

# Welcome Fall 2020

## Virtually

- Tech Landscape
- Who are We?
- Course Grading Points
- Course Content Outline

# Current Context: Information Age Arrival

- Computers Driving Change
  - Our new virtual world!
- Data Insanely Important
  - Machine Learning/Artificial Intelligence

# Recent Advances

- Smart Speakers
  - Amazon Echo
  - Google Home
  - Apple Siri
- Speech Recognition
  - Speech-To-Text
  - Trigger Word/Wake-Up Word
    - Siri!
  - Speaker ID
  - Speech Synthesis
- DATA

# Areas of Advances

- Self Driving Cars
- Computer Vision
  - Image Classification/Object Recognition
  - Object Detection
  - Image Segmentation
  - Tracking
- DATA

# Areas of Advances

- Natural Language Processing
  - Text Classification
  - Information Retrieval
  - Name Entity Recognition
  - Machine Translation
  - Parsing, Parts of speech tagging
- DATA

# Areas of Advances

- Robotics
  - Perception
  - Motion Planning
  - Control
- DATA

Regulating the internet giants

# The world's most valuable resource is no longer oil, but data

*The data economy demands a new approach to antitrust rules*



[Print edition | Leaders >](#)

May 6th 2017



# Algorithms Feeding on Data

- Data, Data, Data

# Fresno Tech

# Fresno Tech Irma Olguin

Forbes

Billionaires

Innovation

Leadership

Money

Business

Small Business

Lifestyle

Lists

Advisor

Wheels

Featured

More



**FROM RECYCLING  
BOTTLES FOR CASH  
TO TECH CEO**



**MEET THE LATINX  
FOUNDER BUILDING  
A \$100 MILLION  
TECH HUB IN  
CALIFORNIA'S  
GRITTY CENTRAL  
VALLEY**

# Fresno Tech



## Fresno Data Science Meetup

📍 Fresno, CA

👤 342 members · Public group ?

👤 Organized by Antony R.



## Fresno Python User Group

📍 Fresno, CA

👤 354 members · Public group ?

👤 Organized by Derek P. and 3 others

Share: [f](#) [t](#) [in](#)

# Cloud Notebooks

01\_mascots.ipynb - Colaboratory

File Edit View Insert Runtime Tools Help

Not all teams have a mascot that can be categorized by biological family. ([Pepperdine is a good example.](#)) But for D1 teams with mascots that can be categorized that way, how many mascots fit into each family?

```
[ ] 1 mascots_q = """
2 SELECT
3   IF(tax_family='Canidae',IF(tax_subspecies='familiaris','Canidae lupus familiaris','Canidae &
4   COUNT(id) AS num_mascots
5 FROM `bigquery-public-data.ncaa_basketball.mascots`
6 WHERE tax_family IS NOT NULL
7 GROUP BY tax_group
8 ORDER BY num_mascots DESC
9 """
10
11 mascots = gbq.read_gbq(query=mascots_q, dialect ='standard', project_id=project_id)
12 #mascots = client.query(mascots_q).to_dataframe()
13 plt.figure(figsize=(10,10))
14 plt.pie(mascots.num_mascots, labels=mascots.tax_group, colors=colors);
```

Biological Family	Approximate Number of Mascots
Hominidae	~39
Felidae	~22
Canidae lupus familiaris	~14
Spheniscidae	~1
Pteropodidae	~1
Antechinidae	~1
Aotidae	~1
Castoridae	~1

47.53.59/deployment-with-AML

Test the deployed web service

Make direct calls on the service object

```
In [11]: import json
data1 = [1, 2, 5, 9, 4, 27, 5, 'Memorial Day', True, 0, 0.0, 0.0, 65]
data2 = [[1, 3, 10, 15, 4, 27, 7, 'None', False, 0, 2.0, 1.0, 80],
         [1, 2, 5, 9, 4, 27, 5, 'Memorial Day', True, 0, 0.0, 0.0, 65]]
result = aci_service.run(json.dumps(data1))
print('Predictions for data1')
print(result)
result = aci_service.run(json.dumps(data2))
print('Predictions for data2')
print(result)
```

Predictions for data1  
[22.378127578013636]  
Predictions for data2  
[39.714977643580596, 22.378127578013636]

Consume the Deployed Web Service

Execute the code below to consume the deployed webservice over HTTP.

```
In [12]: import requests
url = aci_service.scoring_uri
print('ACI Service: {} scoring URI is: {}'.format(service_name, url))
headers = {'Content-Type': 'application/json'}

response = requests.post(url, json.dumps(data1), headers=headers)
print('Predictions for data1')
print(response.text)
response = requests.post(url, json.dumps(data2), headers=headers)
print('Predictions for data2')
print(response.text)
```

ACI Service: nyc-taxi-srv scoring URI is: http://84eb2566-cfbf-4e91-88f5-22c43d87f2e2.southcentralus.azurecontainer.io/score  
Predictions for data1  
"[22.378127578013636]"  
Predictions for data2  
"[39.714977643580596, 22.378127578013636]"

# Salesforce Tower San Francisco

- Tallest Building in SF when completed in 2018.

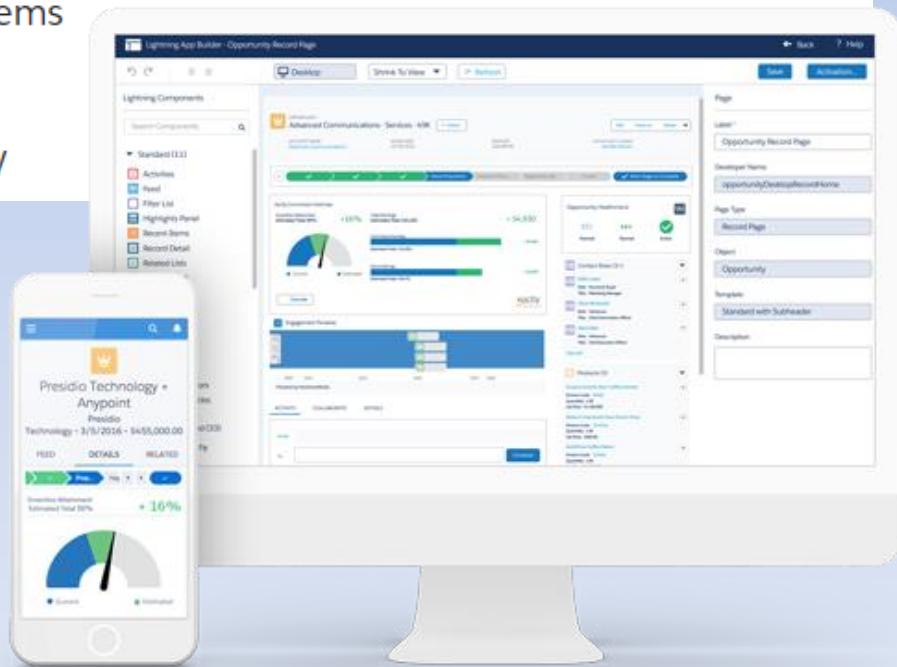




# Connect to your customers in a whole new way with the world's #1 CRM platform.

Integrate every part of your company that interacts with customers – including marketing, sales, service, and more – onto one CRM platform, Salesforce Customer 360 :

- ✓ Gives teams a shared view of every customer
- ✓ Automates workflows using artificial intelligence
- ✓ Integrates easily with existing data and systems
- ✓ Keeps your business data safe and secure
- ✓ Customizes and scales for any size company



# Professional Certification

- Google Cloud

- <https://cloud.google.com/certification/machine-learning-engineer>
- Professional Machine Learning Engineer

The screenshot shows the Google Cloud website with the navigation bar at the top. Below the navigation, there's a sidebar on the left with links for the certification, including 'Exam overview', 'Exam guide', 'Certification home', 'Certification FAQs', 'Exam terms & conditions', 'Register', 'Certified directory', and 'Google Cloud training'. The main content area features the title 'Professional Machine Learning Engineer BETA' in large bold letters. A detailed description follows: 'A Professional Machine Learning Engineer designs, builds, and productionizes ML models to solve business challenges using Google Cloud technologies and knowledge of proven ML models and techniques. The ML Engineer is proficient in all aspects of model architecture, data pipeline interaction, and metrics interpretation and needs familiarity with application development, infrastructure management, data engineering, and security.' Below this, a section titled 'The Professional Machine Learning Engineer BETA exam assesses your ability to:' lists several competencies with checkmarks: 'Frame ML problems', 'Architect ML solutions', 'Prepare and process data', 'Develop ML models', 'Automate & orchestrate ML pipelines', and 'Monitor, optimize, and maintain ML solutions'.

# Professional Certification

- Amazon
  - Amazon Web Services

## AWS Certified Machine Learning – Specialty

Validate your ability to build, train, tune, and deploy machine learning models using the AWS Cloud.

[Schedule an exam](#)

**AWS CERTIFICATION EXAM READINESS TRAINING | AUGUST 26**

**Preparing to Take the AWS Certified SysOps Administrator — Associate Certification Exam?**

Join our free virtual exam readiness webinar led by AWS-accredited instructors.

[Register now »](#)

The AWS Certified Machine Learning - Specialty certification is intended for individuals who perform a development or data science role. It validates a candidate's ability to design, implement, deploy, and maintain machine learning (ML) solutions for given business problems.

**Abilities Validated by the Certification**

- Select and justify the appropriate ML approach for a given business problem
- Identify appropriate AWS services to implement ML solutions
- Design and implement scalable, cost-optimized, reliable, and secure ML solutions



AWS Certified Machine Learning - Specialty (0:37)

# Qwiklabs

## Baseline: Data, ML, AI

Introductory   6 Steps   5 hours   10 Credits

Big data, machine learning, and artificial intelligence are today's hot computing topics, but these fields are quite specialized and introductory material is hard to come by. Fortunately, Google Cloud provides user-friendly services in these areas and Qwiklabs has you covered with this introductory-level quest, so you can take your first steps with tools like Big Query, Cloud Speech API, and AI Platform. Want extra help? [1-minute videos walk you through key concepts for each lab.](#)

Data

Machine Learning

### Prerequisites:

This Quest builds on a basic understanding of Google Cloud services. It is recommended that the student have earned a Badge by completing the hands-on labs in the [Getting Started - Create and Manage Cloud Resources Quest](#) before beginning.

### Quest Complete!

Congrats! You completed this quest and earned a badge. Become a cloud expert and start another.





# Who I Am

- Computer Science
- California State University, Fresno

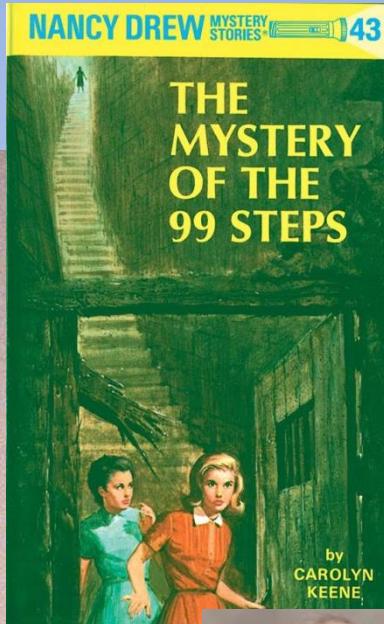
## David Ruby, Lecturer

- From Fresno – 10/62
- Graduated Fresno State – 5/84
- Teaching at Fresno State – 1/13
- First Generation College Grad

# Key Idea: Abstraction

- Abstract thinking critical.
- Developing ability with abstractions important.
- My story involves abstractions.

# My Story



- Family Memories

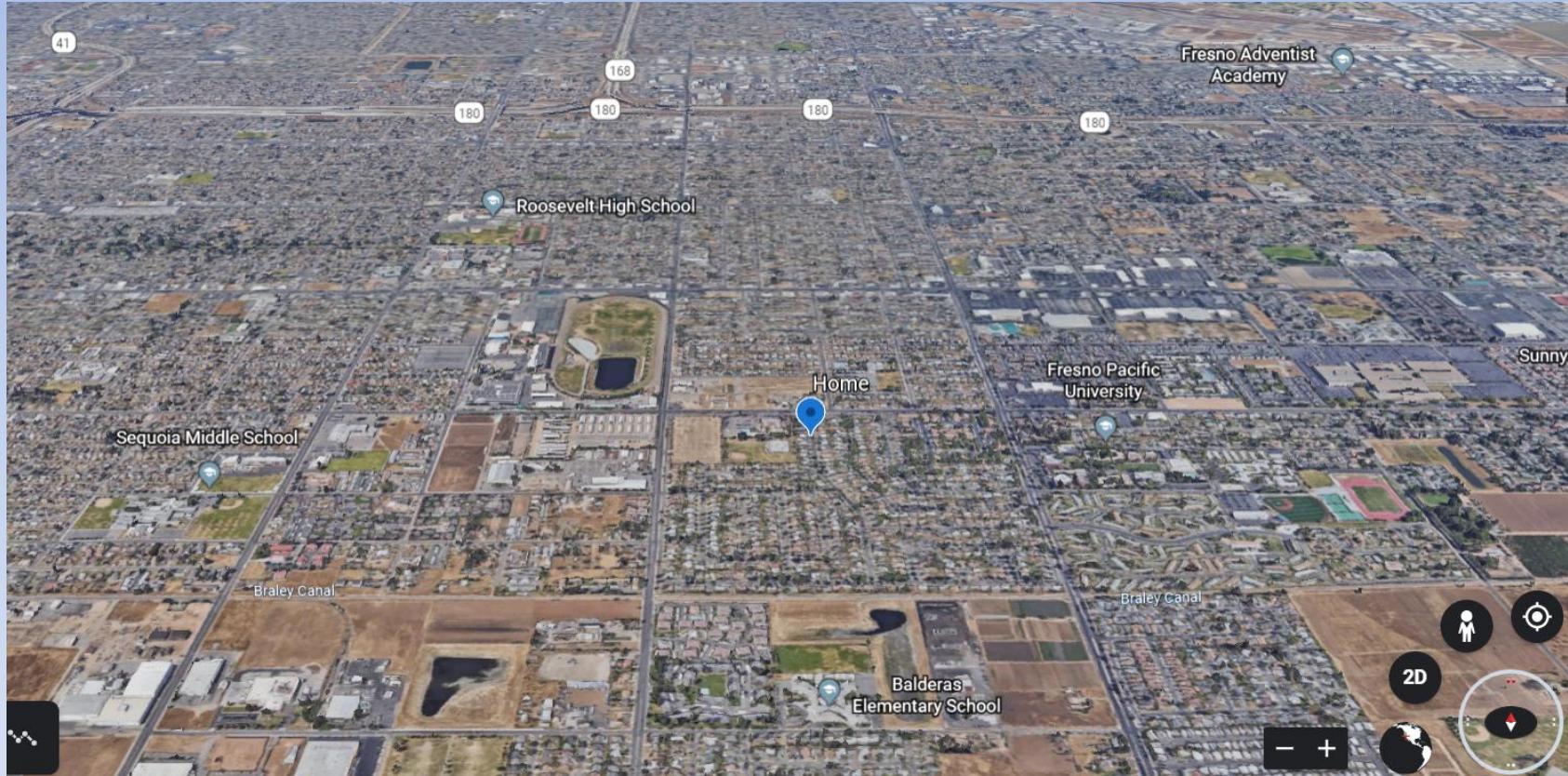


Interest  
In  
Puzzles

# Mother & Father grew up in Fresno



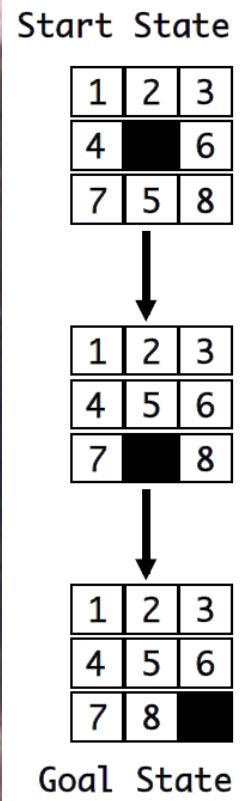
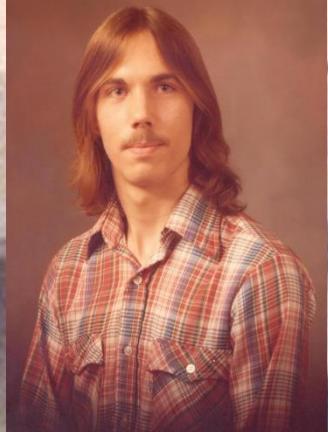
# Father Floyd Fresno Career Custodian Ending @ Sequoia Middle School



- Uncle Walt career custodian @ Roosevelt High School

# My Memories..

- Home Hedges Maze.
- Also – First time w/ Sliding Tile Puzzle



# Teaching & Learning

- How should I teach?
  - Much research going on.
- How have I learned?
  - Initial approach at teaching.

# PhD Thesis: How did you solve it???

File: main.cpp

```
38 int Move(string& board, int dir){  
39     int row, col;  
40     LocBlank(board, row, col);  
41     if ((dir==1 || dir == -3) && row > 0){  
42         Swap(board, row, col, row-1, col);  
43     }  
44     else if ((dir == 2 || dir == -4) && col <  
45         Swap(board, row, col, row, col+1);  
46     }  
47     else if ((dir == 3 || dir == -1) && row <  
48         Swap(board, row, col, row+1, col);  
49     }  
50     else if ((dir == 4 || dir == -2) && col >  
51         Swap(board, row, col, row, col-1);  
52     }  
53     else {  
54         return -1;  
55     }  
56     return 0;  
57 }
```

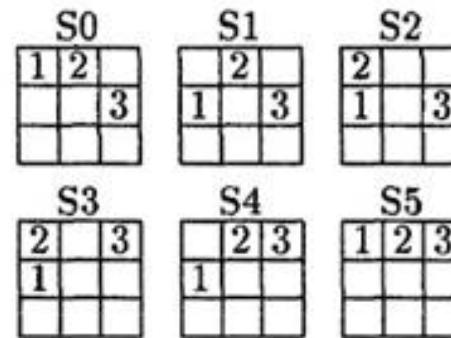


Figure 2: Steppingstones from Memory

## ■ AAAI DIGITAL LIBRARY

*AAAI relies on your generous support through membership and donations. If you find these resources useful, we would be grateful for your support.*

### AI Magazine Articles

### Conference Proceedings Papers

### Funding Tutorial

### Presidential Addresses

### Policy Reports

### Symposia Papers

### Workshop Papers

## ■ OTHER LINKS

### AAAI Home Page

## ■ PROCEEDINGS OF THE NINTH NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE

*Sponsored by the Association for the Advancement of Artificial Intelligence*

Thomas L. Dean and Kathleen McKeown, Program Cochairs

July 14-19, 1991, Anaheim, California. Published by The AAAI Press, Menlo Park, California. This proceedings is also available in book and CD format.

**Please Note:** Abstracts are linked to individual titles, and will appear in a separate browser window. Full-text versions of the papers are linked to the abstract text. Access to full text may be restricted to AAAI members. PDF file sizes may be large!

### Contents

#### Volume One

Preface

Conference Organization

Best Paper Award

#### Case-Based Reasoning

##### Case-Based Reasoning

Rules and Precedents as Complementary Warra  
L. Karl Branting, University of Wyoming; and Bi

An Indexing Vocabulary for Case-Based Explain:  
David B. Leake, Indiana University / 10

#### Volume Two

##### Learning

##### Learning Search Control

Synthesizing UNIX Shell Scripts Using Derivational Analogy: An Empirical Assessment  
Sanjay Bhansali and Mehdi T. Harandi, University of Illinois at Urbana-Champaign / 521

##### SteppingStone: An Empirical and Analytical Evaluation

David Ruby and Dennis Kibler, University of California, Irvine / 527

##### STATIC: A Problem-Space Compiler for PRODIGY

Oren Etzioni, University of Washington / 533

##### Integrating Abstraction and Explanation-Based Learning in PRODIGY

Craig A. Knoblock, Carnegie Mellon University; Steven Minton, Sterling Federal Systems; and Oren Etzioni, University of Washington / 541

##### Learning Theory and MDL

##### Learning with Many Irrelevant Features

Hussein Almuallim and Thomas G. Dietterich, Oregon State University / 547

##### Analyses of Instance-Based Learning Algorithms

Marc K. Albert, University of California, Irvine; and David W. Aha, The Turing Institute / 553

##### Regularity and Structure

Alexander Botta, New York University / 559

##### A Minimal Encoding Approach to Feature Discovery

Mark Derthick, MCC / 565

##### Learning Connectionist Representations

Error-Correcting Output Codes: A General Method for Improving Multiclass Inductive Learning Programs  
Thomas G. Dietterich and Ghulum Bakiri, Oregon State University / 572

Analysis of the Internal Representations in Neural Networks for Machine Intelligence  
Lai-Wan Chan, The Chinese University of Hong Kong / 578

# Computer Science / Memories

- Puzzles
- Abstractions
- Memories

<b>Assignment/Examination/Presentation</b>	<b>Points</b>
In-Class/Presentations	100
Quizzes	100
Qwiklabs	200
Project	200
Midterm	150
Final	250

# Attendance

- Class is constructed with students & instructor.

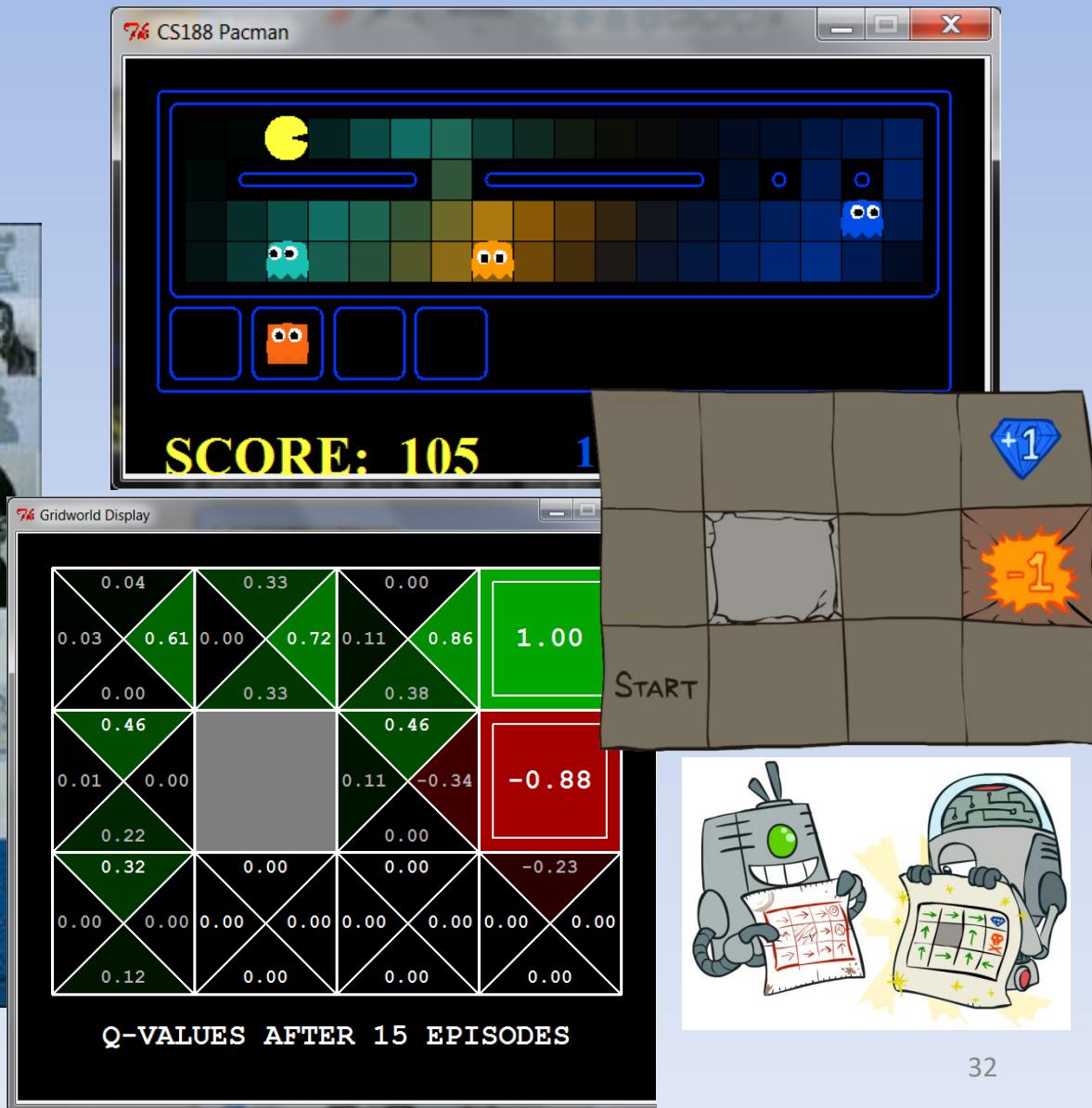
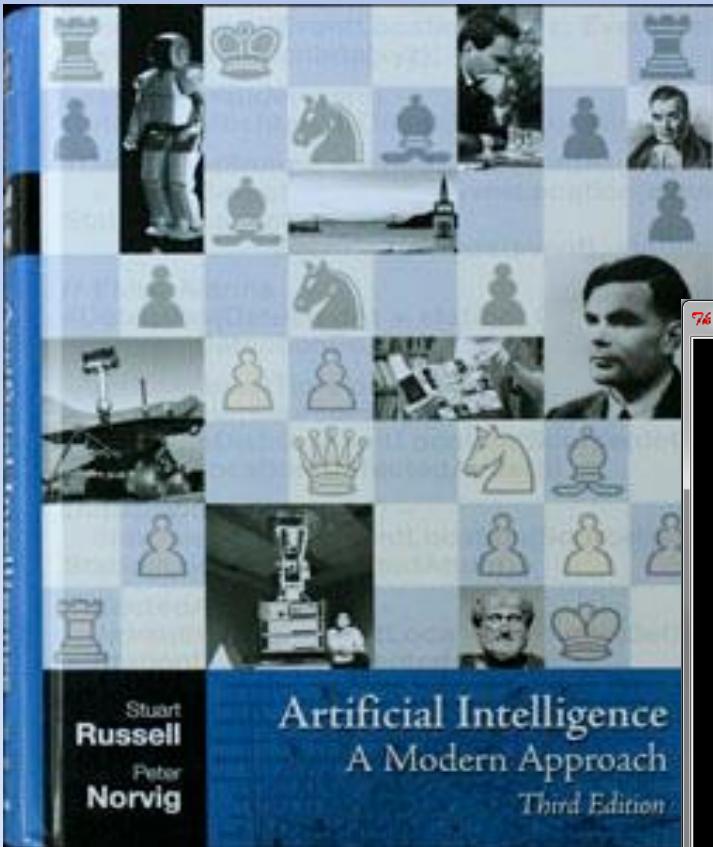
# Presentations

# Quizzes

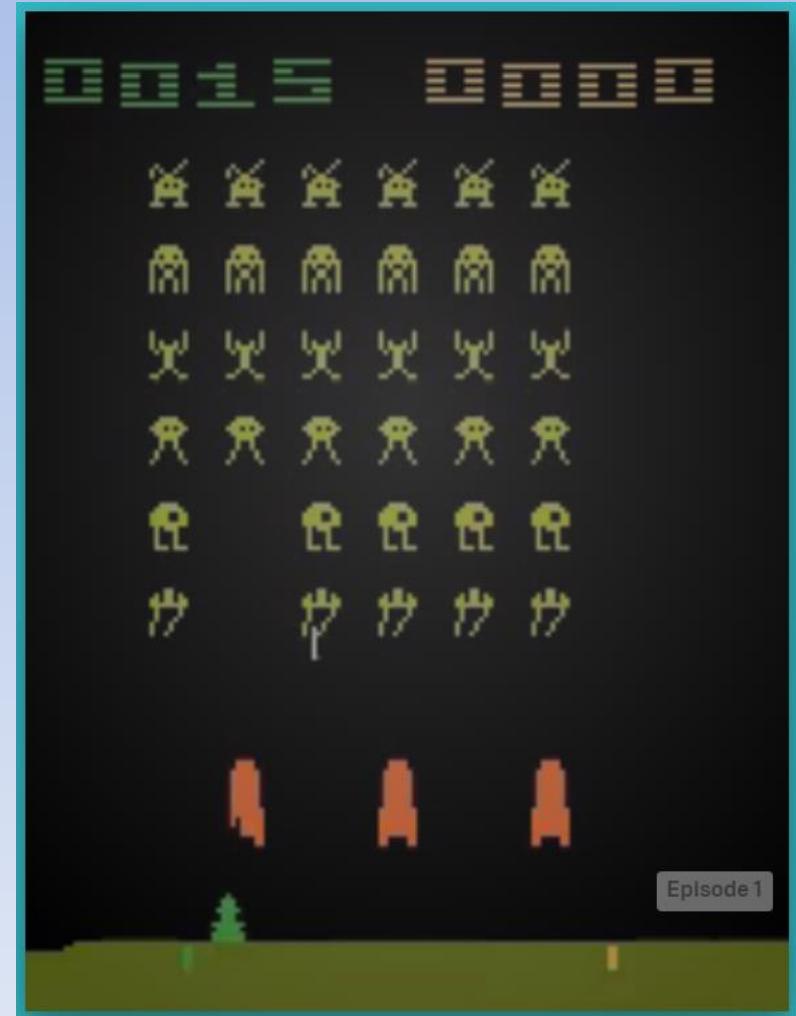
# CSci 166: Principles of AI

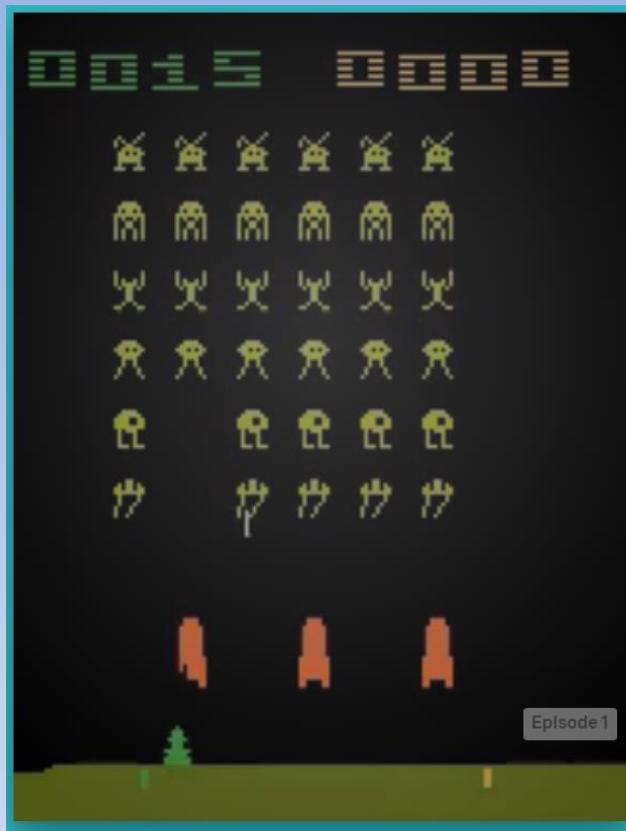
## AI – Part 2

- AI



# AI w/ Games



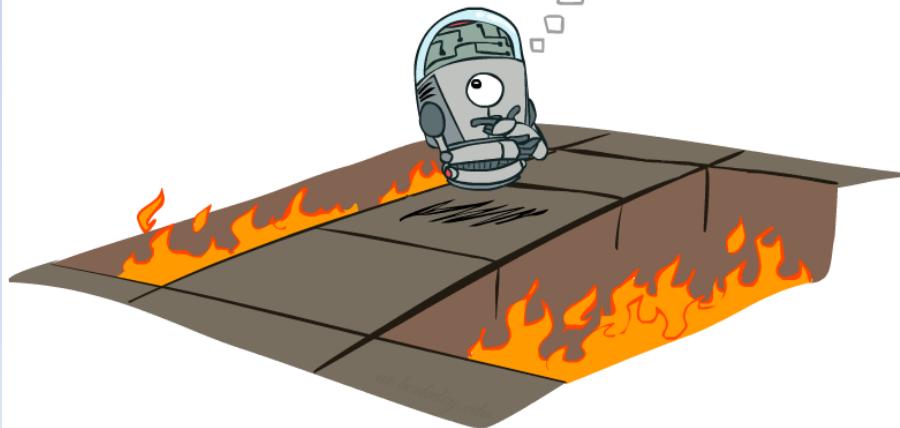
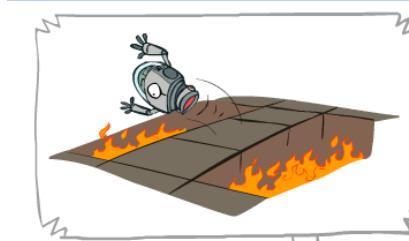
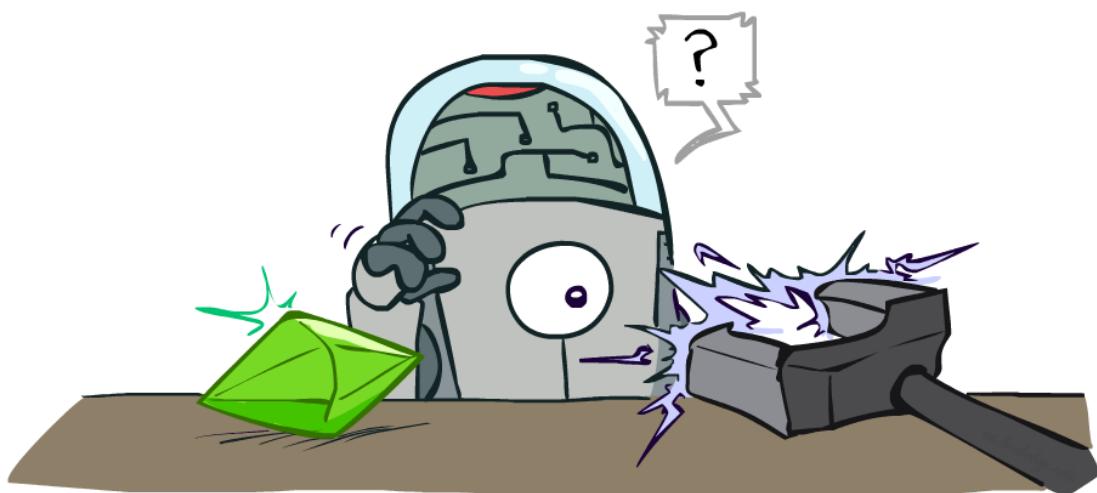


# AI w/ Games

- How can a computer learn to play video games?
- Big rewards require learning sequence of actions.

# Learning w/ MDP (Chapter 21)

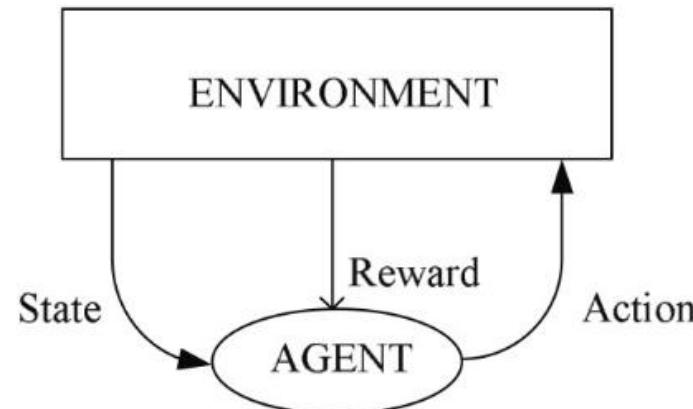
## Reinforcement Learning



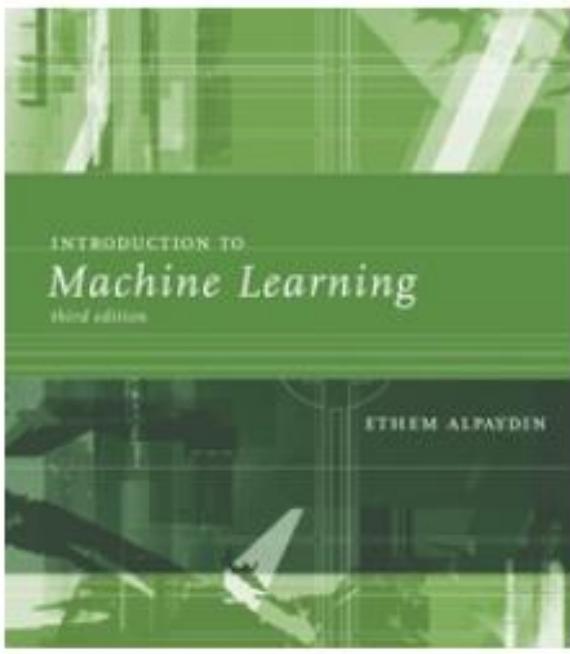
# Reinforcement Learning

518

18 Reinforcement Learning



**Figure 18.1** The agent interacts with an environment. At any state of the environment, the agent takes an action that changes the state and returns a reward.



# Reinforcement Learning

O'REILLY®

Hands-On  
Machine Learning  
with Scikit-Learn  
& TensorFlow

CONCEPTS, TOOLS, AND TECHNIQUES  
TO BUILD INTELLIGENT SYSTEMS



Aurélien Géron

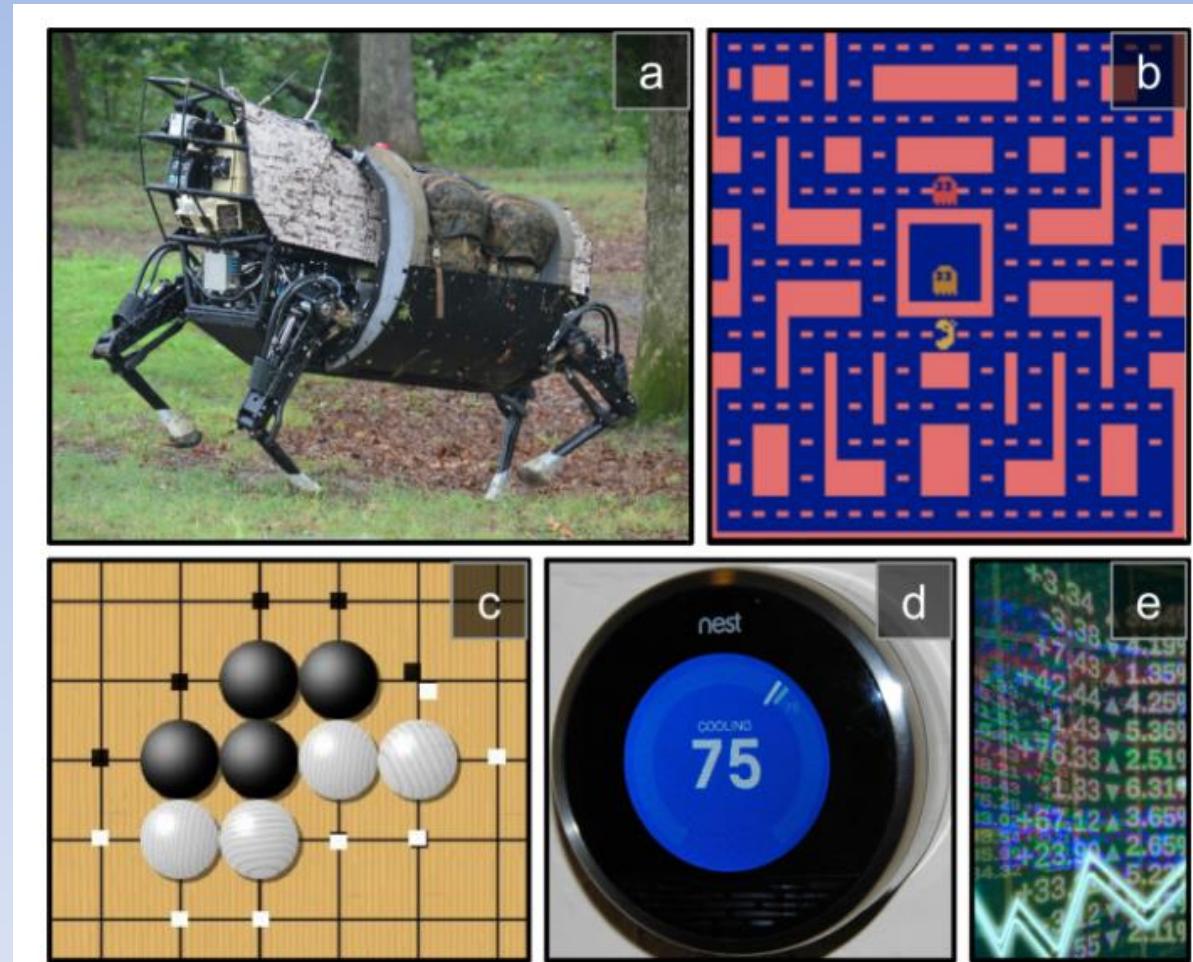
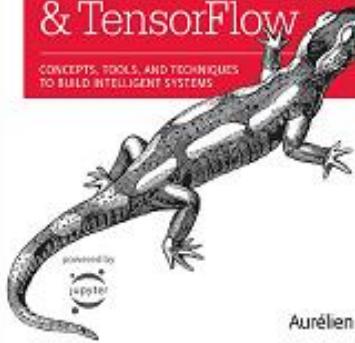


Figure 16-1. Reinforcement Learning examples: (a) walking robot, (b) Ms. Pac-Man, (c) Go player, (d) thermostat, (e) automatic trader<sup>5</sup>

# Hands-On Machine Learning with Scikit-Learn & TensorFlow

CONCEPTS, TOOLS, AND TECHNIQUES  
TO BUILD INTELLIGENT SYSTEMS

Aurélien Géron

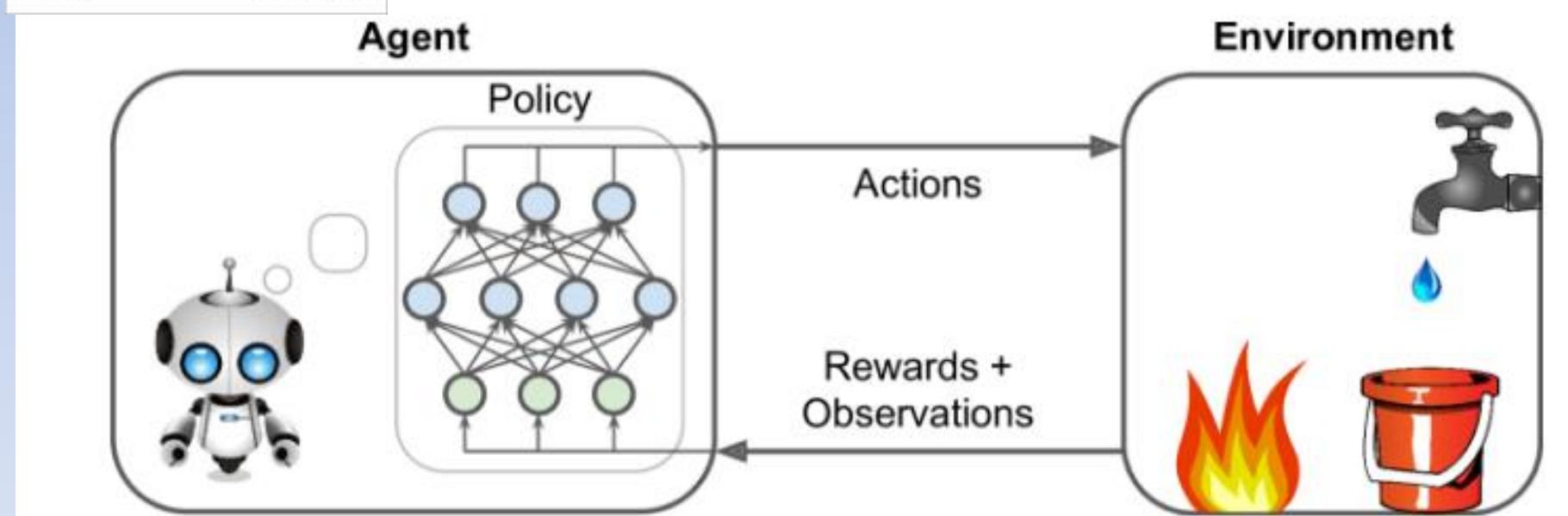


Figure 16-2. Reinforcement Learning using a neural network policy

# Reinforcement Learning

The screenshot shows the AWS training and certification platform. At the top, there's a navigation bar with links for Dashboard, Learning Library, Certification, Support, and Partner Training. A search bar is also present. The main content area features a green header for the course 'AWS DeepRacer: Driven by Reinforcement Learning'. The header includes an 'E-LEARNING' icon, the course title, and duration information (90 MINUTES, ENGLISH). Below the header, there's a section titled 'ABOUT' with a description of the course's purpose: to provide hands-on experience with reinforcement learning (RL) by helping users build, train, and deploy models in AWS DeepRacer. It also mentions the AWS DeepRacer League. Further down, there's a section about the course structure, stating it starts with orienting users to AWS DeepRacer before diving into RL, providing opportunities to build, train, and evaluate their own RL models, and then deploying them to a physical car. Finally, it notes that by the end of the course, users will have an optimized RL model for the AWS DeepRacer League. To the right of the main content, there's a sidebar with a 'IN PROGRESS' status message indicating the course was started on Aug 13, 2020. It has a 'RESUME >' button and a refresh/circular arrow icon. Below this, there's a section titled 'LANGUAGES AVAILABLE' with a link to register for Japanese.

aws training and certification

Dashboard Learning Library Certification Support Partner Training

Search for training

E-LEARNING

## AWS DeepRacer: Driven by Reinforcement Learning

90 MINUTES ENGLISH

### ABOUT

This course is designed to give you hands-on experience with reinforcement learning (RL) by helping you build, train and deploy models in AWS DeepRacer so you can compete in the official AWS DeepRacer League.

The course starts off by orienting you to AWS DeepRacer before diving into RL. You'll have several opportunities to build, train, and evaluate your own RL models, and then deploy them to the physical car.

By the end of the course, you'll have your own optimized RL model that you can enter into a virtual or physical circuit of the AWS DeepRacer League.

IN PROGRESS  
You started this course on Aug 13, 2020

RESUME >

LANGUAGES AVAILABLE

This training is also available in other languages. Click the links below to register for this training in your preferred language.

[Japanese](#)

# Reinforcement Learning

AWS DeepRacer

Contact Sales   Support ▾   English ▾   My Account ▾

Products   Solutions   Pricing   Documentation   Learn   Partner Network   AWS Marketplace   Customer Enablement   Events   Explore More   Q

Overview   League   F1 Event   Schedule and Standings   Getting Started   Racing Tips   Pricing   FAQs

## Developers, start your engines

Developers of all skill levels can get hands on with machine learning through a cloud based 3D racing simulator, fully autonomous 1/18th scale race car driven by reinforcement learning, and global racing league.

Build a model   Buy Now



**FEATURED SERVICE**

**Compete in the AWS DeepComposer Chartbusters Challenge**

Create your own AI-generated tunes and top the musical charts by generating original compositions.

Start composing ➤

# Reinforcement Learning

The screenshot shows the AWS DeepRacer League homepage. At the top, there's a navigation bar with links for Products, Solutions, Pricing, Documentation, Learn, Partner Network, AWS Marketplace, Customer Enablement, Events, Explore More, and a search bar. The main title "AWS DeepRacer League" is prominently displayed above a banner image of a race track at night. Below the banner, a large orange button says "Race online". To the right of the banner is a large, shiny silver and purple logo for "DEEPRACER LEAGUE" featuring a racing helmet. A text block welcomes visitors to the world's first global autonomous racing league, driven by reinforcement learning, and encourages them to get on the track to compete online in monthly Virtual Circuit races or at Summit Circuit race events worldwide. Below this, a "FEATURED SERVICE" section promotes the "AWS DeepComposer Chartbusters Challenge", which allows users to create their own AI-generated tunes and top the musical charts by generating original compositions. An orange button labeled "Start composing >" is shown. The bottom section, titled "Virtual Circuit", describes how to race online via the AWS DeepRacer console. It mentions joining from anywhere, competing in time trial races, and taking on challenges like head-to-head racing. It highlights the AWS Free Tier, which covers up to 10 hours of training. To the right of this text is a video player showing a man in a blue shirt sitting on a couch, with shelves of racing memorabilia in the background. The video player has a play button in the center.

aws

Contact Sales Support English My Account

Products Solutions Pricing Documentation Learn Partner Network AWS Marketplace Customer Enablement Events Explore More Q

AWS DeepRacer Overview League F1 Event Schedule and Standings Getting Started Racing Tips Pricing FAQs

# AWS DeepRacer League

Welcome to the world's first global autonomous racing league, driven by reinforcement learning. It's time to race for prizes and glory, and a chance to advance to the AWS DeepRacer Championship. Get on the track to compete online in monthly Virtual Circuit races or at Summit Circuit race events worldwide.

Race online

**FEATURED SERVICE**

**Compete in the AWS DeepComposer Chartbusters Challenge**

Create your own AI-generated tunes and top the musical charts by generating original compositions.

Start composing >

## Virtual Circuit

Race online via the AWS DeepRacer console

Join the AWS DeepRacer League from anywhere in the world. Compete in time trial races and take on new challenges such as head-to-head racing via the AWS DeepRacer console. New tracks will be released each month, with 24 chances to win an expenses paid trip to re:Invent 2020! With AWS Free Tier we've got you covered for up to 10 hours of training, so you can enter the AWS DeepRacer League at no cost to you.

Play

AWS DeepRacer League F1 Pre-Am Event

# Project

[Environments](#) [Documentation](#)

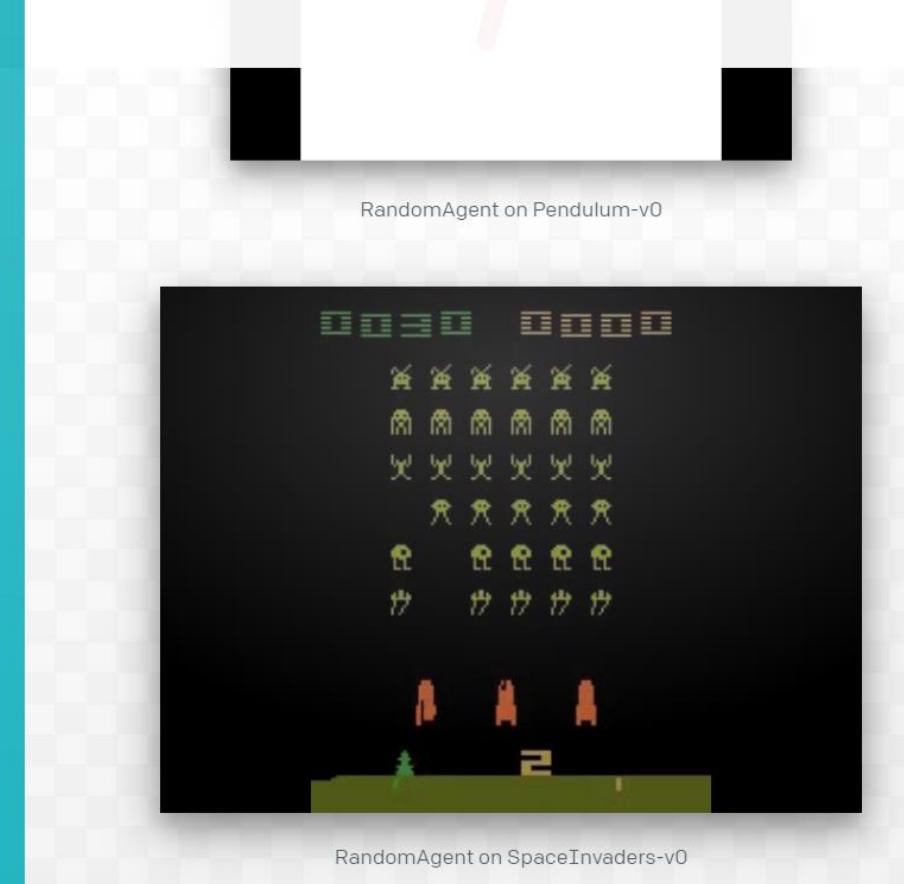


## Gym

Gym is a toolkit for developing and comparing reinforcement learning algorithms. It supports teaching agents everything from walking to playing games like Pong or Pinball.

[View documentation >](#)

[View on GitHub >](#)



# OpenAi.Com

RESEARCH SYSTEMS

ABOUT BLOG

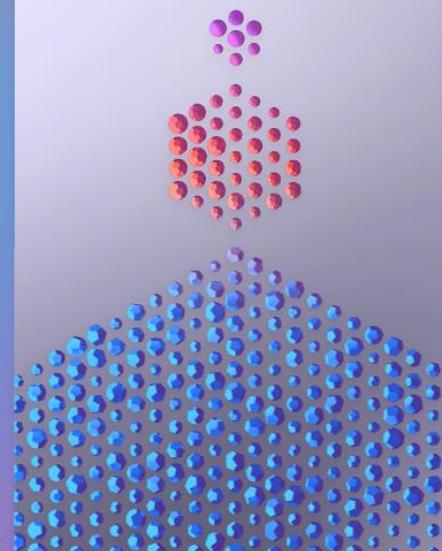


Discovering and enacting  
the path to safe artificial  
general intelligence.

RESEARCH SYSTEMS

Improving Language  
Understanding with  
Unsupervised Learning

JUN 11  
2018



[VIEW RESEARCH >](#)

# Gym w/ Google Colab

There are [four methods from Gym](#) that are going to be useful to us in order to save the gumdrop.

- make allows us to build the environment or game that we can pass actions to
- reset will reset an environment to it's starting configuration and return the state of the player
- render displays the environment for human eyes
- step takes an action and returns the player's next state.

Let's make, reset, and render the game. The output is an ANSI string with the following characters:

- S for starting point
- F for frozen
- H for hole
- G for goal
- A red square indicates the current position

**Note:** Restart the kernel if the above libraries needed to be installed

```
In [3]: import gym
import numpy as np
import random
env = gym.make('FrozenLake-v0', is_slippery=False)
state = env.reset()
env.render()

SFFF
FHFH
FFFH
HFFG
```

If we print the state we'll get 0. This is telling us which square we're in. Each square is labeled from 0 to 15 from left to right, top to bottom, like this:

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

```
In [4]: print(state)

0
```

# Reinforcement Learning: But First .... MDP

- Reinforcement Learning is built on a problem solving model.
- First look at model.

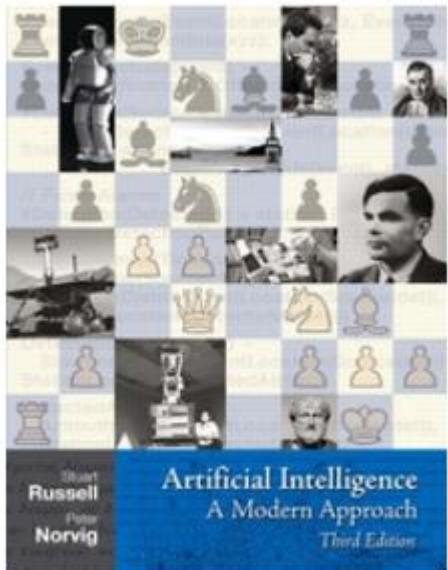
# Markov Decision Process (MDP)

## Chapter 17

17

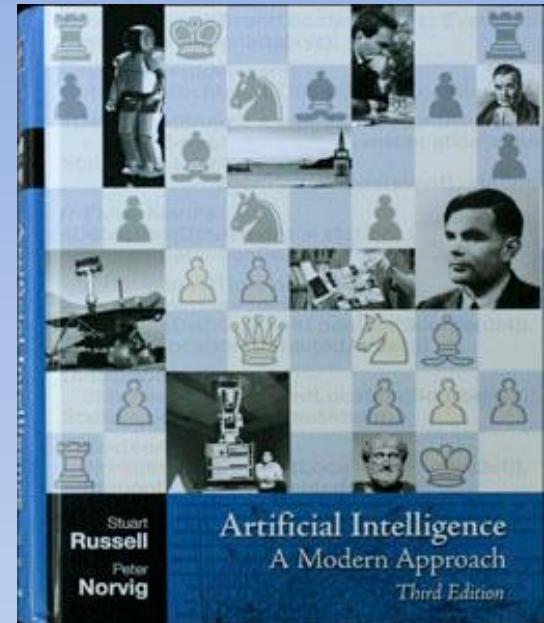
MAKING COMPLEX  
DECISIONS

*In which we examine methods for deciding what to do today, given that we may decide again tomorrow.*



# But First ...

# Dealing w/ Uncertainty

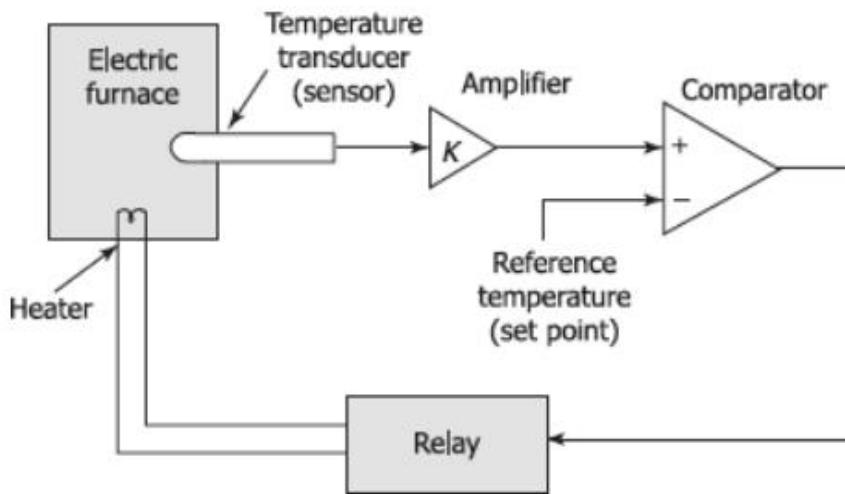


## IV Uncertain knowledge and reasoning

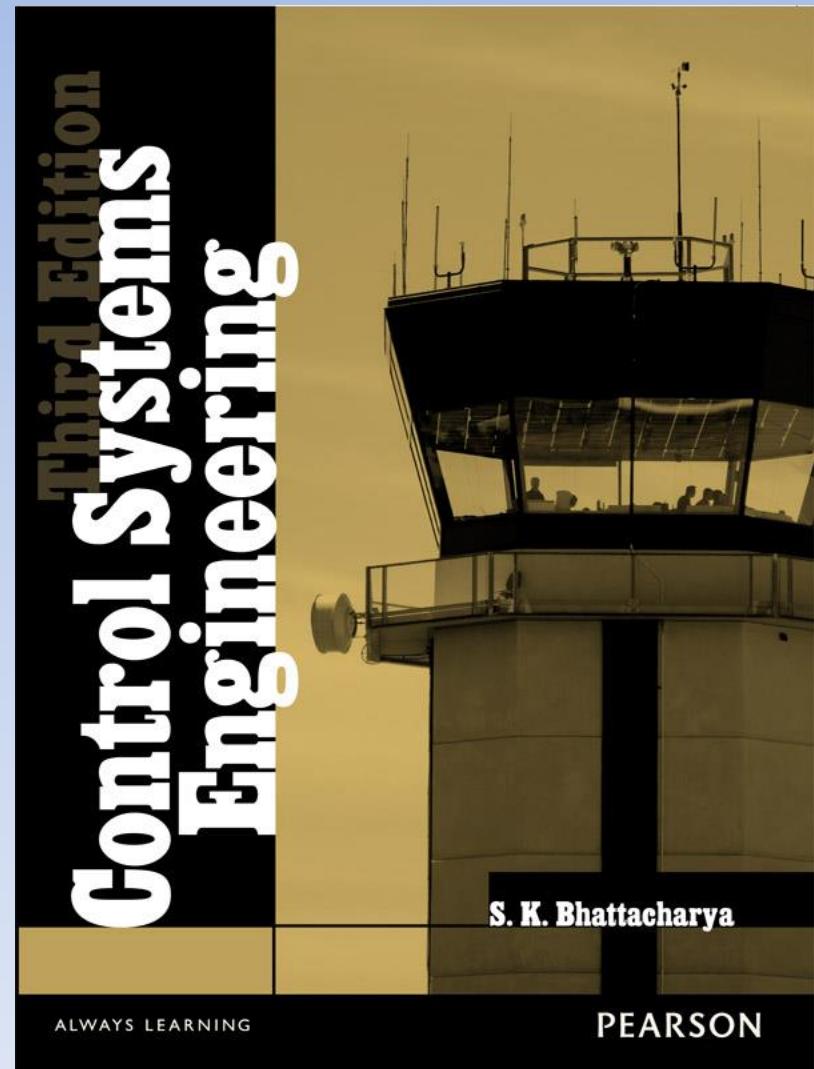
<b>13 Quantifying Uncertainty</b>	<b>480</b>
13.1 Acting under Uncertainty . . . . .	480
13.2 Basic Probability Notation . . . . .	483
13.3 Inference Using Full Joint Distributions . . . . .	490
13.4 Independence . . . . .	494
13.5 Bayes' Rule and Its Use . . . . .	495
13.6 The Wumpus World Revisited . . . . .	499
13.7 Summary, Bibliographical and Historical Notes, Exercises . . . . .	503



# MDP w/ Control Systems



**Fig. 1.10** Temperature control system



# MDP w/ Operations Research



## Table of Contents

Table of Contents	
<input type="text"/> Search contents	
	Introduction to Operations Research 10th Edition Frederick Hillier
Cover	Introduction to Operations Research
v	Title
vi	Copyright
xi	TABLE OF CONTENTS

## CHAPTER 19

### Markov Decision Processes

# The Bayesian Network (Chapter 14)

- Exploit Islands of Tractability in High Dimensional Space Probability Distributions
  - Worst Case is Intractable
  - Real World Frequently NOT Worst Case
  - Efficiently Exploit properties in Real World Probability Distributions to induce Tractability
- Primary Tool:

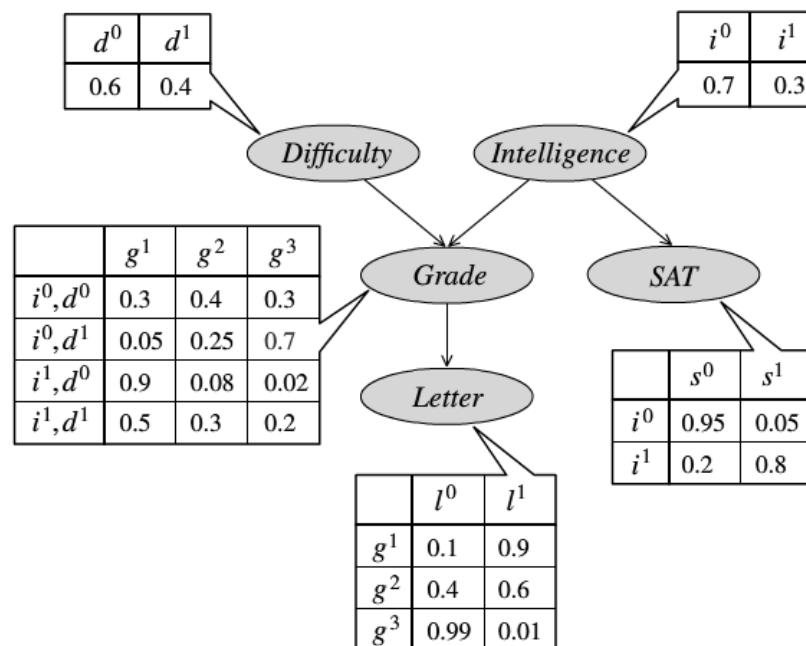
**INDEPENDENCE!**

# Bayesian Networks : (CPD's)

- Each variable is associated with a conditional probability distribution (CPD) that specifies a distribution CPD over the values of X given each possible joint assignment of values to its parents in the model.
- For a node with no parents, the CPD is conditioned on the empty set of variables.

3.2. Bayesian Networks

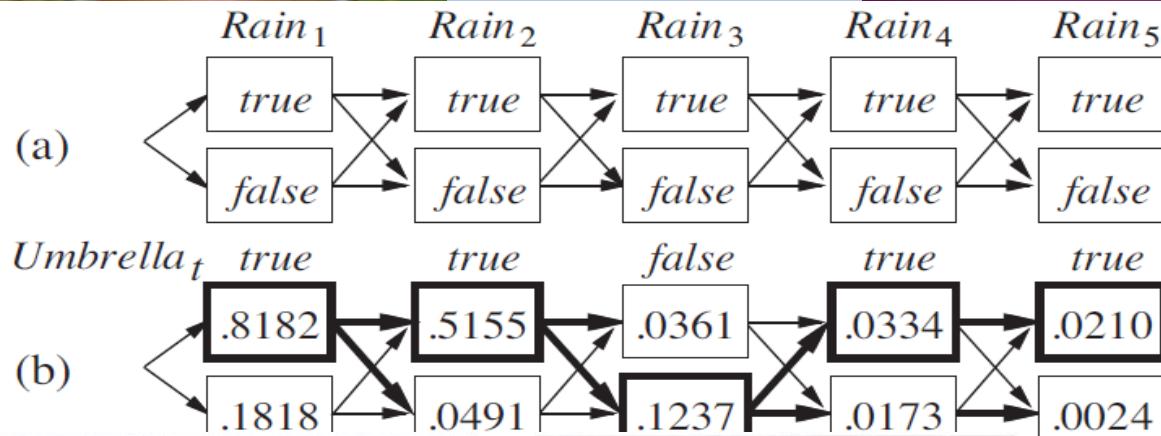
53





# Principles of AI

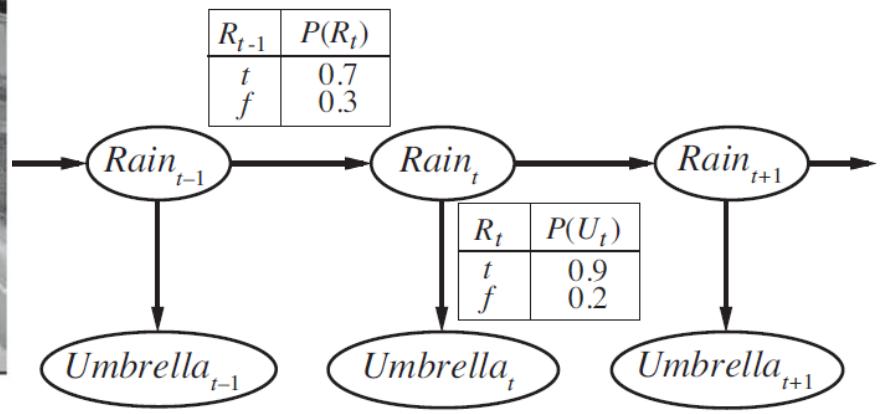
## Chapter 15: Probabilistic Reasoning over Time



(a)



(b)



# Chapter 16:

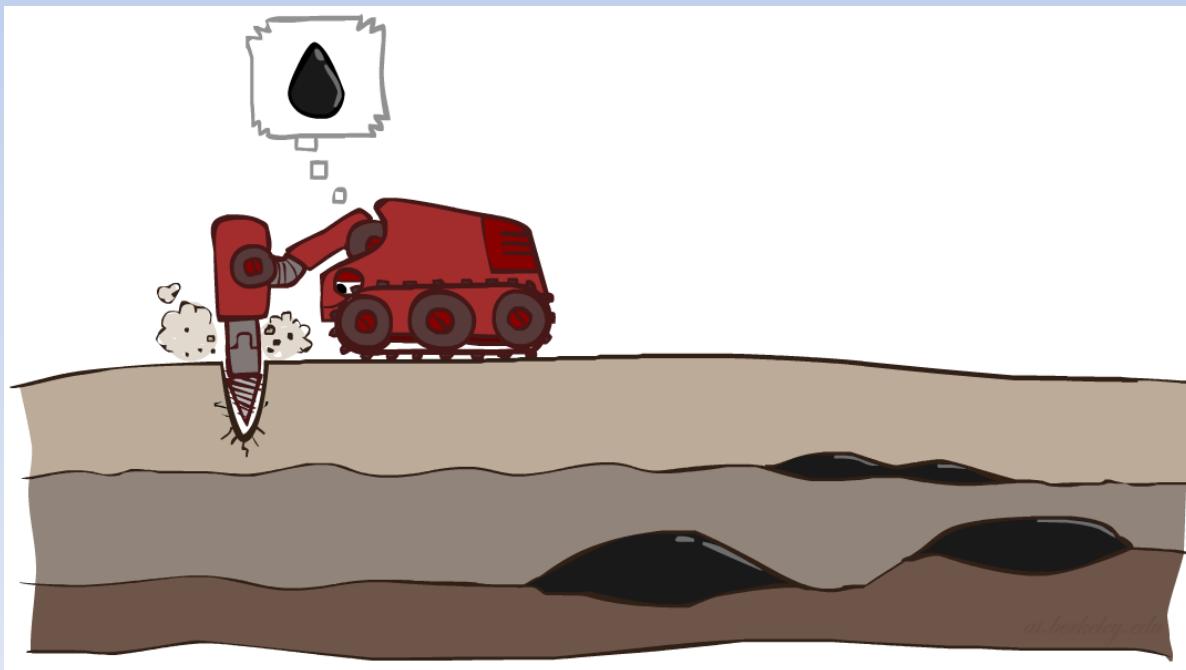
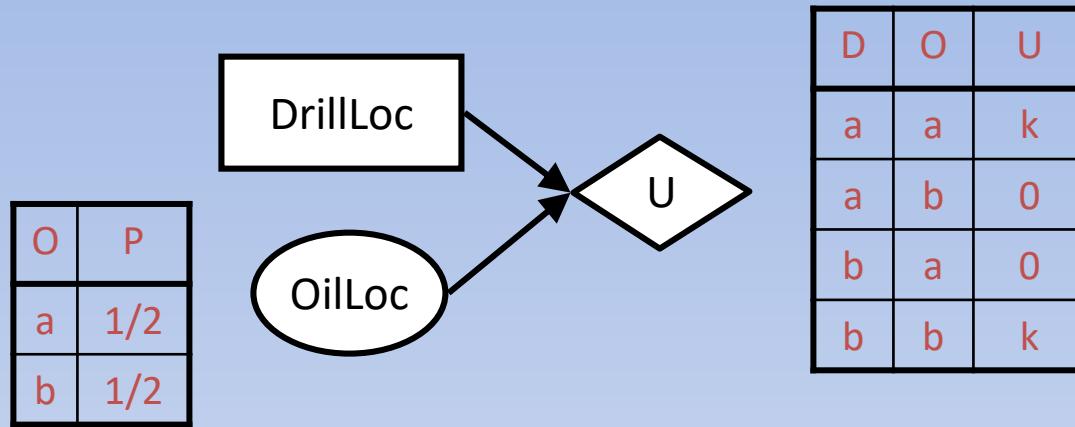
# Making Simple Decisions

## 16 MAKING SIMPLE DECISIONS

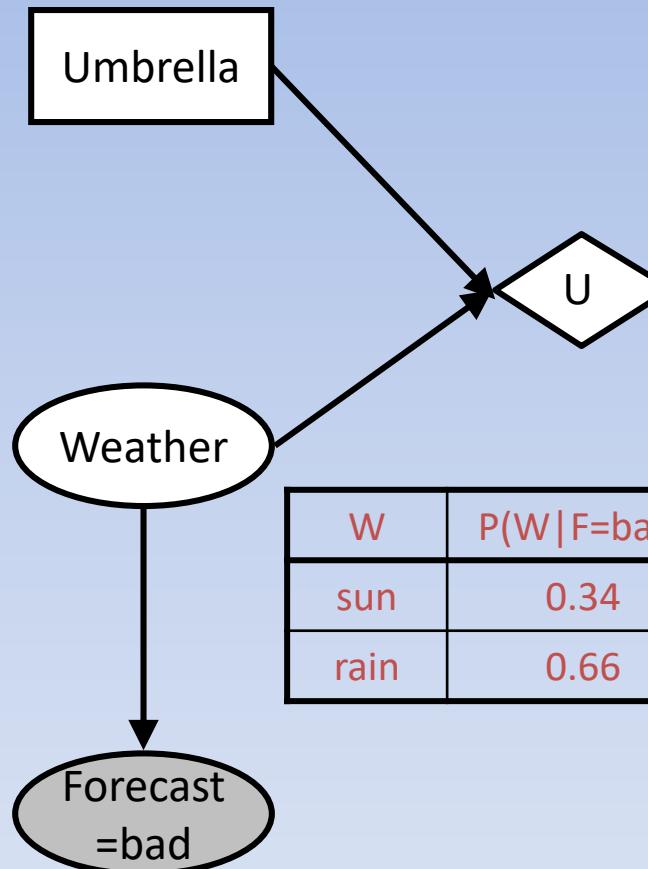
*In which we see how an agent should make decisions so that it gets what it wants—  
on average, at least.*

<b>16 Making Simple Decisions</b>	<b>610</b>
16.1 Combining Beliefs and Desires under Uncertainty . . . . .	610
16.2 The Basis of Utility Theory . . . . .	611
16.3 Utility Functions . . . . .	615
16.4 Multiattribute Utility Functions . . . . .	622
16.5 Decision Networks . . . . .	626
16.6 The Value of Information . . . . .	628
16.7 Decision-Theoretic Expert Systems . . . . .	633
16.8 Summary, Bibliographical and Historical Notes, Exercises . . . . .	636

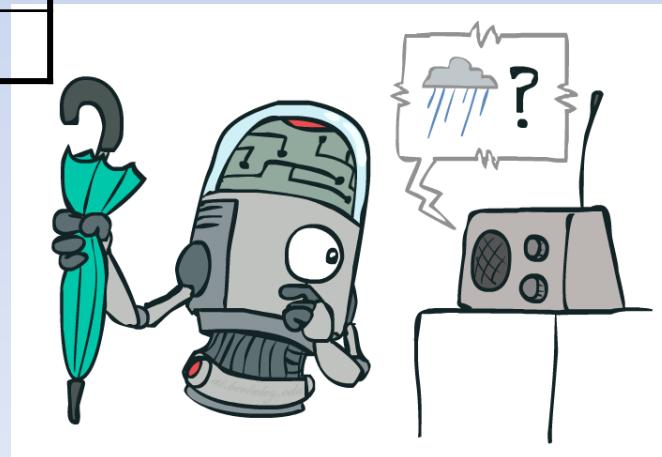
# Value of Information



# Decision Making w/ Uncertainty



A	W	$U(A,W)$
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



# Chapter 20 :

# Learning Probabilistically

- Naïve Bayes

# Chapter 22: Natural Language

## • N-Gram Model

ONE TWO ONE TWO THREE TWO THREE

"TWO" has the following predictions:

THREE: 0.666, ONE: 0.333

"THREE" occurred 2 times after a "TWO" and "ONE" occurred 1 time after a "TWO", for a total of 3 occurrences after a "TWO".

Your program will run against the following text, ignoring all punctuation i.e. Hardcode it into your program:

Mary had a little lamb its fleece was white as snow;  
And everywhere that Mary went the lamb was sure to go.  
was against the rule;  
see a lamb at school.  
ill it lingered near,  
appear.  
ger children cry; "Why, Mary loves  
y."

CHARACTERS

### 22.1.1 N-gram character models

Ultimately, a written text is composed of **characters**—letters, digits, punctuation, and spaces in English (and more exotic characters in some other languages). Thus, one of the simplest language models is a probability distribution over sequences of characters. As in Chapter 15, we write  $P(c_{1:N})$  for the probability of a sequence of  $N$  characters,  $c_1$  through  $c_N$ . In one Web collection,  $P(\text{"the"}) = 0.027$  and  $P(\text{"zgq"}) = 0.000000002$ . A sequence of written symbols of length  $n$  is called an  $n$ -gram (from the Greek root for writing or letters), with special case “unigram” for 1-gram, “bigram” for 2-gram, and “trigram” for 3-gram. A model of the probability distribution of  $n$ -letter sequences is thus called an  **$n$ -gram model**. (But be careful: we can have  $n$ -gram models over sequences of words, syllables, or other units; not just over characters.)

An  $n$ -gram model is defined as a **Markov chain** of order  $n - 1$ . Recall from page 568 that in a Markov chain the probability of character  $c_i$  depends only on the immediately preceding characters, not on any other characters. So in a trigram model (Markov chain of order 2) we have

$$P(c_i | c_{1:i-1}) = P(c_i | c_{i-2:i-1}).$$

# Chapter 22: NLP

## Language Identification w/ N-Grams

- $P(\text{text} \mid \text{language})$  is calculated

$$\begin{aligned}\ell^* &= \underset{\ell}{\operatorname{argmax}} P(\ell \mid c_{1:N}) \\ &= \underset{\ell}{\operatorname{argmax}} P(\ell) P(c_{1:N} \mid \ell) \\ &= \underset{\ell}{\operatorname{argmax}} P(\ell) \prod_{i=1}^N P(c_i \mid c_{i-2:i-1}, \ell)\end{aligned}$$

# Chapter 22: NLP

## Spam or Ham w/ N-Grams

- $P(\text{message} \mid \text{Ham})$  is calculated
- $P(\text{message} \mid \text{Spam})$  is calculated

$$\operatorname{argmax}_{c \in \{\text{spam}, \text{ham}\}} P(c \mid \text{message}) = \operatorname{argmax}_{c \in \{\text{spam}, \text{ham}\}} P(\text{message} \mid c) P(c)$$

# Introduce Yourself

- Name
- Sophomore/Junior/Senior
- Tech Interests
  - <https://bit.ly/c166f20Intro>

<b>Assignment/Examination/Presentation</b>	<b>Points</b>
In-Class/Presentations	100
Quizzes	100
Qwiklabs	200
Project	200
Midterm	150
Final	250