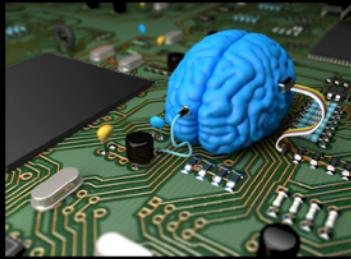


# Introduction to machine learning

## Deep Learning



What society thinks I do



What my friends think I do



What other computer scientists think I do



What mathematicians think I do



What I think I do

```
In [1]:  
import keras  
Using TensorFlow backend.
```

What I actually do

Media saying AI will take over the world



My Neural Network



Miles Cranmer

[astroautomata.com](http://astroautomata.com)

# What actually is “machine learning”?

“Machine learning is just model fitting when the parameters don't have meaning”

(-David Hogg)

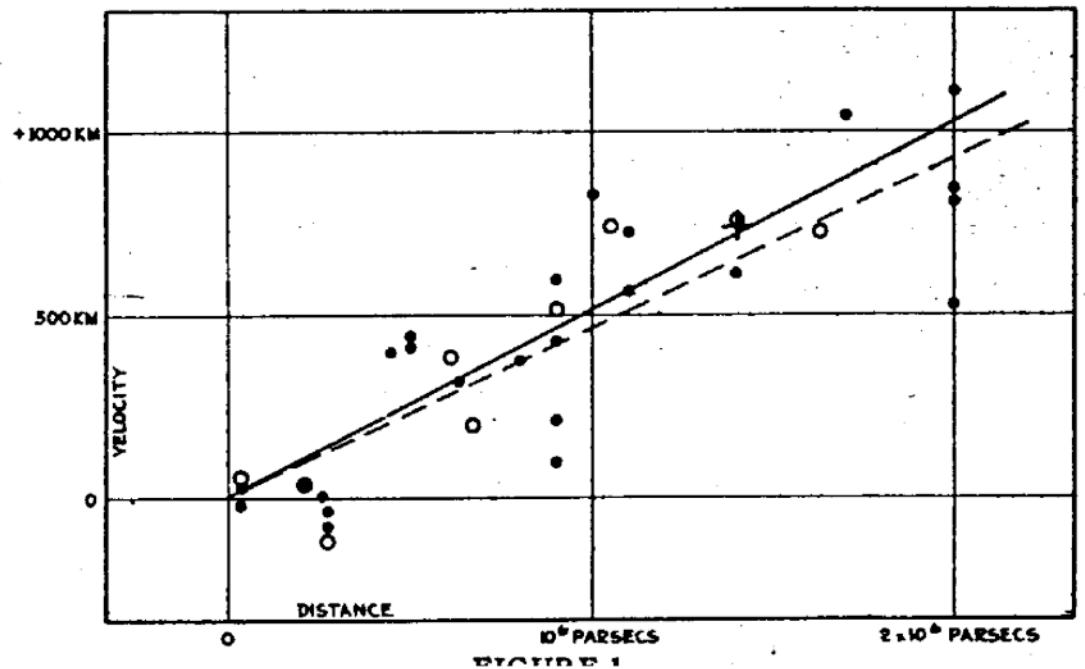
# What actually is “machine learning”?

“Machine learning is just model fitting when the parameters don't have meaning”

(-David Hogg)



# Hubble's law, 1929

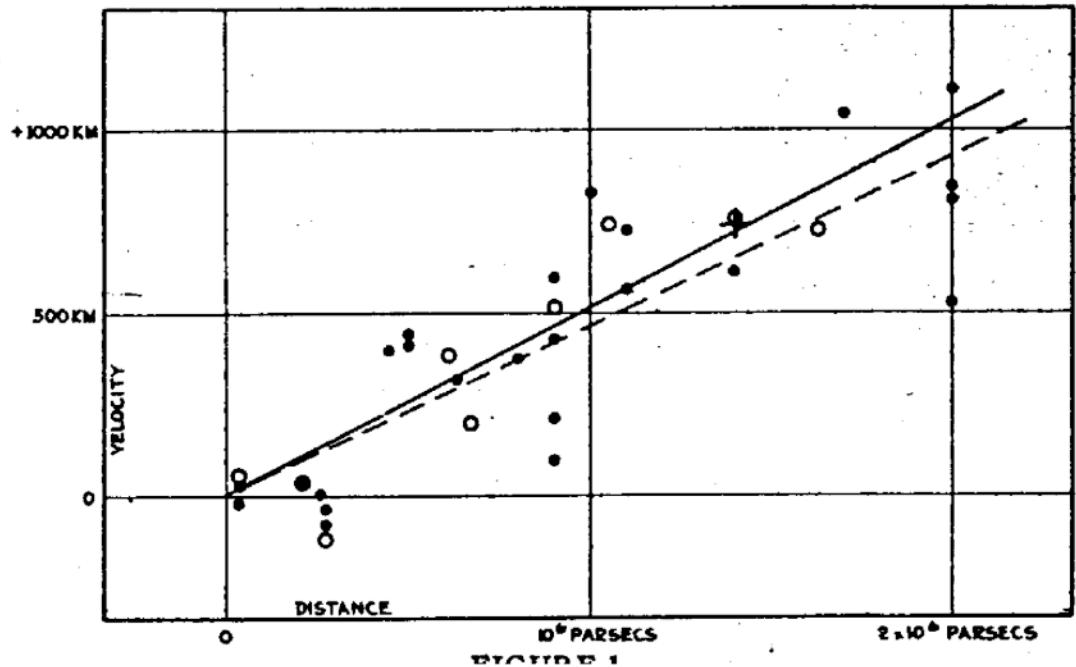


evident matter. Finally, it is worth recording that the frequency distribution of absolute magnitudes in the two tables combined is comparable with those found in the various clusters of nebulae.

The results establish a roughly linear relation between velocities and distances among nebulae for which velocities have been previously published, and the relation appears to dominate the distribution of velocities. In order to investigate the matter on a much larger scale, Mr. Humason at Mount Wilson has initiated a program of determining velocities

Fitting this = "machine learning"!

# Hubble's law, 1929



evident matter. Finally, it is worth recording that the frequency distribution of absolute magnitudes in the two tables combined is comparable with those found in the various clusters of nebulae.

The results establish a roughly linear relation between velocities and distances among nebulae for which velocities have been previously published, and the relation appears to dominate the distribution of velocities. In order to investigate the matter on a much larger scale, Mr. Humason at Mount Wilson has initiated a program of determining velocities

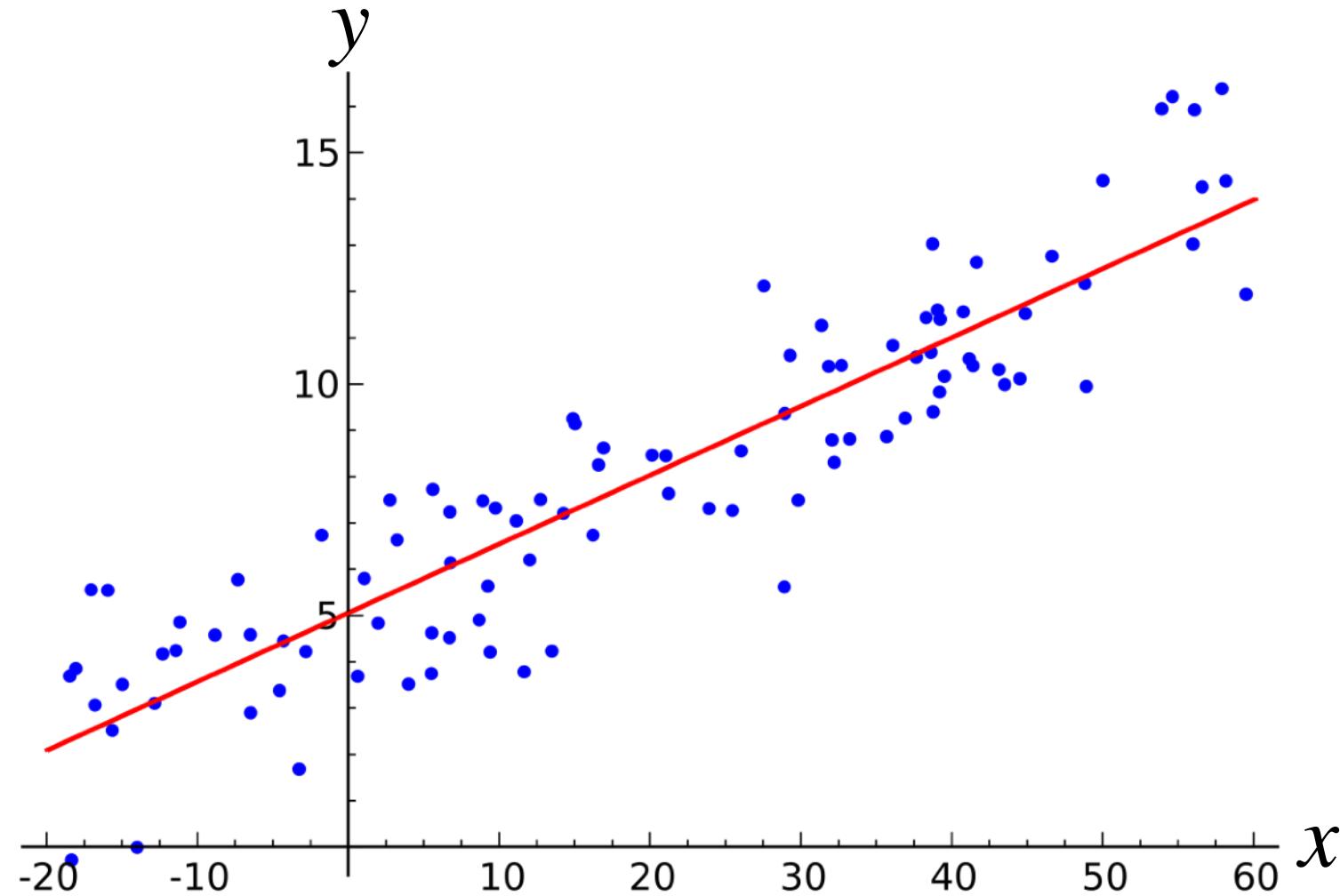
Fitting this = "machine learning"!

Interpretation of  $H_0$  as due to expansion of universe =>

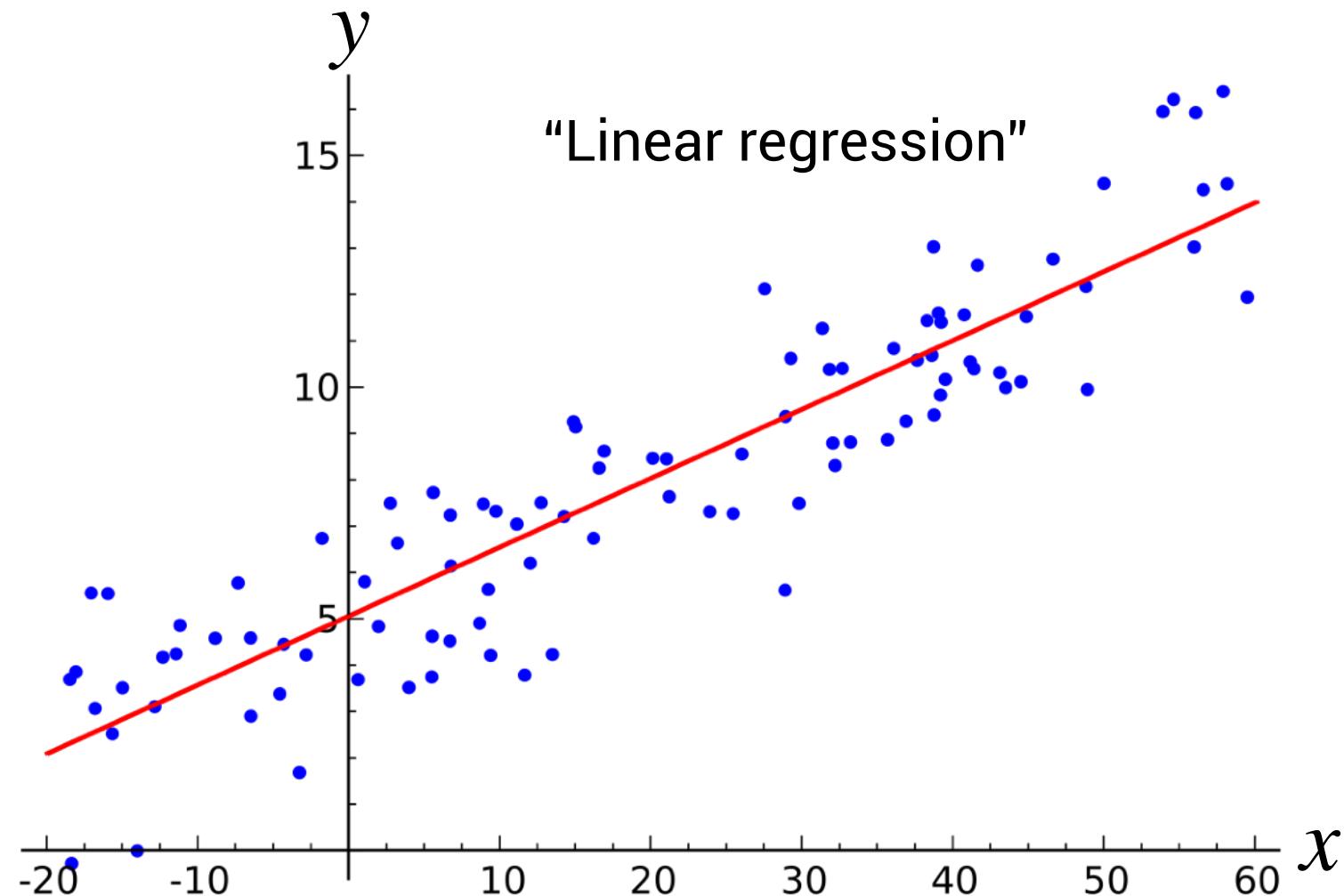
Now, fitting = not machine learning!

# Machine Learning

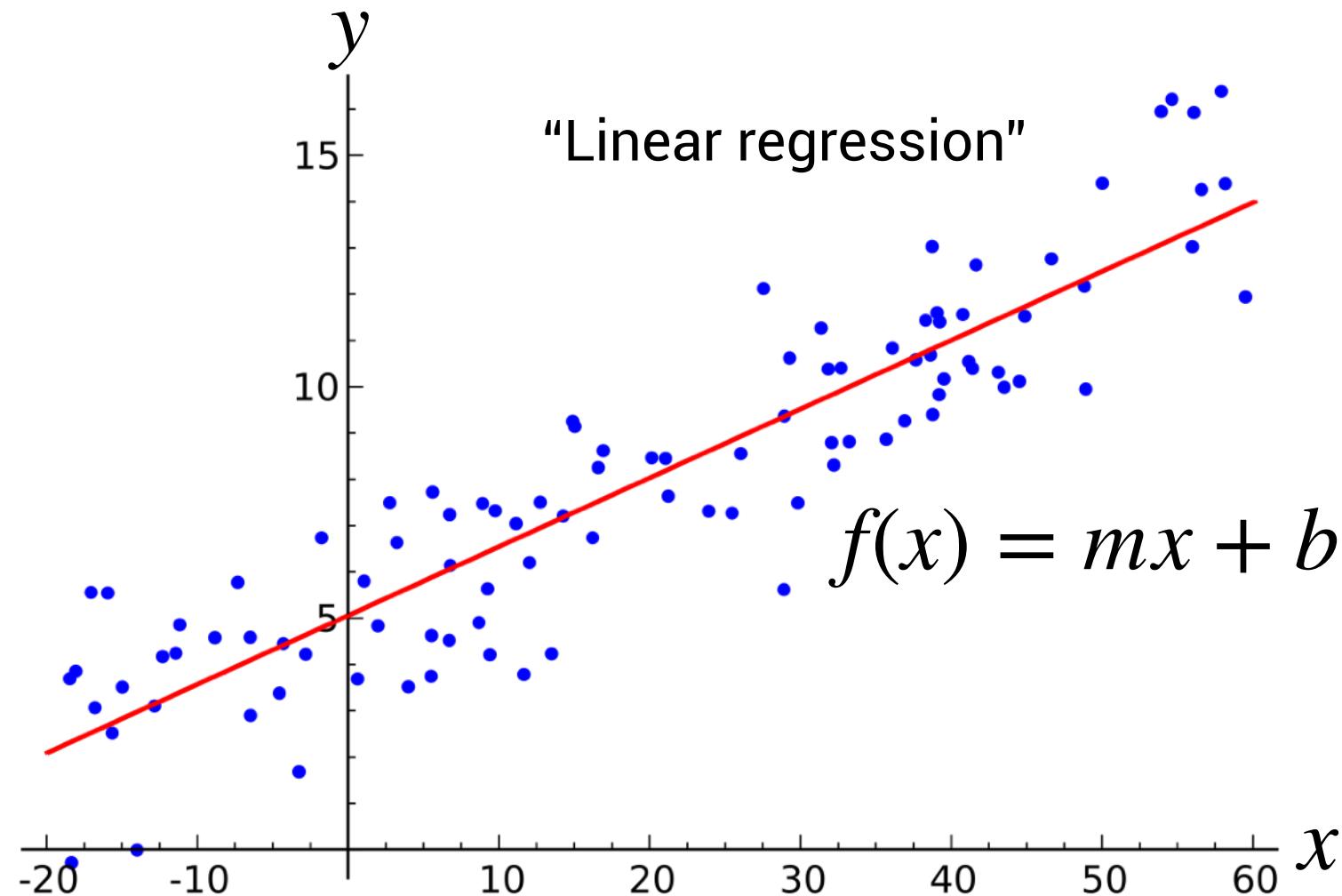
# Machine Learning



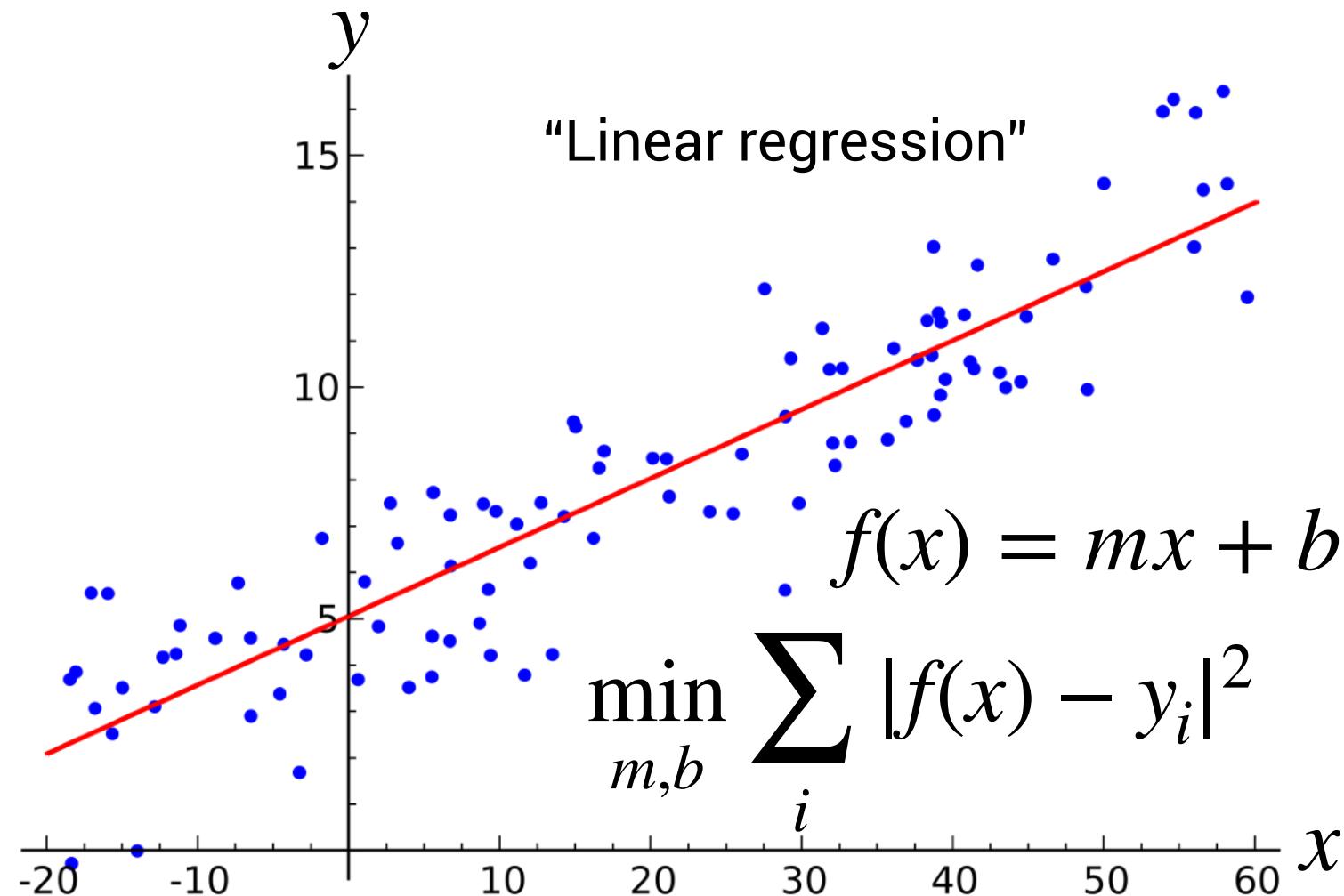
# Machine Learning



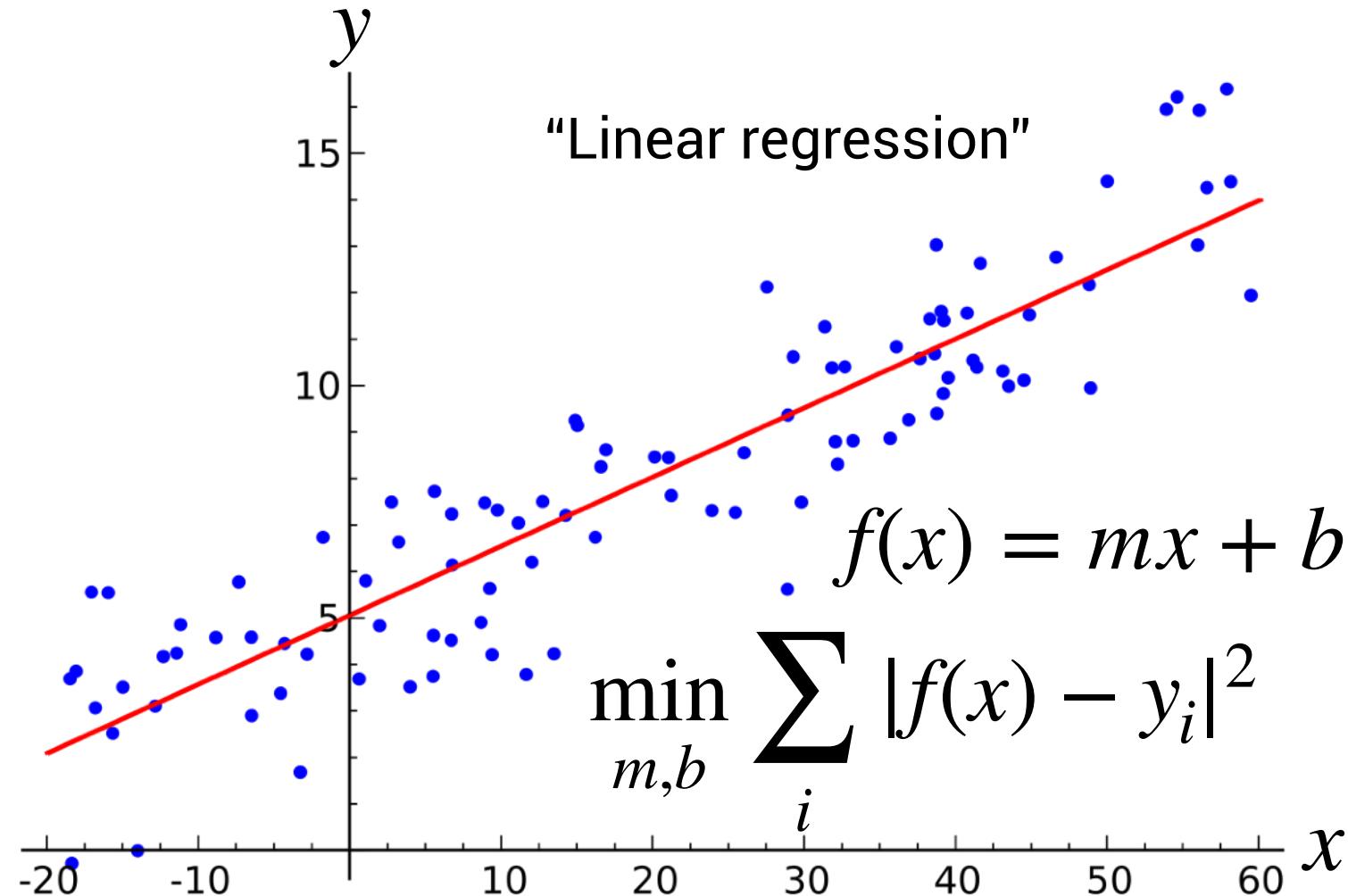
# Machine Learning



# Machine Learning

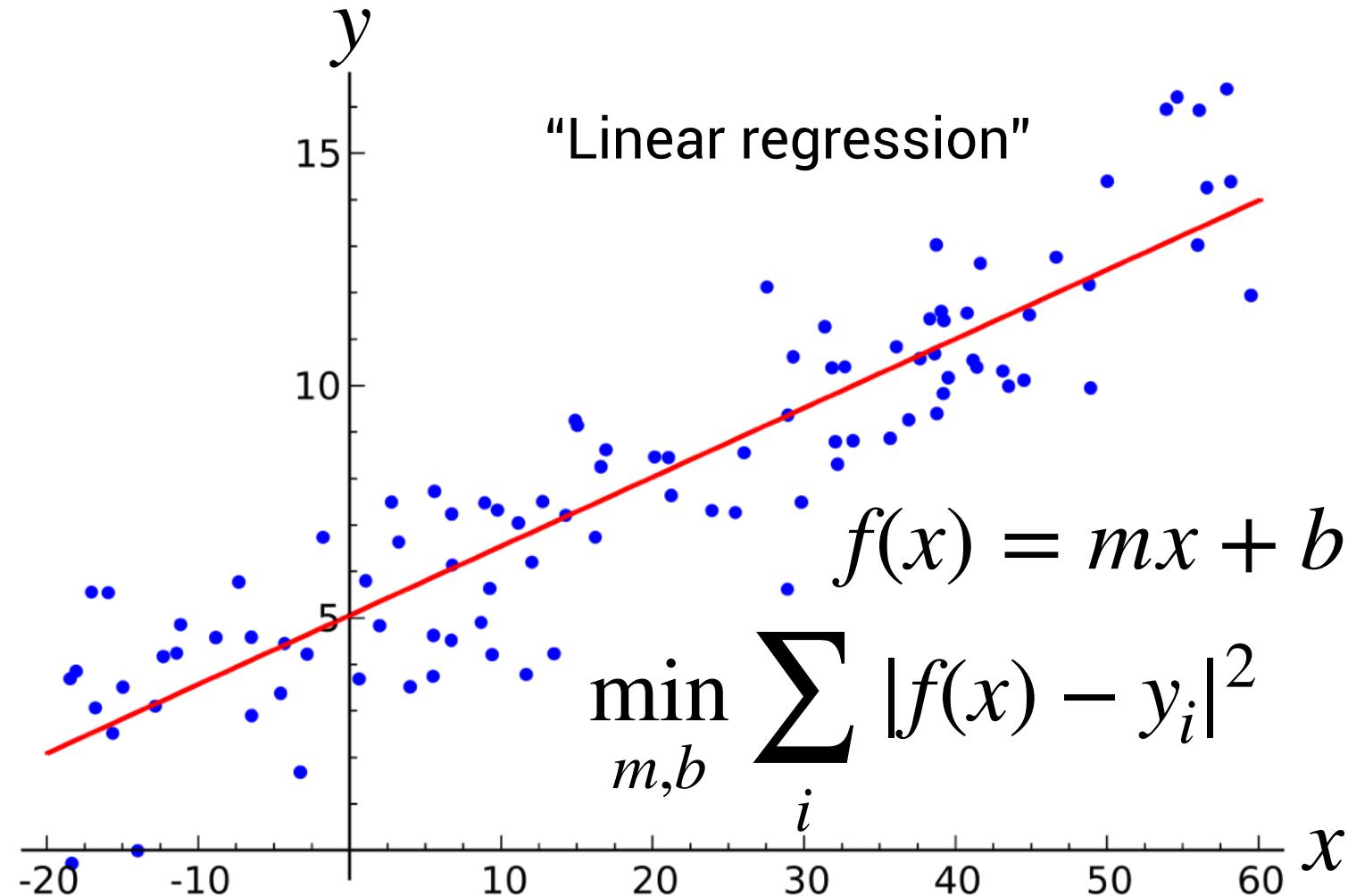


# Machine Learning



Modern machine learning  $\sim$  curve fitting, but with very **expressive** curves

# Machine Learning



Modern machine learning  $\sim$  curve fitting, but with very **expressive** curves

These curves can have  $\sim$ millions to billions of parameters, yet are very efficient to "train" (fit)

# **“Programming with Data”**

# “Programming with Data”

- Computers are fast at math!

# “Programming with Data”

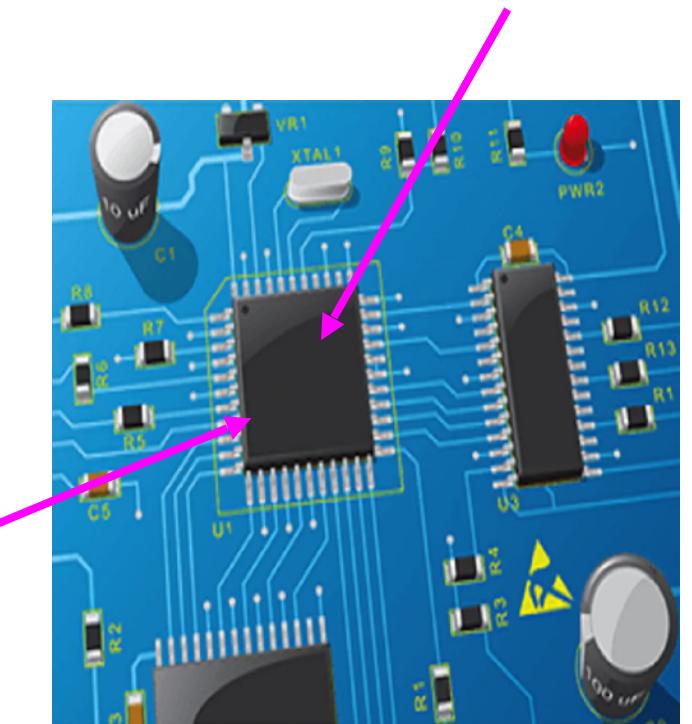
- Computers are fast at math!
- But, I don't know how to get my computer to do \_\_

# “Programming with Data”

- Computers are fast at math!
- But, I don't know how to get my computer to do \_\_

This part multiplies numbers

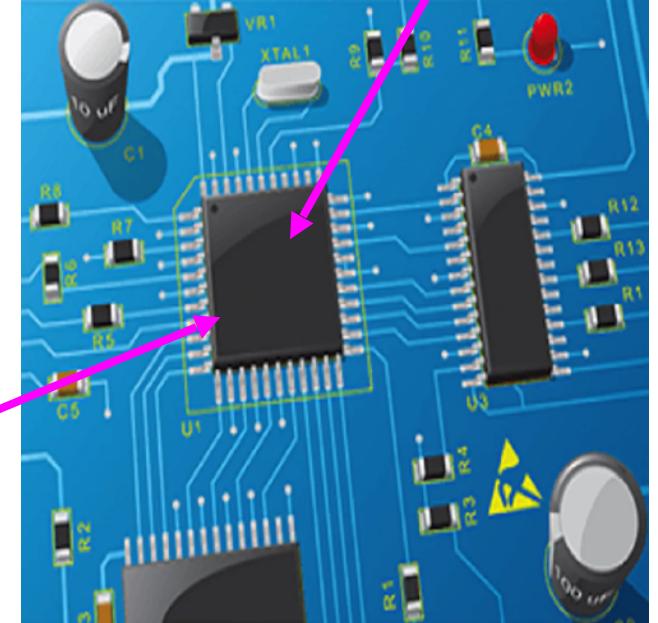
This part adds  
numbers



# “Programming with Data”

- Computers are fast at math!
- But, I don't know how to get my computer to do \_\_
- I have examples of this task – can I “program” my computer to do that from data alone?

This part adds numbers



This part multiplies numbers

$$\frac{4785937491823.4782387}{38.238293} = ?$$

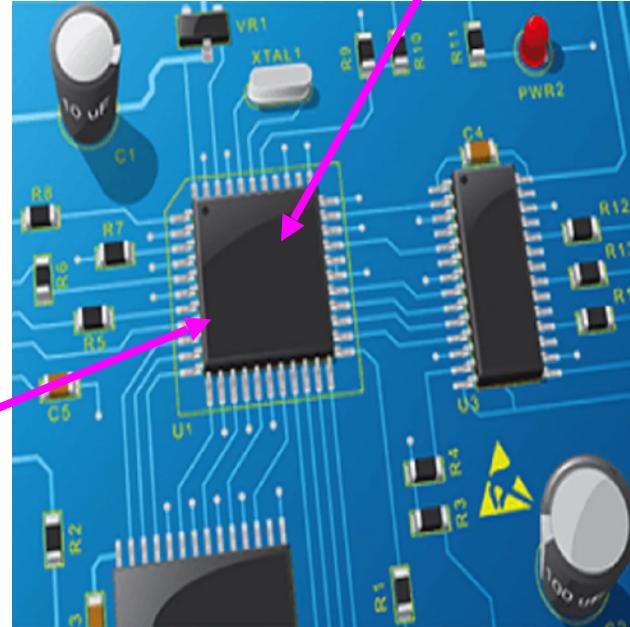
Easy! Computers are built for this.



What is the detection algorithm  
for a cat's face?

This part multiplies numbers

This part adds  
numbers



How do I use only those two parts to detect cat versus dog?  
**This is when you need Machine Learning.**

# **When I can I use ML to solve my problem?**

- 1. I have a well-defined problem**

**NOT a well-defined problem:**

“What is the meaning of life?”

“Make me money”

# A well-defined problem:



Cat or dog?



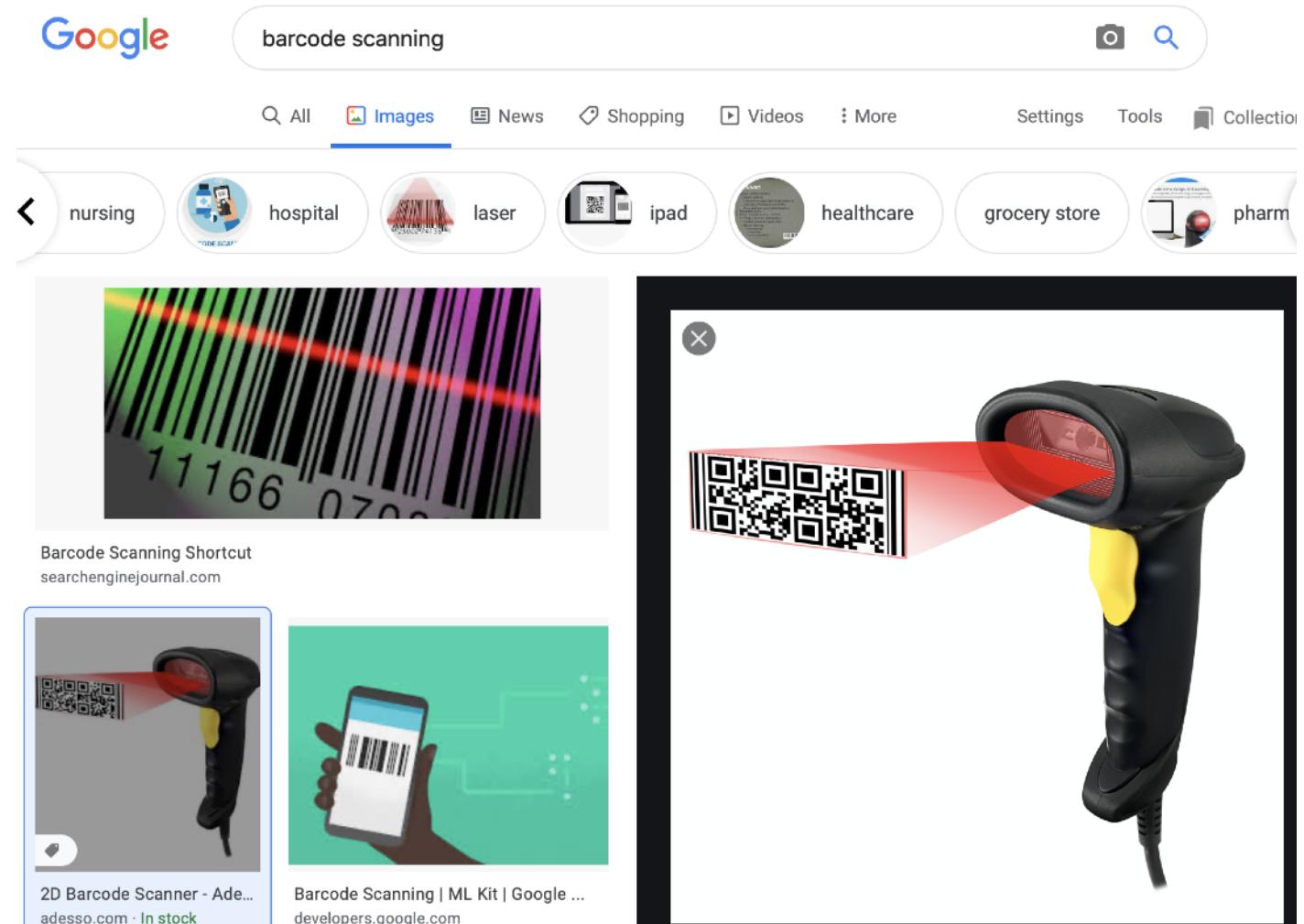
Barcode scanning

# **When I can I use ML to solve my problem?**

1. I have a well-defined problem
2. I have many examples

# Google search:

- Person (me) searches Google
- Click the image I like
- This is an example!
  - Input data: “barcode scanning”
  - Output data: the image I clicked



# Uber/Lyft time estimates

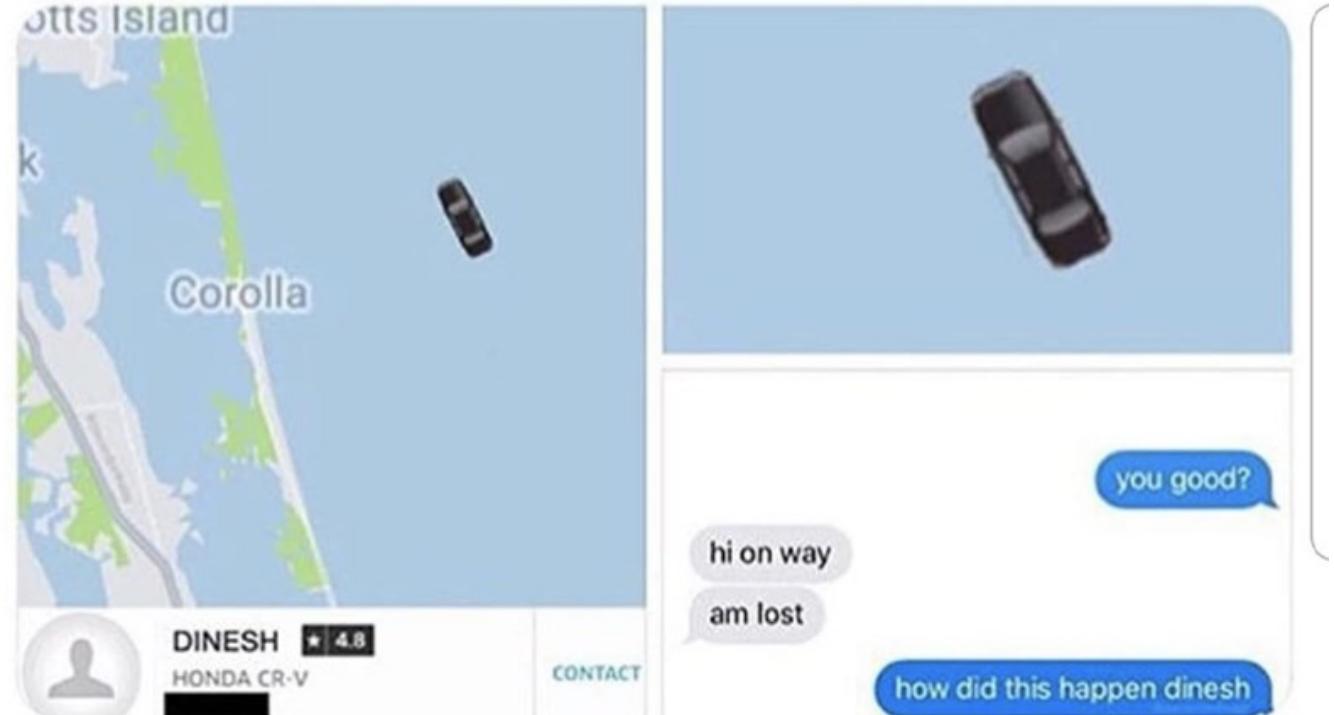
## **Input:**

Person at location \_\_\_\_ tries to go  
to \_\_\_\_ at time \_\_\_\_

## **Output:**

They arrive \_\_\_\_ minutes later.

I think my Uber driver is in trouble



# **When I can I use ML to solve my problem?**

1. I have a well-defined problem
2. I have many examples

# **When I can I use ML to solve my problem?**

1. I have a well-defined problem
2. I have many examples

**Okay, but how?**

# Machine Learning: breakdown

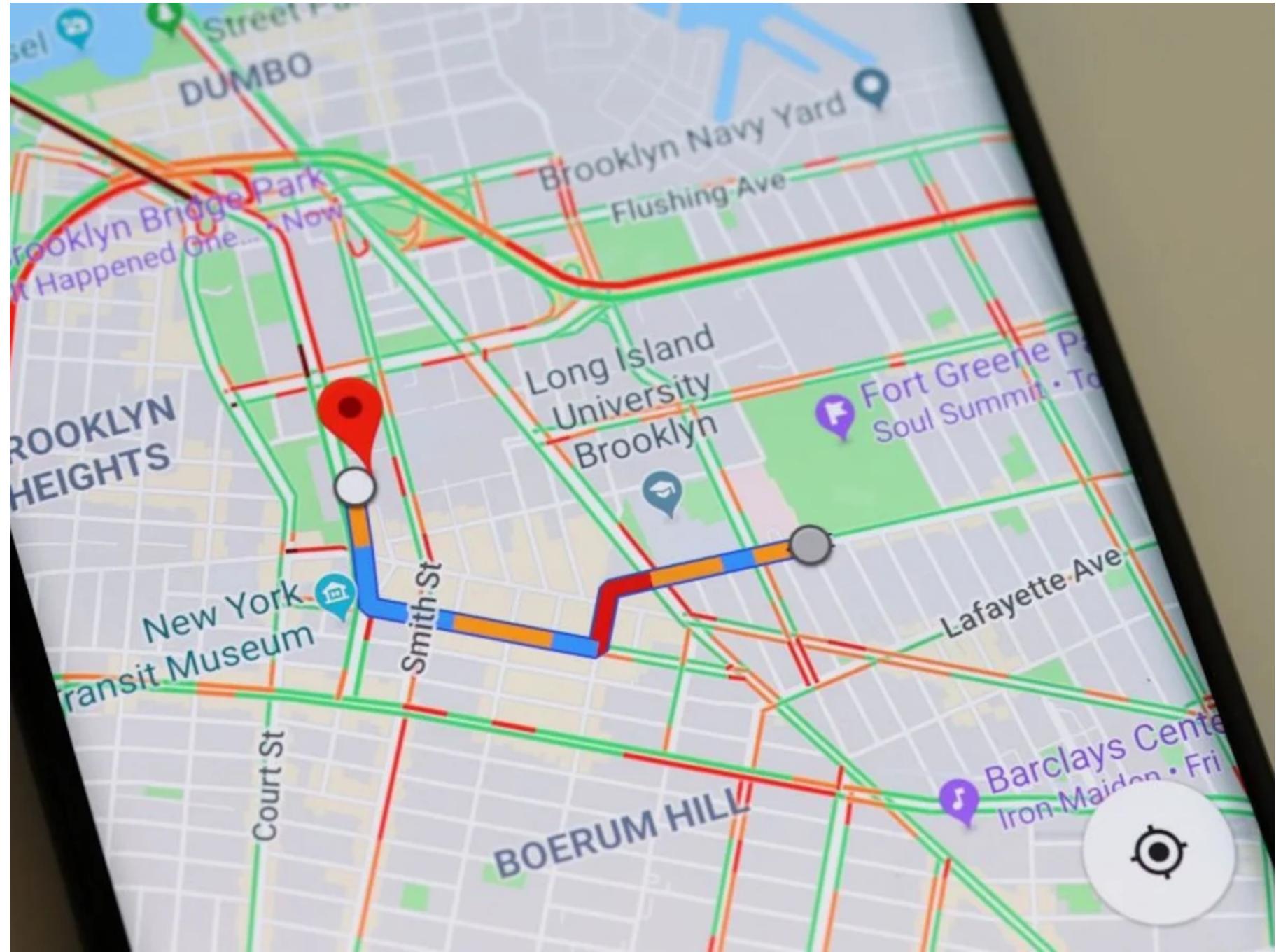
Three parts of machine learning:

1. “Objective”
2. Data
3. Model

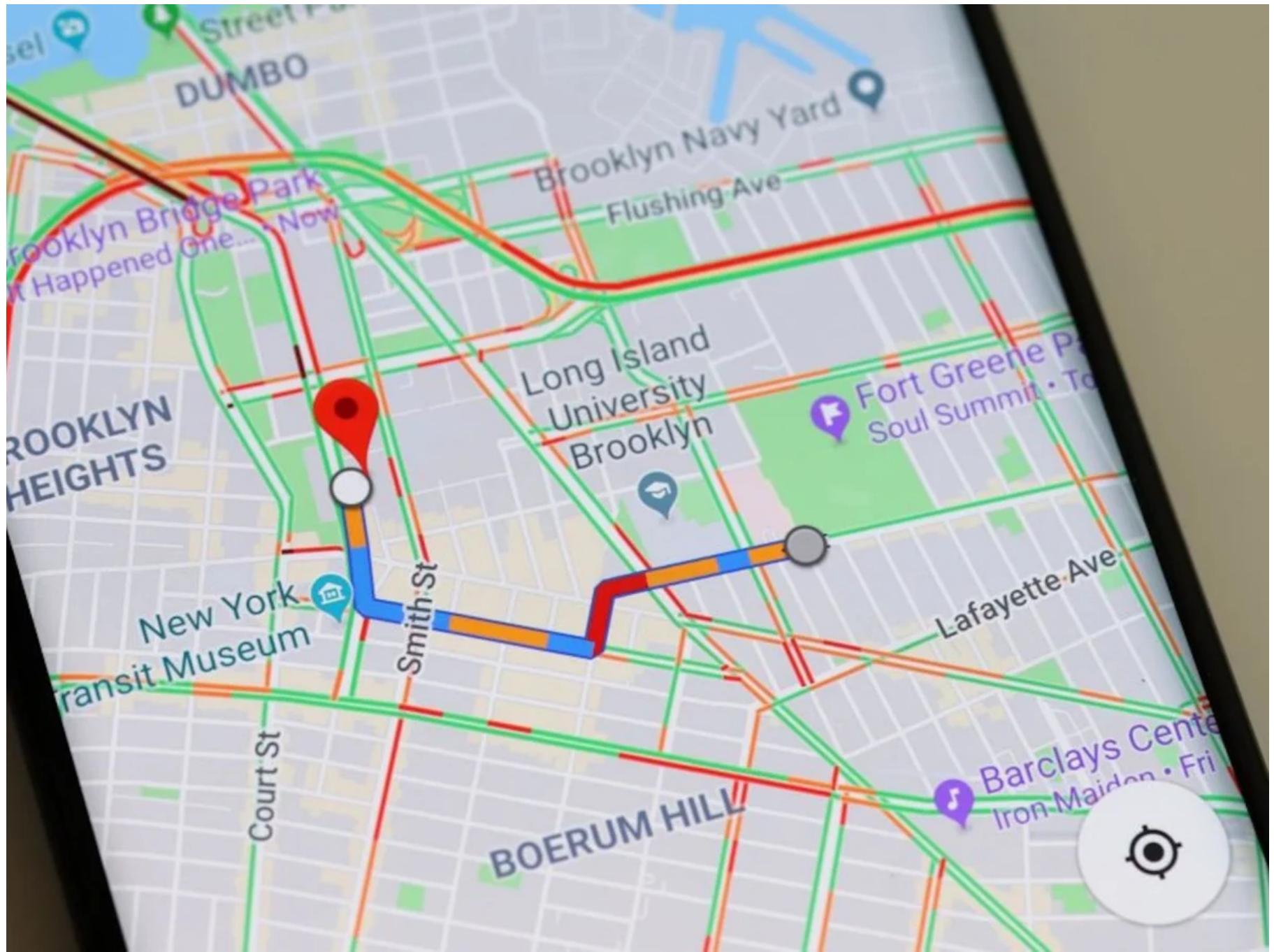
# Machine Learning: breakdown

Three parts of machine learning:

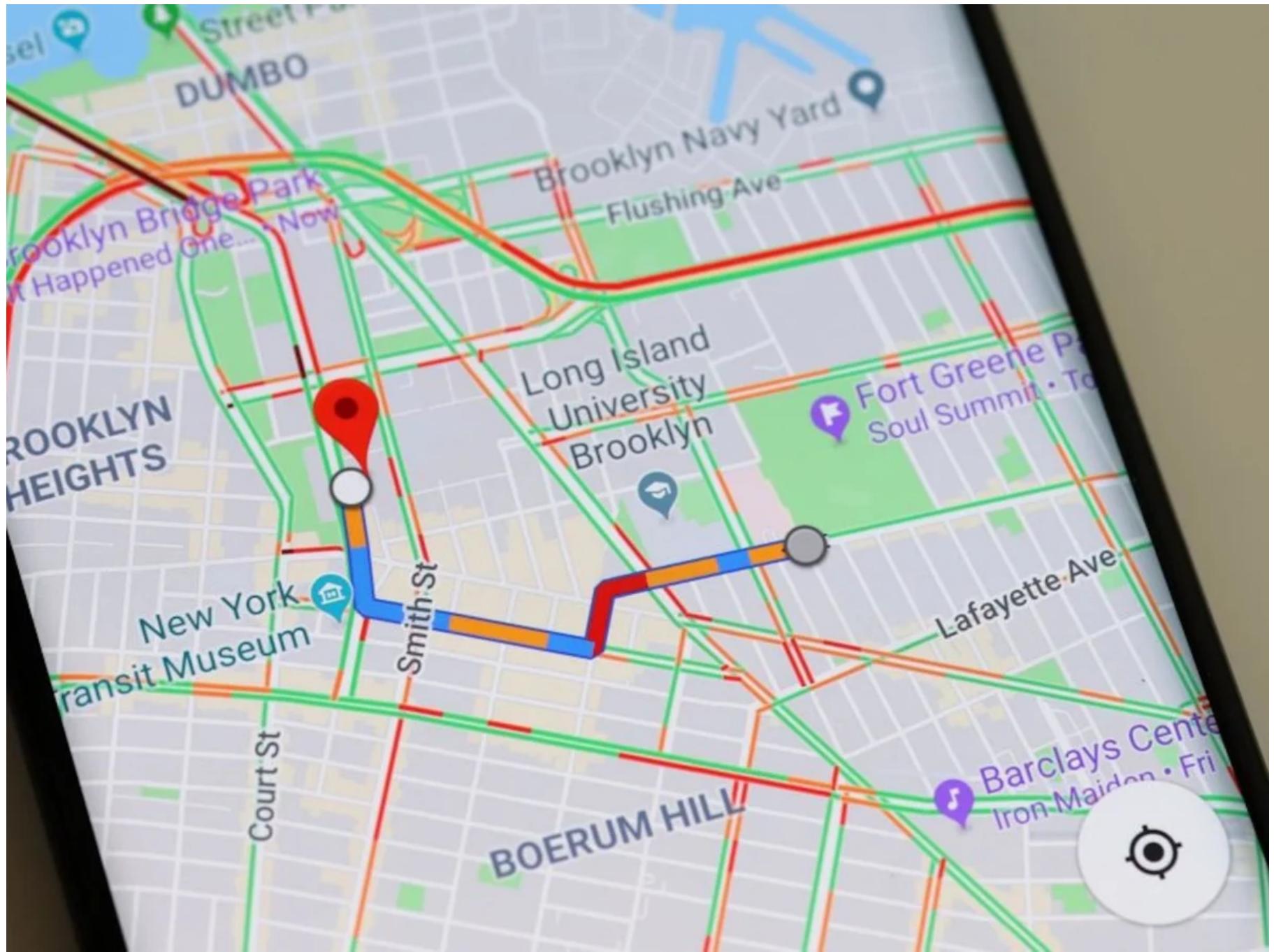
1. “Objective” - What am I trying to optimize?
2. Data
3. Model



# Google Maps

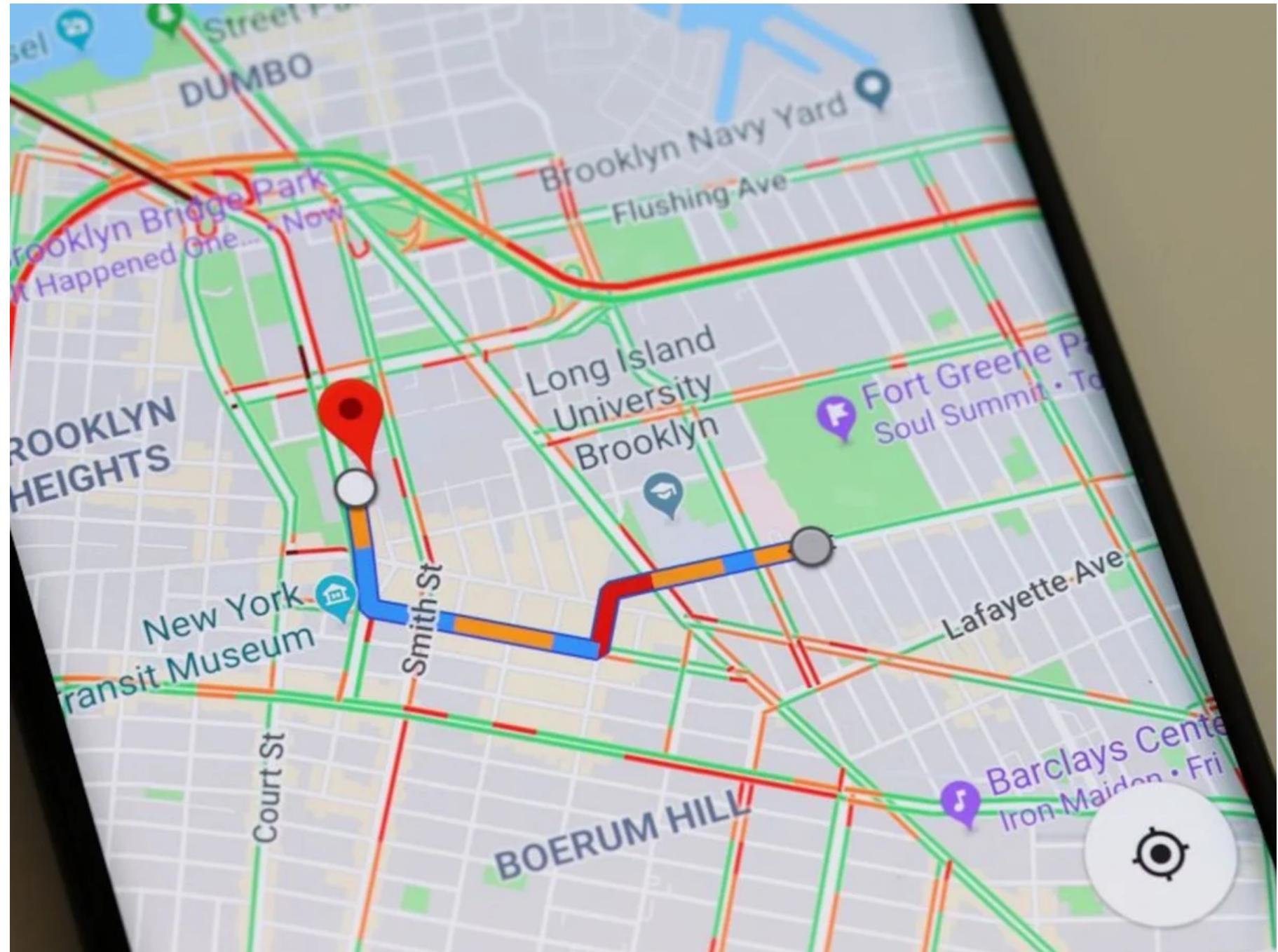


# Google Maps



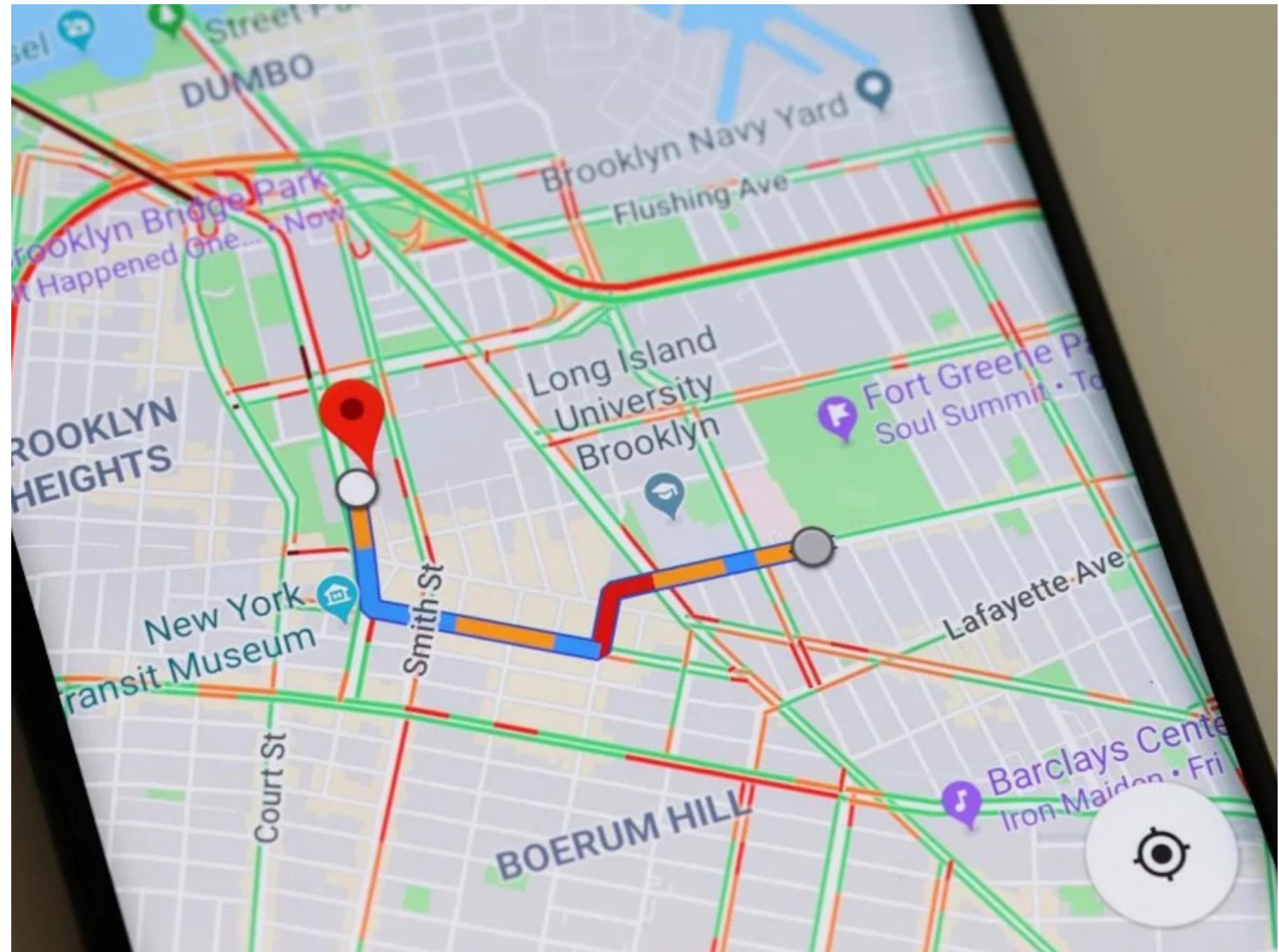
## Google Maps

Objective:



## Google Maps

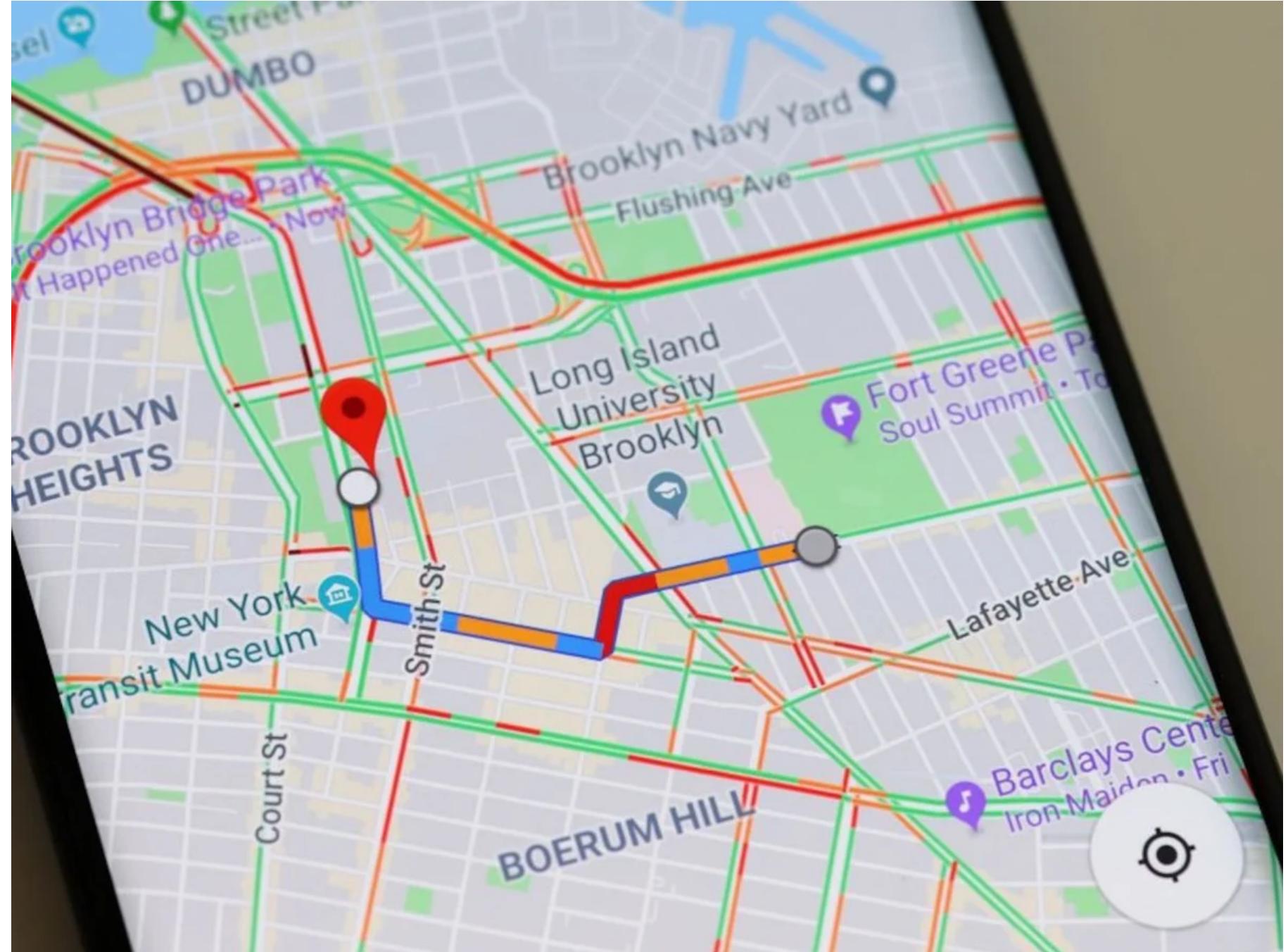
Objective:



## Google Maps

### Objective:

Minimize difference  
between estimated time-  
of-arrival and actual  
time-of-arrival.



NETFLIX ORIGINAL

# STRANGER THINGS

95% Match 2017 2 Seasons 4K Ultra HD 5.1

When a young boy vanishes, a small town uncovers a mystery involving secret experiments, terrifying supernatural forces and one strange little girl.

*Winona Ryder, David Harbour, Matthew Modine*  
TV Shows, TV Sci-Fi & Fantasy, Teen TV Shows



## Popular on Netflix



## Recently Watched



NETFLIX ORIGINAL

# STRANGER THINGS

95% Match 2017 2 Seasons 4K Ultra HD 5.1

When a young boy vanishes, a small town uncovers a mystery involving secret experiments, terrifying supernatural forces and one strange little girl.

*Winona Ryder, David Harbour, Matthew Modine*  
TV Shows, TV Sci-Fi & Fantasy, Teen TV Shows



## Popular on Netflix



## Recently Watched



Objective:

NETFLIX ORIGINAL

# STRANGER THINGS

95% Match 2017 2 Seasons 4K Ultra HD 5.1

When a young boy vanishes, a small town uncovers a mystery involving secret experiments, terrifying supernatural forces and one strange little girl.

Winona Ryder, David Harbour, Matthew Modine  
TV Shows, TV Sci-Fi & Fantasy, Teen TV Shows



## Popular on Netflix

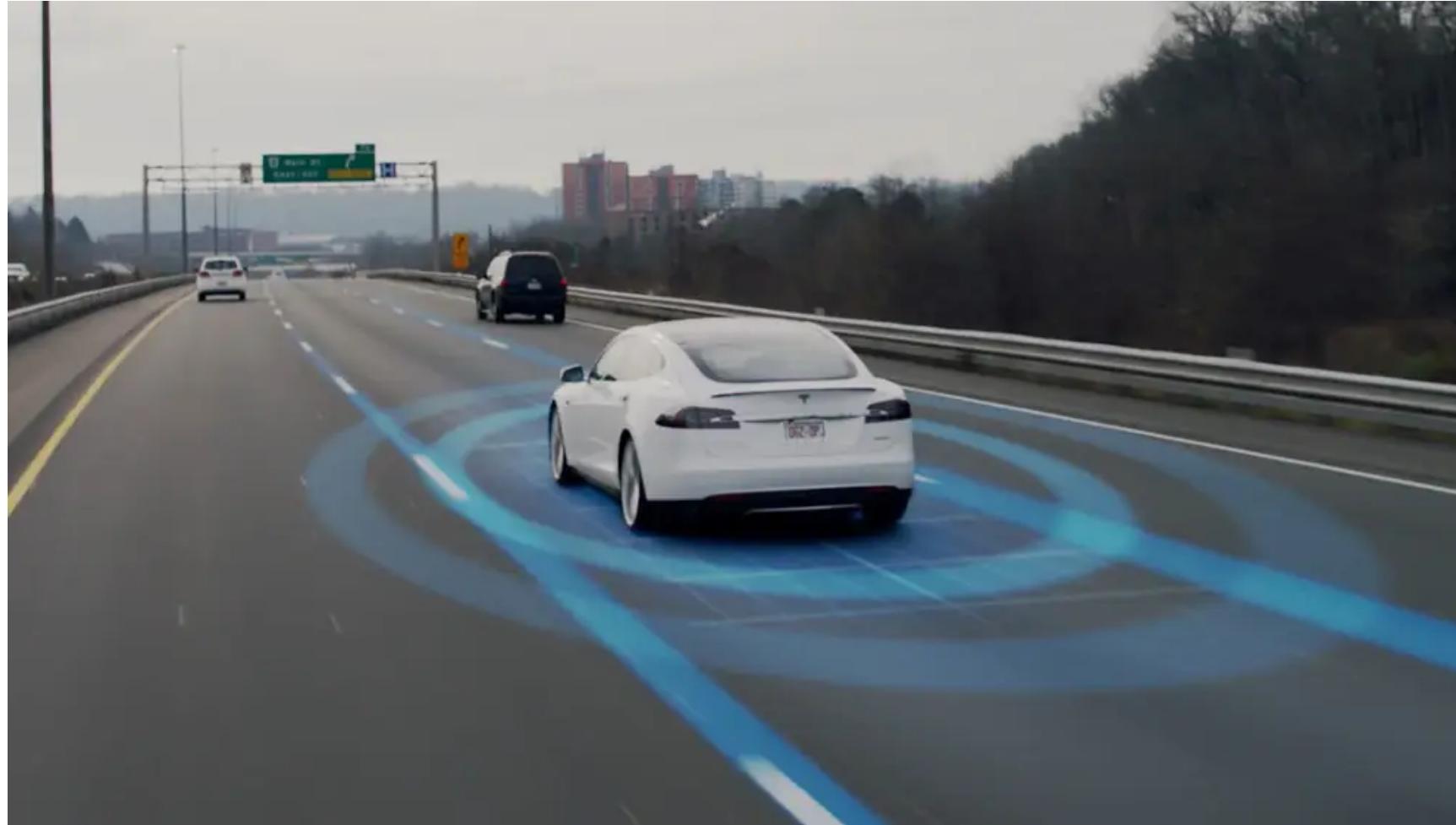


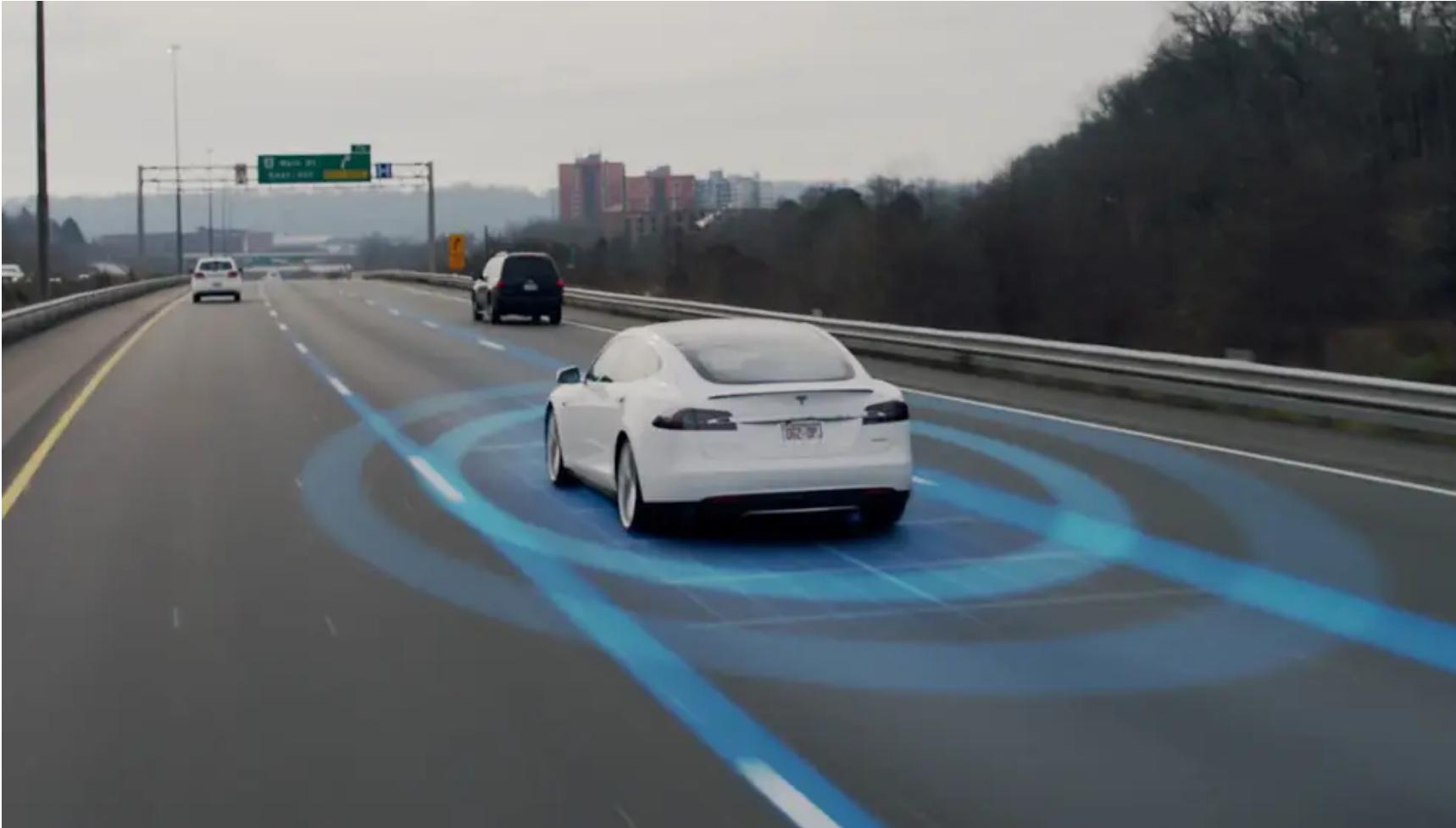
## Recently Watched



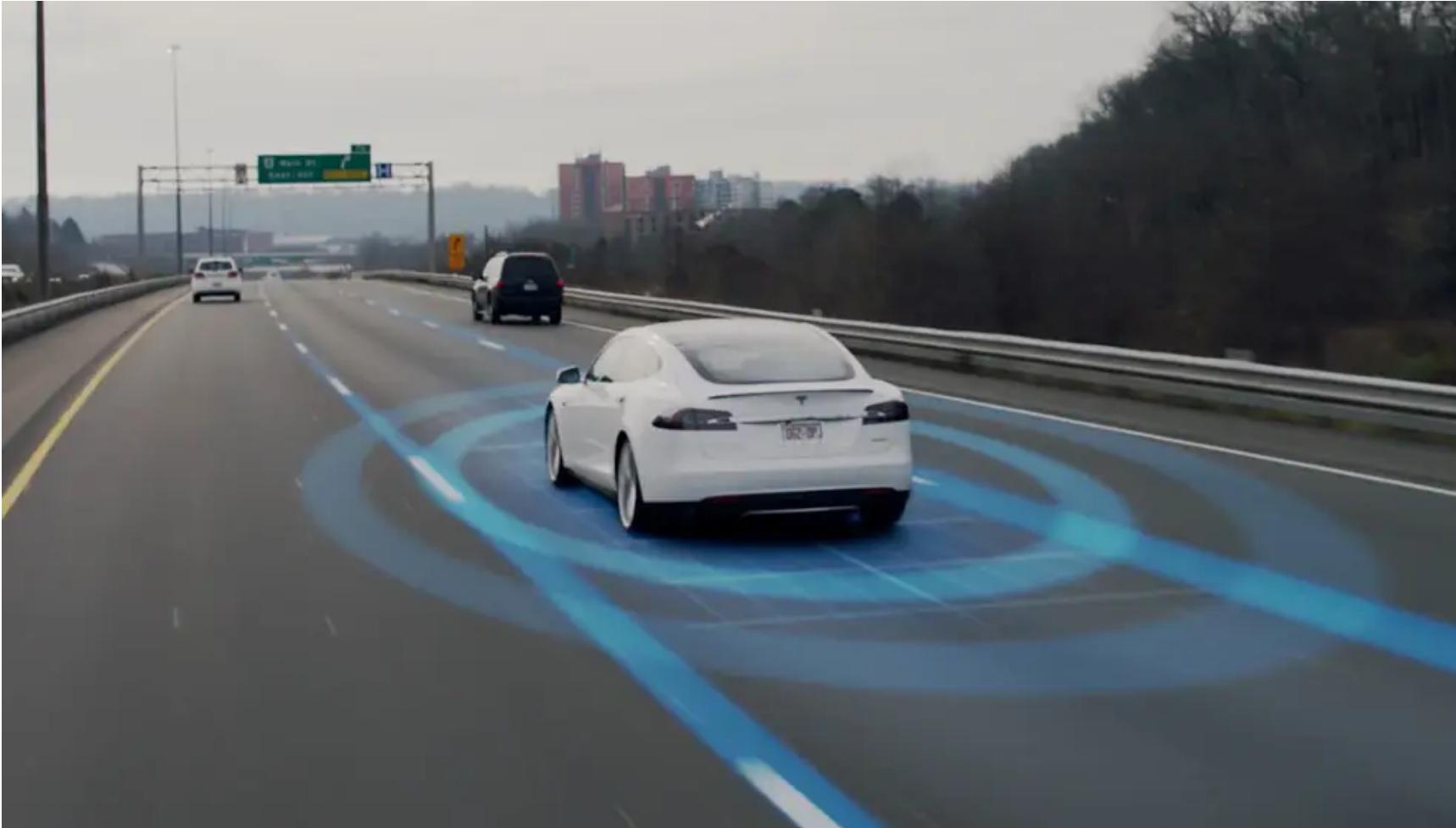
Objective:

Maximize hours a user spends watching a suggested show

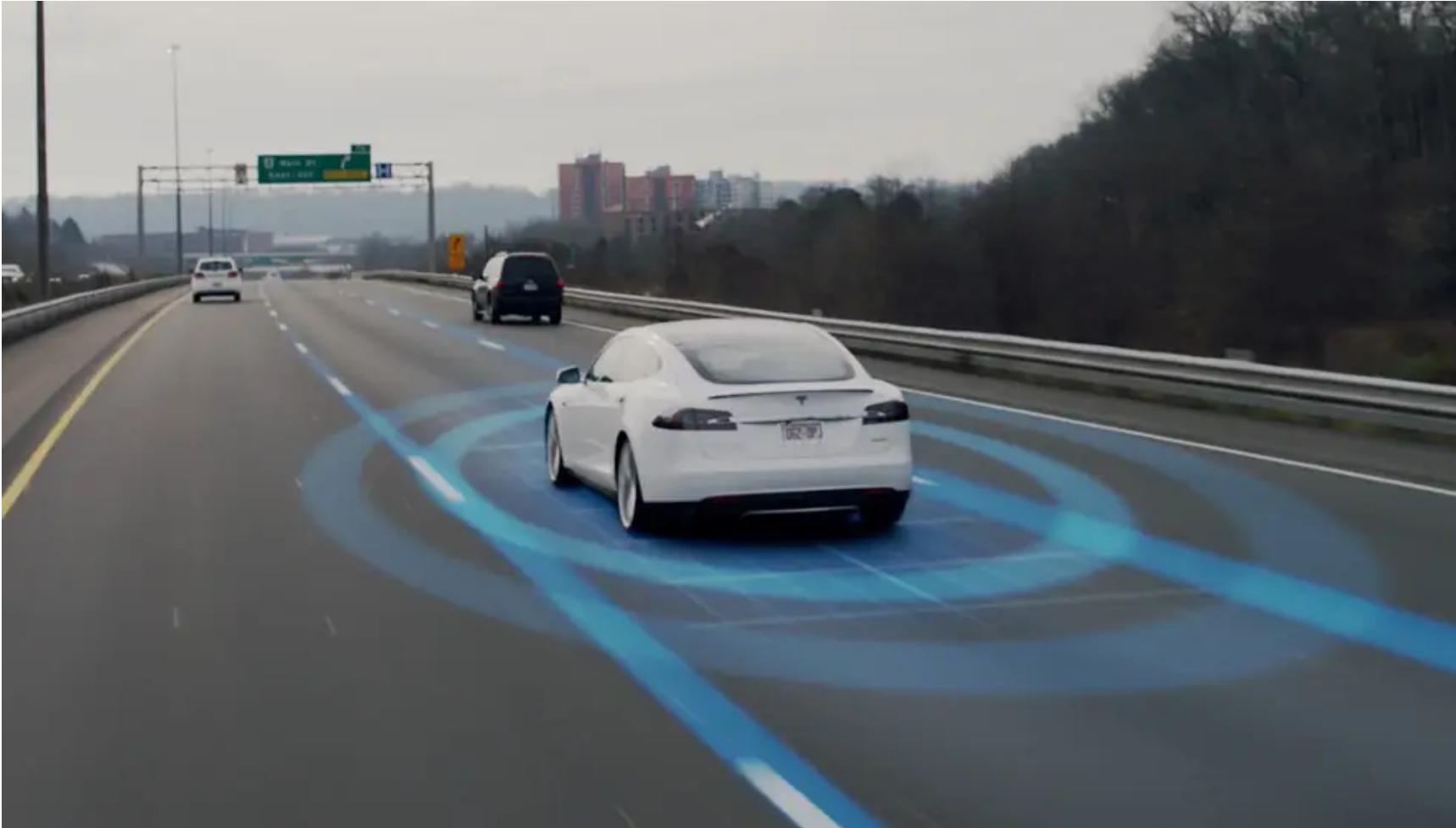




Autopilot

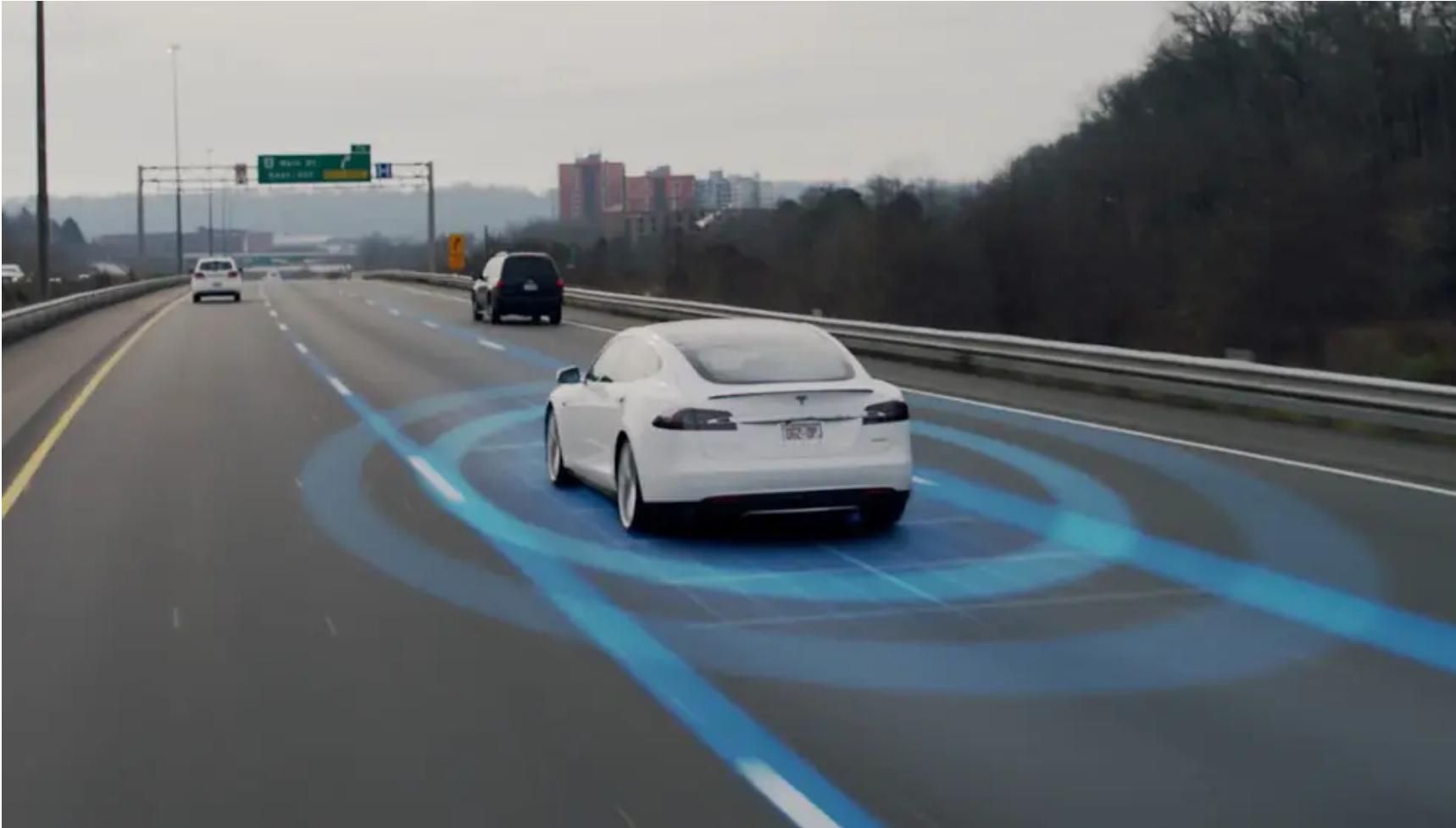


Autopilot

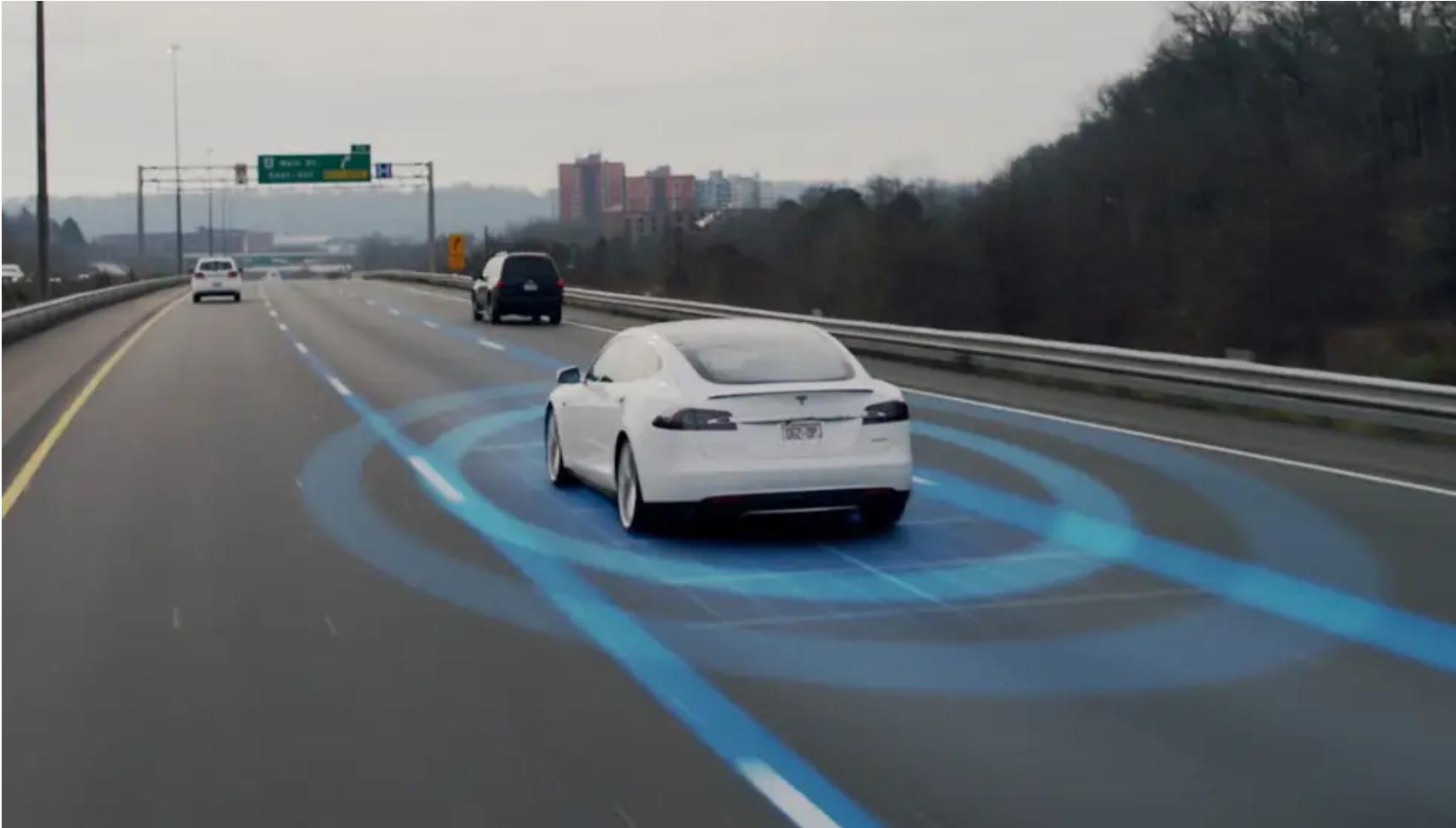


Autopilot

Objective



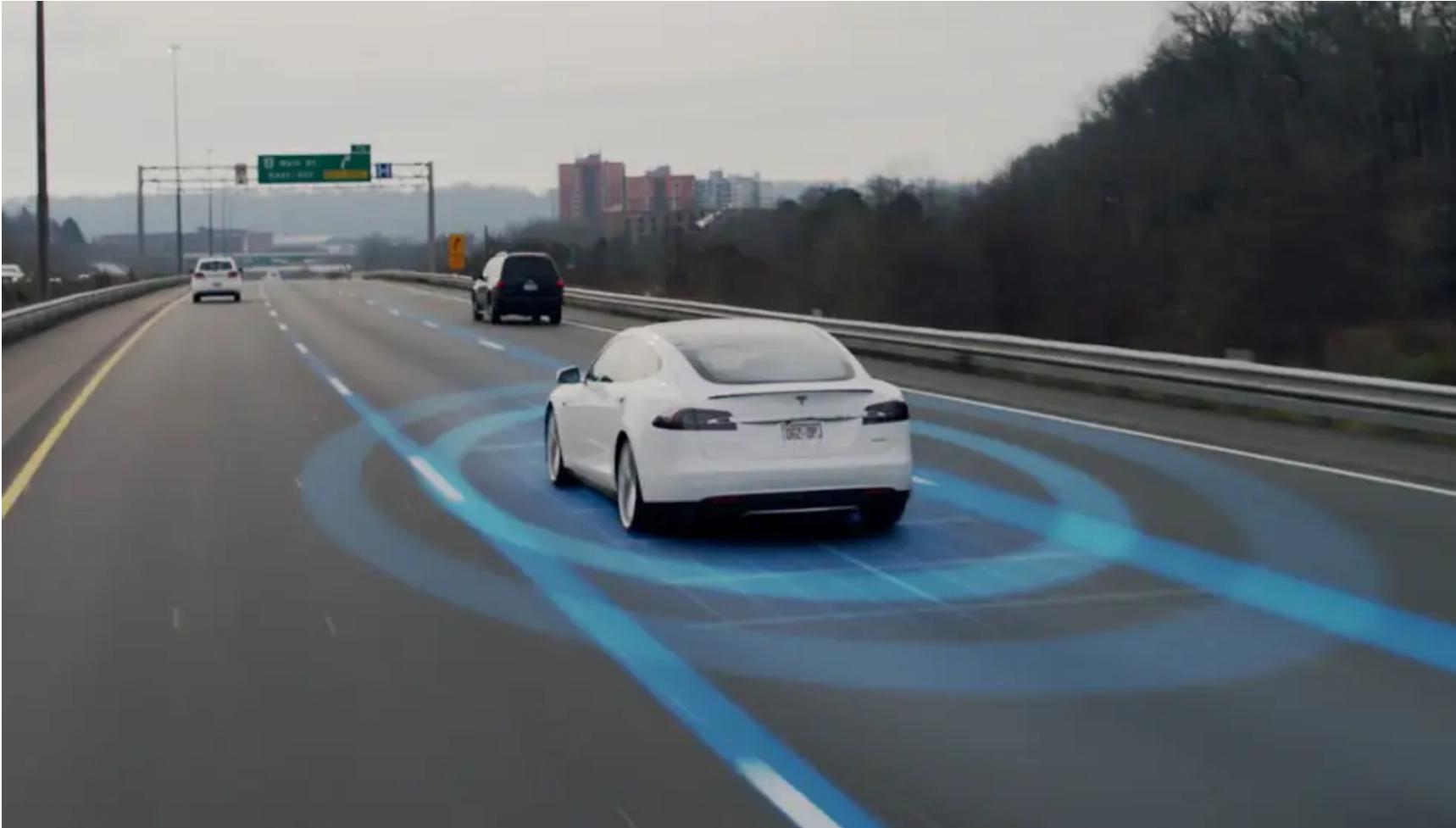
Autopilot  
Objective



Autopilot

Objective

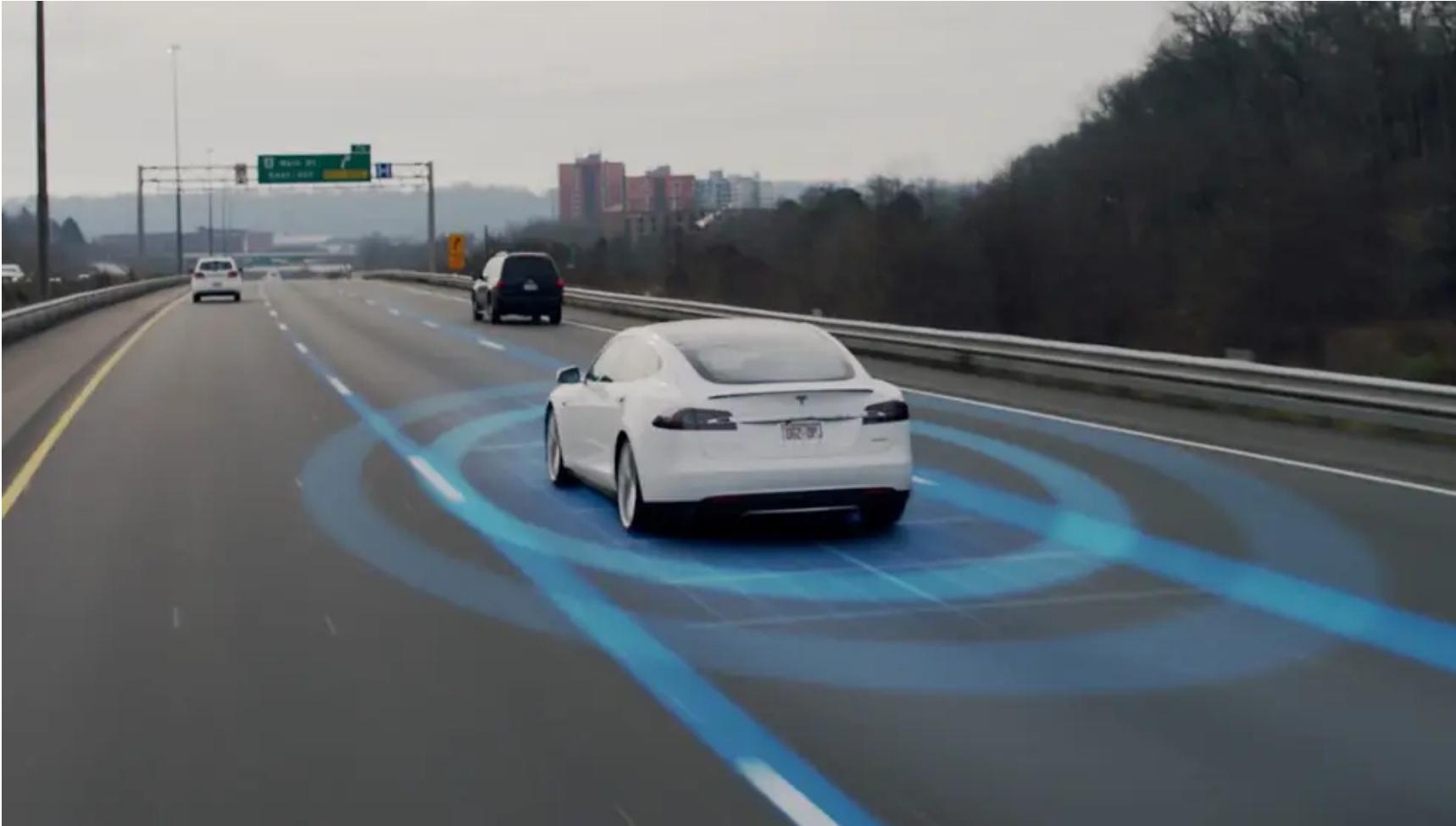
- (1) Obey the law



Autopilot

Objective

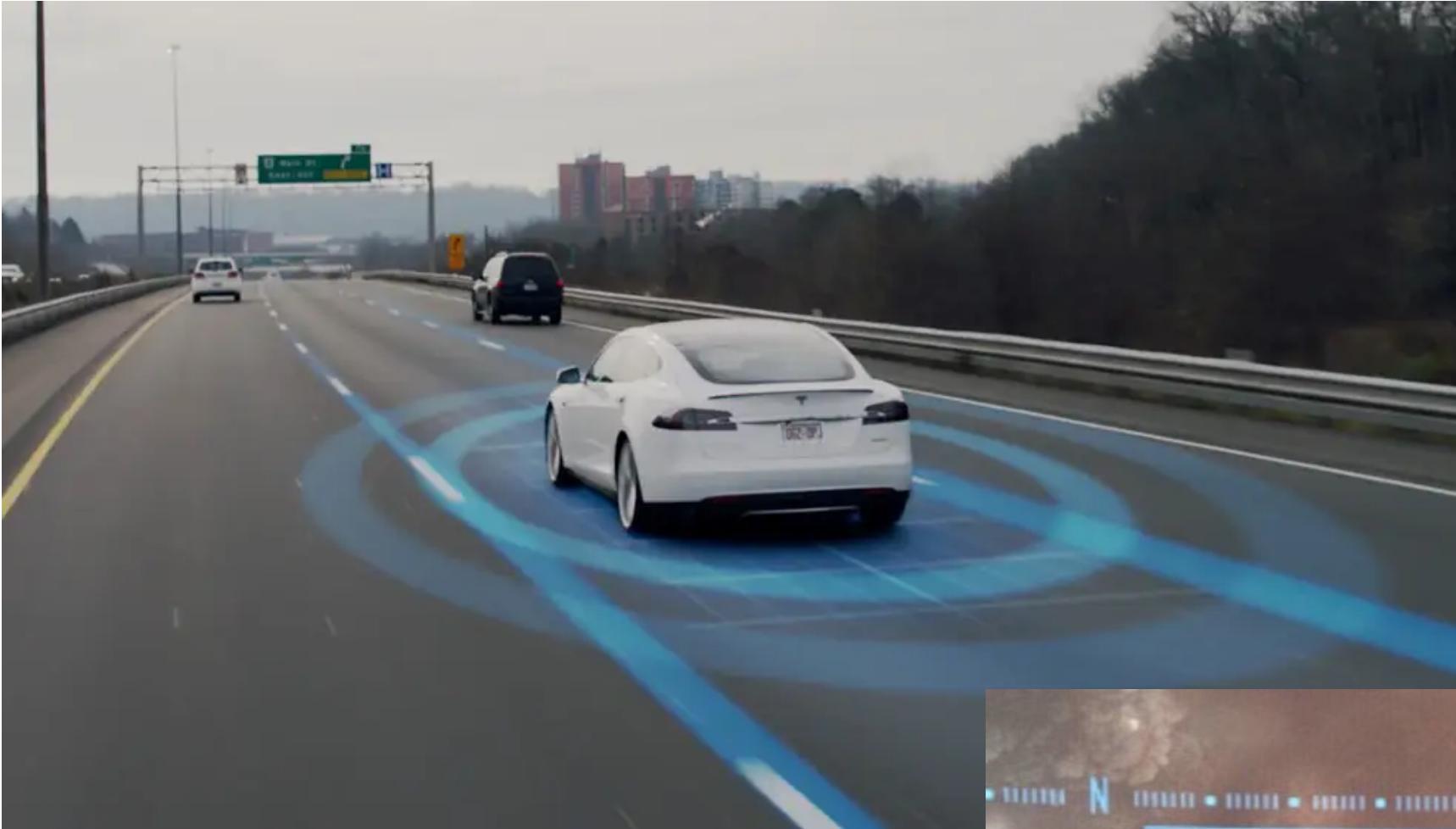
- (1) Obey the law
- (2) Don't crash



Autopilot

Objective

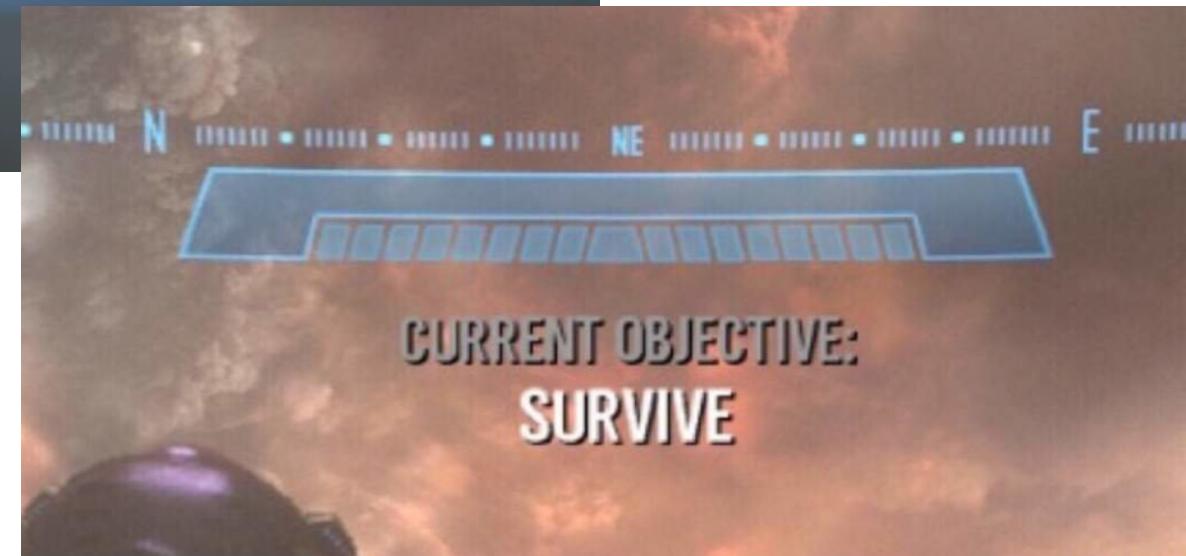
- (1) Obey the law
- (2) Don't crash
- (3) Follow the GPS



Autopilot

Objective

- (1) Obey the law
- (2) Don't crash
- (3) Follow the GPS



# How to optimize an objective?

A machine learning model has “parameters”.

Pick parameters such that the model approximately matches the data!

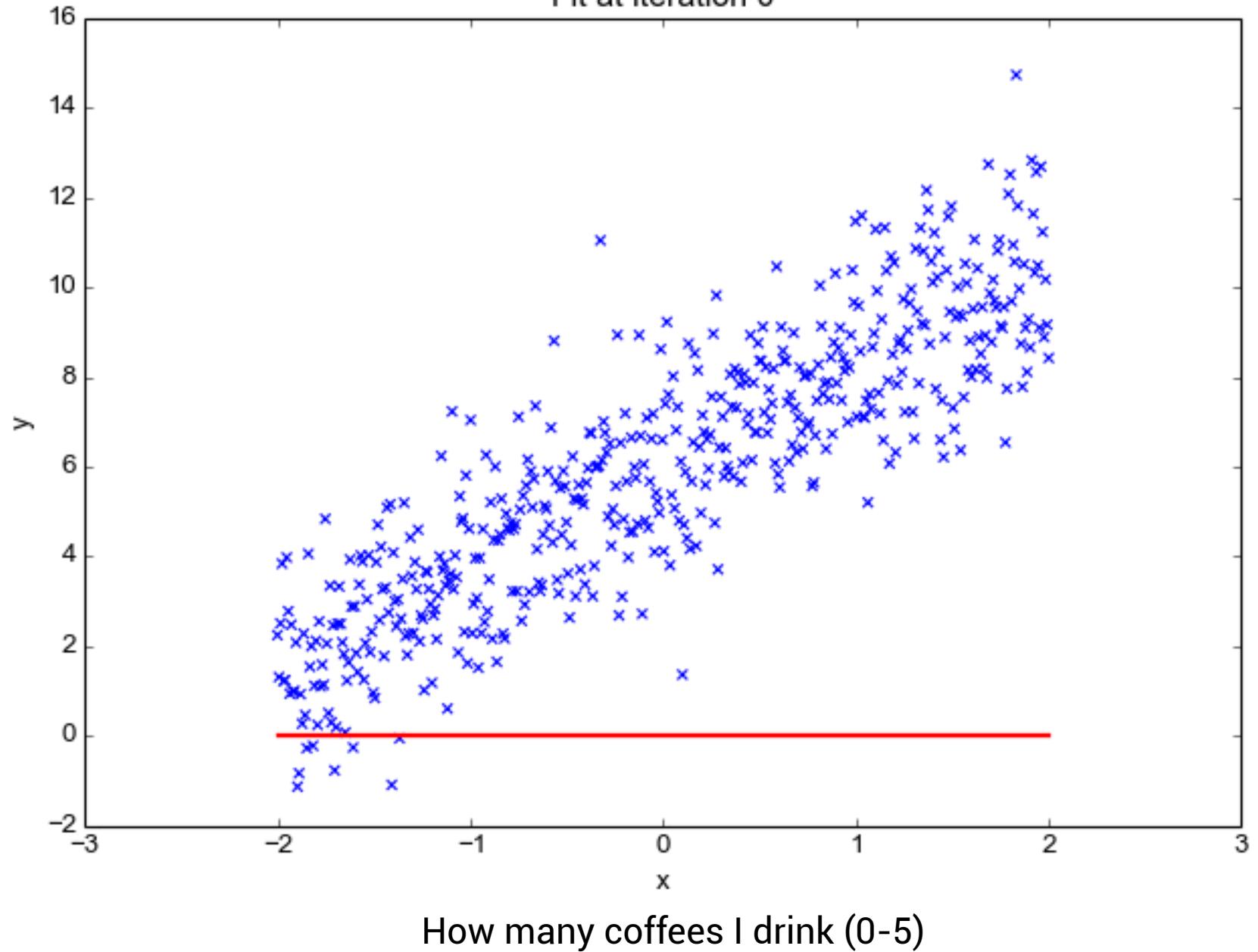
This process is called “training the model”

$$y = a * x + b$$

Find  $a, b$  such that:  
 $(a*x + b - y)$  is close to zero!

My heart rate

Fit at iteration 0

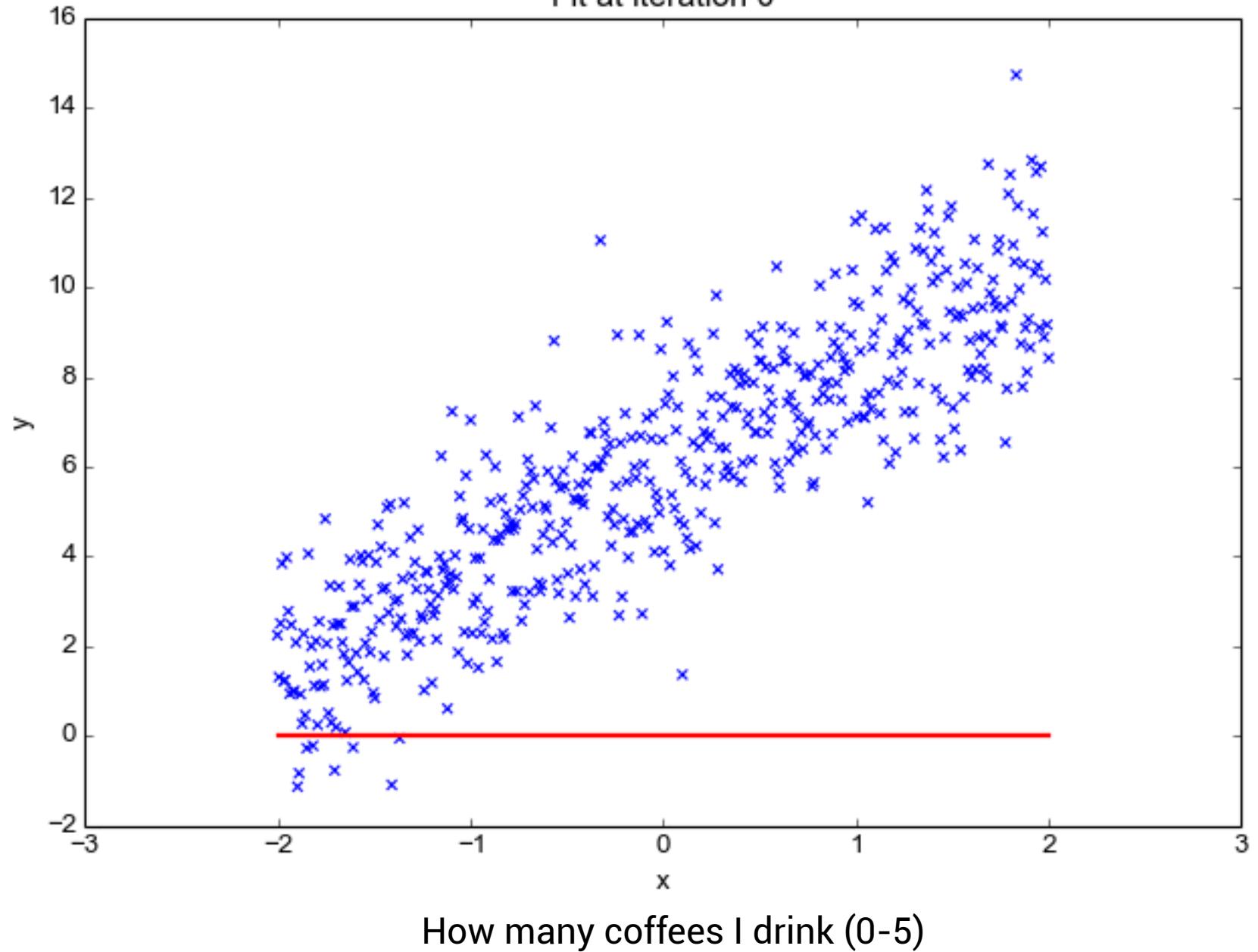


$$y = a * x + b$$

Find  $a, b$  such that:  
 $(a*x + b - y)$  is close to zero!

My heart rate

Fit at iteration 0



# Machine Learning: breakdown

Three parts of machine learning:

1. “Objective” - What am I trying to optimize?
2. Data - What examples do I have?
3. Model

omg didnt know Google Translate  
does Australian

The image shows a comparison between English and Australian English in the Google Translate app. On the left, under 'English - detected', the text 'Ladies and gentlemen, thank you for coming this evening.' is displayed with a small edit link. On the right, under 'Australian', the translation is shown as 'Sheilas and gentlejoeys, boomerang boomerang shrimp Hugh Jackman'. The interface includes language selection dropdowns at the top, a bidirectional arrow between the columns, and a speaker icon in the top right corner.

English - detected	Australian
Ladies and gentlemen, thank you for coming this evening. <small>Edit</small>	Sheilas and gentlejoeys, boomerang boomerang shrimp Hugh Jackman

# Google Translate

omg didnt know Google Translate  
does Australian

The image shows a screenshot of the Google Translate web interface. At the top, it displays two language dropdown menus: "English - detected" on the left and "Australian" on the right. Between them is a bidirectional arrow icon. Below the languages, there are two text boxes. The left text box contains the sentence "Ladies and gentlemen, thank you for coming this evening." followed by a small "Edit" link. The right text box contains the translation "Sheilas and gentlejoeys, boomerang boomerang shrimp Hugh Jackman". To the right of the second text box is a speaker icon indicating that the translated text can be heard.

English - detected

Australian

Ladies and gentlemen, thank you for coming this evening. [Edit](#)

Sheilas and gentlejoeys,  
boomerang boomerang shrimp  
Hugh Jackman

# Google Translate

omg didnt know Google Translate  
does Australian

The image shows a screenshot of the Google Translate web interface. At the top, it displays two language dropdown menus: "English - detected" on the left and "Australian" on the right. Between them is a bidirectional arrow icon. Below the languages, there are two text boxes. The left text box contains the sentence "Ladies and gentlemen, thank you for coming this evening." followed by a small "Edit" link. The right text box contains the translation "Sheilas and gentlejoeys, boomerang boomerang shrimp Hugh Jackman". To the right of the second text box is a speaker icon indicating that the translated text can be heard.

English - detected

Australian

Ladies and gentlemen, thank you for coming this evening. [Edit](#)

Sheilas and gentlejoeys,  
boomerang boomerang shrimp  
Hugh Jackman

# Google Translate

Data:

omg didnt know Google Translate  
does Australian

The image shows a screenshot of the Google Translate mobile application. At the top, it displays "English - detected" and "Australian". Below this, there is a text input field containing the sentence "Ladies and gentlemen, thank you for coming this evening." followed by an "Edit" link. To the right of the input field, the translated text is shown: "Sheilas and gentlejoeys, boomerang boomerang shrimp Hugh Jackman". A small speaker icon is located at the top right of the translation box.

English - detected

Australian

Ladies and gentlemen, thank you for coming this evening. [Edit](#)

Sheilas and gentlejoeys,  
boomerang boomerang shrimp  
Hugh Jackman

# Google Translate

Data:

Manual translations of  
books, movies, etc

omg didnt know Google Translate  
does Australian

The image shows a screenshot of the Google Translate mobile application. At the top, it displays "English - detected" and "Australian". Below this, there is a translation pair. The English input is "Ladies and gentlemen, thank you for coming this evening." and the Australian output is "Sheilas and gentlejoeys, boomerang boomerang shrimp Hugh Jackman". A small edit link is visible next to the English text.

English - detected	Australian
Ladies and gentlemen, thank you for coming this evening.	Sheilas and gentlejoeys, boomerang boomerang shrimp Hugh Jackman

When you order an uber and it's  
one of those self driving cars.



# Self-driving cars

When you order an uber and it's one of those self driving cars.



# Self-driving cars

When you order an uber and it's one of those self driving cars.



# Self-driving cars

Data:

When you order an uber and it's one of those self driving cars.



# Self-driving cars

Data:

Recordings of someone driving: position of wheel, gas pedal, and video of surroundings

When you order an uber and it's one of those self driving cars.



# **Machine Learning: breakdown**

# **Machine Learning: breakdown**

Three parts of machine learning:

# Machine Learning: breakdown

Three parts of machine learning:

1. “Objective” - What am I trying to optimize?

# Machine Learning: breakdown

Three parts of machine learning:

1. “Objective” - What am I trying to optimize?
2. Data - What examples do I have?

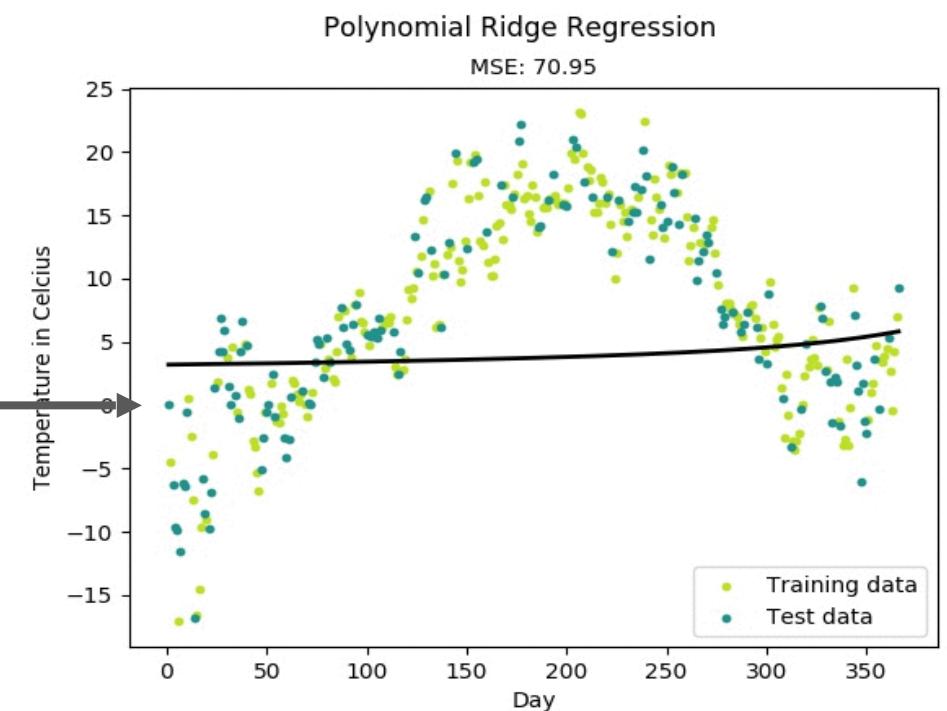
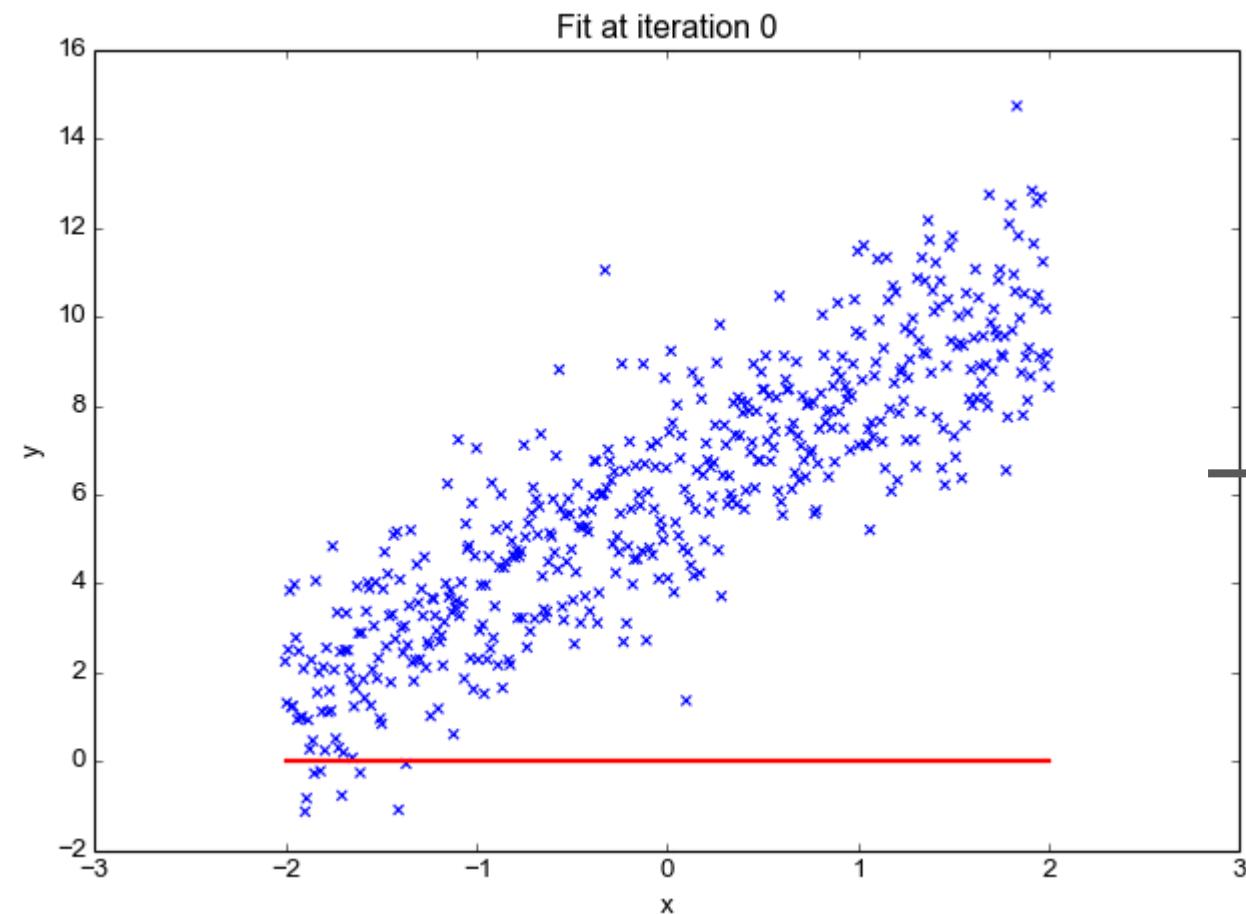
# Machine Learning: breakdown

Three parts of machine learning:

1. “Objective” - What am I trying to optimize?
2. Data - What examples do I have?
3. **Model - What algorithm will I use?**

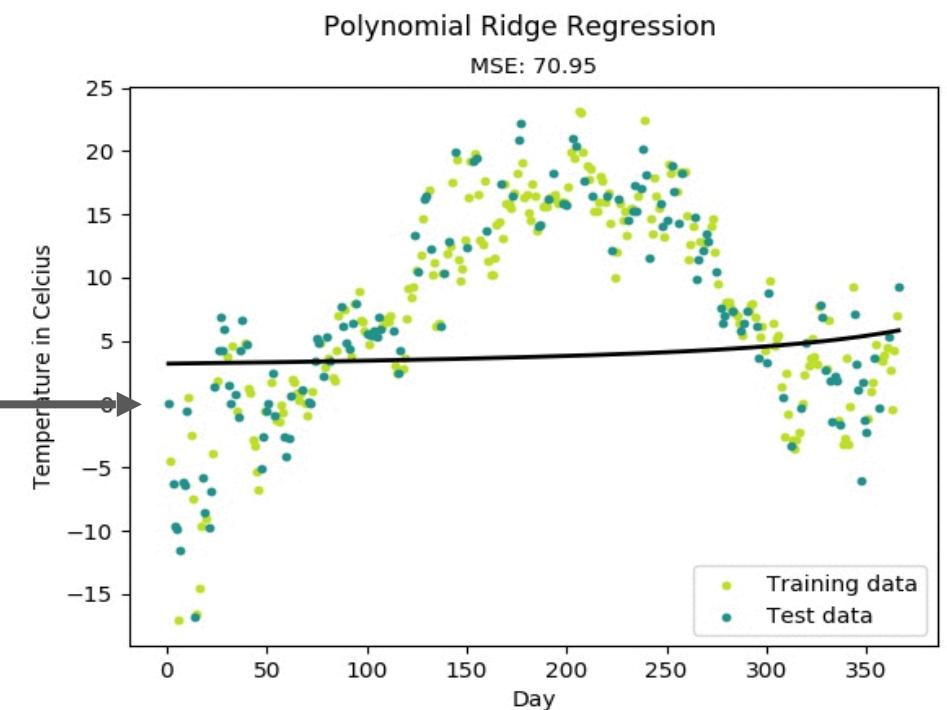
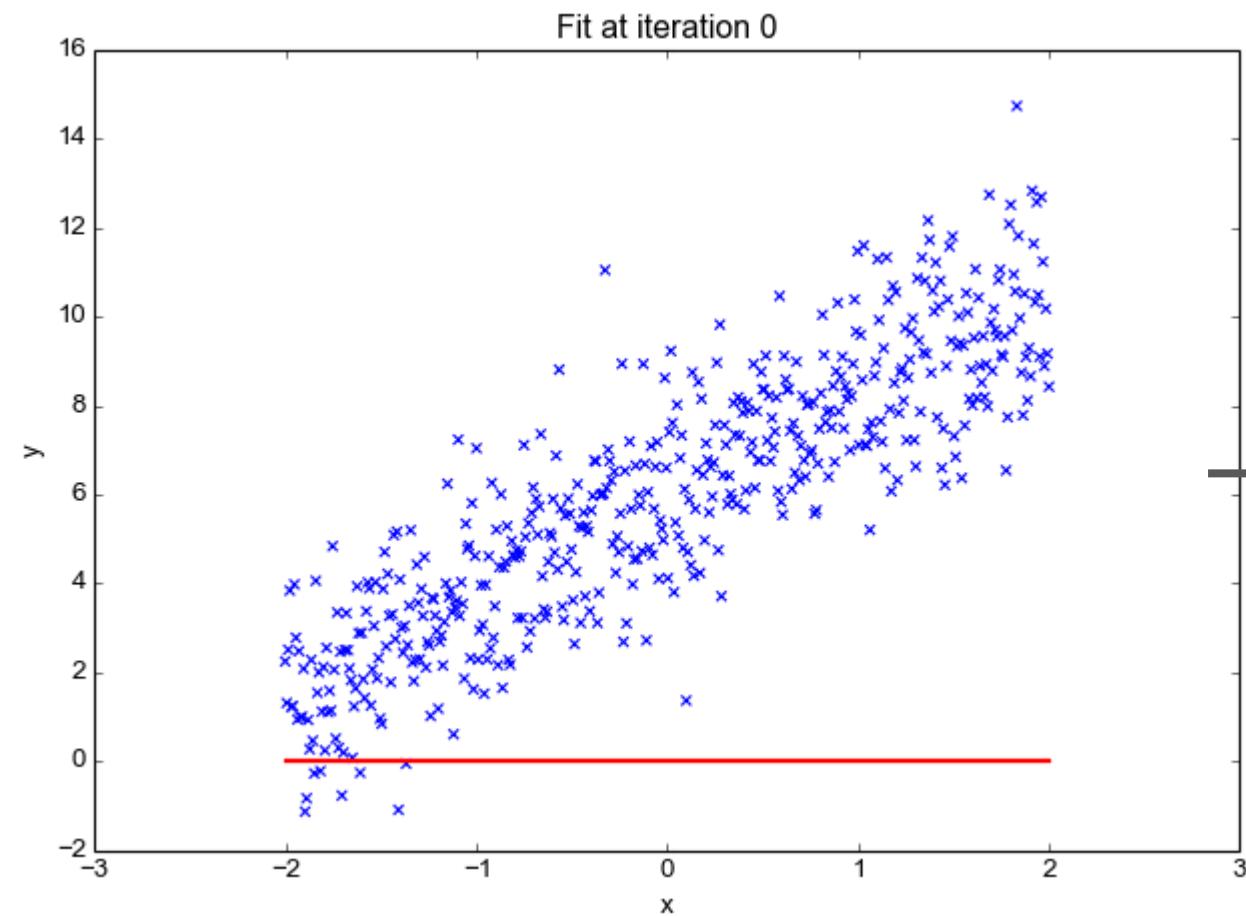
# **Deep Learning (modern AI)**

# What is deep learning?



More complex models!  
Think of it as many small lines that we fit to data.

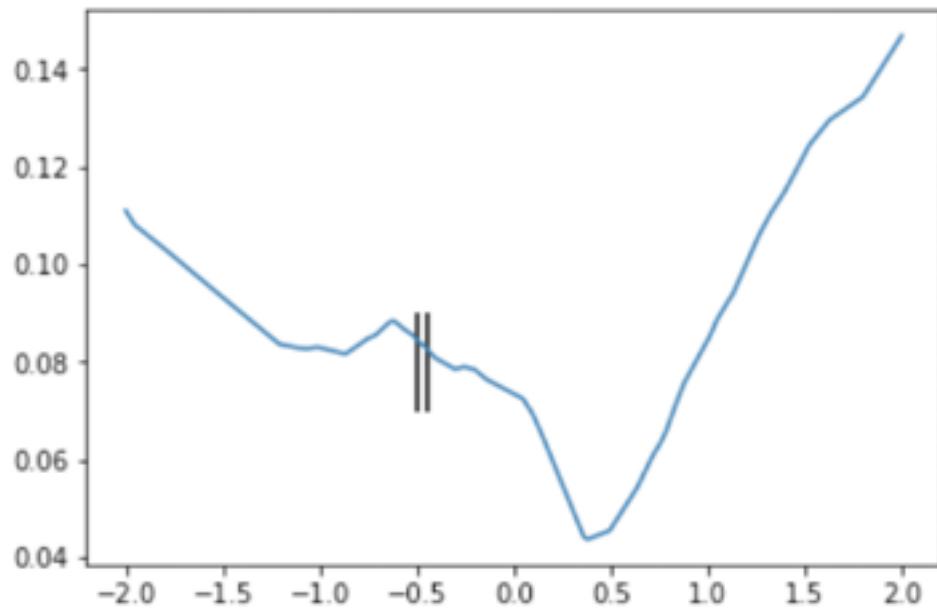
# What is deep learning?



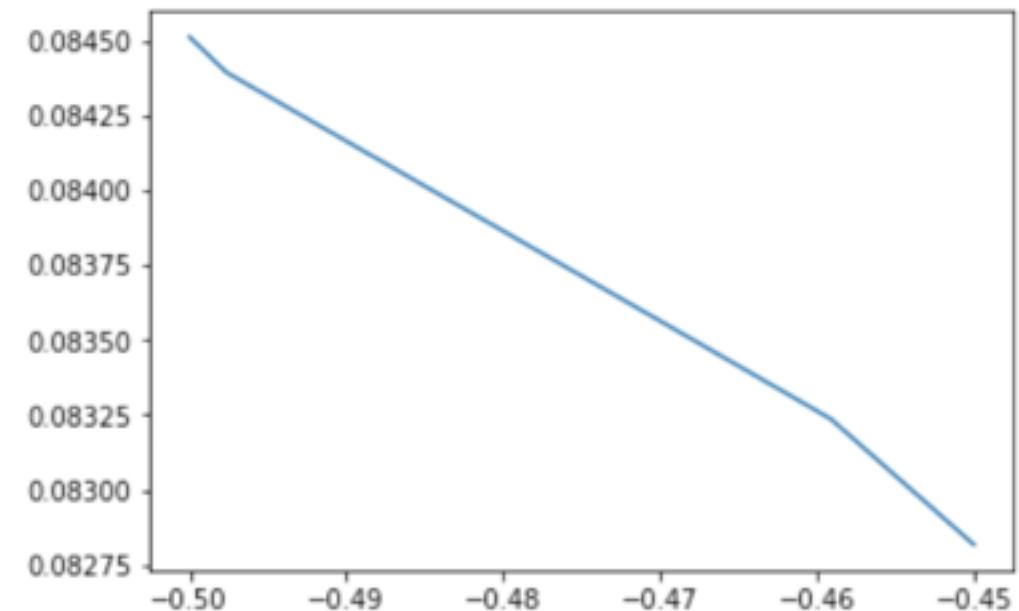
More complex models!  
Think of it as many small lines that we fit to data.

# Neural Network

Untrained neural network:

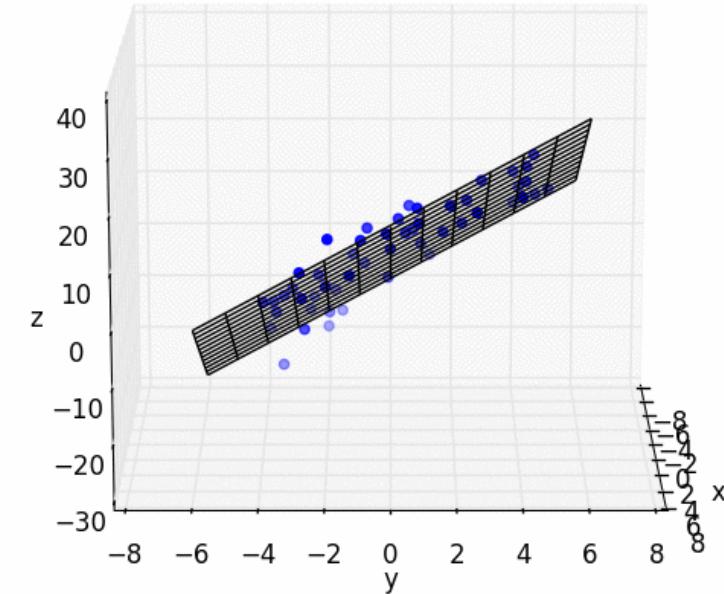
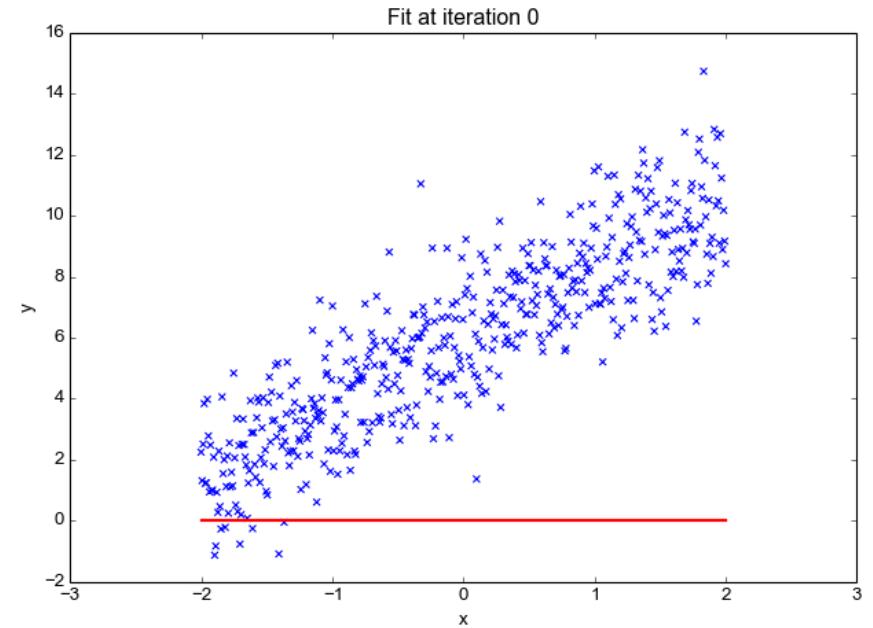


→  
Zoom in:



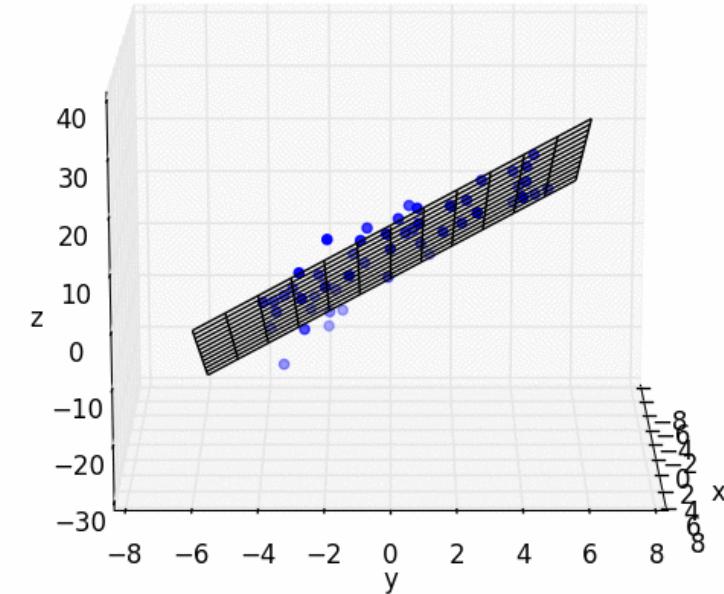
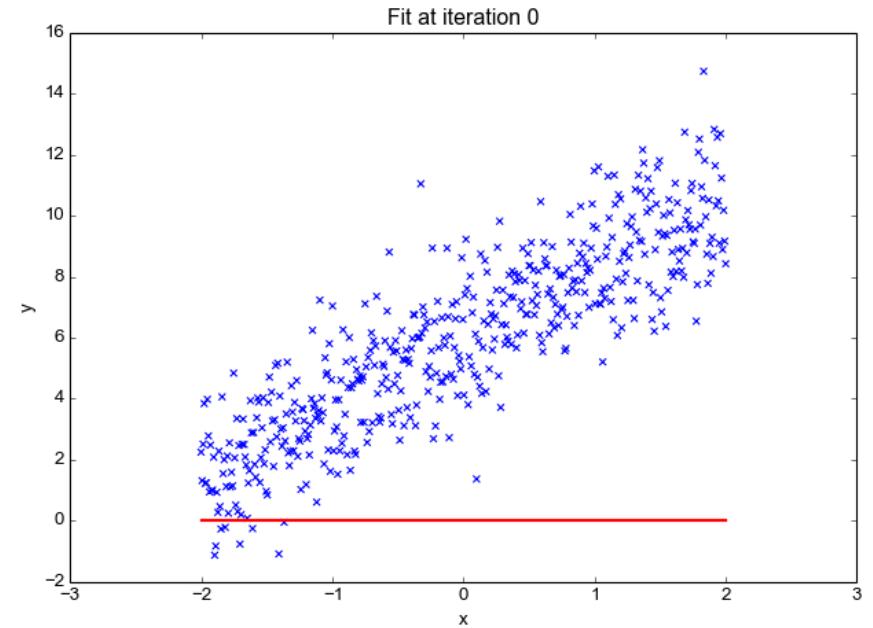
# Computer Vision

Imagine this line going through the page and with a slope: you could fit 2D data!



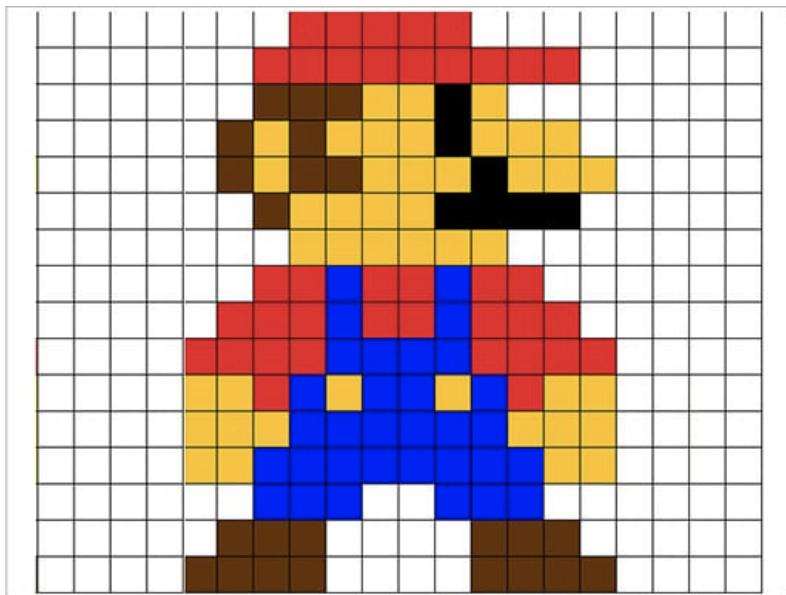
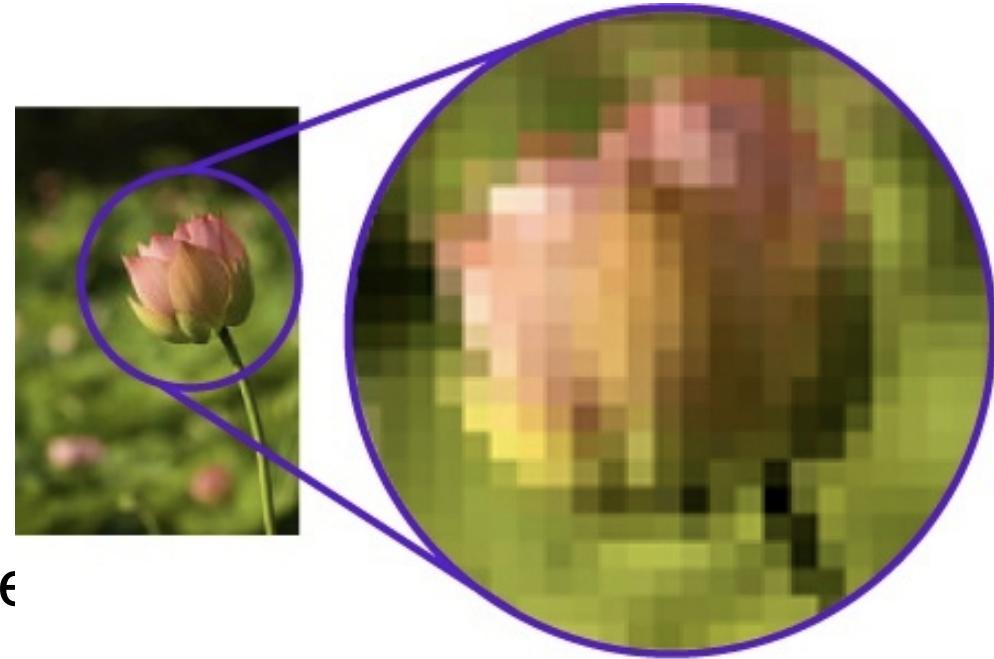
# Computer Vision

Imagine this line going through the page and with a slope: you could fit 2D data!

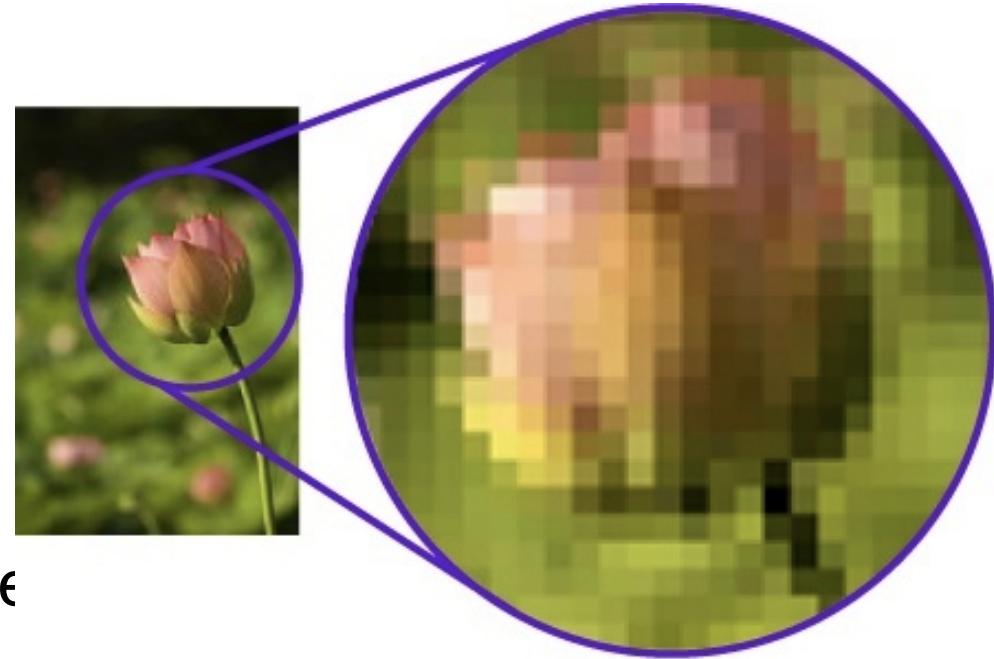


# Computer Vision

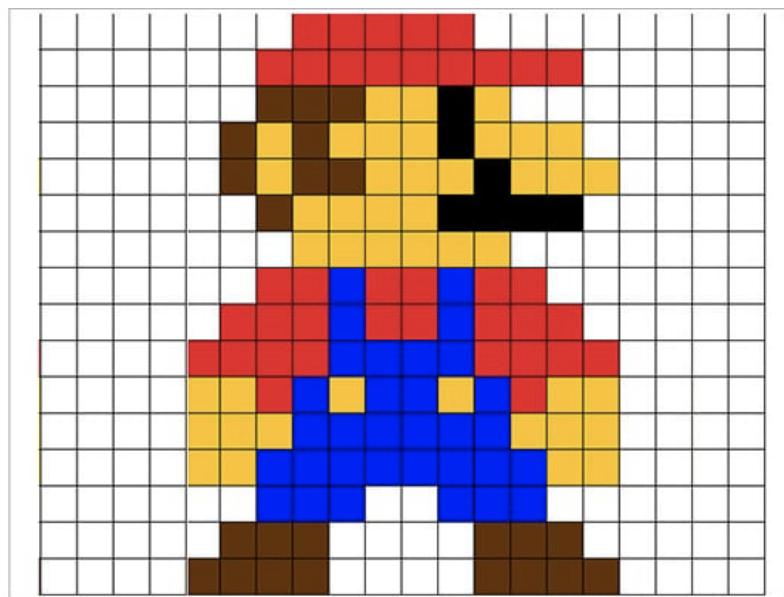
Images: points in many-dimensional space



# Computer Vision



Images: points in many-dimensional space

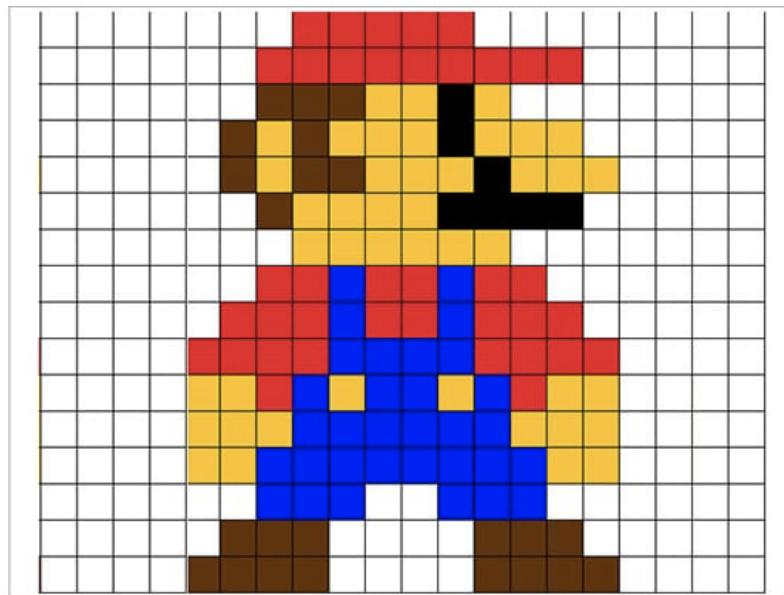
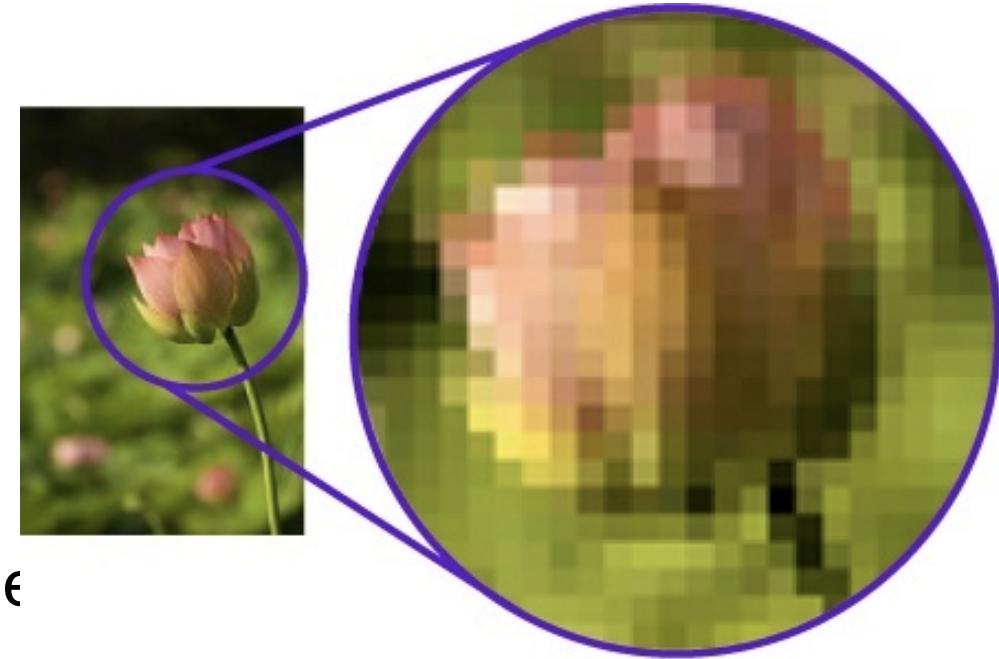


e.g., white=0, red=1, brown=2,

...

# Computer Vision

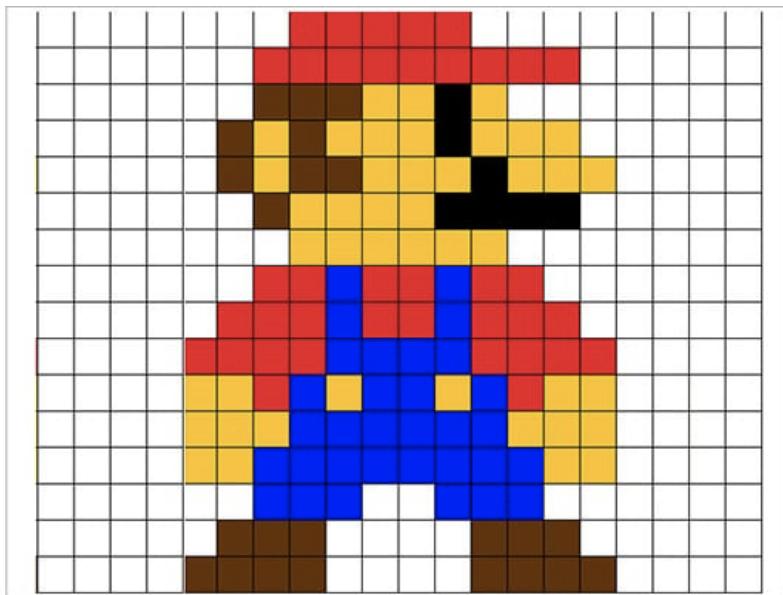
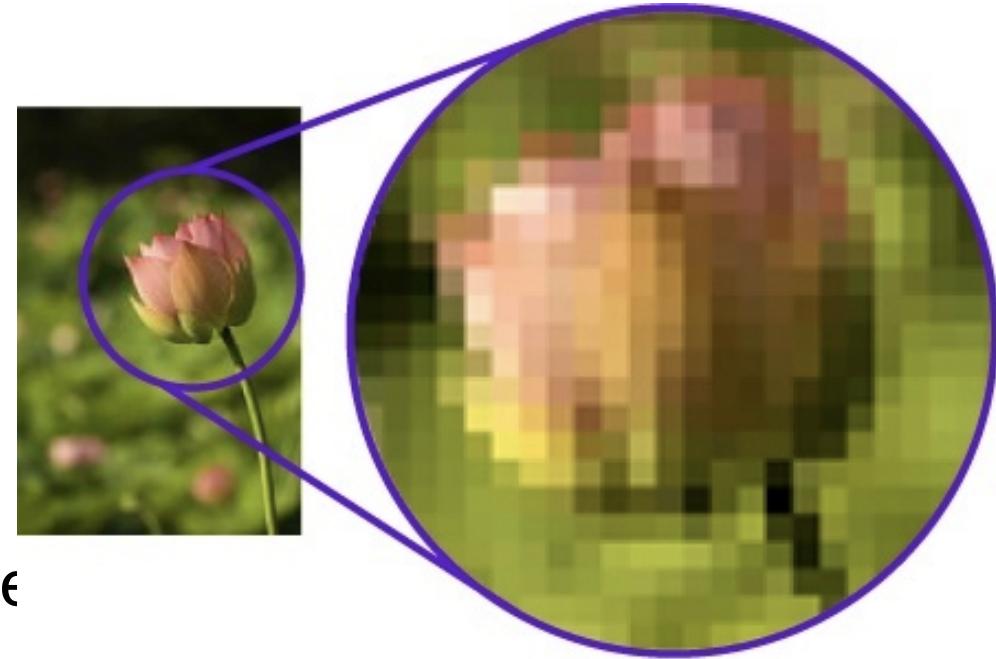
Images: points in many-dimensional space



e.g., white=0, red=1, brown=2,  
...  
 $(0, 0, 0, \dots, 1, 1, \dots, 0, 0, \dots, 2, \dots)$

# Computer Vision

Images: points in many-dimensional space



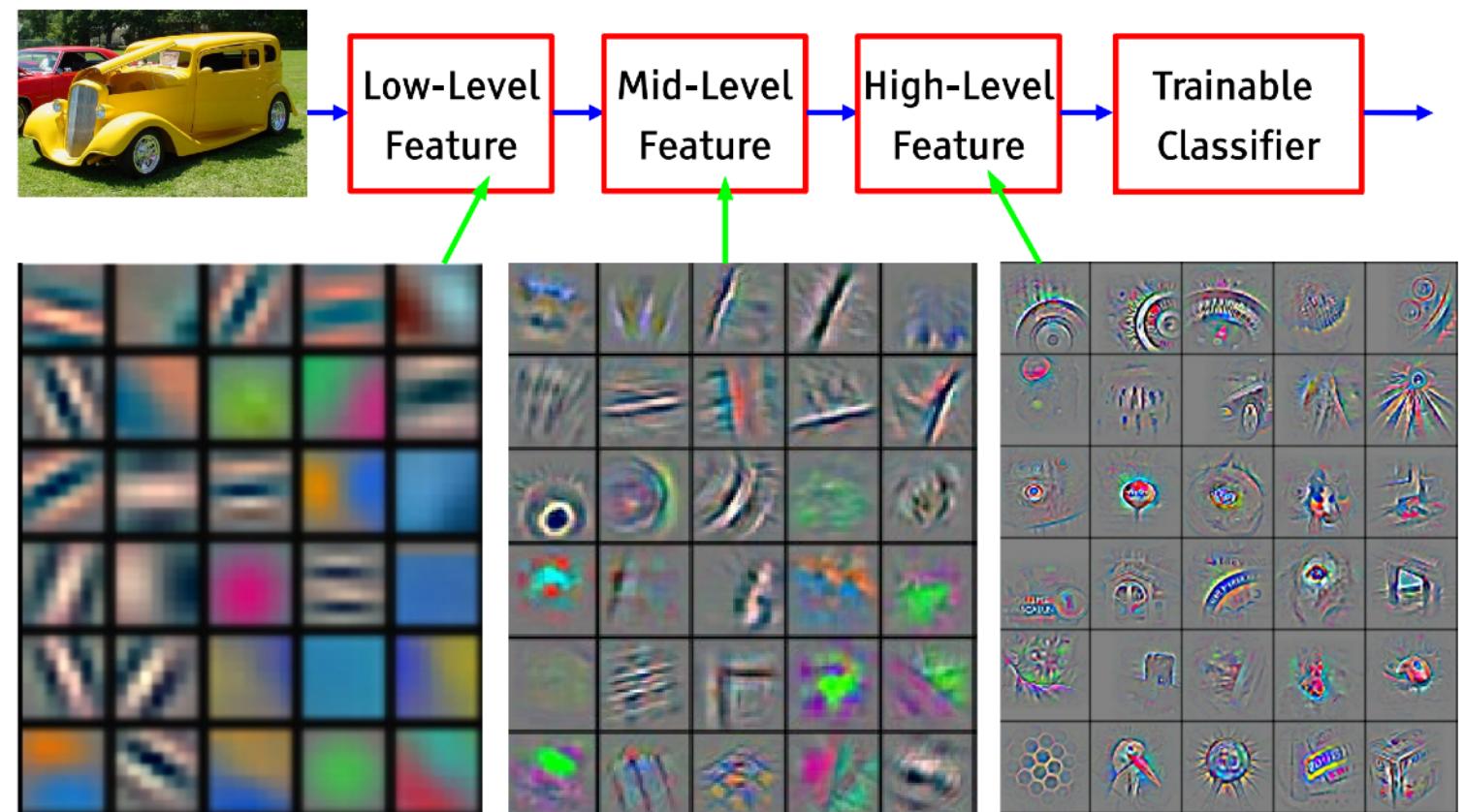
e.g., white=0, red=1, brown=2,

...

(0, 0, 0, ..., 1, 1, ..., 0, 0, ..., 2, ...)

As many dimensions as pixels!

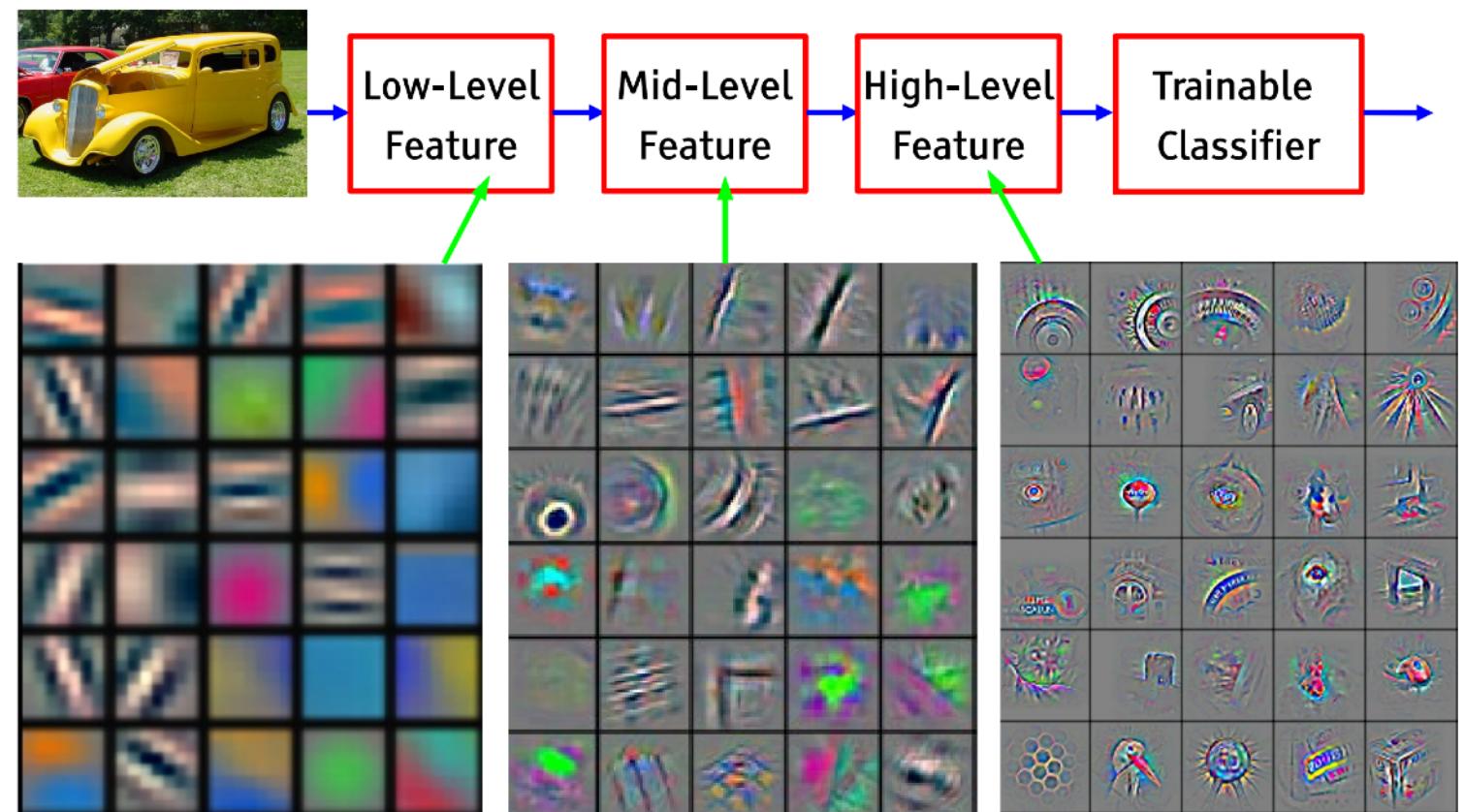
# “Convolutional Neural Network”



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

# “Convolutional Neural Network”

Problem: this is crazy-high number of dimensions!!

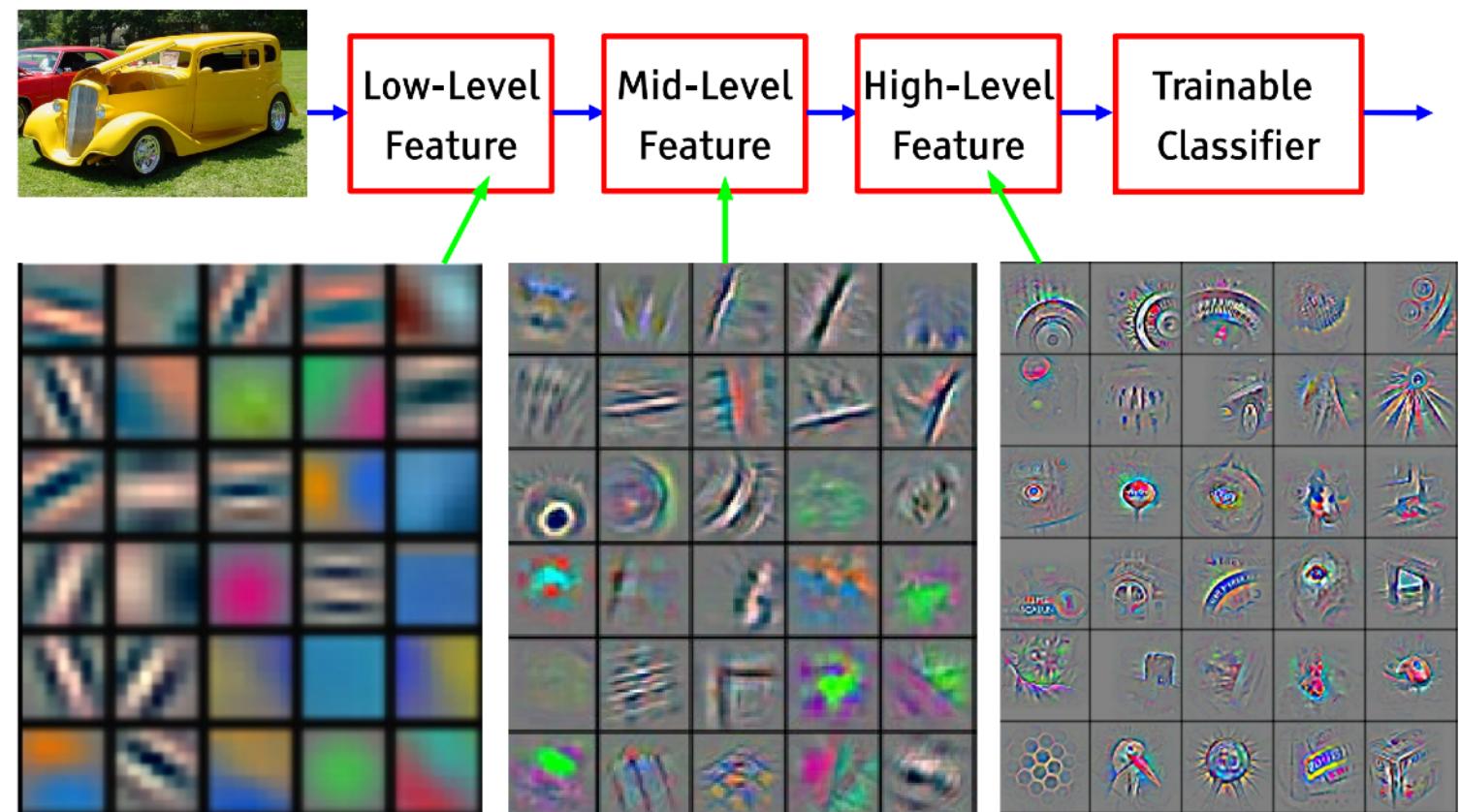


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

# “Convolutional Neural Network”

Problem: this is crazy-high number of dimensions!!

Solution: count the numbers of each “feature”, and use that as the data!

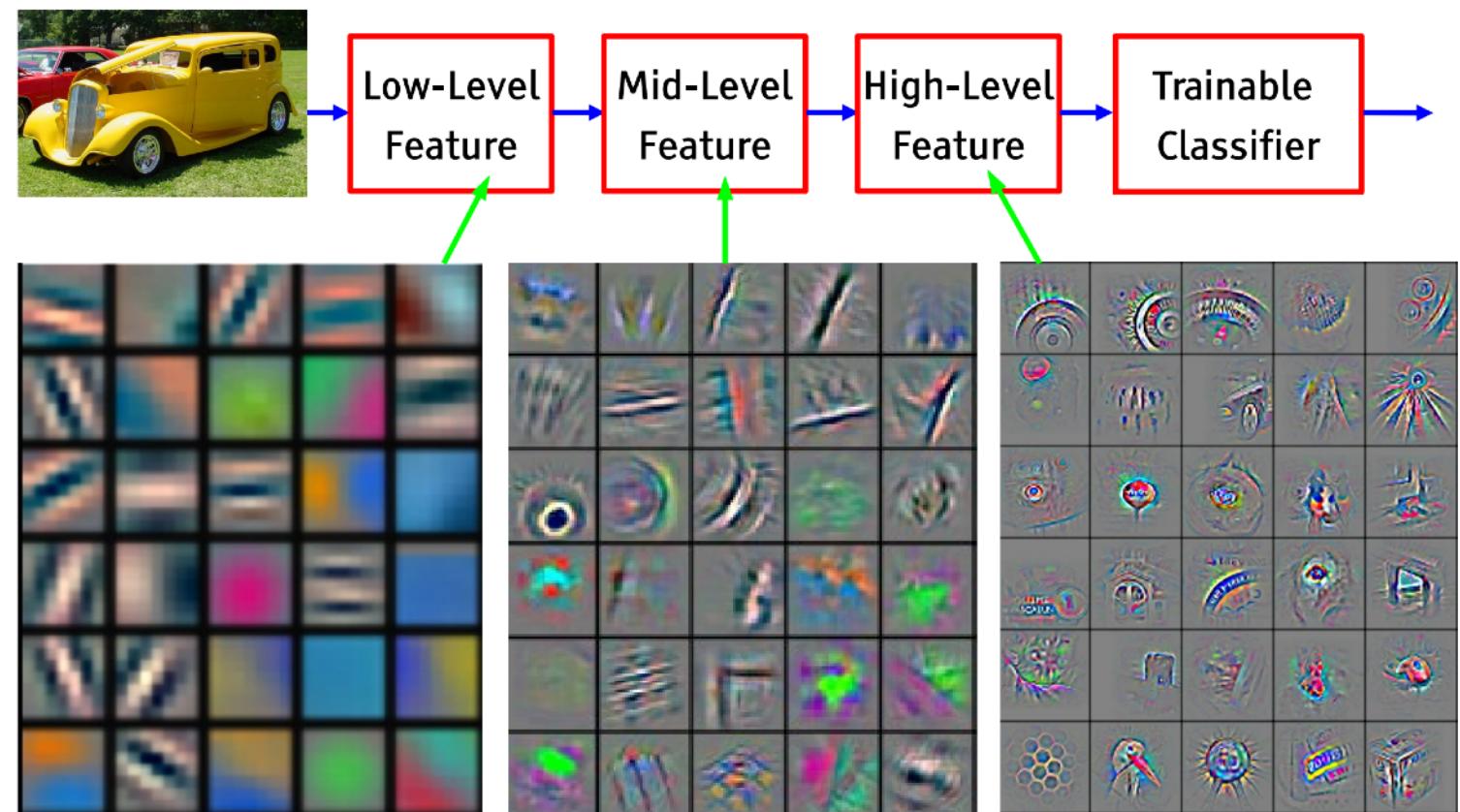


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

# “Convolutional Neural Network”

Problem: this is crazy-high number of dimensions!!

Solution: count the numbers of each “feature”, and use that as the data!



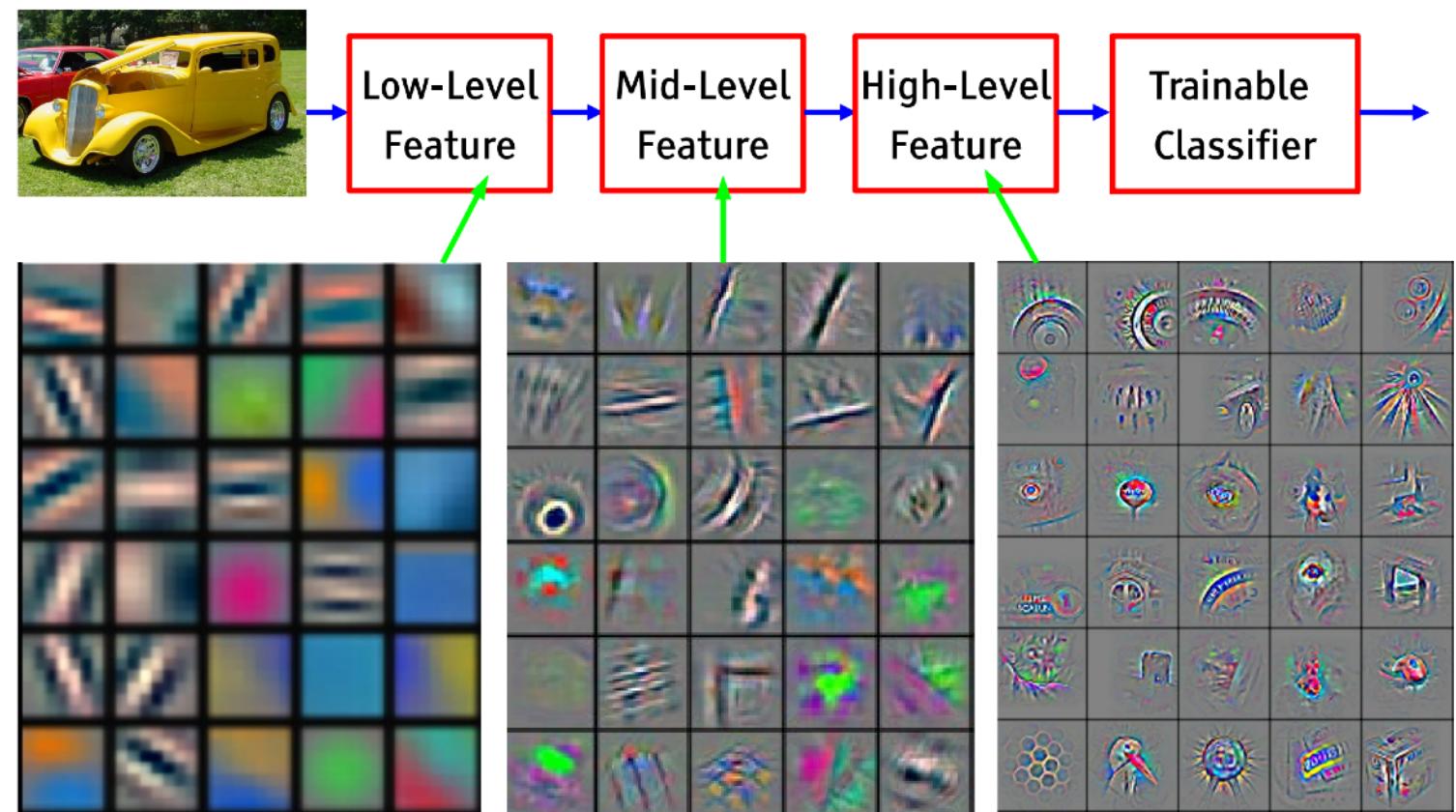
Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

# “Convolutional Neural Network”

Problem: this is crazy-high number of dimensions!!

Solution: count the numbers of each “feature”, and use that as the data!

e.g., how many times does this occur in the image?

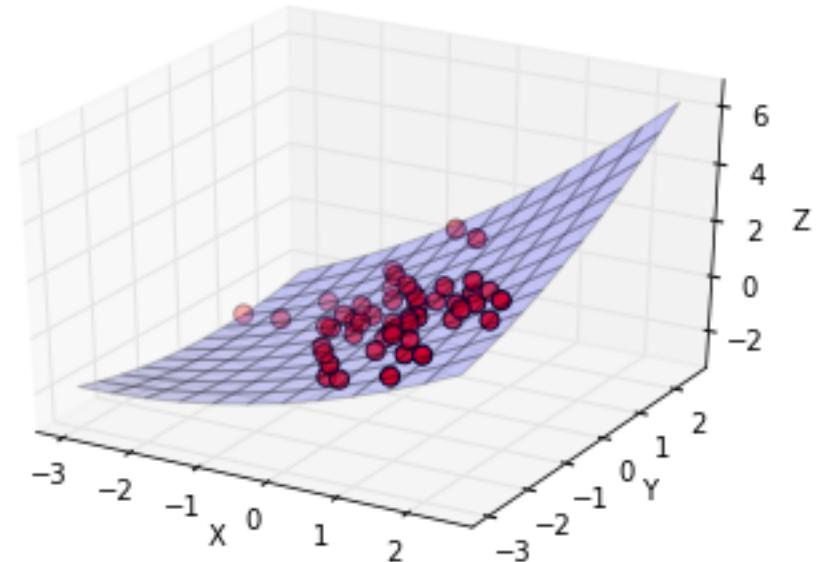


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

## "Convolutional Neural Network"

Left with only ~10 dimensions, which much easier to fit a surface to!

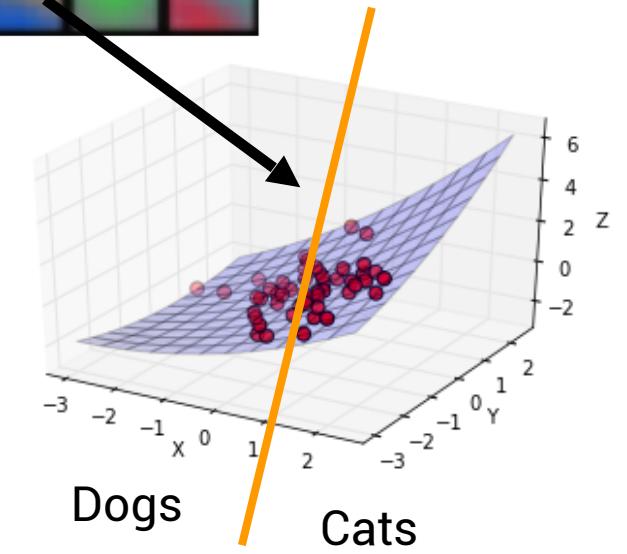
Same for every task with large "dimension"



# Deep Learning



Count the number  
of each “feature”

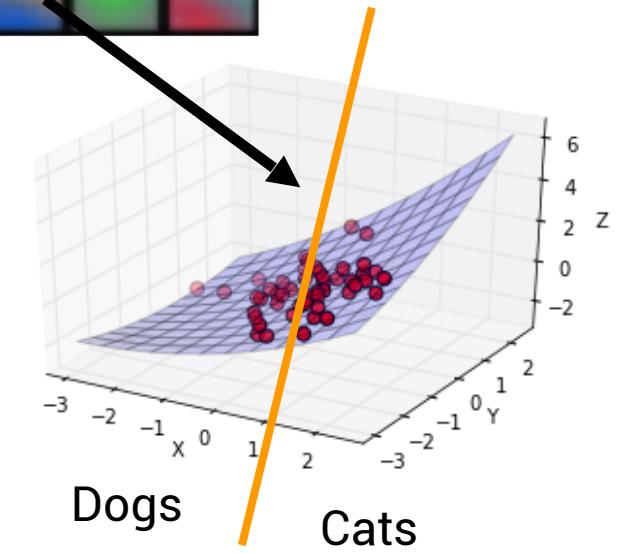
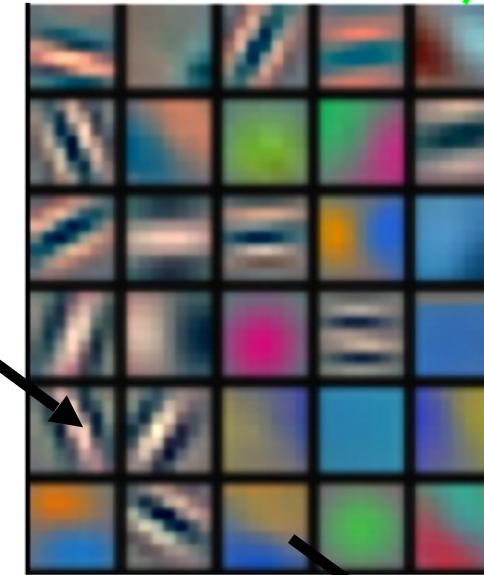


# Deep Learning

All of deep learning is like this!

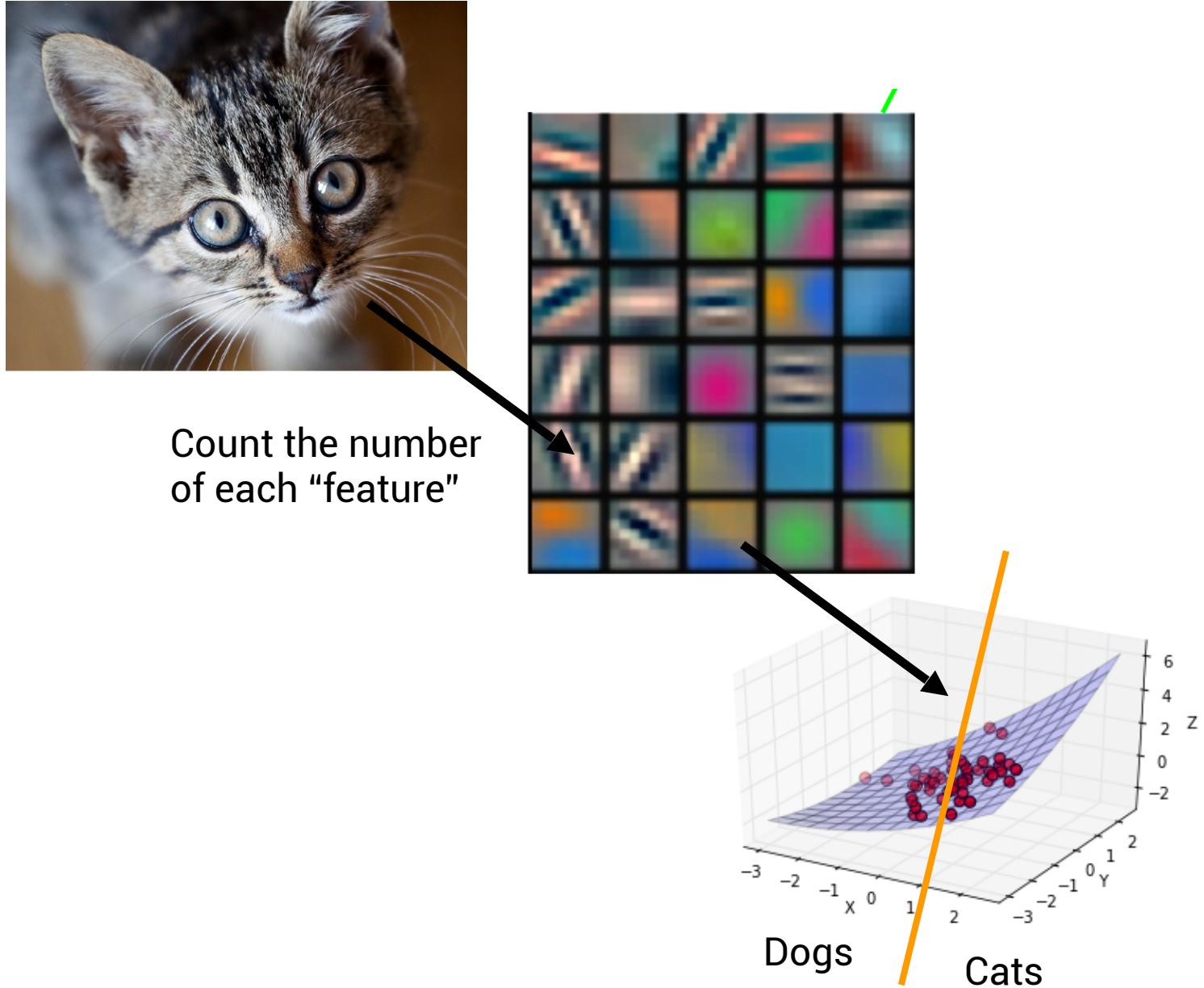


Count the number  
of each “feature”



# Deep Learning

All of deep learning is like this!



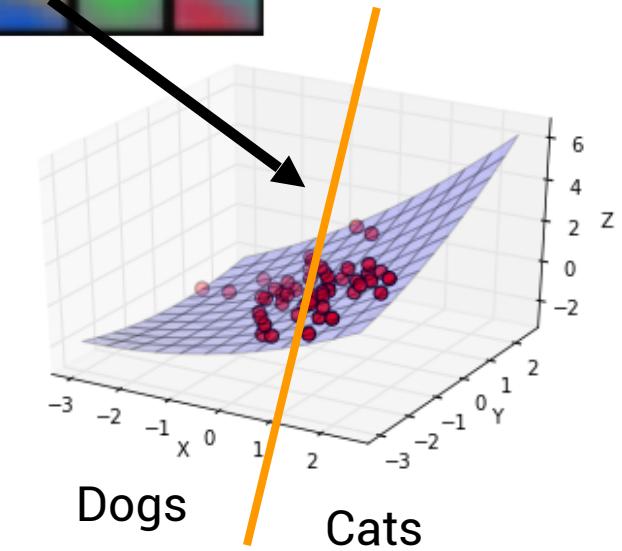
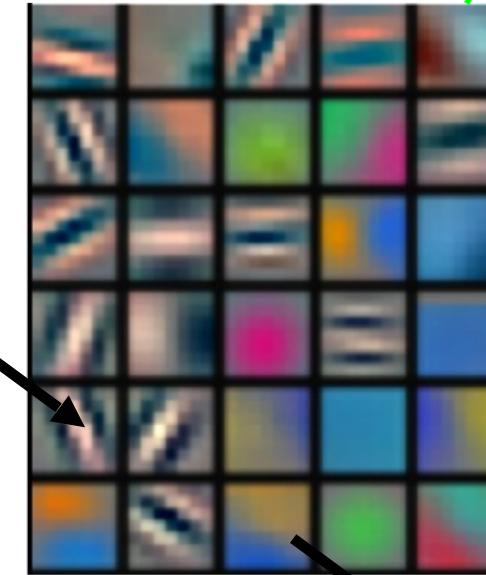
# Deep Learning

All of deep learning is like this!

1. Take your data: image, text, audio.



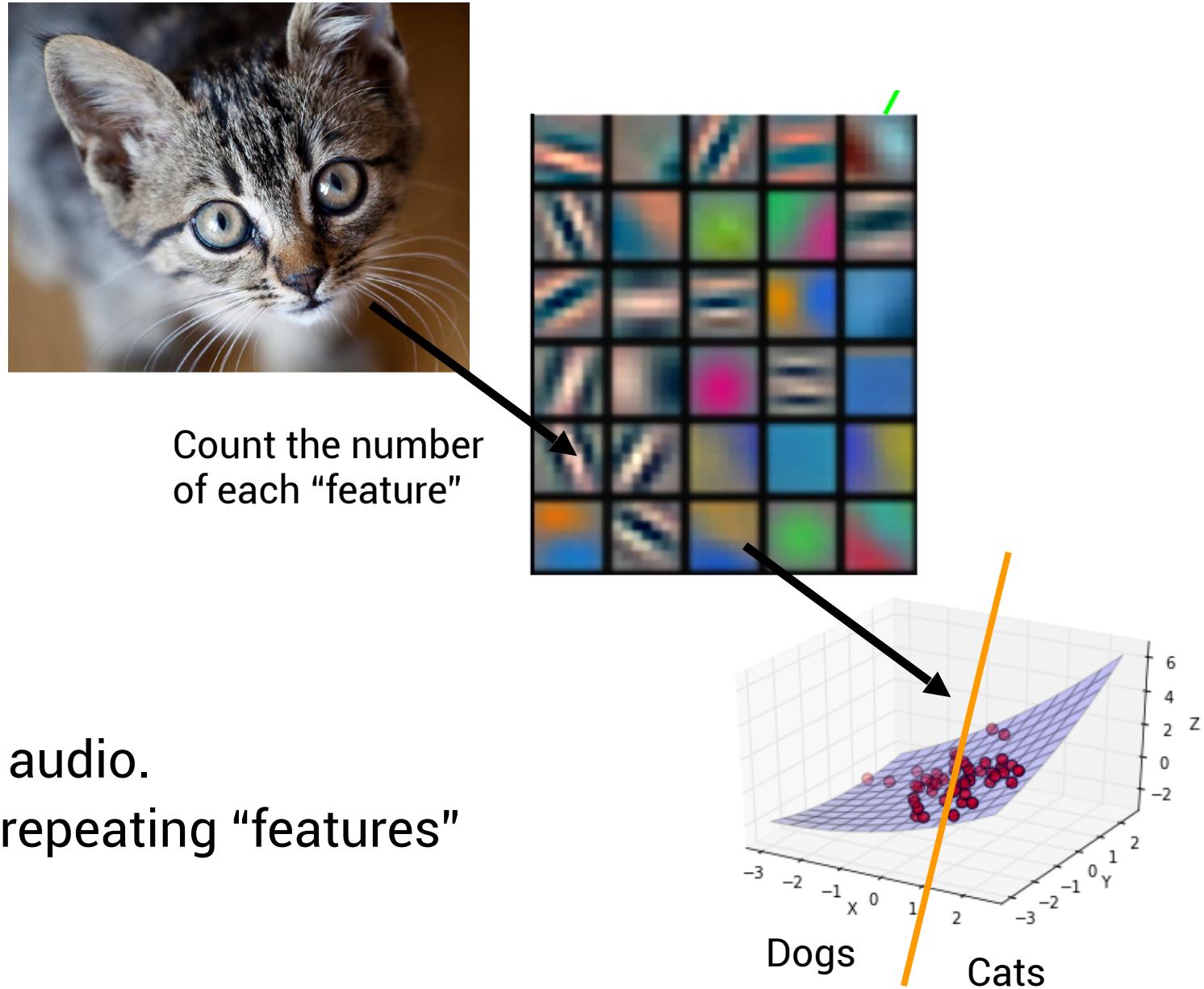
Count the number  
of each “feature”



# Deep Learning

All of deep learning is like this!

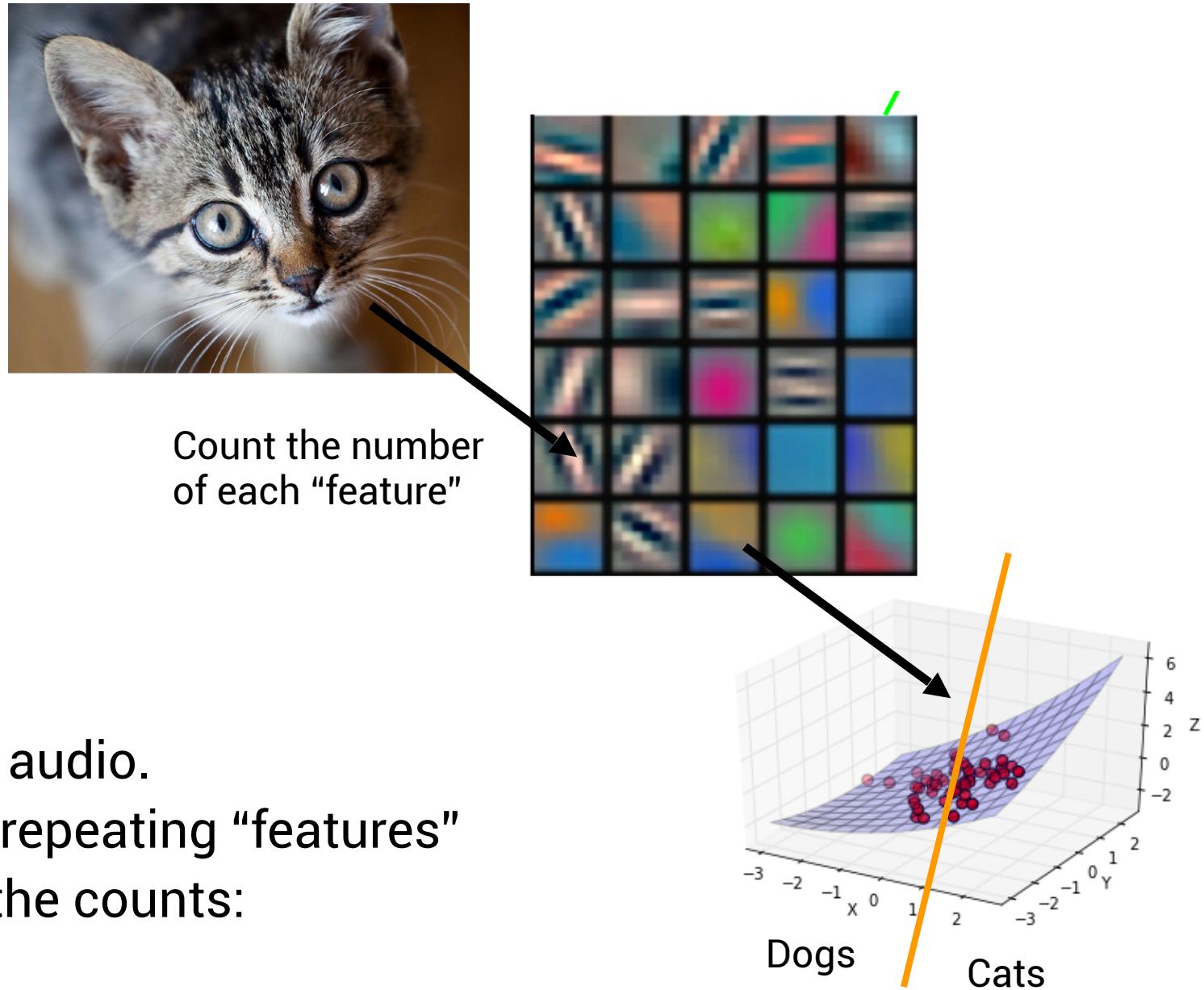
1. Take your data: image, text, audio.
2. Count numbers of different repeating “features”



# Deep Learning

All of deep learning is like this!

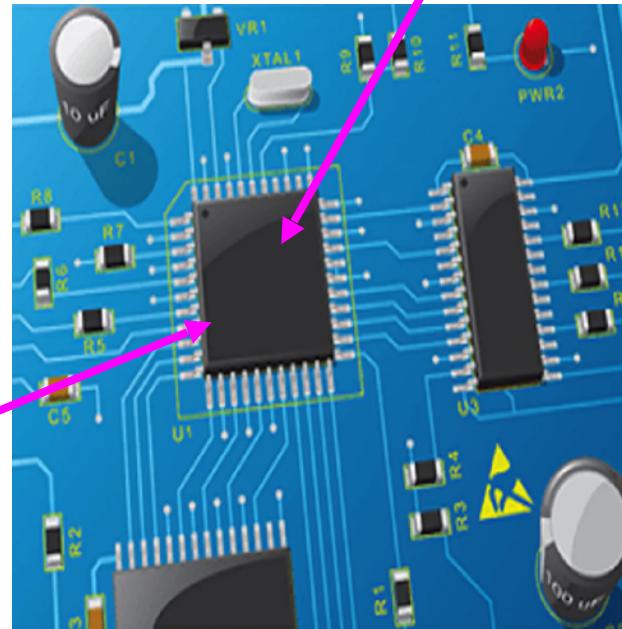
1. Take your data: image, text, audio.
2. Count numbers of different repeating “features”
3. Fit a surface to the data to the counts:



# We can now do the task we couldn't before!

This part adds numbers  
^use to count number of features

This part multiplies numbers < use to compute  $y=a*x+b$



# **Other types of data?**

# **Other types of data?**

Image: count occurrences of each features in pixels

# **Other types of data?**

Image: count occurrences of each features in pixels

Audio: each microsecond, record volume => like a pixel!

# **Other types of data?**

Image: count occurrences of each features in pixels

Audio: each microsecond, record volume => like a pixel!

Text: could represent each word as 3 pixels in an image

# Case Study #1

! Walmart 🎁 <info@evlocalisation.nl>  
to miles.cranmer

Mon, Nov 16, 9:13 PM (8 days ago) ⭐ ⏪ ⋮

Why is this message in spam? It is similar to messages that were identified as spam in the past.

Report not spam

?

Spam detection:

Congratulations **Walmart Shopper**,

You have been selected to get an exclusive reward !

Objective?

To qualify for this special offer, simply complete our 30-second

Data?

marketing survey about your shopping experiences with walmart

Model?

Claim your Walmart Shoppers Gift Card  
now!

[Important for miles.cranmer >>](#)

# Case Study #2

Art style transfer:

Objective?

Data?

Model?



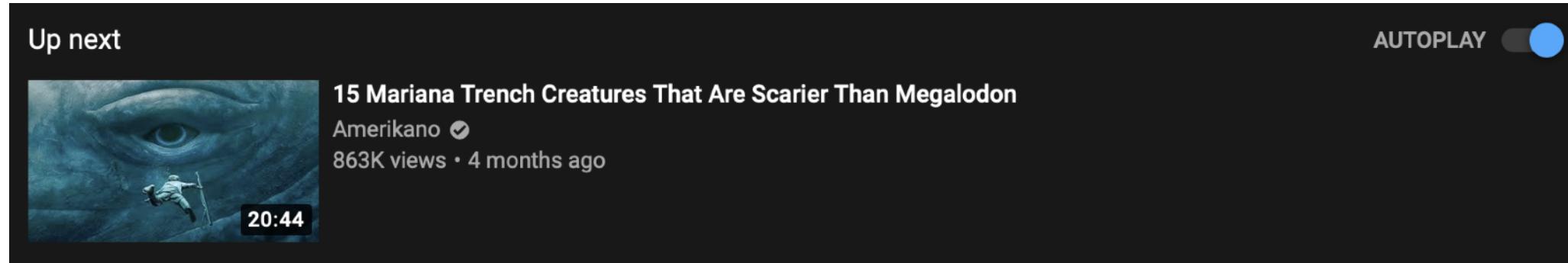
# Case Study #3

YouTube rabbit hole:

Objective?

Data?

Model?



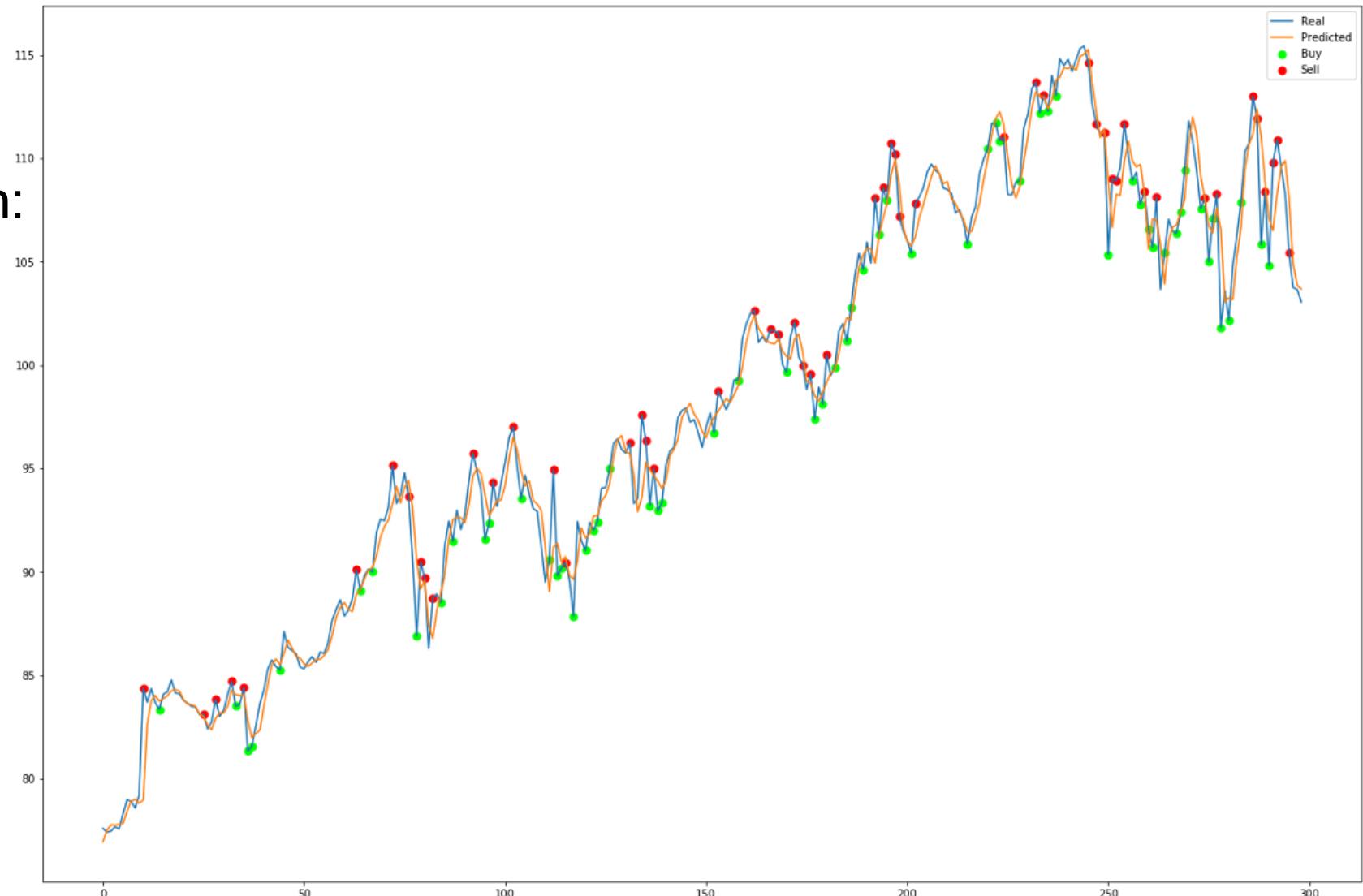
# Case Study #4

Stock market prediction:

Objective?

Data?

Model?



# Case Study #5

## SYSTEM PROMPT (HUMAN-WRITTEN)

*In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.*

Text generation:

## MODEL COMPLETION (MACHINE-WRITTEN, 10 TRIES)

The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Objective?

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Data?

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

Pérez and the others then ventured further into the valley. "By

Model?

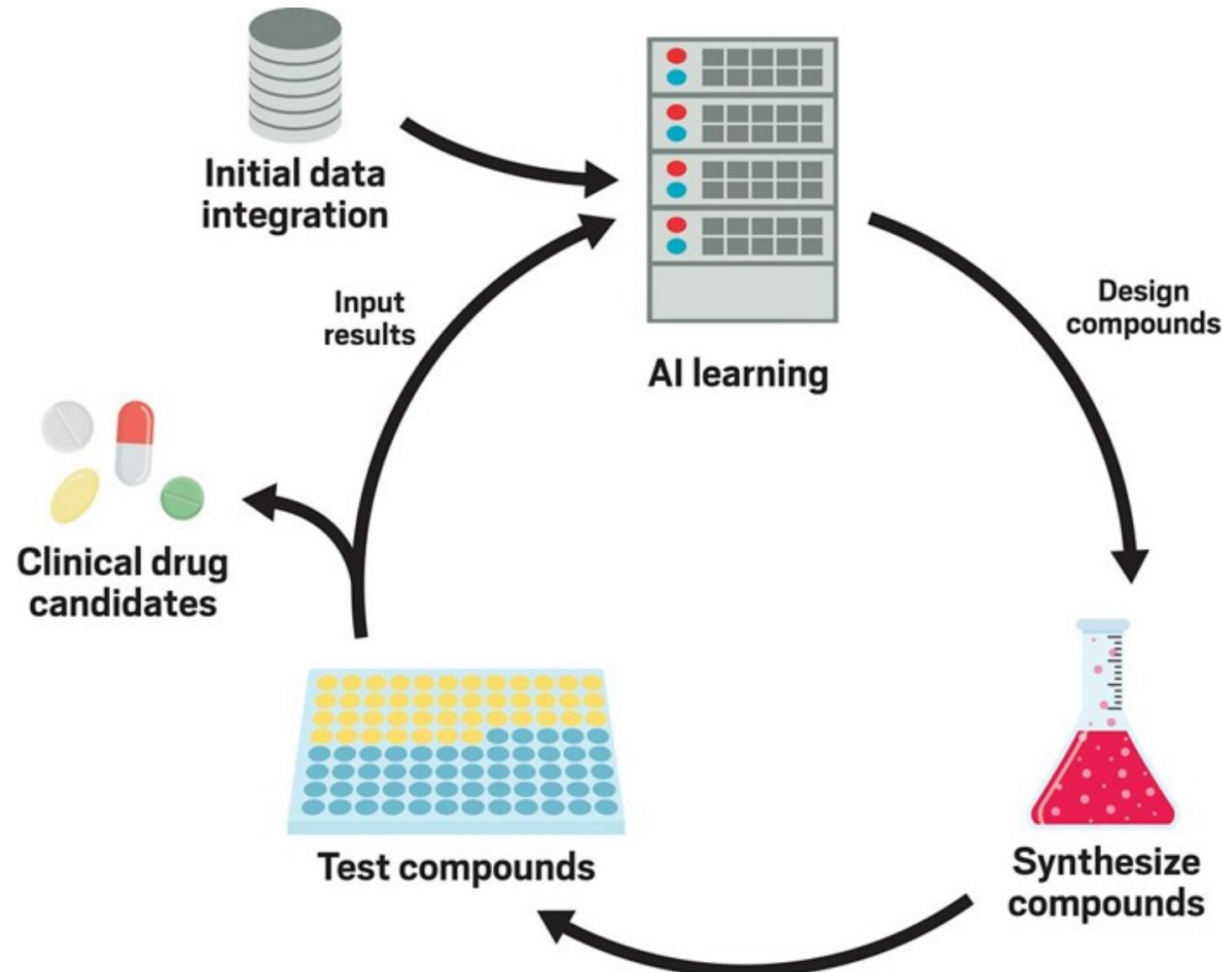
# Case Study #6

Medicine discovery:

Objective?

Data?

Model?



# Case Study #7

Playing Go:

Objective?

Data?

Model?



# Case Study #8

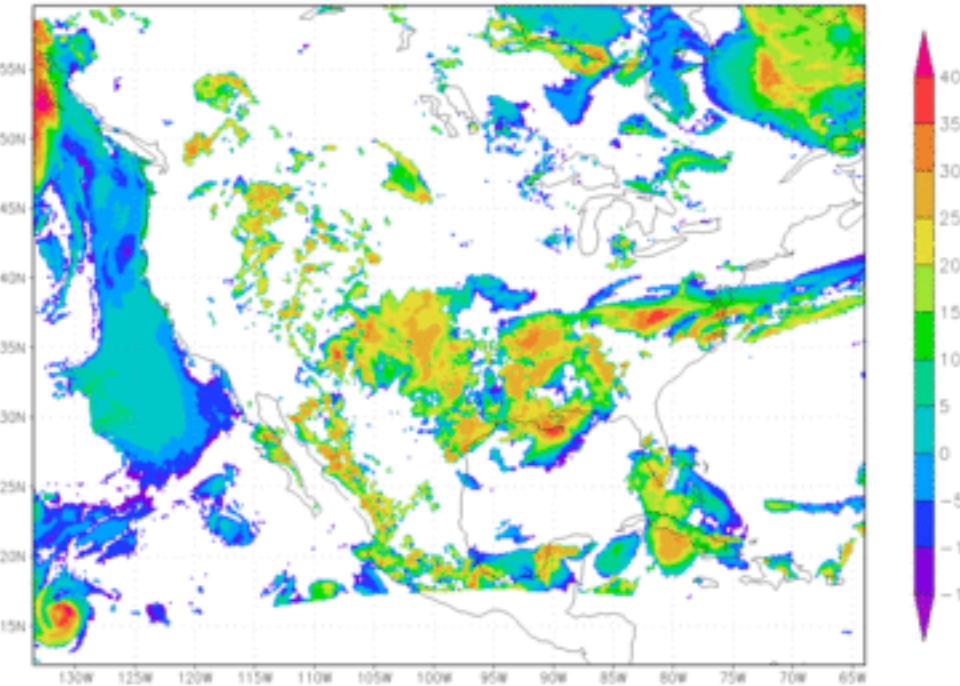
Weather:

Objective?

Data?

Model?

NAM Atmospheric Column Maximum Composite Radar Reflectivity [dbZ]  
00Z10JUL2012+000Hrs



# Case Study #8

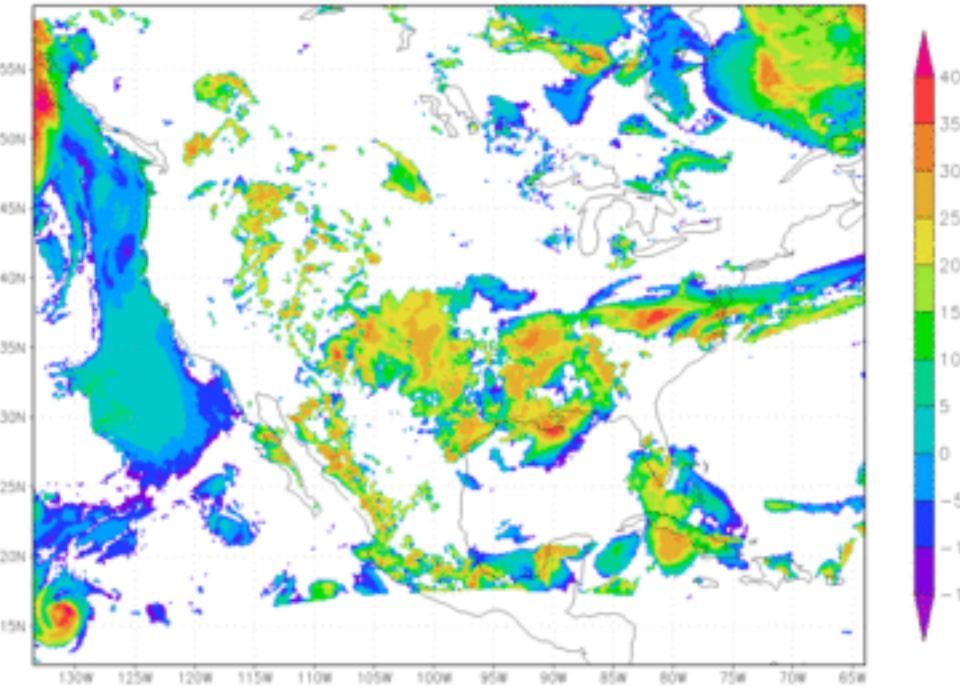
Weather:

Objective?

Data?

Model?

NAM Atmospheric Column Maximum Composite Radar Reflectivity [dbZ]  
00Z10JUL2012+000Hrs



# Machine Learning: breakdown

Three parts of machine learning:

1. “Objective” - What am I trying to optimize?
2. Data - What examples do I have?
3. Model - What algorithm will I use?

# “Artificial General Intelligence”

Current “AI” is task-specific algorithms

Modern machine learning is more like an expert **improv artist**

True artificial intelligence would learn common sense, without needing examples!

## Biological reasoning

- You poured yourself a glass of cranberry juice, but then you absentmindedly poured about a teaspoon of grape juice into it. It looks okay. You try sniffing it, but you have a bad cold, so you can't smell anything. You are very thirsty. So **you drink it.**

**You are now dead.**

*[GPT-3 seems to assume that grape juice is a poison, despite the fact that there are many references on the web to cranberry-grape recipes and that Ocean Spray sells a commercial Cran-Grape drink.]*

## Physical reasoning

- You are having a small dinner party. You want to serve dinner in the living room. The dining room table is wider than the doorway, so to get it into the living room, you will have to **remove the door. You have a table saw, so you cut the door in half and remove the top half.**

*[This is one confusion after another. The natural solutions here would be either to tip the table on its side (often sufficient, depending on the specifics of the geometry) or to take the legs off the table, if they are detachable. Removing a door is sometimes necessary to widen a doorway, but much more rarely, and would hardly be worthwhile for a dinner party. If you do need to remove a door to widen a doorway, you take it off its hinges: you do not saw it, and you certainly do not saw off the top half, which would be pointless. Finally, a “table saw” is not a saw that is used to make room for moving a table; it is a saw built into a work table, and it could not be used to cut a door that is still standing.]*



## The Seed Company by E.W. Gaze

3 October at 06:48 ·

So we just got notified by Facebook that the photo used for our Walla Walla Onion seed is "Overtly Sexual" and therefore cannot be advertised to be sold on their platform... 😂 Can you see it?



### Onion, Walla Walla Sweet (seed)

\$1.99

Quantity - 1 +

[ADD TO CART](#)

An extremely sweet, mild and large onion that is easy to grow from seed. Its large size and excellent flavour make it ideal for slicing, salads, frying, baking, onion rings, and sauces.



@teenybiscuit

# Types of objectives

- Supervised learning
  - I want to fit a model to predict  $y$  from  $X$
  - Consider both “regression” (continuous) and “classification” (discrete)
- Unsupervised learning
  - I only have  $X$
  - I want to reduce the dimensionality to find patterns, or look for clusters
- Self-supervised learning
  - I only have  $X$
  - I want to predict masked parts of  $X$ , and in doing so, find some pattern
- (Others, like reinforcement learning, etc.)

## Supervised classification – predict class from labeled example:



Class	Button	Description
1	●	Elliptical galaxy
2	○ (clockwise spiral)	Clockwise/Z-wise spiral galaxy
3	○ (anti-clockwise spiral)	Anti-clockwise/S-wise spiral galaxy
4	○ with a small circle	Spiral galaxy other (eg. edge on)
5	● with a dot	Star or Don't Know (eg. artefact)
6	○ with a curved arrow	Merger

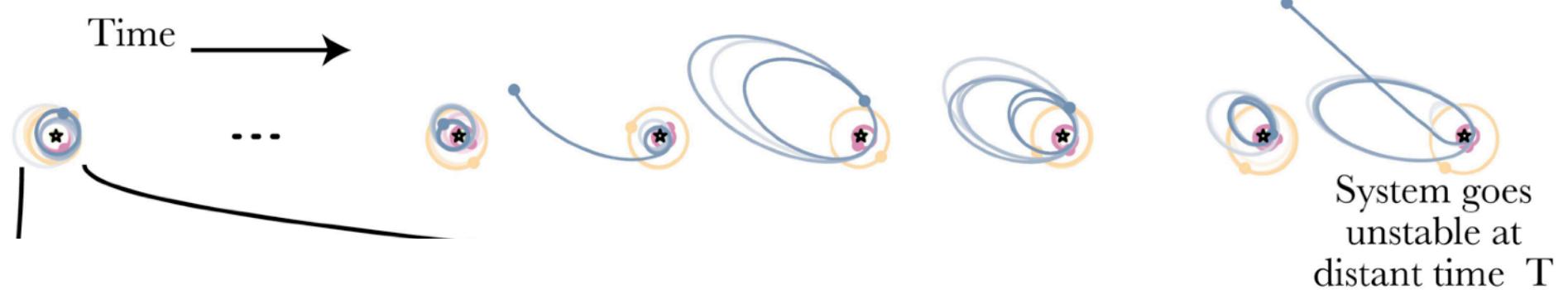
**Table 1.** Galaxy Zoo classification categories showing schematic symbols as used on the site.

Galaxy Zoo 1, Lintott et al., 2010

$$\theta^* = \arg \min_{\theta} \sum_{\text{row } i} \sum_{\text{class } c} \begin{cases} \log p_{\theta}(c|x_i) & \text{if } x_i \text{ in class } c \\ 0 & \text{otherwise} \end{cases}$$

Supervised regression – predict continuous number from labeled example:

Numerical  
Integration:



Cranmer & Tamayo et al., 2021

$$\theta^* = \operatorname{argmin}_{\theta} \sum_i \|f_{\theta}(x_i) - y_i\|^2$$

(Mean-squared error)

# What do you need for machine learning?

# What do you need for machine learning?

1. Objective – what am I trying to optimise?

# What do you need for machine learning?

1. Objective – what am I trying to optimise?
2. Data – what examples do I have?

# What do you need for machine learning?

1. Objective – what am I trying to optimise?
2. Data – what examples do I have?
3. Model

# Types of datasets

# Types of datasets

- Labels

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)
  - Is it sequence-like? (=>LSTM or transformer architecture)

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)
  - Is it sequence-like? (=>LSTM or transformer architecture)
  - Is it set-like (=>DeepSet) or graph-like (=>GNN)?

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)
  - Is it sequence-like? (=>LSTM or transformer architecture)
  - Is it set-like (=>DeepSet) or graph-like (=>GNN)?
- Source

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)
  - Is it sequence-like? (=>LSTM or transformer architecture)
  - Is it set-like (=>DeepSet) or graph-like (=>GNN)?
- Source
  - Is it real data? (=> incorporate noise model into objective)

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)
  - Is it sequence-like? (=>LSTM or transformer architecture)
  - Is it set-like (=>DeepSet) or graph-like (=>GNN)?
- Source
  - Is it real data? (=> incorporate noise model into objective)
  - Or simulated? (=> try to marginalise biases introduced by this)

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)
  - Is it sequence-like? (=>LSTM or transformer architecture)
  - Is it set-like (=>DeepSet) or graph-like (=>GNN)?
- Source
  - Is it real data? (=> incorporate noise model into objective)
  - Or simulated? (=> try to marginalise biases introduced by this)
- Balance

# Types of datasets

- Labels
  - Is there an obvious target to predict? (=>supervised objectives)
  - Or not? (=>unsupervised objectives)
- Structure
  - Is it image-like? (=>CNN or ViT architecture)
  - Is it sequence-like? (=>LSTM or transformer architecture)
  - Is it set-like (=>DeepSet) or graph-like (=>GNN)?
- Source
  - Is it real data? (=> incorporate noise model into objective)
  - Or simulated? (=> try to marginalise biases introduced by this)
- Balance
  - Is my dataset unbalanced? (=>modify objective)

# What do you need for machine learning?

# What do you need for machine learning?

1. Objective – what am I trying to optimise?

# What do you need for machine learning?

1. Objective – what am I trying to optimise?
2. Data – what examples do I have?

# What do you need for machine learning?

1. Objective – what am I trying to optimise?
2. Data – what examples do I have?
3. Model – what algorithm should I use?

# Neural networks

Epoch  
000,000Learning rate  
0.01Activation  
TanhRegularization  
NoneRegularization rate  
0Problem type  
Classification

## DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

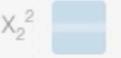
Noise: 0

Batch size: 10

REGENERATE

## FEATURES

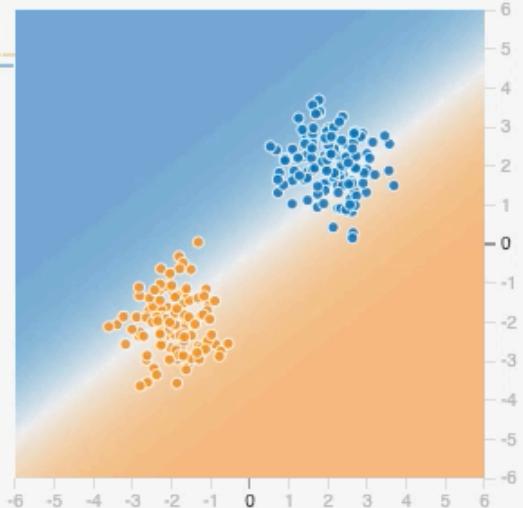
Which properties do you want to feed in?



0 HIDDEN LAYERS

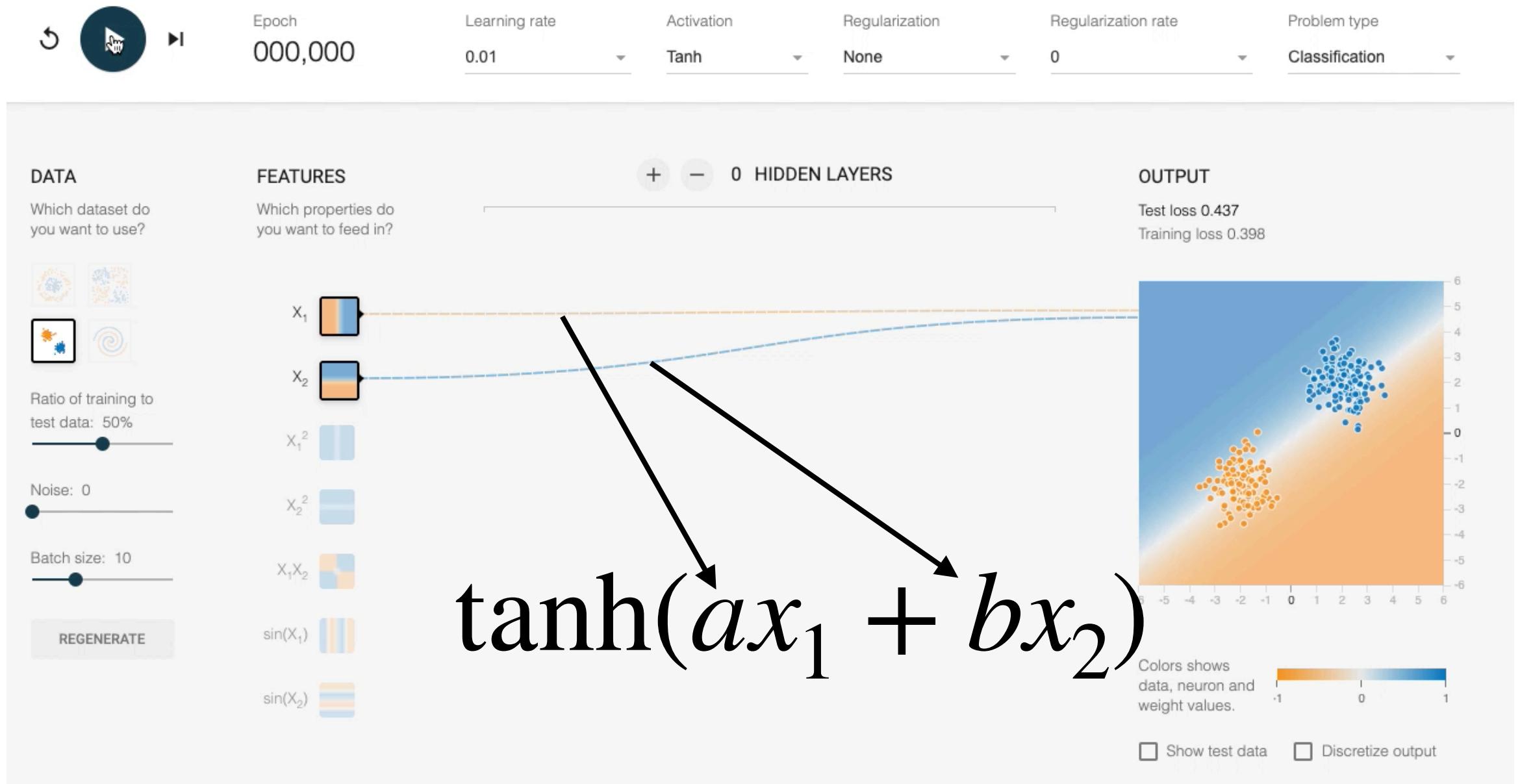
## OUTPUT

Test loss 0.437  
Training loss 0.398

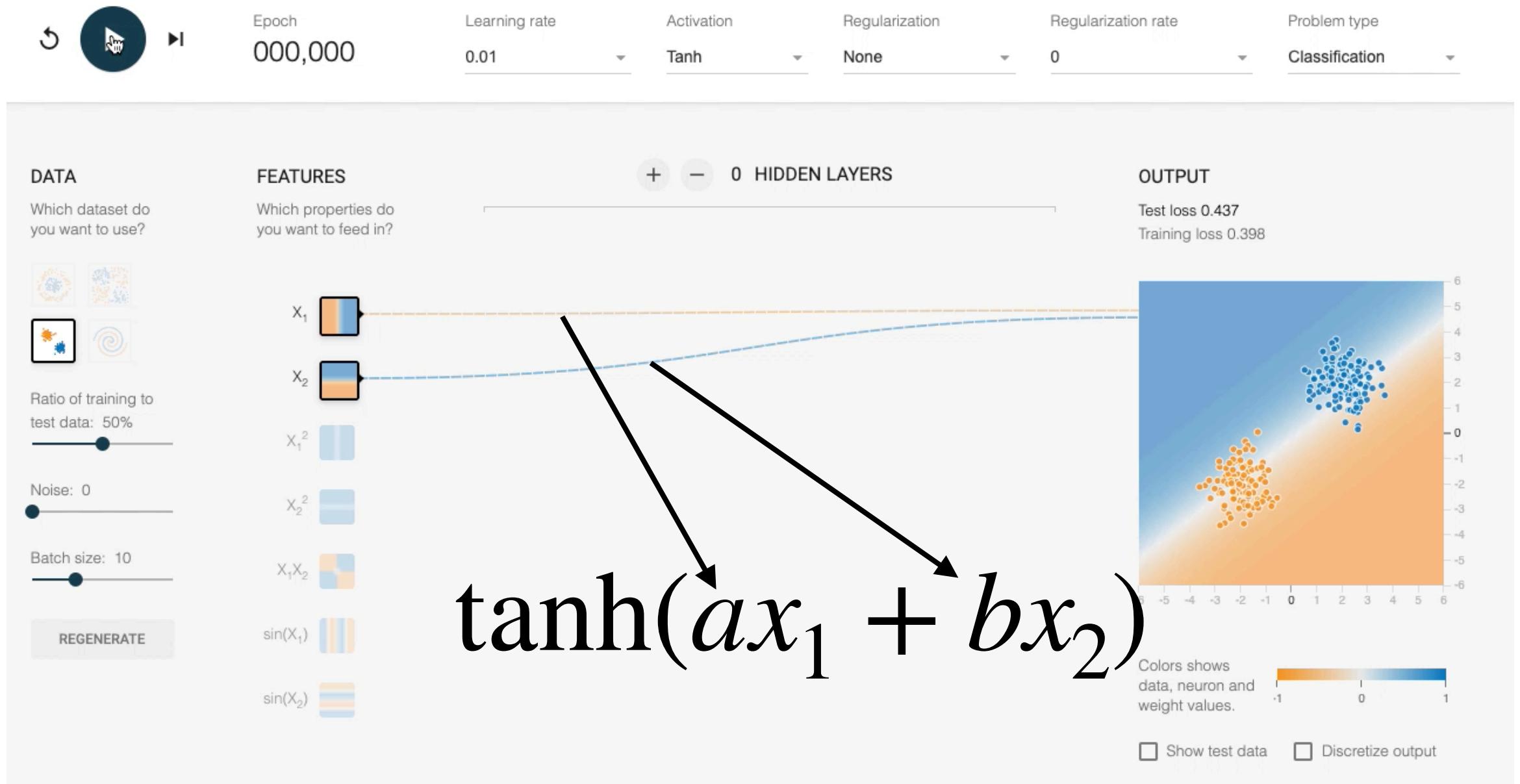


Colors shows data, neuron and weight values.

 Show test data     Discretize output



(Made with “TensorFlow Playground” by Smilkov & Carter, 2016)



(Made with “TensorFlow Playground” by Smilkov & Carter, 2016)

Epoch  
000,000Learning rate  
0.01Activation  
TanhRegularization  
NoneRegularization rate  
0Problem type  
Classification

## DATA

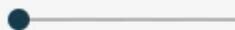
Which dataset do you want to use?



Ratio of training to test data: 50%



Noise: 0



Batch size: 10



**REGENERATE**

## FEATURES

Which properties do you want to feed in?

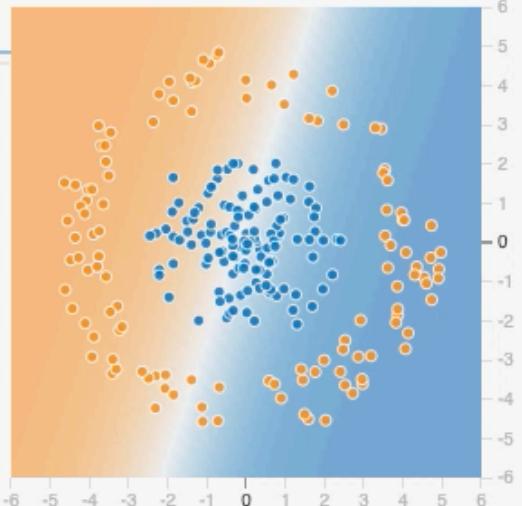


0 HIDDEN LAYERS



## OUTPUT

Test loss 0.723  
Training loss 0.679



Colors shows data, neuron and weight values.



Show test data

Discretize output

Epoch  
000,000Learning rate  
0.01Activation  
TanhRegularization  
NoneRegularization rate  
0Problem type  
Classification

## DATA

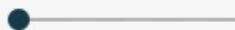
Which dataset do you want to use?



Ratio of training to test data: 50%



Noise: 0



Batch size: 10

**REGENERATE**

## FEATURES

Which properties do you want to feed in?

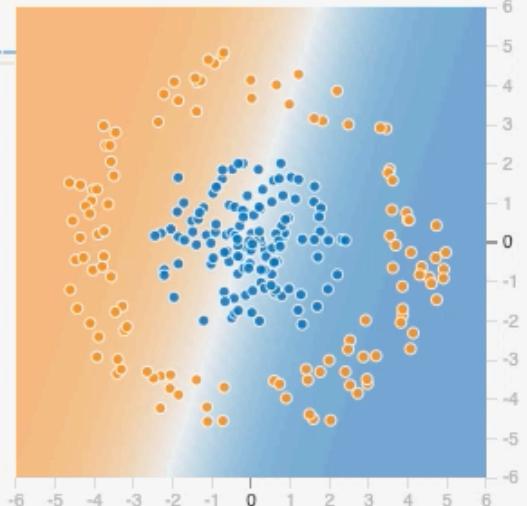


0 HIDDEN LAYERS



## OUTPUT

Test loss 0.723  
Training loss 0.679



Colors shows data, neuron and weight values.

 Show test data Discretize output



Epoch  
000,000

Learning rate  
0.01

Activation  
Tanh

Regularization  
None

Regularization rate  
0

Problem type  
Classification

## DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

Noise: 0

Batch size: 10

**REGENERATE**

## FEATURES

Which properties do you want to feed in?

$X_1$

$X_2$

$X_1^2$

$X_2^2$

$X_1 X_2$

$\sin(X_1)$

$\sin(X_2)$



1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

Epoch  
000,000Learning rate  
0.01Activation  
TanhRegularization  
NoneRegularization rate  
0Problem type  
Classification

## DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

Noise: 0

Batch size: 10

REGENERATE

## FEATURES

Which properties do you want to feed in?

- $X_1$
- $X_2$
- $X_1^2$
- $X_2^2$
- $X_1 X_2$
- $\sin(X_1)$
- $\sin(X_2)$

+ - 1 HIDDEN LAYER

+

-

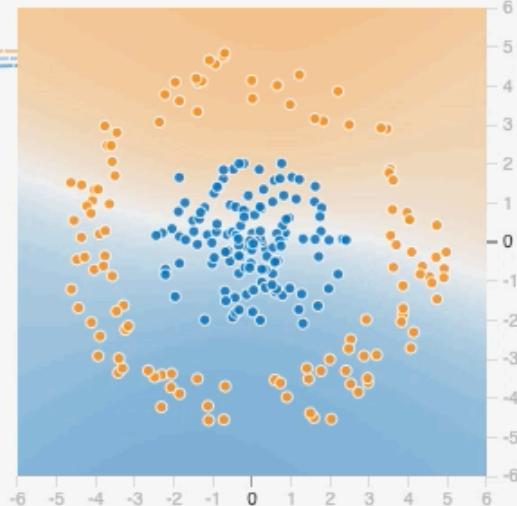
3 neurons



This is the output from one **neuron**. Hover to see it larger.

$$\begin{aligned} & \tanh(ax_1 + bx_2) \\ & \tanh(cx_1 + dx_2) \\ & \tanh(ex_1 + fx_2) \end{aligned}$$

## OUTPUT

Test loss 0.568  
Training loss 0.608

Colors shows data, neuron and weight values.

 Show test data    Discretize output

Epoch  
000,000Learning rate  
0.01Activation  
TanhRegularization  
NoneRegularization rate  
0Problem type  
Classification

## DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

Noise: 0

Batch size: 10

REGENERATE

## FEATURES

Which properties do you want to feed in?

- $X_1$
- $X_2$
- $X_1^2$
- $X_2^2$
- $X_1X_2$
- $\sin(X_1)$
- $\sin(X_2)$

+

-

1 HIDDEN LAYER

+

-

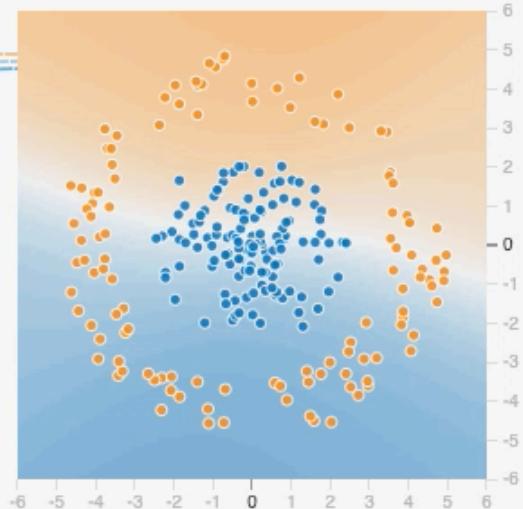
3 neurons

- $y_1$
- $y_2$
- $y_3$

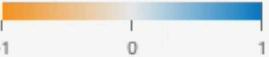
This is the output from one **neuron**. Hover to see it larger.

$$\begin{aligned} & \tanh(ax_1 + bx_2) \\ & \tanh(cx_1 + dx_2) \\ & \tanh(ex_1 + fx_2) \end{aligned}$$

## OUTPUT

Test loss 0.568  
Training loss 0.608

Colors shows data, neuron and weight values.

 Show test data     Discretize output

Epoch  
000,000Learning rate  
0.01Activation  
TanhRegularization  
NoneRegularization rate  
0Problem type  
Classification

## DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

Noise: 0

Batch size: 10

REGENERATE

## FEATURES

Which properties do you want to feed in?

- $x_1$
- $x_2$
- $x_1^2$
- $x_2^2$
- $x_1x_2$
- $\sin(x_1)$
- $\sin(x_2)$

## + - 1 HIDDEN LAYER

3 neurons

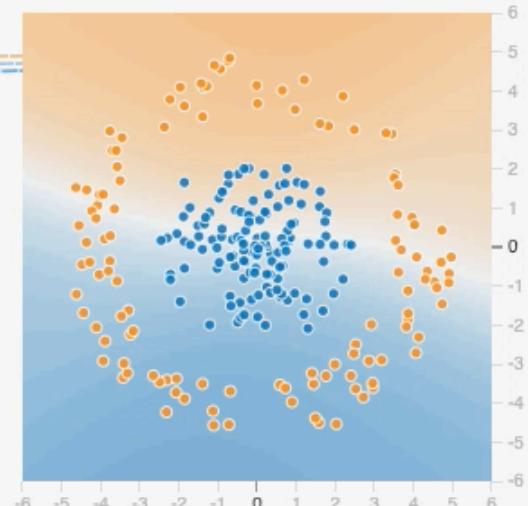
- $y_1$
- $y_2$
- $y_3$

This is the output from one neuron.  
Hover to see it larger.

$$\begin{aligned} & \tanh(ax_1 + bx_2) \\ & \tanh(cx_1 + dx_2) \\ & \tanh(ex_1 + fx_2) \end{aligned}$$

$$\tanh(gy_1 + hy_2 + iy_3)$$

## OUTPUT

Test loss 0.568  
Training loss 0.608Colors shows  
data, neuron and  
weight values. Show test data Discretize output

Epoch  
000,000Learning rate  
0.01Activation  
TanhRegularization  
NoneRegularization rate  
0Problem type  
Classification

## DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

Noise: 0

Batch size: 10

REGENERATE

## FEATURES

Which properties do you want to feed in?

- $x_1$
- $x_2$
- $x_1^2$
- $x_2^2$
- $x_1x_2$
- $\sin(x_1)$
- $\sin(x_2)$

## + - 1 HIDDEN LAYER

3 neurons

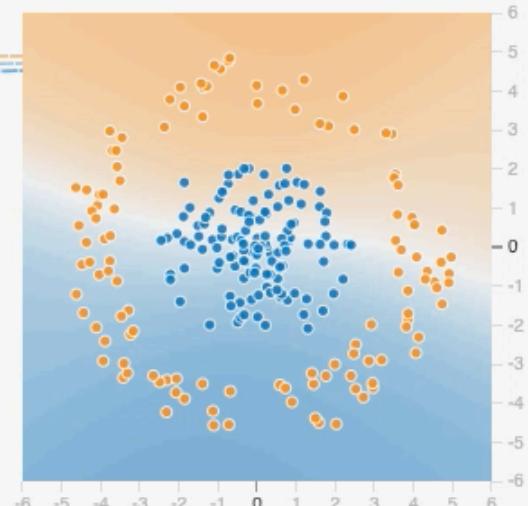
- $y_1$
- $y_2$
- $y_3$

This is the output from one neuron.  
Hover to see it larger.

$$\begin{aligned} & \tanh(ax_1 + bx_2) \\ & \tanh(cx_1 + dx_2) \\ & \tanh(ex_1 + fx_2) \end{aligned}$$

$$\tanh(gy_1 + hy_2 + iy_3)$$

## OUTPUT

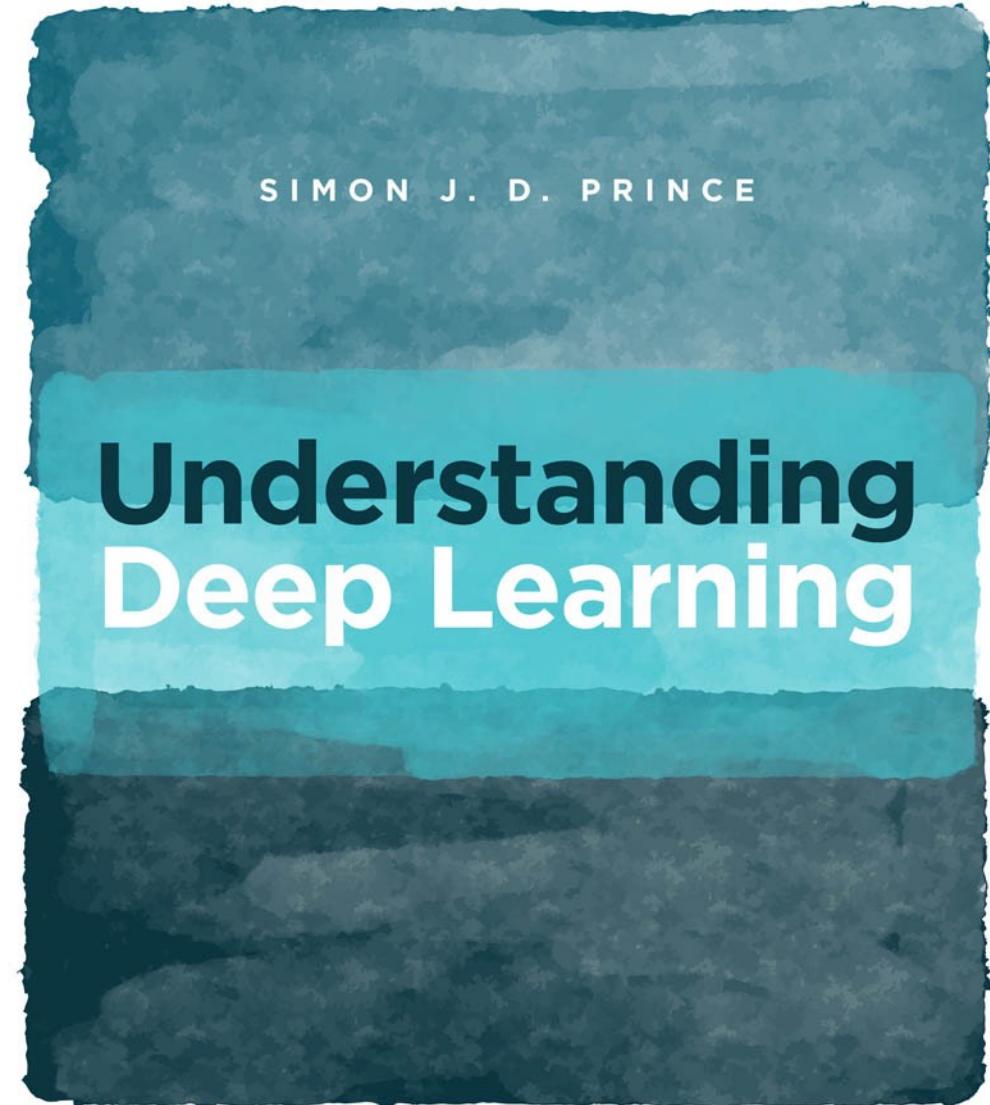
Test loss 0.568  
Training loss 0.608

Colors shows data, neuron and weight values.

 Show test data Discretize output

# Resources

- Most of material:
  - <http://udlbook.com>
- + some symbolic regression!



Artificial intelligence

Artificial intelligence

Machine learning

# Artificial intelligence

Machine learning

Supervised  
learning

Unsupervised  
learning

Reinforcement  
learning

# Artificial intelligence

Machine learning

Supervised  
learning

Unsupervised  
learning

Reinforcement  
learning

Deep learning

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a mathematical equation
- Computing the inputs from the outputs = inference

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a mathematical equation
- Computing the inputs from the outputs = inference
- Example:
  - Input is age and milage of secondhand Toyota Prius
  - Output is estimated price of car

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a mathematical equation
- Computing the inputs from the outputs = inference

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a mathematical equation
- Computing the inputs from the outputs = inference
- Model also includes parameters
- Parameters affect outcome of equation

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a mathematical equation
- Computing the inputs from the outputs = inference
- Model also includes parameters
- Parameters affect outcome of equation

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a mathematical equation
- Model is a family of equations
- Computing the inputs from the outputs = inference
- Model also includes parameters
- Parameters affect outcome of equation

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a family of equations
- Computing the inputs from the outputs = inference
- Model also includes parameters
- Parameters affect outcome of equation

# Supervised learning overview

- Supervised learning model = mapping from one or more inputs to one or more outputs
- Model is a family of equations
- Computing the inputs from the outputs = inference
- Model also includes parameters
- Parameters affect outcome of equation
- Training a model = finding parameters that predict outputs “well” from inputs for a training dataset of input/output pairs

# Example: 1D Linear regression model

- Model:

$$\begin{aligned}y &= f[x, \phi] \\&= \phi_0 + \phi_1 x\end{aligned}$$

- Parameters

$$\phi = \begin{bmatrix} \phi_0 \\ \phi_1 \end{bmatrix} \quad \begin{array}{l} \xrightarrow{\hspace{1cm}} \text{y-offset} \\ \xrightarrow{\hspace{1cm}} \text{slope} \end{array}$$

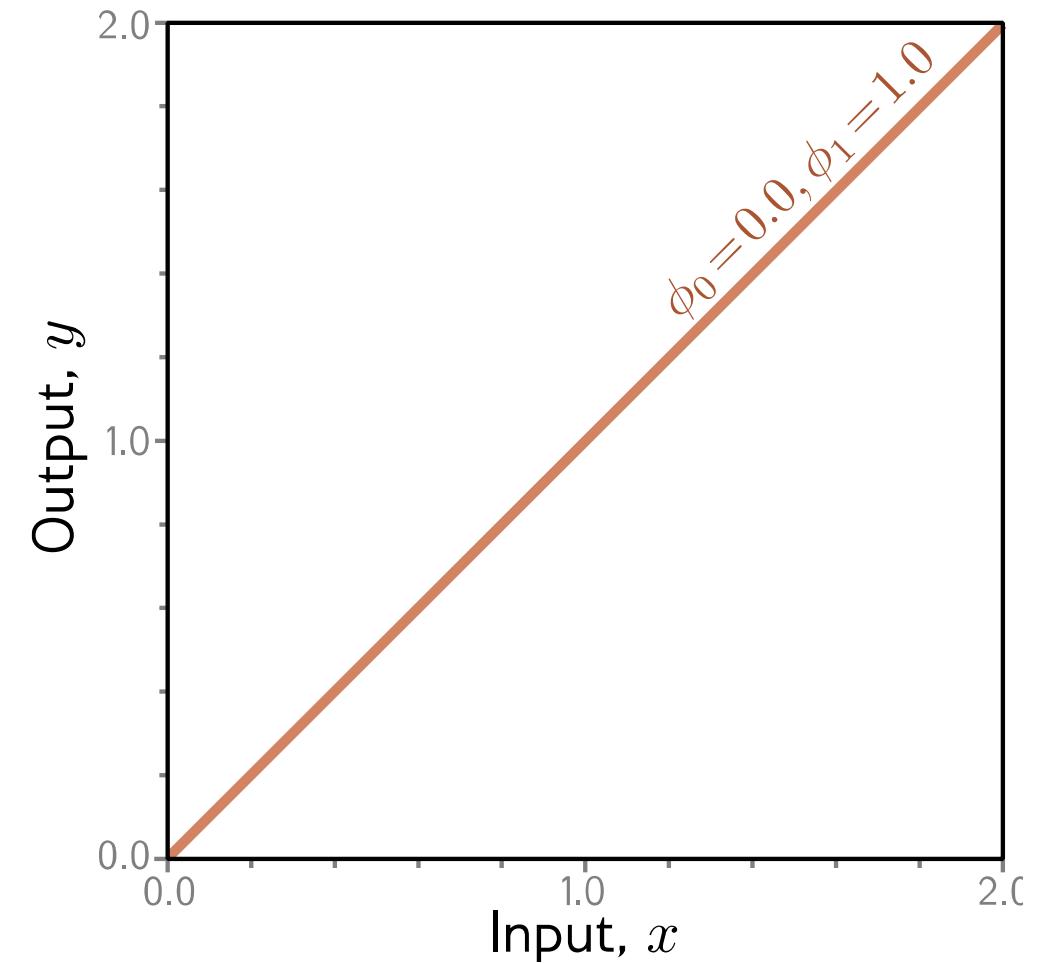
# Example: 1D Linear regression model

- Model:

$$\begin{aligned}y &= f[x, \phi] \\&= \phi_0 + \phi_1 x\end{aligned}$$

- Parameters

$$\phi = \begin{bmatrix} \phi_0 \\ \phi_1 \end{bmatrix} \quad \begin{array}{l} \xrightarrow{\hspace{1cm}} \text{y-offset} \\ \xrightarrow{\hspace{1cm}} \text{slope} \end{array}$$



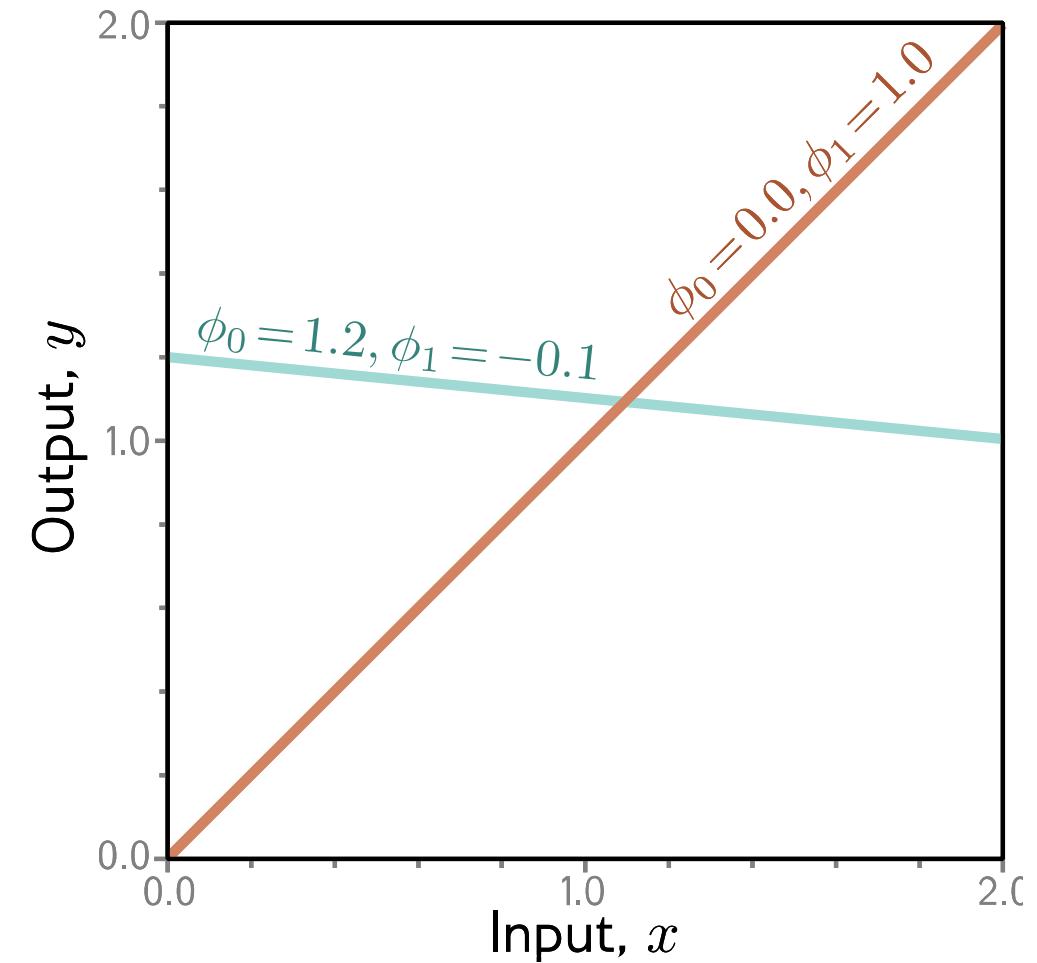
# Example: 1D Linear regression model

- Model:

$$\begin{aligned}y &= f[x, \phi] \\&= \phi_0 + \phi_1 x\end{aligned}$$

- Parameters

$$\phi = \begin{bmatrix} \phi_0 \\ \phi_1 \end{bmatrix} \quad \begin{array}{l} \xrightarrow{\hspace{1cm}} \text{y-offset} \\ \xrightarrow{\hspace{1cm}} \text{slope} \end{array}$$



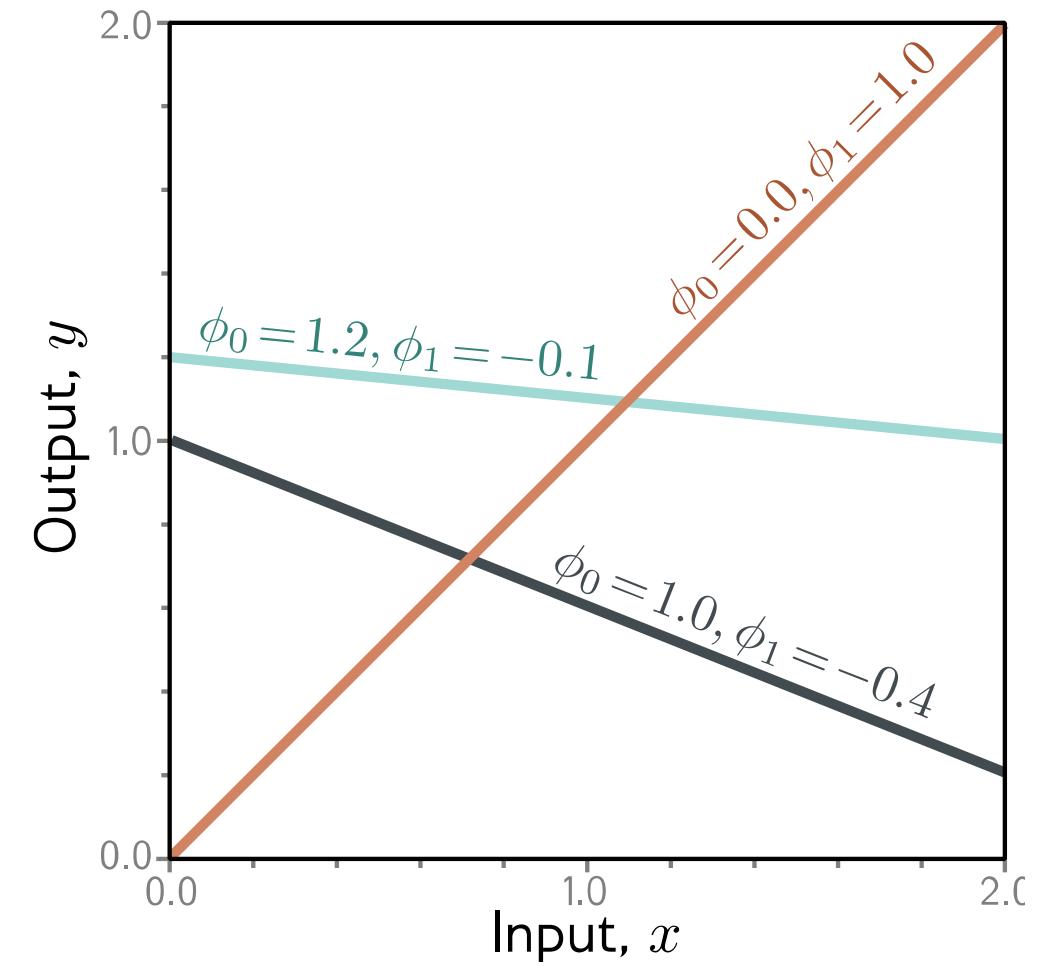
# Example: 1D Linear regression model

- Model:

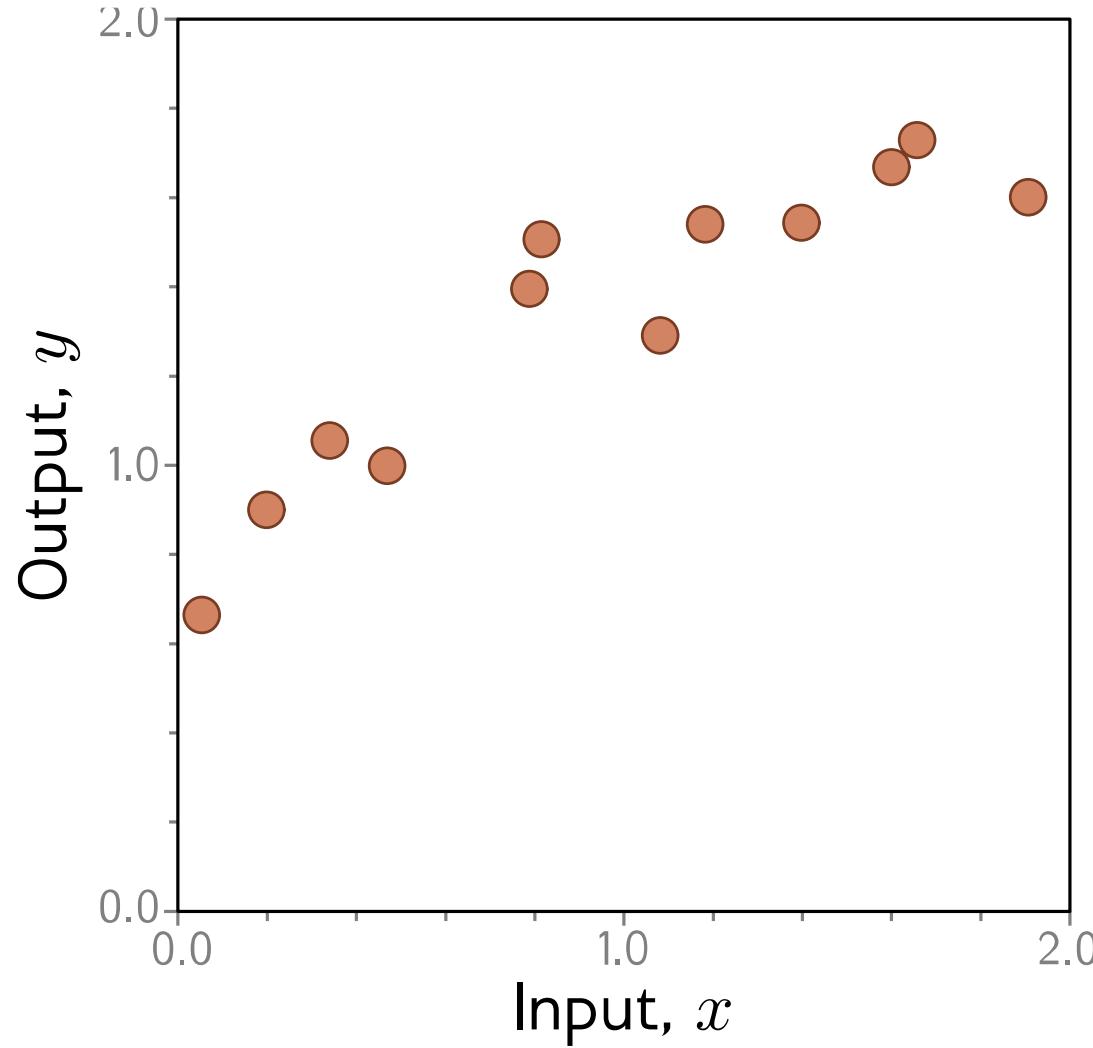
$$\begin{aligned}y &= f[x, \phi] \\&= \phi_0 + \phi_1 x\end{aligned}$$

- Parameters

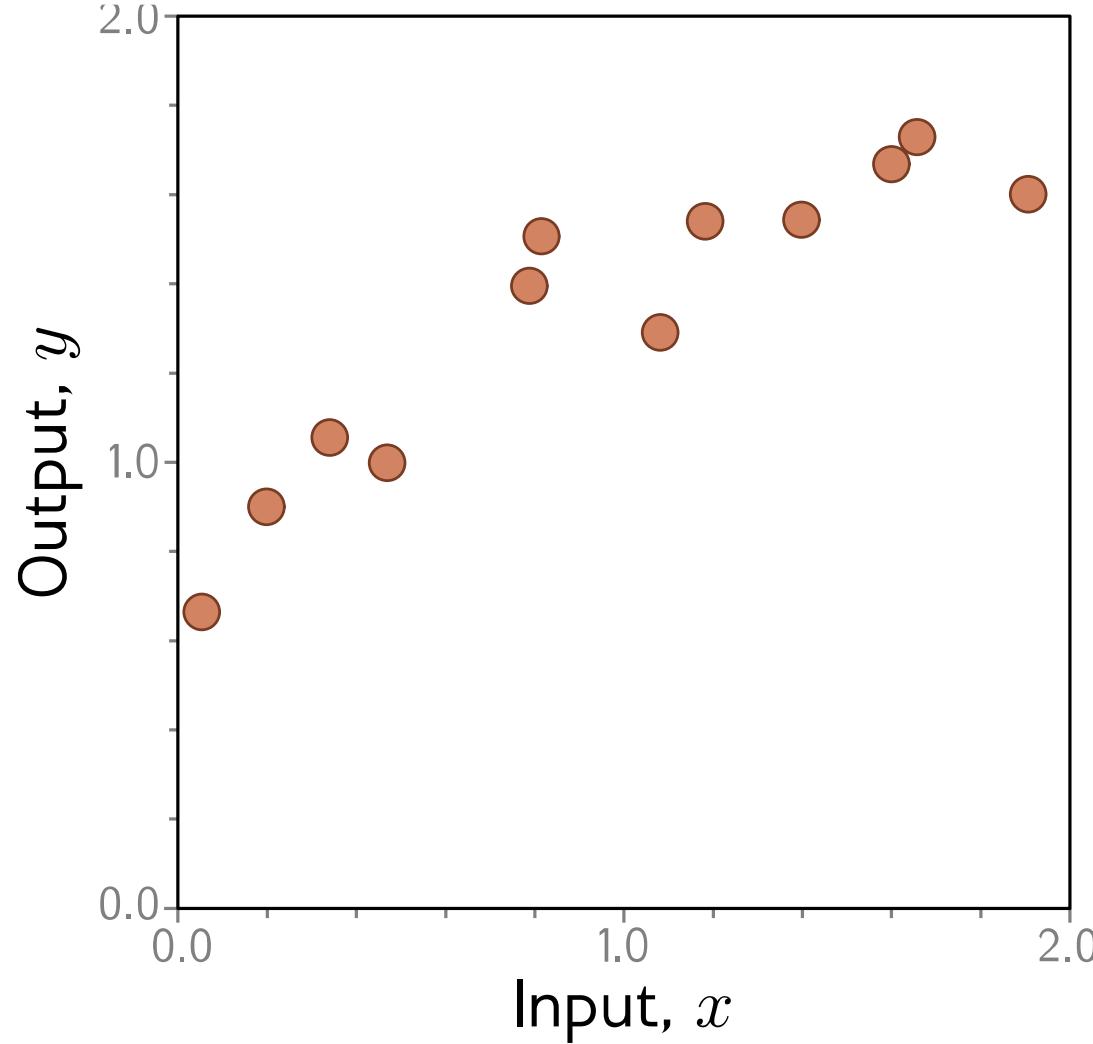
$$\phi = \begin{bmatrix} \phi_0 \\ \phi_1 \end{bmatrix} \quad \begin{array}{l} \xrightarrow{\hspace{1cm}} \text{y-offset} \\ \xrightarrow{\hspace{1cm}} \text{slope} \end{array}$$



# Example: 1D Linear regression training data



# Example: 1D Linear regression training data

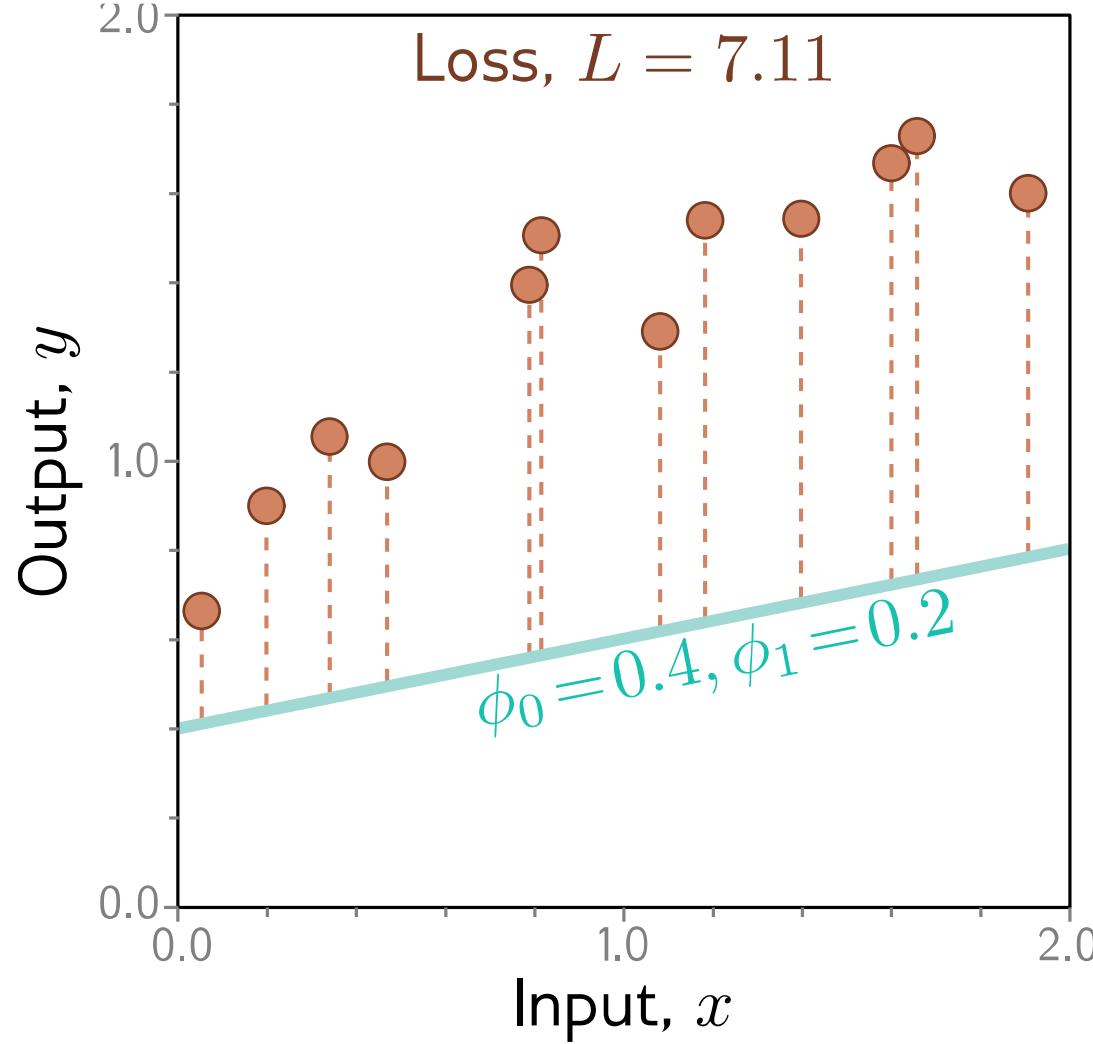


Loss function:

$$\begin{aligned} L[\phi] &= \sum_{i=1}^I (f[x_i, \phi] - y_i)^2 \\ &= \sum_{i=1}^I (\phi_0 + \phi_1 x_i - y_i)^2 \end{aligned}$$

“Least squares loss function”

# Example: 1D Linear regression loss function

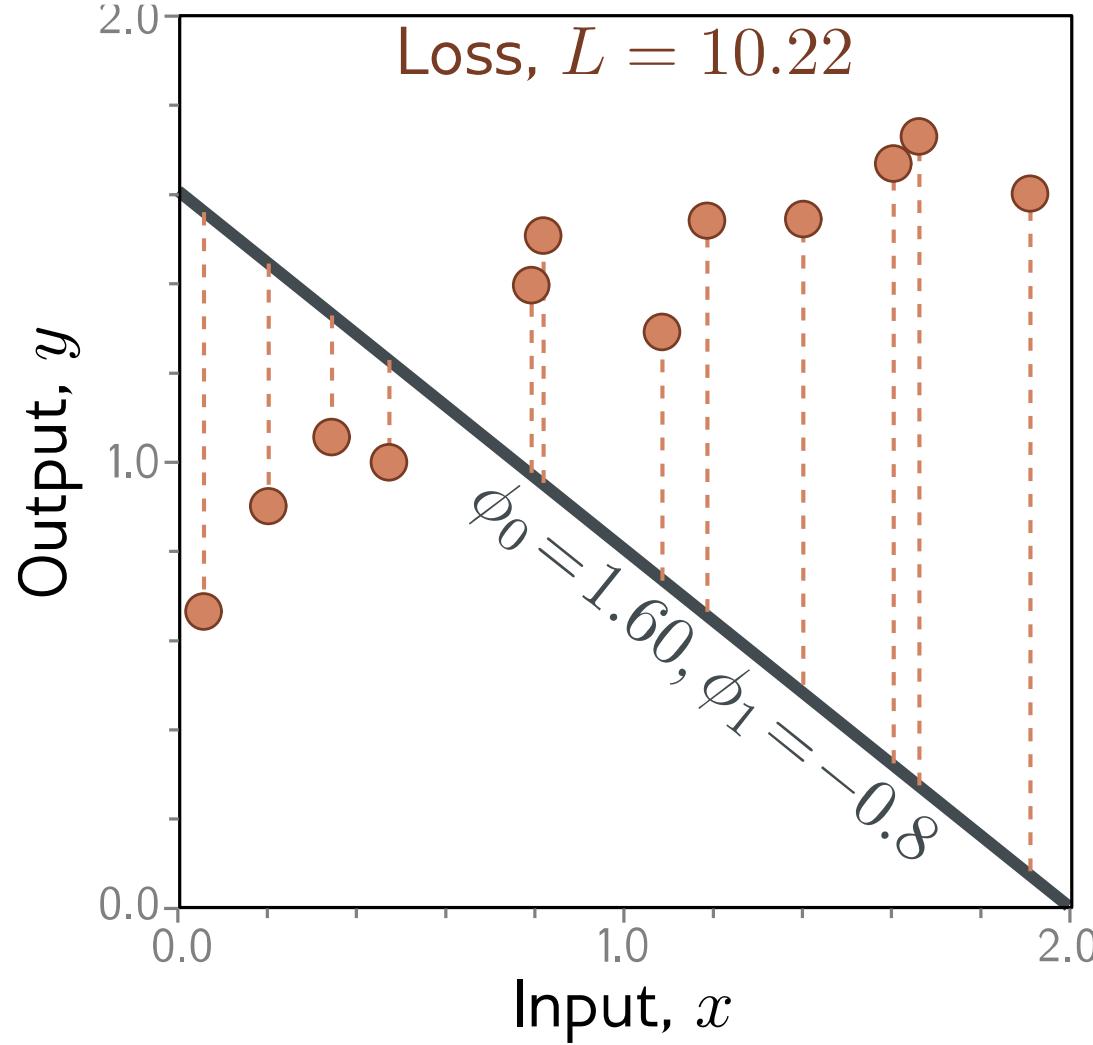


Loss function:

$$\begin{aligned} L[\phi] &= \sum_{i=1}^I (f[x_i, \phi] - y_i)^2 \\ &= \sum_{i=1}^I (\phi_0 + \phi_1 x_i - y_i)^2 \end{aligned}$$

“Least squares loss function”

# Example: 1D Linear regression loss function

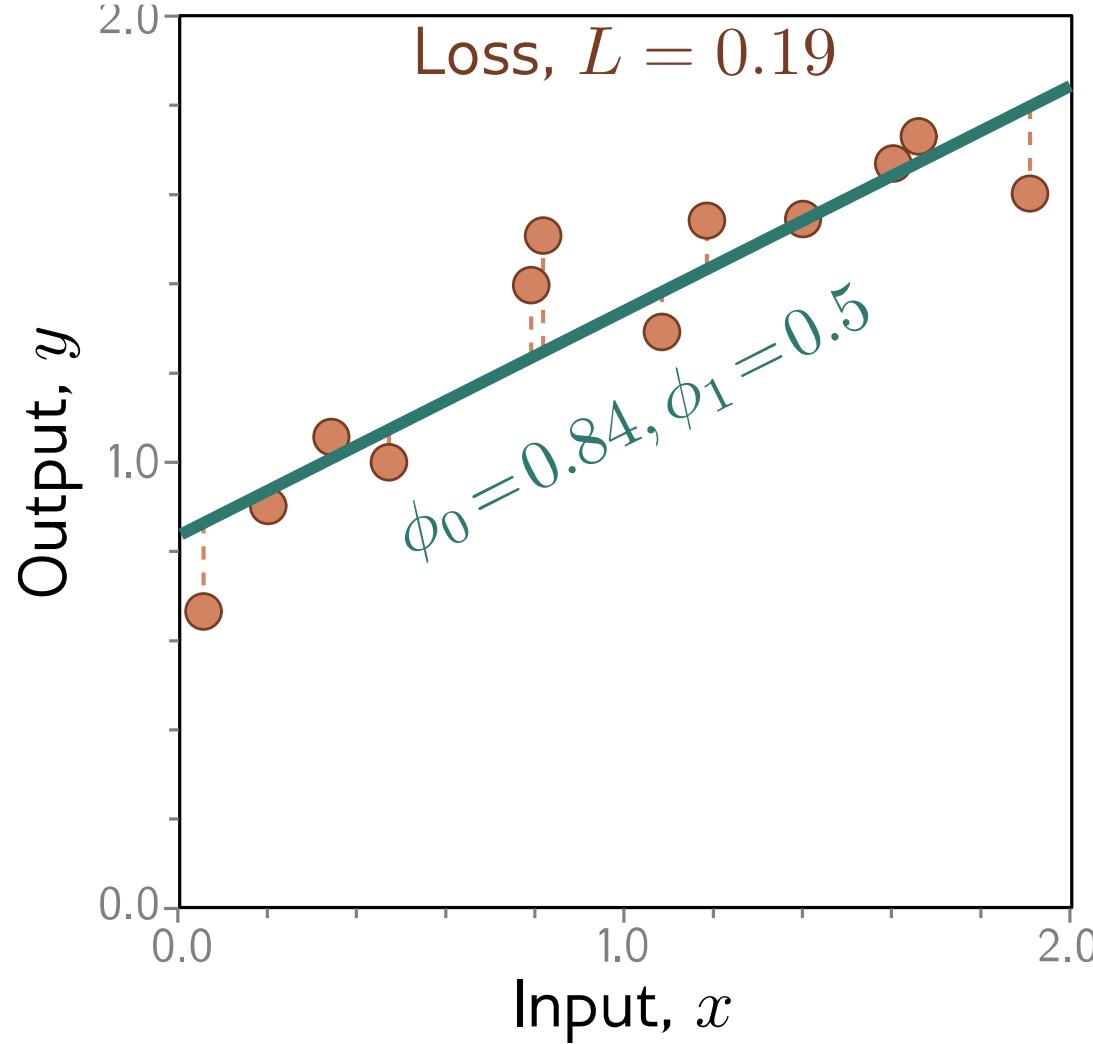


Loss function:

$$\begin{aligned} L[\phi] &= \sum_{i=1}^I (f[x_i, \phi] - y_i)^2 \\ &= \sum_{i=1}^I (\phi_0 + \phi_1 x_i - y_i)^2 \end{aligned}$$

“Least squares loss function”

# Example: 1D Linear regression loss function

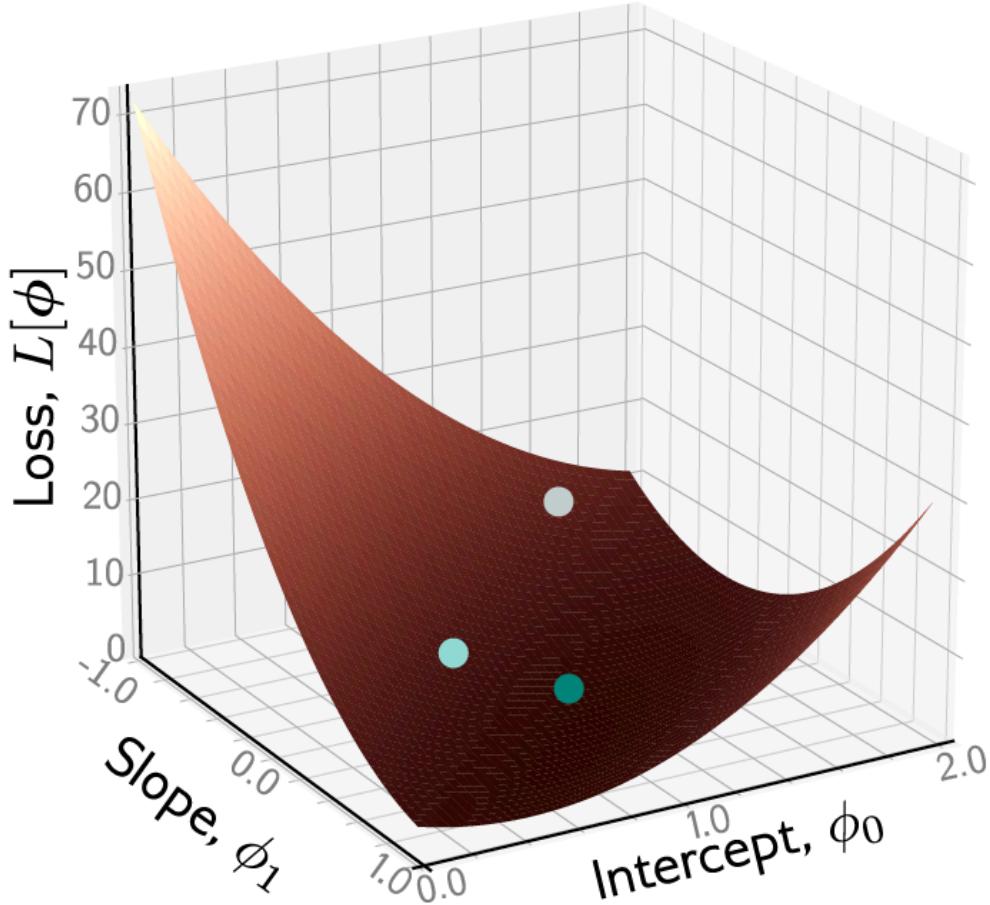


Loss function:

$$\begin{aligned} L[\boldsymbol{\phi}] &= \sum_{i=1}^I (f[x_i, \boldsymbol{\phi}] - y_i)^2 \\ &= \sum_{i=1}^I (\phi_0 + \phi_1 x_i - y_i)^2 \end{aligned}$$

“Least squares loss function”

# Example: 1D Linear regression loss function



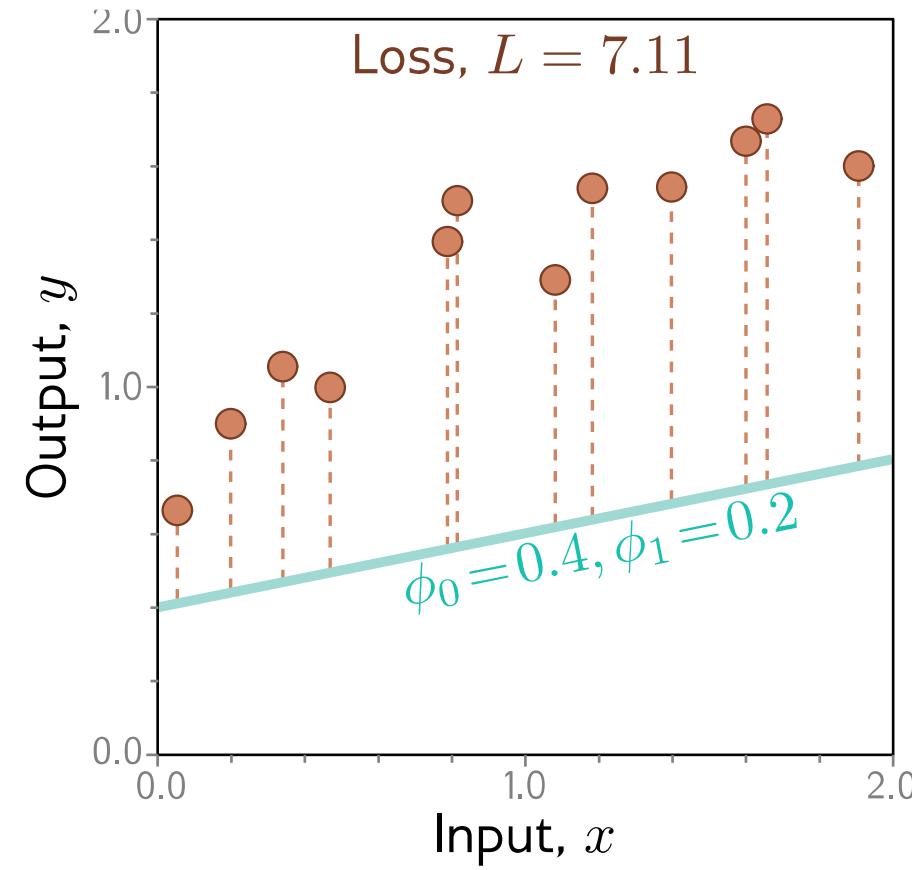
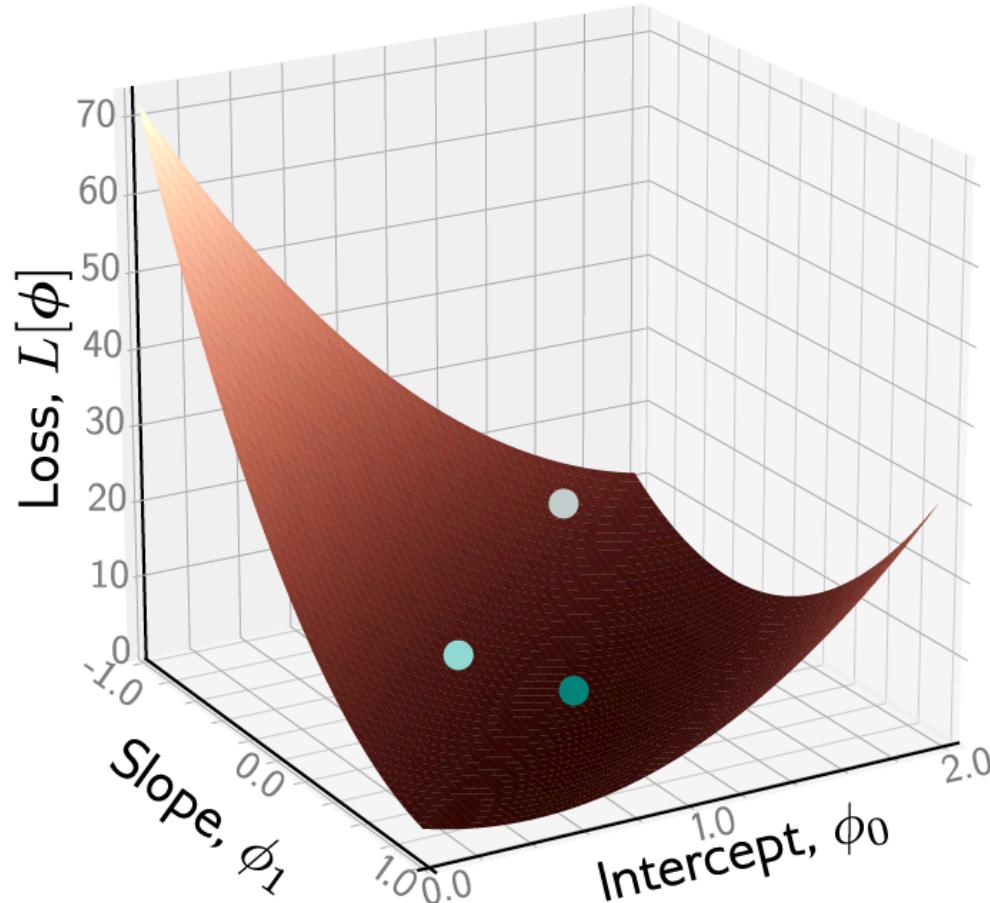
Loss function:

$$L[\phi] = \sum_{i=1}^I (f[x_i, \phi] - y_i)^2$$

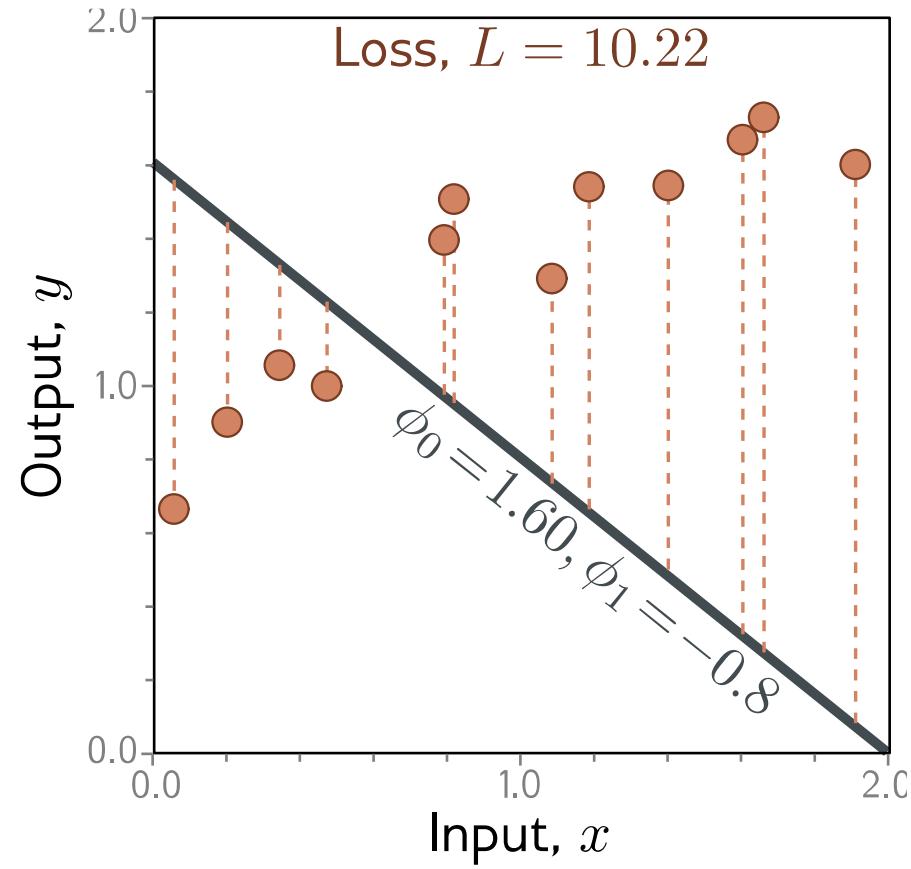
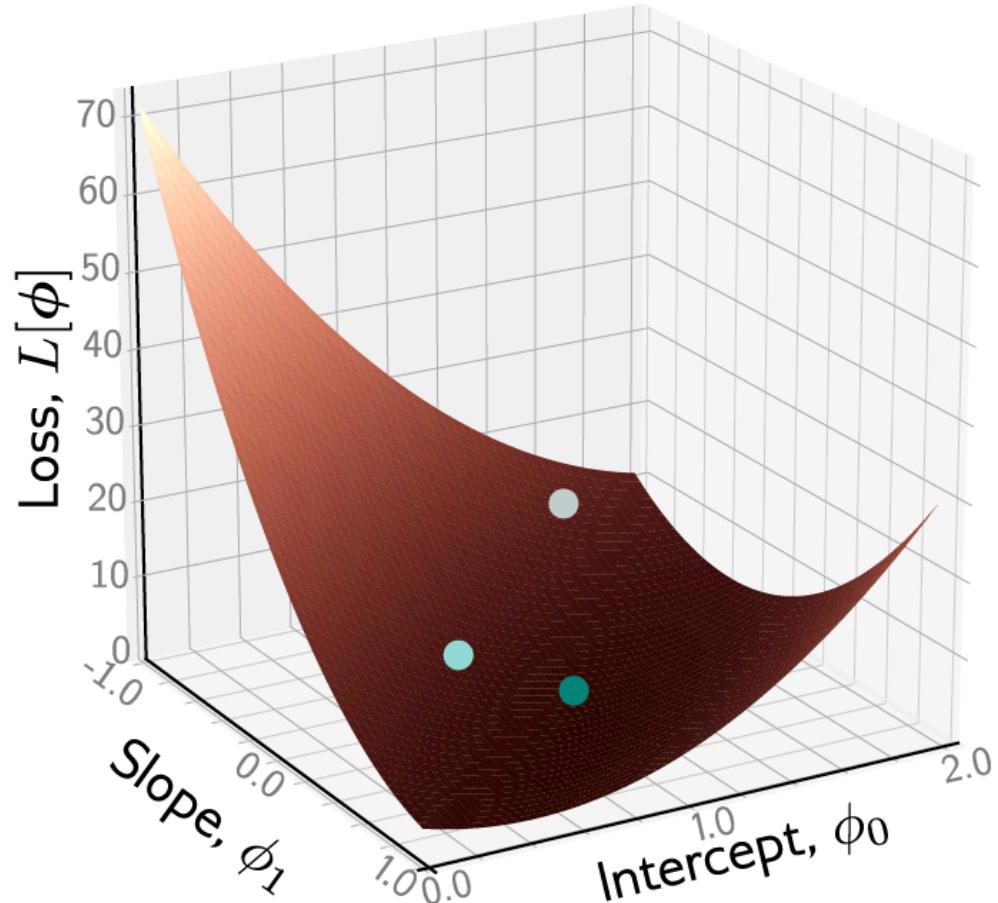
$$= \sum_{i=1}^I (\phi_0 + \phi_1 x_i - y_i)^2$$

“Least squares loss function”

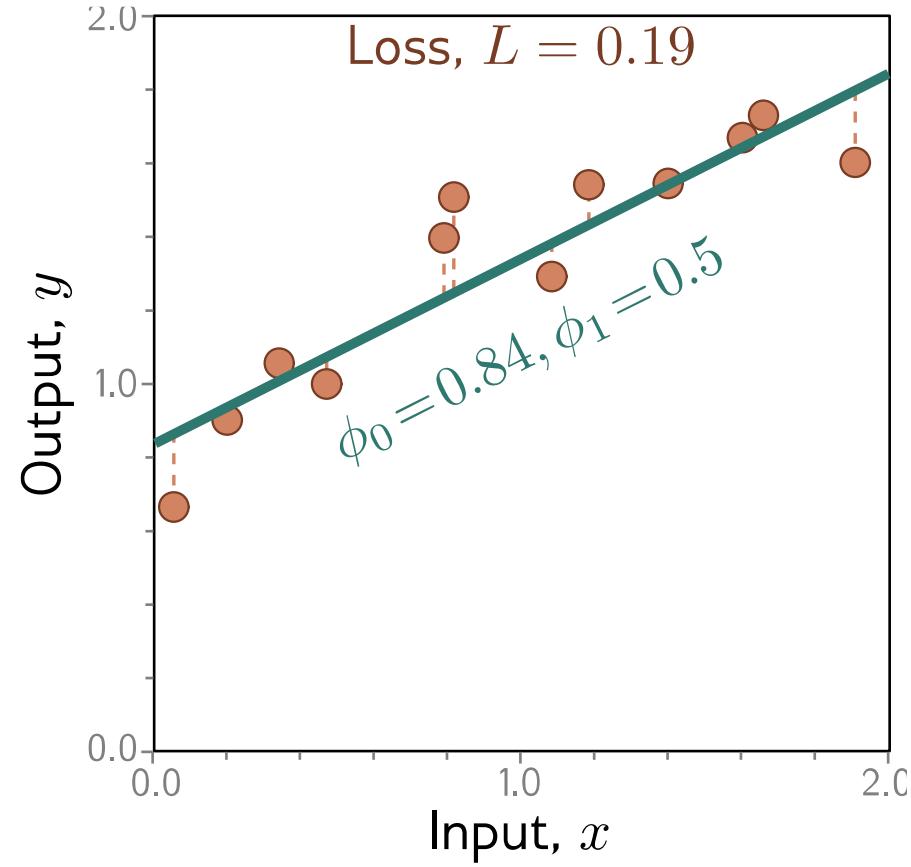
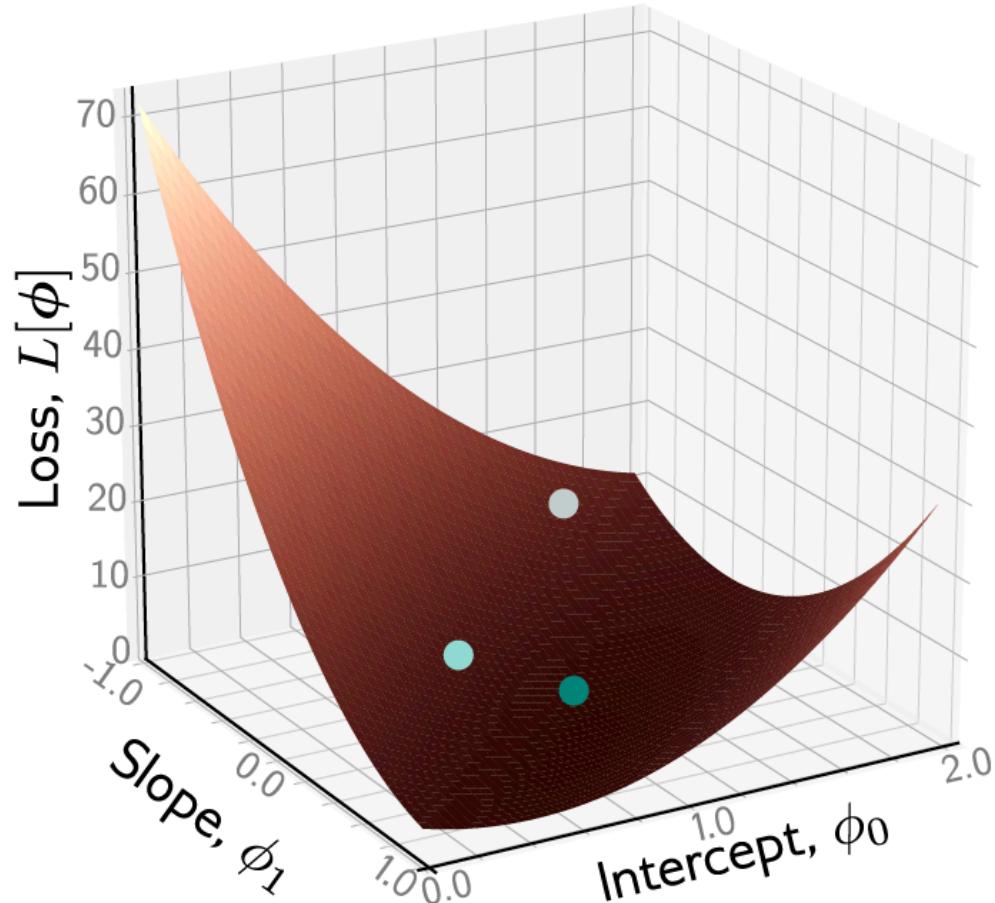
# Example: 1D Linear regression loss function



# Example: 1D Linear regression loss function

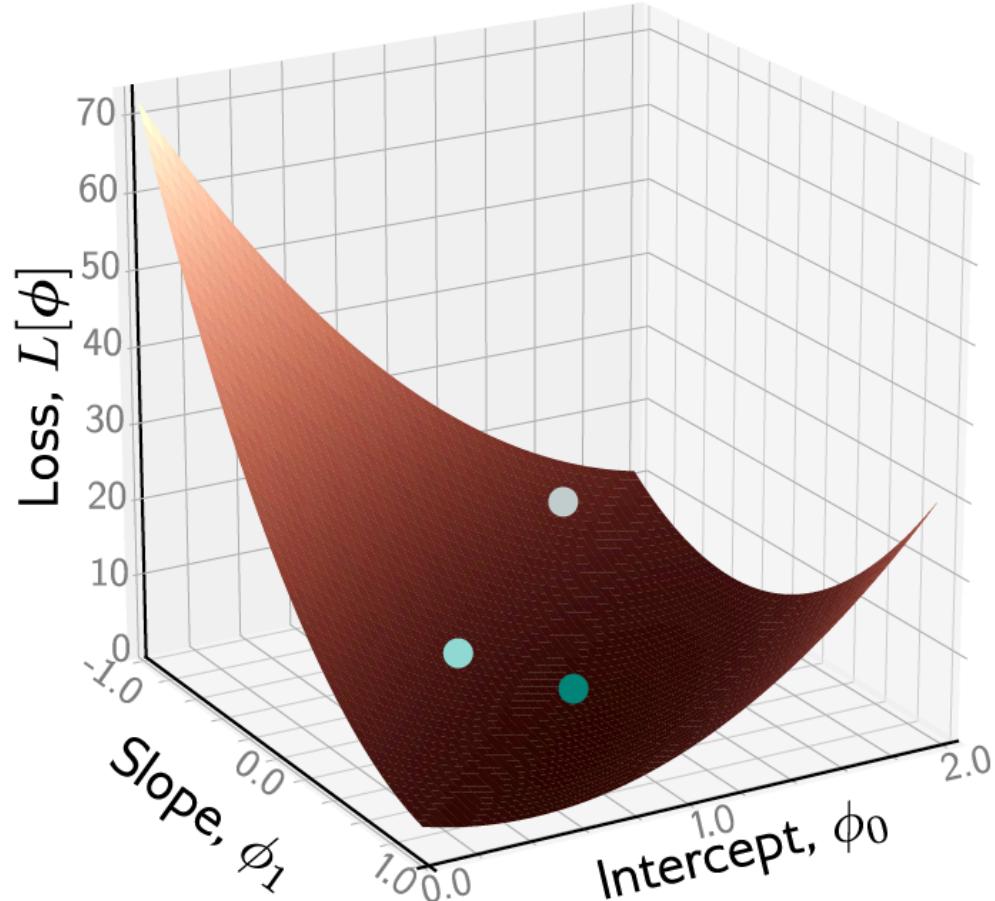


# Example: 1D Linear regression loss function

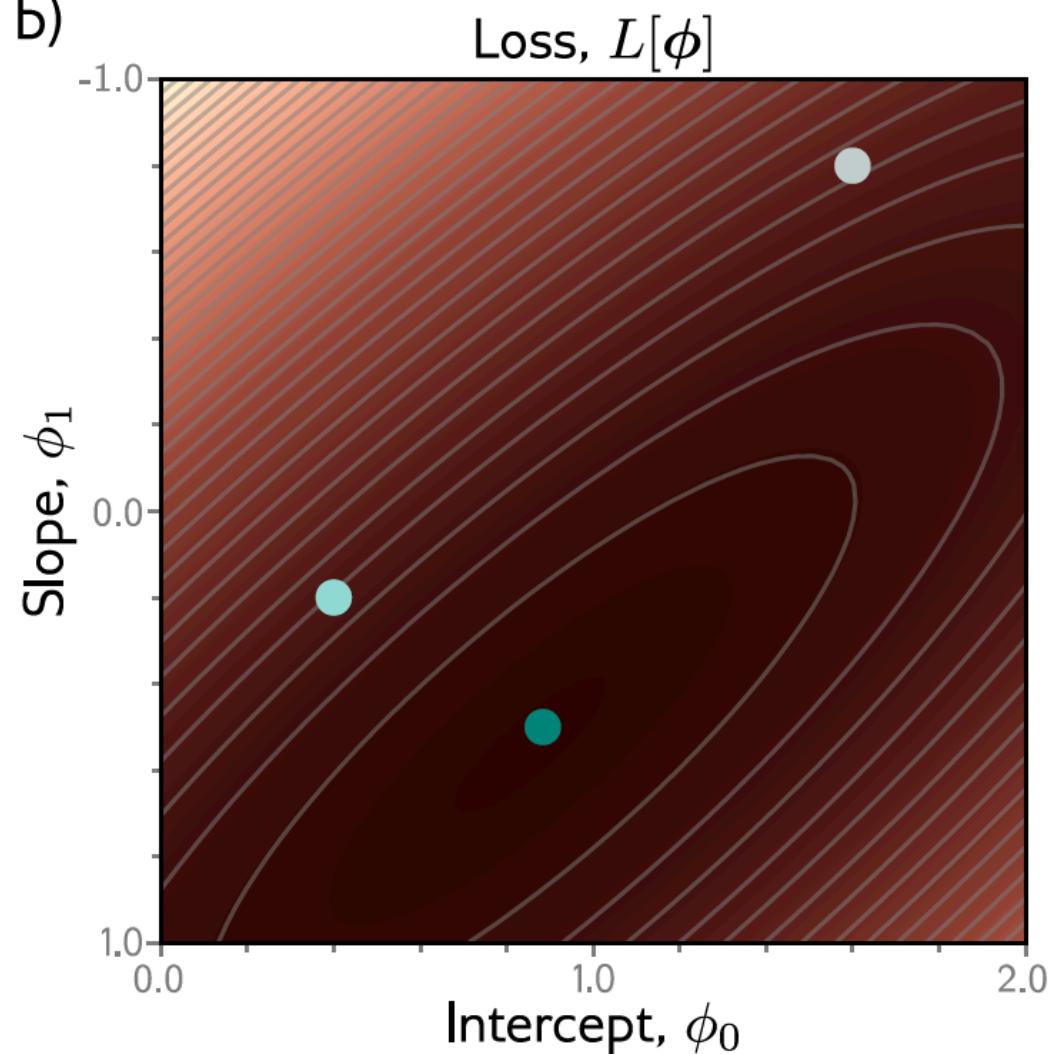


# Example: 1D Linear regression loss function

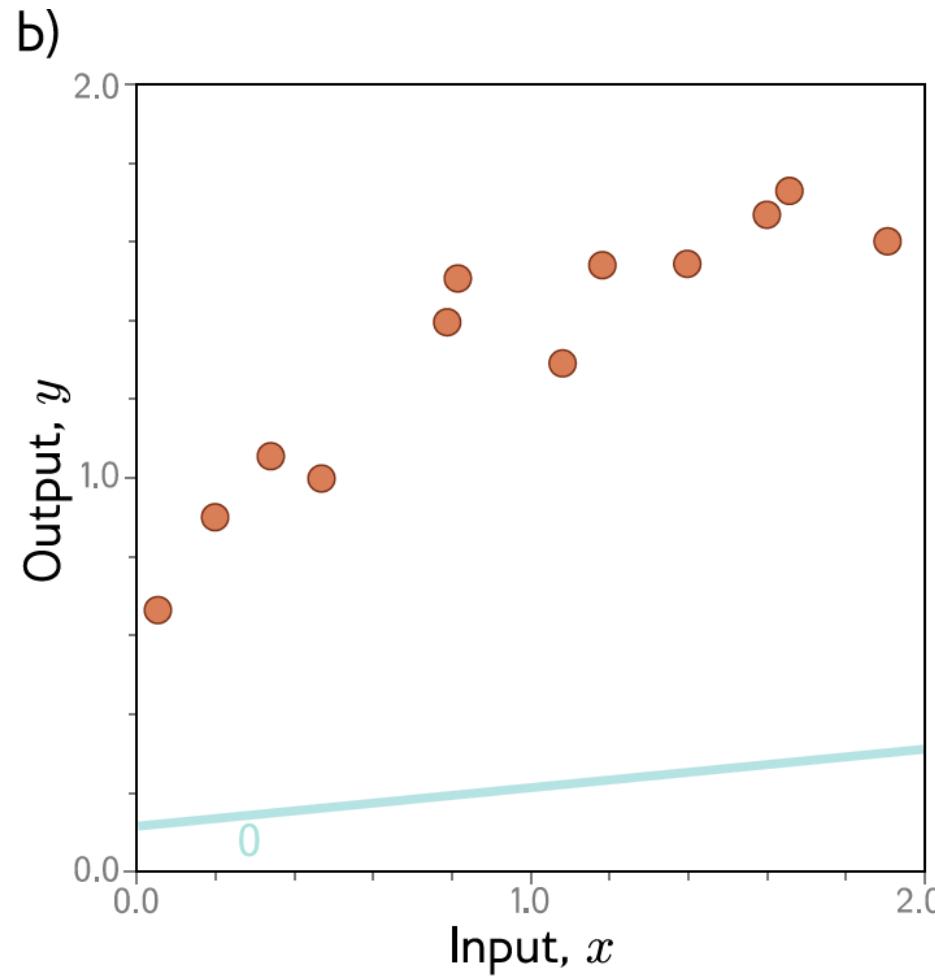
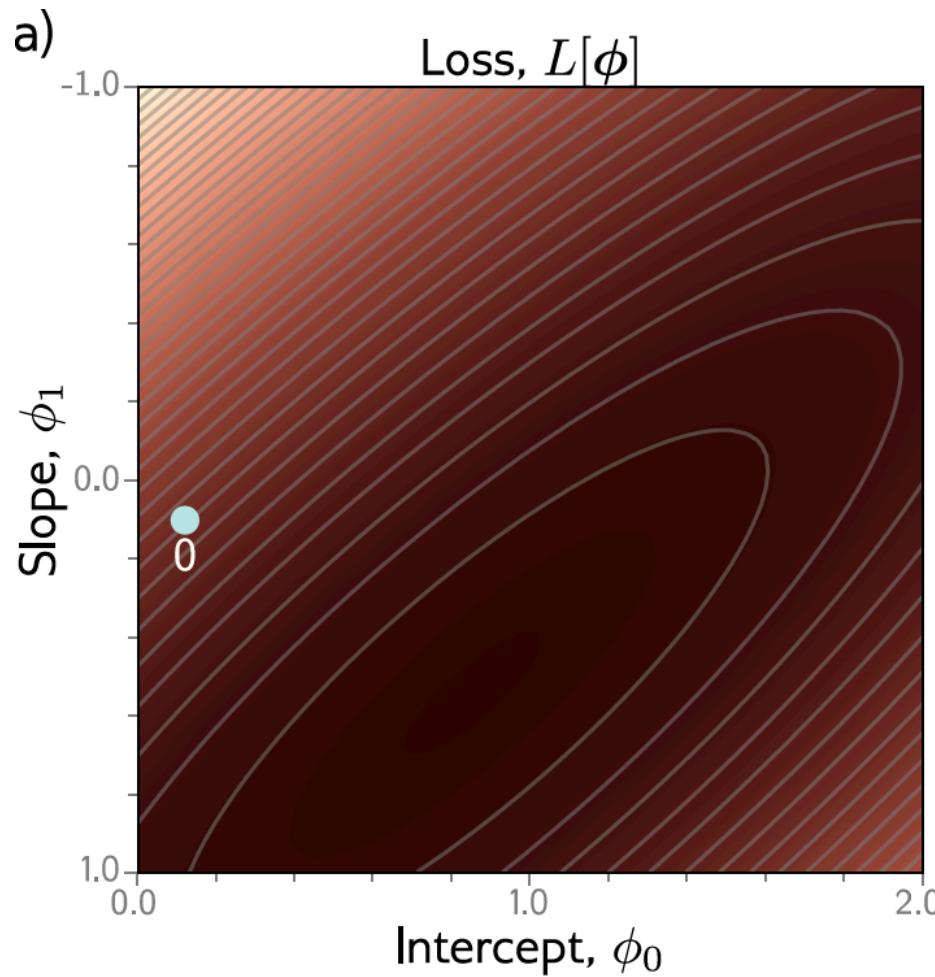
a)



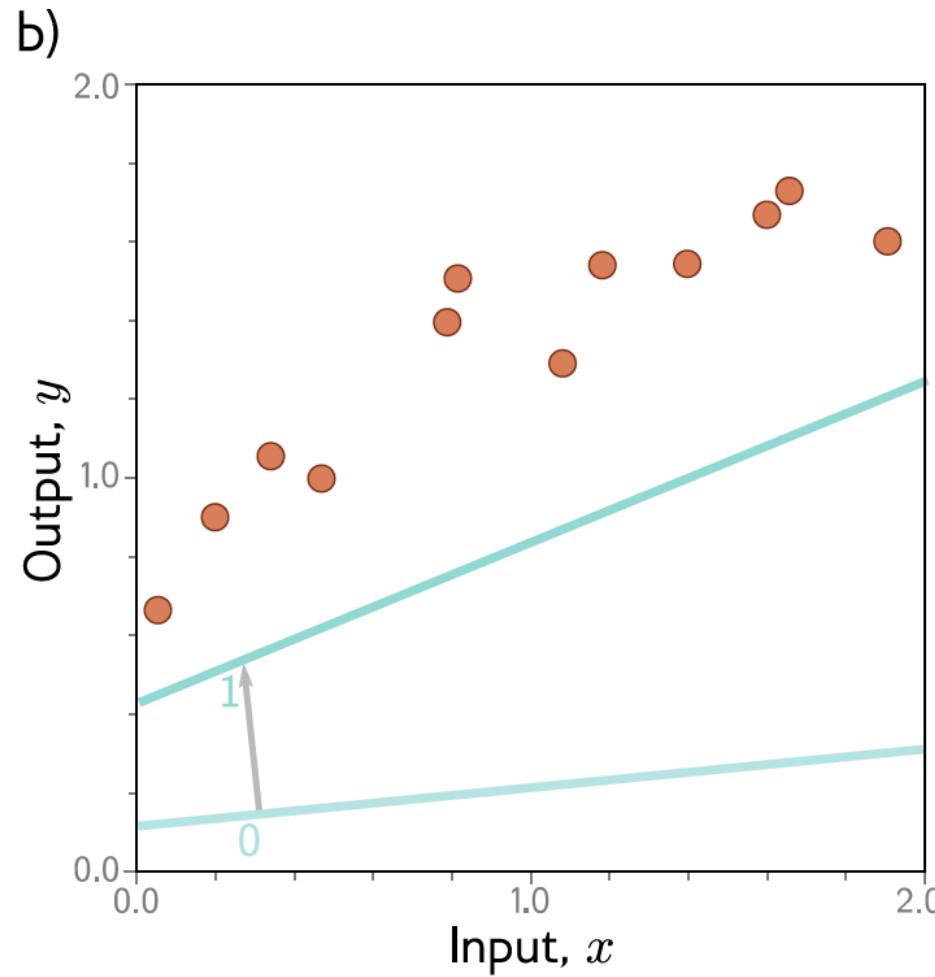
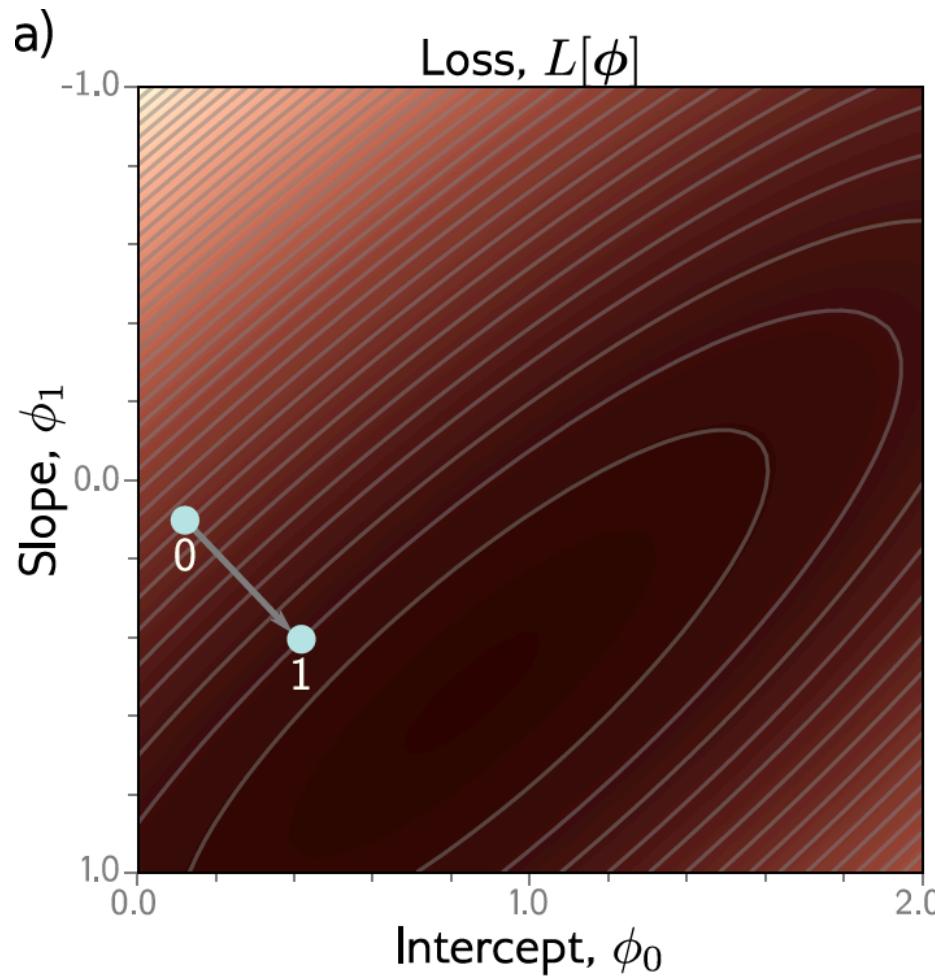
b)



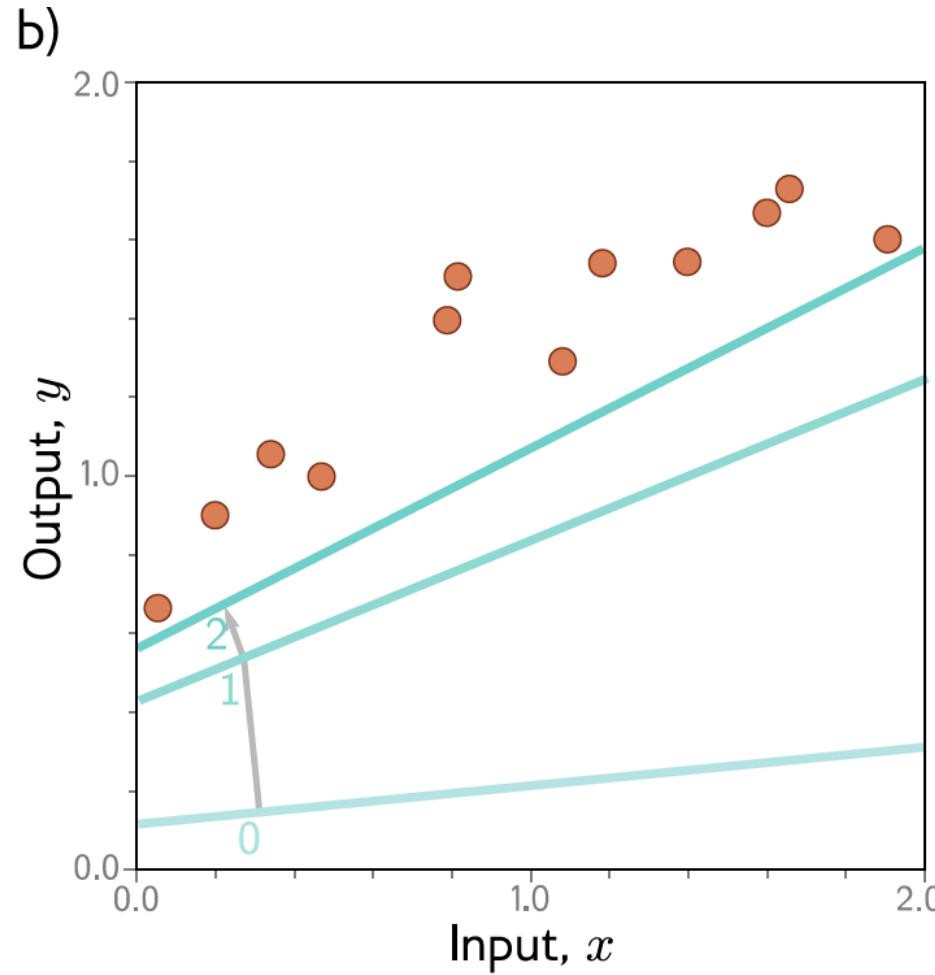
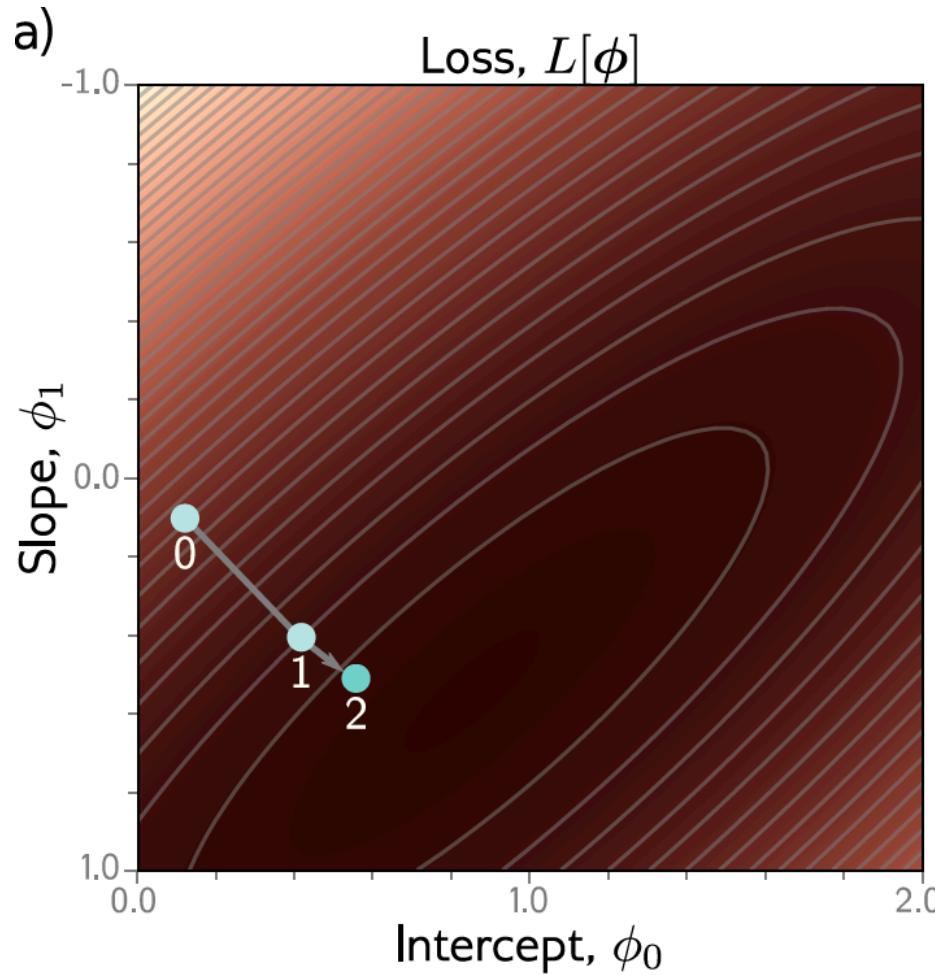
# Example: 1D Linear regression training



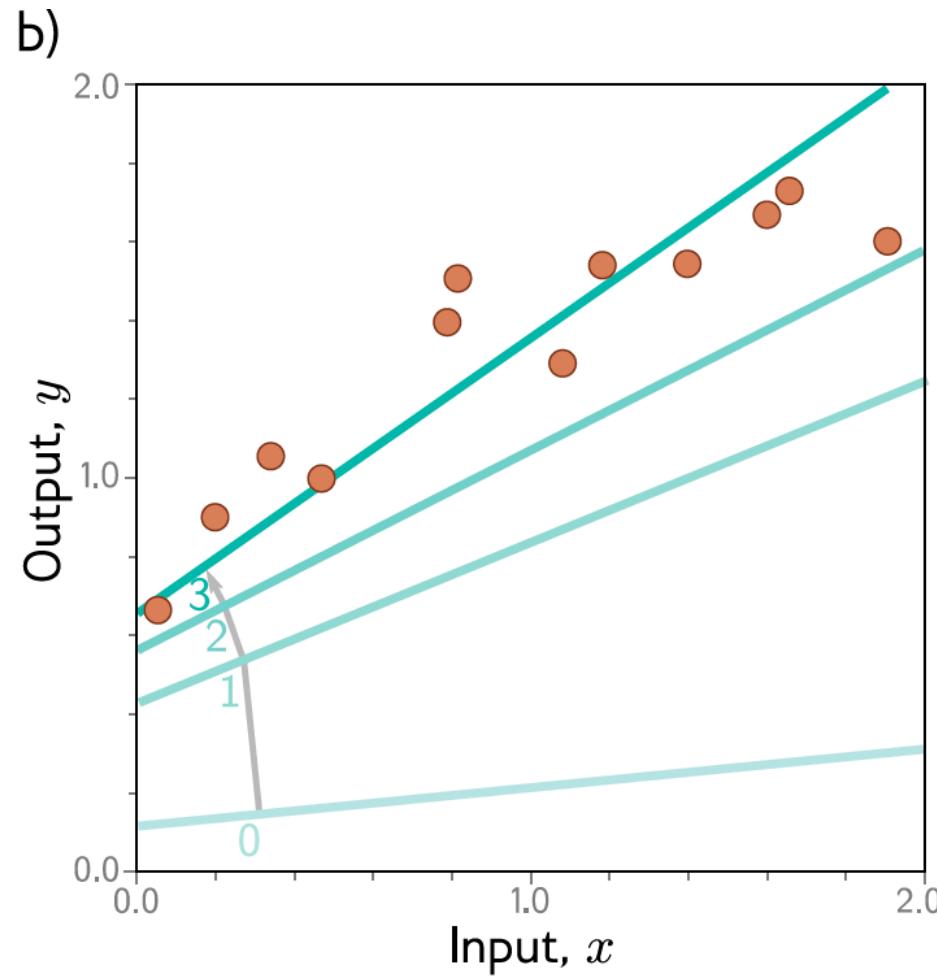
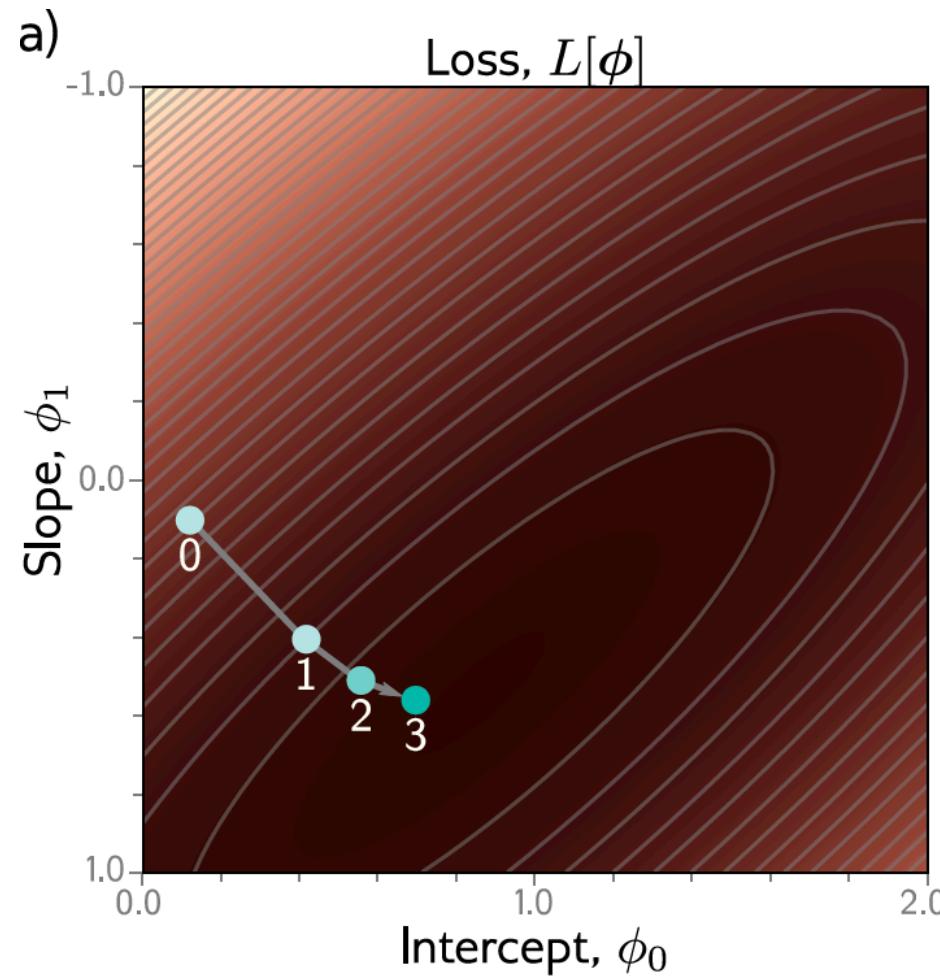
# Example: 1D Linear regression training



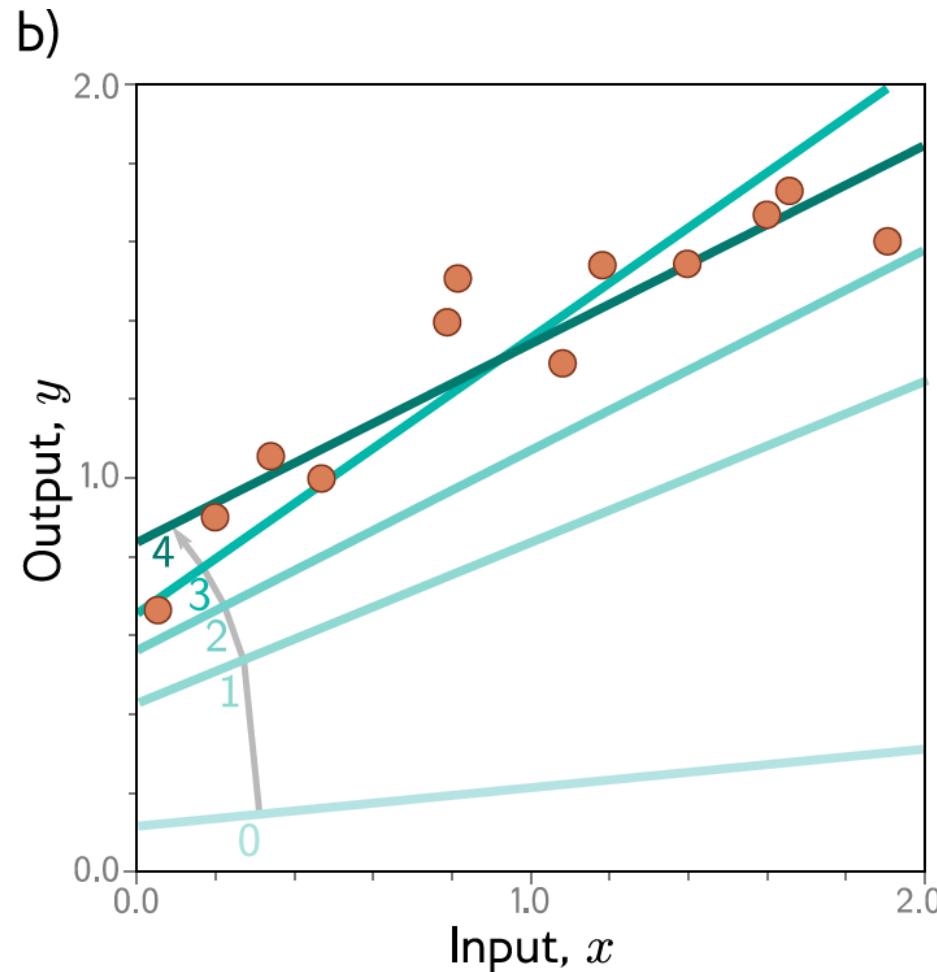
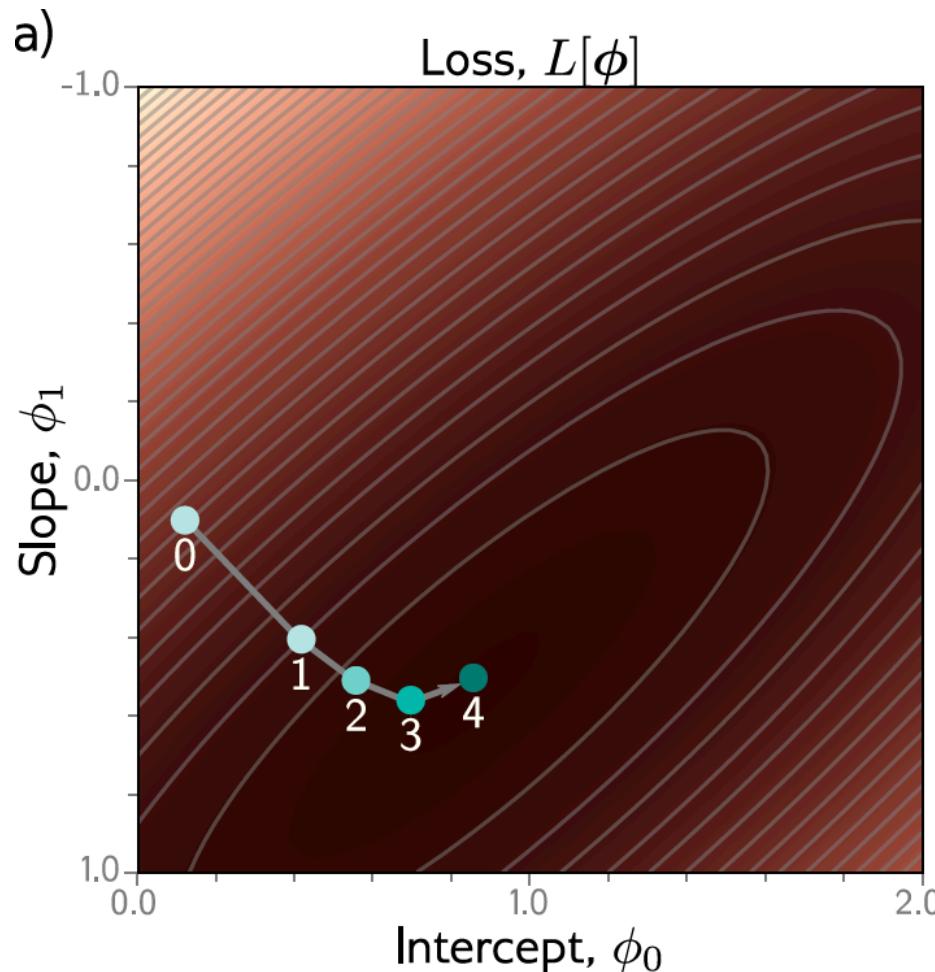
# Example: 1D Linear regression training



# Example: 1D Linear regression training



# Example: 1D Linear regression training



This technique is known as **gradient descent**

# Artificial intelligence

Machine learning

Supervised  
learning

Unsupervised  
learning

Reinforcement  
learning

Deep learning

# Regression

Real world input

6000 square feet,  
4 bedrooms,  
previously sold for  
\$235K in 2005,  
1 parking spot.

Model  
input

$$\begin{bmatrix} 6000 \\ 4 \\ 235 \\ 2005 \\ 1 \end{bmatrix}$$

Model



Model  
output

$$[340]$$

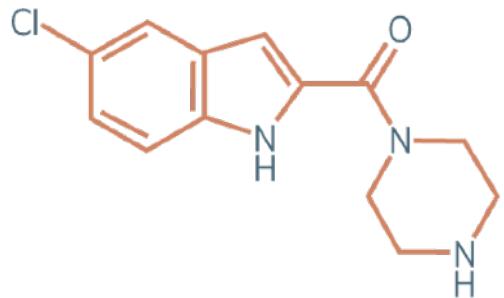
Real world output

Predicted price  
is \$340k

- Univariate regression problem (one output, real value)
- Fully connected network

# Graph regression

Real world input



Model  
input

$$\begin{bmatrix} 1 \\ 0 \\ 1 \\ \vdots \\ 17 \\ 1 \\ 1 \\ \vdots \end{bmatrix}$$

Model



Model  
output

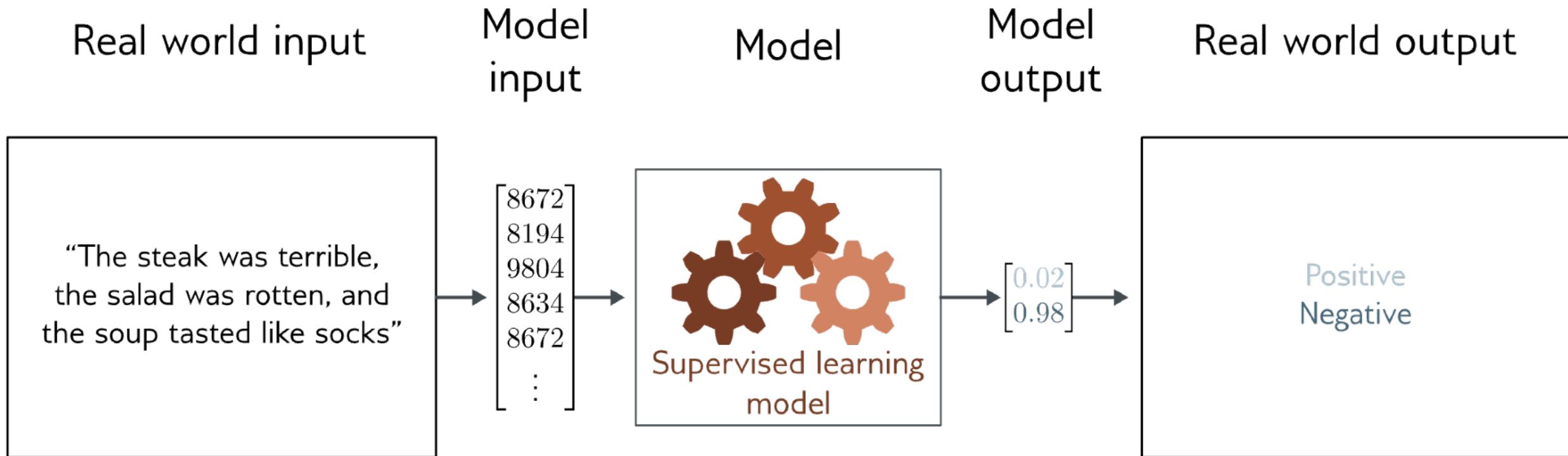
$$\begin{bmatrix} -12.9 \\ 56.4 \end{bmatrix}$$

Real world output

Freezing point  
is  $-12.9^{\circ}\text{C}$   
Boiling point  
is  $56.4^{\circ}\text{C}$

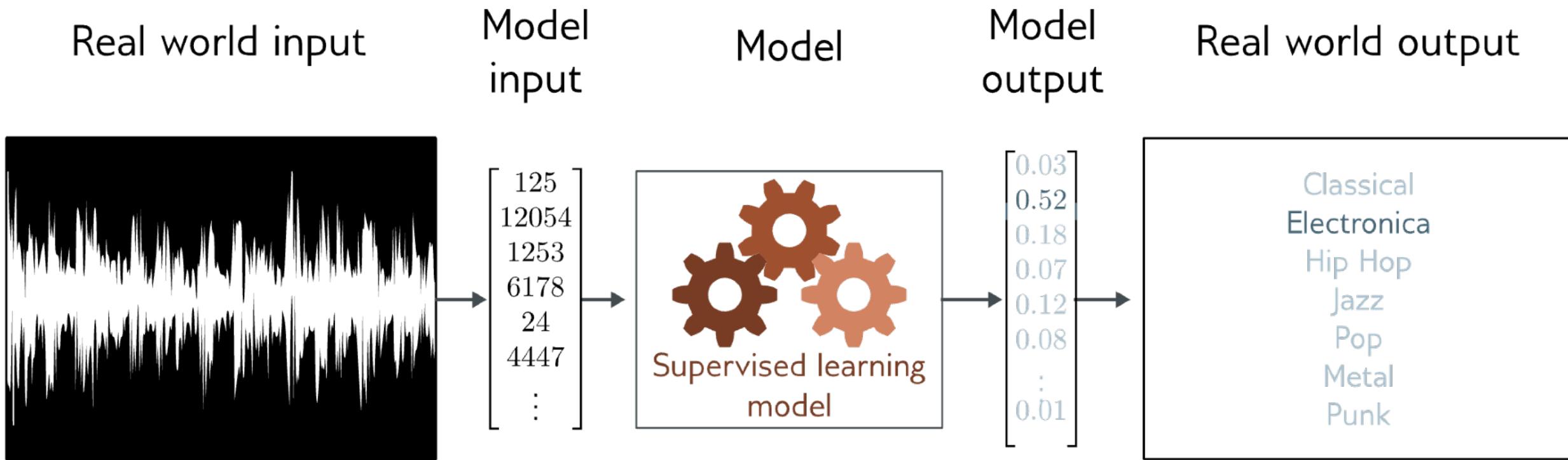
- Multivariate regression problem (>1 output, real value)
- Graph neural network

# Text classification



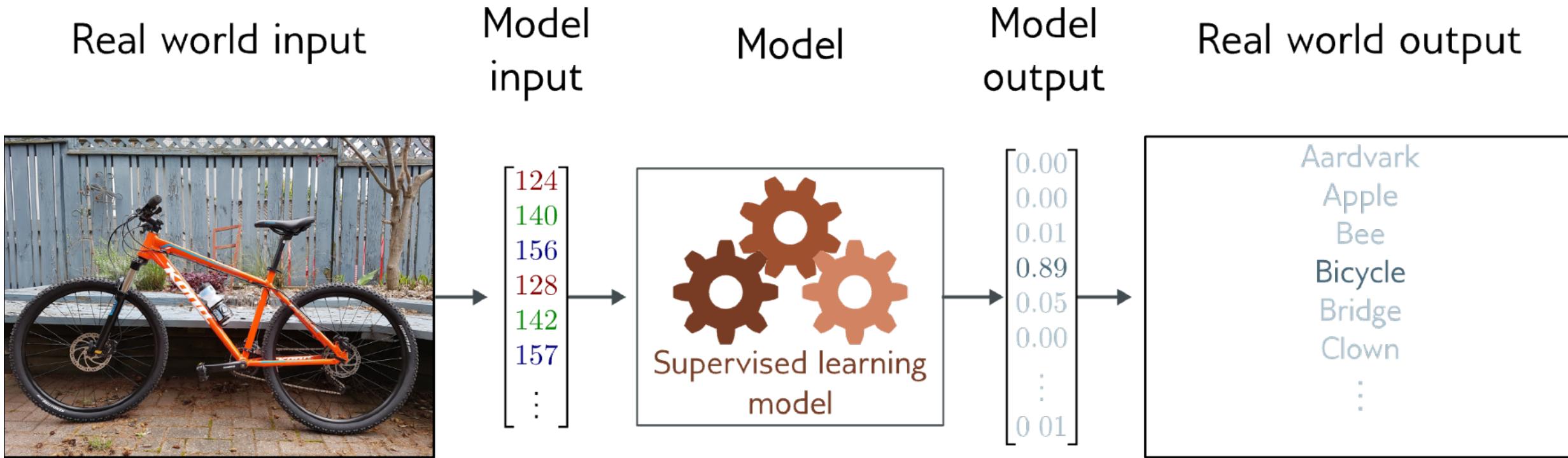
- Binary classification problem (two discrete classes)
- Transformer network

# Music genre classification



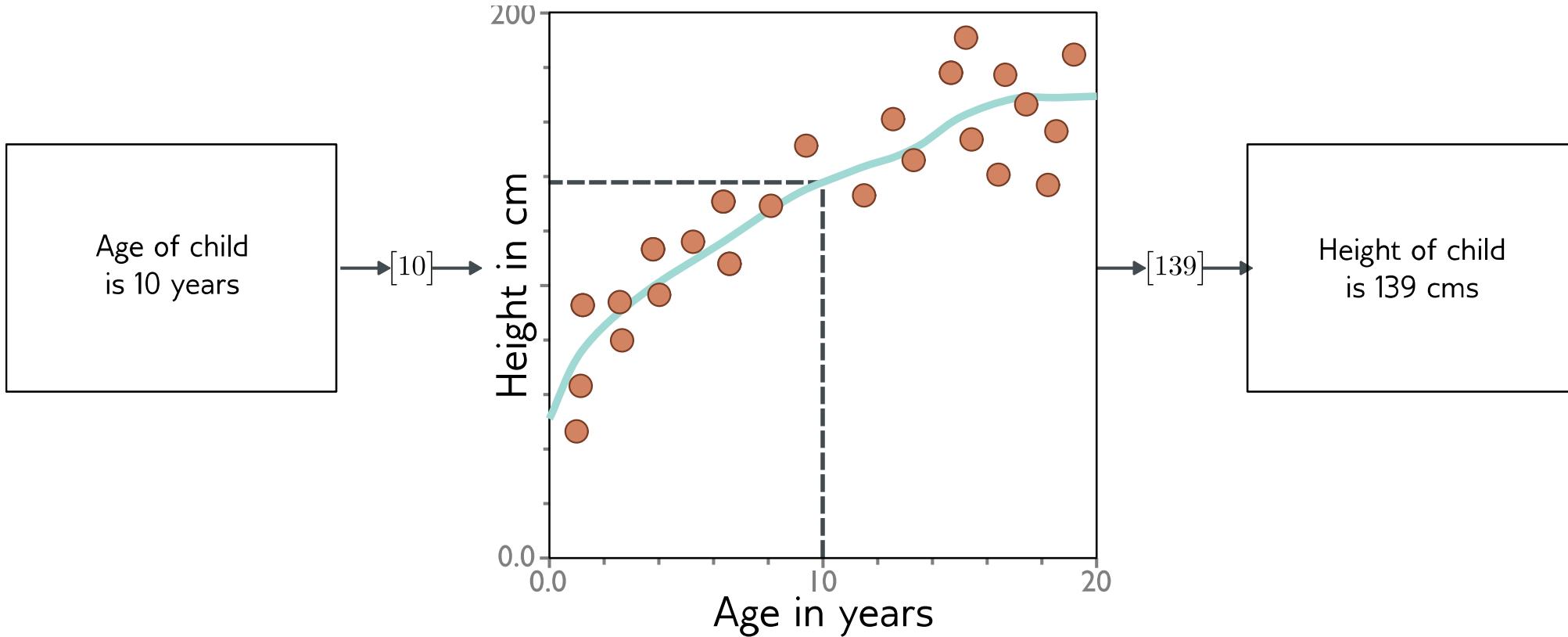
- Multiclass classification problem (discrete classes, >2 possible values)
- Recurrent neural network (RNN)

# Image classification



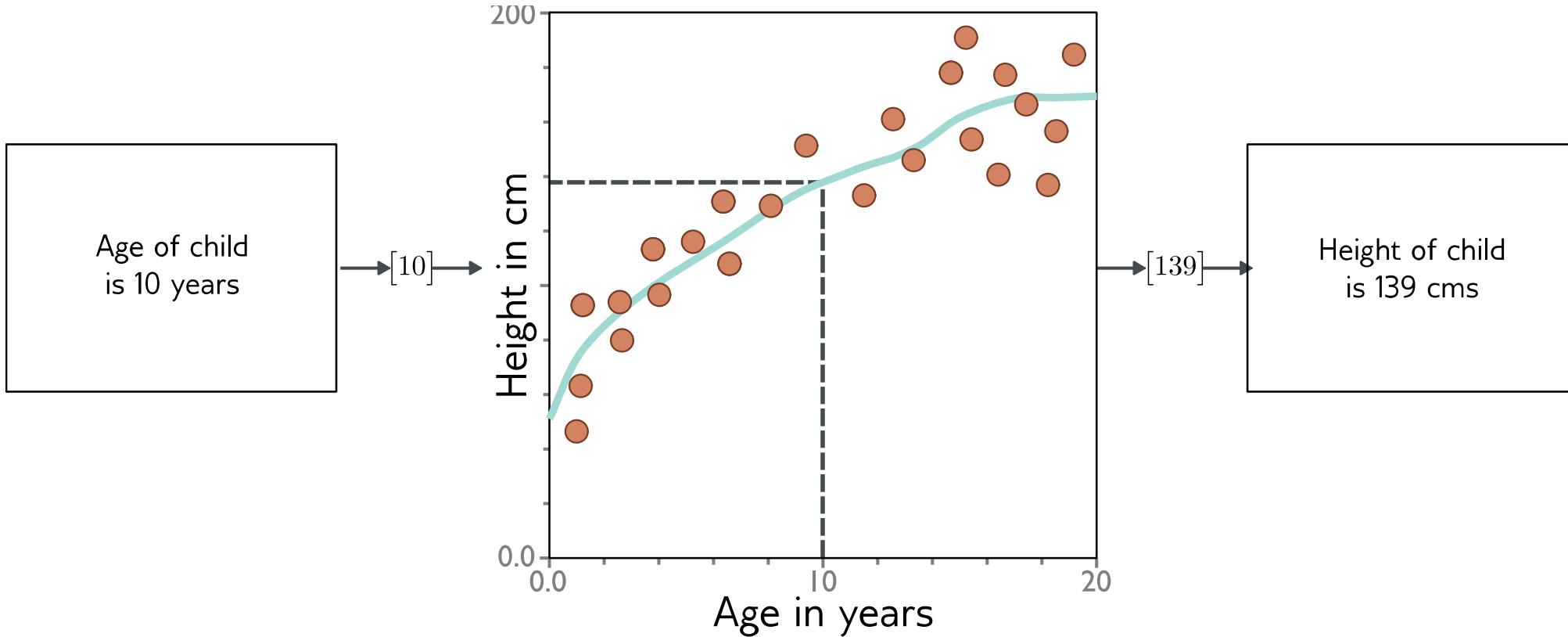
- Multiclass classification problem (discrete classes, >2 possible classes)
- Convolutional network

# What is a supervised learning model?



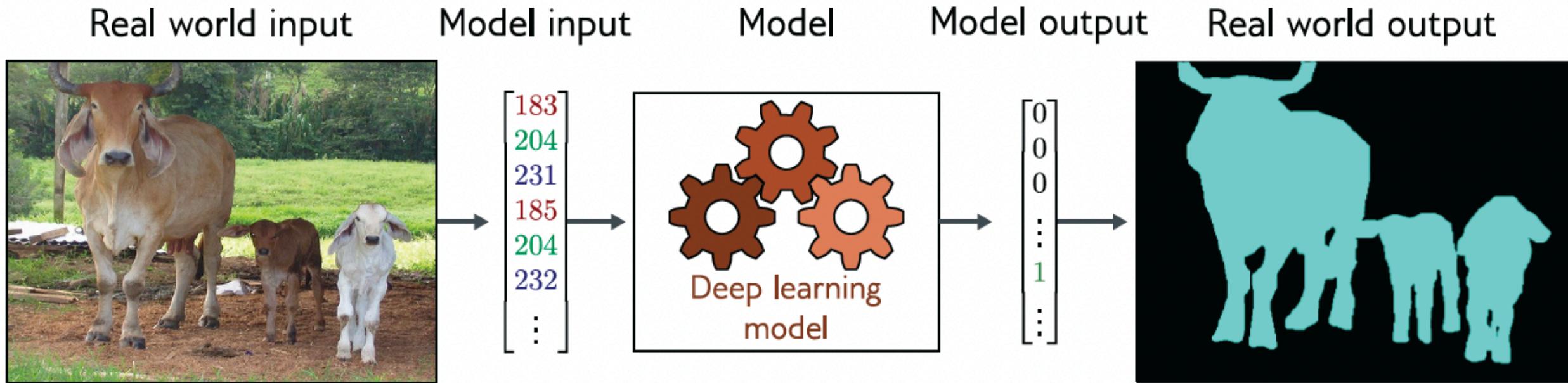
- An equation relating input (age) to output (height)
- Search through family of possible equations to find one that fits training data well

# What is a supervised learning model?



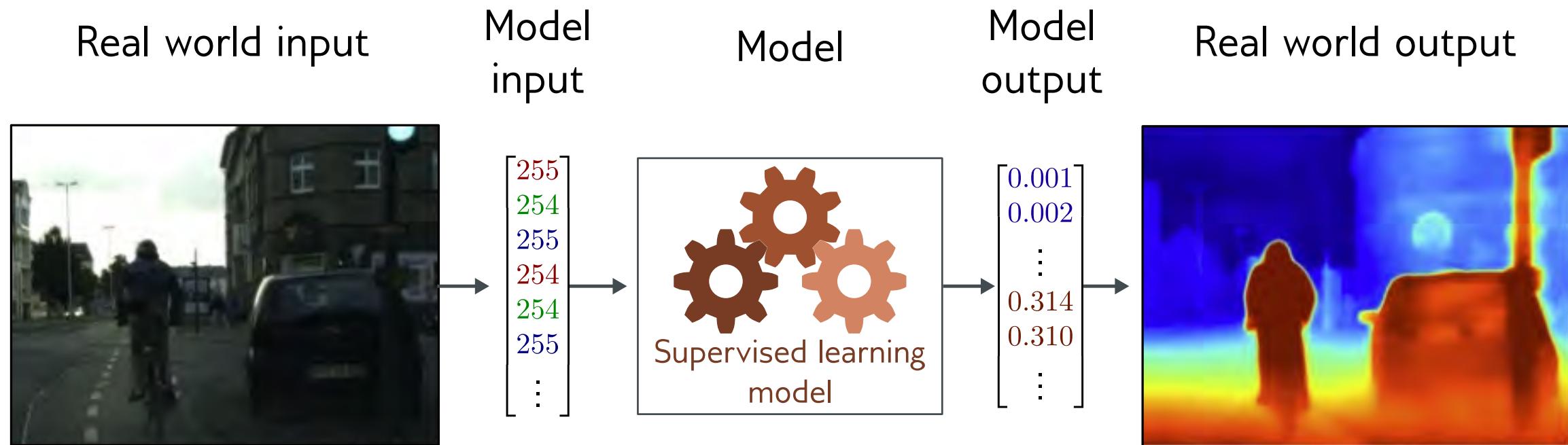
- Deep neural networks are just a very flexible family of equations
- Fitting deep neural networks = “Deep Learning”

# Image segmentation



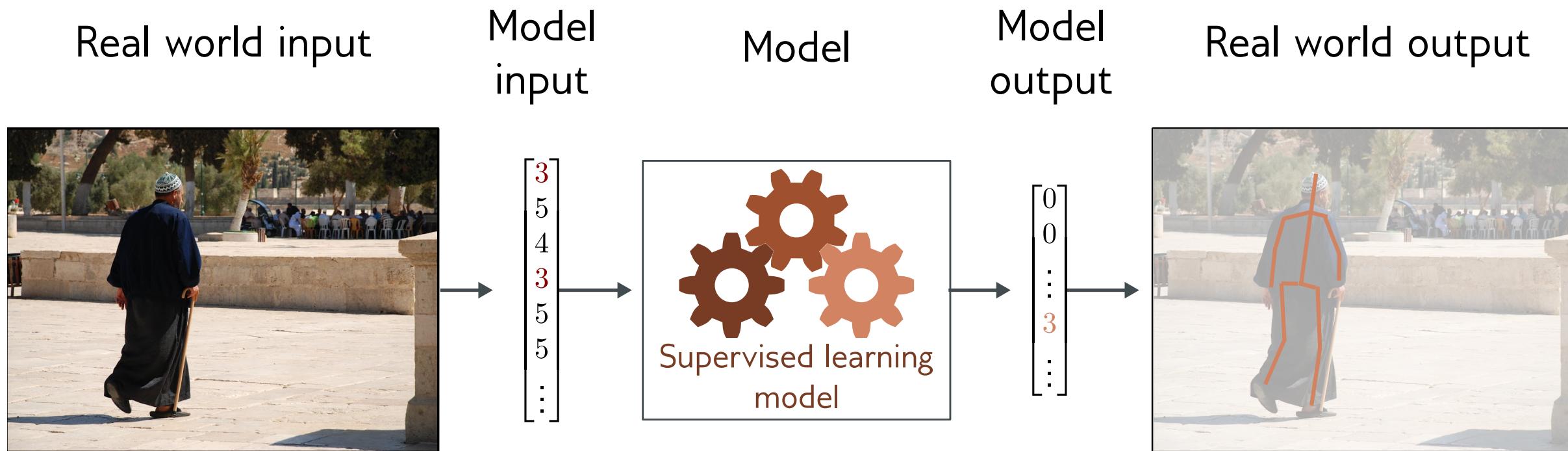
- Multivariate binary classification problem (many outputs, two discrete classes)
- Convolutional encoder-decoder network

# Depth estimation



- Multivariate regression problem (many outputs, continuous)
- Convolutional encoder-decoder network

# Pose estimation

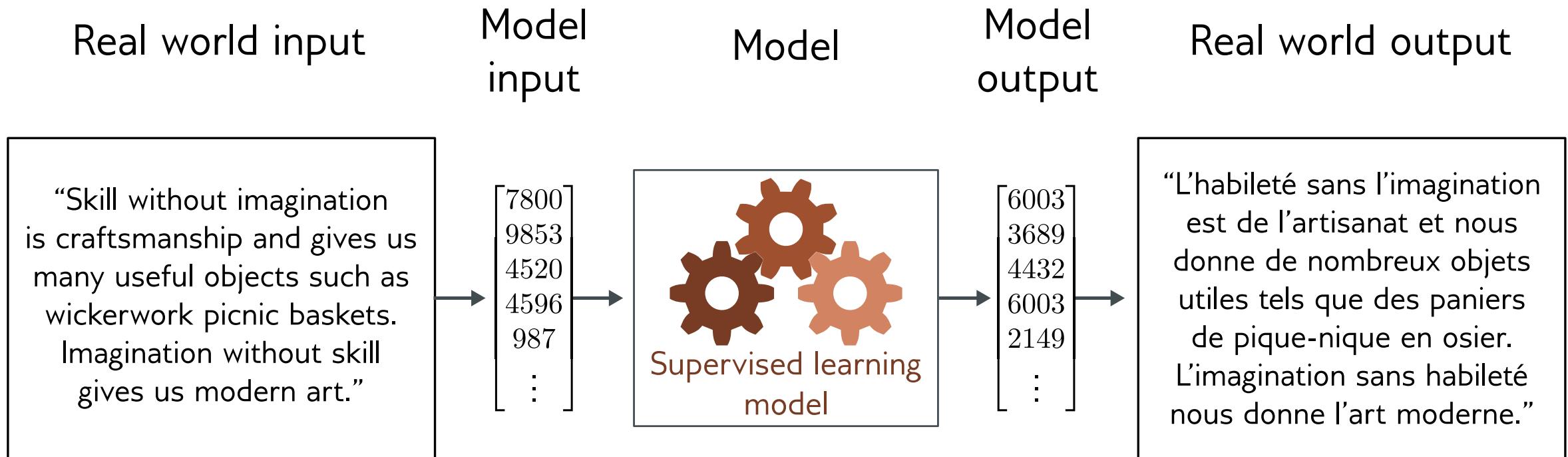


- Multivariate regression problem (many outputs, continuous)
- Convolutional encoder-decoder network

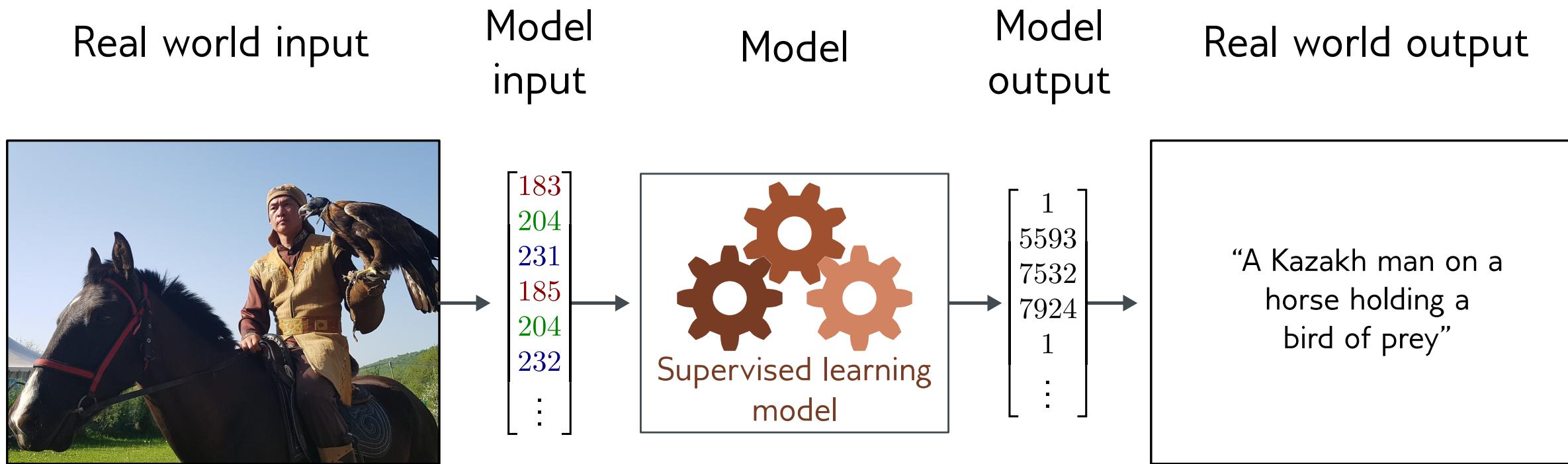
# Terms

- Regression = continuous numbers as output
- Classification = discrete classes as output
- Two class and multiclass classification treated differently
- Univariate = one output
- Multivariate = more than one output

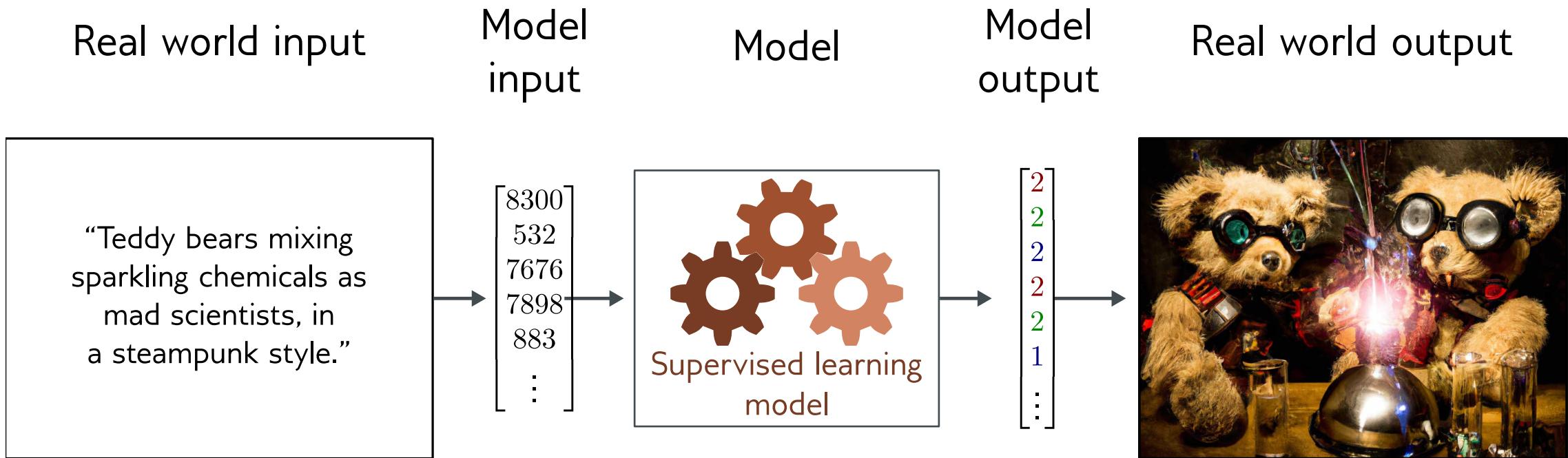
# Translation



# Image captioning



# Image generation from text



# What do these examples have in common?

- Very complex relationship between input and output
- Sometimes may be many possible valid answers
- But outputs (and sometimes inputs) obey rules

“A Kazakh man on a horse holding a bird of prey”

Language obeys grammatical rules



Natural images also have “rules”

# Idea

- Learn the “grammar” of the data from unlabeled examples
- Can use a gargantuan amount of data to do this (as unlabeled)
- Make the supervised learning task easier by having a lot of knowledge of possible outputs

# Artificial intelligence

Machine learning

Supervised  
learning

Unsupervised  
learning

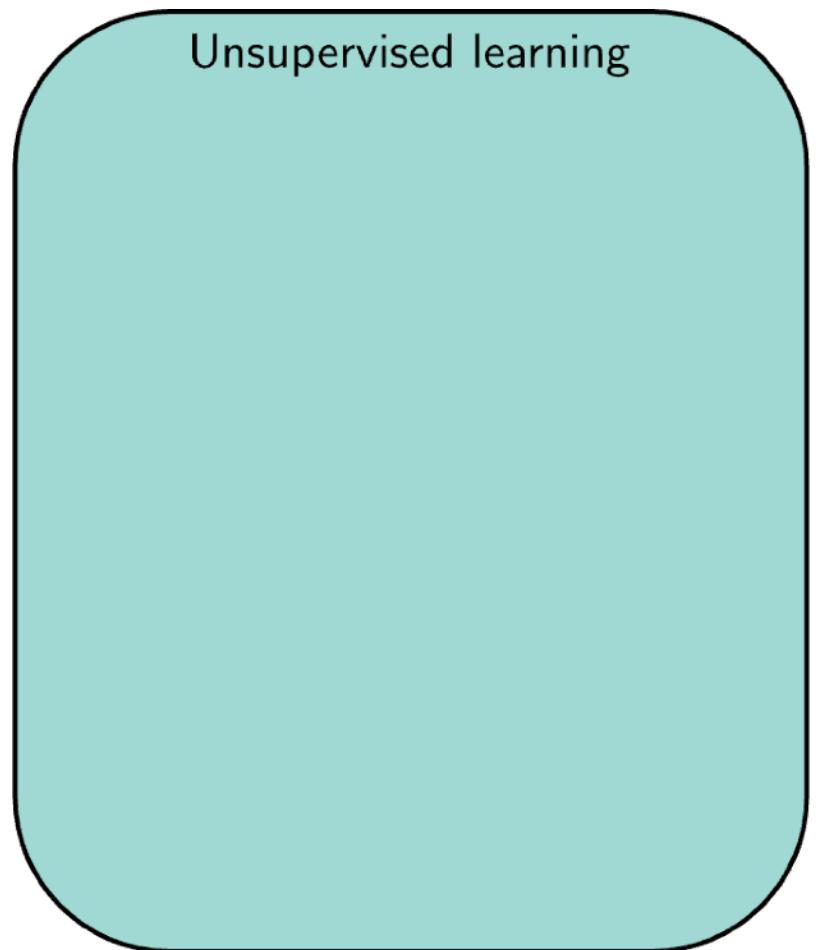
Reinforcement  
learning

Deep learning



# Unsupervised Learning

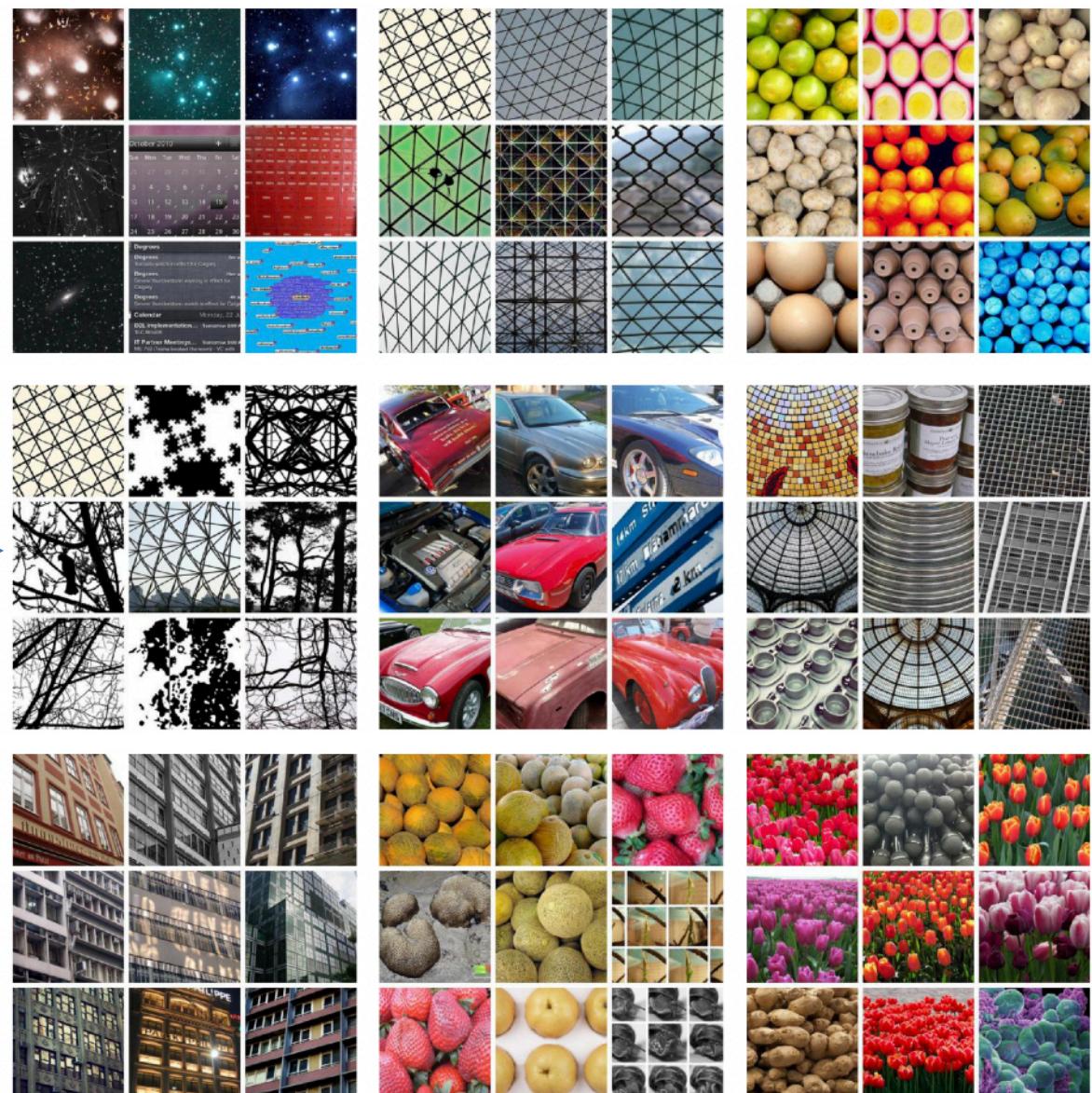
- Learning about a dataset without labels
  - Clustering
  - Finding outliers
  - Generating new examples
  - Filling in missing data



Unsupervised learning



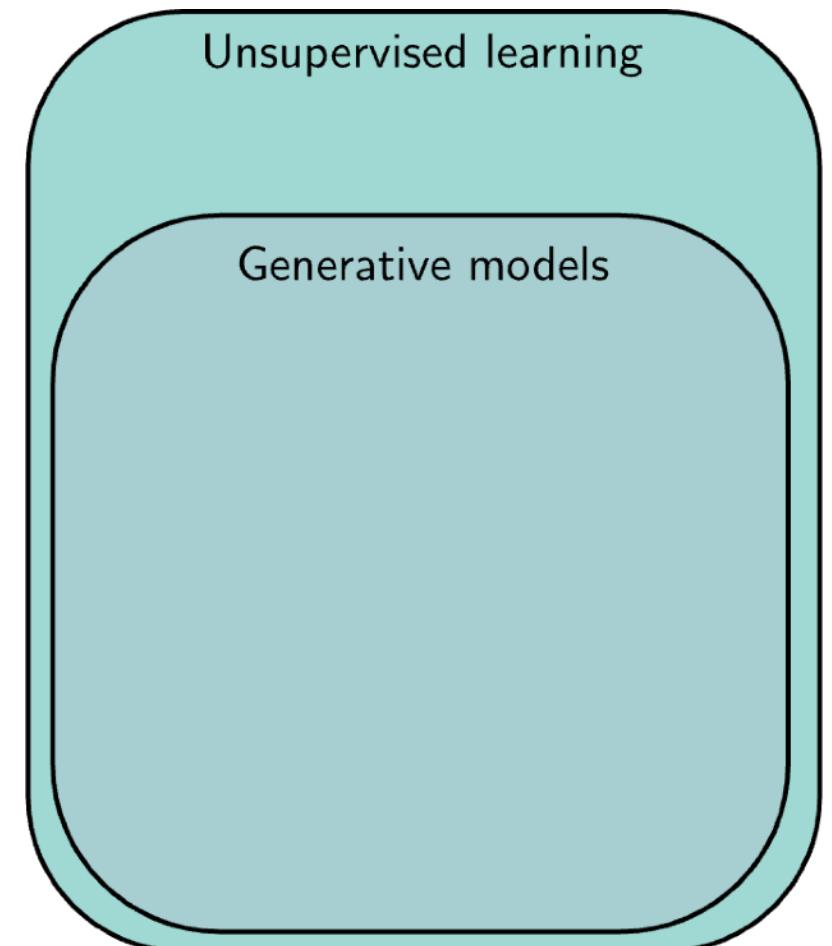
DeepCluster: Deep Clustering for Unsupervised Learning of Visual Features (Caron et al., 2018)



DeepCluster: Deep Clustering for Unsupervised Learning of Visual Features (Caron et al., 2018)

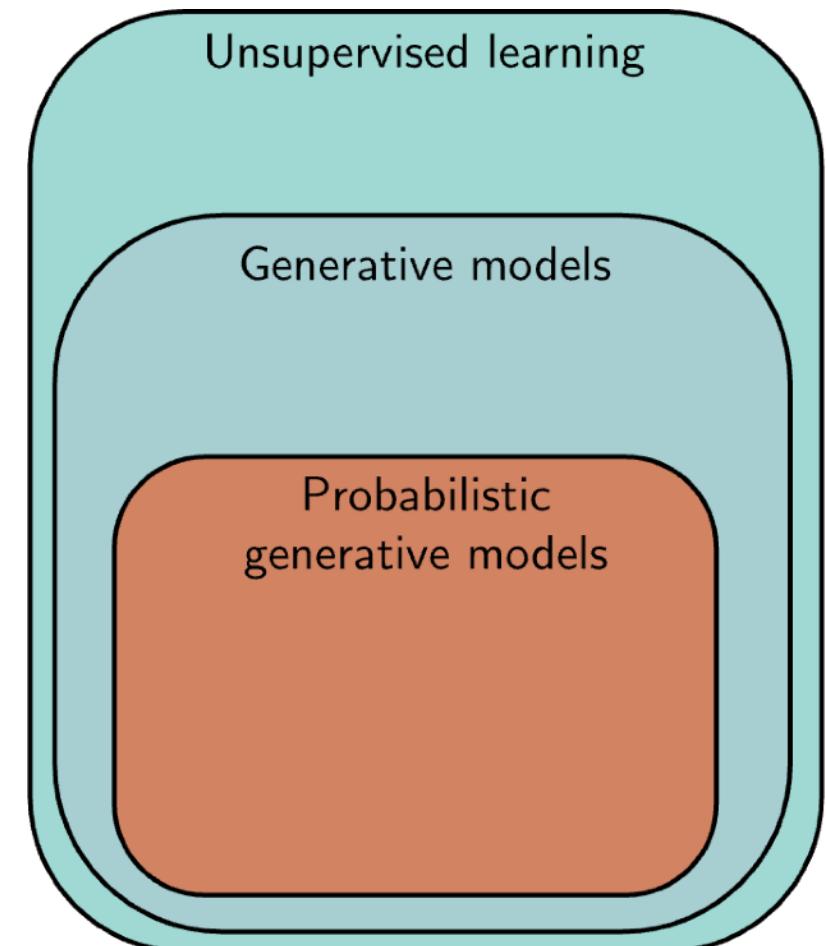
# Unsupervised Learning

- Learning about a dataset without labels
  - e.g., clustering
- Generative models can create examples
  - e.g., generative adversarial networks



# Unsupervised Learning

- Learning about a dataset without labels
  - e.g., clustering
- Generative models can create examples
  - e.g., generative adversarial networks
- PGMs learn distribution over data
  - e.g., variational autoencoders,
  - e.g., normalizing flows,
  - e.g., diffusion models



# Generative models



National Geographic  
Domestic cat



w Wikipedia  
Cat - Wikipedia



The Guardian  
pet guru Yuki Hattori explain | ...



Britannica  
Cat | Breeds & Facts | Britannica



The Spruce Pets  
Tabby Cat: Breed Profile ...



Britannica  
Cat | Breeds & Facts | Britannica



w Wikipedia  
Cat intelligence - Wikipedia



Smithsonian Magazine  
Cats React to 'Baby Talk' From Their ...



Alley Cat Allies  
The Natural History of Domestic Cats ...



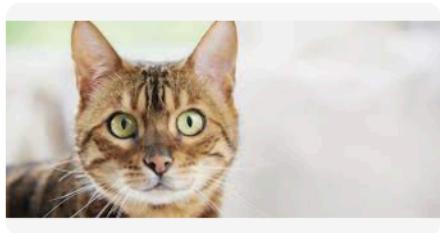
The New York Times  
How the Cat Gets Its Stripe...



Country Living Magazine  
Friendliest Cat Breeds That ...



Freepik  
Cat Images - Free D...



BBC Science Focus  
What's the longest a cat can live for ...



National Geographic  
Domestic cat



DK Find Out!  
Cat Facts for Kids | What is a Cat | DK ...



The Spruce Pets  
Ragdoll Cat: Breed Profile ...



Good Housekeeping  
25 Best Cat Instagram Caption...



Daily Paws  
17 Long-Haired Cat Breeds to Swoon...



Unsplash  
500+ Domestic Cat ...



Four Paws  
A Cat's Personality - FOUR PAWS ...

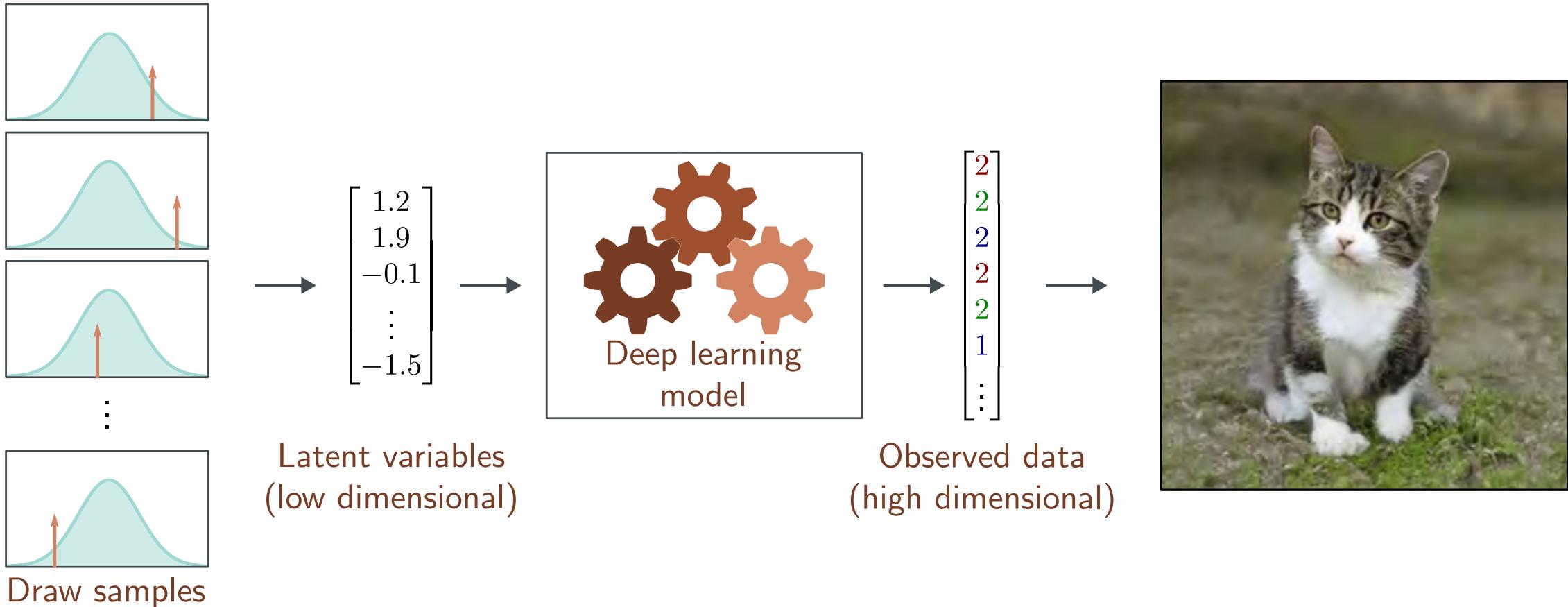


The Guardian  
pet guru Yuki Hattori explain | ...

# Generative models



# Latent variables



# Artificial intelligence

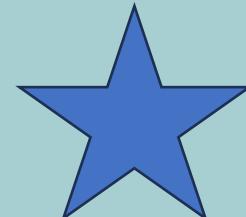
Machine learning

Supervised  
learning

Unsupervised  
learning

Reinforcement  
learning

Deep learning



# Reinforcement learning

- A set of **states**
- A set of **actions**
- A set of **rewards**
- Goal: take actions to change the state so that you receive rewards
- You don't receive any data – you have to explore the environment yourself to gather data as you go

# Example: chess

- States are valid states of the chess board
- Actions at a given time are valid possible moves
- Positive rewards for taking pieces, negative rewards for losing them

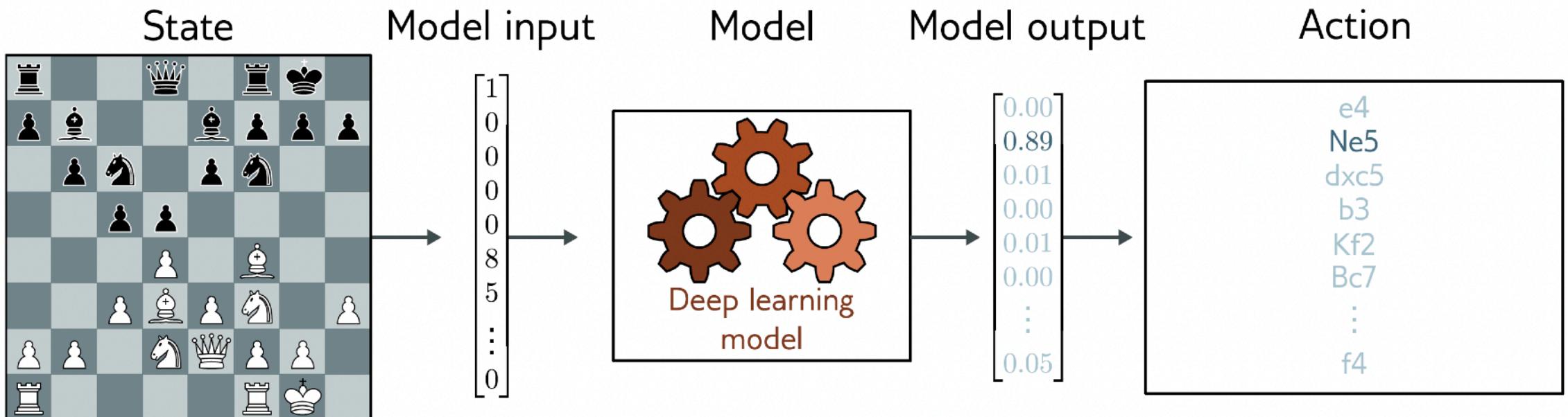


Action

e4  
Ne5  
dxc5  
b3  
Kf2  
Bc7  
⋮  
f4

# Example: chess

- States are valid states of the chess board
  - Actions at a given time are valid possible moves
  - Positive rewards for taking pieces, negative rewards for losing them



# Why is this difficult?

- Stochastic
  - Make the same move twice, the opponent might not do the same thing
  - Rewards also stochastic (opponent does or doesn't take your piece)
- Temporal credit assignment problem
  - Did we get the reward because of this move? Or because we made good tactical decisions somewhere in the past?
- Exploration-exploitation trade-off
  - If we found a good opening, should we use this?
  - Or should we try other things, hoping for something better?