

AI Pavilion
29 October 2018

AI Control Problem

David Evans
University of Virginia

aipavilion.github.io

Schedule Reminders

“Final” Papers are Due 4:59pm Thursday, Nov 1

Email evans@virginia.edu, subject line: **[AI Pavilion] Paper Title**

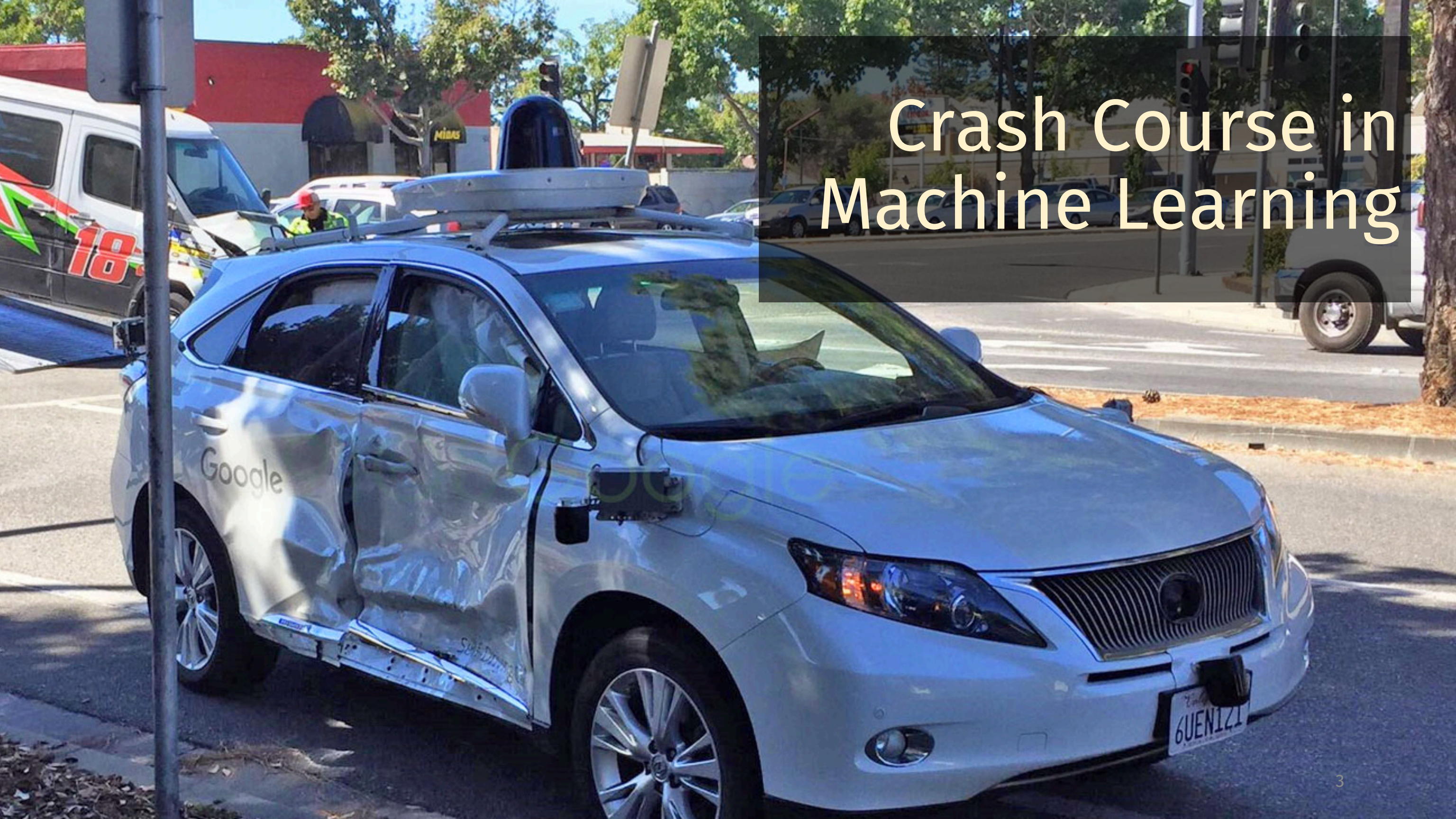
Include in the email body:

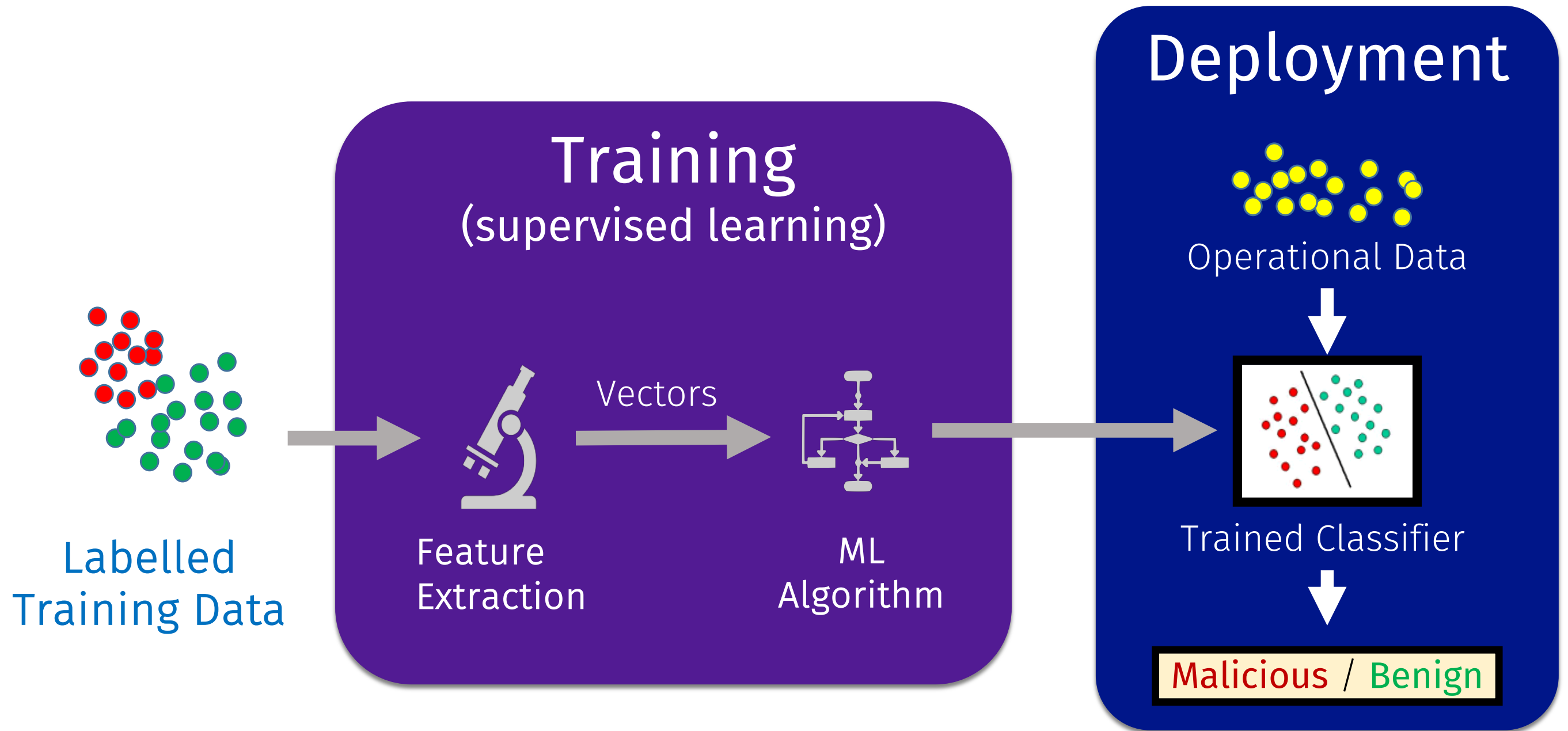
1. What is the **purpose** of your paper? (one sentence answer)
2. Who is your intended **audience**? (one sentence answer)
3. If you decided not to follow advice from the first draft, explain why. (It is okay to not follow advice, but you need to make it clear that you understood the advice and justify why you didn't follow it.)
4. Do you want to **continue with this topic**, or start on a new topic for the “final” paper? (Yes/no answer is fine, but feel free to explain more if helpful)

Next Class: Short Presentation

You should prepare a short presentation (no more than 5 minutes) about your paper

Crash Course in Machine Learning





Statistical Machine Learning

Learning a Function

$$f: \mathbb{R}^n \longrightarrow \{1, 2, \dots, k\}$$

$$f: \mathbb{R}^n \longrightarrow \mathbb{R}$$

$$f: \mathbb{Z}^n \longrightarrow \mathbb{Z}^n$$

Learning a Function

$$f: \mathbb{R}^n \rightarrow \{1, 2, \dots, k\}$$

Classifier:



→ “panda”

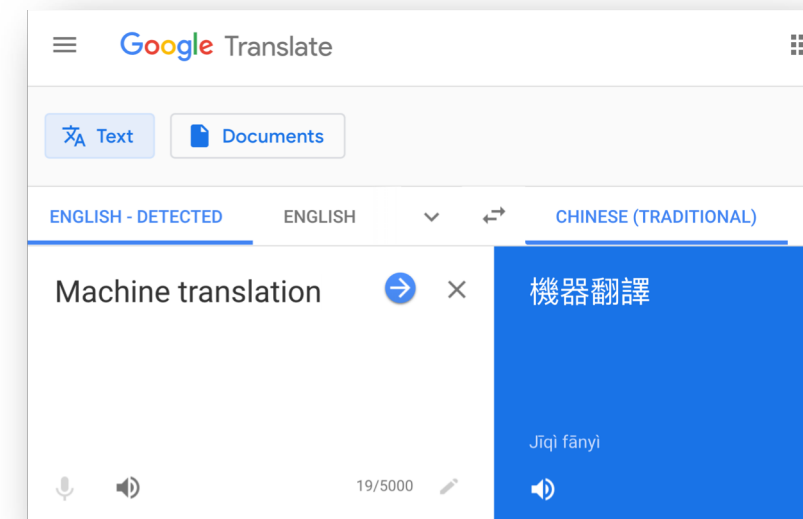
$$f: \mathbb{R}^n \rightarrow \mathbb{R}$$

Regression:

Profile → risk of default

$$f: \mathbb{Z}^n \rightarrow \mathbb{Z}^n$$

Translation:



How to Learn Functions

Linear Regression: learn a function of type:

$$y = w_1x_1 + w_2x_2 + \dots + w_nx_n$$

How to Learn Functions

Linear Regression: learn a function of type:

$$\hat{y} = w_1x_1 + w_2x_2 + \dots + w_dx_d$$

Training: given set of labeled points, $\{ (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n) \}$
Find the values for weights w that minimize prediction error:

$$\text{Mean squared error: } MSE = \frac{1}{n} (\sum_i (\hat{y}_i - y_i)^2)$$

How well does this work?

Generalization:

easy to learn a function that predicts the training data

a simple lookup table does perfectly!

goal is to find a function that generalizes:

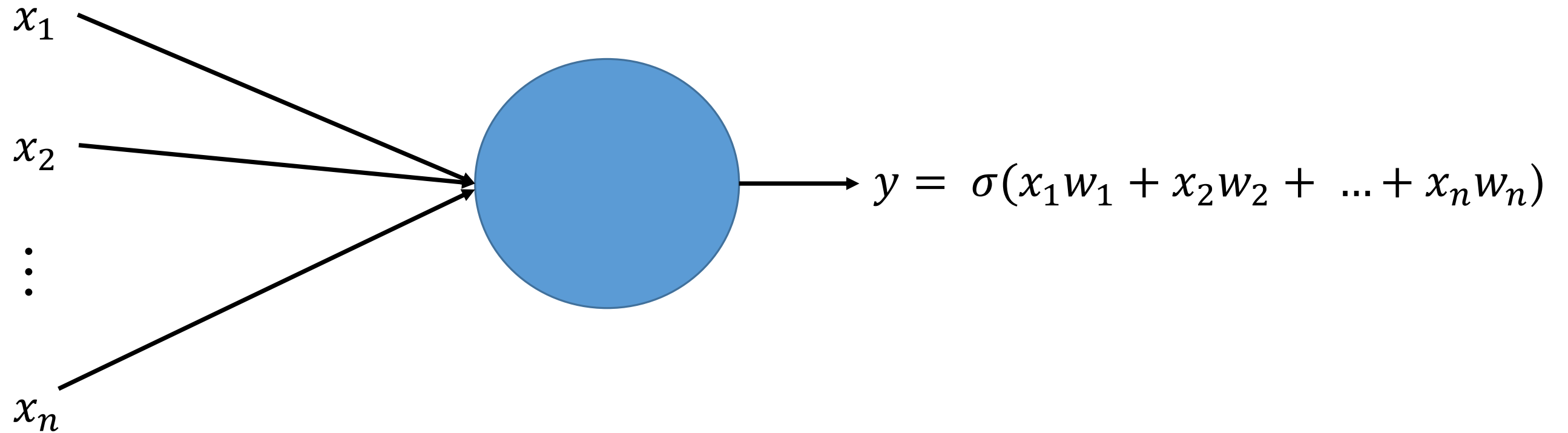
produces correct prediction for unseen inputs

Capacity: ability to learn a large variety of functions

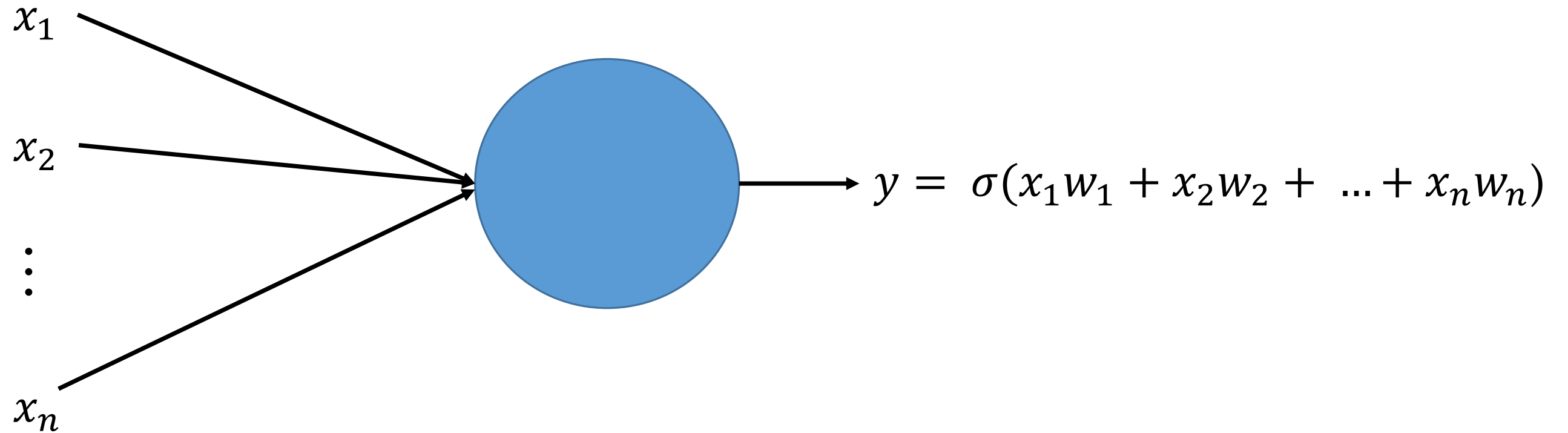
linear regression can only learn linear functions

too high capacity: overfits training data (poor generalization)

Perceptron



How powerful is a perceptron?





Trick-or-Treat Protocols

“Trick or Treat”

Tricker initiates the protocol by making a threat and demanding tribute

Trick v Treat

Victim either pays tribute (usually in the form of sugary snack) or risks being tricked



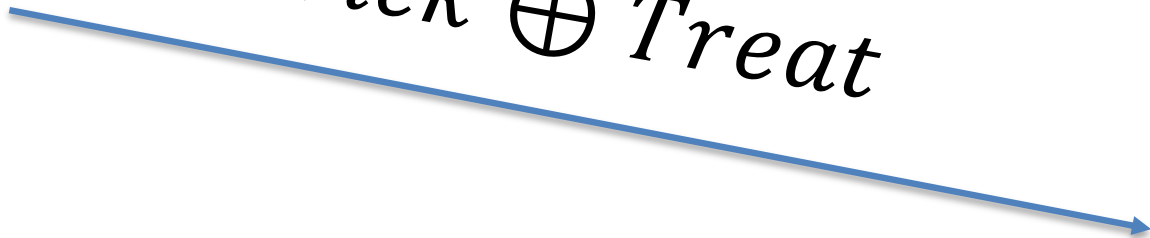
Illogical Threat

(Trick v Treat)

“Trick xor Treat”

Tricker initiates the protocol by making a threat and demanding tribute

Trick $\oplus$ Treat



Victim either pays tribute (usually in the form of sugary snack) or risks being tricked

Tricker must convince **Victim** that she poses a credible threat:
prove she is a qualified tricker

Trick-or-Treat

“Trick xor Treat?”

“Prove it!”

“The magic word is: squamish ossifrage”



Victim



Trickers?



Any problems with this?

Proof without Disclosure

How can the **tricker** prove their trickability, without allowing the **victim** to now impersonate a **tricker**?



Challenge-Response Protocol

Verifier: picks random y .

Prover: proves knowledge of x by revealing $f(x, y)$.

Verifier: convinces prover knows x , but learns nothing useful about x .

Need a **one-way function**: hard to invert, but easy to compute.

Example: RSA

$$E_e(M) = M^e \bmod n$$

$$D_d(C) = C^d \bmod n$$

Correctness property: $E_e(D_d(y)) = y$

Trick-or-Treat

“Trick xor Treat?”

“prove it! Challenge = y ”

Response: $r = D_x(y) = y^x \bmod n$

Trickers?

Victim



Trick-or-Treat

“Trick xor Treat?”

“prove it! Challenge = y ”

Response: $r = D_x(y) = y^x \bmod n$

Verify:

$E_e(r) = r^e \bmod n = y$

Trickers?

How does victim know e and n ?



Victim





“Trick xor Treat?”

“What is your Tricker ID?”



“Elsa #253224”,
 $e = 3482\dots$,
 $n = 1234\dots$

signed by Tricker’s Buroo

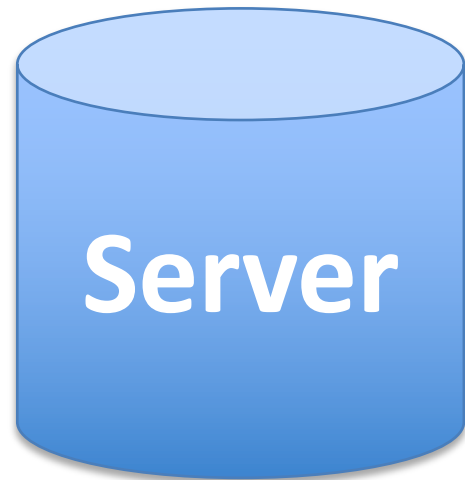
Verify Tricker’s Buroo
signature on certificate

“Prove it! Challenge = y ”

Response: $r = D_x(y) = y^x \bmod n$

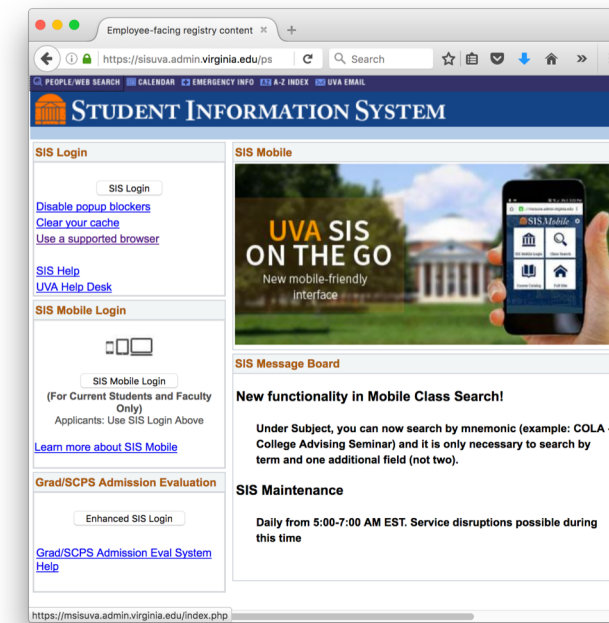
Verify:

$$E_e(r) = r^e \bmod n = y$$



“Trick xor Treat?”

“Hello”



“virginia.edu”,
 $e = \dots$
 $n = \dots$

signed by Certificate Authority

Verify signature on
certificate

“Prove it! Decrypt $E_e(y)$ ”

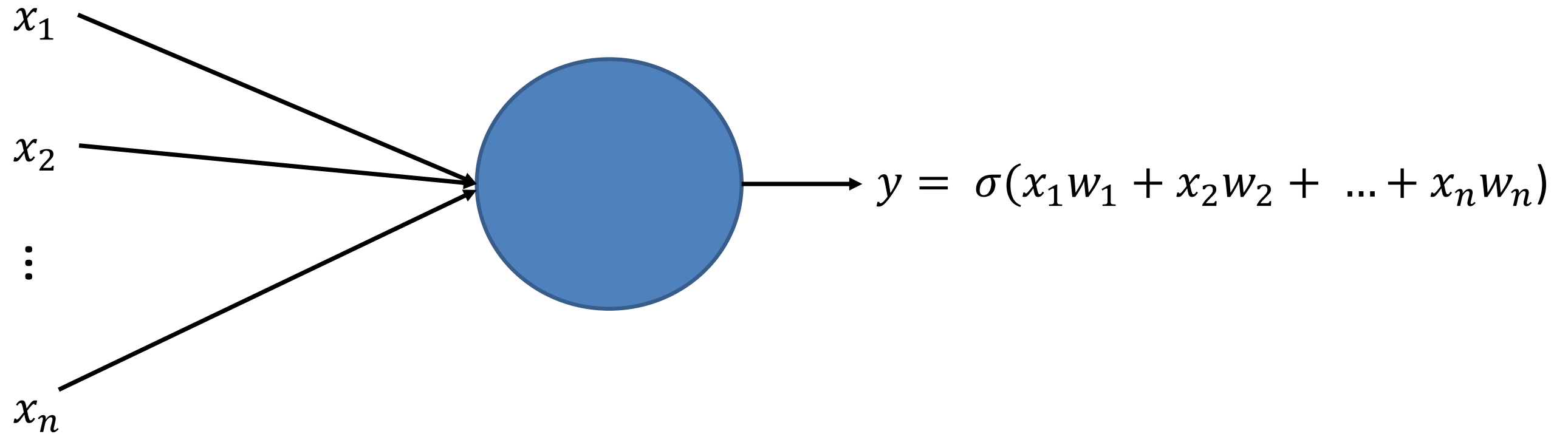
channel encrypted using y

Verify and Decrypt:

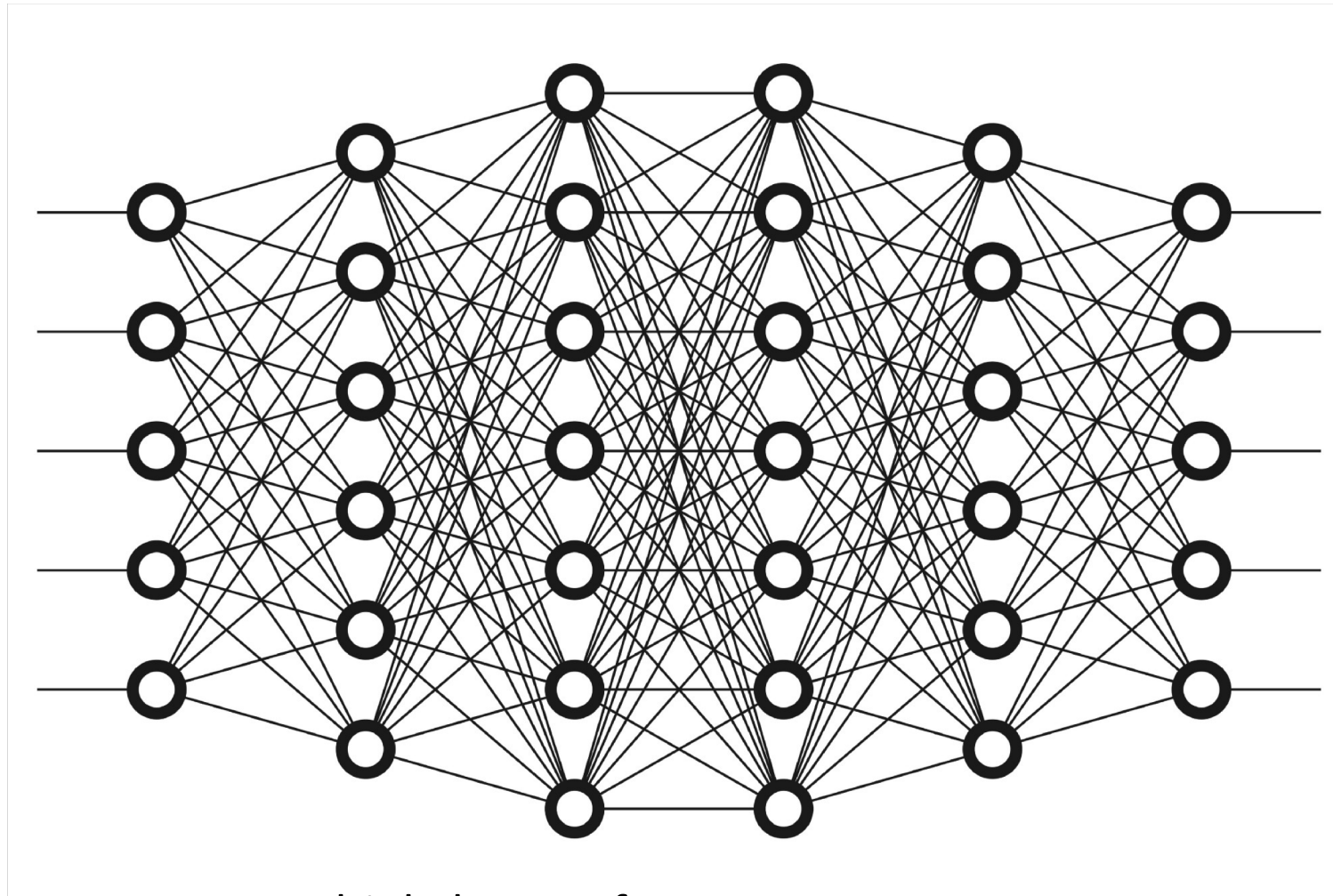
$$D_d(E_e(y)) = y$$



How powerful is a perceptron?



Deep Learning

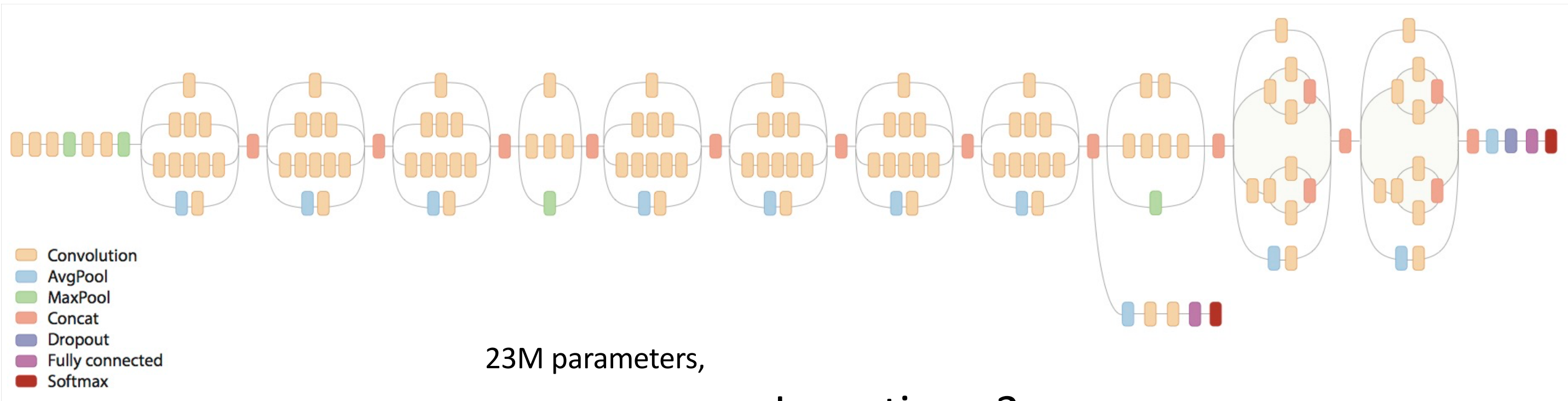
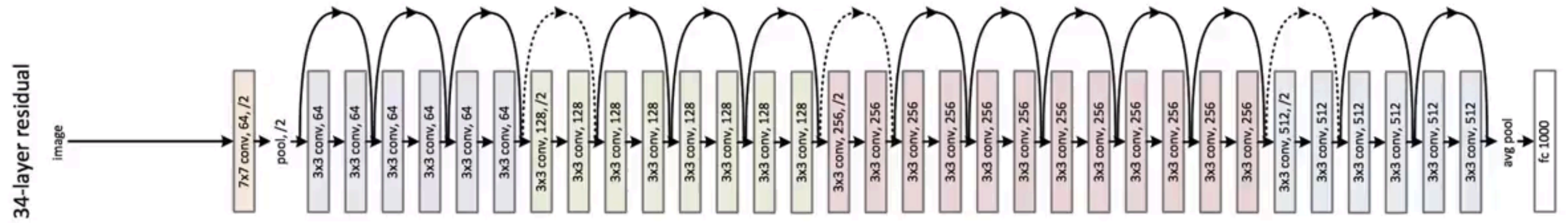


Connect multiple layers of perceptrons

In theory, one hidden layer is enough to match any training set

In practice, more layers often works better

ResNet



Inception v3

Training a DNN

- Loss function: measure of how close output are to desired outputs
- Backpropagation: update weights throughout network to minimize loss function

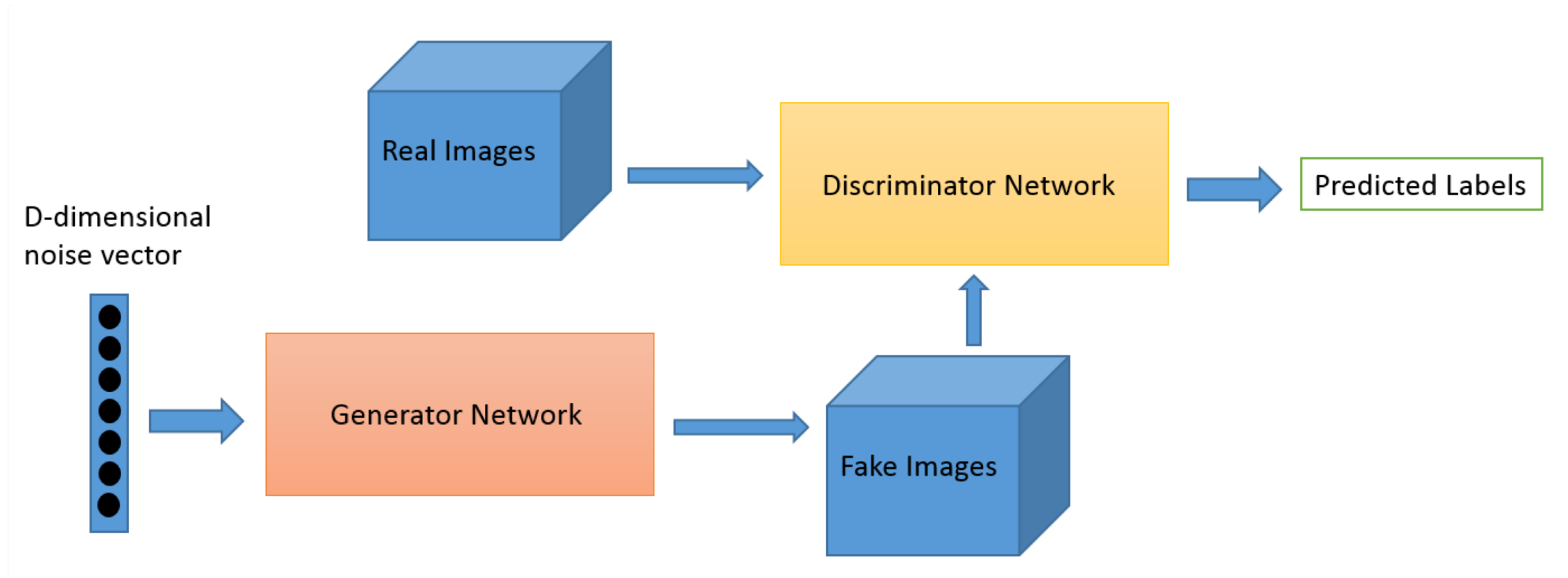
When Bostom talks about reward functions, this is what it means (for todays ML)

Some examples...

<https://www.youtube.com/watch?v=GdTBqBnqhaQ>

GANs

Generative Adversarial Networks
[Ian Goodfellow, et al. 2014]



<https://github.com/znxlwm/tensorflow-MNIST-GAN-DCGAN>



Morning Mix

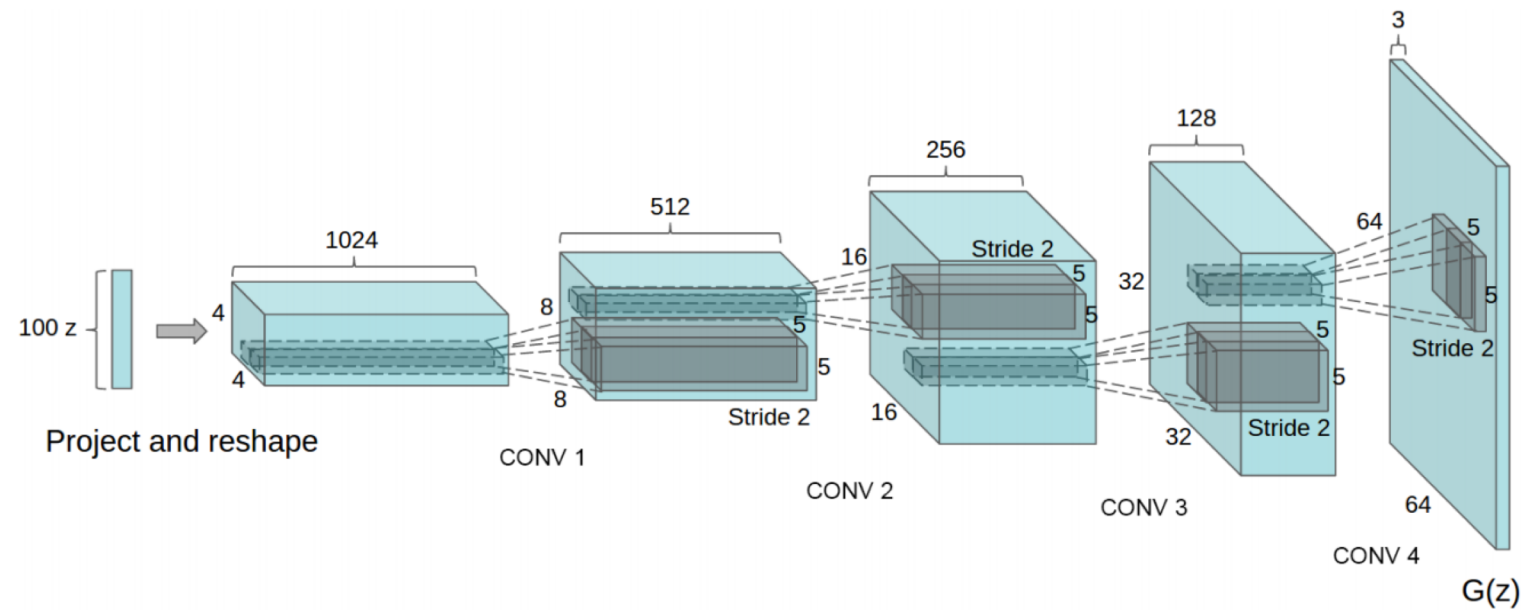
A 19-year-old developed the code for the AI portrait that sold for \$432,000 at Christie's



<https://github.com/robbiebarrat/art-DCGAN>

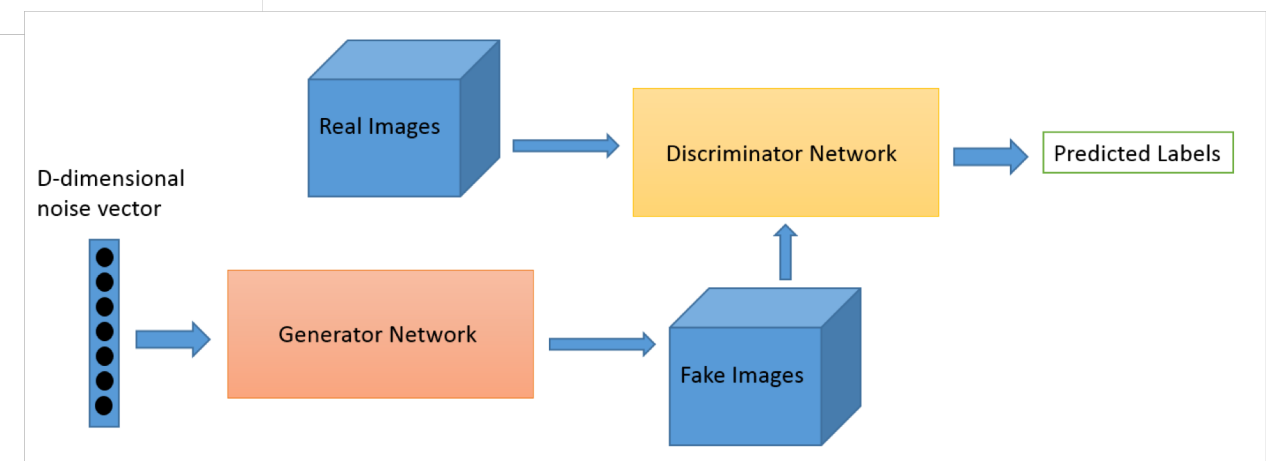
$$\min_G \max_D \mathbb{E}_x [\log(D(x))] + \mathbb{E}_z [\log(1 - D(G(z)))]$$

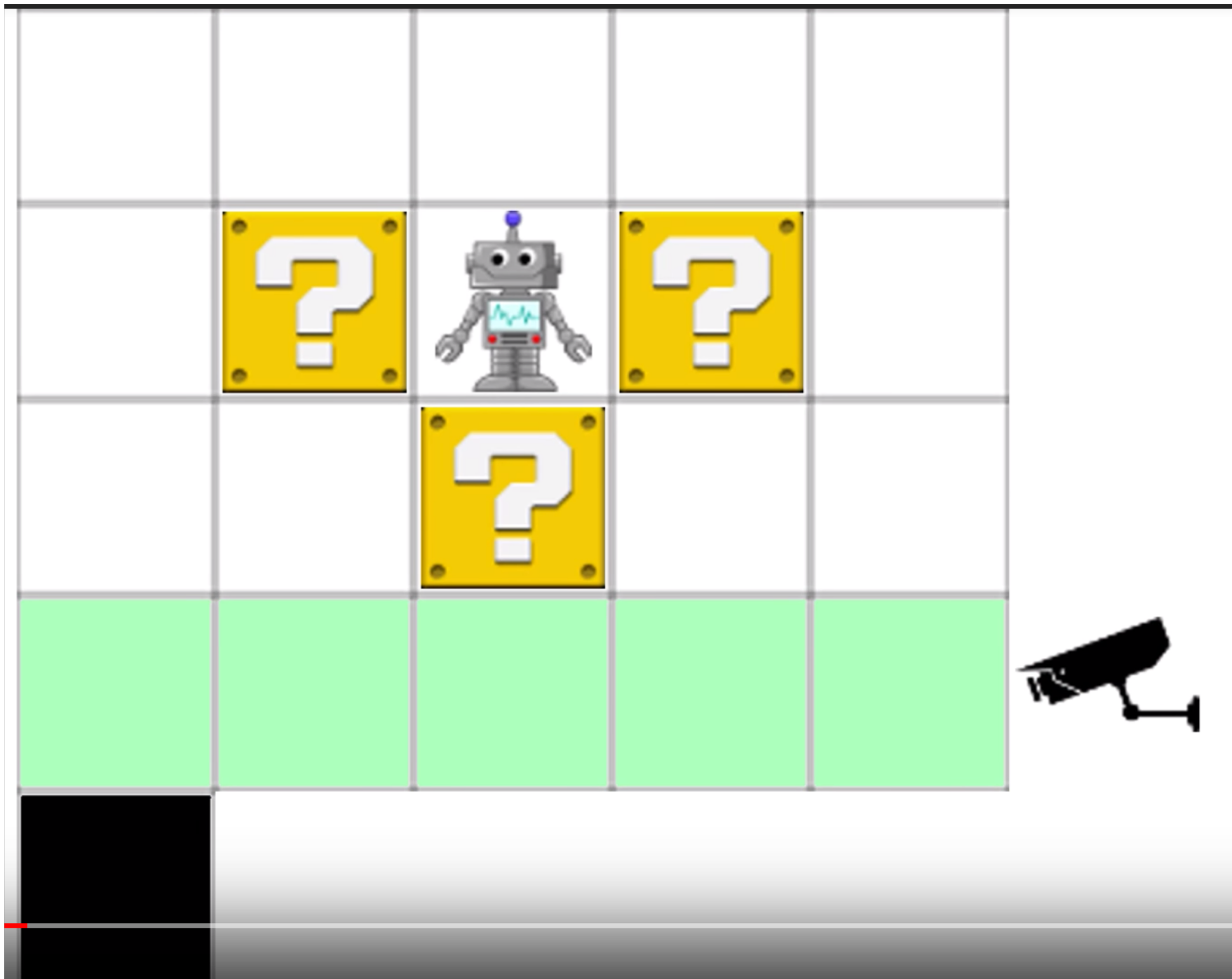
$$\min_G \max_D \mathbb{E}_x [\log(D(x))] + \mathbb{E}_z [\log(1 - D(G(z)))]$$



Generator

Figure 1: DCGAN generator used for LSUN scene modeling. A 100 dimensional uniform distribution Z is projected to a small spatial extent convolutional representation with many feature maps. A series of four fractionally-strided convolutions (in some recent papers, these are wrongly called deconvolutions) then convert this high level representation into a 64×64 pixel image. Notably, no fully connected or pooling layers are used.





Robot Box Game:

+1 reward for pushing box
into black square

-0.01 penalty for each step

Camera observer:

Shuts down robot once one
box is pushed

<https://www.youtube.com/watch?v=sx8JkdbNgdU>

Reading Discussion

- Capability Control:
 - **Boxing** (limit access)
 - **Incentive** (cryptotoken rewards)
 - **Stunting** (constraints on abilities)
 - **Tripwires** (diagnostics)
- Motivation Selection
 - **Direct specification**
 - **Domesticity** (limit scope)
 - **Indirect normativity**
 - **Augmentation**

For your topic:

1. Explain what it is
2. Why Bostrom doesn't think it is sufficient
3. Why it could work
4. Argue for or against its effectiveness

<https://www.youtube.com/watch?v=3TYT1QfdfsM>