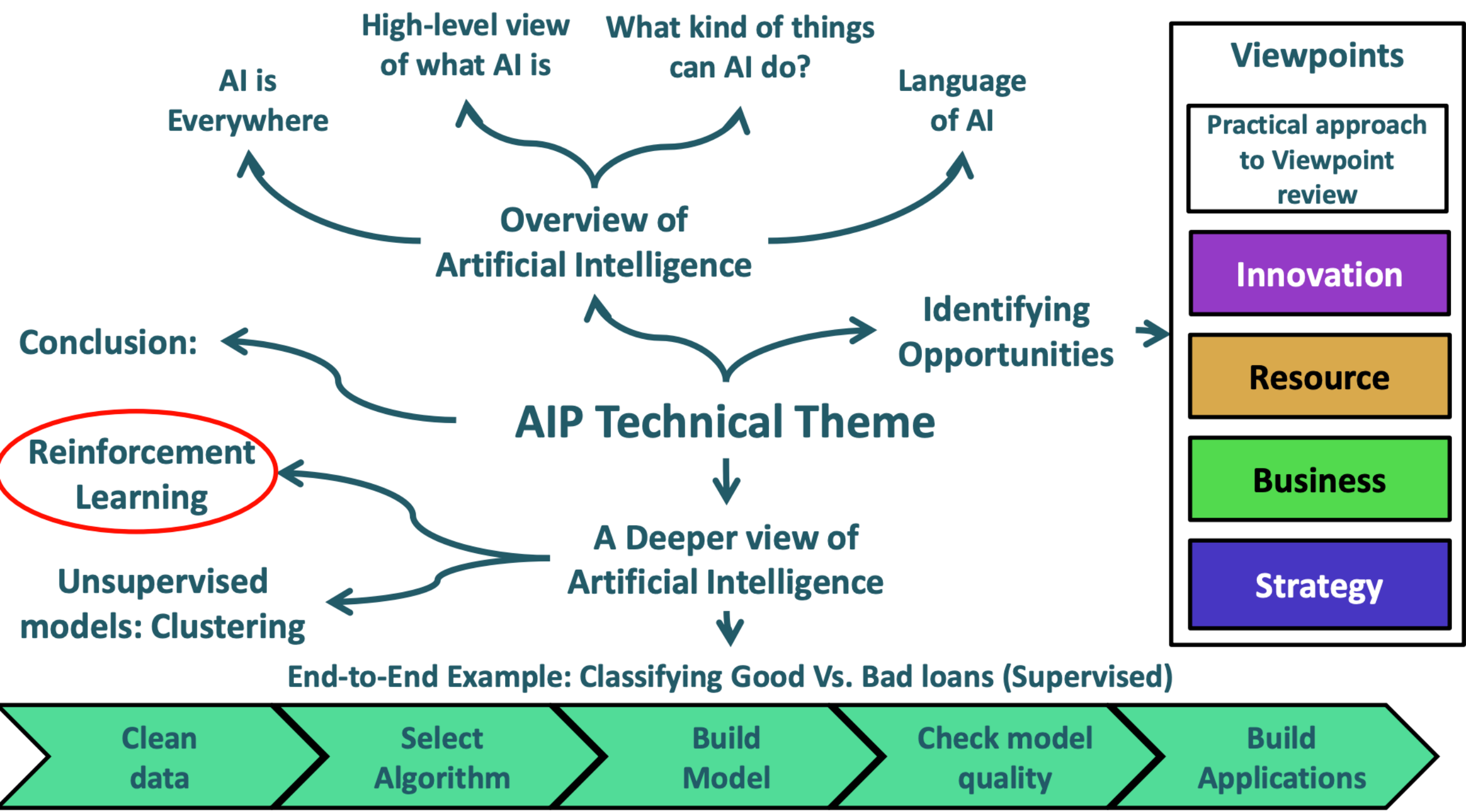
immerse
EDUCATION

Reinforcement Learning

Elizabeth Savochkina | 8th June

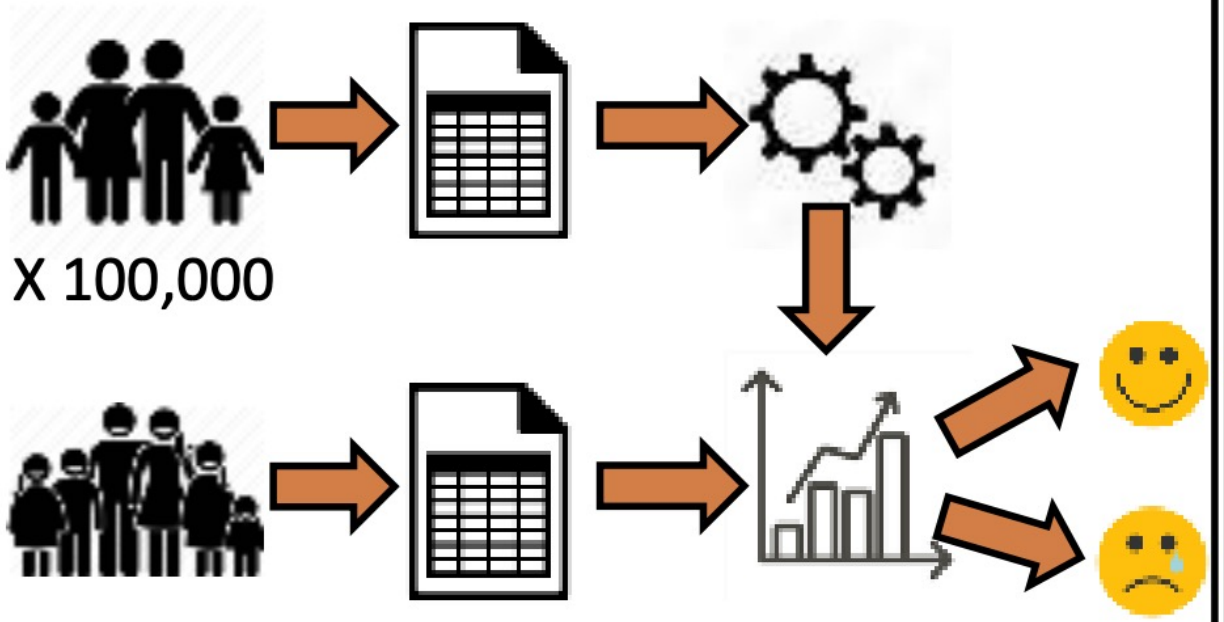

High-level view of what AI is
What kind of things can AI do?
AI is Everywhere
Language of AI
Overview of Artificial Intelligence
Identifying Opportunities
Conclusion:
AIP Technical Theme
Reinforcement Learning
A Deeper view of Artificial Intelligence
Unsupervised models: Clustering
Viewpoints
Practical approach to Viewpoint review
Innovation
Resource
Business
Strategy
End-to-End Example: Classifying Good Vs. Bad loans (Supervised)
Clean data
Select Algorithm
Build Model
Check model quality
Build Applications

Shape of a Day

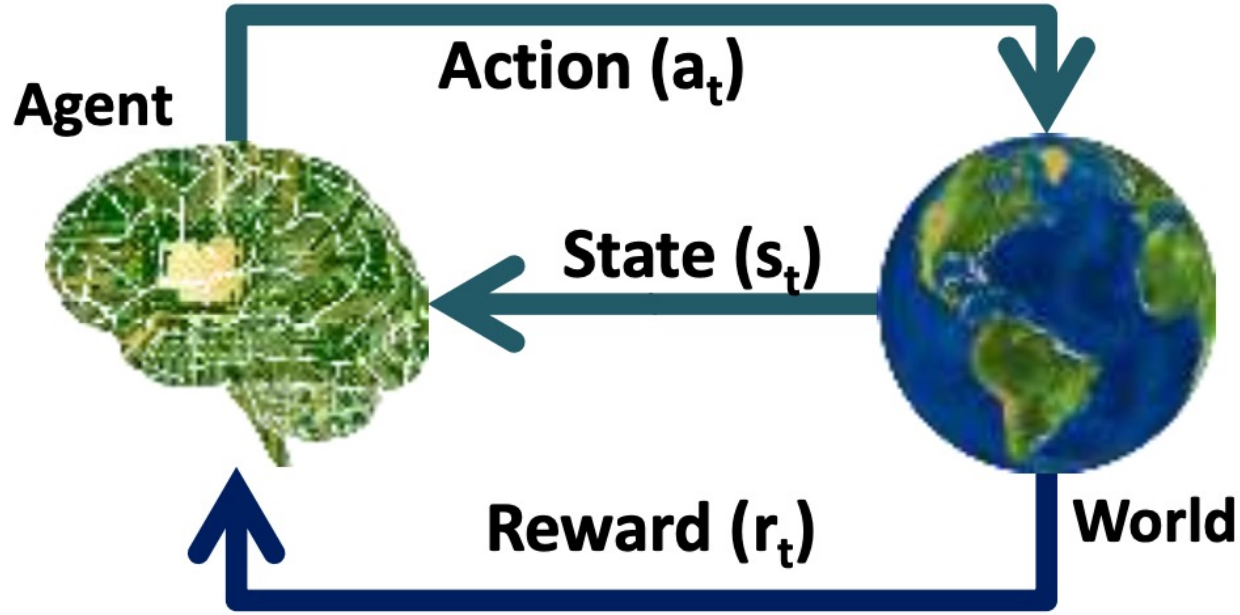
Registration 9.00am to 9.30am
K-NN clustering workshop 9.30am to 10.30am
Reinforcement Learning 10.30am to 11.30am
Break 11.30am to 13.30 pm
Keeping up with AI + **Machine Learning updates +** **Key Industry Players** 13.30am to 14.30pm
BPMN + **Project Discussion +** **coding homework exercises** 14.30am to 15.30pm

Reinforcement : A Different Way of Learning

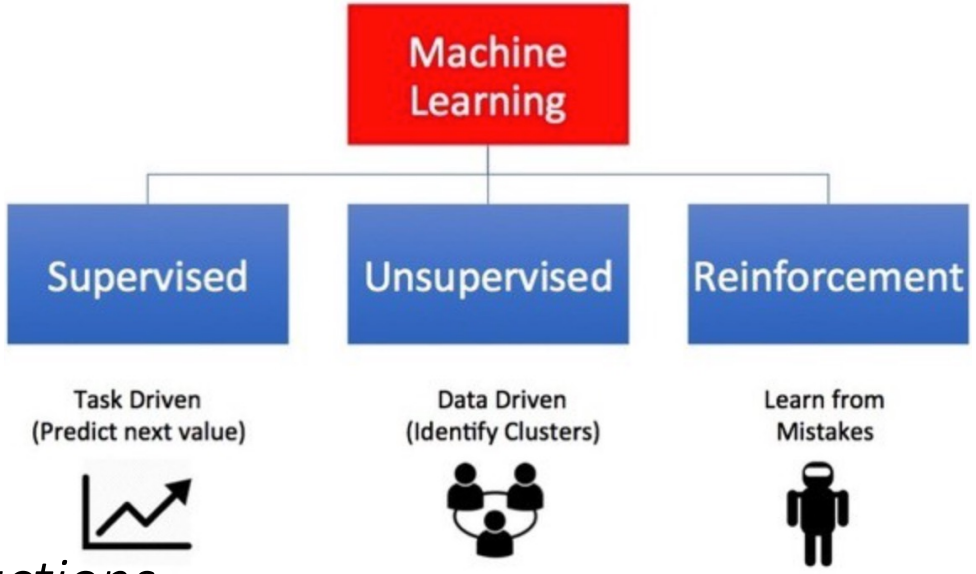
Supervised Learning



Reinforcement Learning



ML Differences



Supervised (SL) vs Reinforcement Learning (RL):

- In SL the feedback/labels provided is/are *correct set of actions*
- RL uses *rewards and punishments* as signals for positive and negative behaviour

Unsupervised (UL) vs RL (what is the goal?):

- The goal of UL is to find similarities and differences between data points
- The goal od RL to find a suitable action model that *would maximize the total cumulative reward* of the agent.

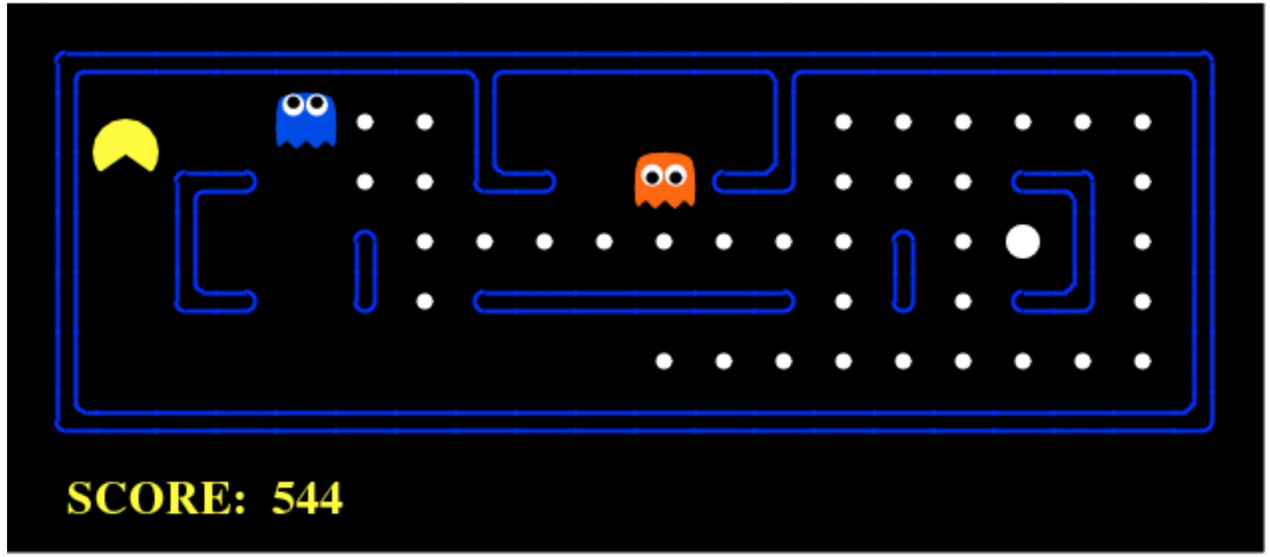
How to formulate a basic Reinforcement Learning problem?

Environment — Physical world in which the agent operates
State — Current situation of the agent (more in the next slide)
Reward — Feedback from the environment
Policy — Method to map agent's state to actions
Value — Future reward that an agent would receive by taking an action in a particular state

PacMan game:
- the goal of the agent (yellow) is to eat food in the grid while avoiding the ghosts on its way.
Reward: Food
Punishment: Gets killed by ghosts

What are the states?
- location of the agent in the grid world

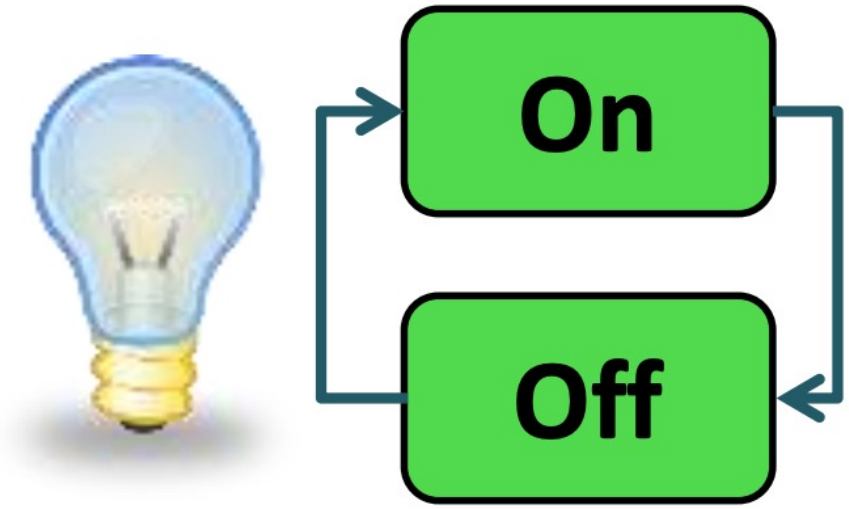
What is the total cumulative reward?
- agent winning the game!



What is ‘State’?

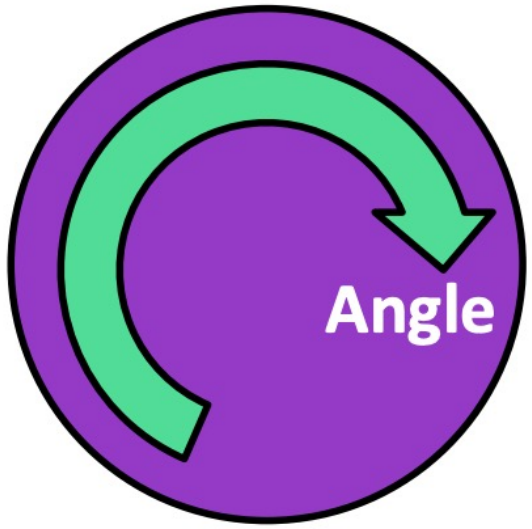
- ‘State’ is the condition that an object is in
- A description of:
  - ‘where’ an object is
  - ‘How’ an object is
- Data describing the objects condition

Light-Bulb



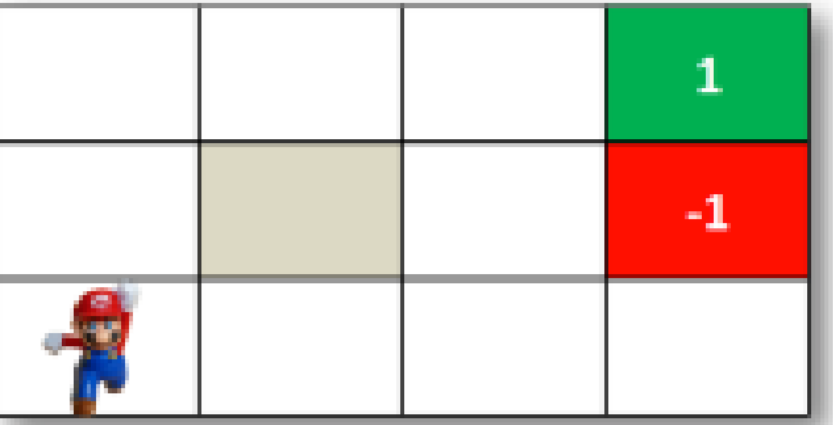
Discrete State

Wheel



Continuous State

Q-Learning : Mapping States to Actions



S 1	S2	S3	1 S4
S5		S6	-1 S7
S8	S9	S10	S11

State	Action ←	↑	→	↓	Reward
S1					-0.01
S2					-0.01
S3					-0.01
S4					1
S5					-0.01
S6					-0.01
S7					-1
S8					-0.01
S9					-0.01
S10					-0.01
S11					-0.01
S12					-0.01

Each time we enter a state there is either a 'cost' or a 'reward'

The task is to learn the best policy for playing this game

Updating the Q-Table

State	Action ←	↑	→	↓
S1	0	0	0	0
S2	0	0	0	0
S3	0	0	0	0
S4	0	0	0	0
S5	0	0	0	0
S6	0	0	0	0
S7	0	0	0	0
S8	0	0	0	0
S9	0	0	0	0
S10	0	0	0	0
S11	0	0	0	0
S12	0	0	0	0

S 1	S2	S3	1 S4
S5		S6	-1 S7
S8	S9	S10	. S11

	←	↑	→	↓
S3	-0.01	-0.01	1	-0.01

Exploration

As we iterate, information about the value of each State, and each State-Action pair propagates through the model

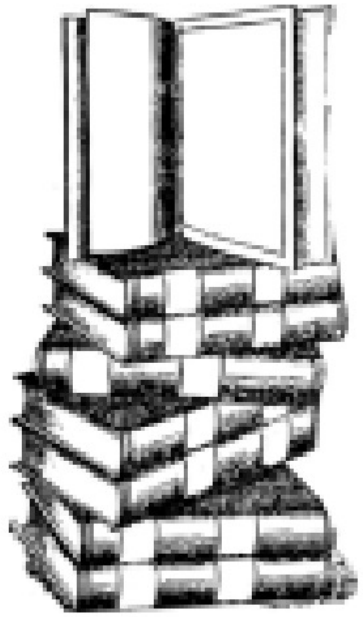
S 1	S2	S3	1 S4	
S5		S6	-1 S7	
S8	S9	S10	S11	

	←	↑	→	↓
S2	-0.01	-0.01	1	-0.01
S3	-0.01	-0.01	1	-0.01

Action-Value: expected reward for taking action in a particular state

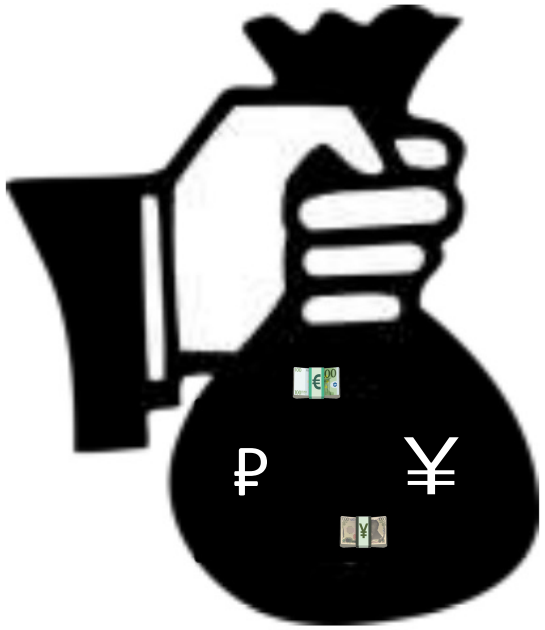
State-Action (Q-function): specified how good it is for an agent to perform a particular action in a state with a policy π

Three Important Factors for Learning



Leaning Rate

α or ‘Alpha’



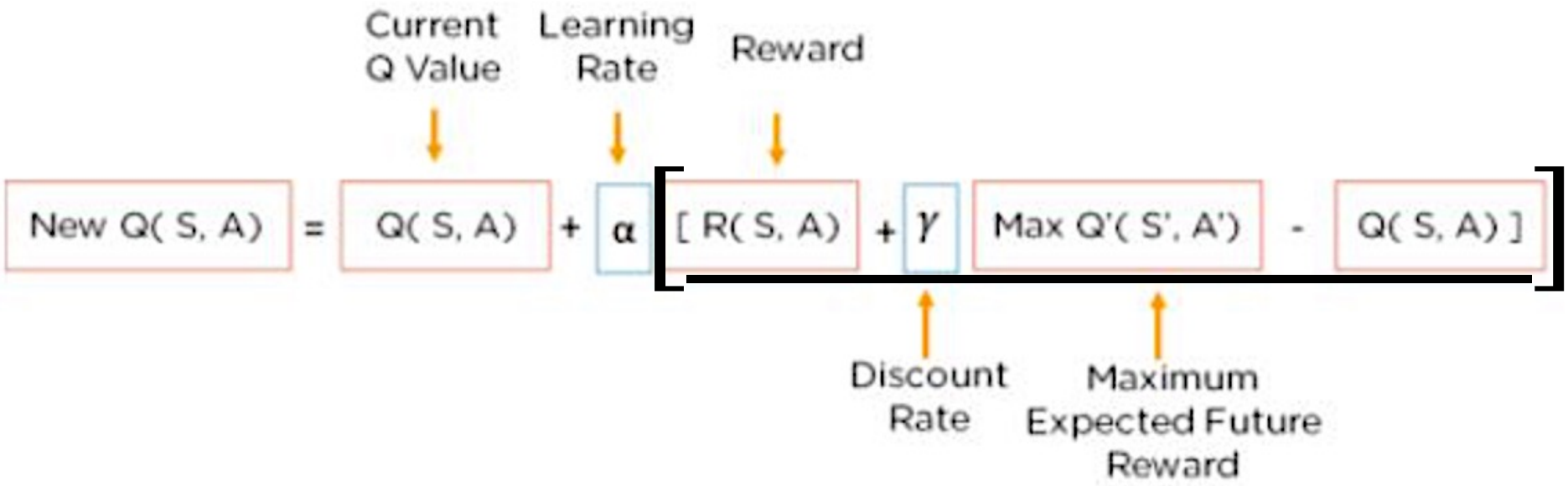
Discount Rate

γ or ‘Gamma’

Exploration Rate

ε or ‘Epsilon’

Bellman Equation



Leaning Rate
α or ‘Alpha’

Learning Rate – Alpha (α)

- The importance of new knowledge versus old knowledge
- The rate at which new information overrides old information

Alpha 1

Observation	1	2	3	4	5	6	7
See a red Iris	✓	✓	✓	✓	✗	✗	✗
See a blue Iris	✗	✗	✗	✗	✓	✓	✓

Model

Iris is red	100%	100%	100%	100%	0%	0%	0%
Iris is Blue	0%	0%	0%	0%	100%	100%	100%

Learning Rate – Alpha (α) = 0

- Alpha = 1
  - 100% observation
  - No memory
  - The last thing I saw was the truth
- Alpha = 0
  - 100% memory
  - No learning
  - The first thing I saw was the truth

Alpha 0

Observation	1	2	3	4	5	6	7
See a red Iris	✓	✓	✓	✓	✗	✗	✗
See a blue Iris	✗	✗	✗	✗	✓	✓	✓

Model

Iris is red	100%	100%	100%	100%	100%	100%	100%
Iris is Blue	0%	0%	0%	0%	0%	0%	0%

Learning Rate – Alpha (α) = 0.2

- Alpha = 0.2
  - 80% memory, 20% new observations
  - Balancing memories against new information
- Optimum alpha will depend on amount of experience and variation observations

Alpha 0.2

Observation	1	2	3	4	5	6	7
See a red Iris	✓	✓	✓	✓	✗	✗	✗
See a blue Iris	✗	✗	✗	✗	✓	✓	✓

Model

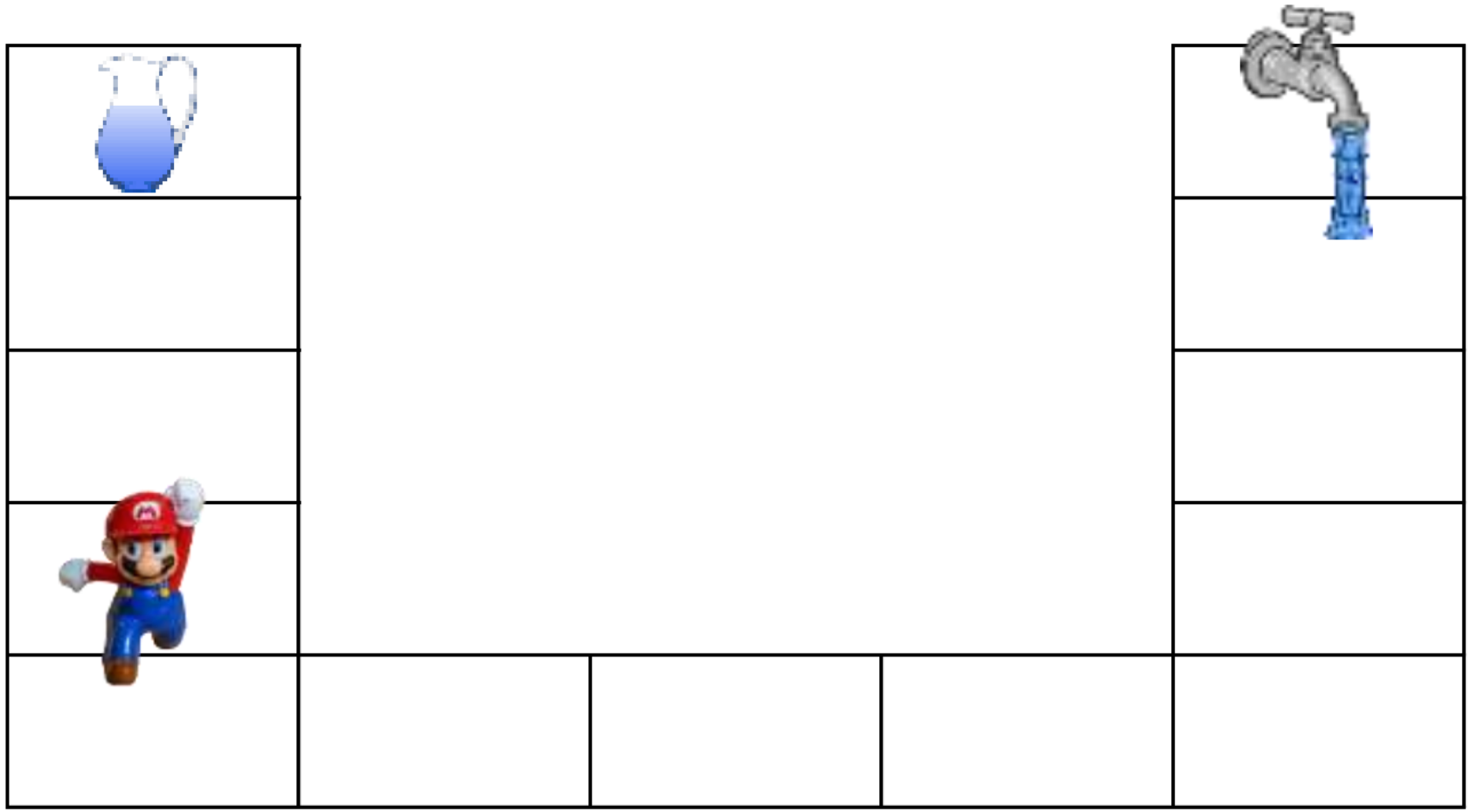
Iris is red	100%	100%	100%	100%	80%	64%	51%
Iris is Blue	0%	0%	0%	0%	20%	36%	49%

Demonstration 5.1

Discount Rate
γ or ‘Gamma’

Discount Rate – Gamma (γ)

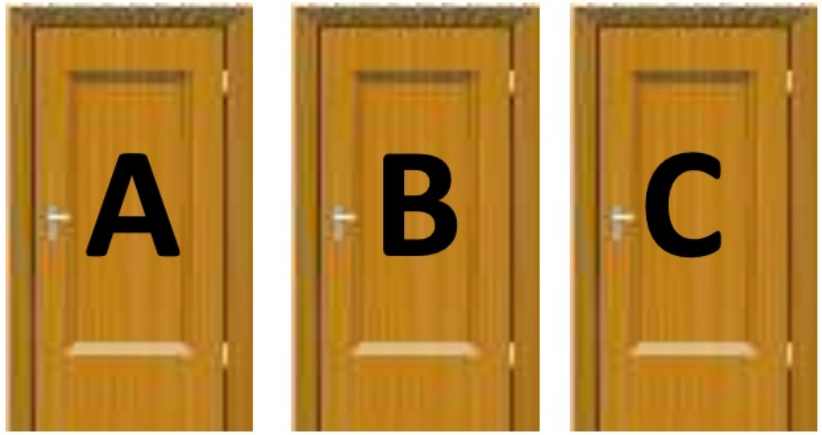
- How much does the Agent value quick rewards Vs. long-term rewards
- If each step made has a cost …
- Should Mario take a short walk for a small reward or a long walk for a much larger reward?



Exploration Rate
ε or ‘Epsilon’

Exploration Rate – Epsilon (ε)

- To what extent does the agent explore new options Vs. exploiting previous knowledge?
- Let's say that our agent, Mario, is confronted with 3 doors ..
- And it costs him $1 to open each door



- He opens door 'A' and gets a reward of $2 ..
- .. Thus learning that doors of type 'A' has a total value of $1
- **Question**: When confronted with 3 identical doors .. What should Mario do?

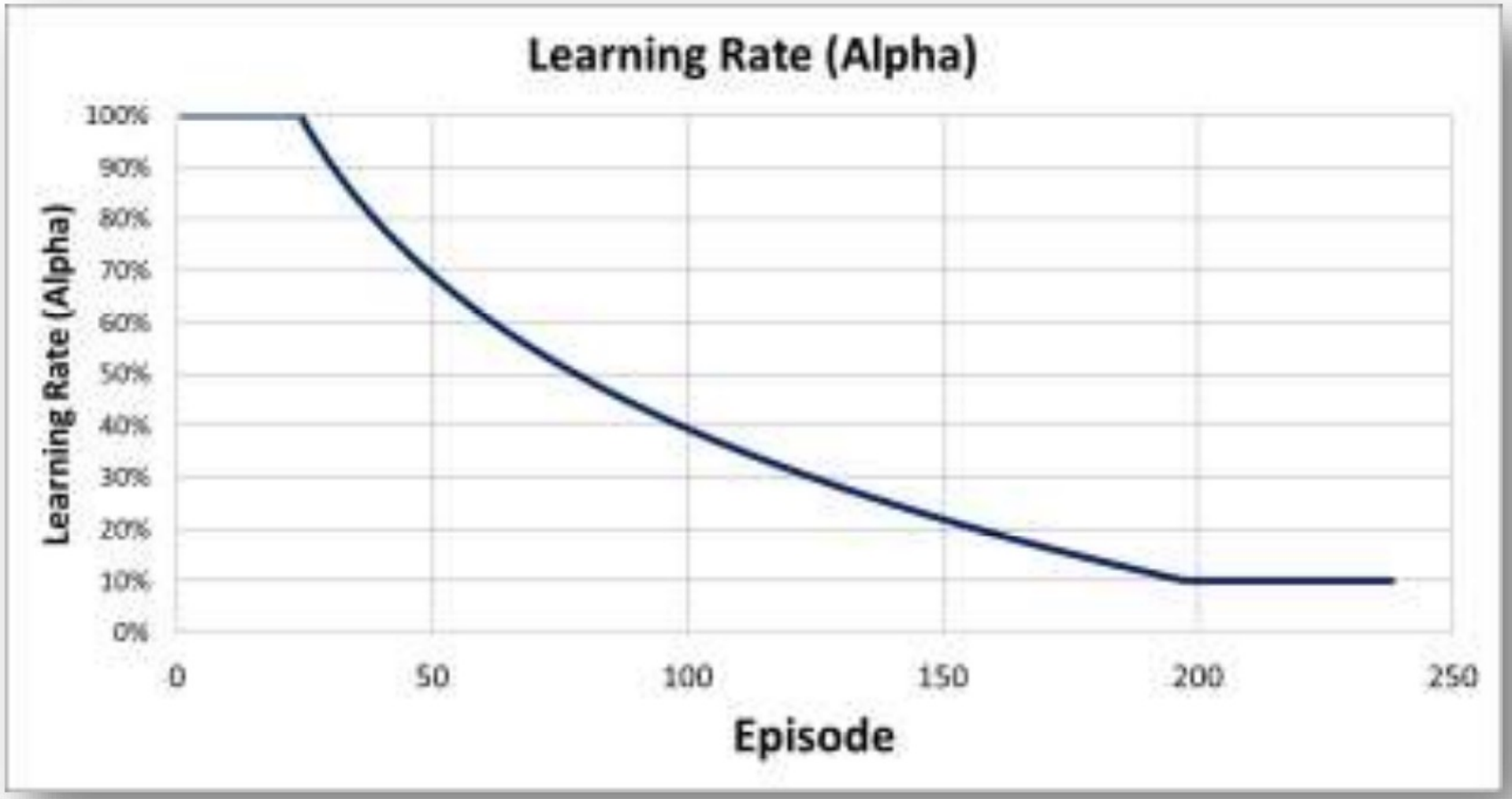
Exploration Rate – Epsilon (ε)

- Our agent Mario could exploit the knowledge he already has ..
- ..and always open door A and always get a 1$ reward
- But maybe there is a huge reward hidden behind doors of type C!

- Unless Mario occasionally explores the environment by selecting random doors he may not find the best solution
- Epsilon is a measure of the frequency of exploring rather than exploiting

Changing Alpha and Epsilon Over time

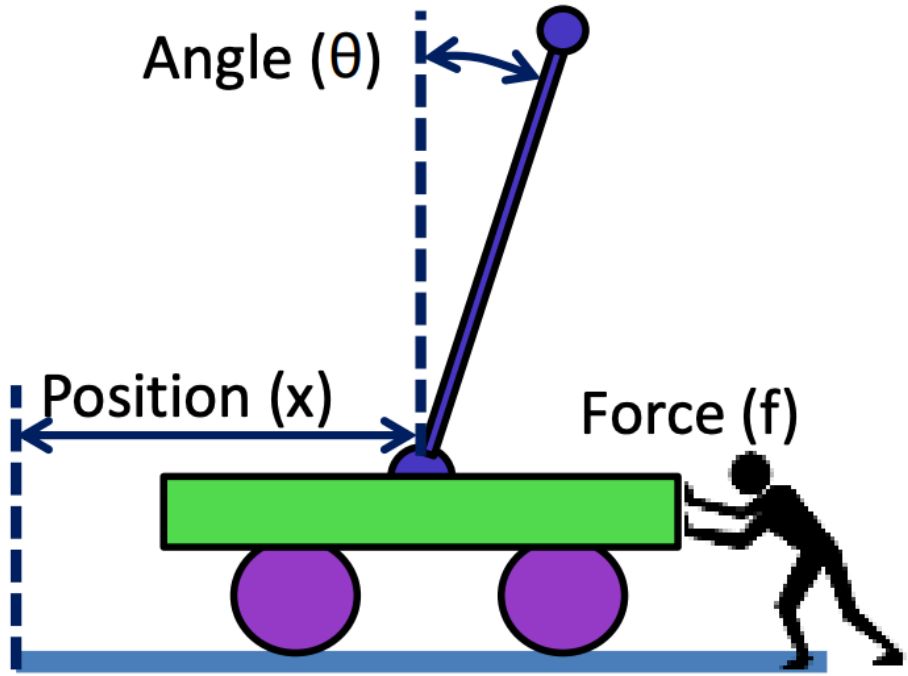
- Learning Rate (alpha) and Exploration Rate (epsilon) are normally updated during learning
- For example, alpha may change the 'style' of learning from:
  - 'mostly based on new information' to
  - 'mostly based on experience'



A Practical Example: Cart-Pole from AI Gym

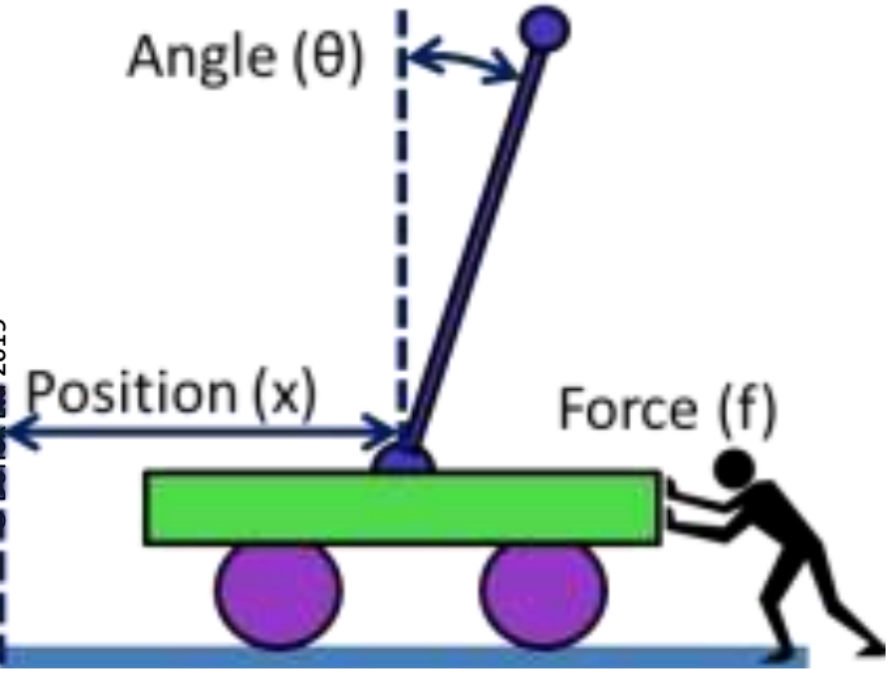
Learning to Control the Cart-Pole : Q-Learning

- The Goal is to balance the pole
- The agent observes the 'State':
  - Position (x)
  - Angle of the pole (θ)
  - Velocity (dx/dt)
  - Angular velocity (dθ/dt)



- The 'Action' is a force the agent can apply to the cart
- The agent receives 1 point reward for each time step during which the pole is upright
- Goal is to get 200 points

Simplifying Observations

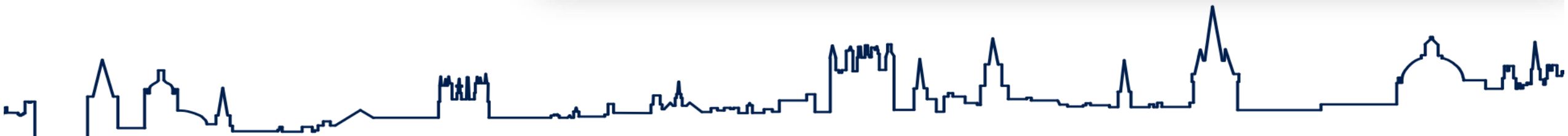


- An issue with Cart-Pole is each of the state observations is represented by a decimal number
- This means that we would need a massive Q-table to represent every possible value of the 4 observables
- This is often a limitation with using Q-Tables
- But in this case we can just split the observations into more discrete ‘buckets’
  - 0 > angle <= 20 -> 0
  - 20 > angle <= 40 -> 1

Q-Table for the Cart-Pole

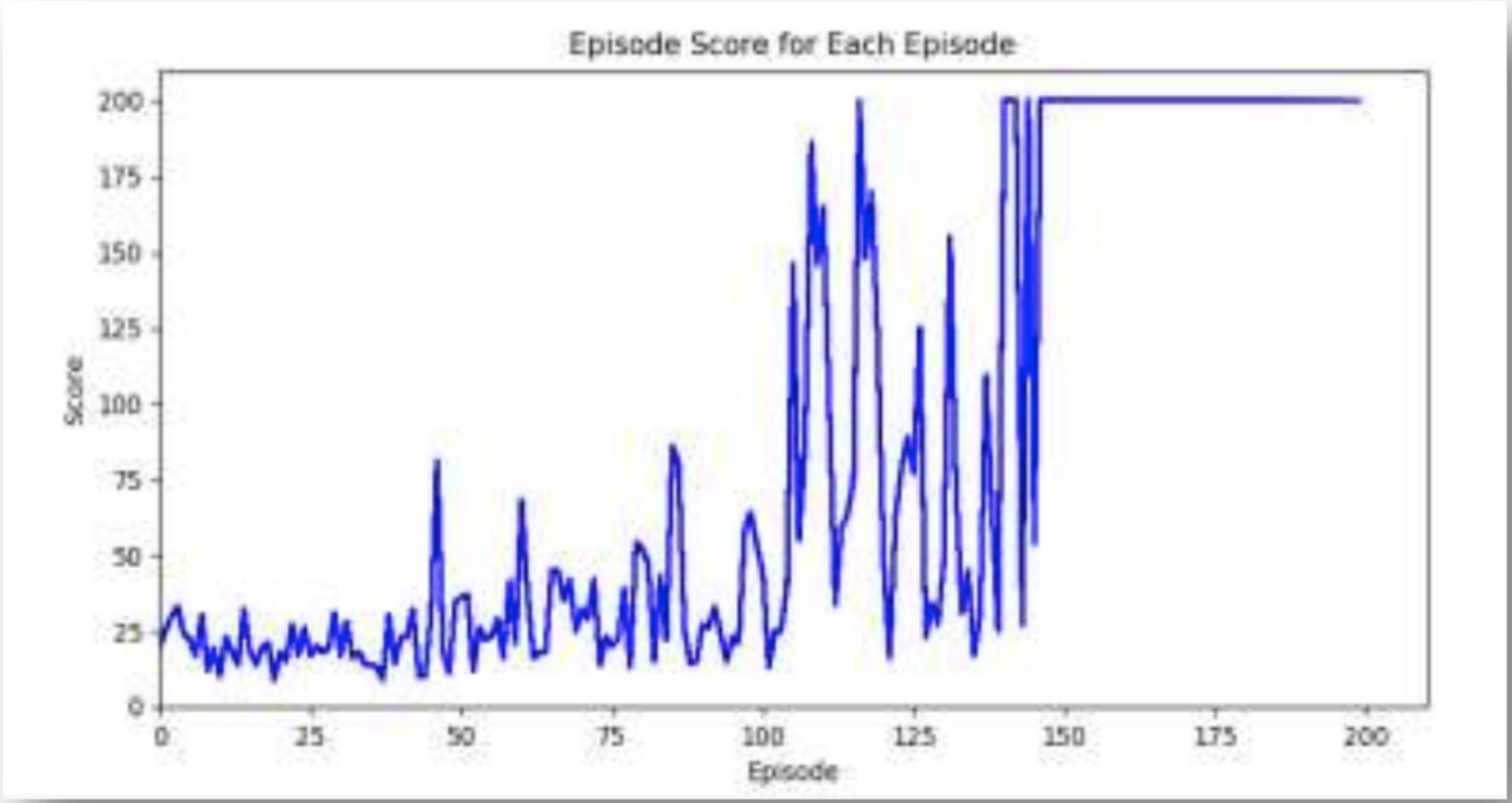
- The programme runs many episodes of the cart-pole task
- At the start it takes mostly random actions, observes the results and stores them in the q-table
- Later, actions are selected that previously gave the best score

State				Action	
Position	Angle	Velocity	Angular Velocity	Force Left	Force Right

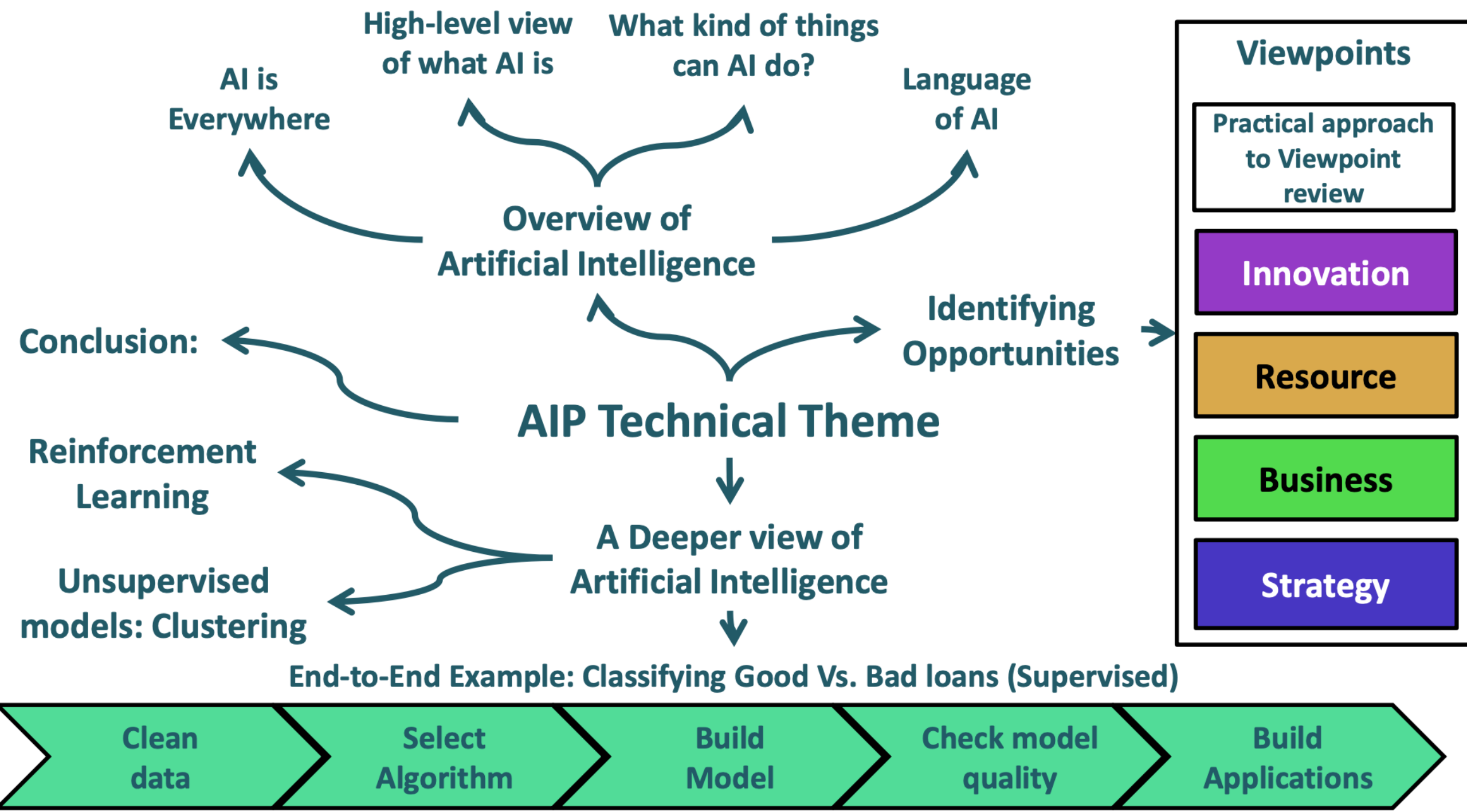


Demonstration

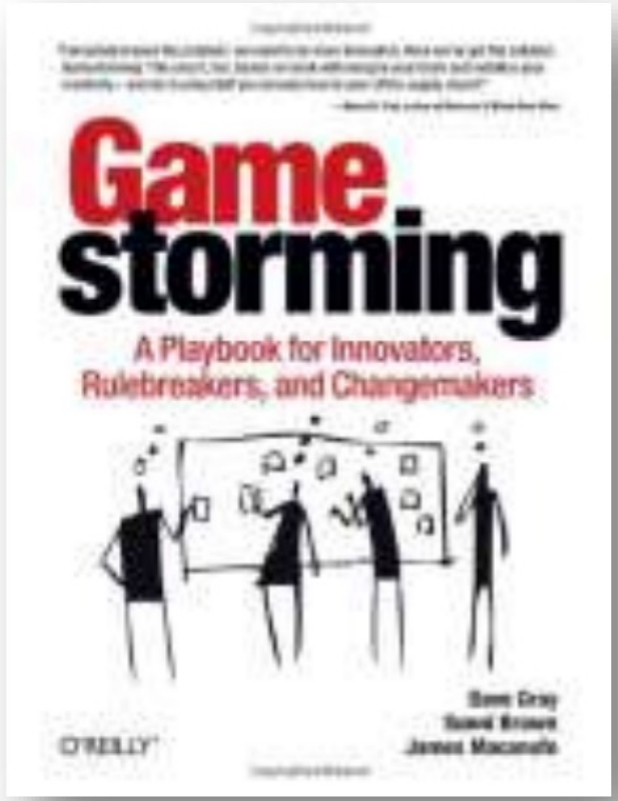
- The solution in Python is 107 lines of code …
  - 4 to import pre-written library functions
  - 15 to configure the model
  - 7 to split measurements into ‘buckets’
  - 1 to select an action
  - 1 to do the learning
  - 14 to plot the graph
  - 25 to run the episodes
  - + comments and white-space

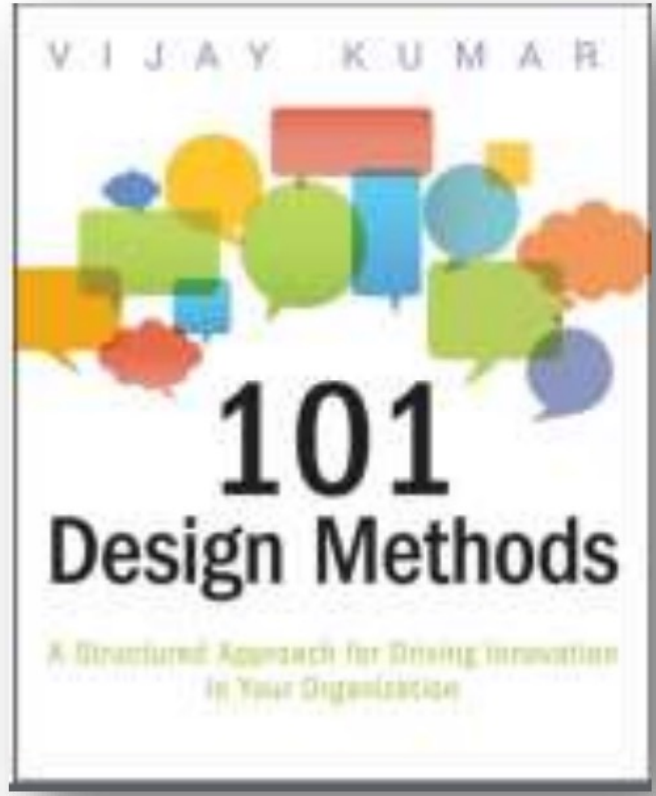


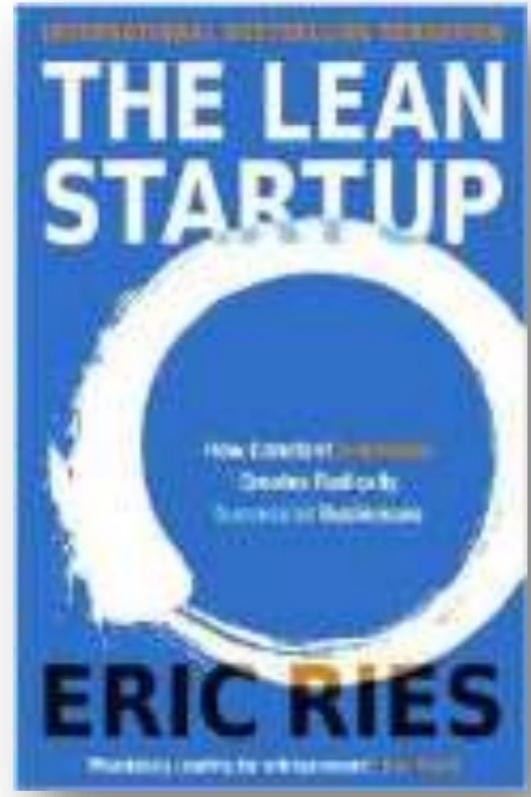
Demonstration 5.3, 5.4, 5.5


High-level view of what AI is
What kind of things can AI do?
AI is Everywhere
Language of AI
Overview of Artificial Intelligence
Conclusion:
Identifying Opportunities
AIP Technical Theme
Reinforcement Learning
A Deeper view of Artificial Intelligence
Unsupervised models: Clustering
Viewpoints
Practical approach to Viewpoint review
Innovation
Resource
Business
Strategy
End-to-End Example: Classifying Good Vs. Bad loans (Supervised)
Clean data
Select Algorithm
Build Model
Check model quality
Build Applications

Conclusion :
What Next?

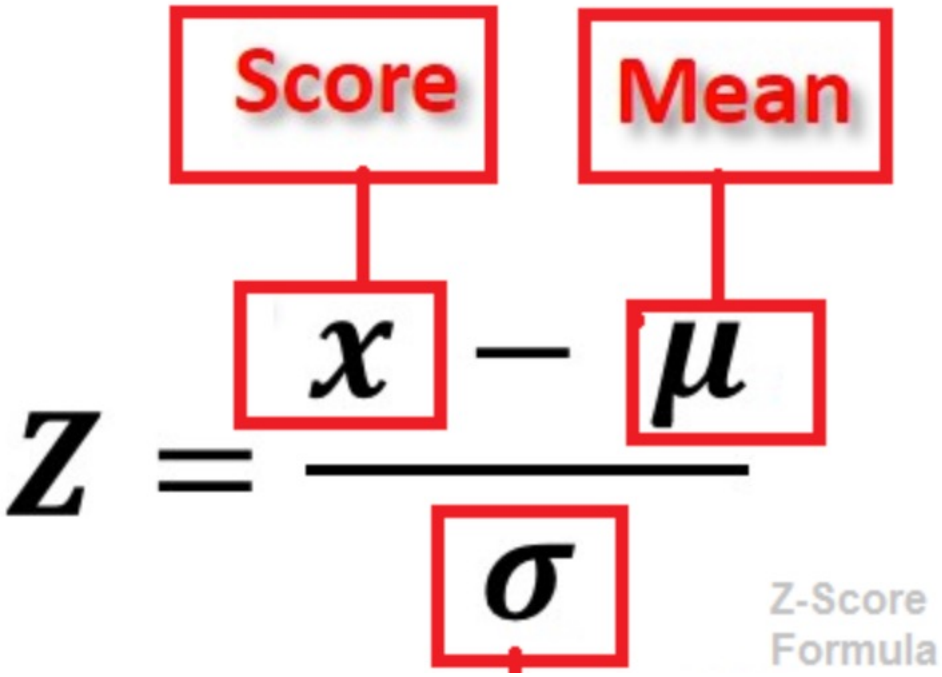

Game storming
A Playbook for Innovators, Rulebreakers, and Changemakers
O'REILLY


VIJAY KUMAR
101
Design Methods


THE LEAN STARTUP
ERIC RIES

Z-normalization

$$s = \sqrt{\frac{\sum (x - \bar{x})^2}{n}}$$



$$\text{Standard deviation} = \sqrt{\frac{\sum (\text{every individual value of marks} - \text{mean of marks})^2}{n}}$$

Mean of marks = 8 + 10 + 15 + 20 /4 = 13.25

$$= \sqrt{\frac{(8 - 13.25)^2 + (10 - 13.25)^2 + (15 - 13.25)^2 + (20 - 13.25)^2}{4}}$$

$$= \sqrt{\frac{(-5.25)^2 + (-3.25)^2 + (1.75)^2 + (6.75)^2}{4}}$$

$$= \sqrt{\frac{27.56 + 10.56 + 3.06 + 45.56}{4}} = \sqrt{\frac{86.74}{4}} = \sqrt{21.6} = 4.6$$

marks
8
10
15
20

Mean = 13.25

Standard deviation = 4.6

$$ZScore = \frac{x - \mu}{\sigma} = \frac{8 - 13.25}{4.6} = -1.14$$

$$ZScore = \frac{x - \mu}{\sigma} = \frac{10 - 13.25}{4.6} = -0.7$$

$$ZScore = \frac{x - \mu}{\sigma} = \frac{15 - 13.25}{4.6} = 0.3$$

$$ZScore = \frac{x - \mu}{\sigma} = \frac{20 - 13.25}{4.6} = 1.4$$