From Learning and Control to Deep Reinforcement Learning

Benjamin Recht
University of California, Berkeley

I took four courses in graduate school that completely revolutionized my thinking:

I took four courses in graduate school that completely revolutionized my thinking:

6.432: Detection and Estimation with Wornell
9.520 Statistical Learning Theory with Poggio
6.24x: Complex systems with Megretski
6.253: Convex optimization with Berteskas

I took four courses in graduate school that completely revolutionized my thinking:

6.432: Detection and Estimation with Wornell
9.520 Statistical Learning Theory with Poggio
6.24x: Complex systems with Megretski
6.253: Convex optimization with Berteskas

4 of these were LIDS courses!

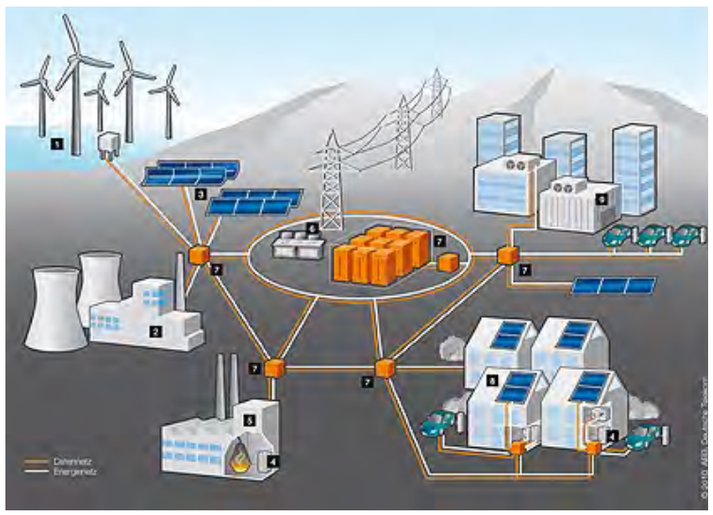
I took four courses in graduate school that completely revolutionized my thinking:

6.432: Detection and Estimation with Wornell
9.520 Statistical Learning Theory with Poggio
6.24x: Complex systems with Megretski
6.253: Convex optimization with Berteskas

4 of these were LIDS courses!

All are prerequisites for modern RL, but I never took an RL course…


nature
THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE
At last — a computer program that can beat a champion Go player PAGE 484
ALL SYSTEMS GO
CONSERVATION
SONGBIRDS À LA CARTE
Illegal harvest of millions of Mediterranean birds
PAGE 452
RESEARCH ETHICS
SAFEGUARD TRANSPARENCY
Don't let openness backfire on individuals
PAGE 459
POPULAR SCIENCE
WHEN GENES GOT 'SELFISH'
Dawkins's calling card 40 years on
PAGE 462
NATUREASIA.COM
28 January 2016
Vol. 529, No. 7587



trustable, scalable, predictable

Reinforcement Learning is the study of how to use past data to enhance the future manipulation of a dynamical system

Control Theory 🤔

~~Reinforcement Learning~~ is the study of how to use past data to enhance the future manipulation of a dynamical system

What is ML?

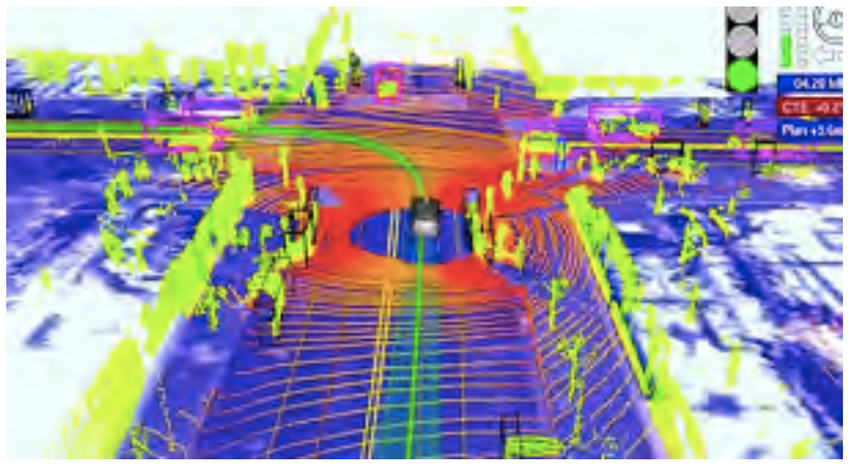
What is ML?

using past **data** to **learn** about and/or **act** upon the world

What is ML?

using past **data** to **learn** about and/or **act** upon the world

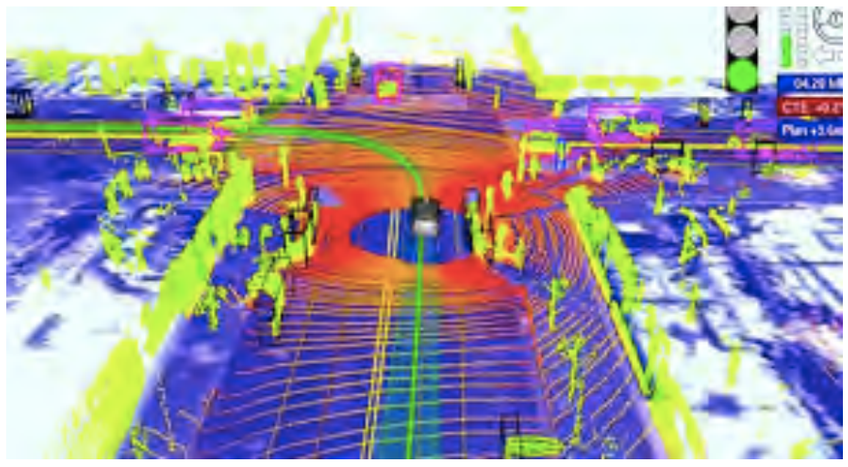
Environments too complex



What is ML?

using past **data** to **learn** about and/or **act** upon the world

Environments too complex

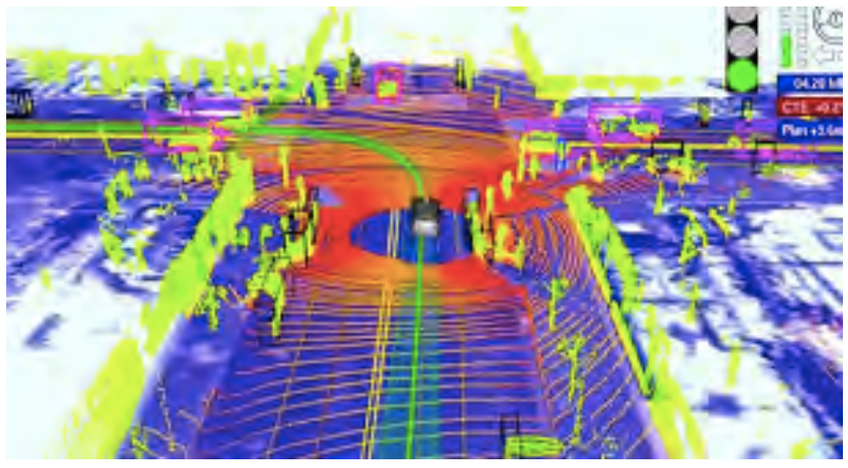


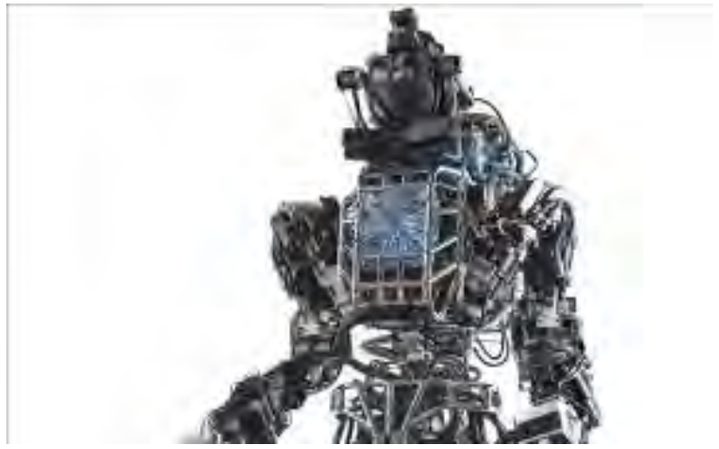
Sensing too complex

What is ML?

using past **data** to **learn** about and/or **act** upon the world

Environments	Sensing	Models
too complex	too complex	too complex





What is Control?

What is Control?

using **feedback** to **mitigate** the effects of **dynamic uncertainty**

What is Control?

using **feedback** to **mitigate** the effects of **dynamic uncertainty**

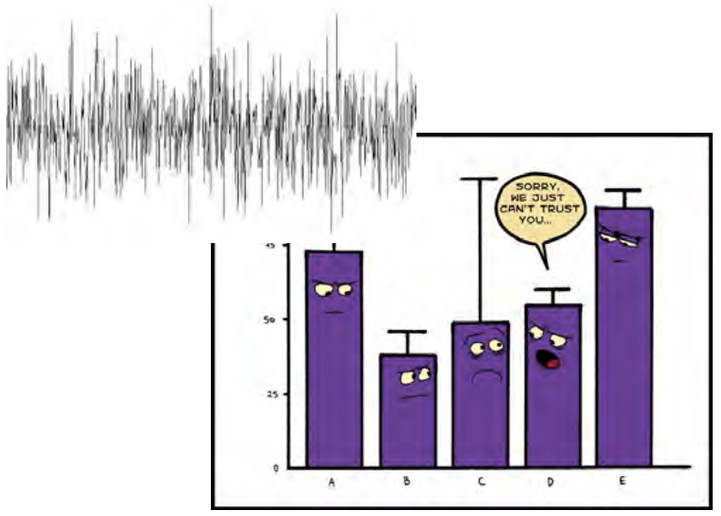
Environments are uncertain

What is Control?

using **feedback** to **mitigate** the effects of **dynamic uncertainty**

Environments are uncertain

Sensing/components are uncertain

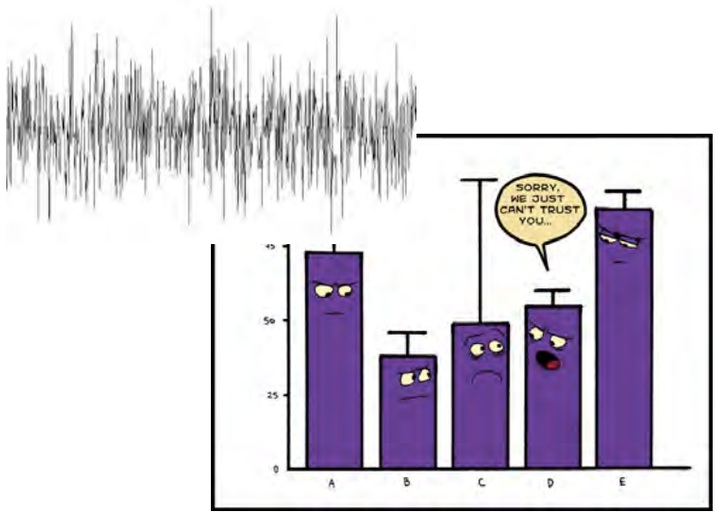


What is Control?

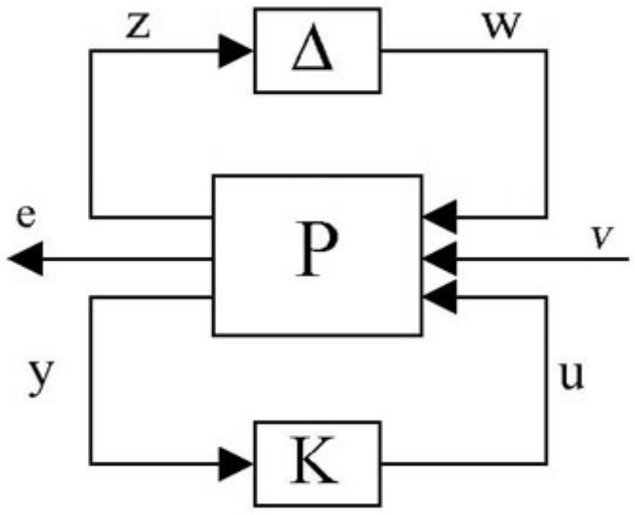
using **feedback** to **mitigate** the effects of **dynamic uncertainty**

Environments are uncertain

Sensing/components are uncertain



Models are uncertain

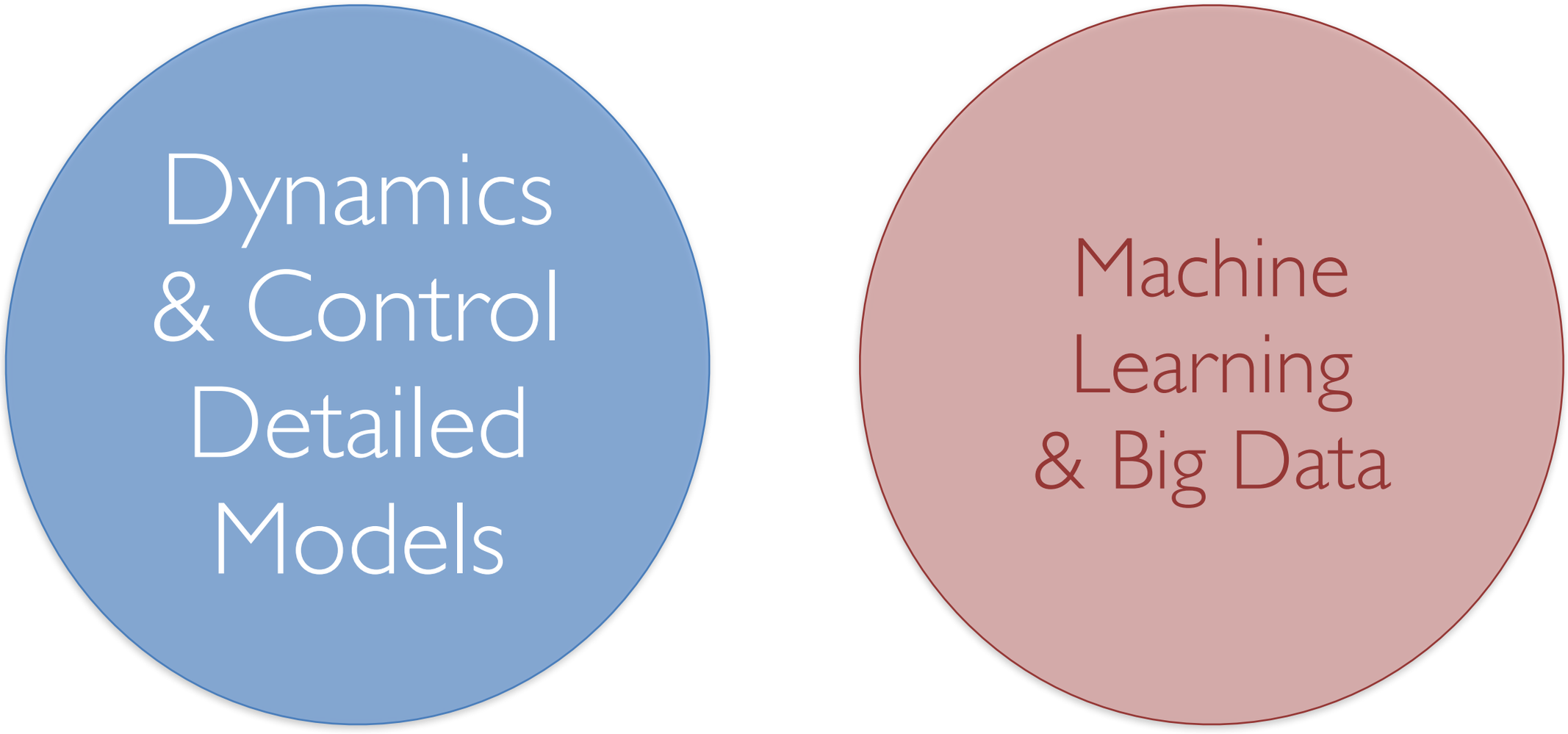


How do we get the best of both?

How do we get the best of both?

Dynamics & Control Detailed Models

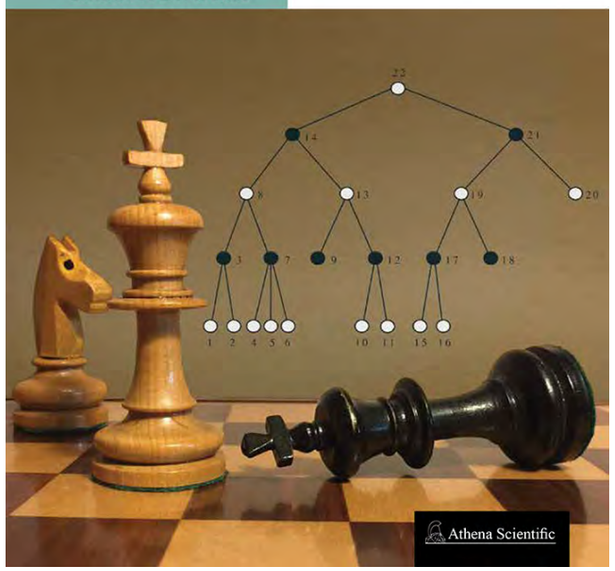
How do we get the best of both?



VOLUME 1 • 4th EDITION

Dynamic Programming and Optimal Control

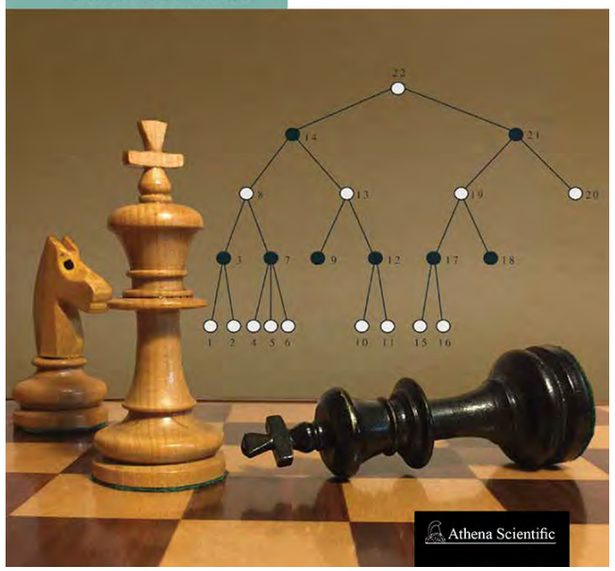
Dimitri P. Bertsekas



VOLUME 1 • 4th EDITION

Dynamic Programming and Optimal Control

Dimitri P. Bertsekas

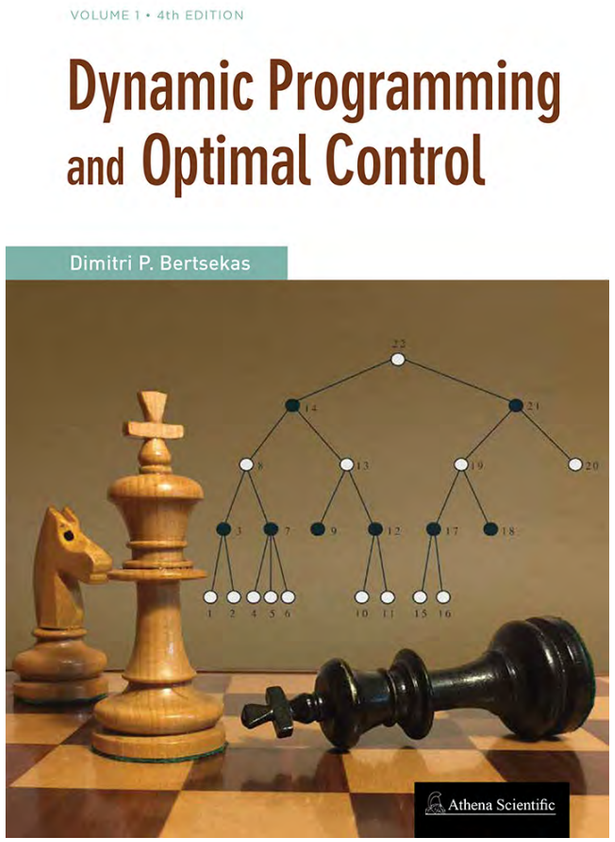


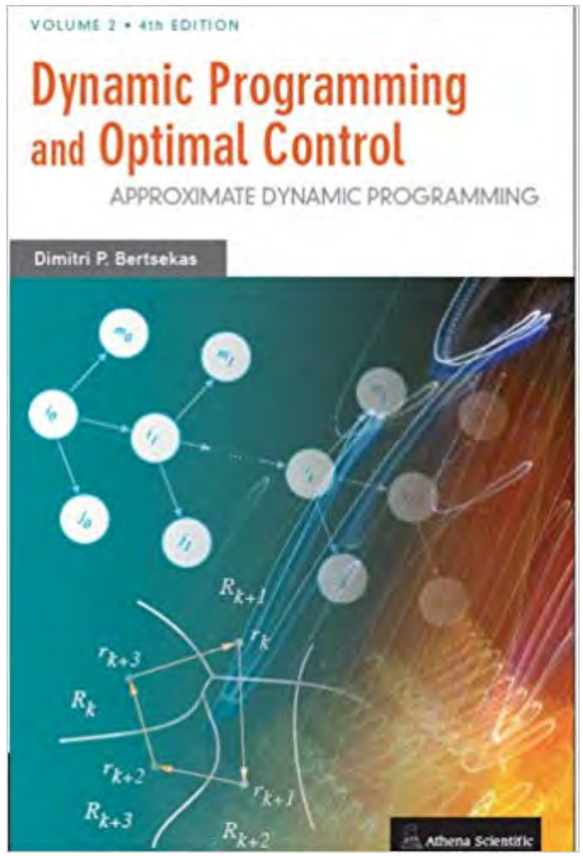
VOLUME 2 • 4th EDITION

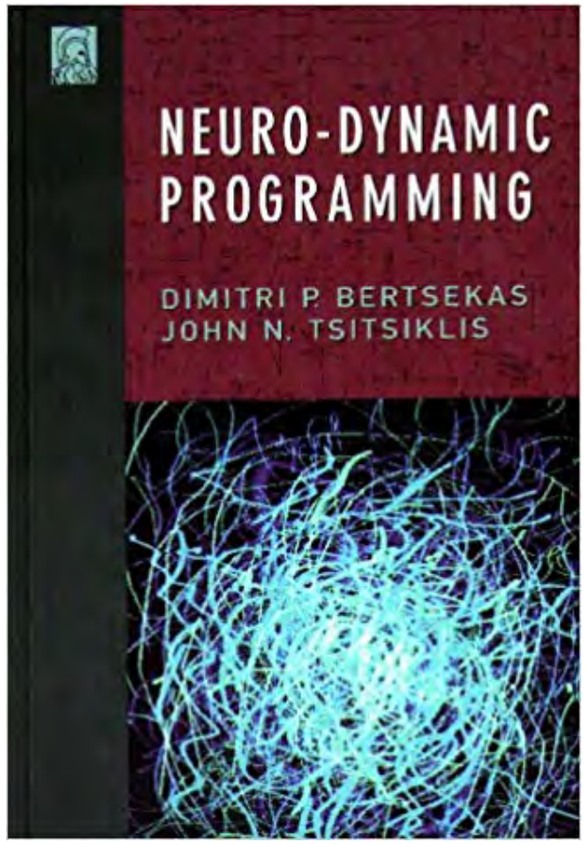
Dynamic Programming and Optimal Control

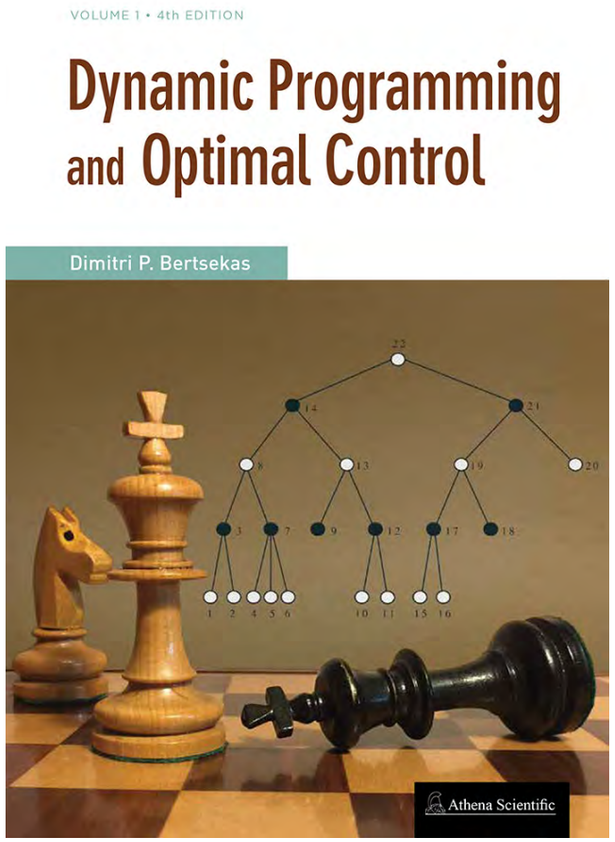
APPROXIMATE DYNAMIC PROGRAMMING

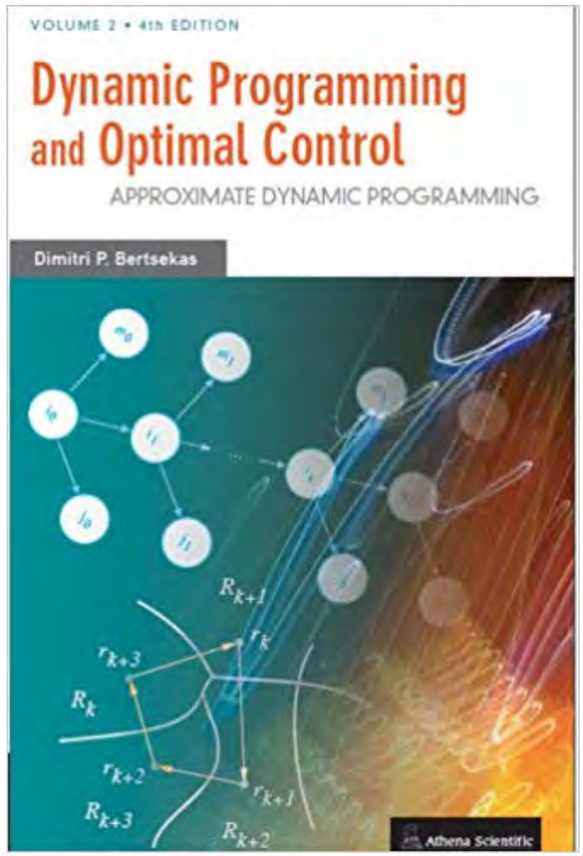
Dimitri P. Bertsekas

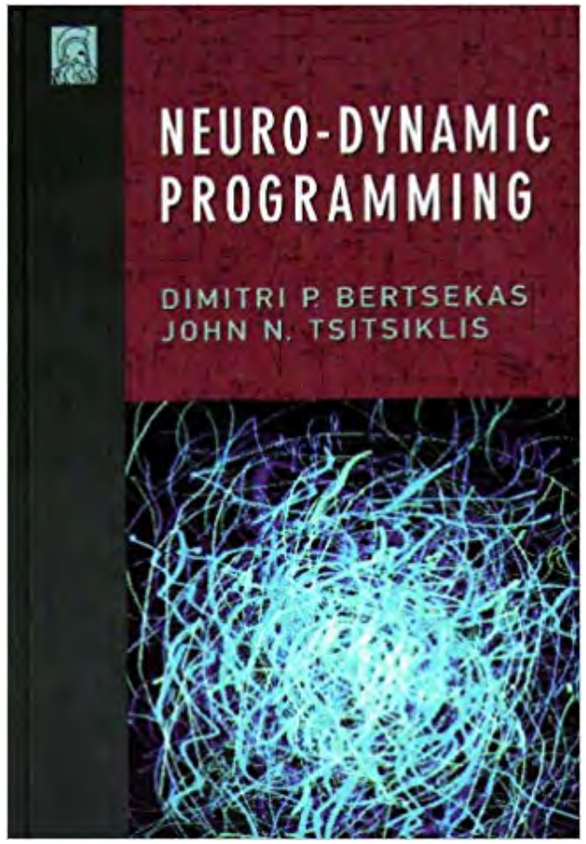

VOLUME 1 • 4th EDITION
Dynamic Programming
and Optimal Control
Dimitri P. Bertsekas
Athena Scientific

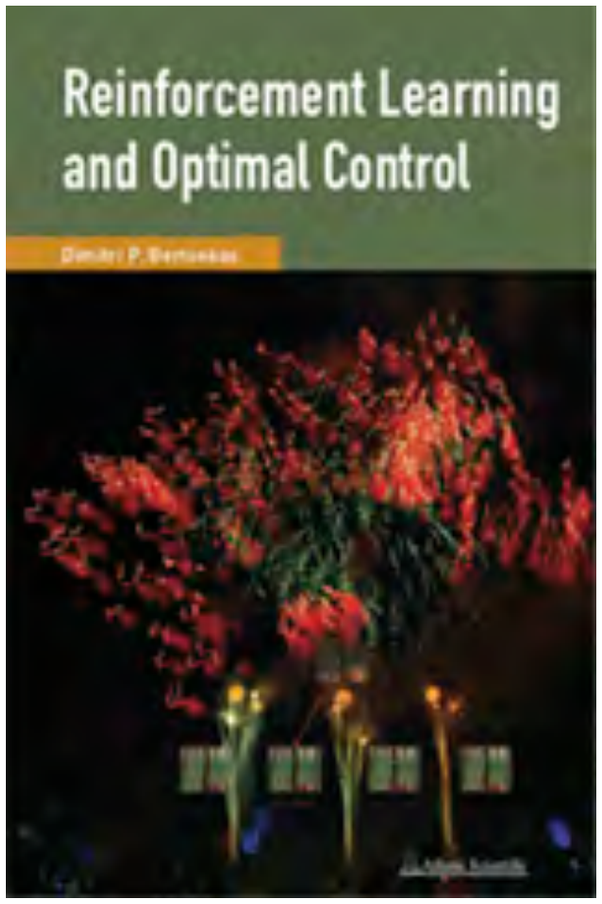

VOLUME 2 • 4th EDITION
Dynamic Programming
and Optimal Control
APPROXIMATE DYNAMIC PROGRAMMING
Dimitri P. Bertsekas
Athena Scientific


NEURO-DYNAMIC
PROGRAMMING
DIMITRI P. BERTSEKAS
JOHN N. TSITSIKLIS

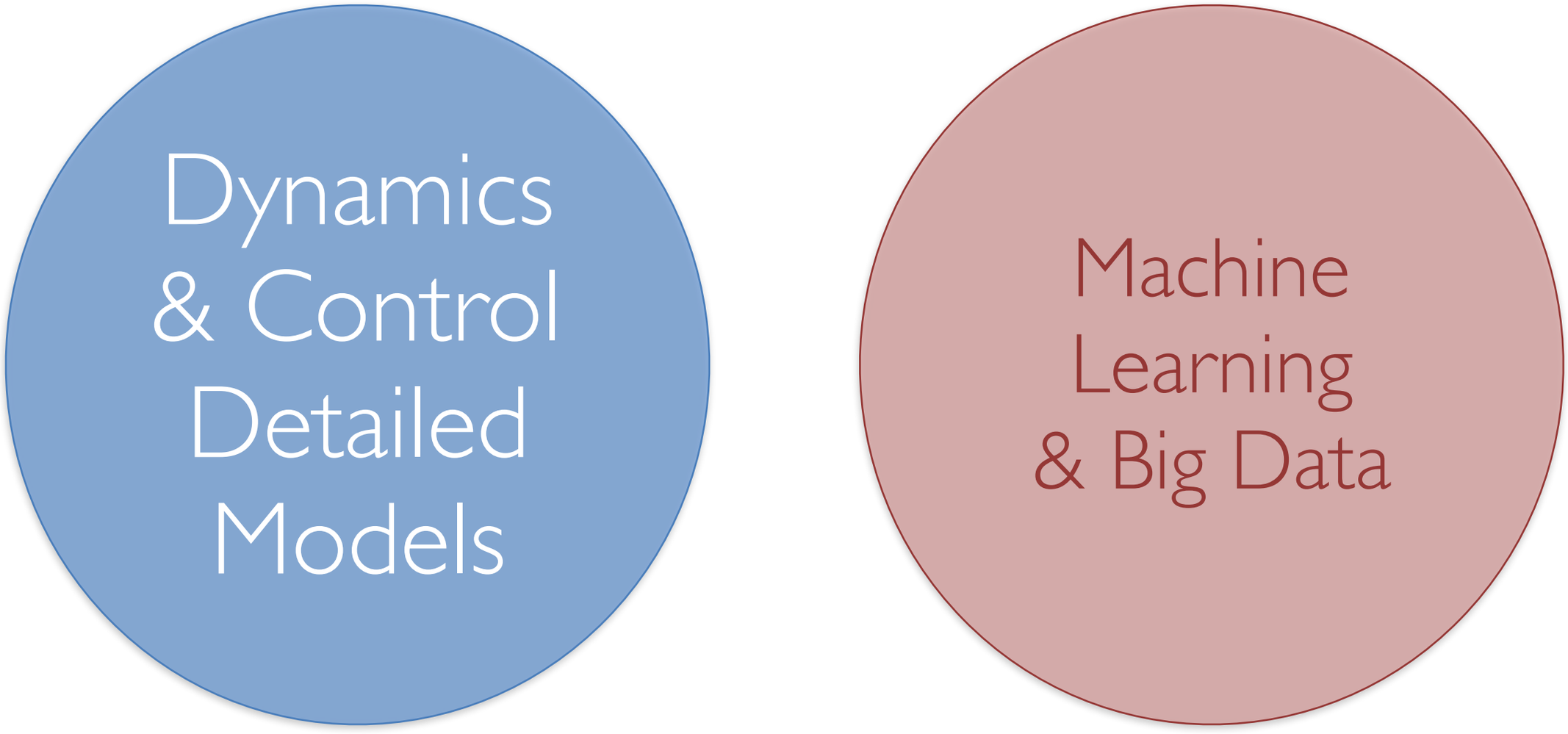

VOLUME 1 • 4th EDITION
Dynamic Programming
and Optimal Control
Dimitri P. Bertsekas
Athena Scientific


VOLUME 2 • 4th EDITION
Dynamic Programming
and Optimal Control
APPROXIMATE DYNAMIC PROGRAMMING
Dimitri P. Bertsekas
Athena Scientific

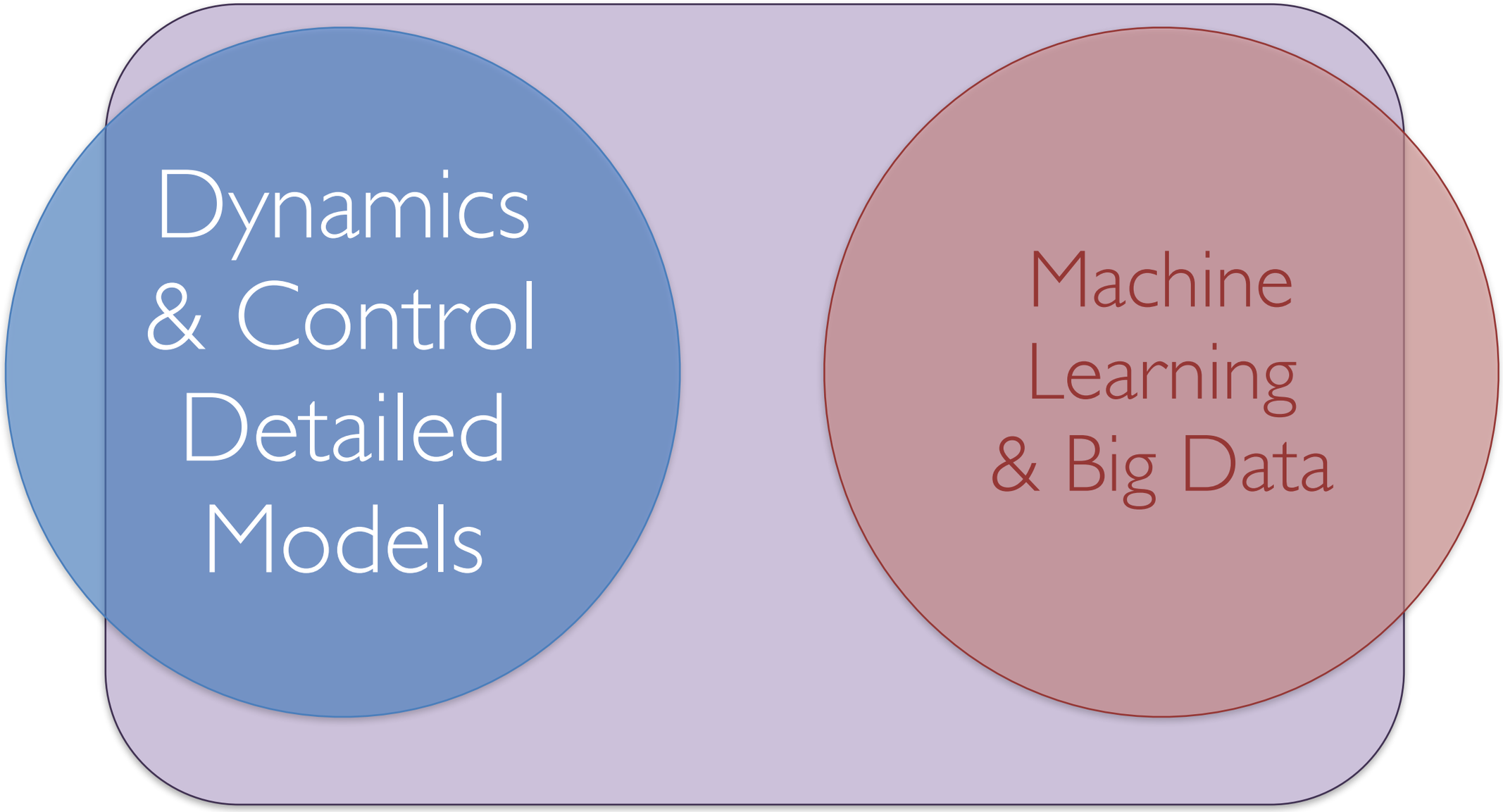

NEURO-DYNAMIC
PROGRAMMING
DIMITRI P. BERTSEKAS
JOHN N. TSITSIKLIS


Reinforcement Learning
and Optimal Control

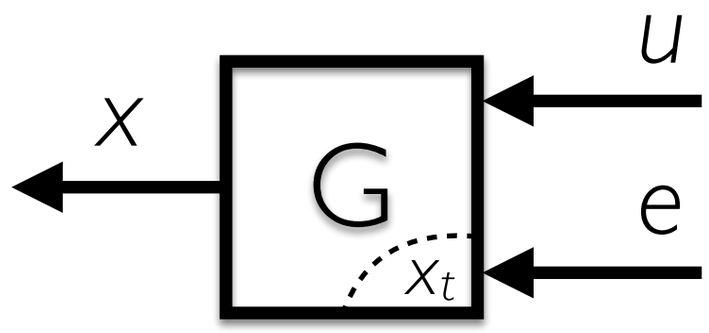
How do we get the best of both?



How do we get the best of both?



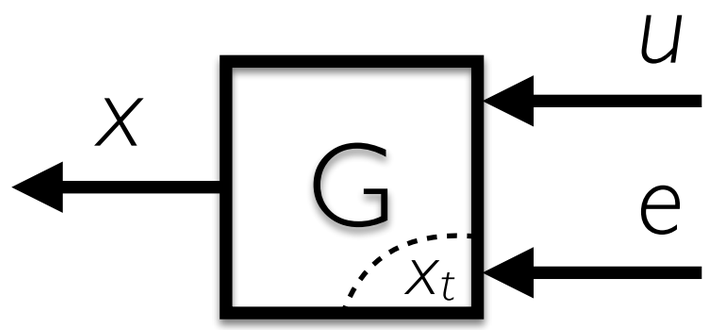
RL Methods



$$\begin{aligned} \text{minimize} \quad & \mathbb{E}_e \left[\sum_{t=1}^{T} C_t(x_t, u_t) \right] \\ \text{s.t.} \quad & x_{t+1} = f_t(x_t, u_t, e_t) \\ & u_t = \pi_t(\tau_t) \end{aligned}$$

How to solve optimal control when the model *f* is unknown?

RL Methods



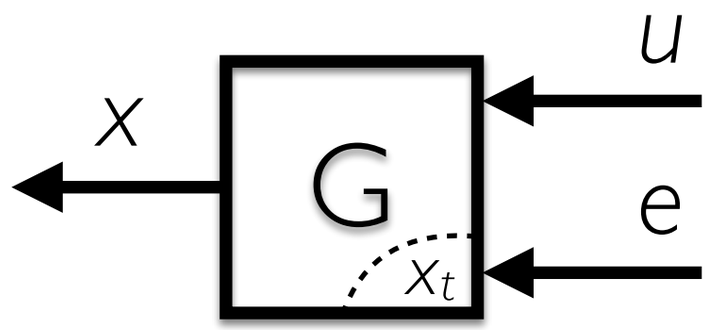
$$\begin{aligned} \text{minimize} \quad & \mathbb{E}_e \left[\sum_{t=1}^{T} C_t(x_t, u_t) \right] \\ \text{s.t.} \quad & x_{t+1} = f_t(x_t, u_t, e_t) \\ & u_t = \pi_t(\tau_t) \end{aligned}$$

model-based

How to solve optimal control when the model *f* is unknown?

- **Model-based:** fit model from data (aka, standard engineering practice)

RL Methods



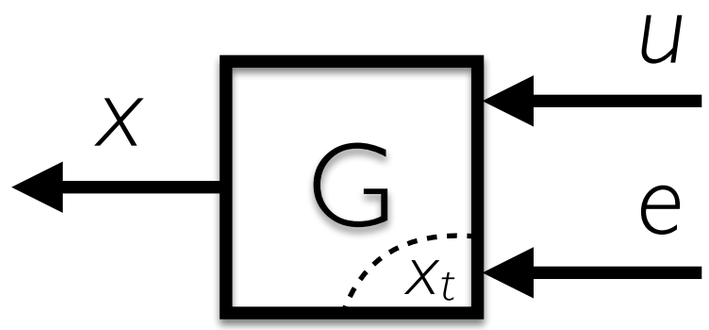
$$\begin{aligned} \text{minimize} \quad & \mathbb{E}_e \left[\sum_{t=1}^{T} C_t(x_t, u_t) \right] \\ \text{s.t.} \quad & x_{t+1} = f_t(x_t, u_t, e_t) \\ & u_t = \pi_t(\tau_t) \end{aligned}$$

model-based

How to solve optimal control when the model *f* is unknown?

- **Model-based:** fit model from data (aka, standard engineering practice)
- **Model-free**

RL Methods



minimize $\mathbb{E}_e \left[\sum_{t=1}^{T} C_t(x_t, u_t) \right]$

s.t. $x_{t+1} = f_t(x_t, u_t, e_t)$

$u_t = \pi_t(\tau_t)$

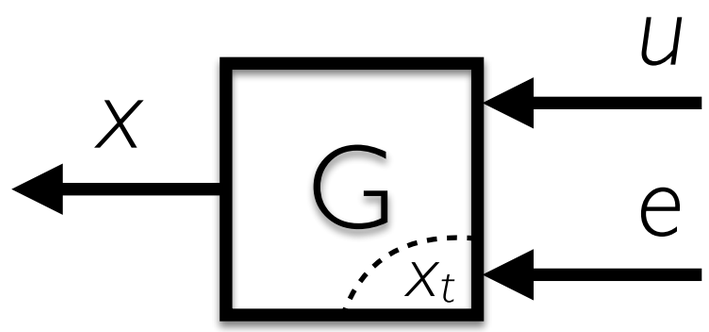
approximate dynamic programming

model-based

How to solve optimal control when the model *f* is unknown?

- **Model-based:** fit model from data (aka, standard engineering practice)
- **Model-free**
  - **Approximate dynamic programming:** estimate cost from data

RL Methods



minimize $\mathbb{E}_e \left[\sum_{t=1}^{T} C_t(x_t, u_t) \right]$

s.t. $x_{t+1} = f_t(x_t, u_t, e_t)$

$u_t = \pi_t(\tau_t)$

approximate dynamic programming

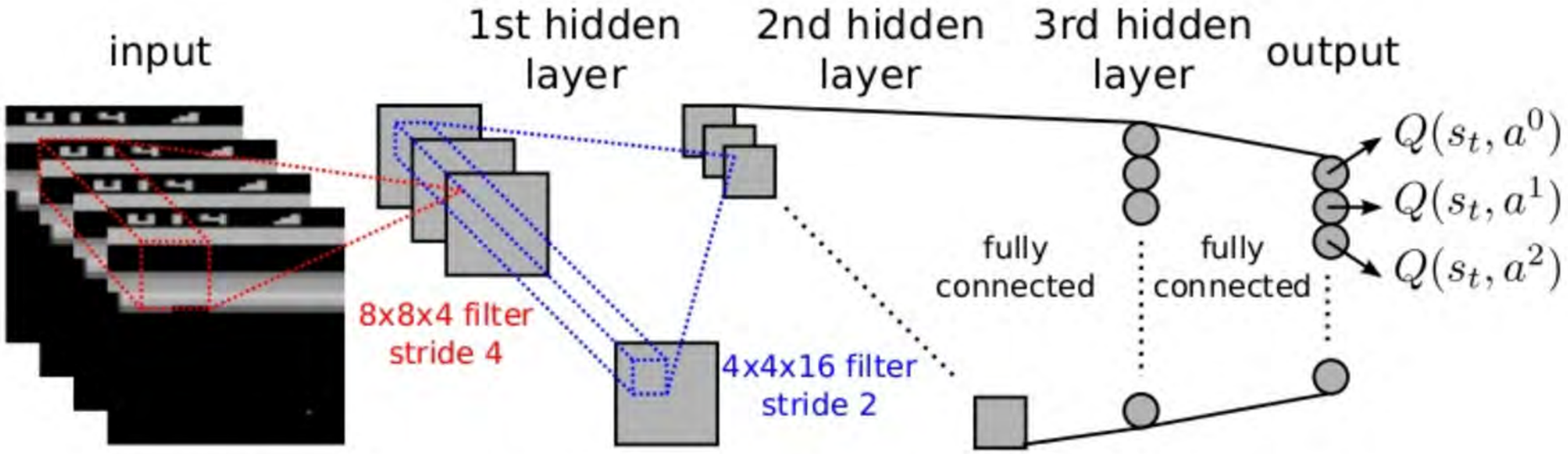
model-based

direct policy search

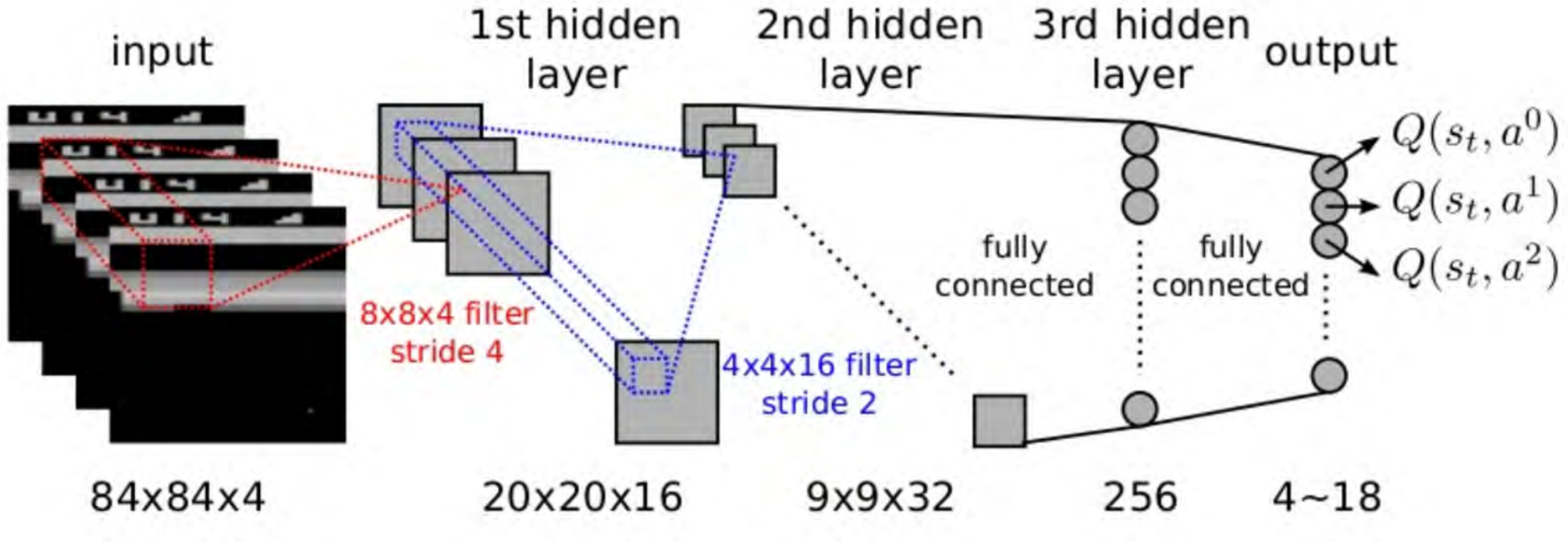
How to solve optimal control when the model *f* is unknown?

- **Model-based:** fit model from data (aka, standard engineering practice)
- **Model-free**
  - **Approximate dynamic programming:** estimate cost from data
  - **Direct policy search:** search for actions from data

Deep Reinforcement Learning

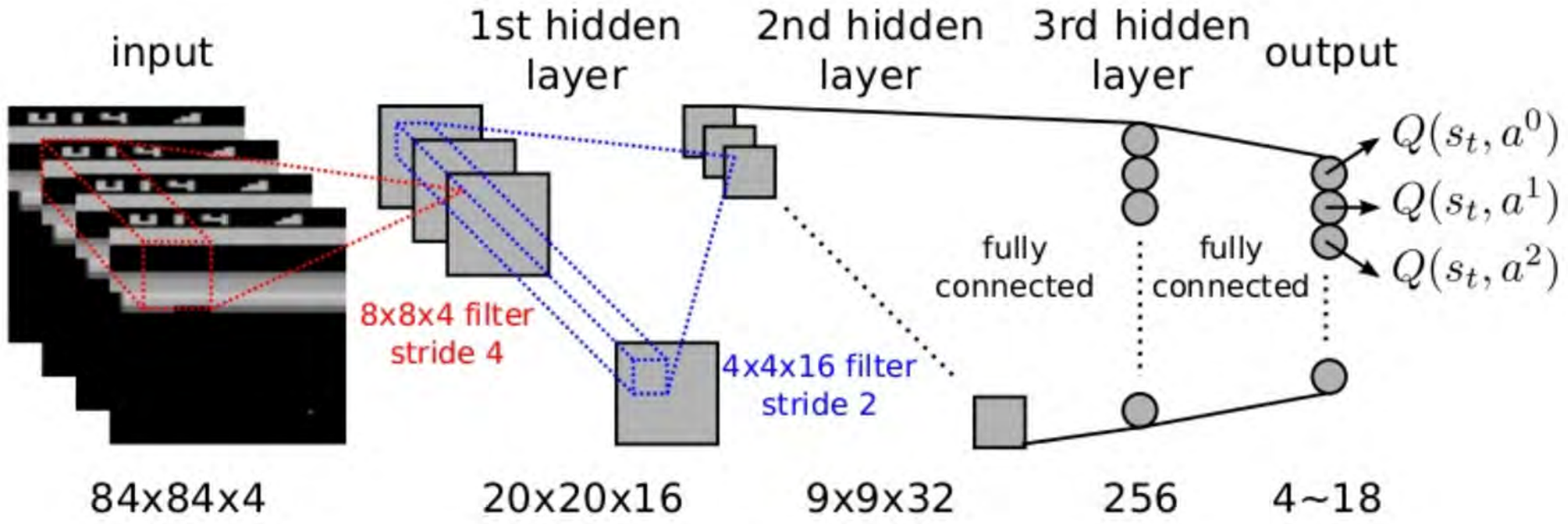


Deep Reinforcement Learning



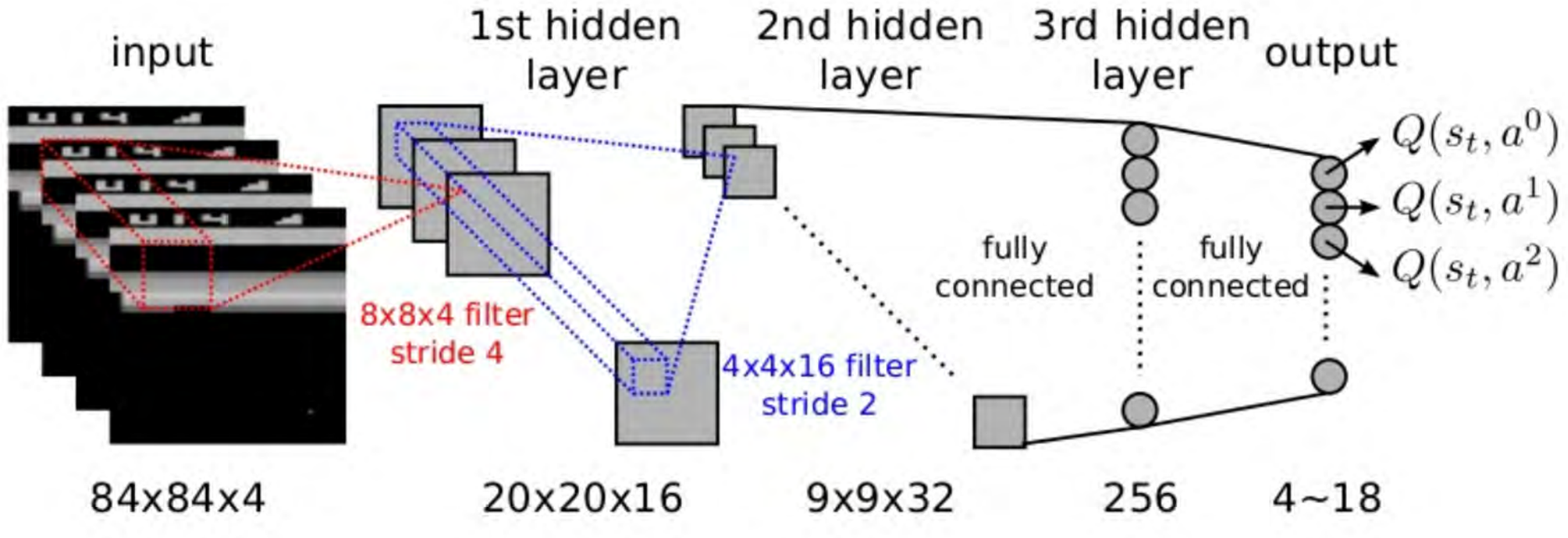
- Simply parameterize value function or policy as a deep net

Deep Reinforcement Learning



- Simply parameterize value function or policy as a deep net
- All of the ideas have been here since NDP!

Deep Reinforcement Learning



- Simply parameterize value function or policy as a deep net
- All of the ideas have been here since NDP!
- Most of these algorithms don't really "work."

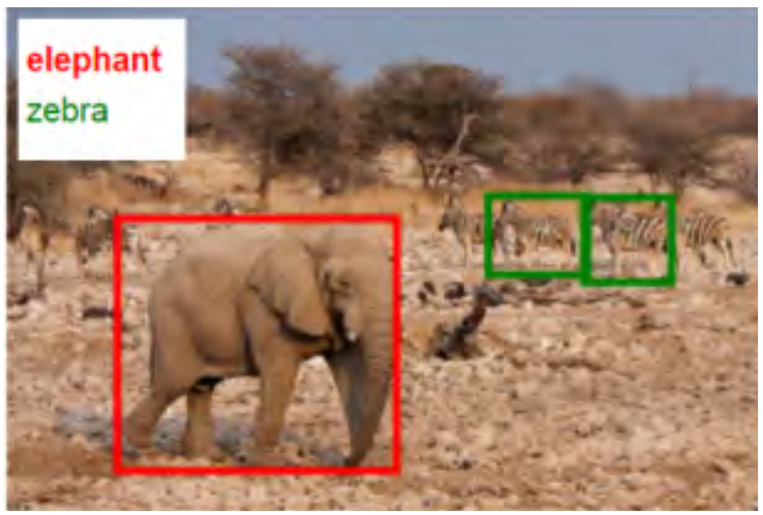
What is ML good for in control?

What is ML good for in control?

- Fundamentally, almost all machine learning successes are in *nonparametric prediction* (mostly classification).

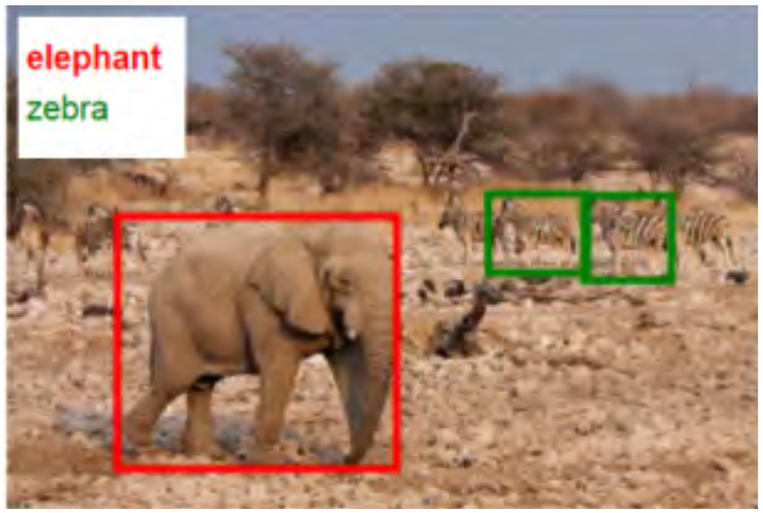
What is ML good for in control?

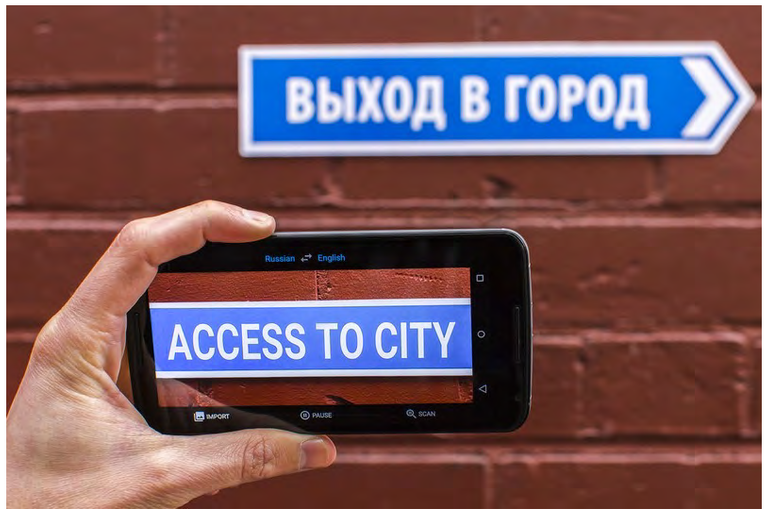
- Fundamentally, almost all machine learning successes are in *nonparametric prediction* (mostly classification).



What is ML good for in control?

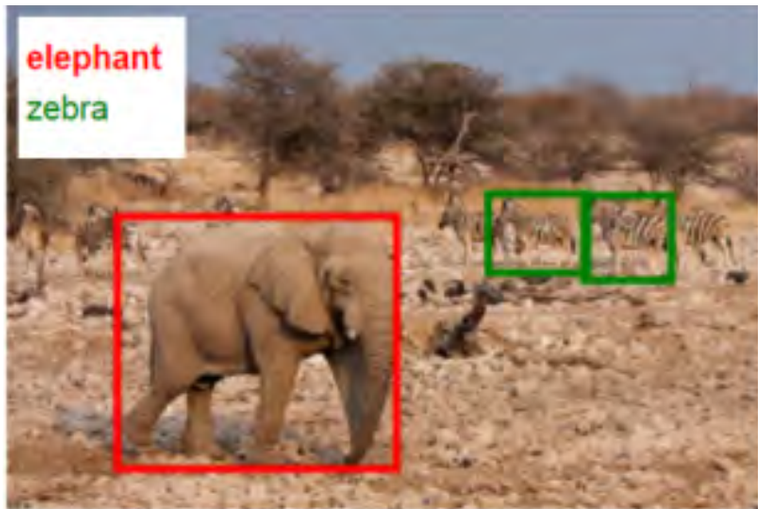
- Fundamentally, almost all machine learning successes are in *nonparametric prediction* (mostly classification).

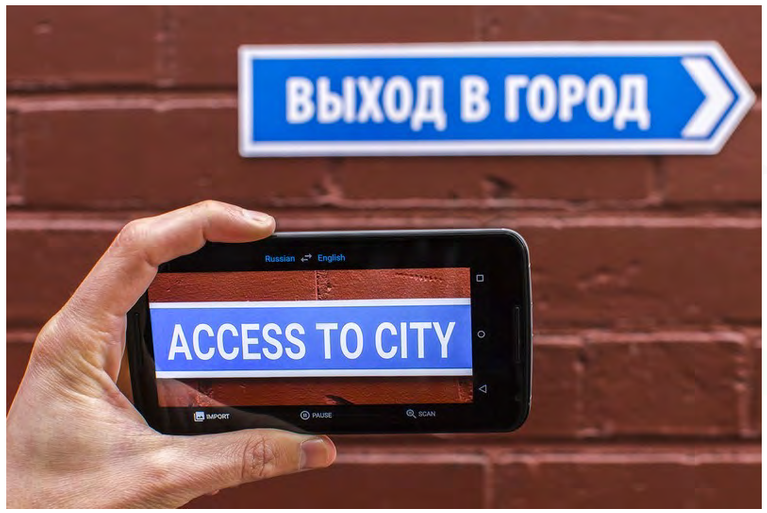


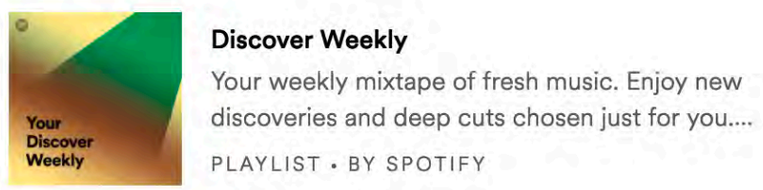


What is ML good for in control?

- Fundamentally, almost all machine learning successes are in *nonparametric prediction* (mostly classification).

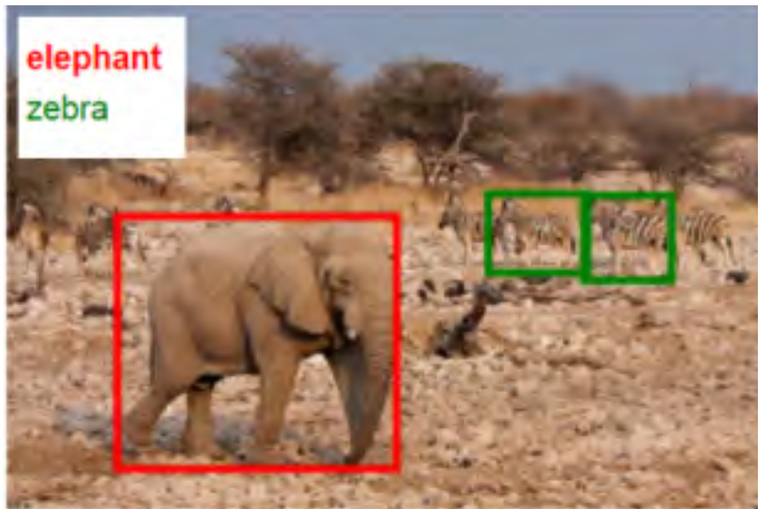


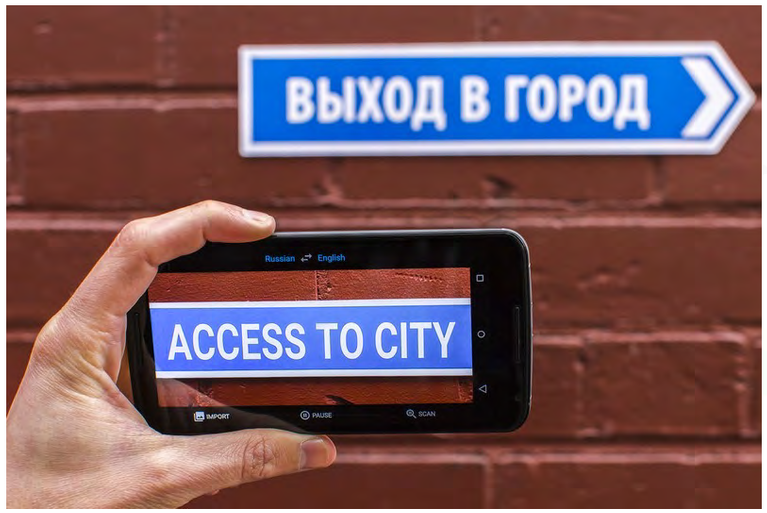


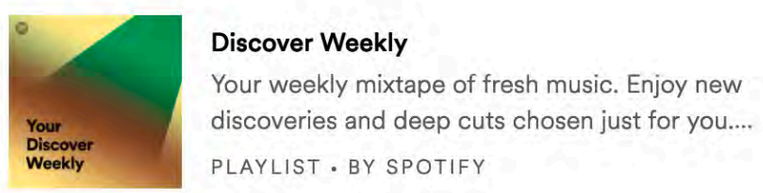


What is ML good for in control?

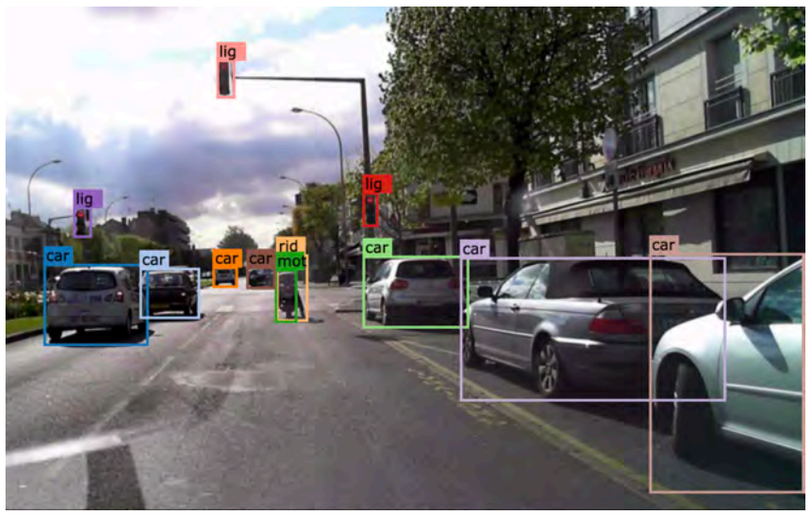
- Fundamentally, almost all machine learning successes are in *nonparametric prediction* (mostly classification).





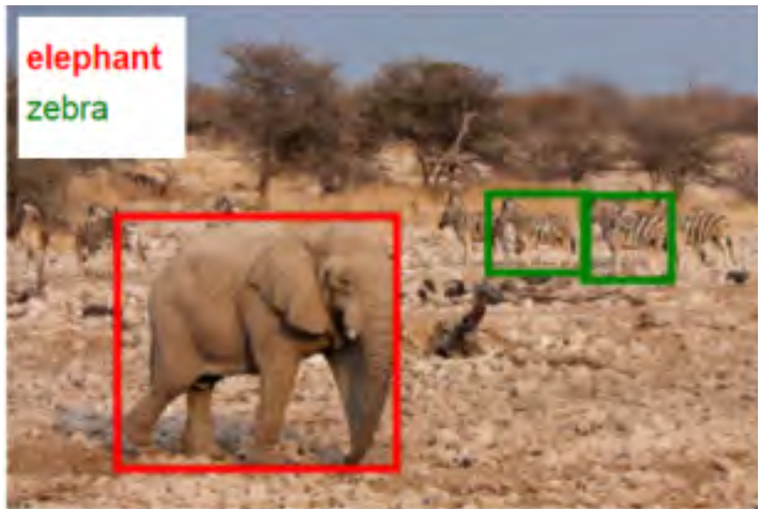


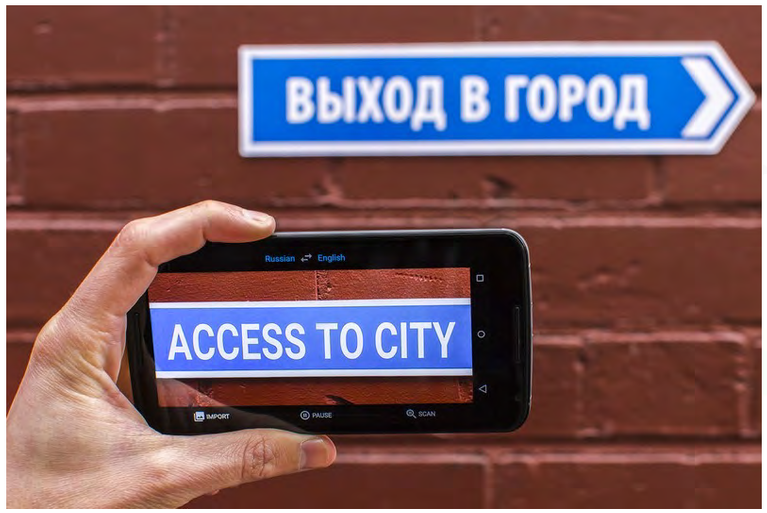
Perceptual sensors in the loop

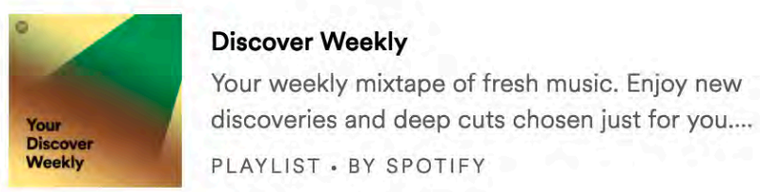


What is ML good for in control?

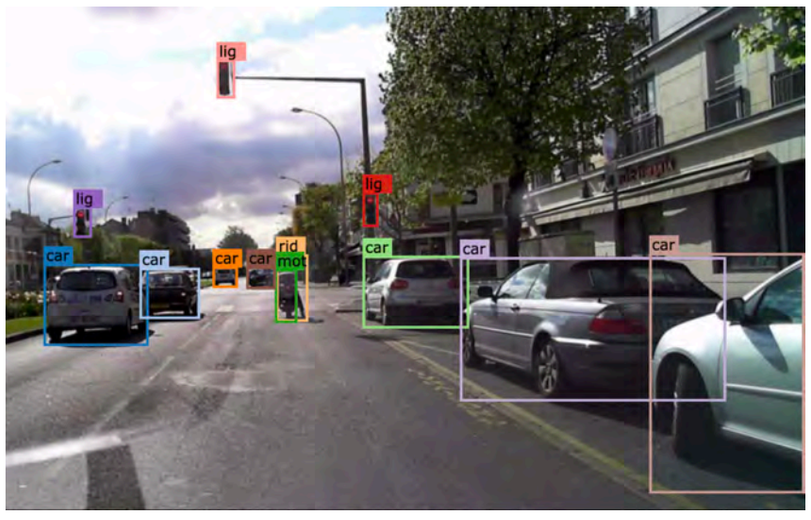
- Fundamentally, almost all machine learning successes are in *nonparametric prediction* (mostly classification).



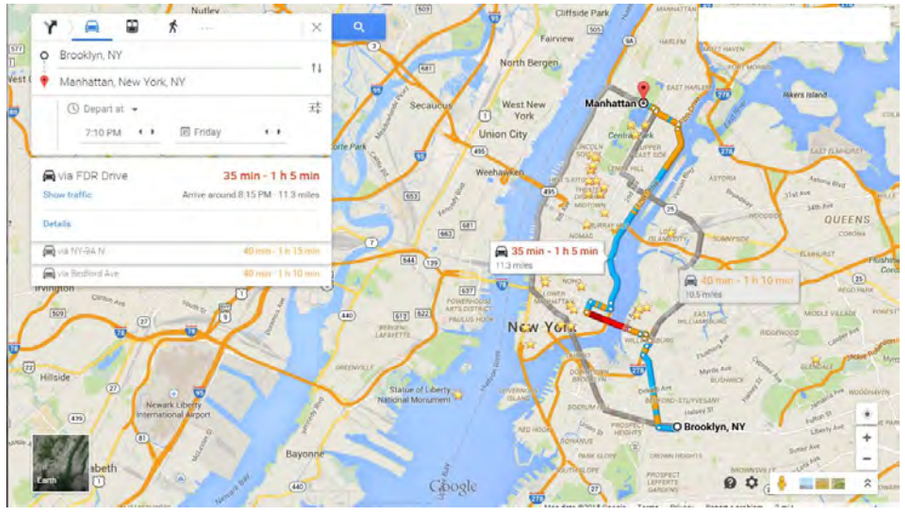




Perceptual sensors in the loop

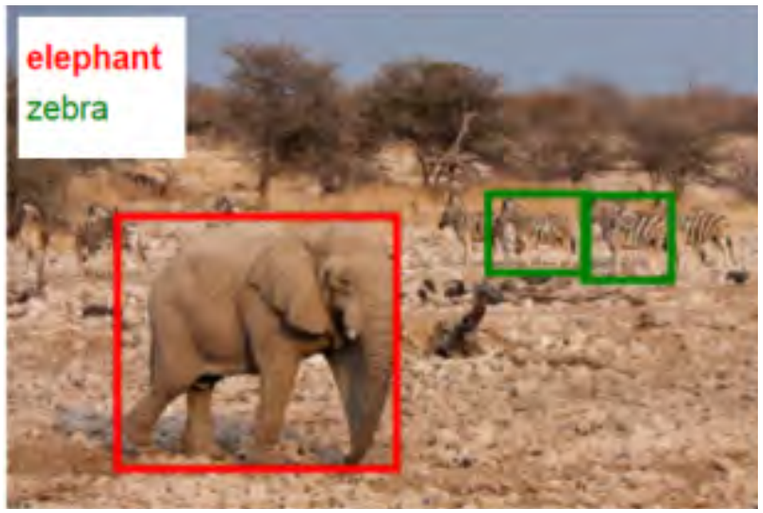


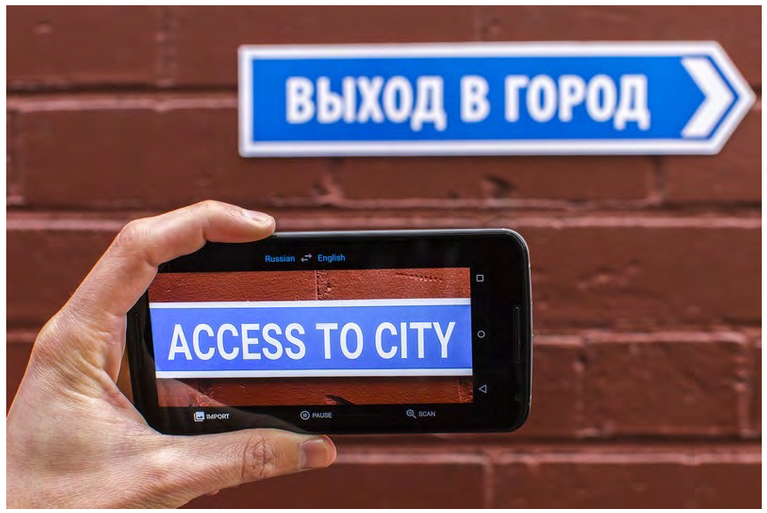
Forecasting in MPC

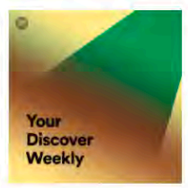


What is ML good for in control?

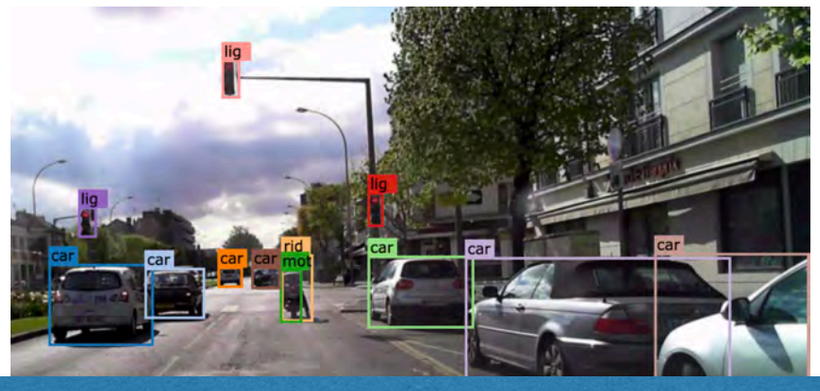
- Fundamentally, almost all machine learning successes are in *nonparametric prediction* (mostly classification).



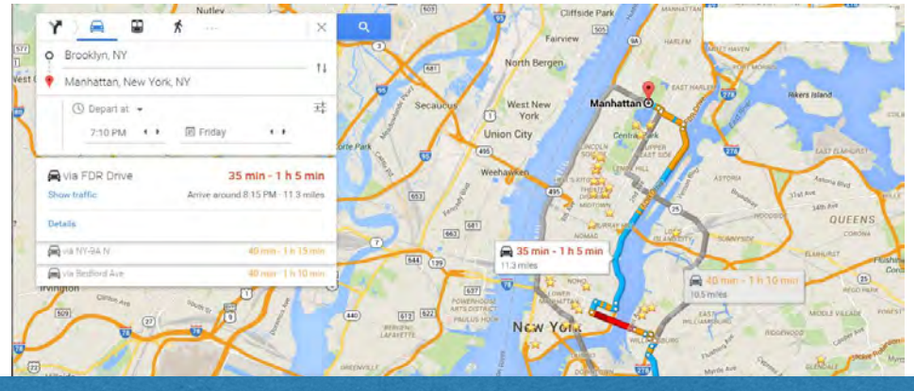




Perceptual sensors in the loop

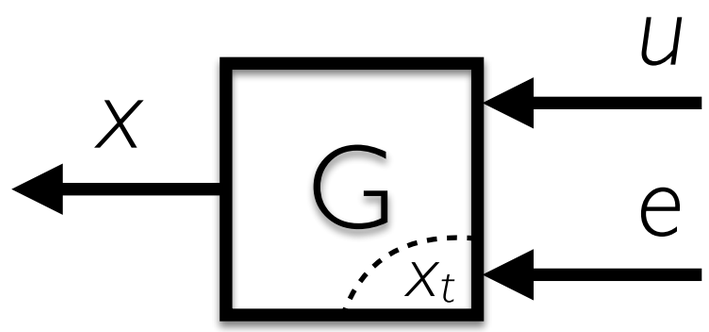


Forecasting in MPC



How to incorporate uncertain predictive perception in trustable, scalable, predictable autonomy?

managing uncertainty and learning in optimal control



minimize $\mathbb{E}_e \left[\sum_{t=1}^{T} C_t(x_t, u_t) \right]$ ← changing costs

s.t. $x_{t+1} = f_t(x_t, u_t, e_t)$ ← uncertain dynamics.

$u_t = \pi_t(\tau_t)$

$x_t \in \mathcal{X}, \quad u_t \in \mathcal{U}$ ← safety constraints

$z_t = g(x_t)$ ← perceptual sensing

How to incorporate uncertain predictive perception in trustable, scalable, predictable autonomy?

Actionable Intelligence is the study of how to use past data to enhance the future manipulation of a dynamical system

Actionable Intelligence is the study of how to use past data to enhance the future manipulation of a dynamical system

Actionable Intelligence is the study of how to use past data to enhance the future manipulation of a dynamical system

Actionable Intelligence is the study of how to use past data to enhance the future manipulation of a dynamical system

Recommendations for you in Electronics

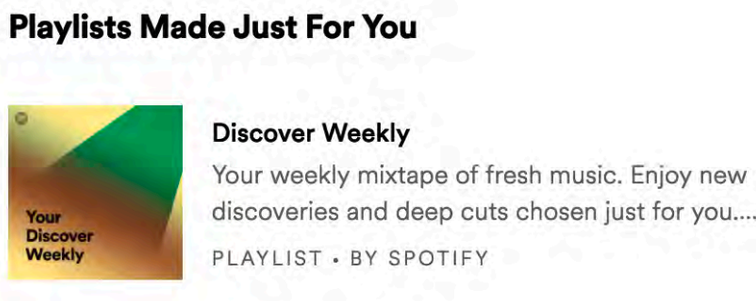
Playlists Made Just For You

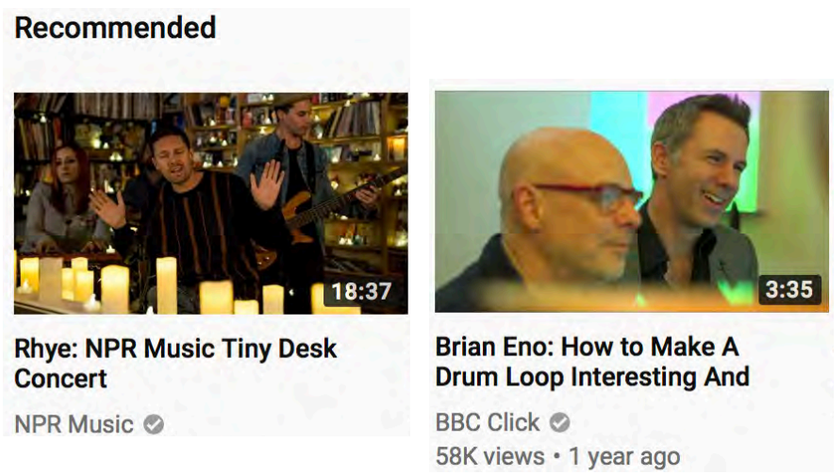
Discover Weekly

Your weekly mixtape of fresh music. Enjoy new discoveries and deep cuts chosen just for you....

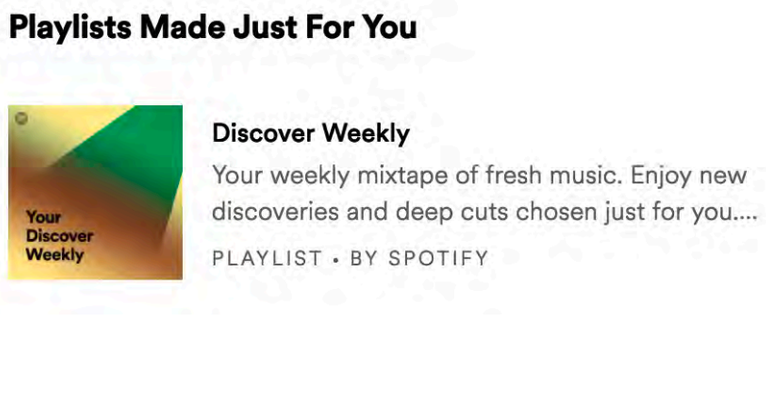
PLAYLIST • BY SPOTIFY

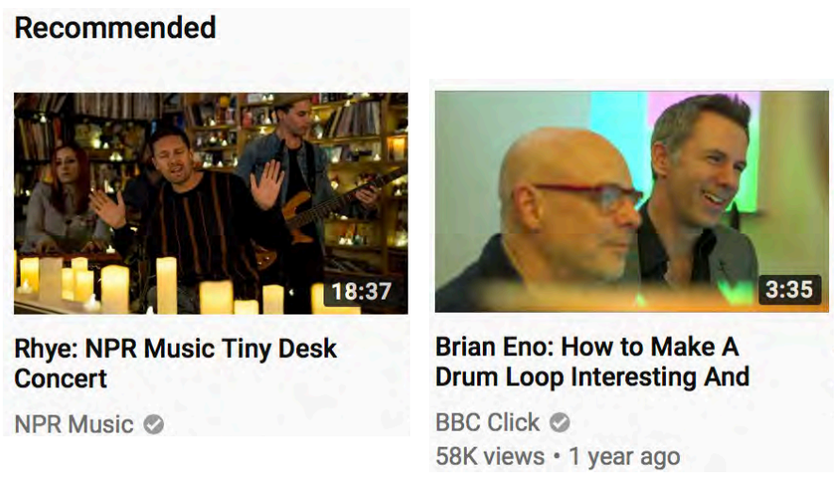
Actionable Intelligence is the study of how to use past data to enhance the future manipulation of a dynamical system



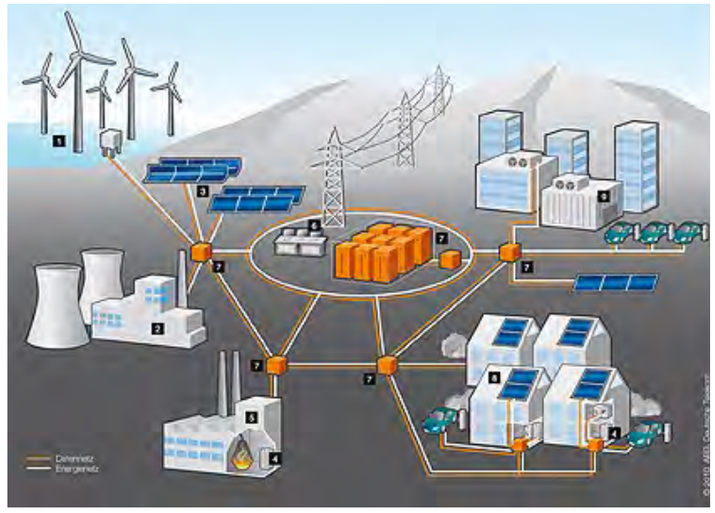


Actionable Intelligence is the study of how to use past data to enhance the future manipulation of a dynamical system





As soon as a machine learning system is unleashed in feedback with humans, that system is an actionable intelligence system, not a machine learning system.



Actionable Intelligence

trustable, scalable, predictable

L4DC 2020 – Learning for Dynamics and Control

UC Berkeley, June 10-11, 2020

Mark your calendars!

Deadline: November 15th, 2019

6-page papers

Formal call for papers will be out shortly!

Local organizers: Ben Recht, Claire Tomlin

L4DC.org