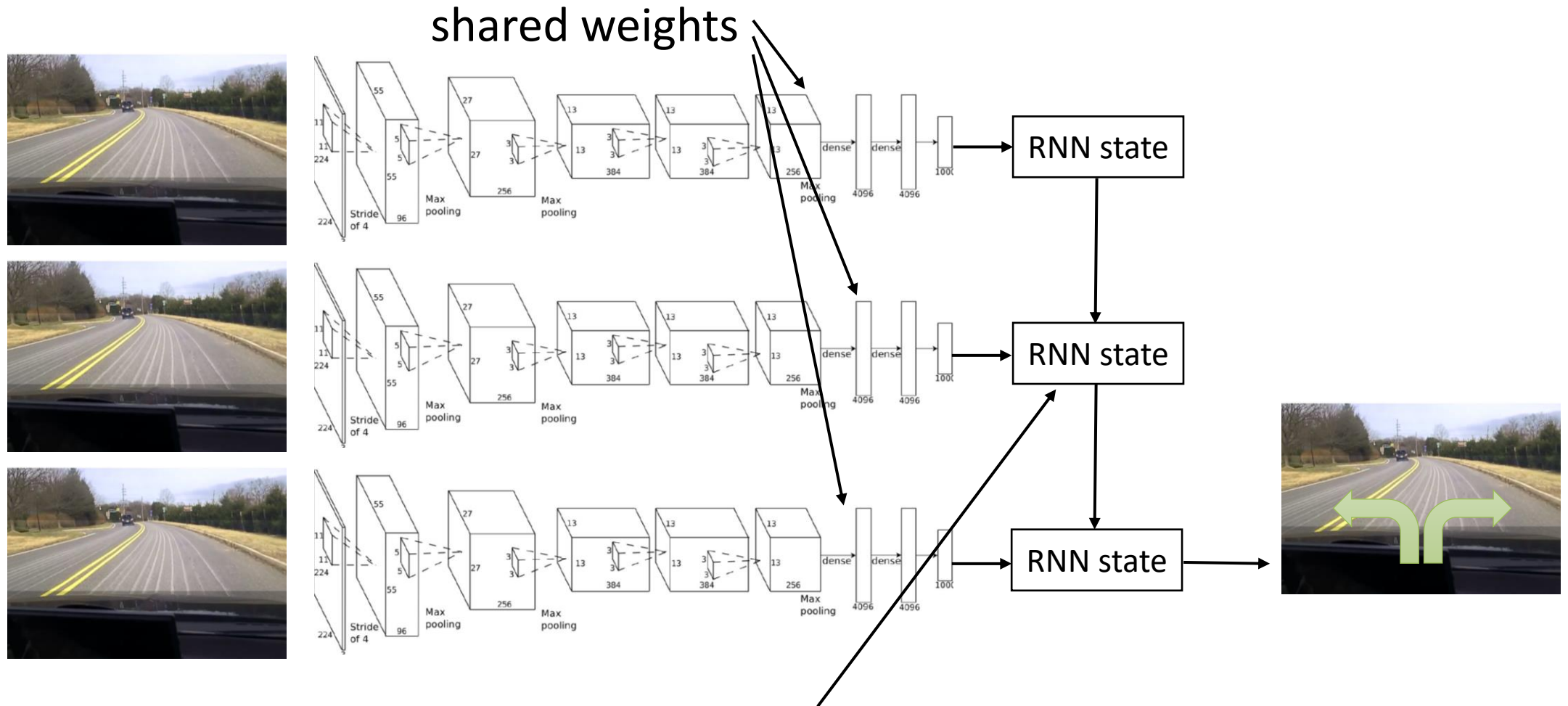
# How can we use the whole history?



variable number of frames,  
too many weights

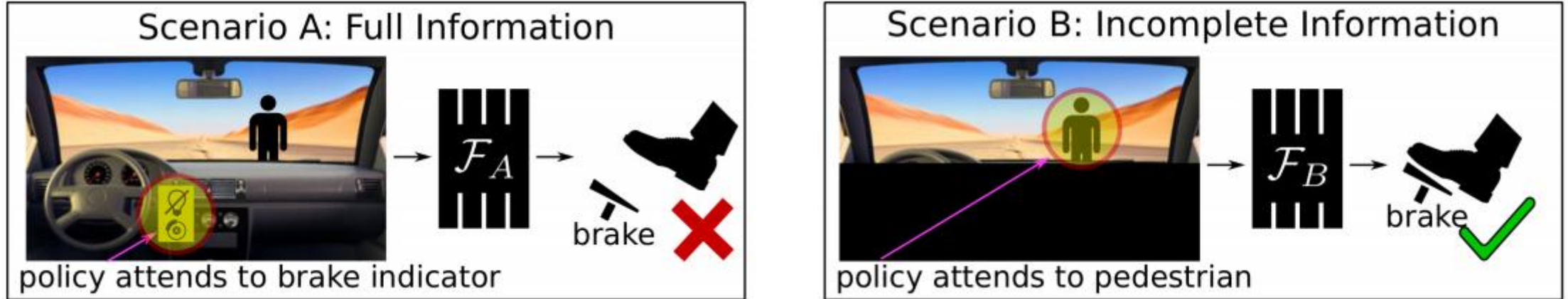


# How can we use the whole history?



Typically, LSTM cells work better here

# Aside: why might this work **poorly**?



“causal confusion”

see: de Haan et al., “Causal Confusion in Imitation Learning”

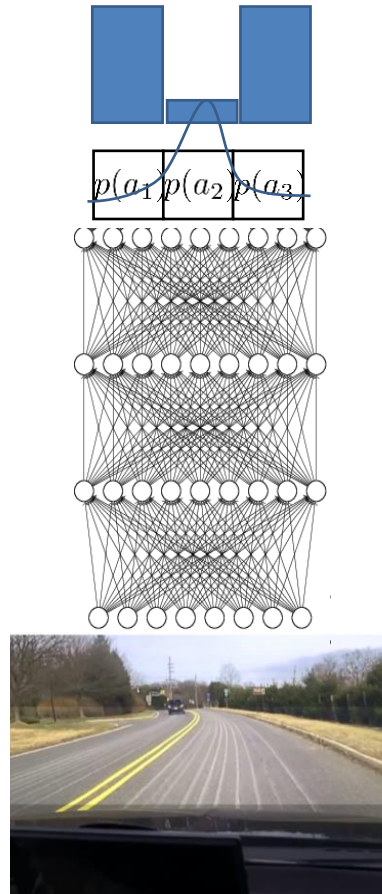
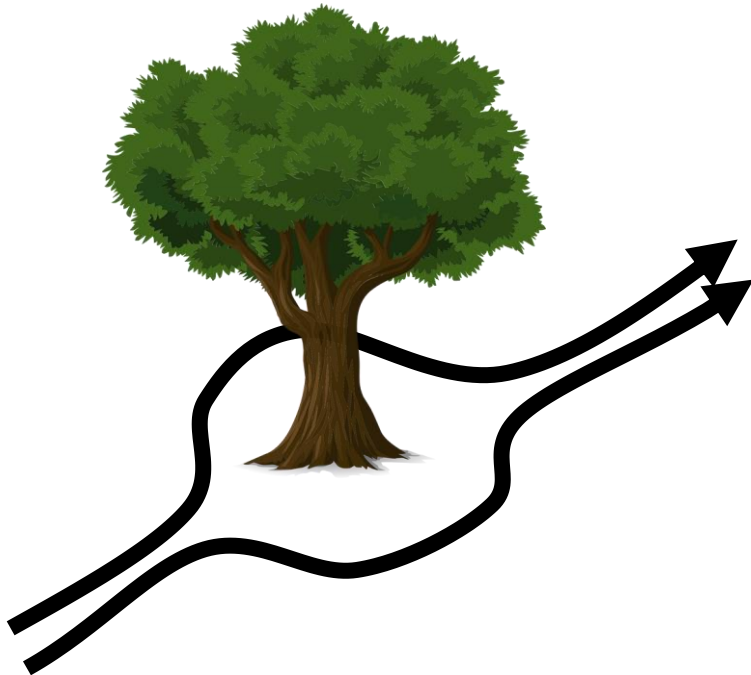
**Question 1:** Does including history mitigate causal confusion?

**Question 2:** Can DAgger mitigate causal confusion?

# Why might we fail to fit the expert?

1. Non-Markovian behavior

➔ 2. Multimodal behavior



1. Output mixture of Gaussians

2. Latent variable models

3. Autoregressive discretization

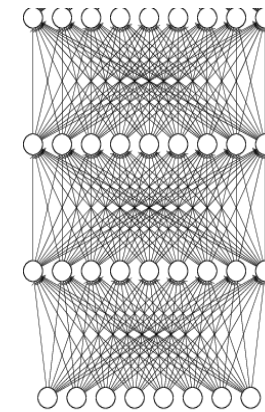
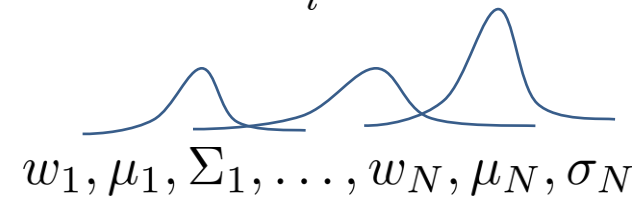




# Why might we fail to fit the expert?

- ➔ 1. Output mixture of Gaussians
- 2. Latent variable models
- 3. Autoregressive discretization

$$\pi(\mathbf{a}|\mathbf{o}) = \sum_i w_i \mathcal{N}(\mu_i, \Sigma_i)$$



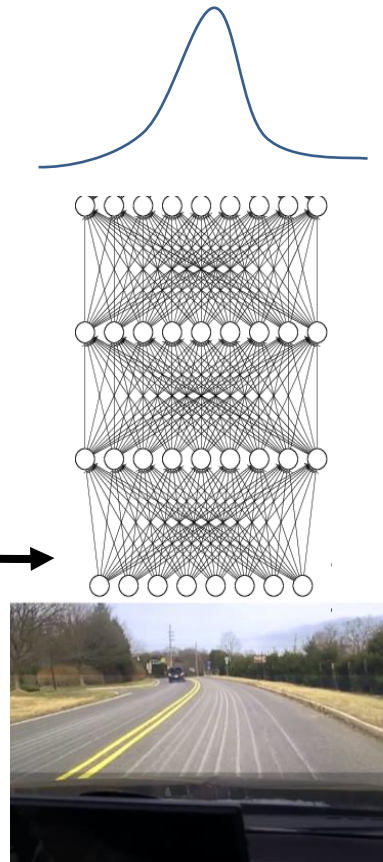
# Why might we fail to fit the expert?

1. Output mixture of Gaussians
- ➔ 2. Latent variable models
3. Autoregressive discretization

Look up some of these:

- Conditional variational autoencoder
- Normalizing flow/realNVP
- Stein variational gradient descent

$$\xi \sim \mathcal{N}(0, \mathbf{I})$$

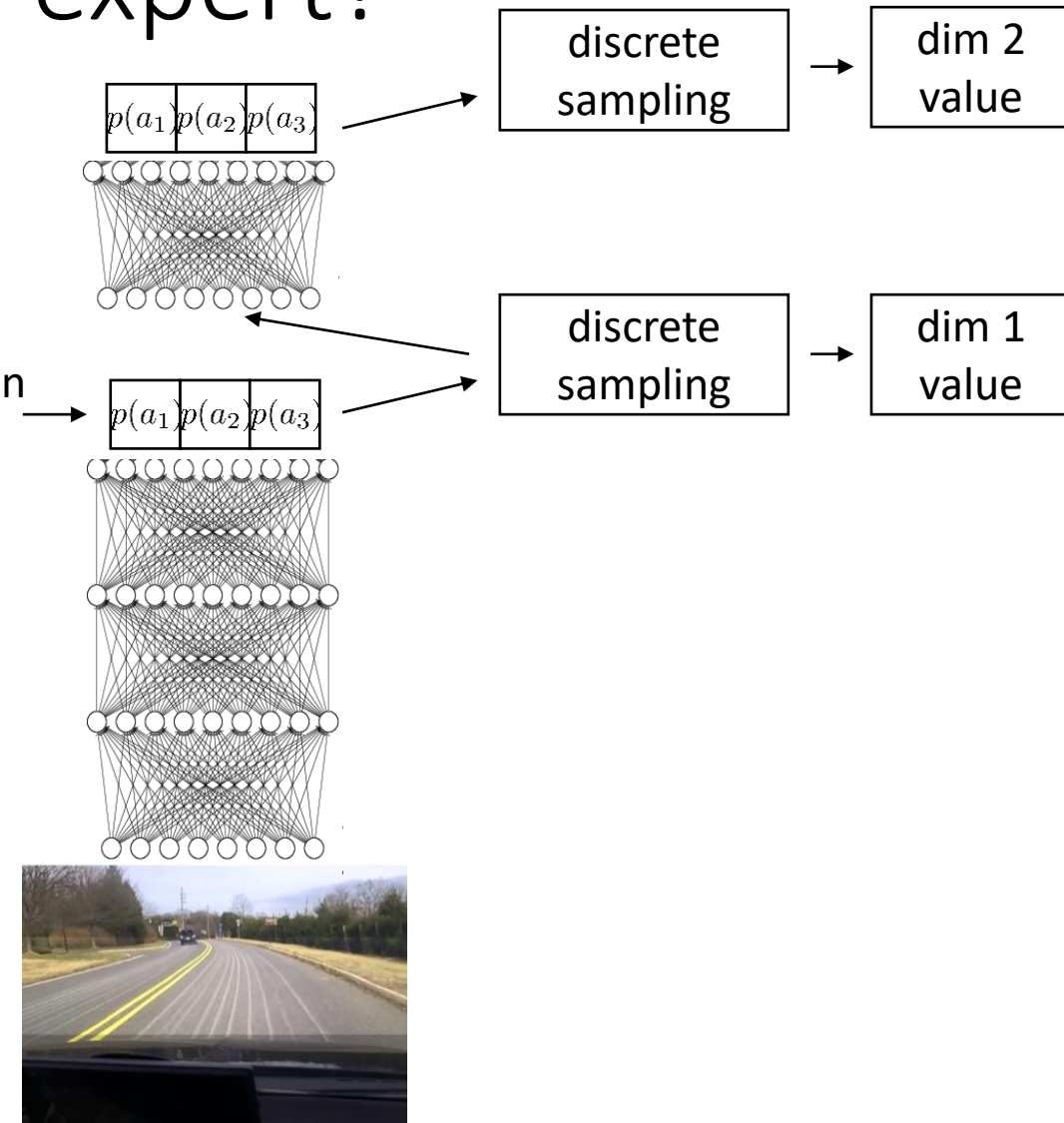


# Why might we fail to fit the expert?

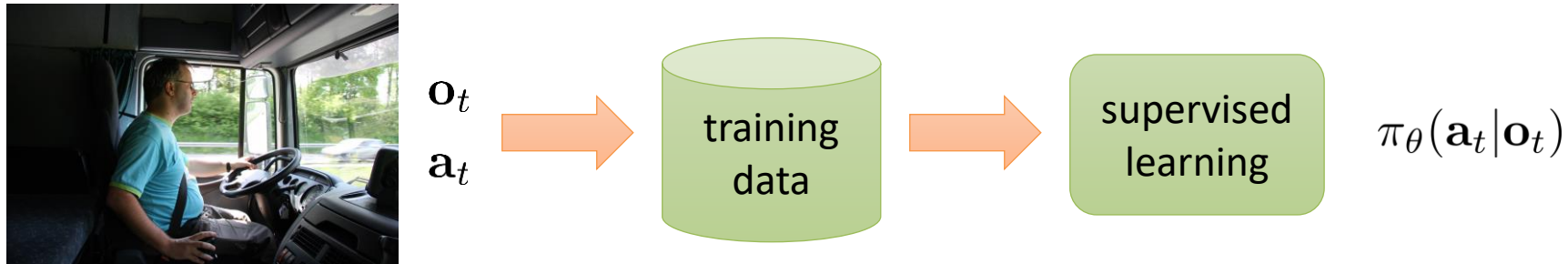
1. Output mixture of Gaussians

2. Latent variable models (discretized) distribution over dimension 1 **only**

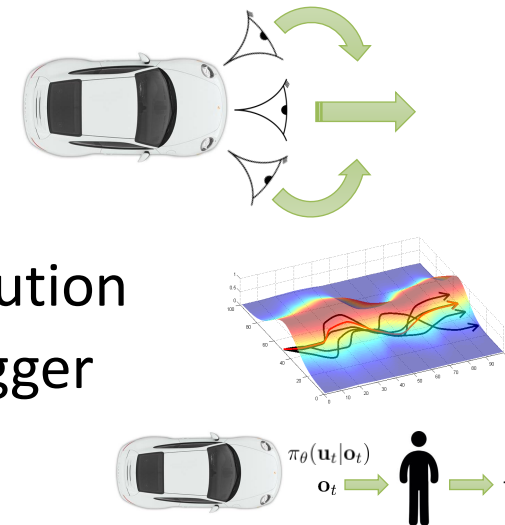
➡ 3. Autoregressive discretization



# Imitation learning: recap



- Often (but not always) insufficient by itself
  - Distribution mismatch problem
- Sometimes works well
  - Hacks (e.g. left/right images)
  - Samples from a stable trajectory distribution
  - Add more **on-policy** data, e.g. using Dagger
  - Better models that fit more accurately

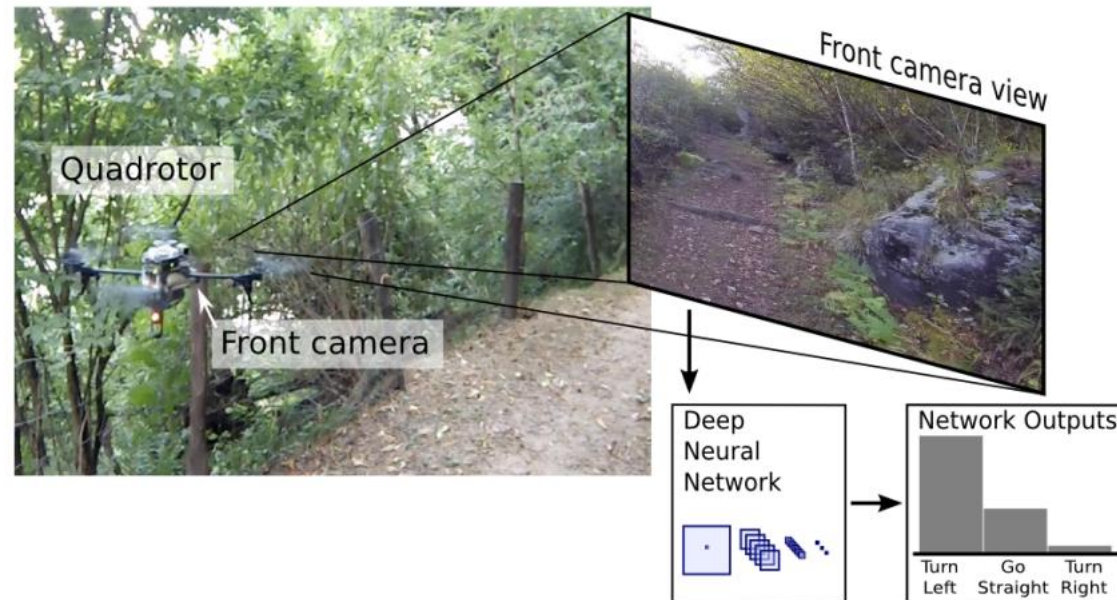


A case study: trail following from  
human demonstration data

# Case study 1: trail following as classification

## A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots

Alessandro Giusti<sup>1</sup>, Jérôme Guzzi<sup>1</sup>, Dan C. Cireşan<sup>1</sup>, Fang-Lin He<sup>1</sup>, Juan P. Rodríguez<sup>1</sup>  
Flavio Fontana<sup>2</sup>, Matthias Faessler<sup>2</sup>, Christian Forster<sup>2</sup>  
Jürgen Schmidhuber<sup>1</sup>, Gianni Di Caro<sup>1</sup>, Davide Scaramuzza<sup>2</sup>, Luca M. Gambardella<sup>1</sup>



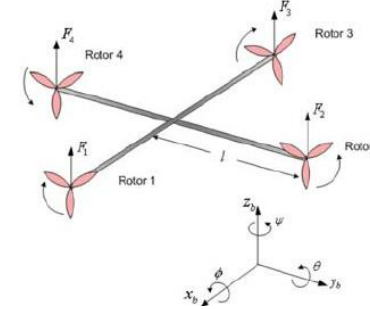
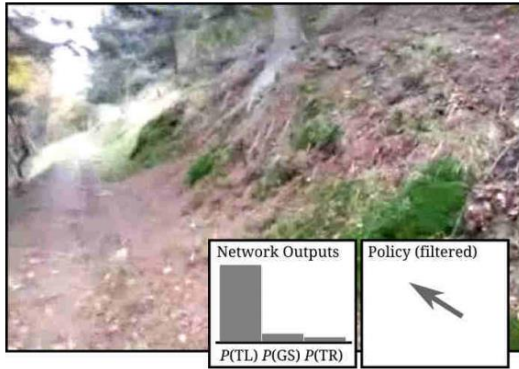




Cost functions, reward functions, and a  
bit of theory

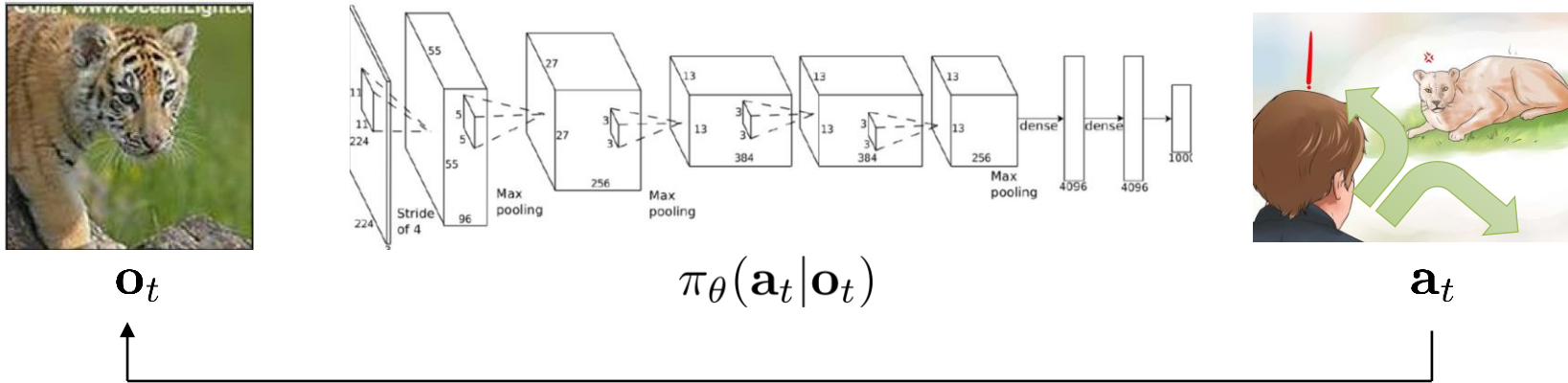
# Imitation learning: what's the problem?

- Humans need to provide data, which is typically finite
  - Deep learning works best when data is plentiful
- Humans are not good at providing some kinds of actions



- Humans can learn autonomously; can our machines do the same?
  - Unlimited data from own experience
  - Continuous self-improvement

# Terminology & notation



$\mathbf{s}_t$  – state

$\mathbf{o}_t$  – observation

$\mathbf{a}_t$  – action

$c(\mathbf{s}_t, \mathbf{a}_t)$  – cost function

$r(\mathbf{s}_t, \mathbf{a}_t)$  – reward function

$$\min_{\theta} E_{\mathbf{s} \sim p(\mathbf{s}), \mathbf{a} \sim \pi_{\theta}(\mathbf{a} | \mathbf{s})} \left[ \sum_t \delta(\mathbf{s}_t, \mathbf{a}_t) \right]$$

# Aside: notation

$\mathbf{s}_t$  – state

$\mathbf{a}_t$  – action

$r(\mathbf{s}, \mathbf{a})$  – reward function



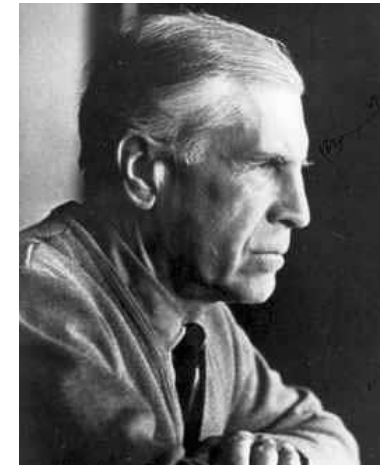
Richard Bellman

$\mathbf{x}_t$  – state

$\mathbf{u}_t$  – action

$c(\mathbf{x}, \mathbf{u})$  – cost function

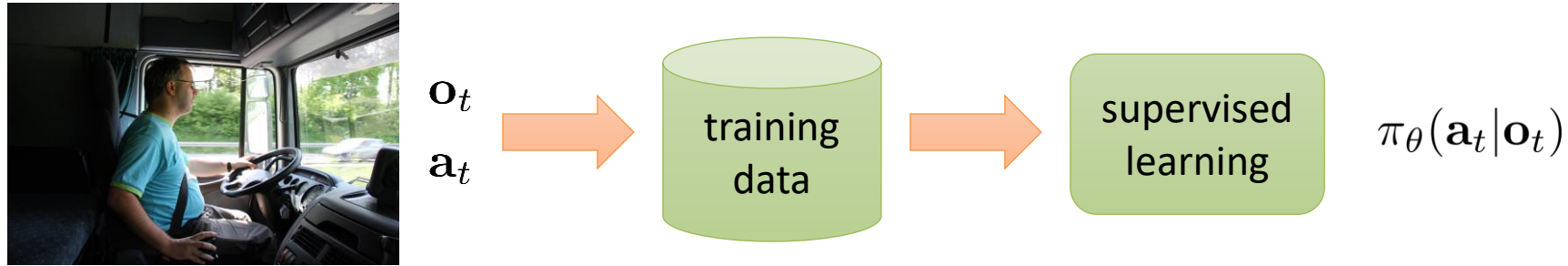
$$r(\mathbf{s}, \mathbf{a}) = -c(\mathbf{x}, \mathbf{u})$$



Lev Pontryagin

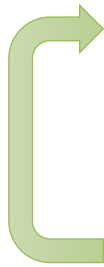
Cost functions, reward functions, and a  
bit of theory

# A cost function for imitation?

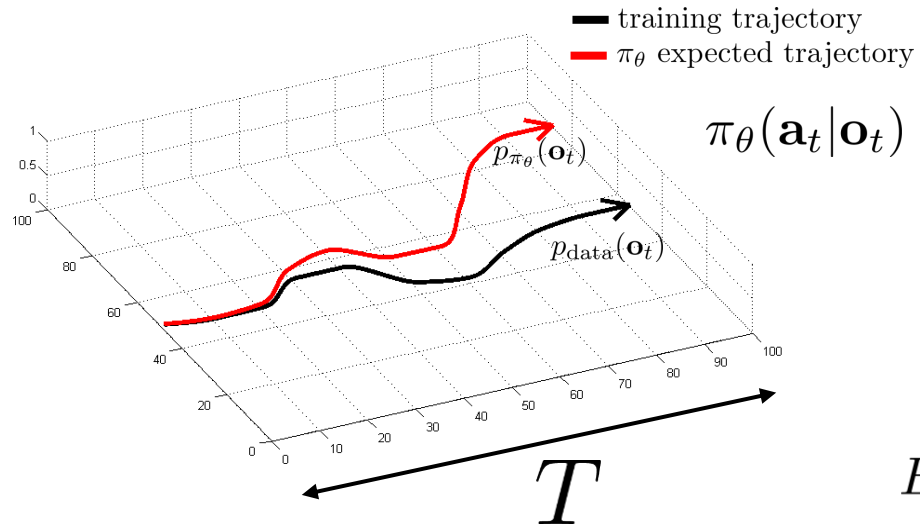


$$r(\mathbf{s}, \mathbf{a}) = \log p(\mathbf{a} = \pi^*(\mathbf{s}) | \mathbf{s})$$

$$c(\mathbf{s}, \mathbf{a}) = \begin{cases} 0 & \text{if } \mathbf{a} = \pi^*(\mathbf{s}) \\ 1 & \text{otherwise} \end{cases}$$

- 
1. train  $\pi_{\theta}(\mathbf{a}_t | \mathbf{o}_t)$  from human data  $\mathcal{D} = \{\mathbf{o}_1, \mathbf{a}_1, \dots, \mathbf{o}_N, \mathbf{a}_N\}$
  2. run  $\pi_{\theta}(\mathbf{a}_t | \mathbf{o}_t)$  to get dataset  $\mathcal{D}_{\pi} = \{\mathbf{o}_1, \dots, \mathbf{o}_M\}$
  3. Ask human to label  $\mathcal{D}_{\pi}$  with actions  $\mathbf{a}_t$
  4. Aggregate:  $\mathcal{D} \leftarrow \mathcal{D} \cup \mathcal{D}_{\pi}$

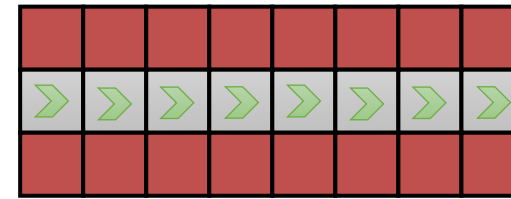
# Some analysis



$$c(\mathbf{s}, \mathbf{a}) = \begin{cases} 0 & \text{if } \mathbf{a} = \pi^*(\mathbf{s}) \\ 1 & \text{otherwise} \end{cases}$$

assume:  $\pi_\theta(\mathbf{a} \neq \pi^*(\mathbf{s}) | \mathbf{s}) \leq \epsilon$

for all  $\mathbf{s} \in \mathcal{D}_{\text{train}}$



$$E \left[ \sum_t c(\mathbf{s}_t, \mathbf{a}_t) \right] \leq \underbrace{\epsilon T + O(\epsilon T^2)}_{T \text{ terms, each } O(\epsilon T)}$$



# More general analysis

assume:  $\pi_\theta(\mathbf{a} \neq \pi^*(\mathbf{s})|\mathbf{s}) \leq \epsilon$

~~for all  $\mathbf{s} \in \mathcal{D}_{\text{train}}$~~  for  $\mathbf{s} \sim p_{\text{train}}(\mathbf{s})$

actually enough for  $E_{p_{\text{train}}(\mathbf{s})}[\pi_\theta(\mathbf{a} \neq \pi^*(\mathbf{s})|\mathbf{s})] \leq \epsilon$

if  $p_{\text{train}}(\mathbf{s}) \neq p_\theta(\mathbf{s})$ :

$$p_\theta(\mathbf{s}_t) = \underbrace{(1 - \epsilon)^t}_{\text{probability we made no mistakes}} p_{\text{train}}(\mathbf{s}_t) + (1 - (1 - \epsilon)^t) \underbrace{p_{\text{mistake}}(\mathbf{s}_t)}_{\text{some other distribution}}$$

$$c(\mathbf{s}, \mathbf{a}) = \begin{cases} 0 & \text{if } \mathbf{a} = \pi^*(\mathbf{s}) \\ 1 & \text{otherwise} \end{cases}$$

with DAgger,  $p_{\text{train}}(\mathbf{s}) \rightarrow p_\theta(\mathbf{s})$

$$E \left[ \sum_t c(\mathbf{s}_t, \mathbf{a}_t) \right] \leq \epsilon T$$

# More general analysis

assume:  $\pi_\theta(\mathbf{a} \neq \pi^*(\mathbf{s})|\mathbf{s}) \leq \epsilon$

for all  $\mathbf{s} \in \mathcal{D}_{\text{train}}$  for  $\mathbf{s} \sim p_{\text{train}}(\mathbf{s})$

$$p_\theta(\mathbf{s}_t) = \underbrace{(1 - \epsilon)^t}_{\text{probability we made no mistakes}} p_{\text{train}}(\mathbf{s}_t) + (1 - (1 - \epsilon)^t) \underbrace{p_{\text{mistake}}(\mathbf{s}_t)}_{\text{some other distribution}}$$

probability we made no mistakes

some *other* distribution

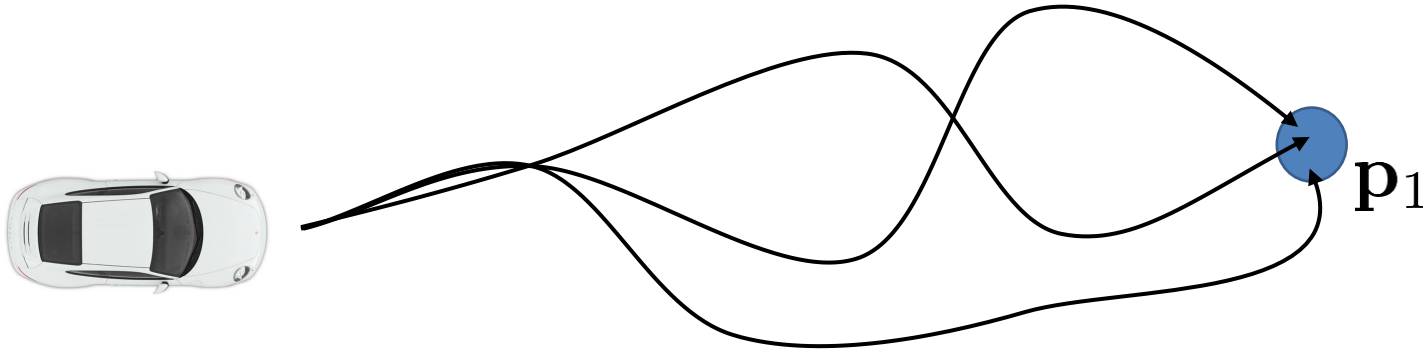
$$|p_\theta(\mathbf{s}_t) - p_{\text{train}}(\mathbf{s}_t)| = (1 - (1 - \epsilon)^t) |p_{\text{mistake}}(\mathbf{s}_t) - p_{\text{train}}(\mathbf{s}_t)| \leq 2(1 - (1 - \epsilon)^t)$$

useful identity:  $(1 - \epsilon)^t \geq 1 - \epsilon t$  for  $\epsilon \in [0, 1]$   $\leq 2\epsilon t$

$$\begin{aligned} \sum_t E_{p_\theta(\mathbf{s}_t)}[c_t] &= \sum_t \sum_{\mathbf{s}_t} p_\theta(\mathbf{s}_t) c_t(\mathbf{s}_t) \leq \sum_t \sum_{\mathbf{s}_t} p_{\text{train}}(\mathbf{s}_t) c_t(\mathbf{s}_t) + |p_\theta(\mathbf{s}_t) - p_{\text{train}}(\mathbf{s}_t)| c_{\max} \\ &\leq \sum_t \epsilon + 2\epsilon t \\ &O(\epsilon T^2) \end{aligned}$$

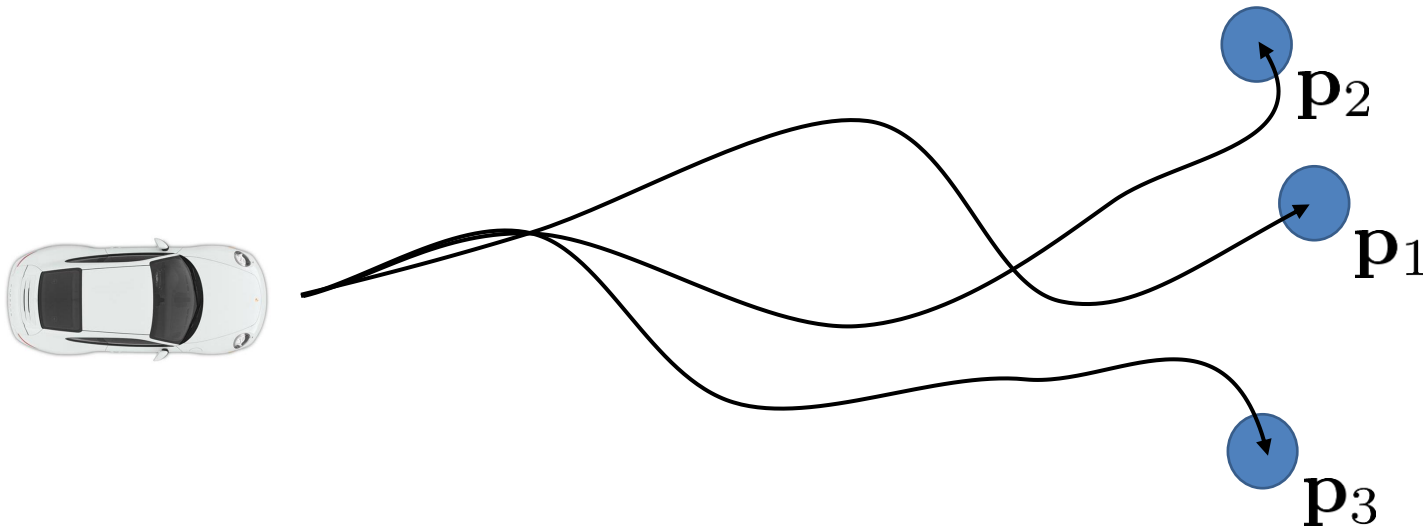
Another way to imitate

# Another imitation idea



$$\pi_{\theta}(\mathbf{a}|\mathbf{s})$$

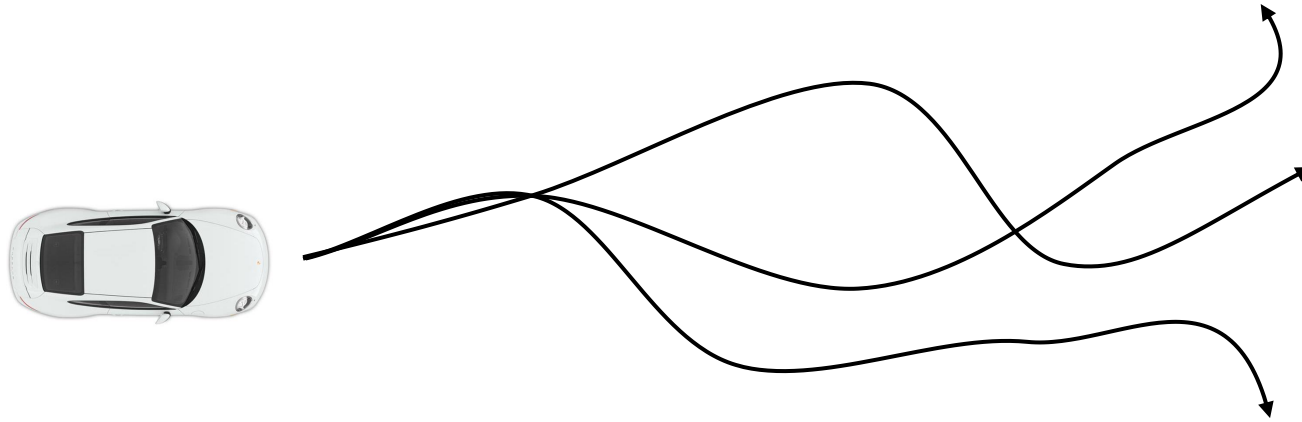
policy for reaching  $\mathbf{p}_1$



$$\pi_{\theta}(\mathbf{a}|\mathbf{s}, \mathbf{p})$$

policy for reaching *any*  $\mathbf{p}$

# Goal-conditioned behavioral cloning



training time:

demo 1:  $\{\mathbf{s}_1, \mathbf{a}_1, \dots, \mathbf{s}_{T-1}, \mathbf{a}_{T-1}, \mathbf{s}_T\}$  ← successful demo for reaching  $\mathbf{s}_T$

demo 2:  $\{\mathbf{s}_1, \mathbf{a}_1, \dots, \mathbf{s}_{T-1}, \mathbf{a}_{T-1}, \mathbf{s}_T\}$  learn  $\pi_\theta(\mathbf{a}|\mathbf{s}, \mathbf{g})$  ← goal state

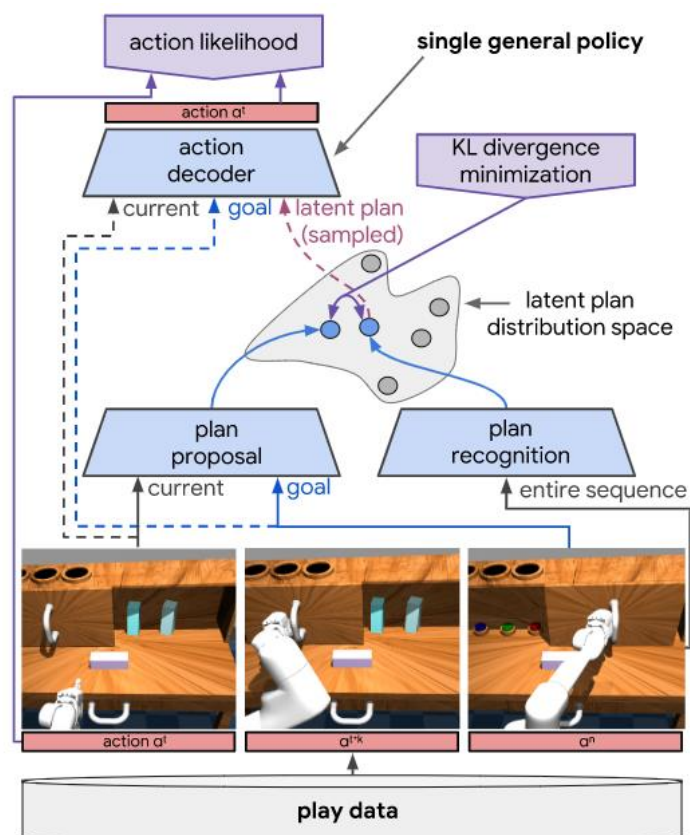
demo 3:  $\{\mathbf{s}_1, \mathbf{a}_1, \dots, \mathbf{s}_{T-1}, \mathbf{a}_{T-1}, \mathbf{s}_T\}$

for each demo  $\{\mathbf{s}_1^i, \mathbf{a}_1^i, \dots, \mathbf{s}_{T-1}^i, \mathbf{a}_{T-1}^i, \mathbf{s}_T^i\}$

maximize  $\log \pi_\theta(\mathbf{a}_t^i | \mathbf{s}_t^i, \mathbf{g} = \mathbf{s}_T^i)$

# Learning Latent Plans from Play

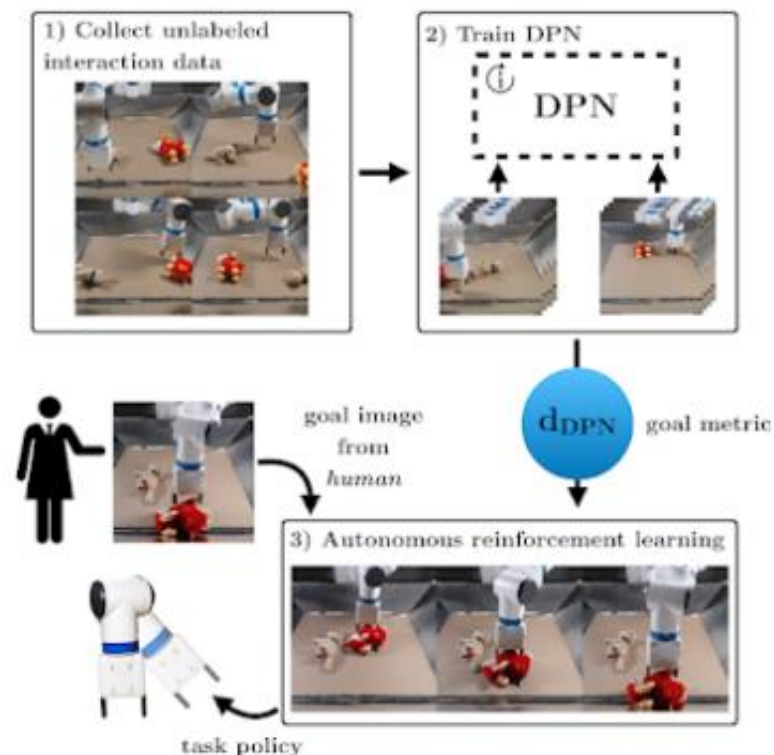
COREY LYNCH   MOHI KHANSARI   TED XIAO   VIKASH KUMAR   JONATHAN TOMPSON   SERGEY LEVINE   PIERRE SERMANET  
Google Brain   Google X   Google Brain   Google Brain   Google Brain   Google Brain   Google Brain



## Unsupervised Visuomotor Control through Distributional Planning Networks

[Tianhe Yu](#), [Gleb Shevchuk](#), [Dorsa Sadigh](#), [Chelsea Finn](#)

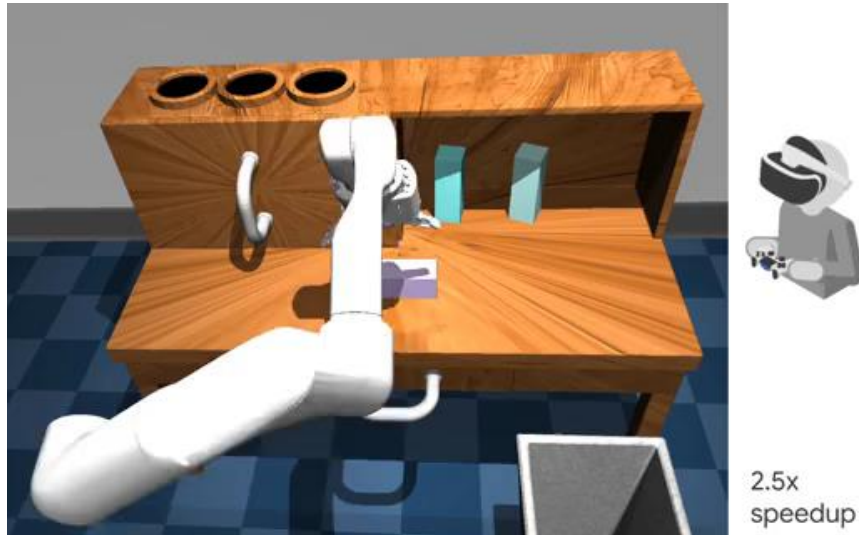
Stanford University



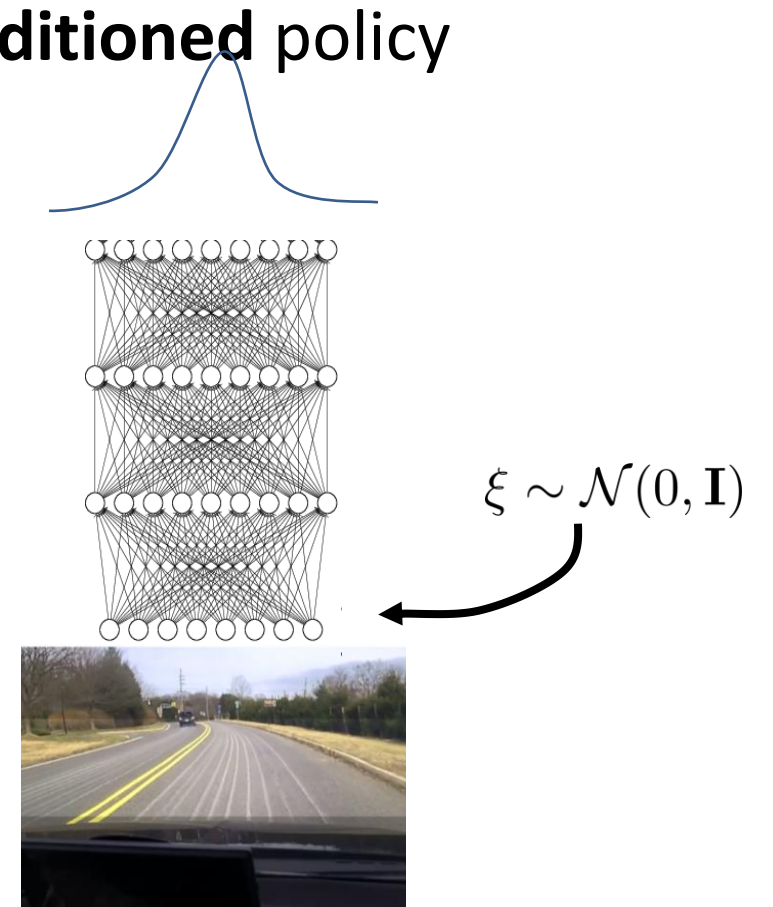
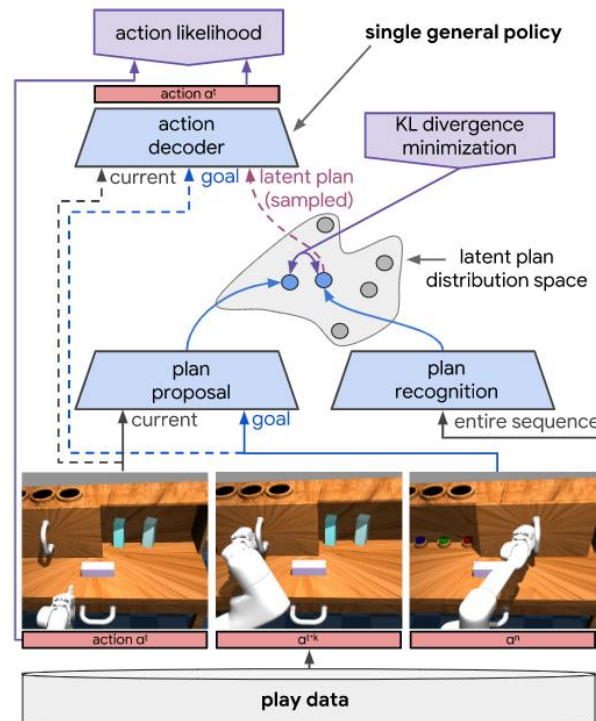
# Learning Latent Plans from Play

COREY LYNCH	MOHI KHANSARI	TED XIAO	VIKASH KUMAR	JONATHAN TOMPSON	SERGEY LEVINE	PIERRE SERMANET
Google Brain	Google X	Google Brain	Google Brain	Google Brain	Google Brain	Google Brain

## 1. Collect data



## 2. Train goal conditioned policy

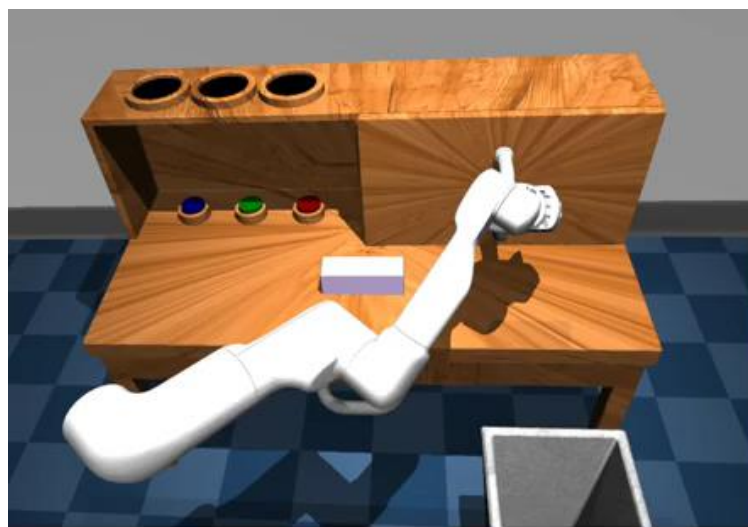




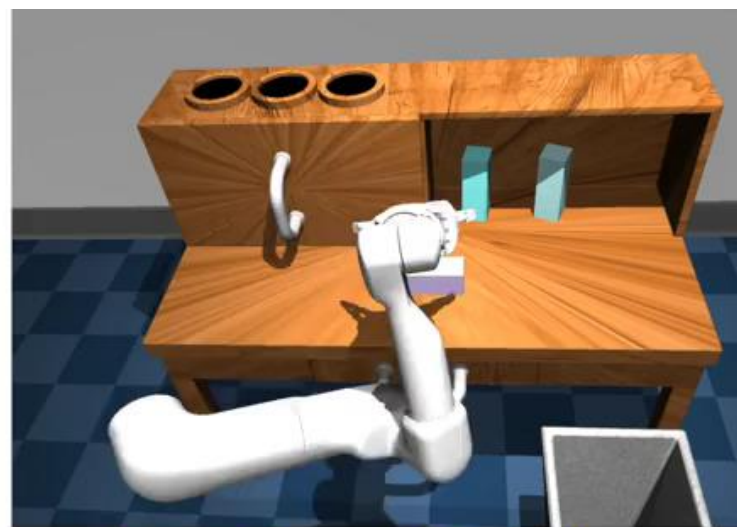
# Learning Latent Plans from Play

COREY LYNCH   MOHI KHANSARI   TED XIAO   VIKASH KUMAR   JONATHAN TOMPSON   SERGEY LEVINE   PIERRE SERMANET  
Google Brain   Google X   Google Brain   Google Brain   Google Brain   Google Brain   Google Brain

## 3. Reach goals



Goal



Single Play-LMP policy



# Going beyond just imitation?

## Learning to Reach Goals via Iterated Supervised Learning

Dibya Ghosh\*  
UC Berkeley

Abhishek Gupta\*  
UC Berkeley

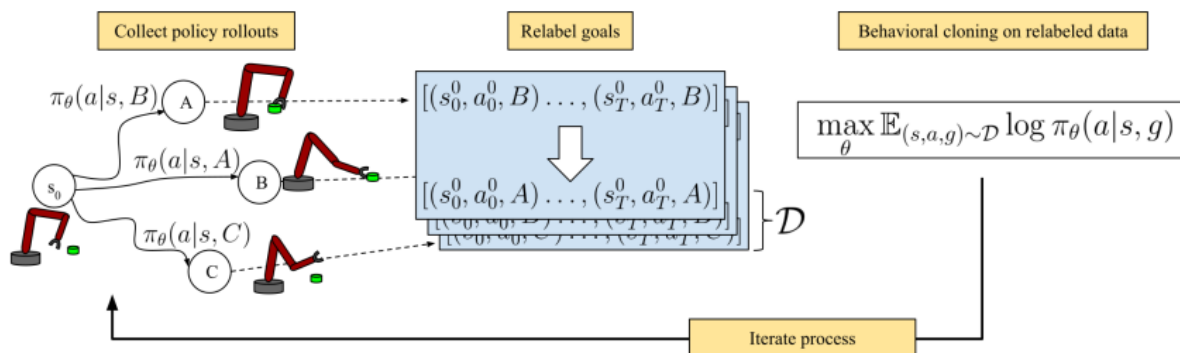
Ashwin Reddy  
UC Berkeley

Justin Fu  
UC Berkeley

Coline Devin  
UC Berkeley

Benjamin Eysenbach  
Carnegie Mellon University

Sergey Levine  
UC Berkeley



- Start with a **random** policy
- Collect data with **random** goals
- Treat this data as “demonstrations” for the goals that were reached
- Use this to improve the policy
- Repeat

# UC Berkeley·CS285 | Deep Reinforcement Learning (2020)

## CS285 (2020)· 课程资料包 @ShowMeAI



视频

中英双语字幕



课件

一键打包下载



笔记

官方笔记翻译



代码

作业项目解析



视频·B 站 [ 扫码或点击链接 ]

<https://www.bilibili.com/video/BV12341167kL>



课件 & 代码·博客 [ 扫码或点击链接 ]

<http://blog.showmeai.tech/cs285>

Berkeley

actor-critic

reinforcement learning

exploration

元学习

exploitation

梯度策略

探索与利用

DQN

迁移学习

逆强化学习

Q-learning / Q 学习

deep q network

meta learning

Awesome AI Courses Notes Cheatsheets 是 [ShowMeAI](#) 资料库的分支系列，覆盖最具知名度的 **TOP20+** 门 AI 课程，旨在为读者和学习者提供一整套高品质中文学习笔记和速查表。

点击课程名称，跳转至课程**资料包**页面，**一键下载**课程全部资料！

机器学习	深度学习	自然语言处理	计算机视觉
Stanford · CS229	Stanford · CS230	Stanford · CS224n	Stanford · CS231n
# Awesome AI Courses Notes Cheatsheets· 持续更新中			
知识图谱	图机器学习	深度强化学习	自动驾驶
Stanford · CS520	Stanford · CS224W	UCBerkeley · CS285	MIT · 6.S094



微信公众号

资料下载方式 2: 扫码点击**底部菜单栏**

称为 **AI 内容创作者**? 回复 [ 添砖加瓦 ]