Applied Machine Learning
ML 101 &Course Introduction

James G. Shanahan [1,2]

[1]Church and Duncan Group,

[2]*School of Informatics, Computing and Engineering, Indiana University*

EMAIL: James_DOT_Shanahan_AT_gmail_DOT_com

Outline

- **AI/ML 101:**
  - Introduction
  - Linear Regression
  - Beyond linear regression
- **Top AI market trends** to watch in 2017 and beyond
  - Key technical developments
  - Case studies
- **ML investor or entrepreneur**
- **Short Case Study in a Python Notebook**
- **Conclusions**
- **Course Logistics**

James G. Shanahan 30+ years in data science

16+

Technology

AI, Systems, Parallel Computing, Hadoop, Spark, Python, R, *Scala*, *Java*

30+years

Domain Expertise

Digital Advertising & Marketing, Web+mobile+local Search, Anticipatory info. systems, Cellular Networks, Social Networks

Leadership, Business Acumen, Teacher

Led teams of R&D, r&D
Xerox Research, AT&T, Turn, NativeX, Adobe
Entrepreneur
Teach at UC Berkeley

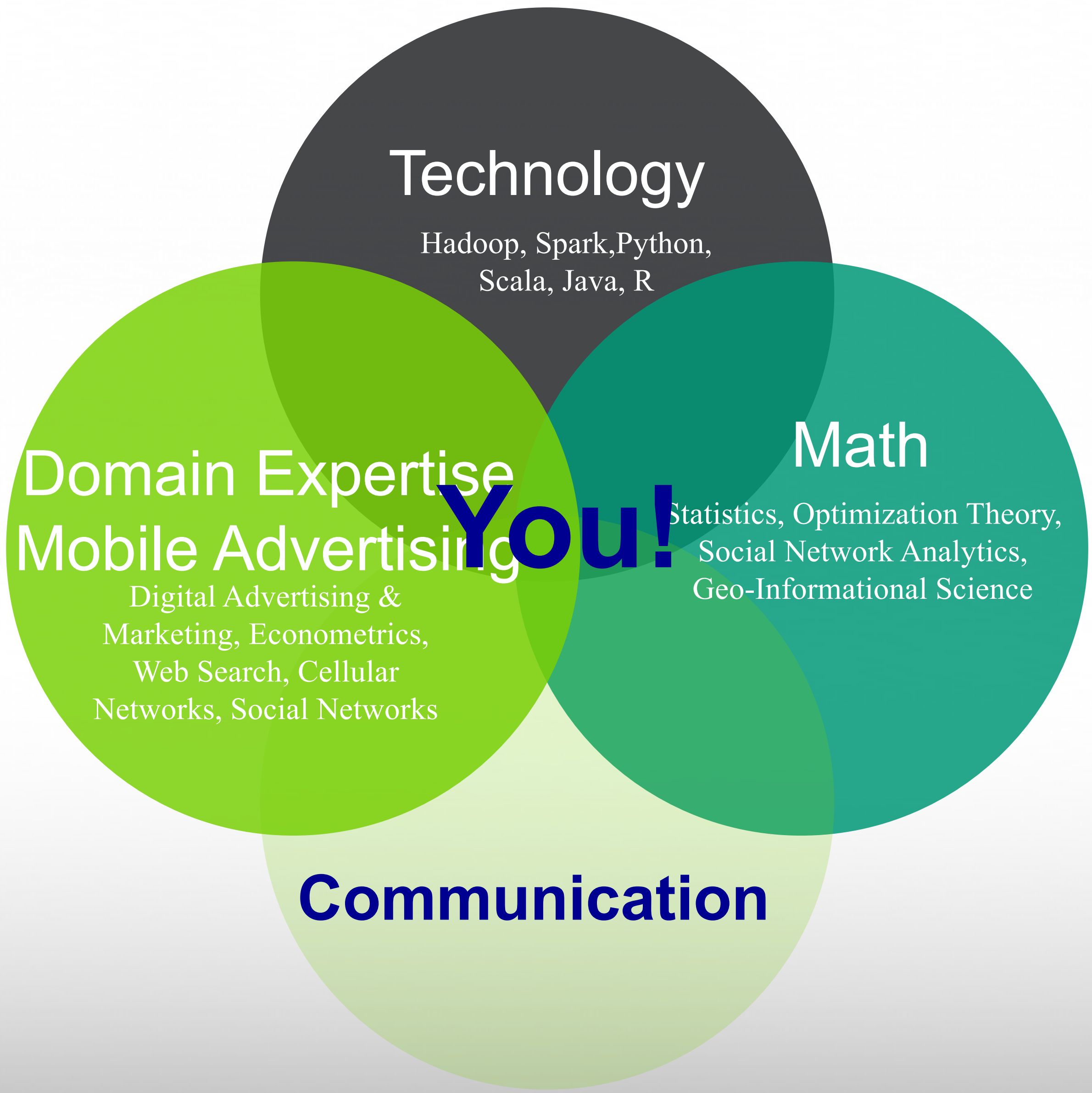
16+

Math&Theory

Statistics, Optimization Theory, Probability
Social Network Analytics, Geo-Informational Science, HCI
Graphs, NLP

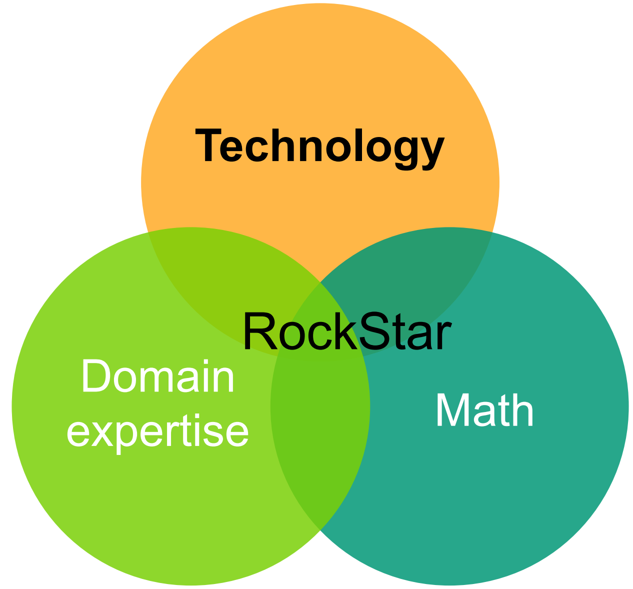
25+

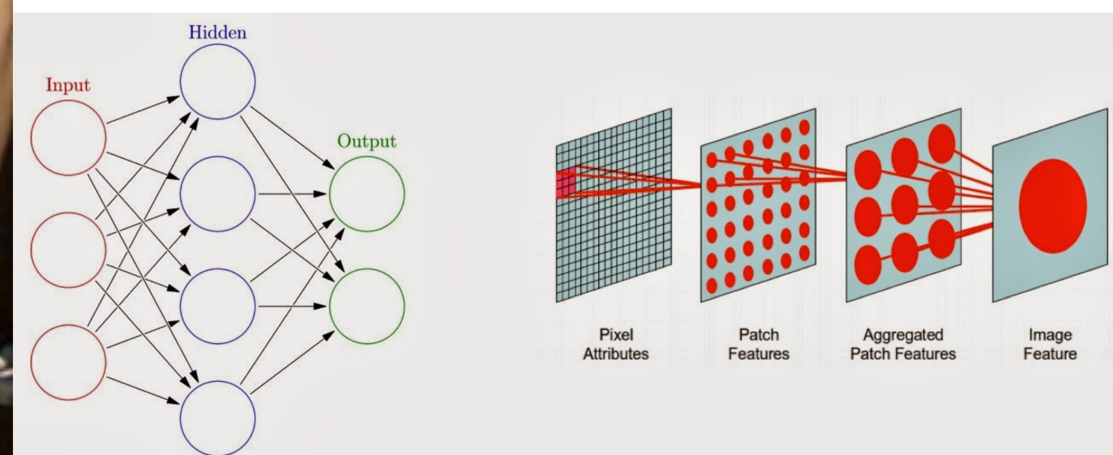
Data Scientist



AI via machine learning at scale

RockStars and Super Models





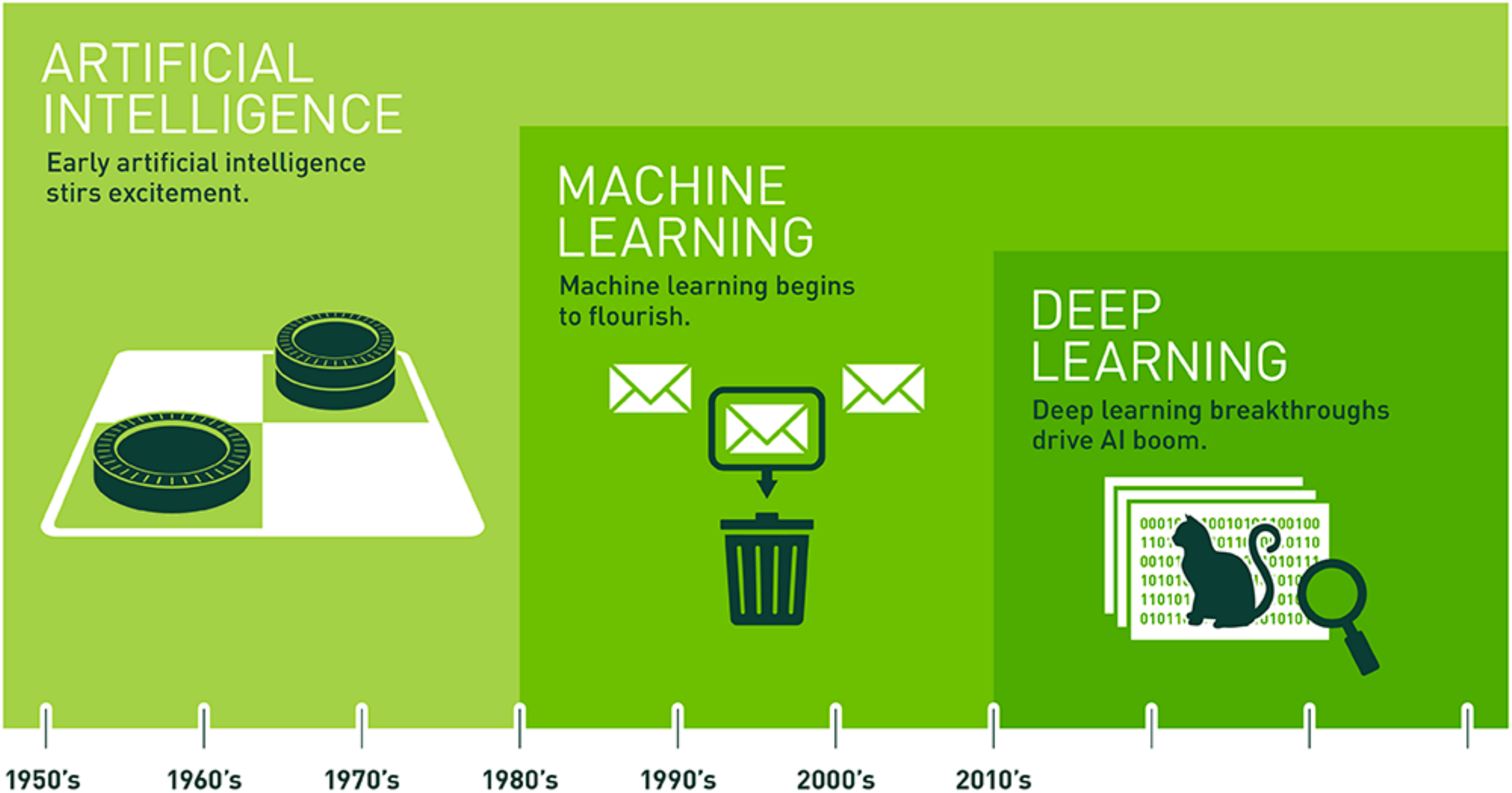
Outline

- **AI/ML 101:**
  - Introduction
  - Linear Regression
  - Beyond linear regression
- **Top AI market trends** to watch in 2017 and beyond
  - Key technical developments
  - Case studies
- **ML investor or entrepreneur**
- **Short Case Study in a Python Notebook**
- **Conclusions**
- **Course Logistics**

Artificial intelligence

- **AI, in its simplest definition, is an umbrella term for technologies that are**
  - inspired by biological systems,
  - giving computers human-like abilities related to seeing, reasoning, hearing, and learning (and maybe things like empathy and introspection).
- **AI encompasses technologies like**
  - machine learning, deep learning,
  - natural language processing (NLP), computer vision,
  - machine reasoning, and strong AI.
- **As an engineer**
  - any device that perceives its environment and takes actions that maximizes its chance of success at some goal
- **AI (software, services, hardware) is a burgeoning industry**
  - Generating over USD 6.5 billion (WorldWide) in revenue in 2015, growing to $300B by 2025s

AI→ML →DL



Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

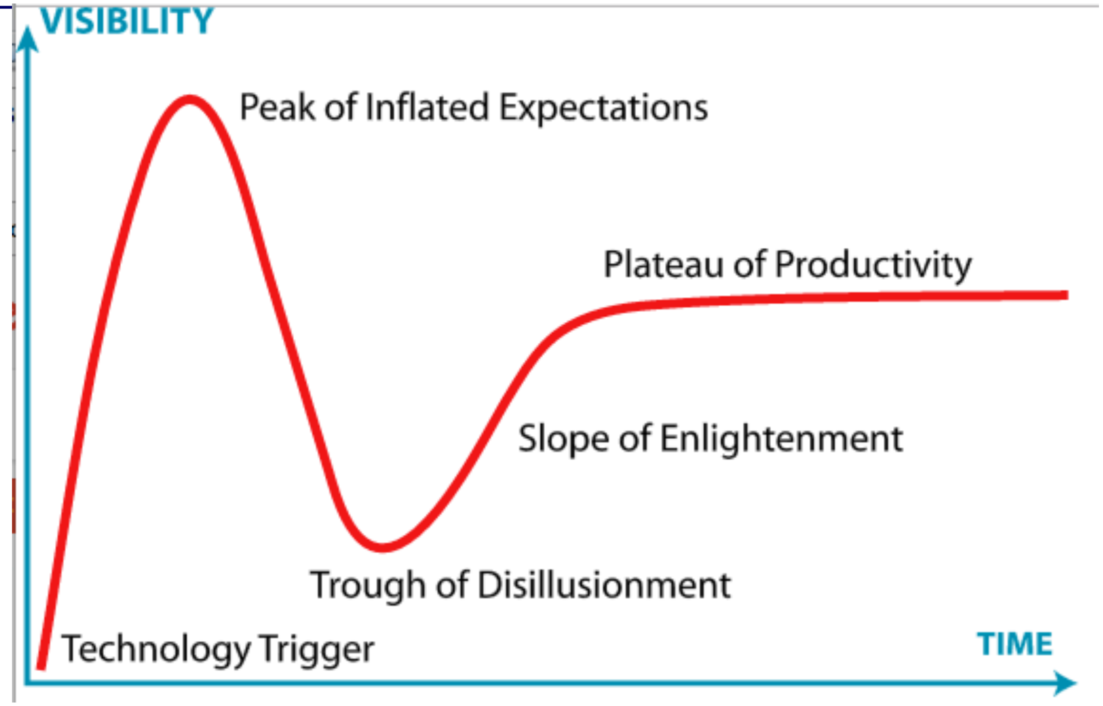
https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/

AI: 4 Hypecyles and 3 winters

1. **1960s-1969 AI**
   – Top-down

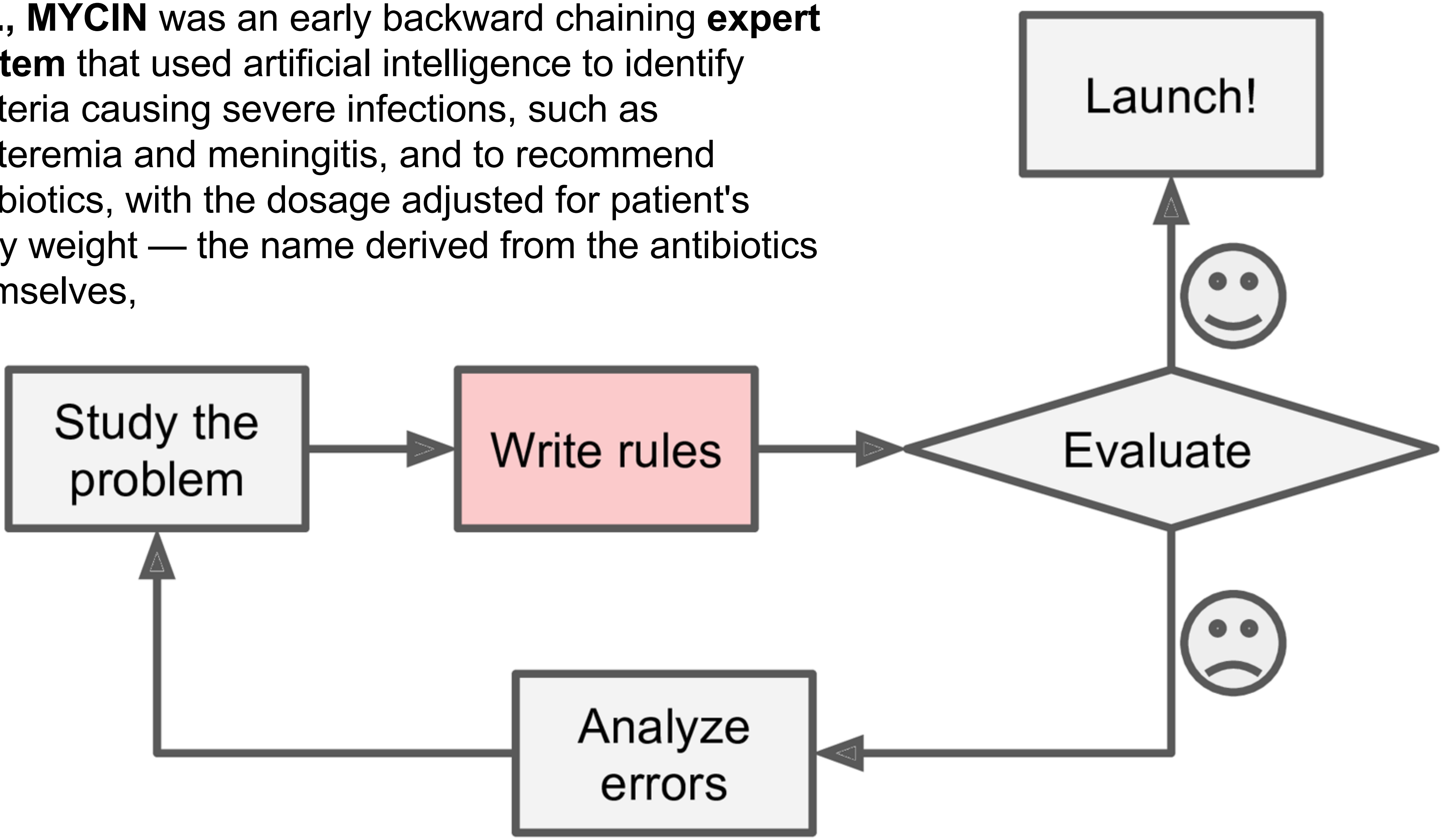
2. **1980-1991 AI**
   – Top-down

3. **2000-2008**
   – Internet companies, Wall Street
4. **2010s - Present**
   – Bottom up, data driven, industry-society wide

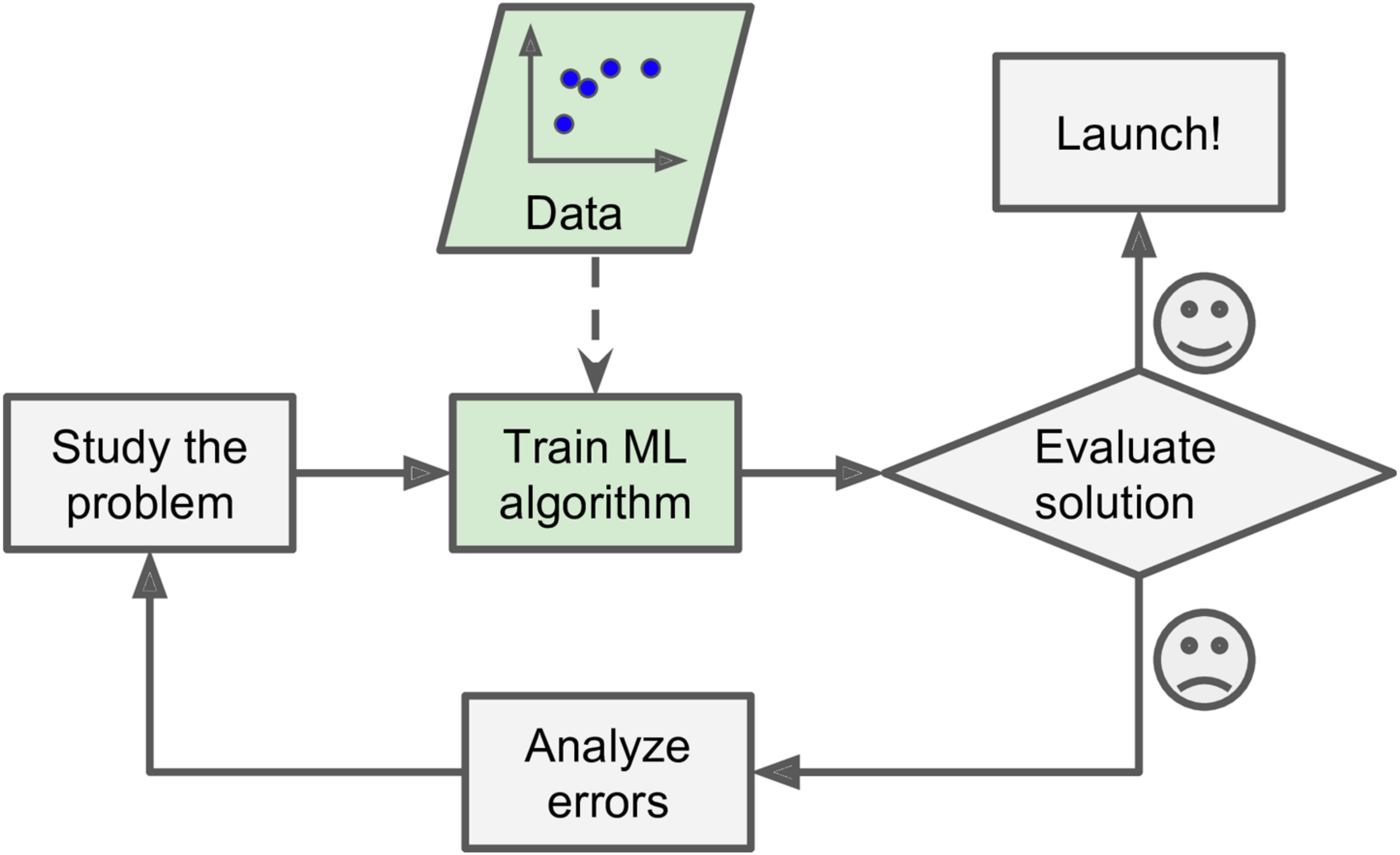


Expert Systems [1980s]

E.g., MYCIN was an early backward chaining **expert system** that used artificial intelligence to identify bacteria causing severe infections, such as bacteremia and meningitis, and to recommend antibiotics, with the dosage adjusted for patient's body weight — the name derived from the antibiotics themselves,



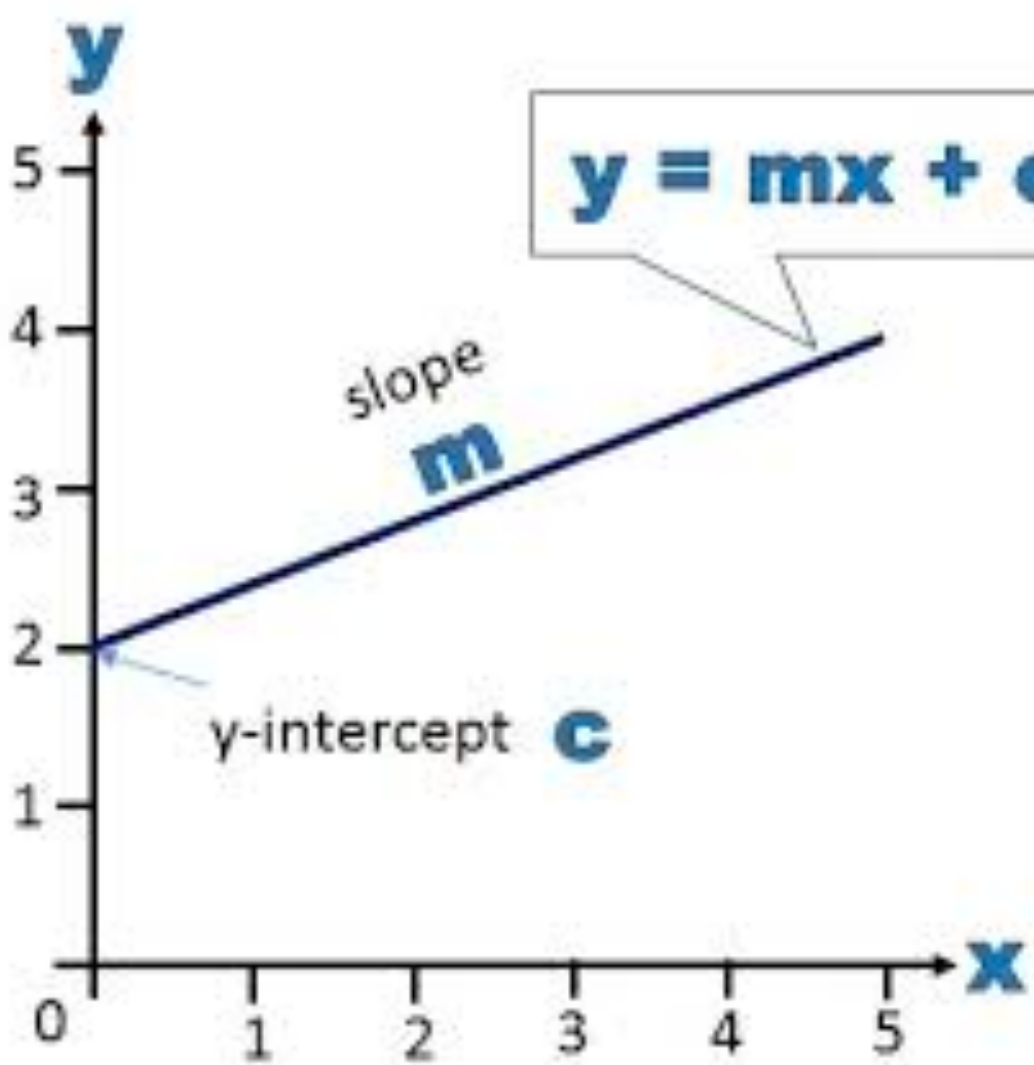
Machine Learning



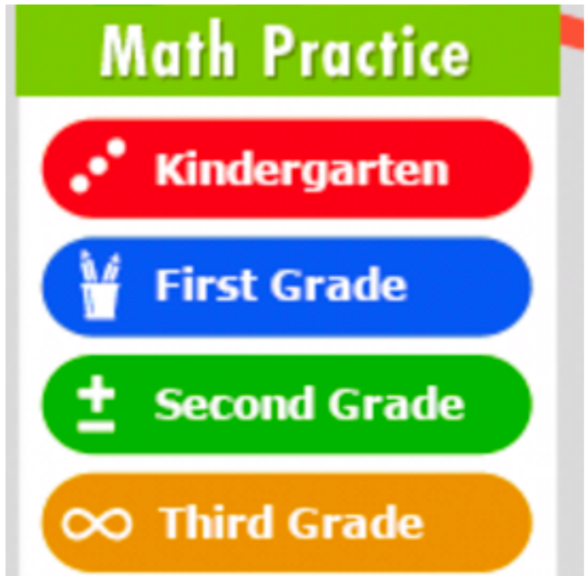
Outline

- **AI/ML 101:**
  - Introduction
  - Linear Regression
  - Beyond linear regression
- **Top AI market trends** to watch in 2017 and beyond
  - Key technical developments
  - Case studies
- **ML investor or entrepreneur**
- **Short Case Study in a Python Notebook**
- **Conclusions**
- **Course Logistics**

Equation of a line



$$f(x) = mx + b$$

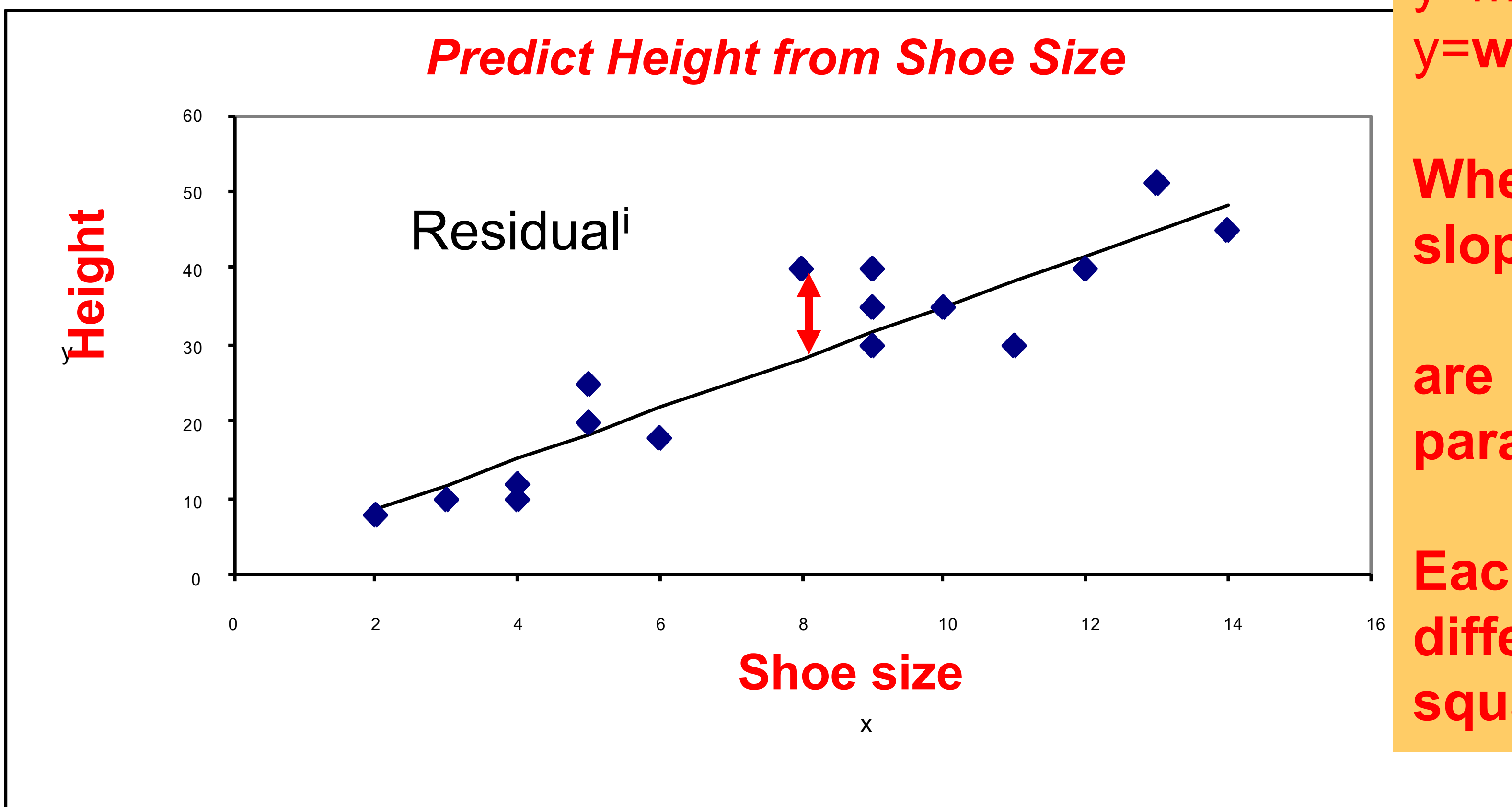


In math an Equation says 1,000 words

Equation of a line

$$\text{Residual}^i = \left(WX^i - y^i\right) \longrightarrow \text{Residual}^i = \left(WX^i - y^i\right)^2$$

Squared error loss gives us a twice differentiable function and thus we can use convex optimization



y=mx + b

y=$\mathbf{w_1}$ * x + $\mathbf{w_0}$

Where w1, w0, slope, intercept

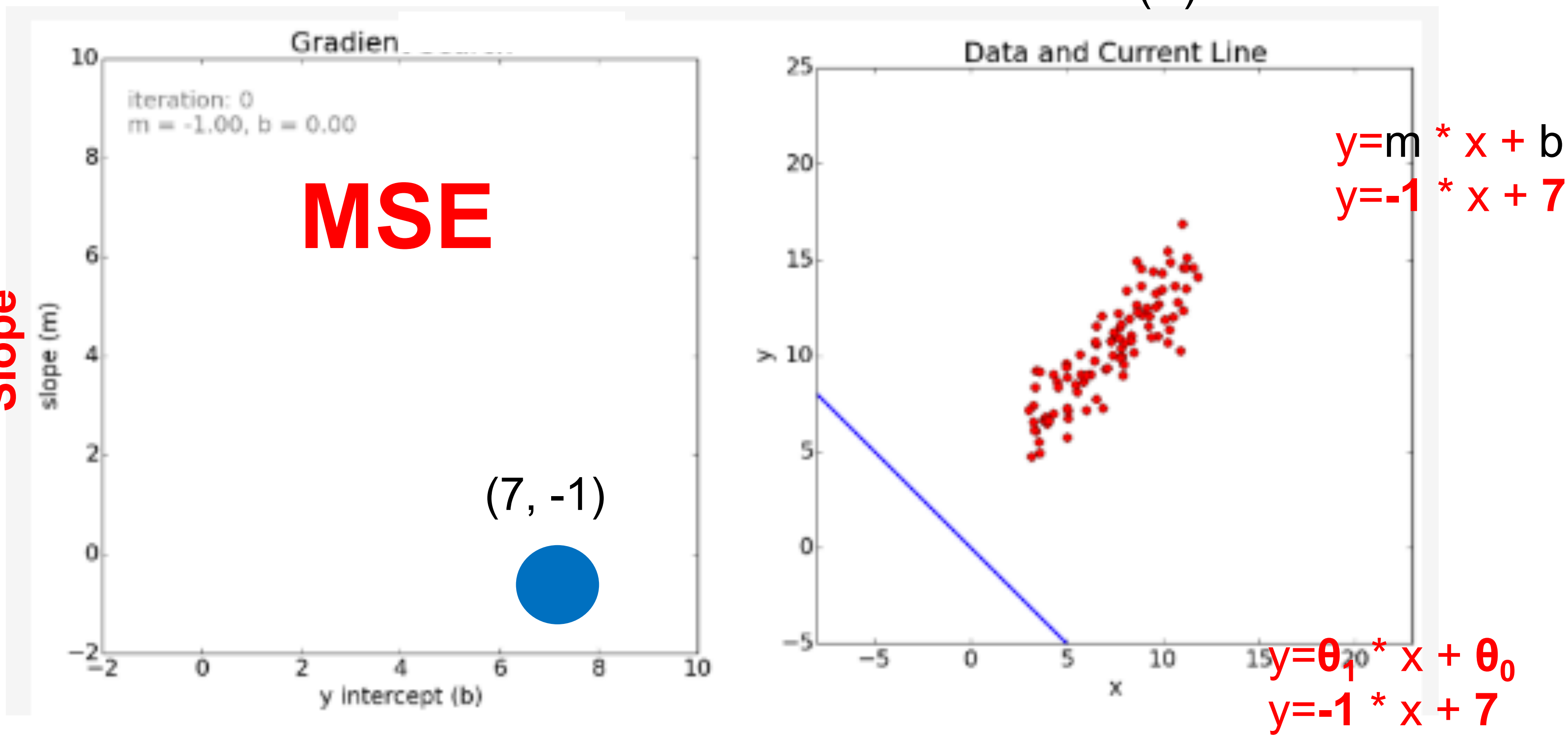
are model parameters

Each pair yields a different sum of squares error

Version Space for linear regression

Model Space
F(W) aka F(θ)

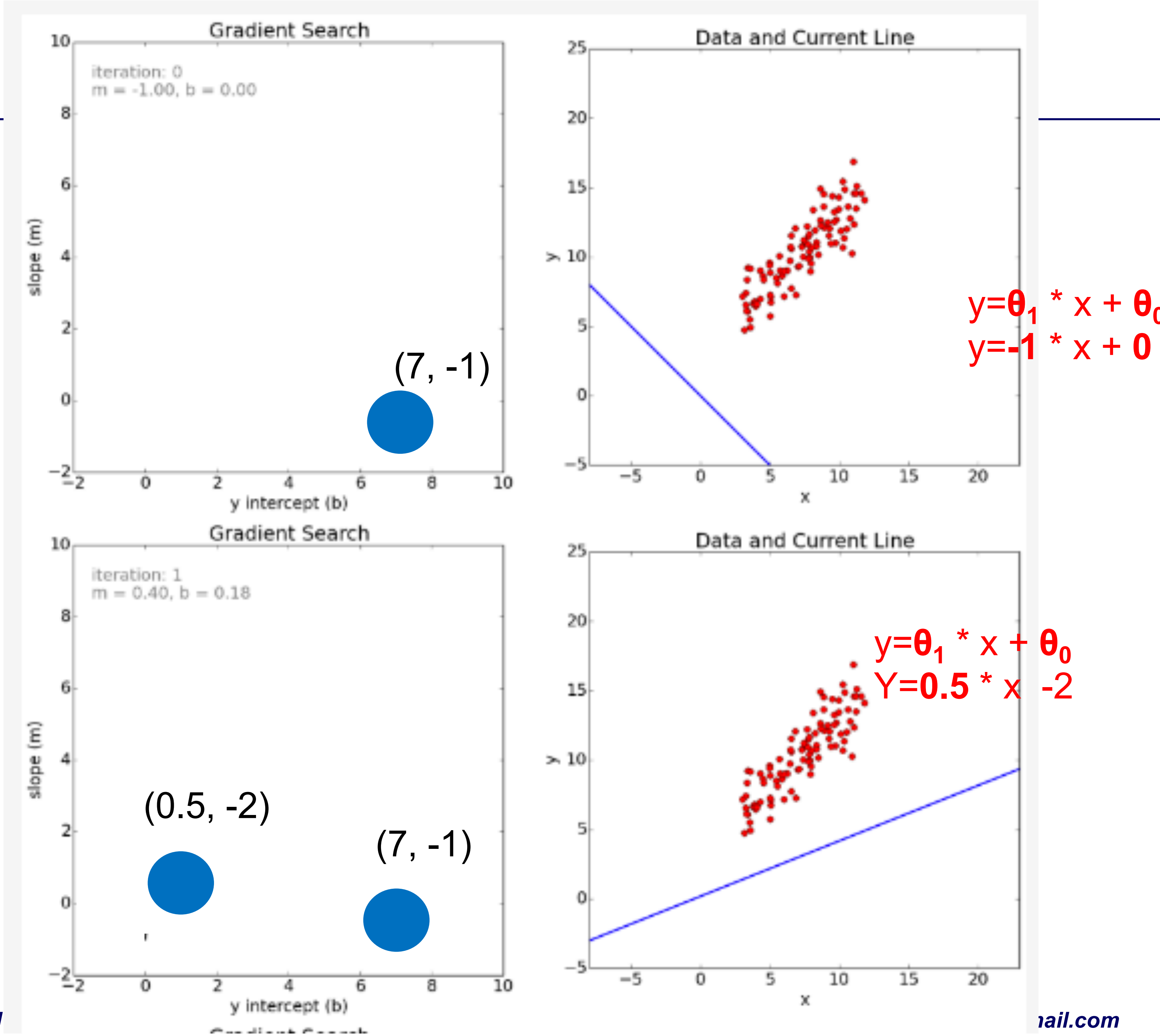
Problem Space
F(X)



Slope

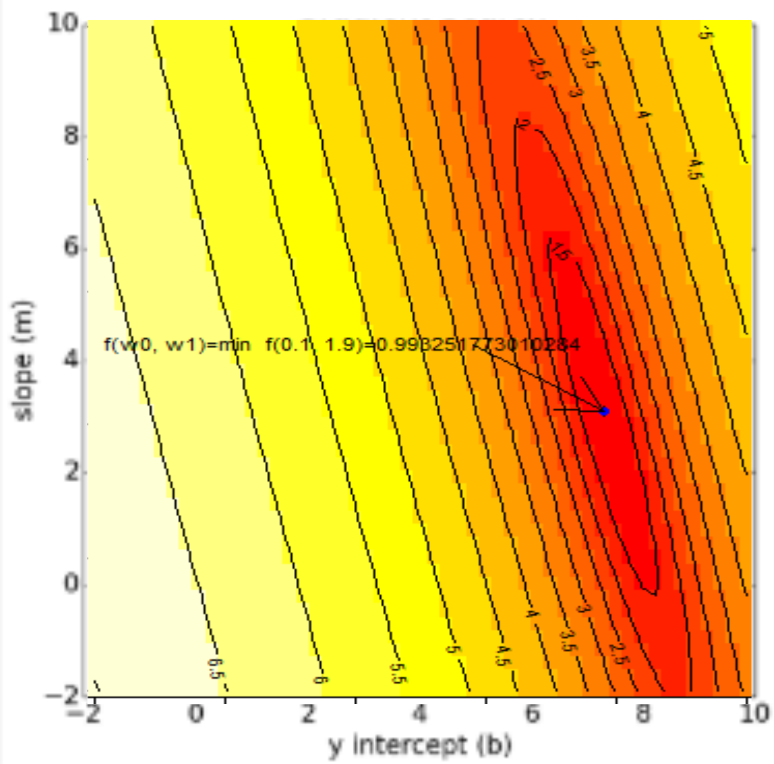
Intercept

$$MSE = \frac{1}{n}\sum \text{Residual}^{\text{i}} = \frac{1}{n}\sum \left(WX^{i} - y^{i}\right)^{2}$$

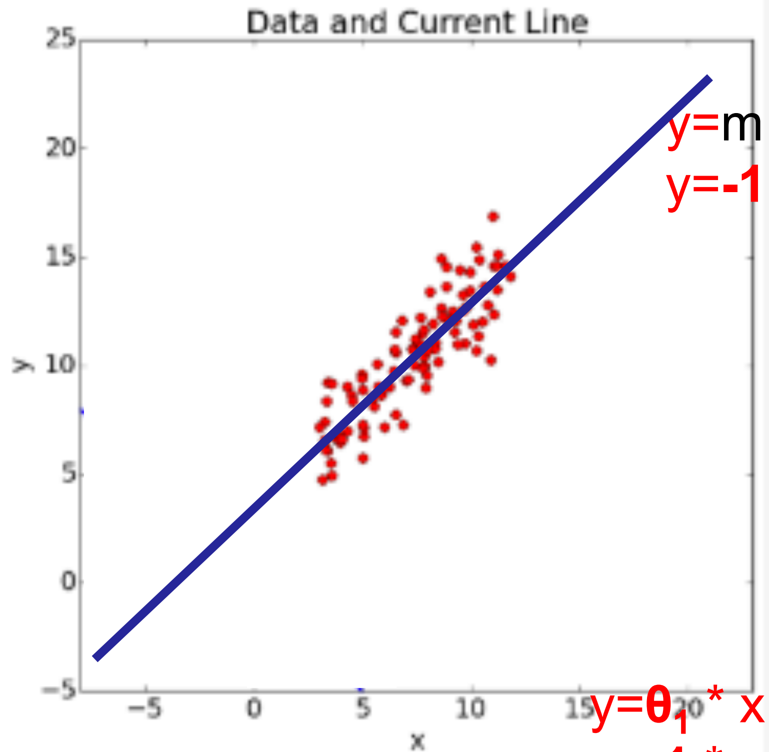

Gradient Search
iteration: 0
m = -1.00, b = 0.00
slope (m)
y intercept (b)
(7, -1)
Data and Current Line
y
x
y=θ1 * x + θ0
y=-1 * x + 0
Gradient Search
iteration: 1
m = 0.40, b = 0.18
(0.5, -2)
(7, -1)
Data and Current Line
y=θ1 * x + θ0
Y=0.5 * x -2

Plot the error for different slopes and intercepts

MSE Surface



Problem Space
F(X)

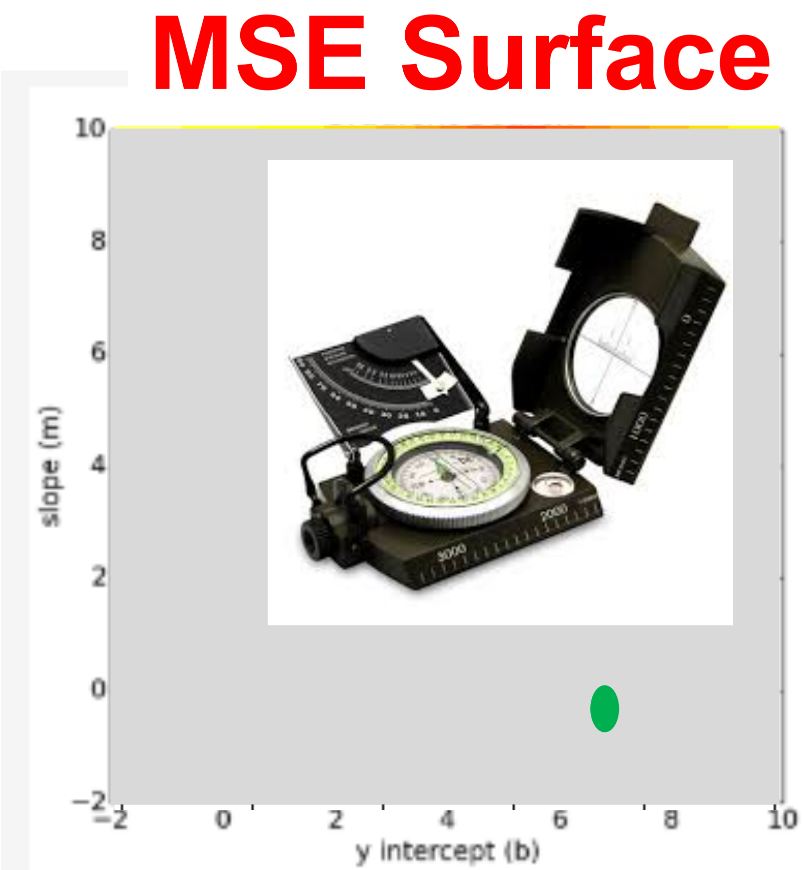


y=m * x + b
y=**-1** * x + **0**

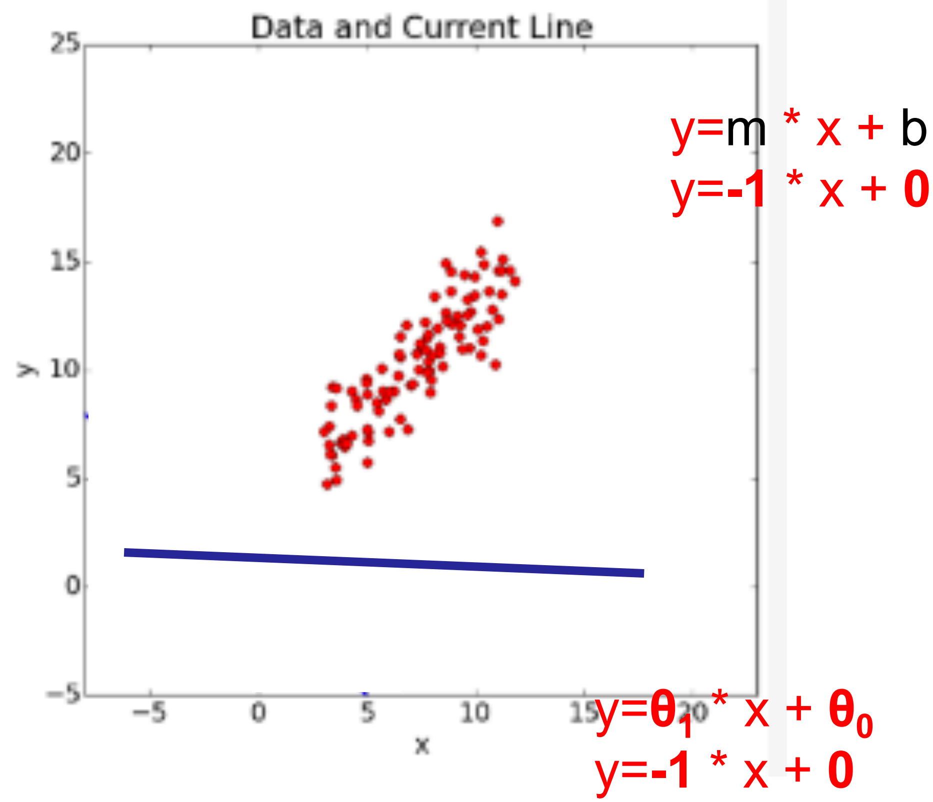
$y = \theta_1 * x + \theta_0$
y=**-1** * x + **0**

Search the MSE Error surface using Calculus (gradient descent):
Tensorflow, Caffe, Torch, Theano, etc

Gradient Compass: tells which way is up/down Without a map



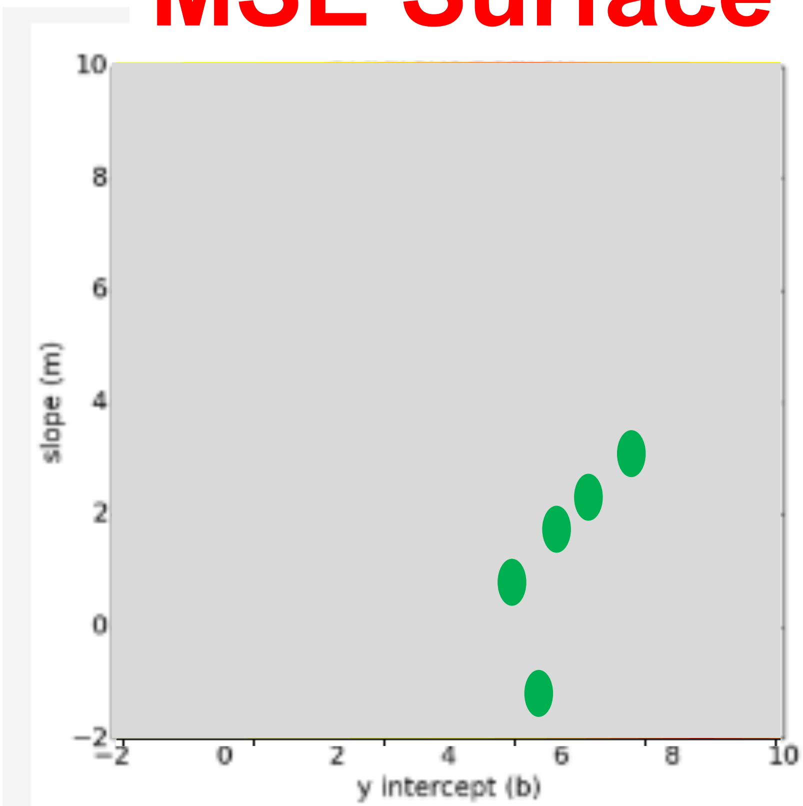
Problem Space
F(X)



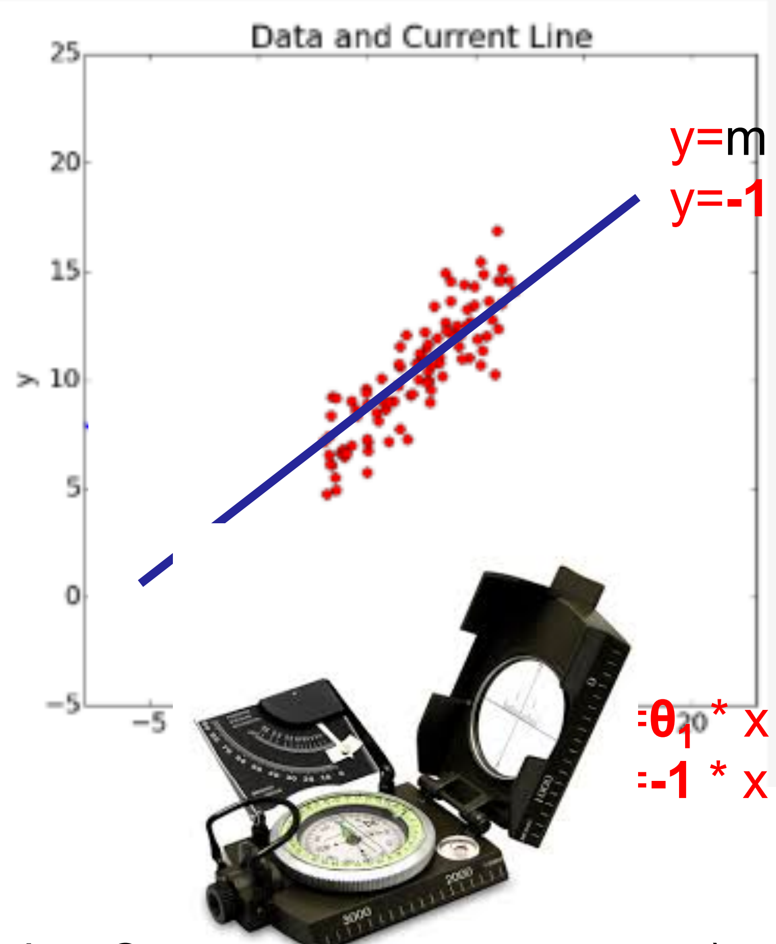
Search the MSE Error surface using Calculus (gradient descent):
Tensorflow, Caffe, Torch, Theano, etc

Using a gradient compass chase down the minimum

MSE Surface



Problem Space
F(X)



$y = m * x + b$

$y = -1 * x + 0$

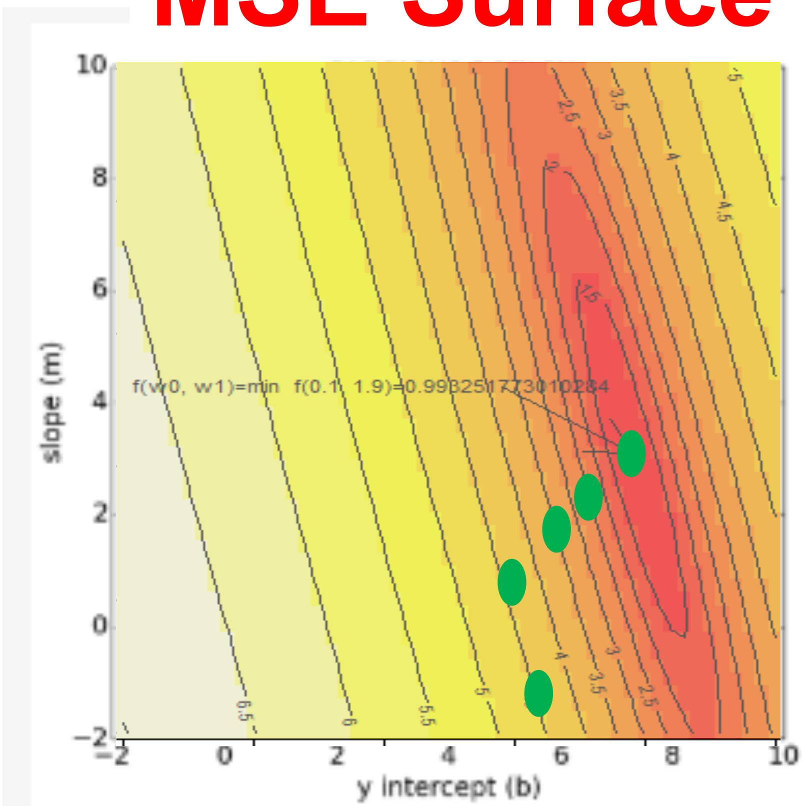
$= \theta_1 * x + \theta_0$

$= -1 * x + 0$

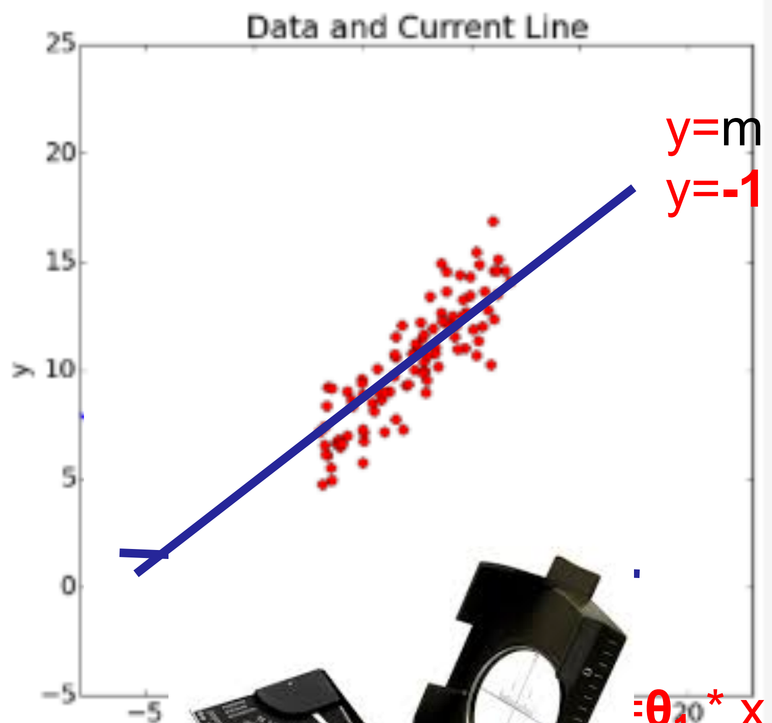
Search the MSE Error surface using Ca ent):
Tensorflow, Caffe, Torch, Theano, etc

Plot the error for different slopes and intercepts

MSE Surface



Problem Space
F(X)



y=m * x + b
y=-1 * x + 0

$=\theta_1 * x + \theta_0$
$=-1 * x + 0$

Search the MSE Error surface using Ca ent):
Tensorflow, Caffe, Torch, Theano, etc

∇ the gradient chorus

- **What is the gradient for linear regression?**

weight *example*

$$\frac{\partial E}{\partial W} = (O - t)\,X$$

$$W = W - \alpha \times \frac{\partial E}{\partial W}$$

$$\begin{bmatrix} 4.32 \\ 3.98 \\ 7.92 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \\ 8 \end{bmatrix} - 0.01 \times 2 \times \begin{bmatrix} 34 \\ 1 \\ 4 \end{bmatrix}$$

$$4.32 = 5 - 0.01 \times (2 \times 34) \text{ for } i = 1$$

- **Chorus**
  - The gradient is the weighted sum of the training data, where the weights are proportional to the error (for each example) !

Machine Learning: Regression

Machine Learning (ML):"*a computer program that improves its performance at some task through experience*" [Mitchell 1997]

GIVEN: Input data is a table of attribute values and associated class values (in the case of supervised learning)
GOAL: Approximate *f(x₁,…,xₙ)->y*

Mimimize $$MSE = \frac{1}{n}\sum \mathrm{Residual}^{\mathrm{i}} = \frac{1}{n}\sum \left(WX^{i} - y^{i}\right)^{2}$$ **Y is real valued**

Instance\Attr	x_1	x_2	…	x_n	*y*
1	3	0	..	7	73
2					76
…	…	…	…	…	…
L (aka m)	0	4	…	8	97

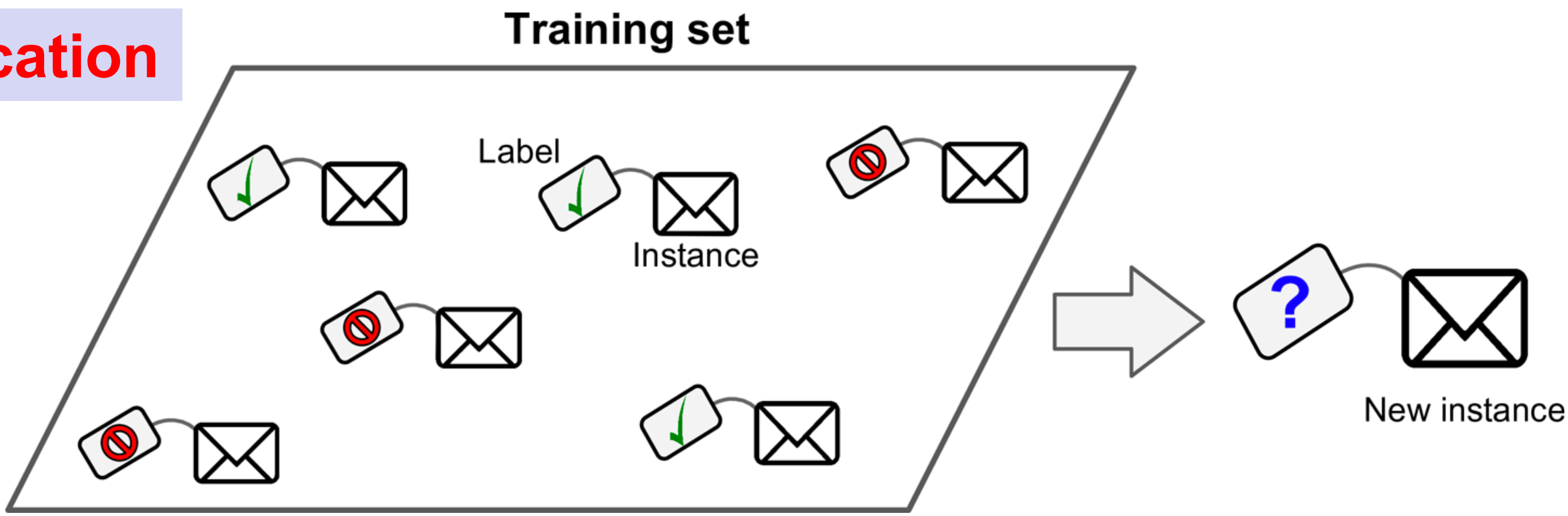
ML in a Nutshell

- **Tens of thousands of machine learning algorithms**
- **Hundreds new every year**
- **Every machine learning algorithm has three components:**
  - **Representation:** ***Slope, intercept***
  - **Objective function** $$\mathrm{MSE}(\mathbf{X}, h_\theta) = \frac{1}{m}\sum_{i=1}^{m}\left(\theta^T \cdot \mathbf{x}^{(i)} - y^{(i)}\right)^2$$
  - **Optimization**
    - Random
    - Brute force
    - Numerical: Gradient descent
    - Analytical: Closed from via Normal Equation

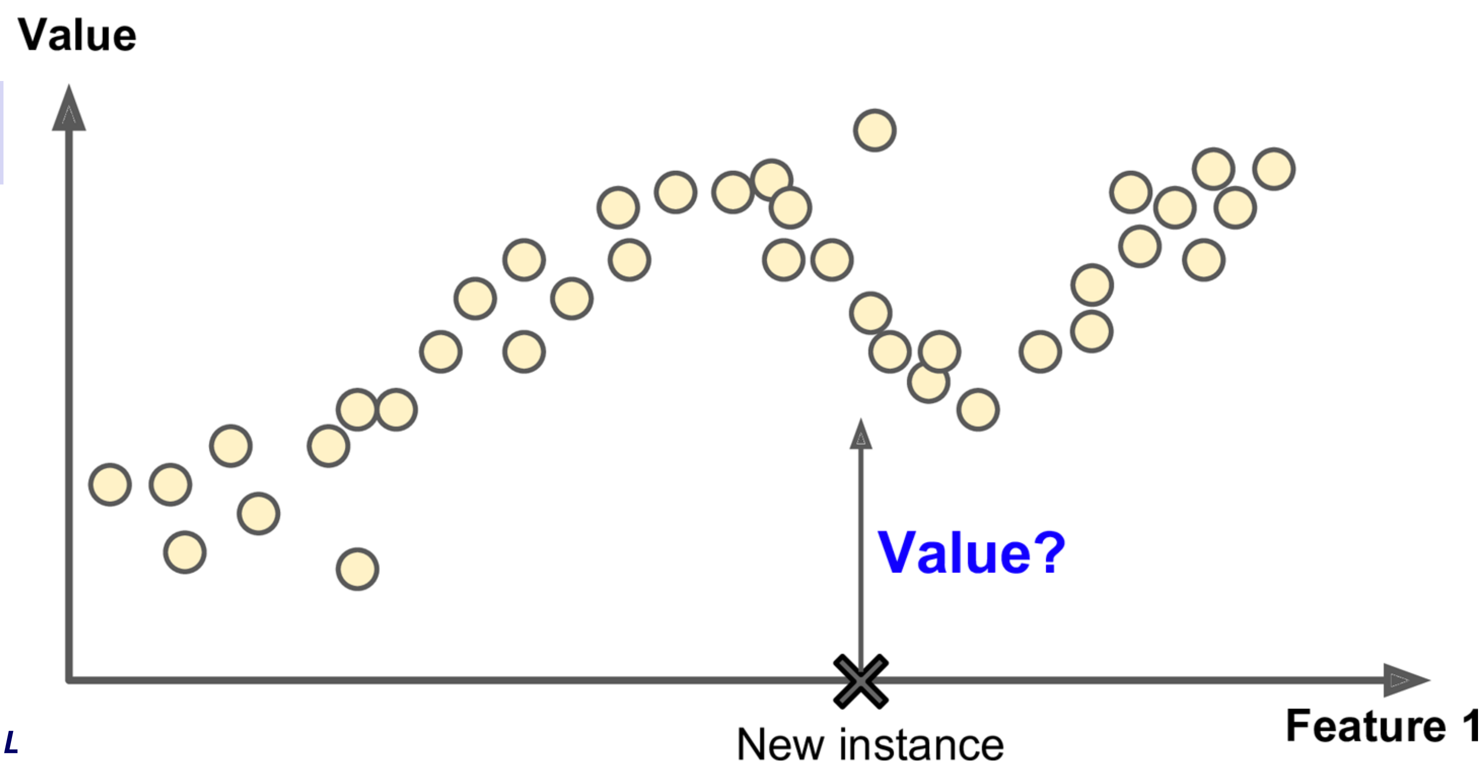
Outline

- **AI/ML 101:**
  - Introduction
  - Linear Regression
  - Beyond linear regression
- **Top AI market trends** to watch in 2017 and beyond
  - Key technical developments
  - Case studies
- **ML investor or entrepreneur**
- **Short Case Study in a Python Notebook**
- **Conclusions**
- **Course Logistics**

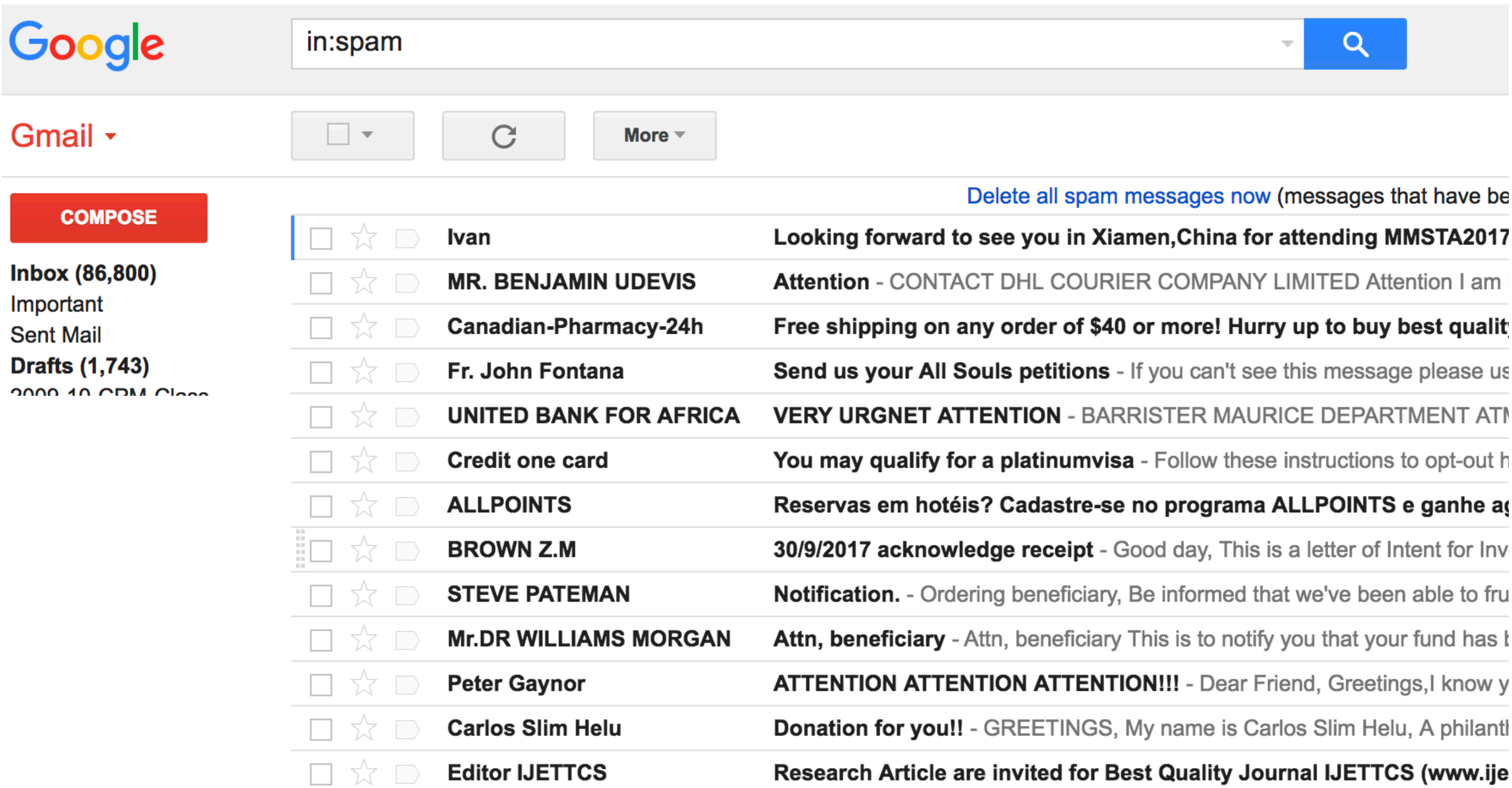
Classification



Regression



Email: SPAM versus Ham



Learning from tagged data (supervised)

Coffee mug

Coffee mug

Coffee mug

Coffee mug

Coffee mug

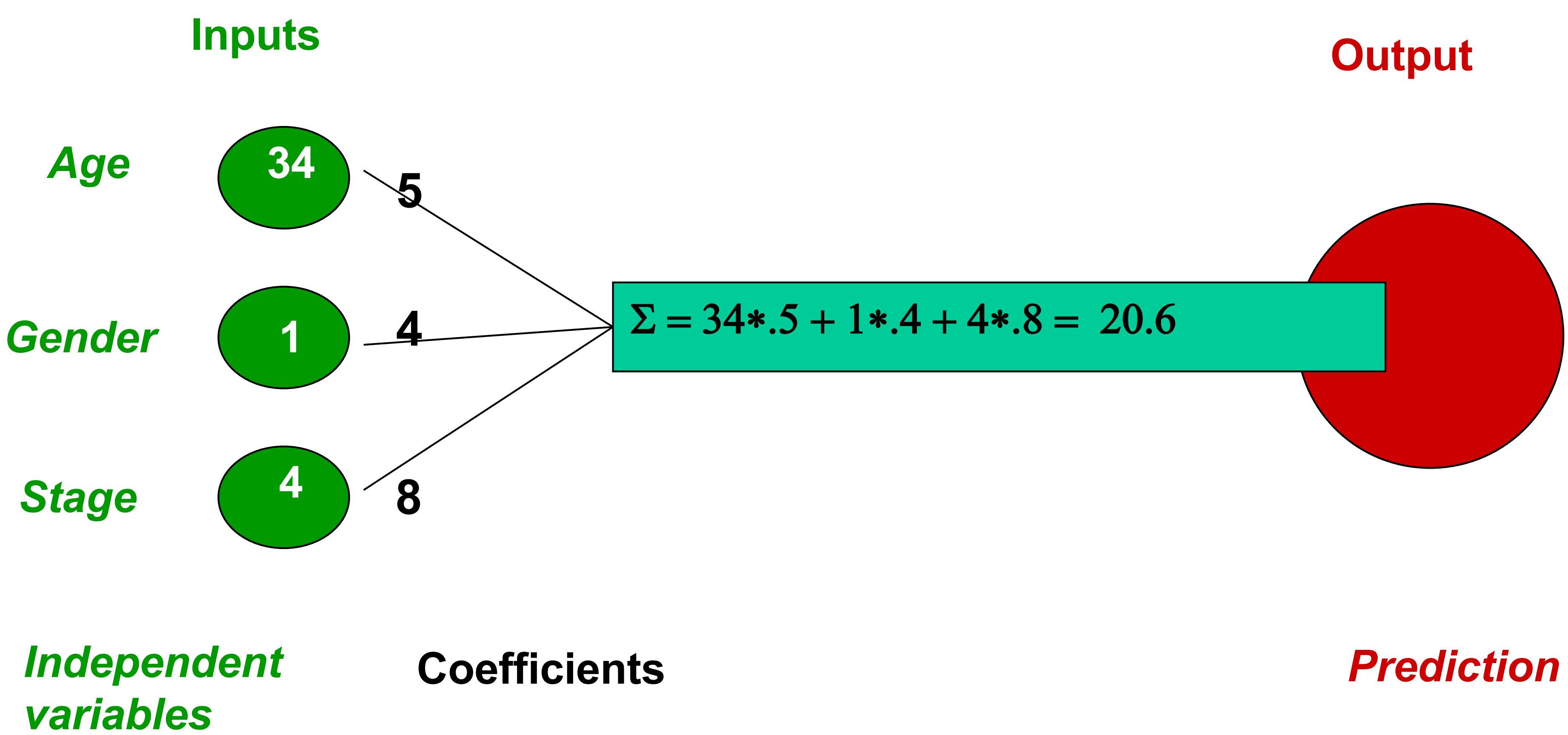
Coffee mug

Testing: What is this?

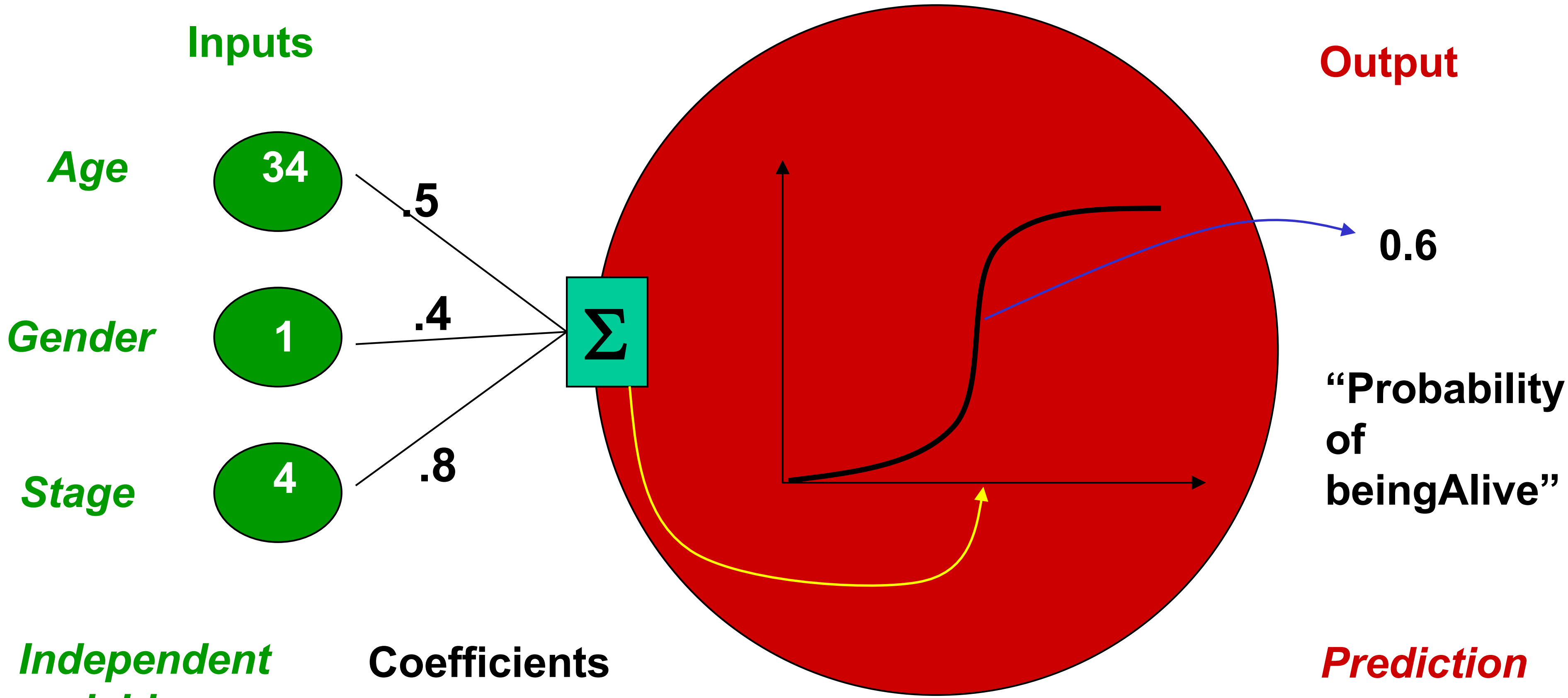
Supervised Learning Algorithms

- k-Nearest Neighbors
- Linear Regression
- Logistic Regression
- Support Vector Machines (SVMs)
- Decision Trees and Random Forests
- Neural networks→ deep learning

Σ is the sum of inputs * weights

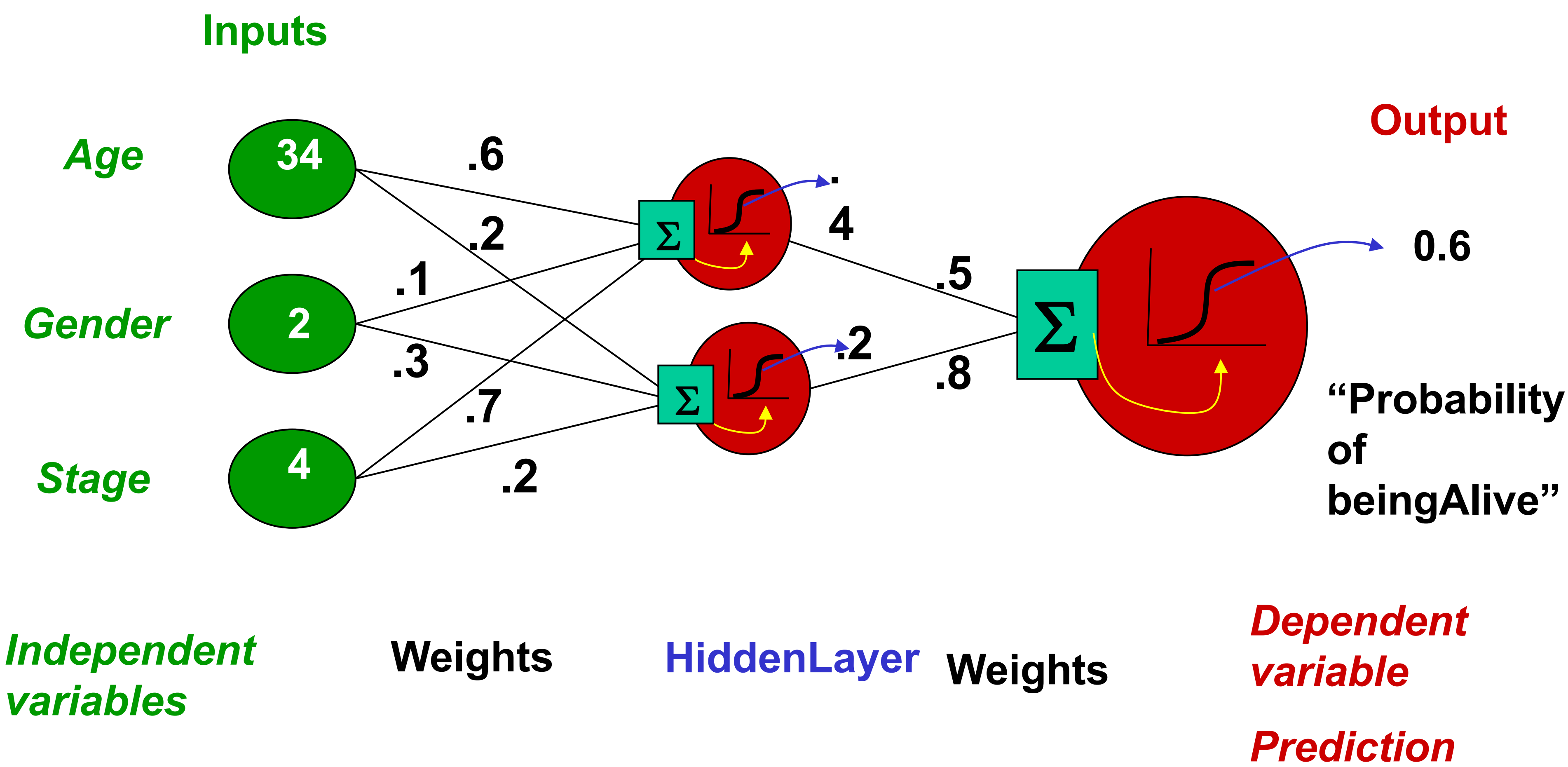


Logistic function

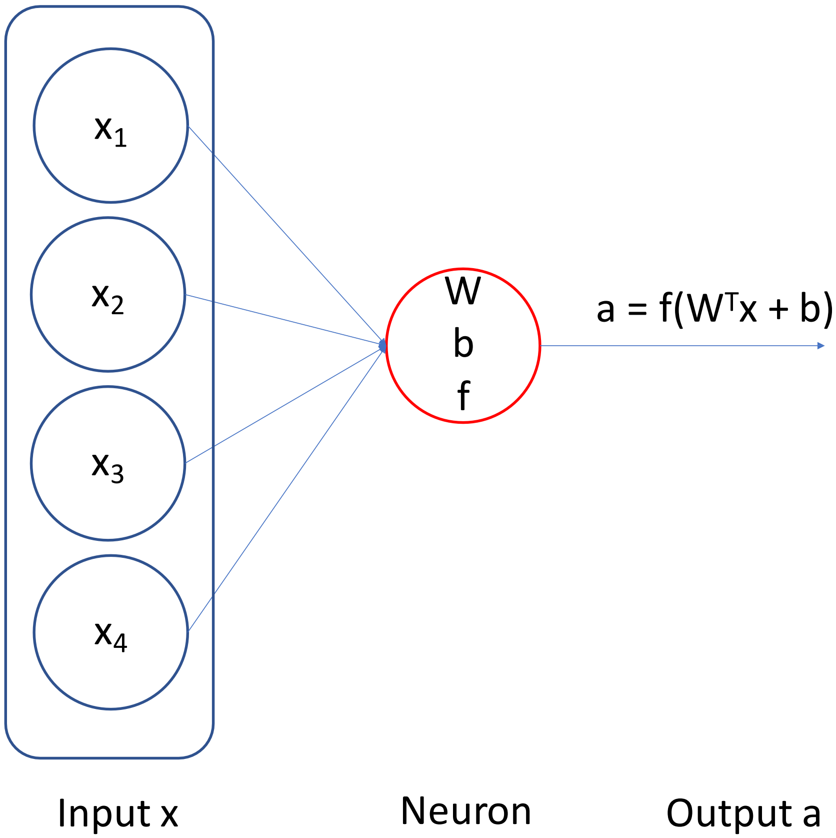


$$p = \frac{1}{1 + e^{-XW}}$$

Neural Network Model



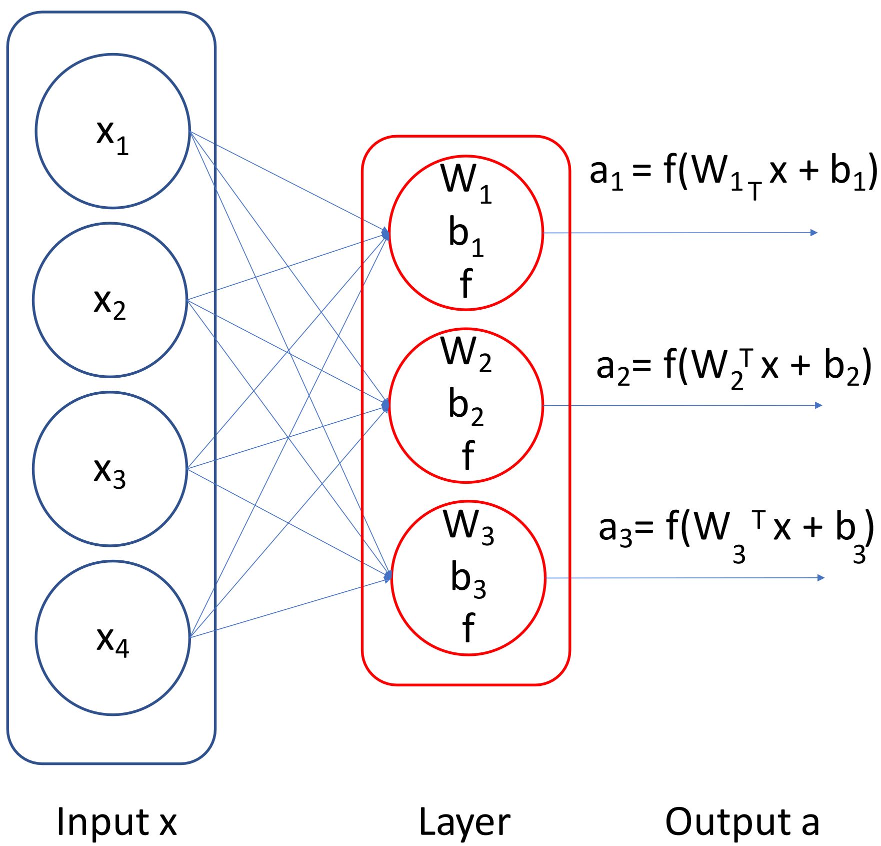
Neuron: Computational Unit



- Input vector: $x = [x_1, x_2, \dots, x_n]$
- Neuron
  - Weight vector: W
  - Bias: b
  - Activation function: f
- Output

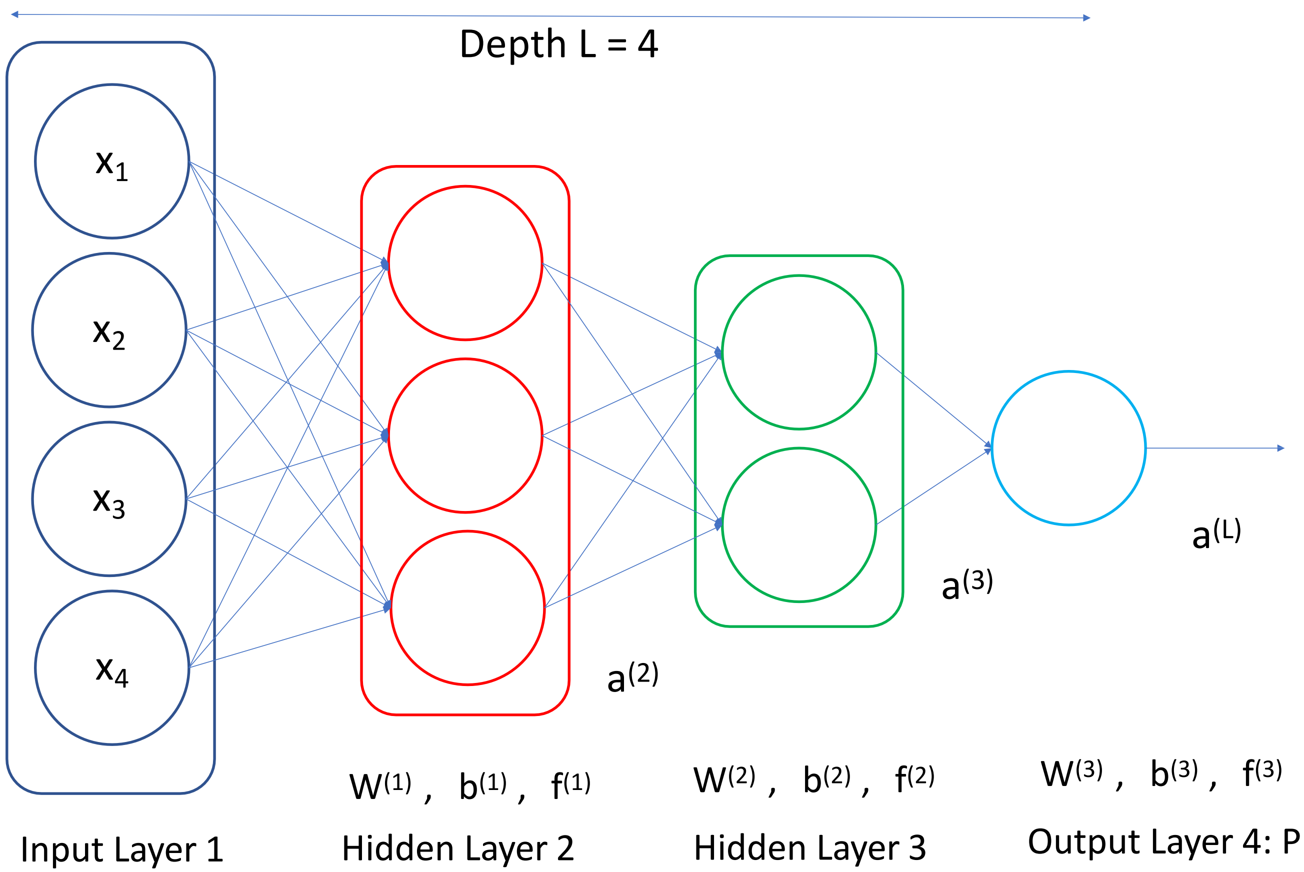
$$a = f(W^T x + b)$$

Layer



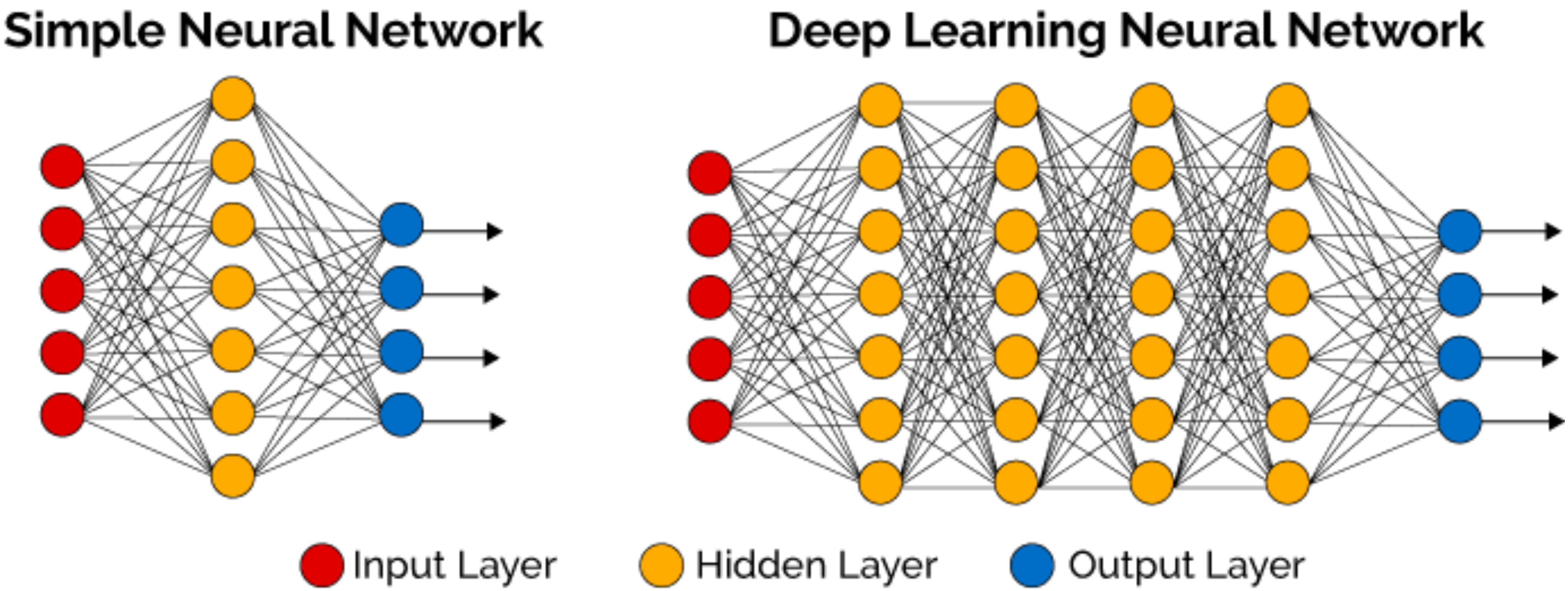
- Layer l: n_l neurons
  - weight matrix: $W = [W_1, \ldots, W_{nl}]$
  - bias vector: $b = [b1, \ldots, b_{nl}]$
  - activation function: f
- output vector
  - $a = f(W^T x + b)$

Feed Forward Network

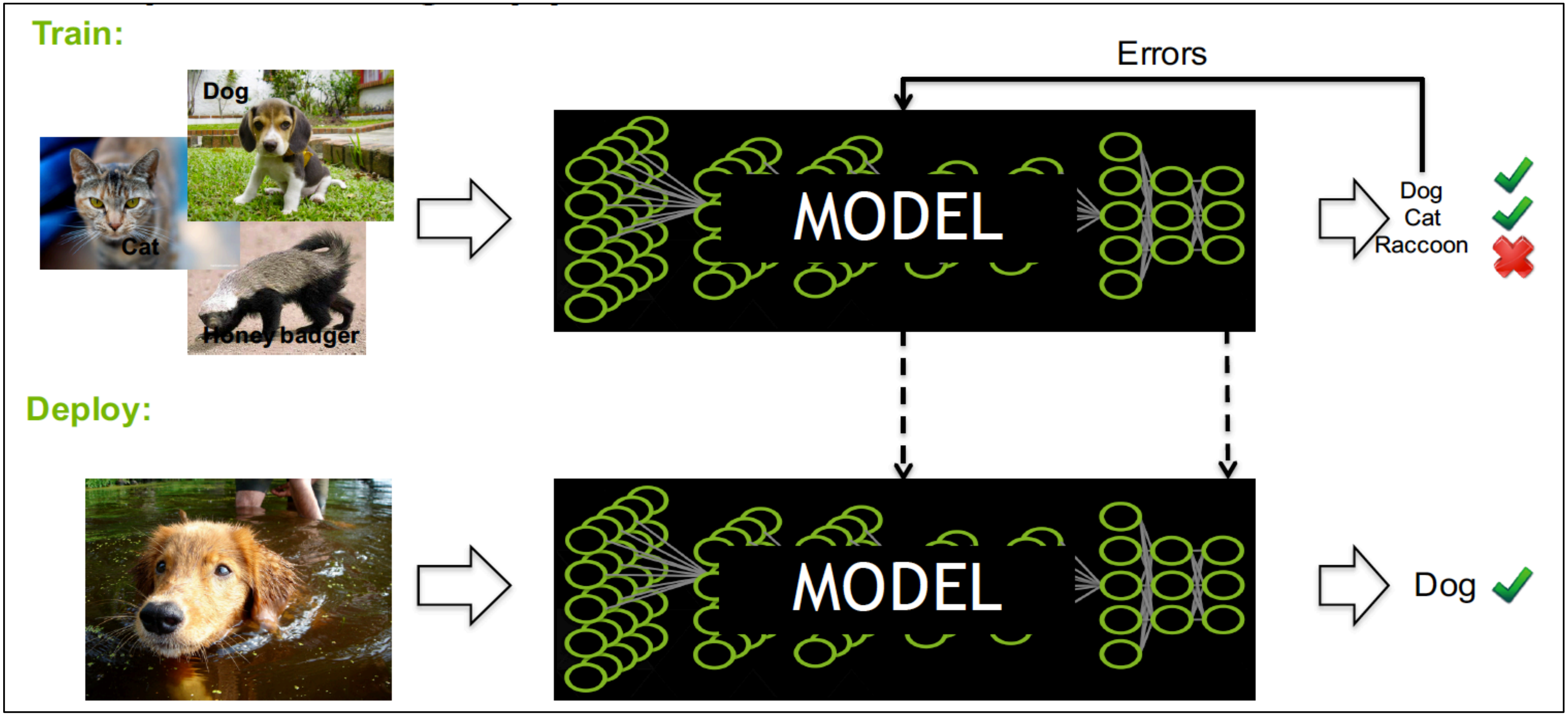


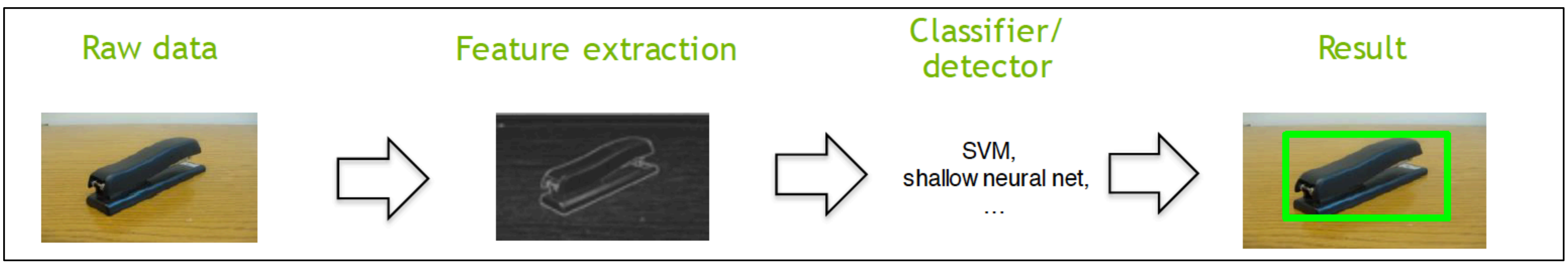
- Depth L layers
- Activation at layer l+1 $a^{(l+1)} = f(W^{(l)T} a^{(l)} + b^{(l)})$
- Output: prediction in supervised learning
- goal: approximate y = F(x)

Deeper neural networks

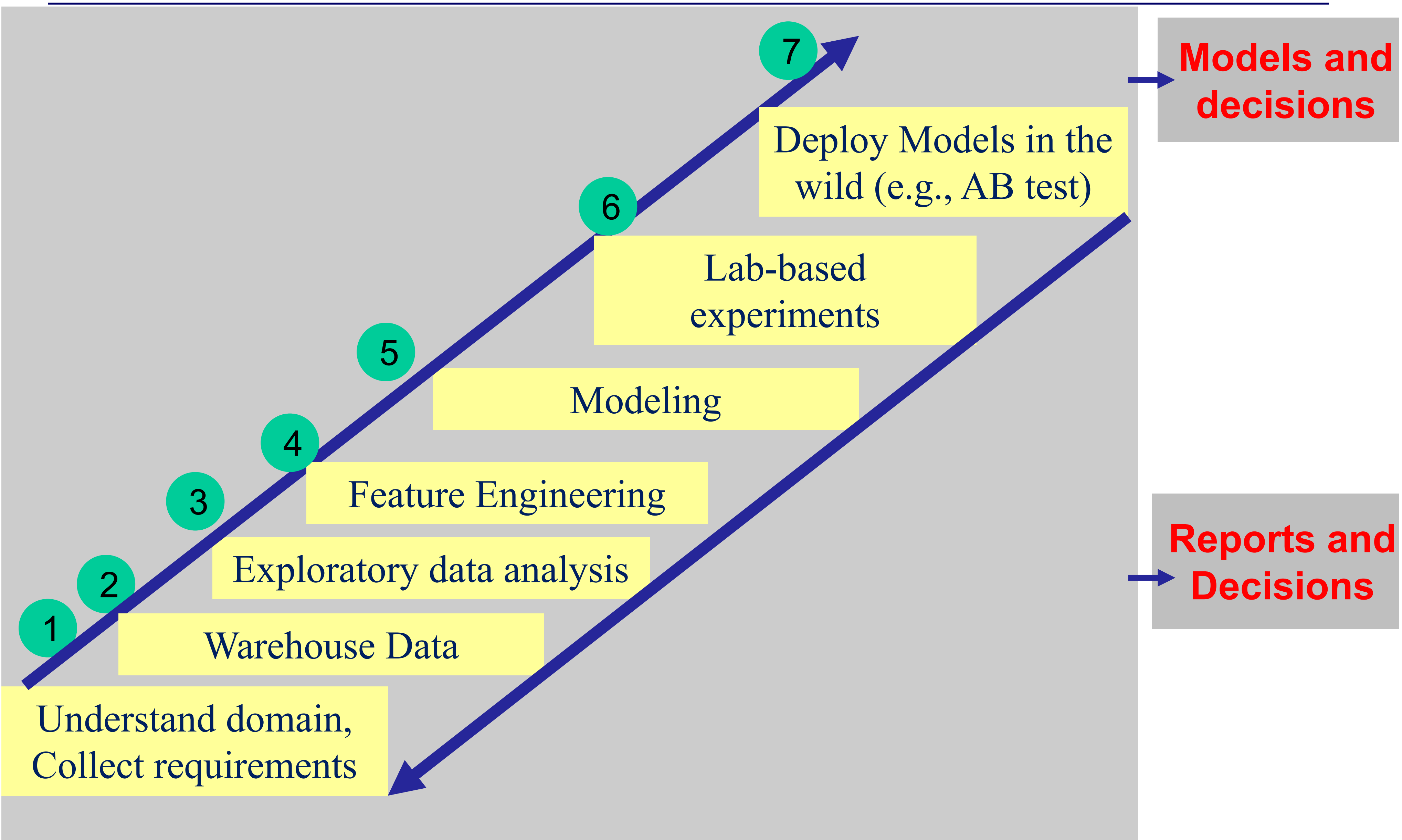


Deep Learning vs Machine Learning

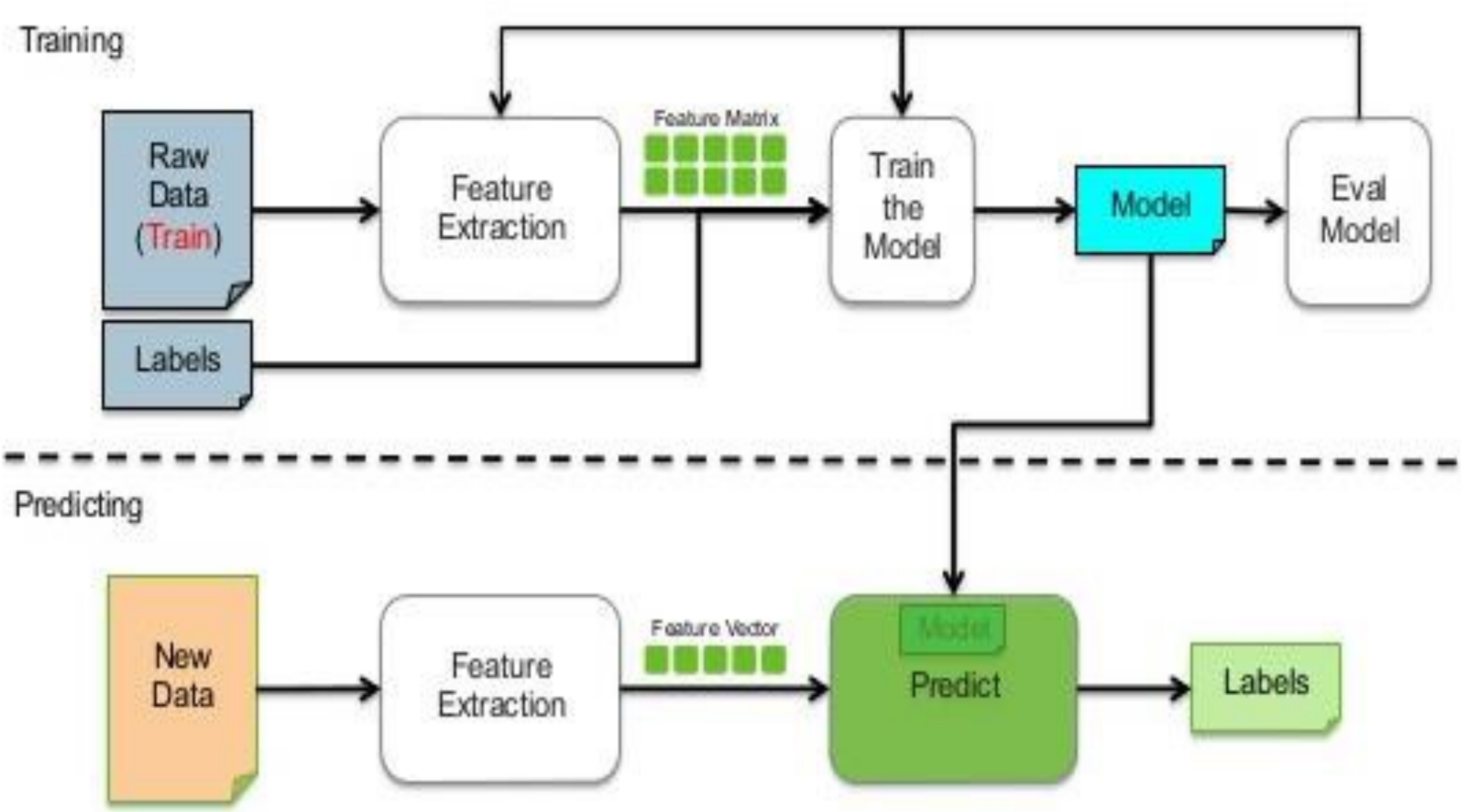




Typical Abstract Data Analytics Pipeline



Machine Learning Pipeline

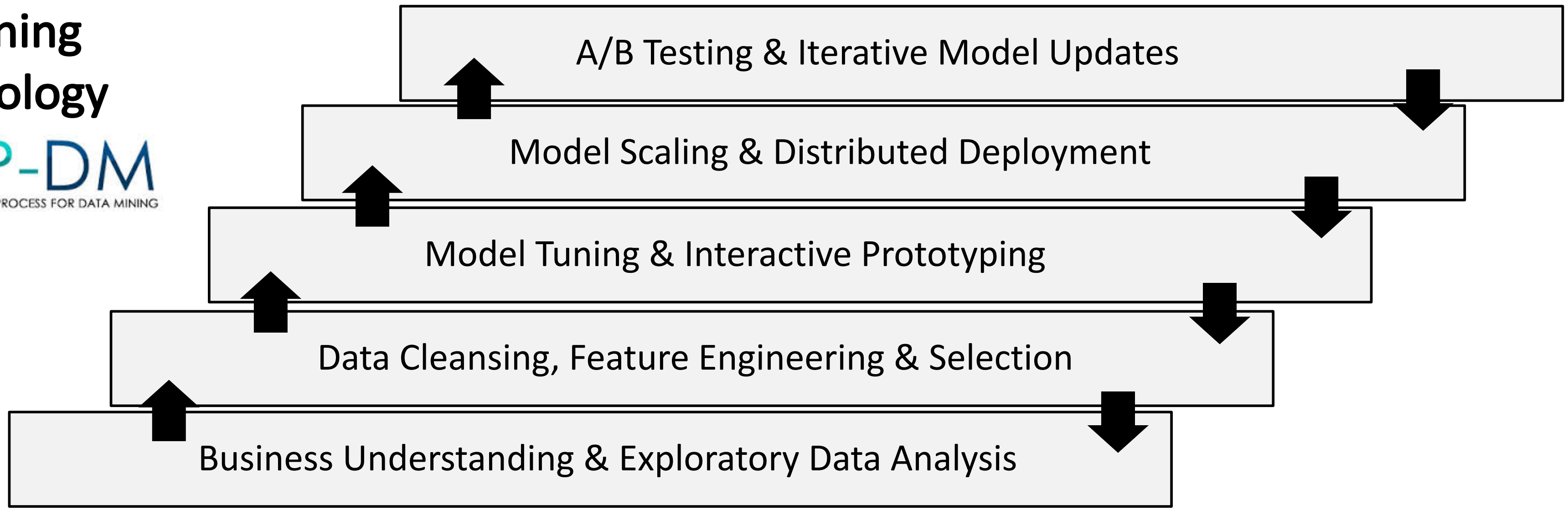


Data Mining

Data Mining Methodology

CRISP-DM

CROSS INDUSTRY STANDARD PROCESS FOR DATA MINING



Full-stack approach to data science

Data Mining Methodology

CRISP-DM

CROSS INDUSTRY STANDARD PROCESS FOR DATA MINING

- A/B Testing & Iterative Model Updates
- Model Scaling & Distributed Deployment
- Model Tuning & Interactive Prototyping
- Data Cleansing, Feature Engineering & Selection
- Business Understanding & Exploratory Data Analysis

AI & Data Applications

python™ jupyter R Scala scikit learn

Big Data

APACHE Spark™ hadoop

Deep Learning

TensorFlow K

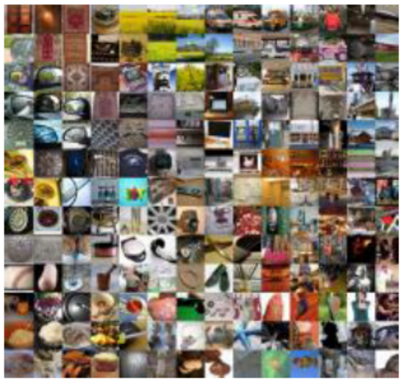
Cloud Computing

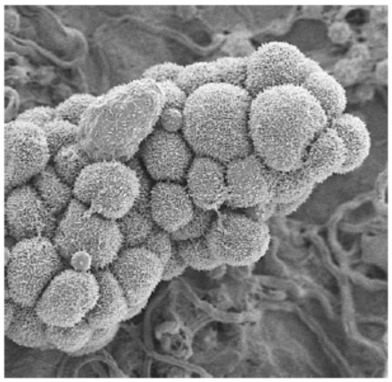
NVIDIA. Microsoft Azure amazon web services docker

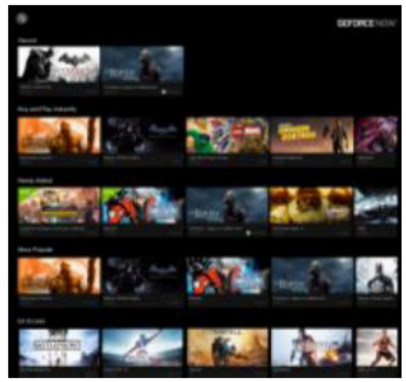
Outline

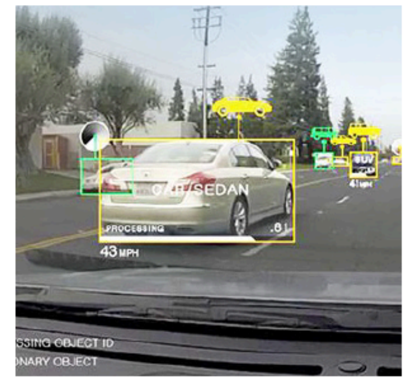
- **AI/ML 101:**
  - Introduction
  - Linear Regression
  - Beyond linear regression
- **Top AI market trends** to watch in 2017 and beyond
  - Key technical developments
  - Case studies
- **ML investor or entrepreneur**
- **Short Case Study in a Python Notebook**
- **Conclusions**
- **Course Logistics**

AI and Data science everywhere









INTERNET & CLOUD

Image Classification
Speech Recognition
Language Translation
Language Processing
Sentiment Analysis
Recommendation

MEDICINE & BIOLOGY

Cancer Cell Detection
Diabetic Grading
Drug Discovery

MEDIA & ENTERTAINMENT

Video Captioning
Video Search
Real Time Translation

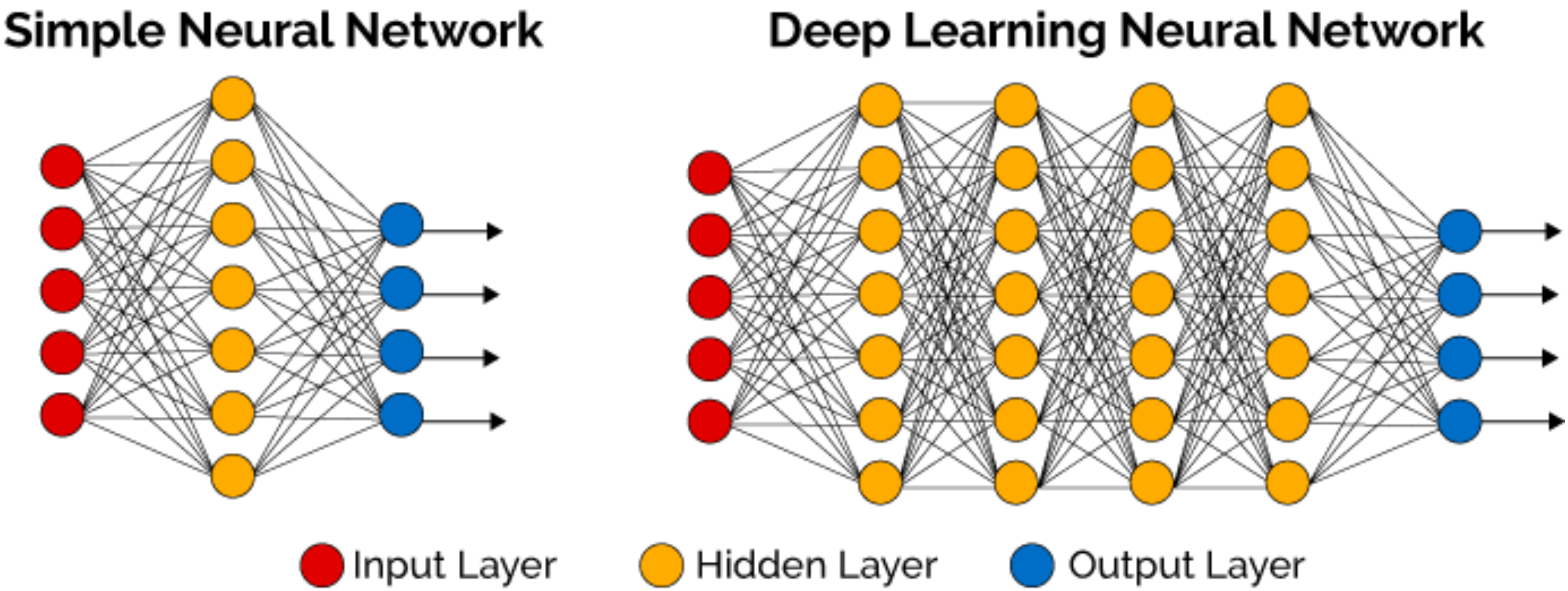
SECURITY & DEFENSE

Face Detection
Video Surveillance
Satellite Imagery

AUTONOMOUS MACHINES

Pedestrian Detection
Lane Tracking
Recognize Traffic Sign

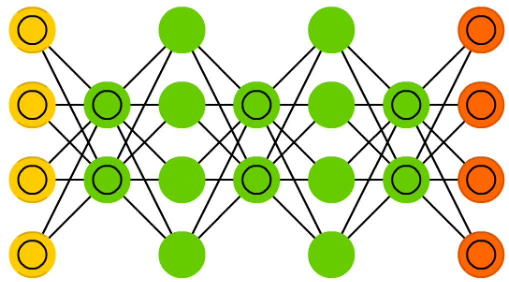
Deeper neural networks



GPU provide a 10X Speedup over CPU

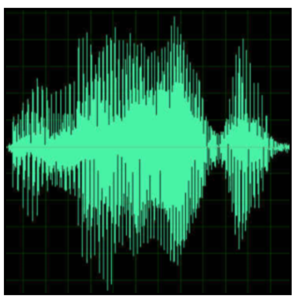
Some cool things that we can do with data

image

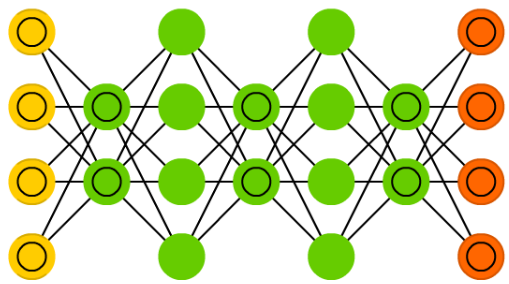


model

Label or generate a description for an image



audio signal



model

Speech transcription and emotion recognition


THE TIMES

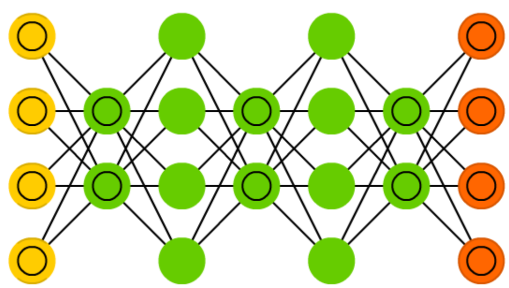
Football League

News | Opinion | Business | Money | Sport | Life | Arts

Football League strugglers want to sign rapist Ched E

Article | Match Centre

news article

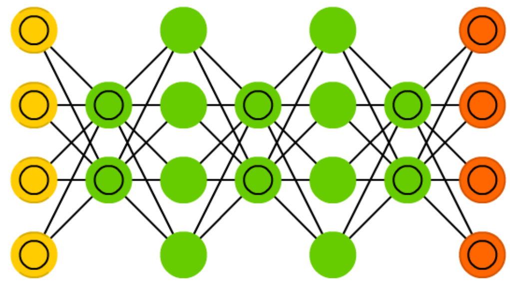


model

Extract entities, detect topics, find similar articles

Some cool things that we can do with data

network →

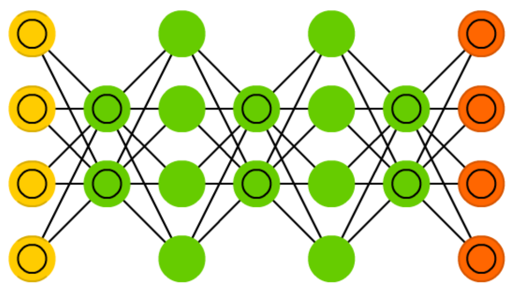


CHURCH & DUNCAN Group Inc.

model →

Detect anomalous behavior, fraud, and clusters

order history →

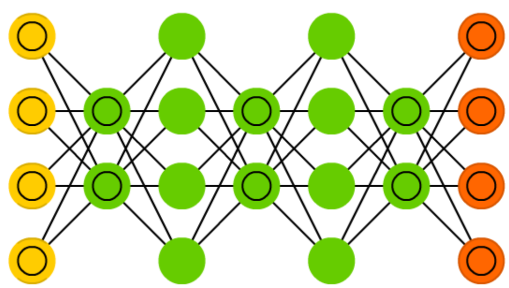


CHURCH & DUNCAN Group Inc.

model →

Suggest a personalized offer/coupon

chat question →

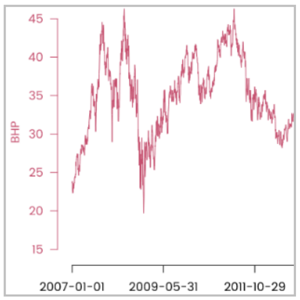


CHURCH & DUNCAN Group Inc.

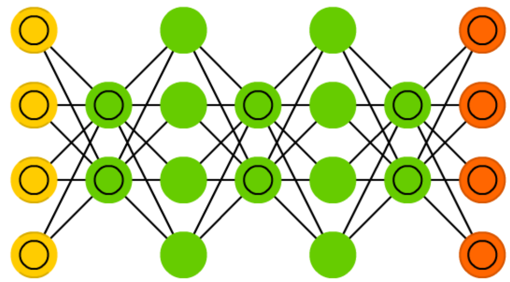
model →

Answer the question, offer extra service, simple AI dialog

Some cool things that we can do with data



time series →

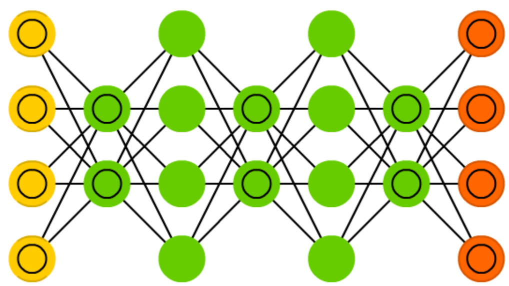


CHURCH & DUNCAN Group Inc.

model →

Predict demand, detect outliers, extract patterns

browse log →

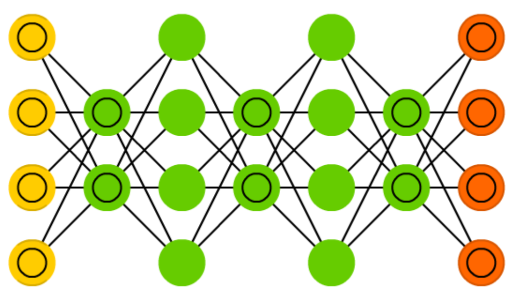


CHURCH & DUNCAN Group Inc.

model →

Decide what native/web ad to show

bank history →



CHURCH & DUNCAN Group Inc.

model →

Decide who to give a loan or how to price insurance

Outline

- **AI/ML 101:**
  - Introduction
  - Linear Regression
  - Beyond linear regression
- **Top AI market trends** to watch in 2017 and beyond
  - Key technical developments
  - Case studies
- **ML investor or entrepreneur**
- **Short Case Study in a Python Notebook**
- **Conclusions**
- **Course Logistics**

Case Studies

- **Advertising and marketing**
- **Playing games**
- **Autonomous vehicles**
- **Consumer**
  - Image tagging
  - Personalized shopping assistance (build vs buy)
- **Automatic stock market trading: Numer.ai**
- **Social Good**
  - Counting Animals in Flickr vs in The Wild

Mobile Ad Spend to Top $100 Billion Worldwide in 2016, 51% of Digital Market

- **US and China will account for nearly 62% of global mobile ad spending next year**

Mobile Internet Ad Spending Worldwide, 2013-2019

	2013	2014	2015	2016	2017	2018	2019
Mobile internet ad spending (billions)	**$19.20**	**$42.63**	**$68.69**	**$101.37**	**$133.74**	**$166.63**	**$195.55**
—% change	117.9%	122.1%	61.1%	47.6%	31.9%	24.6%	17.4%
—% of digital ad spending	16.0%	29.4%	40.2%	51.1%	59.4%	65.9%	70.1%
—% of total media ad spending	3.7%	7.8%	11.9%	16.5%	20.5%	24.1%	26.8%

Note: includes display (banners, video and rich media) and search; excludes SMS, MMS and P2P messaging-based advertising; ad spending on tablets is included
Source: eMarketer, March 2015

186887 www.eMarketer.com

- http://www.emarketer.com/Article/Mobile-Ad-Spend-Top-100-Billion-Worldwide-2016-51-of-Digital-Market/1012299#sthash.FBfZAIaC.dpuf

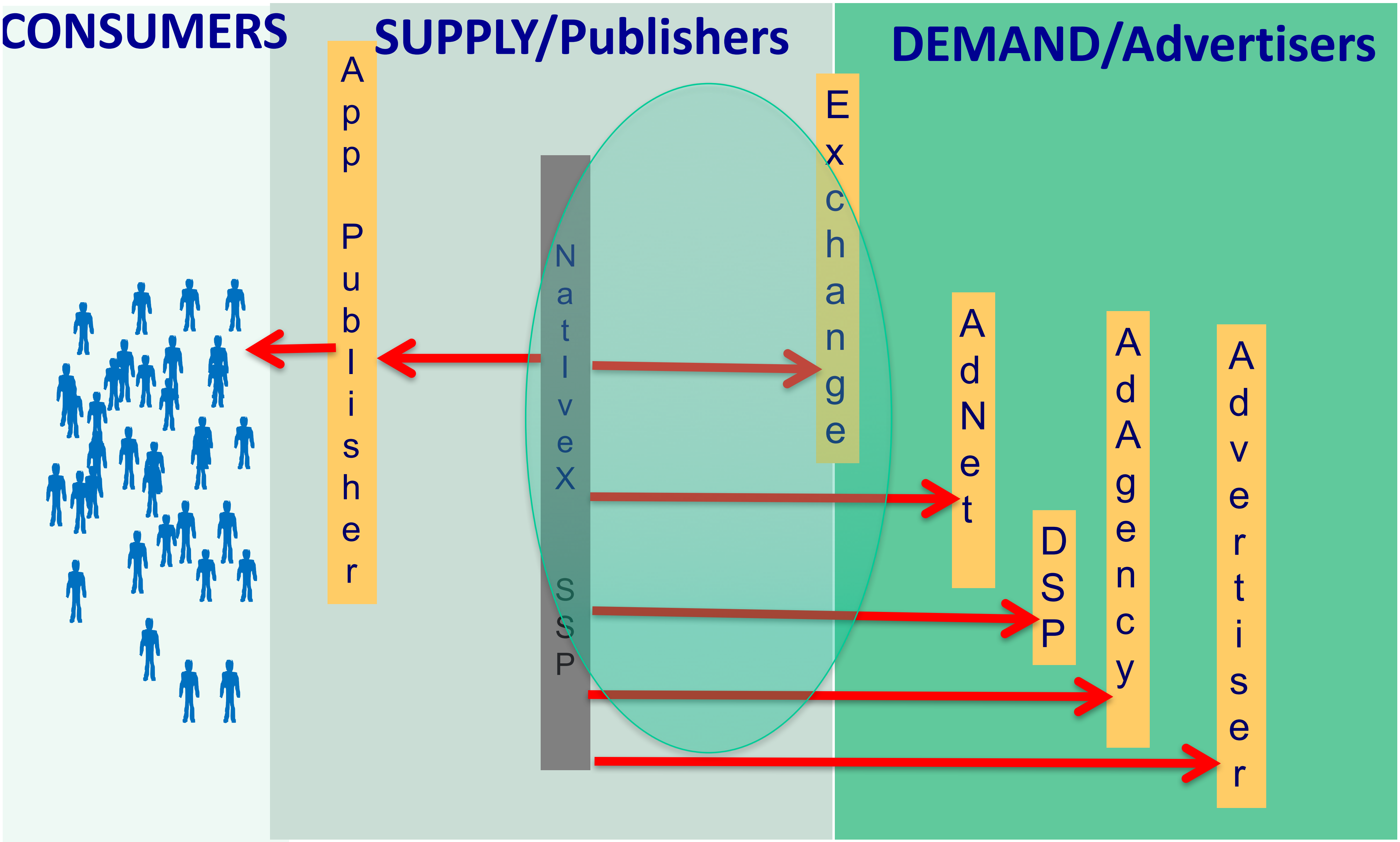
CPMs on Mobile are catching up

- **Mobile Advertising: What is the average CPM on mobile?**
  - The effective cost per thousand impressions (CPM) for desktop web ads is about $3.50, while the CPM for mobile ads is just $0.75.
  - Video-based CPMs typically > $15

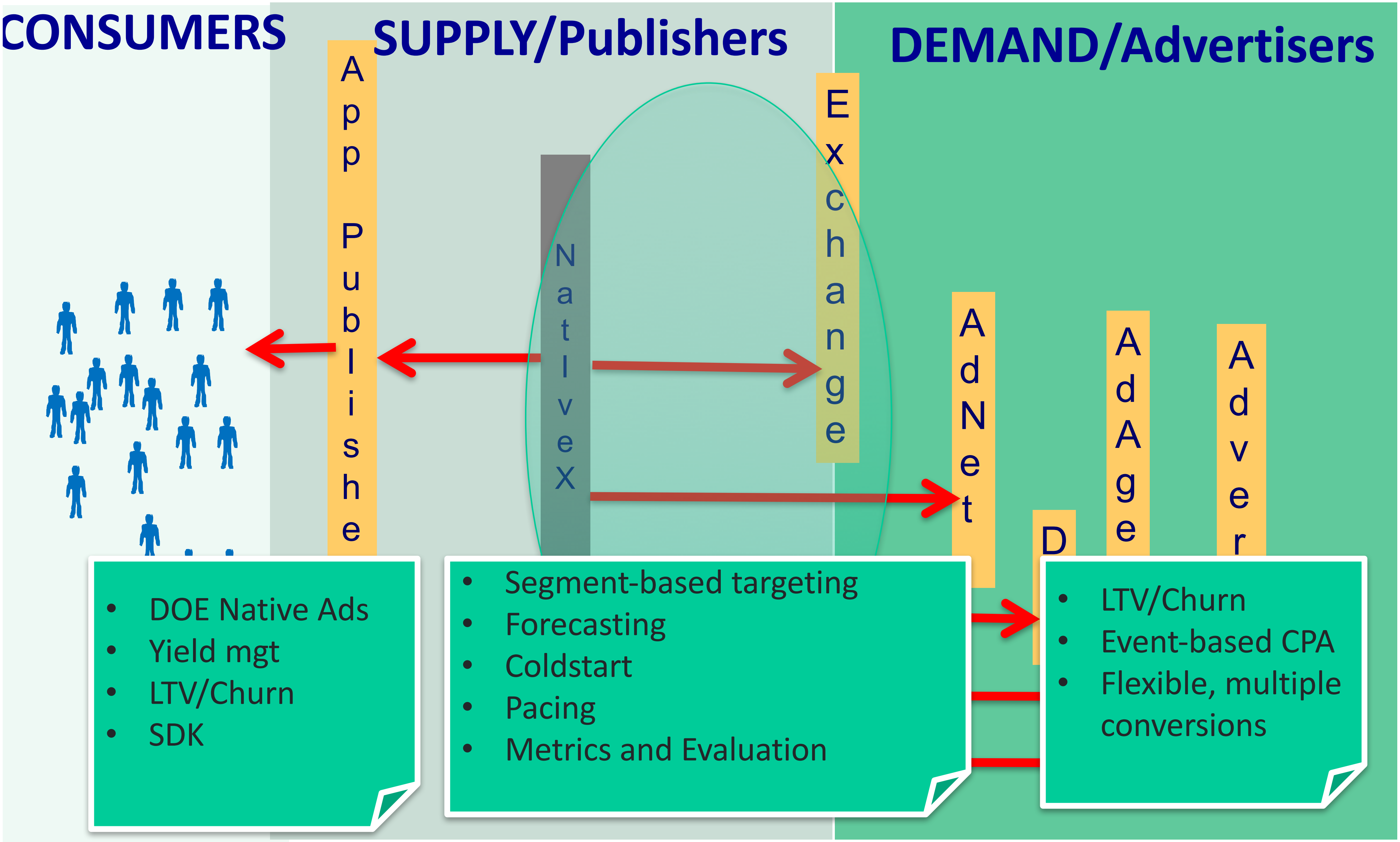
http://www.quora.com/Mobile-Advertising/What-is-the-average-CPM-on-mobile

http://mashable.com/2012/10/23/mobile-ad-prices/

NativeX: Art and Science of Native Mobile Advertising



NativeX: Art and Science of Native Mobile Advertising



Ad serving data pipelines

Realtime Batch

ETL

(Extract, Transform, and Load)

"OLTP" Data Pipeline

- Online Data
- Used in Real Time
- Used for Offer Serving

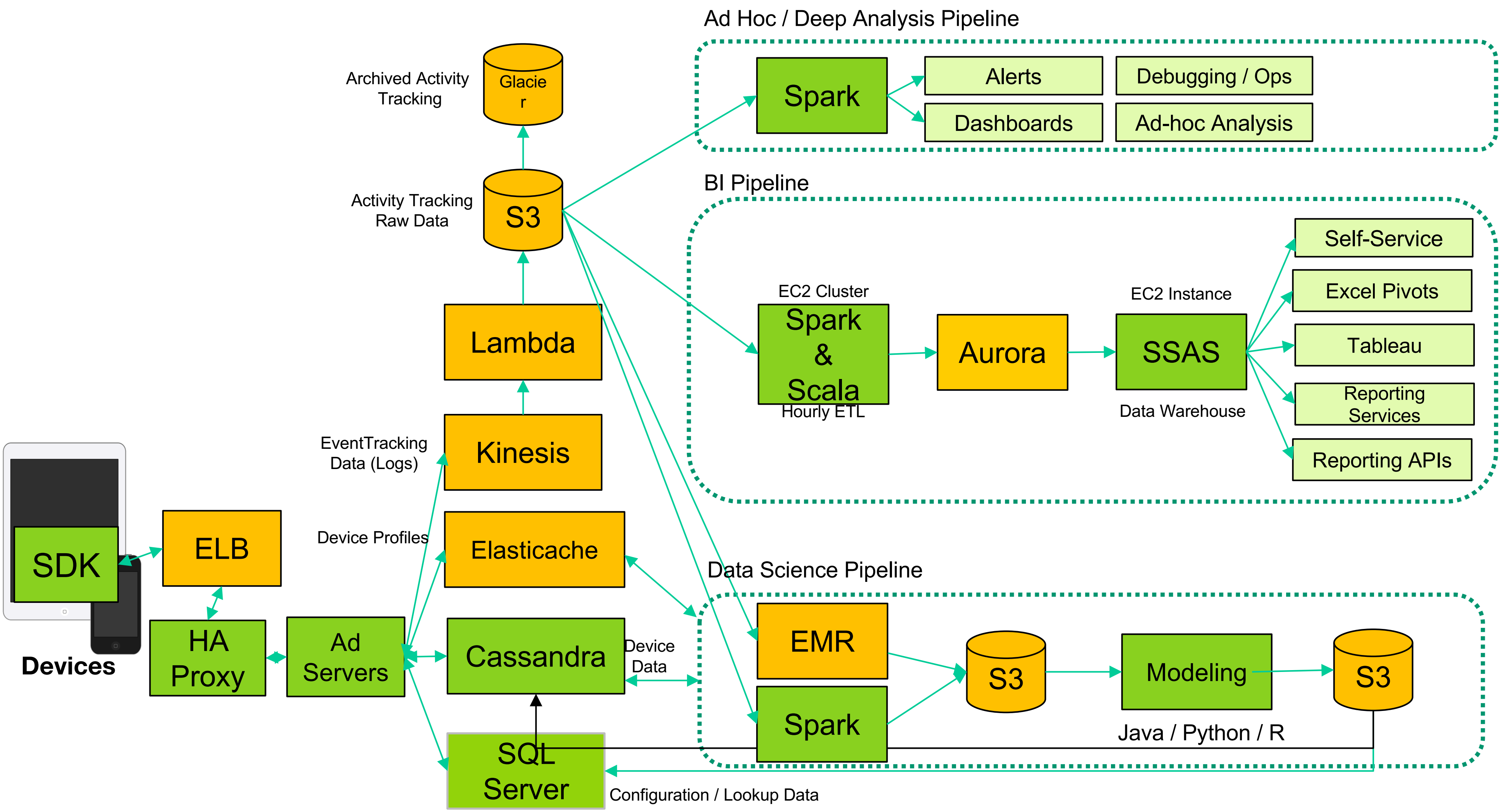
Data Science Predictive Analytics Pi

"OLAP" Data Pipeline

- O line Data
- ging Data
- d for Reporting
- Modeling

Billions of events per day
Process Terabytes of new data
200 milliseconds response times
QPS 1,000,000s

Ad serving architecture



Case Studies

- **Advertising and marketing**
- **Playing games**
- **Autonomous vehicles**
- **Consumer**
  - Image tagging
  - Personalized shopping assistance (build vs buy)
  - Recommender systems
- **Automatic stock market trading: Numer.ai**
- **Social Good**
  - Counting Animals in Flickr vs in The Wild

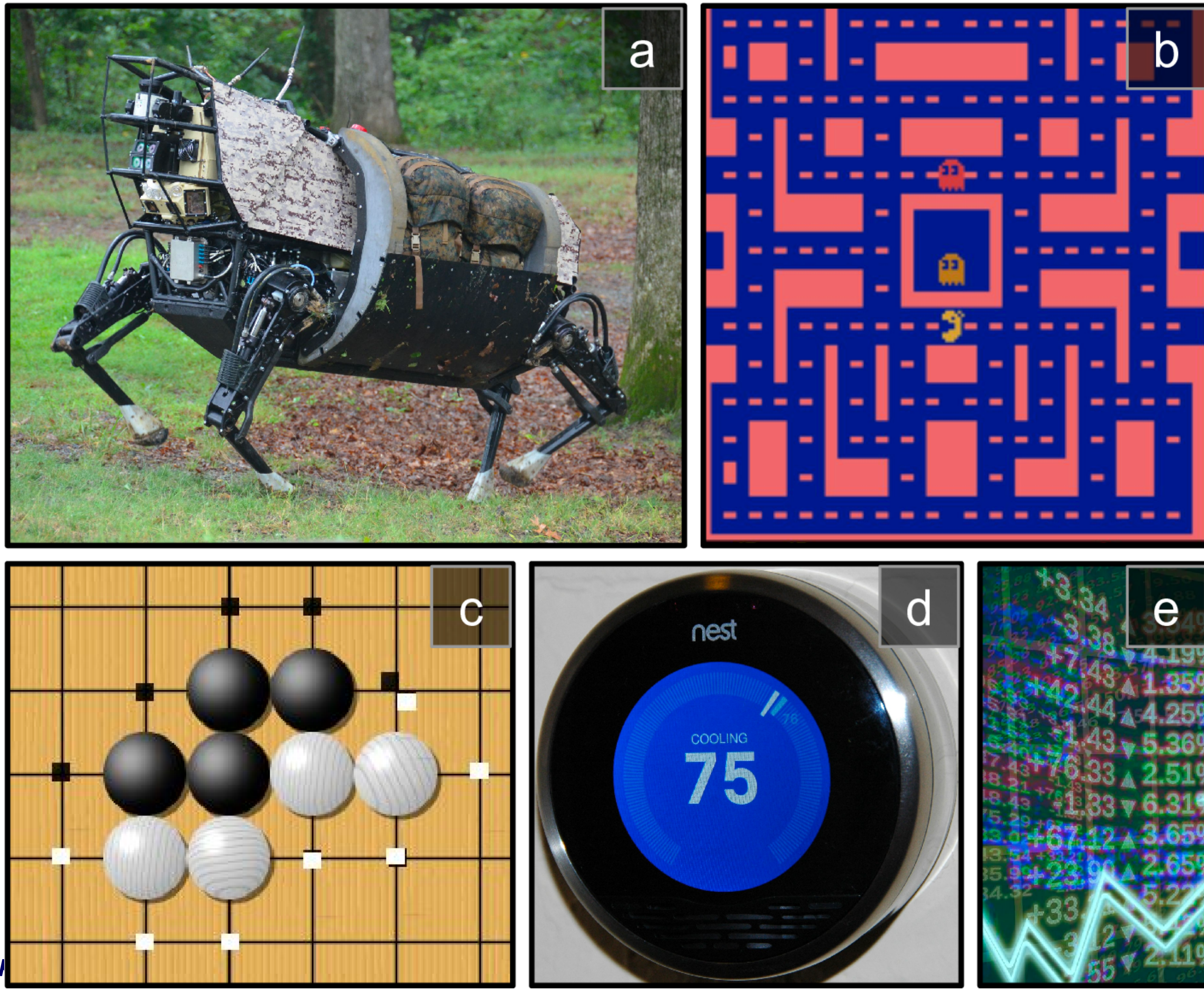
Types of machine learning

- Supervised learning - Generates a function that maps inputs to desired outputs. For example, in a classification problem, the learner approximates a function mapping a vector into classes by looking at input-output examples of the function.

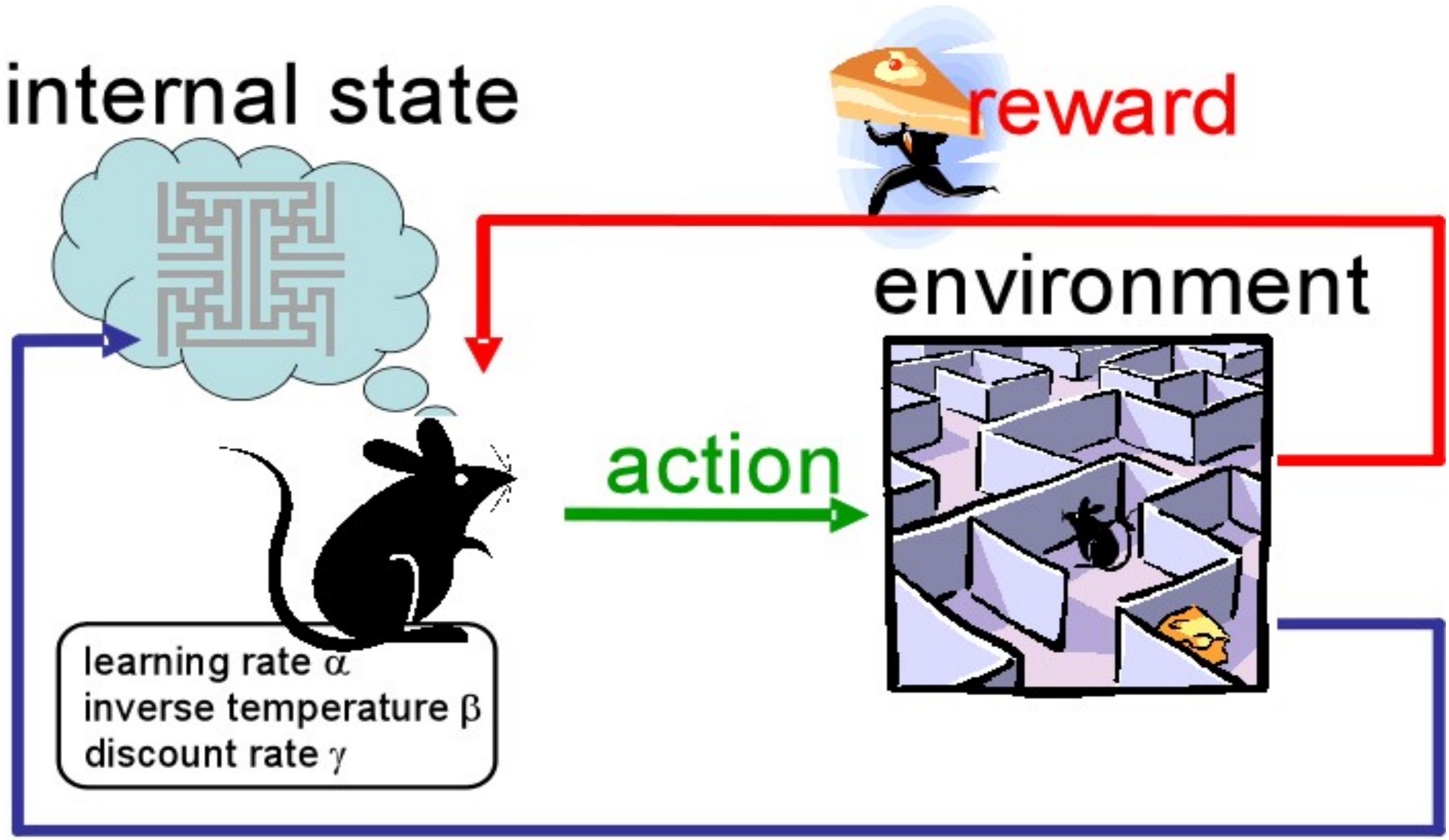
- Reinforcement learning - Learns how to act given an observation of the world. Every action has some impact in the environment, and the environment provides feedback in the form of rewards that guides the learning algorithm.

- Semi-supervised learning - Combines both labeled and unlabeled examples to generate an appropriate function or classifier.
- Unsupervised learning - Models a set of inputs: like clustering
- Transduction - Tries to predict new outputs based on training inputs, training outputs, and test inputs.

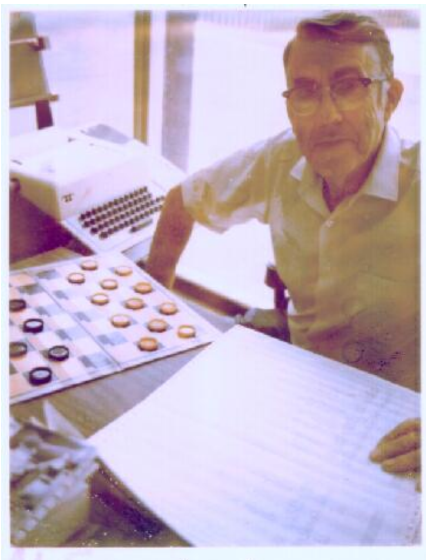
Reinforcement Learning examples



Reinforcement Learning



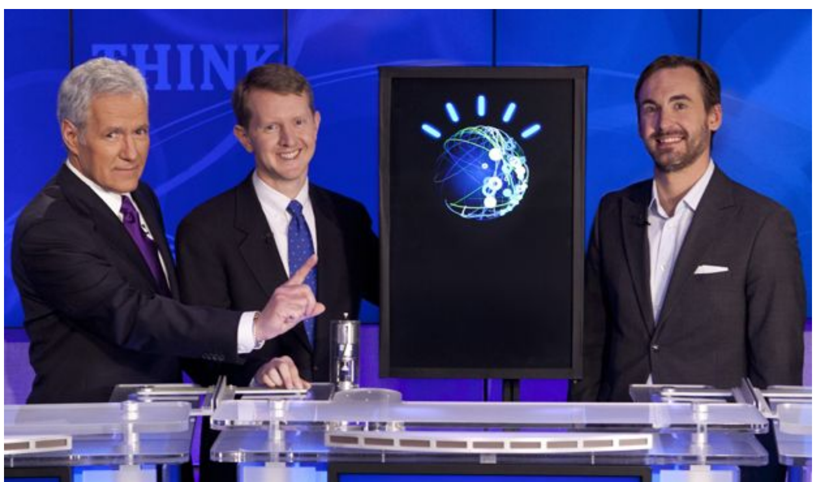
History of Games



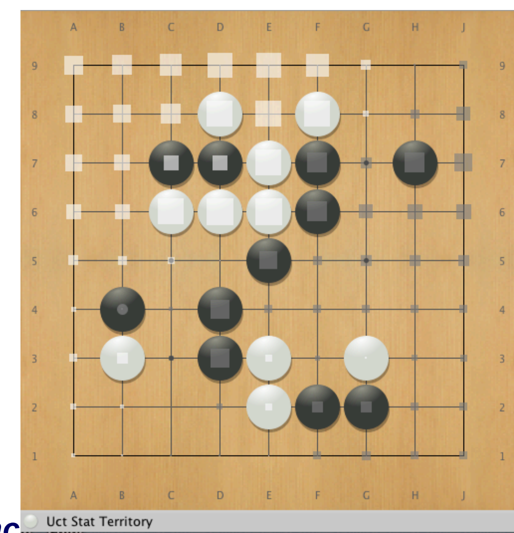
Checkers [1950s]

backgammon

Chess [1997]

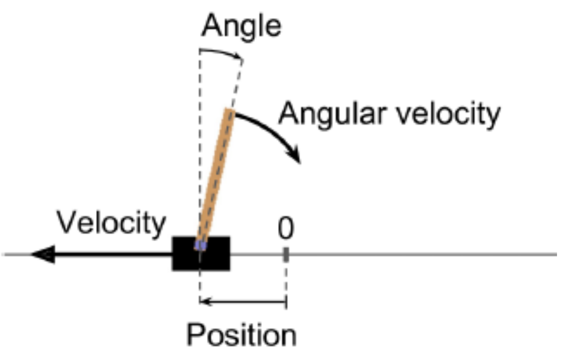


Jeopardy! [2012]



Go [2016]

CartPole example




```python
def basic_policy(obs):
    angle = obs[2]
    return 0 if angle < 0 else 1
```

Simple policy
Go left of the angle is left
Go right if the angle is right

```python
totals = []
for episode in range(500):
    episode_rewards = 0
    obs = env.reset()
    for step in range(1000): # 1000 steps max, we don't want to run forever
        action = basic_policy(obs)
        obs, reward, done, info = env.step(action)
        episode_rewards += reward
        if done:
            break
    totals.append(episode_rewards)
```

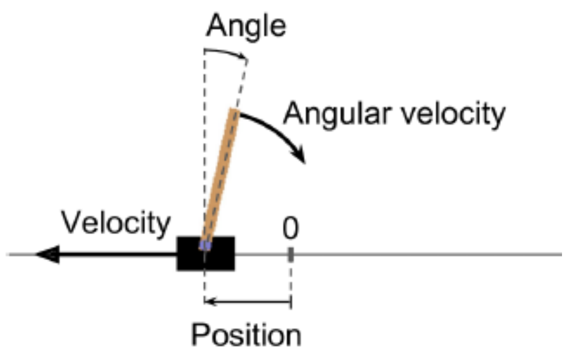
This code is hopefully self-explanatory. Let's look at the result:

```python
>>> import numpy as np
>>> np.mean(totals), np.std(totals), np.min(totals), np.max(totals)
(42.125999999999998, 9.1237121830974033, 24.0, 68.0)
```

https://github.com/ageron/handson-ml

https://www.dropbox.com/s/tyye9744fdf3xks/16_deep_reinforcement_learning.ipynb?dl=0

CartPole example



https://github.com/ageron/handson-ml

https://www.dropbox.com/s/tyye9744fdf3xks/16_deep_reinforcement_learning.ipynb?dl=0

```
def basic_policy(obs):

tot
for
        obs, reward, done, info = env.step(action)
        episode_rewards += reward
        if done:
            break
    totals.append(episode_rewards)
```

Even with 500 tries, this policy never managed to keep the pole upright for more than 68 consecutive steps. Not great. If you look at the simulation in the Jupyter notebooks, you will see that the cart oscillates left and right more and more strongly until the pole tilts too much.

Let's see if a neural network can come up with a better policy.

This code is hopefully self-explanatory. Let's look at the result:

```
>>> import numpy as np
>>> np.mean(totals), np.std(totals), np.min(totals), np.max(totals)
(42.125999999999998, 9.1237121830974033, 24.0, 68.0)
```

Atari Games … all of them

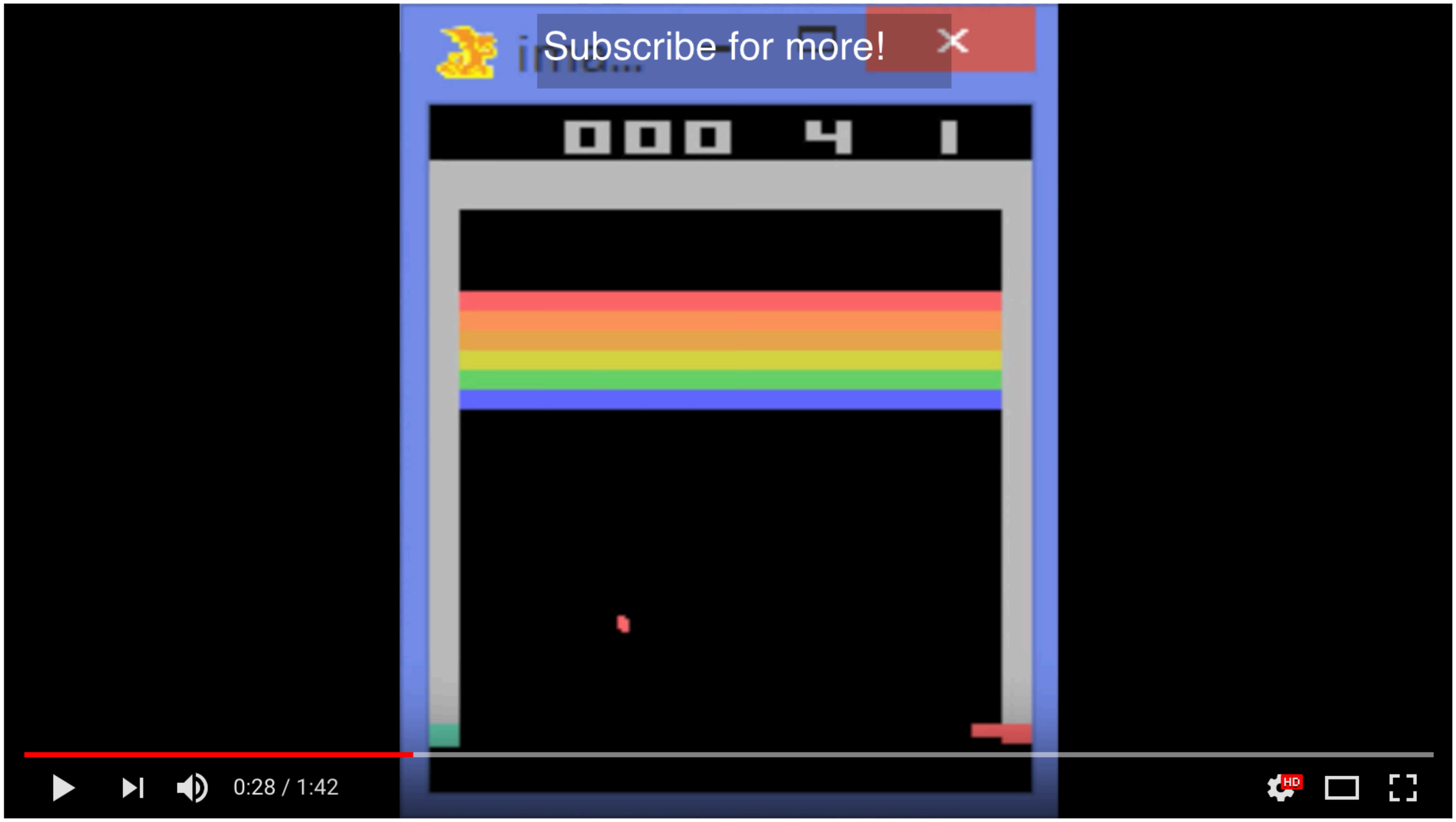
What does $500M look like?

- **Goal:**
  - System that can play any game
  - Only use information human has
    - Raw pixel data
- **Success:**
  - Google's Deep Mind team
  - Used "Deep" Neural Network
  - Can learn any Atari 2600 game
- **Can we get more general?**

- **Video:** https://youtu.be/V1eYniJ0Rnk
- **Code:** https://sites.google.com/a/deepmind.com/dqn/

Atari Breakout: Deep Q Networks

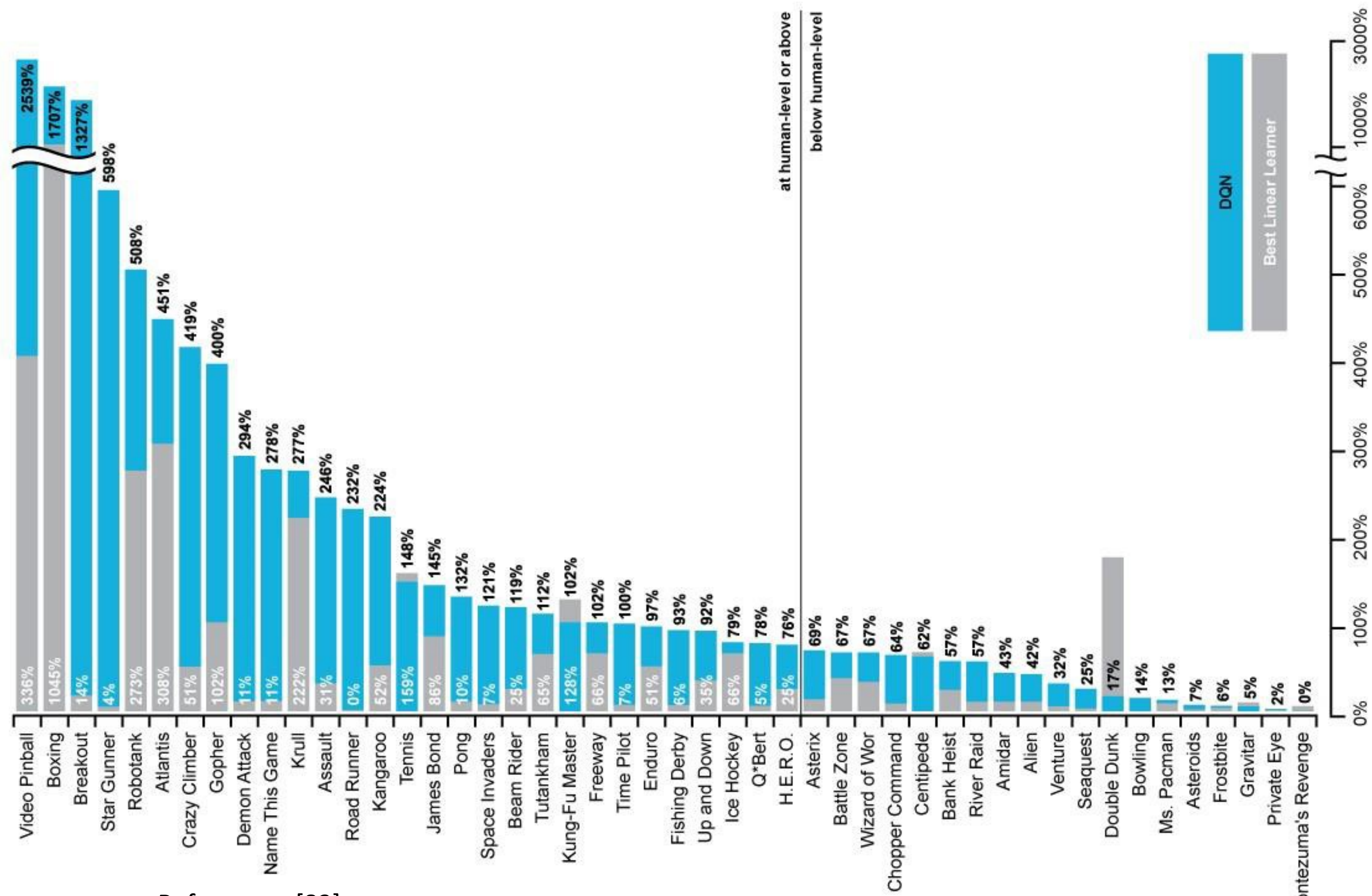
- https://www.youtube.com/watch?v=V1eYniJ0Rnk



Google DeepMind's Deep Q-learning playing Atari Breakout

703,352 views 2K 32 SHARE

DeepMind: DQN Results in Atari



References: [83]

Deep Learning for Self-Driving Cars fridman@mit.edu cars.mit.edu 2017

Train an Drive-atar (avatar) using deepQlearning (to imitate your driving style)

Xbox,
Windows

- Unity gaming engine
  - Has RL capabilities [9/201

Train an Drive-atar (avatar) using deepQlearning (to imitate your driving style)

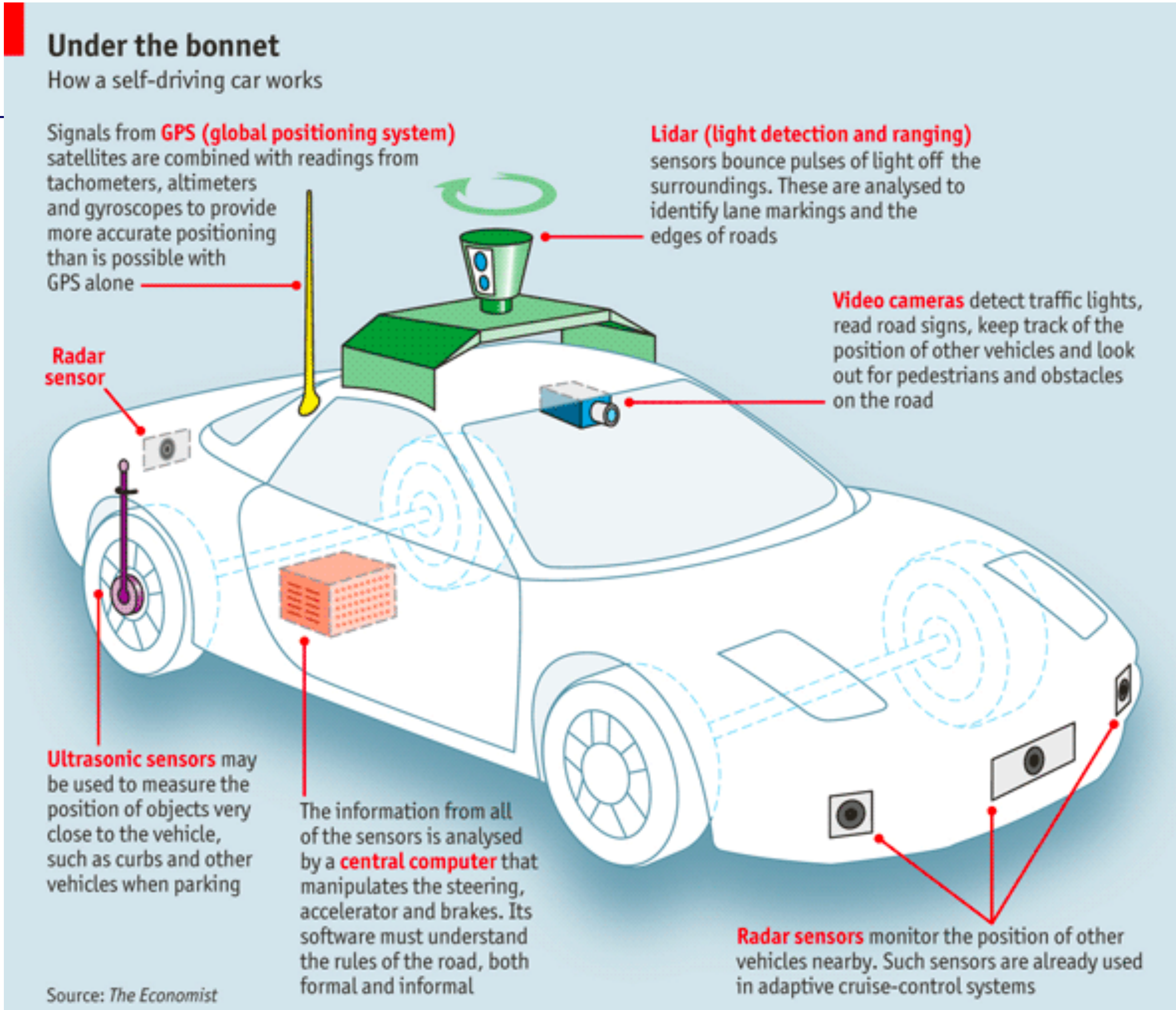
Case Studies

- **Advertising and marketing**
- **Playing games**
- **Autonomous vehicles**
- **Consumer**
  - Image tagging
  - Personalized shopping assistance (build vs buy)
- **Automatic stock market trading: Numer.ai**
- **Social Good**
  - Counting Animals in Flickr vs in The Wild

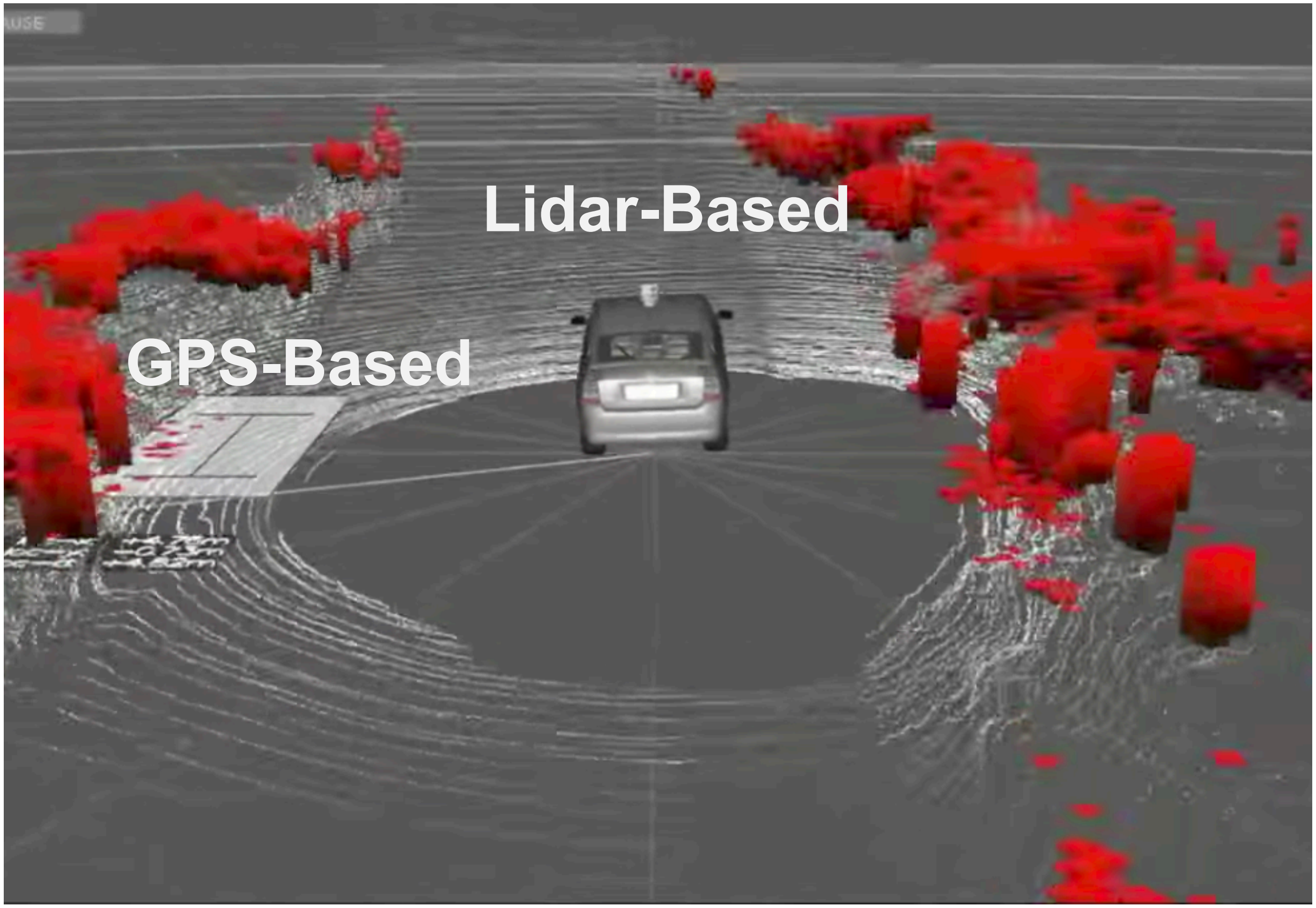
Closing the loop with reinforcement: **AI for cars**

Soon Self-driving Vehicles Should Be a Norm and Safer Than Conventional Vehicles

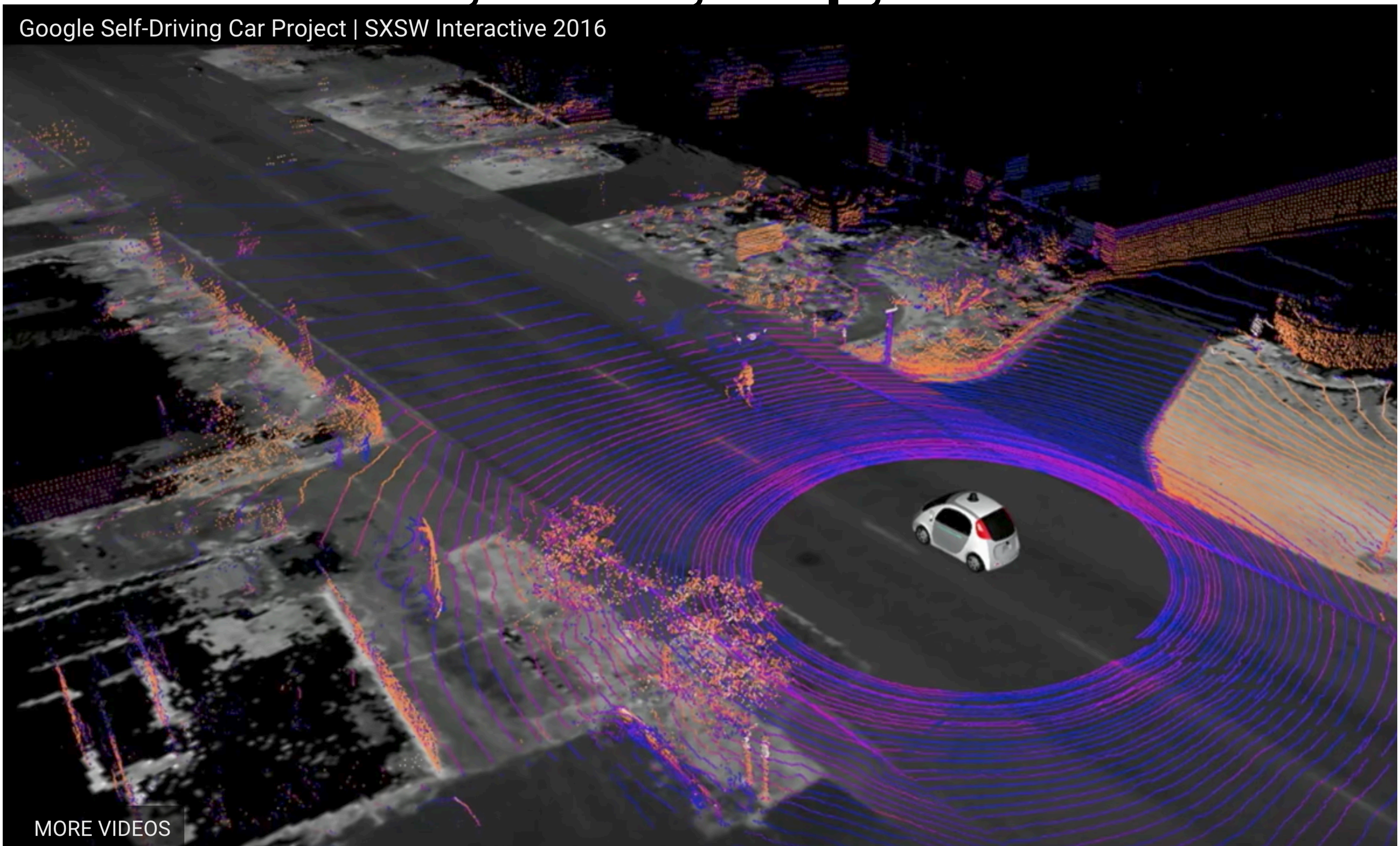
*"Google could boast that its self-driving cars had **logged more than a million miles without causing a single accident**. Google's cars had been involved in a few accidents, but all of them had been the fault of the other driver".*



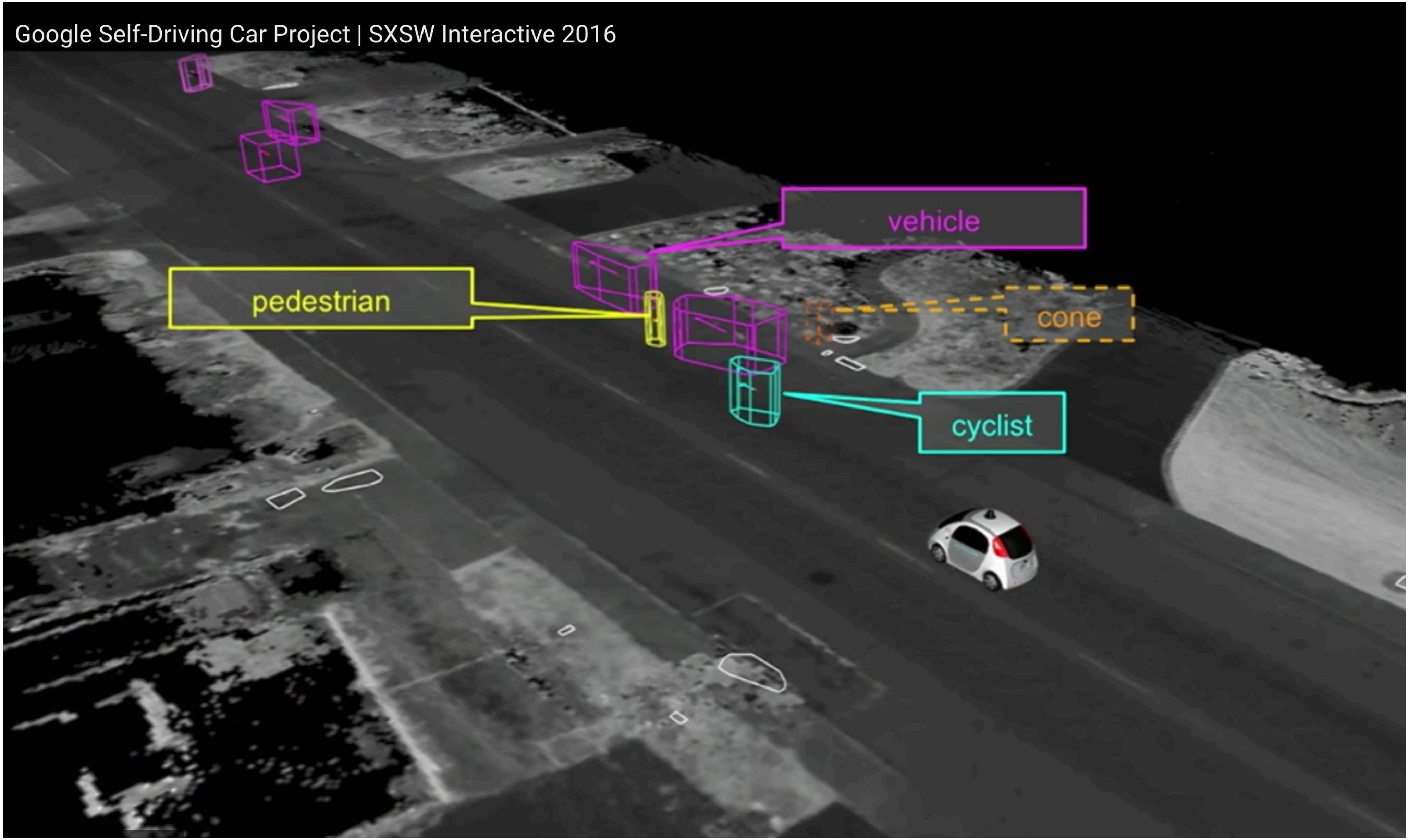
Under the bonnet
How a self-driving car works
Signals from GPS (global positioning system) satellites are combined with readings from tachometers, altimeters and gyroscopes to provide more accurate positioning than is possible with GPS alone
Lidar (light detection and ranging) sensors bounce pulses of light off the surroundings. These are analysed to identify lane markings and the edges of roads
Video cameras detect traffic lights, read road signs, keep track of the position of other vehicles and look out for pedestrians and obstacles on the road
Radar sensor
Ultrasonic sensors may be used to measure the position of objects very close to the vehicle, such as curbs and other vehicles when parking
The information from all of the sensors is analysed by a central computer that manipulates the steering, accelerator and brakes. Its software must understand the rules of the road, both formal and informal
Radar sensors monitor the position of other vehicles nearby. Such sensors are already used in adaptive cruise-control systems
Source: The Economist


Lidar-Based
GPS-Based

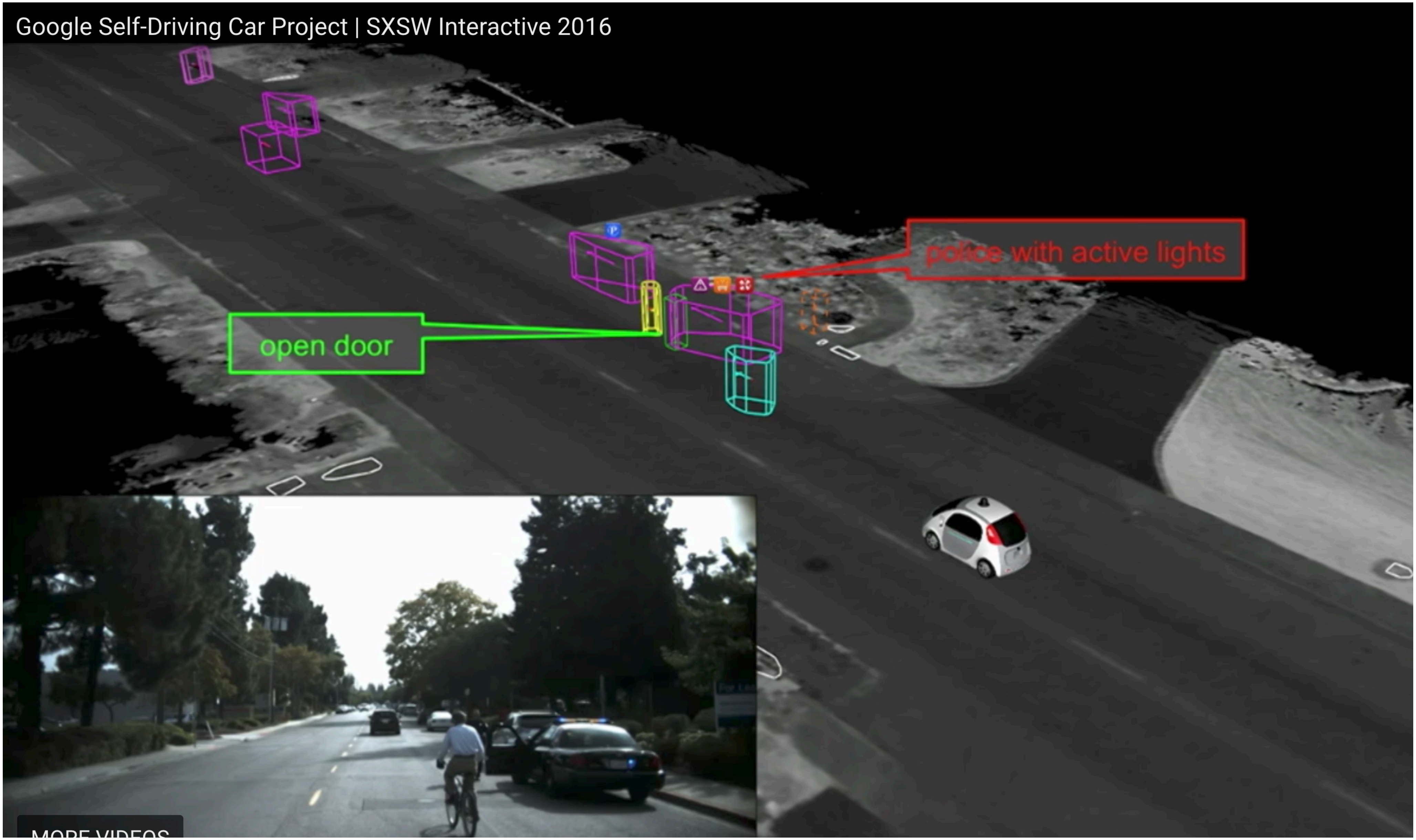
Sensor-based data (millions of inputs) Radar, Laser, Map, Camera data



Highway perception



Reason, plan and act

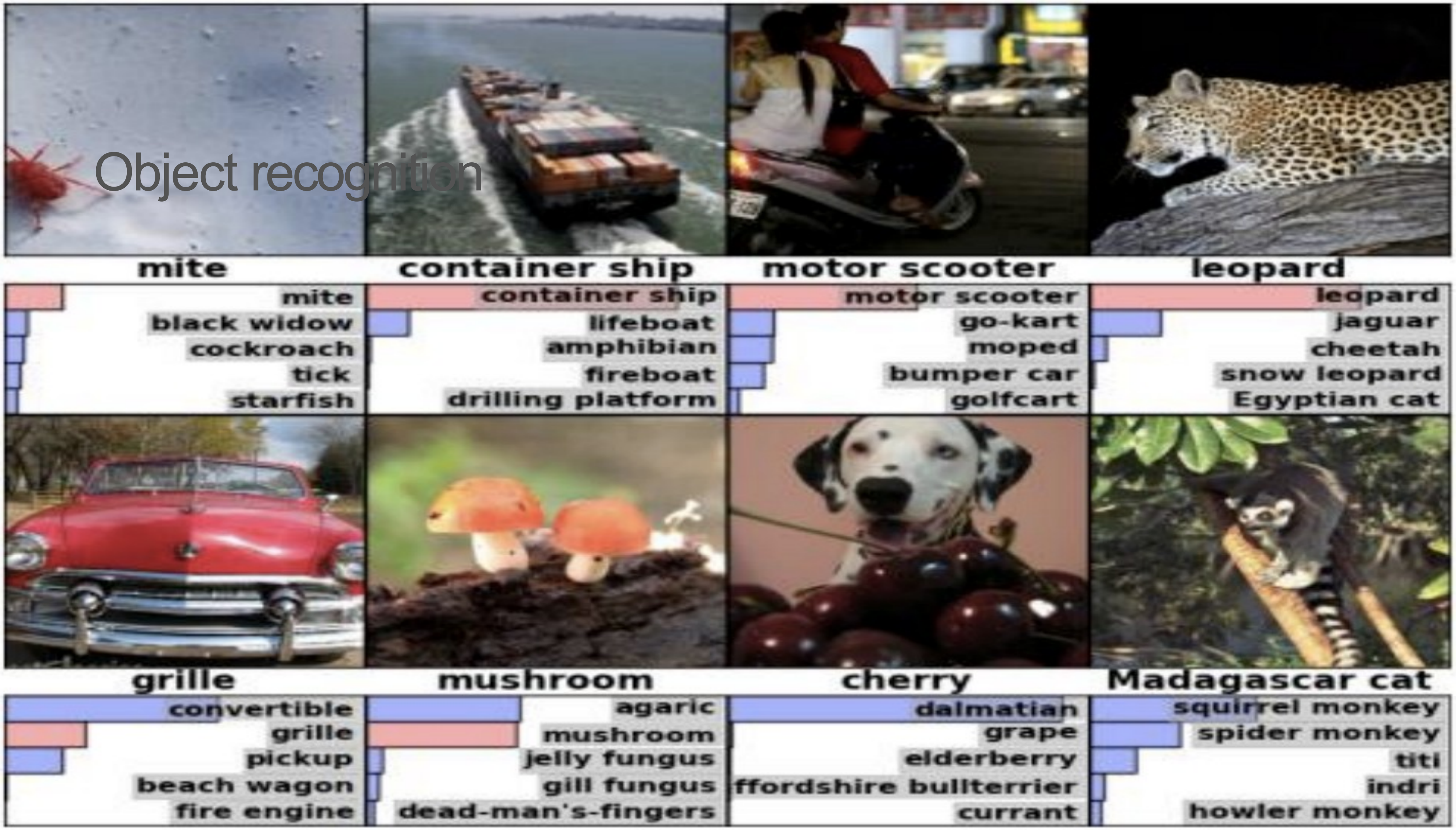



Google Self-Driving Car Project | SXSW Interactive 2016
trajectory
id4:0.96/0.94

Case Studies

- **Advertising and marketing**
- **Playing games**
- **Autonomous vehicles**
- **Consumer**
  - Image tagging
  - Personalized shopping assistance (build vs buy)
- **Automatic stock market trading: Numer.ai**
- **Social Good**
  - Counting Animals in Flickr vs in The Wild

Image classification



http://www.image-net.org/challenges/LSVRC/

Face detection and recognition

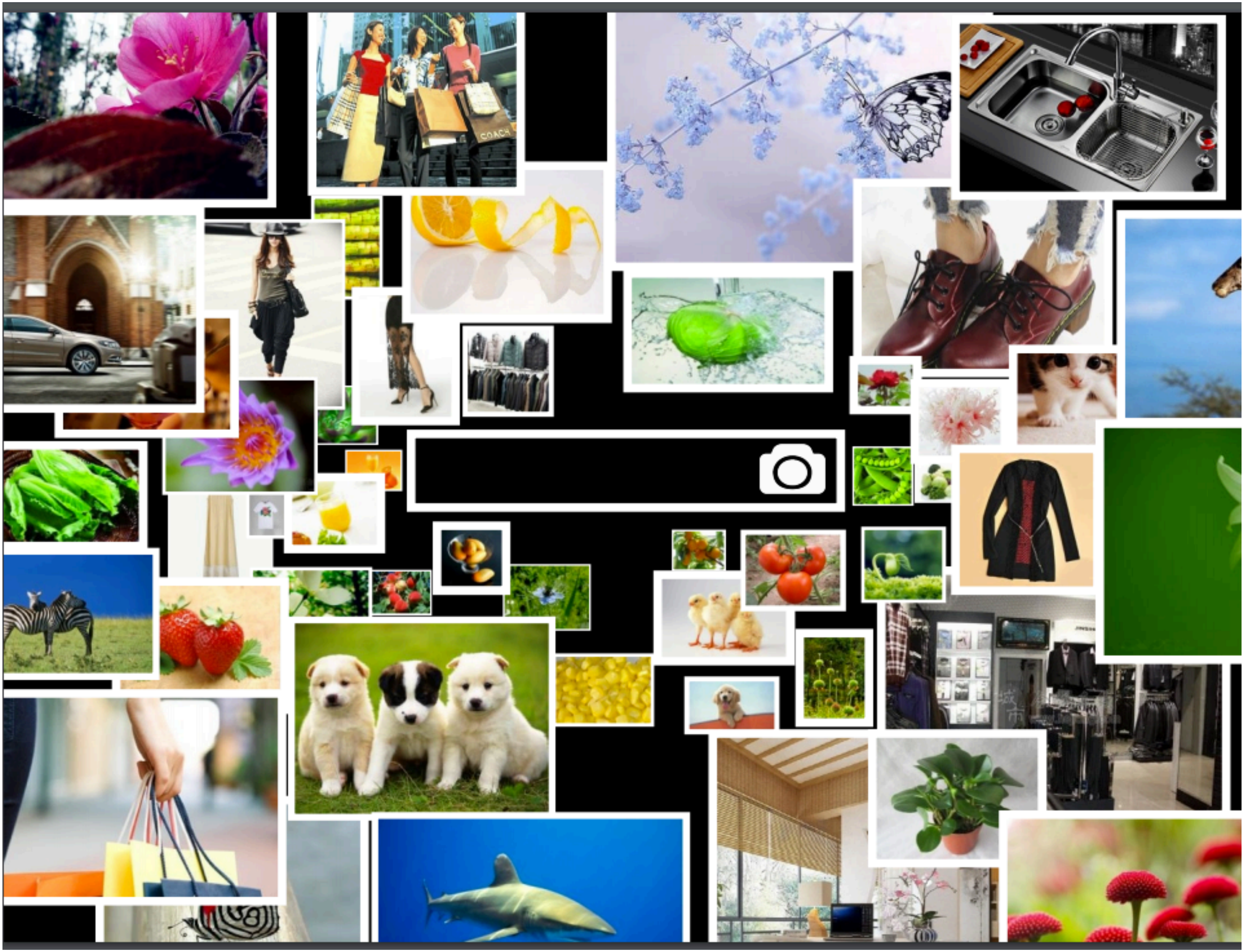
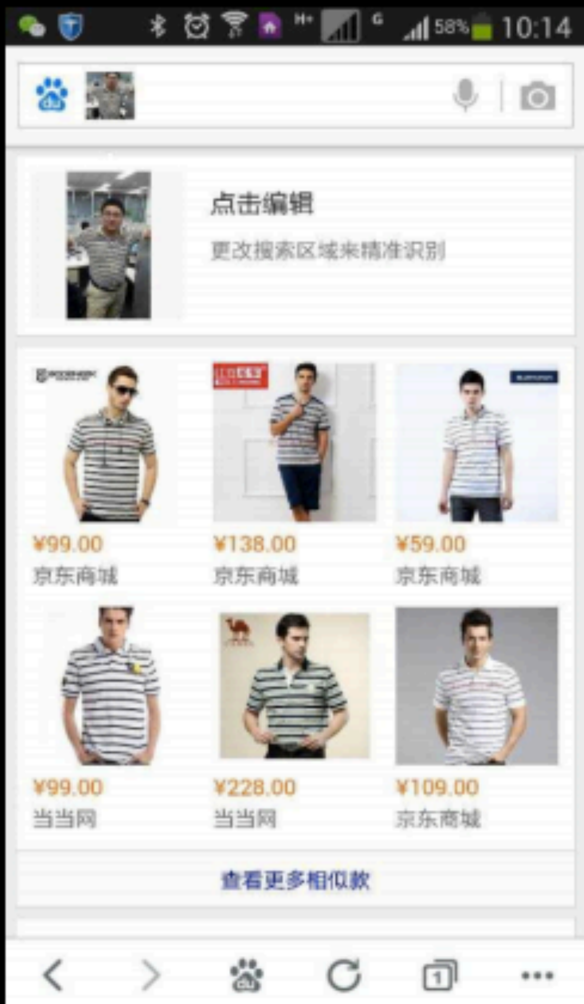
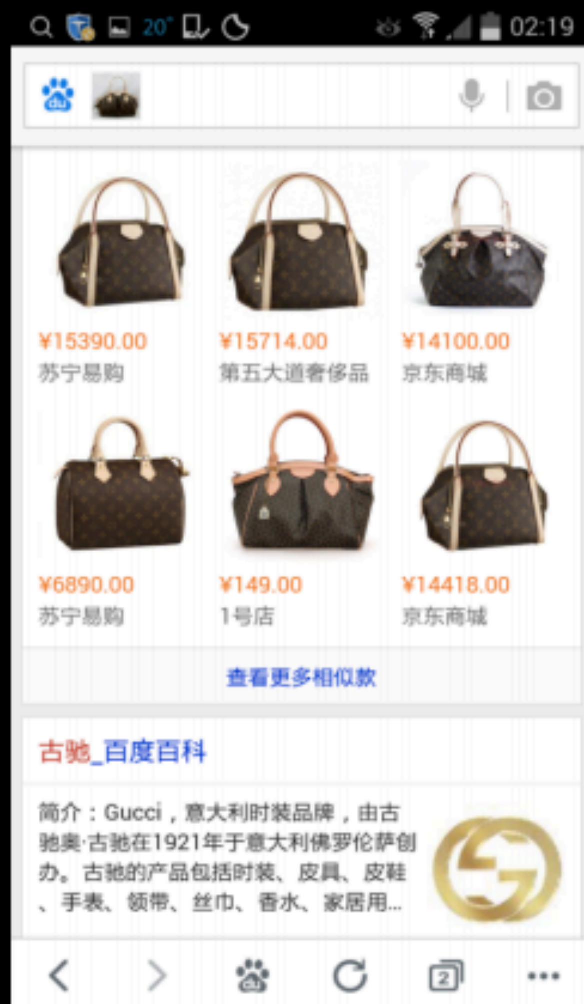


Image queries



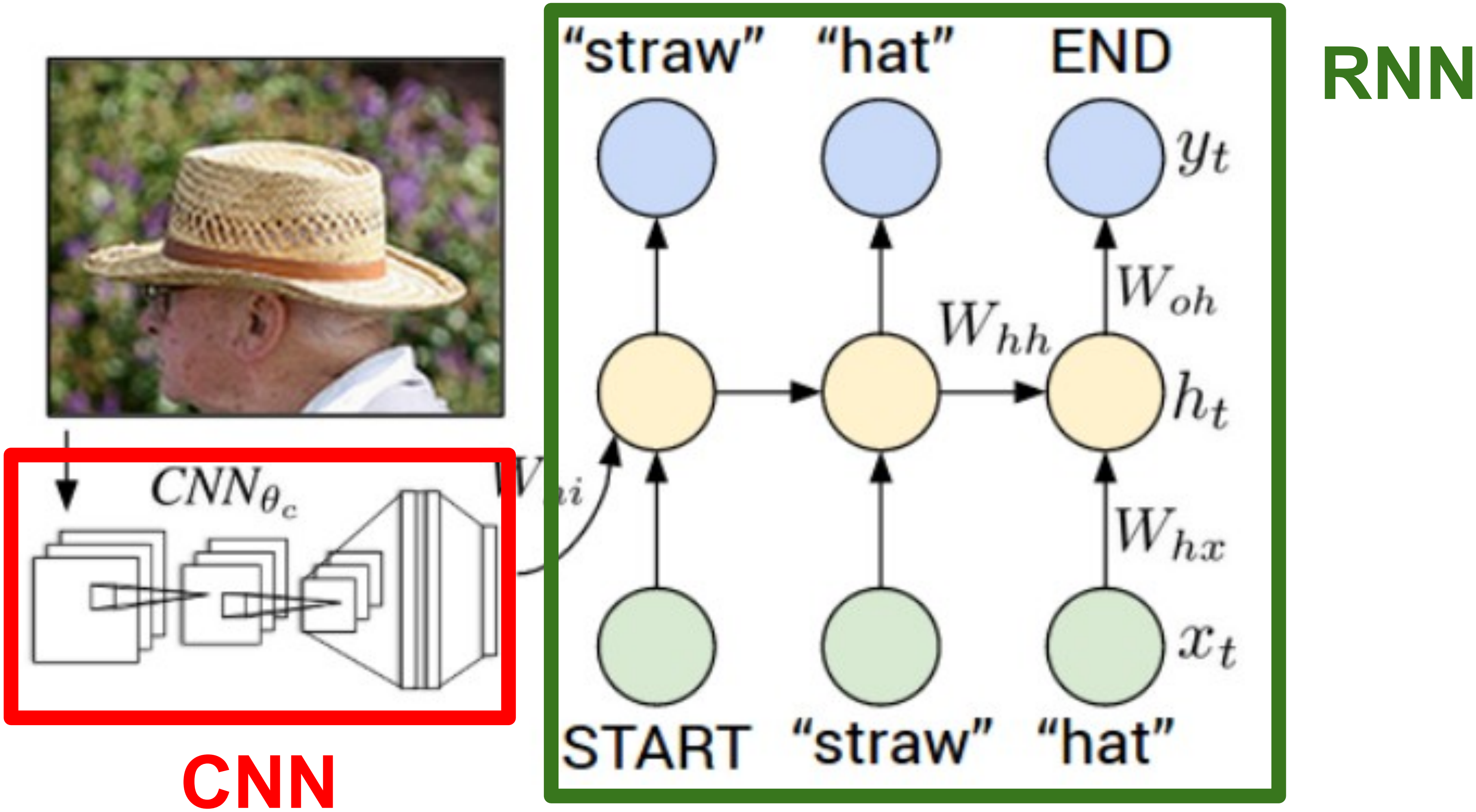
Clothing



Bags

Fruits & Vegetables

Generate the caption for an image



Case Studies

- **Advertising and marketing**
- **Playing games**
- **Autonomous vehicles**
- **Consumer**
  - Image tagging
  - Personalized shopping assistance (build vs buy)
  - Recommender engines
- **Automatic stock market trading: Numer.ai**
- **Social Good**
  - Counting Animals in Flickr vs in The Wild

Virtual Assistants, Chatbots

Virtual Assistants, Chatbots
Horizontal service: access to other apps and services

Siri, Alexa, Google Home, Cortana, etc.

personalized shopping assistance

eBay

- **Artificial intelligence + commerce = highly personalized shopping assistance for everyone**
- **Embedded in chats platforms (FB Messenger); Alexa**

Case Studies

- **Advertising and marketing**
- **Playing games**
- **Autonomous vehicles**
- **Consumer**
  - Image tagging
  - Personalized shopping assistance (build vs buy)
- **Automatic stock market trading: Numer.ai**
- **Social Good**
  - Counting Animals in Flickr vs in The Wild

Numer.ai: Crowdsourced AI Hedge Fund

- Based in San Francisco, Numerai is a hedge fund in which an artificially intelligent system chooses all the trades.
- Several thousand anonymous data scientists compete to create the best trading algorithms—and win bitcoin for their efforts.
- But Numerai has been making trades in this way since 2016, and apparently it's making money.
- Numerai, Quantopian, Pit.ai, a new machine learning-powered hedge fund, based on RL

Case Studies

- **Advertising and marketing**
- **Playing games**
- **Autonomous vehicles**
- **Consumer**
  - Image tagging
  - Personalized shopping assistance (build vs buy)
- **Automatic stock market trading: Numer.ai**
- **Social Good**
  - Counting Animals in Flickr vs in The Wild

Counting Animals in Flickr vs in The Wild

Whale recognition, Kaggle Challenge

$3,000 to deploy a Tracking device

- S. Menon, T. Y. Berger-Wolf , E. Kiciman, L. Joppa, C. V. Stewart, J. Parham, J. Crall, J. Holmberg, J. Van Oast, “Animal Population Estimation Using Flickr Images”, 2nd International Workshop on the Social Web for Environmental and Ecological Monitoring (SWEEM 2017) .