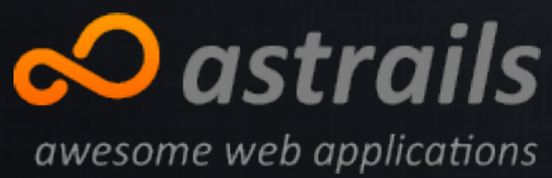
Machine Learning



Boris Nadion
boris@astrails.com
@borisnadion

@borisnadion
boris@astrails.com

astrails

http://astrails.com

awesome web and mobile apps

since 2005

Battle of the Bots: How AI Is Taking Over the World of Cybersecurity

BY EDD GENT ON NOV 09, 2016 | ARTIFICIAL INTELLIGENCE, FEATURED, TECH

2747 2

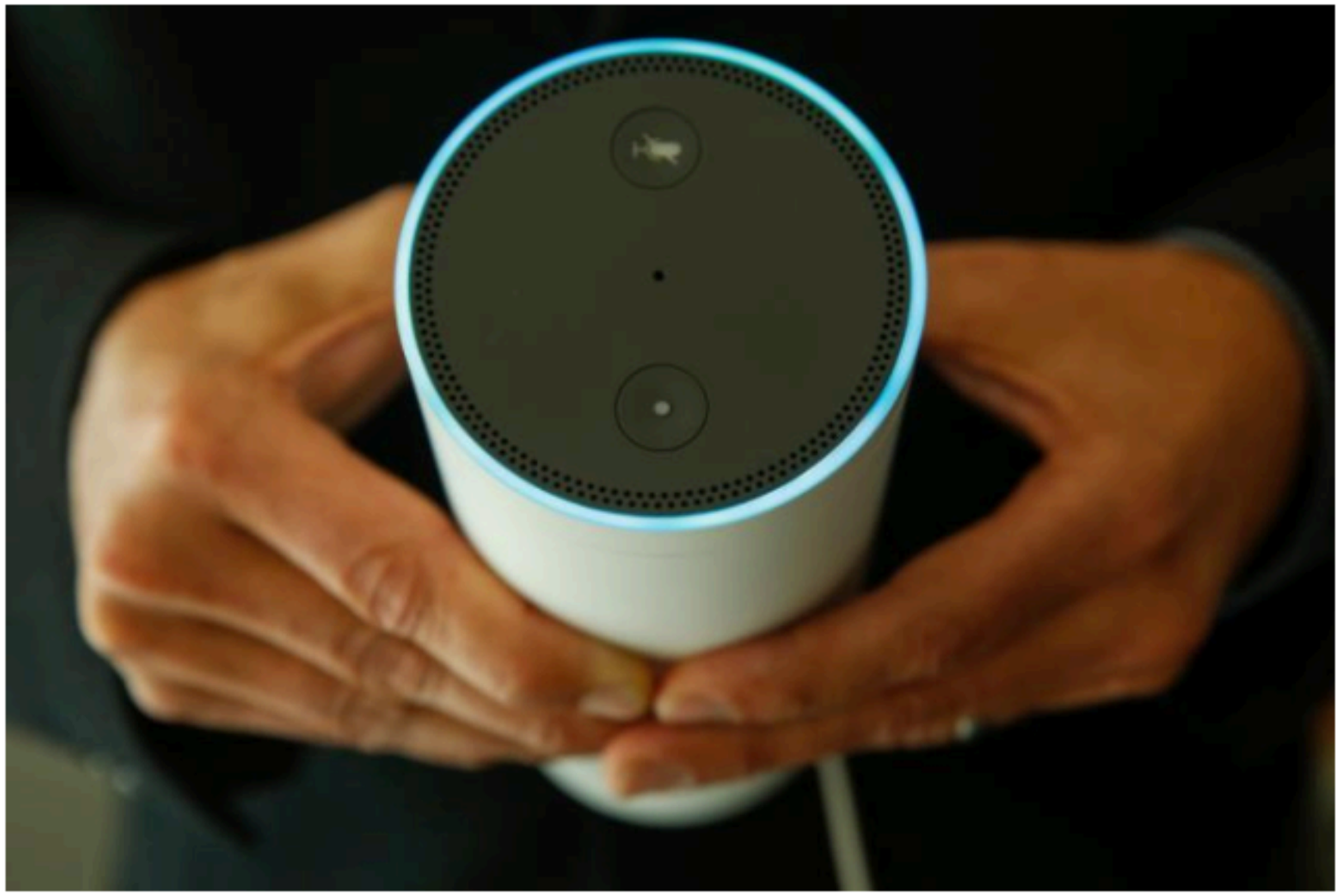
186 348 6 7

AI Revolutionizes Industries, not World Domination

By **John N** - November 10, 2016

Tatiana Shepeleva | Shutterstock.com

Marketing faces death by algorithm unless it finds a new code



AI-powered devices, such as smart speaker Amazon Echo, will play a role in how brands market to consumers in the future CREDIT: BLOOMBERG FINANCE LP/LUKE MACGREGOR

Google DeepMind's AI learns to play with physical objects

She could teach AI a thing or two

Hard Science

End to Illness: Machine Learning Is Revolutionizing How We Prevent Disease

3dnews

IN BRIEF

- The TeraStructure algorithm can analyze genome sets much larger than current systems can efficiently handle, including those as big as 100,000 or 1 million genomes.
- Finding an efficient way to analyze genome databases would allow for personalized healthcare that takes into account any genetic mutations that could exist in a person's DNA.

SHARE

WRITTEN BY

AUTHOR
Jelor Gallego

EDITOR
Kristin Houser
Website

Samsung's Bet on Artificial Intelligence Is a Good One -- If It Can Pull It Off

terms

AI (artificial intelligence)

- the theory and development of computer systems able to perform tasks that normally require human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages

ML (machine learning)

- is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of computer programs that can teach themselves to grow and change when exposed to new data.

without being explicitly programmed

FF NN cost function

$$J(\Theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)} \log(h_\Theta(x^{(i)}))_k + (1 - y_k^{(i)}) \log(1 - (h_\Theta(x^{(i)}))_k)\right]$$

$$h_\Theta(x) \in \mathbb{R}^K$$
$$(h_\Theta(x))_i = i^{th} \text{ output}$$

FF NN Cost Function

$$J(\Theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)} \log(h_\Theta(x^{(i)}))_k + (1 - y_k^{(i)}) \log(1 - (h_\Theta(x^{(i)}))_k)\right]$$

I'm kidding

$$h_\Theta(x) \in \mathbb{R}^K$$
$$(h_\Theta(x))_i = i^{th} \text{ output}$$

cost function with regularization

$$J(\Theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)} \log(h_\Theta(x^{(i)}))_k + (1 - y_k^{(i)}) \log(1 - (h_\Theta(x^{(i)}))_k)\right]$$

$$+\frac{\lambda}{2m}\sum_{l=1}^{L-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(\Theta_{ji}^{(l)})^2$$

$h_\Theta(x) \in \mathbb{R}^K$

$(h_\Theta(x))_i = i^{th}$ output

2 types of ML

supervised learning

unsupervised learning

supervised

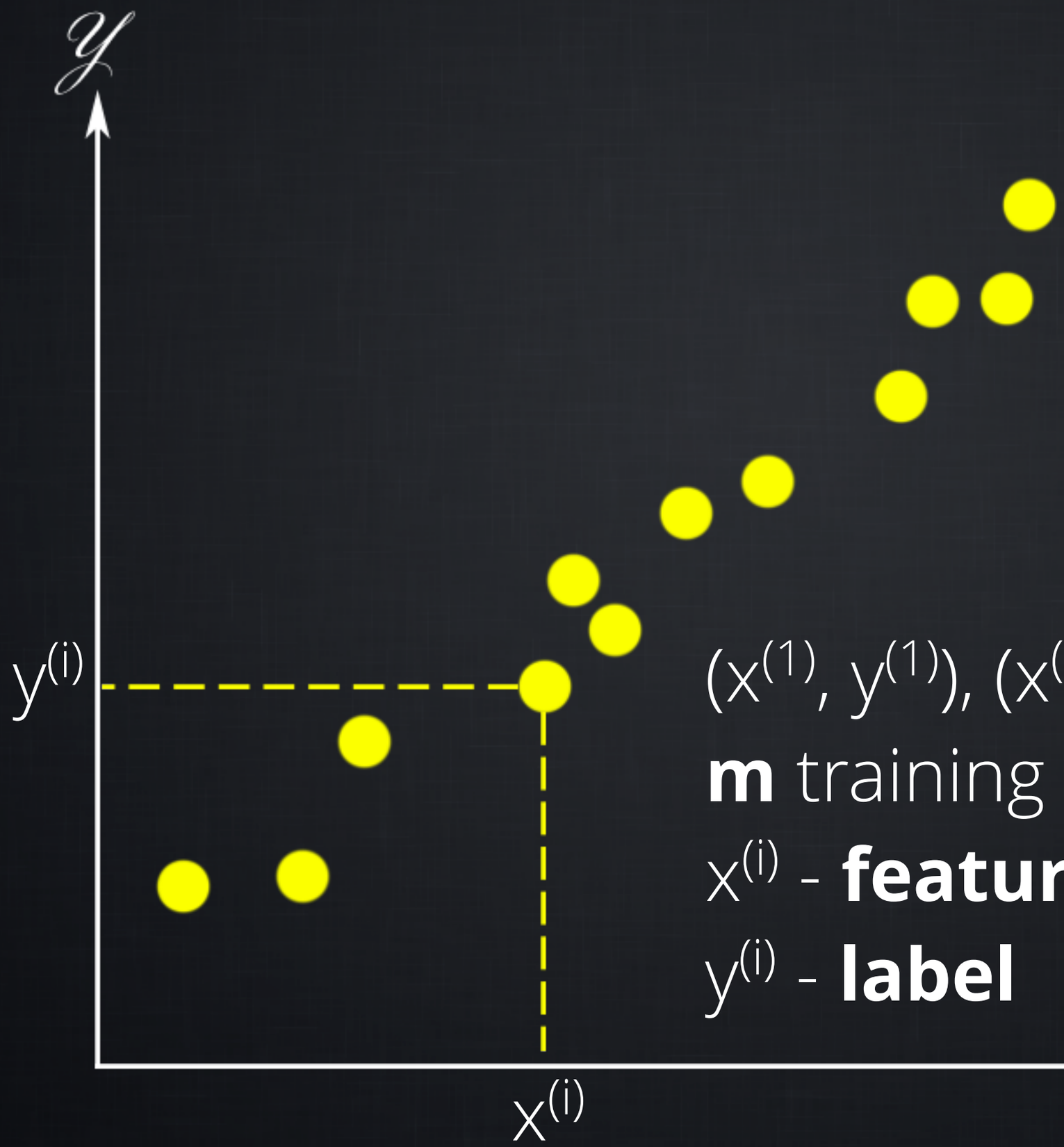
the training data is labeled,
eg. we know the correct answer

unsupervised

the training data is not labeled,
eg. we would figure out hidden correlations by ourselves

linear regression

supervised learning

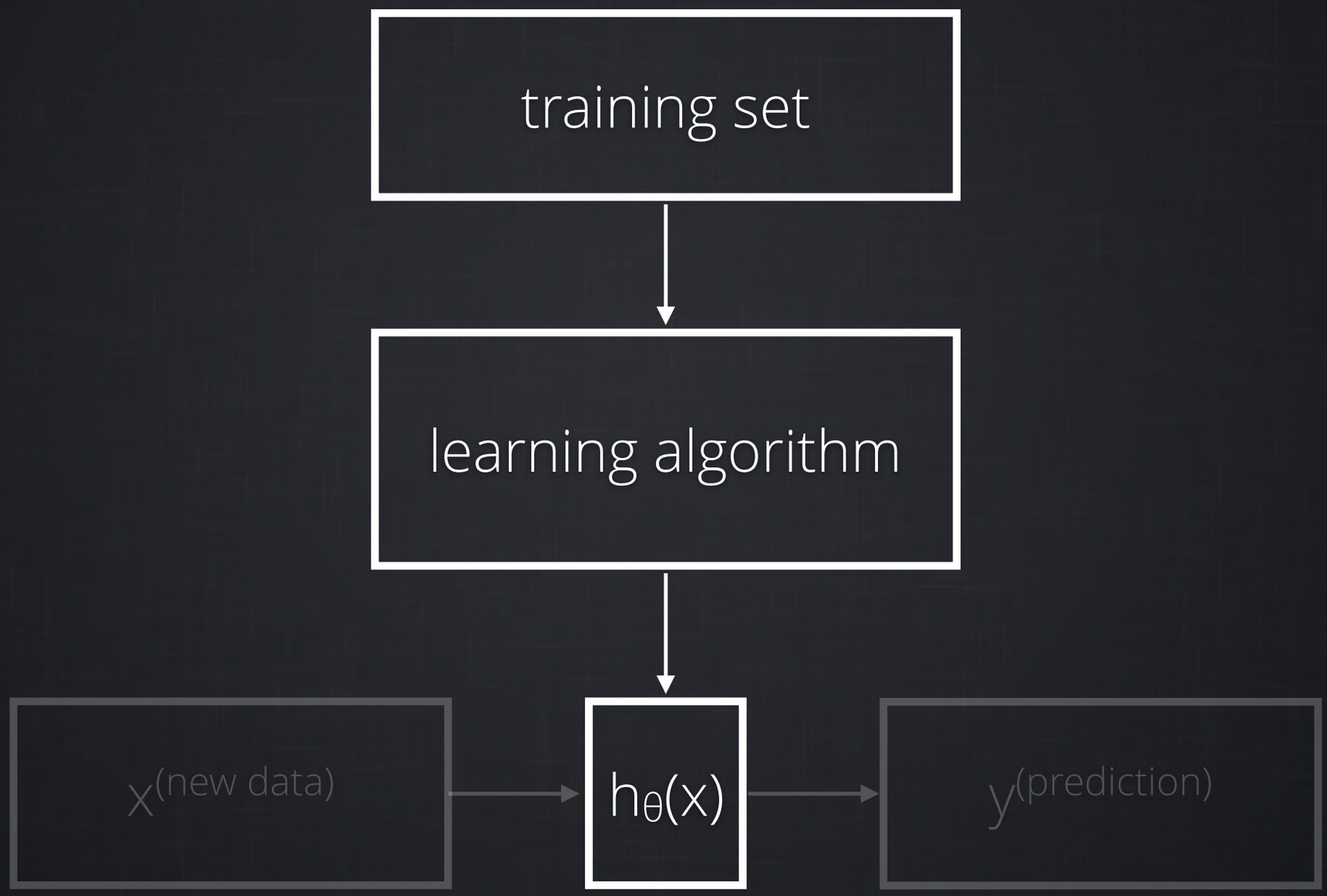


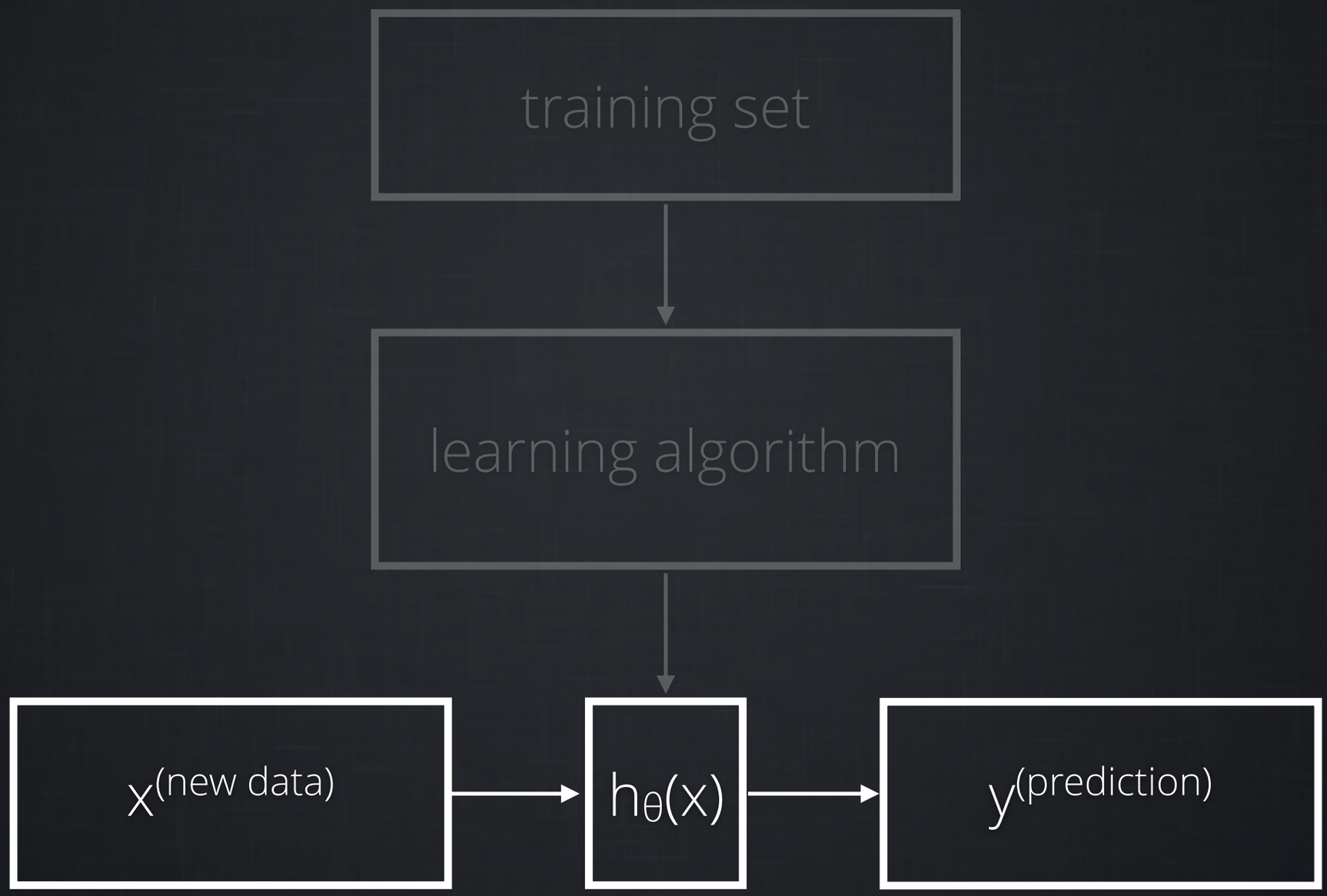
$(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)})...(x^{(m)}, y^{(m)})$

m training examples of $(x^{(i)}, y^{(i)})$

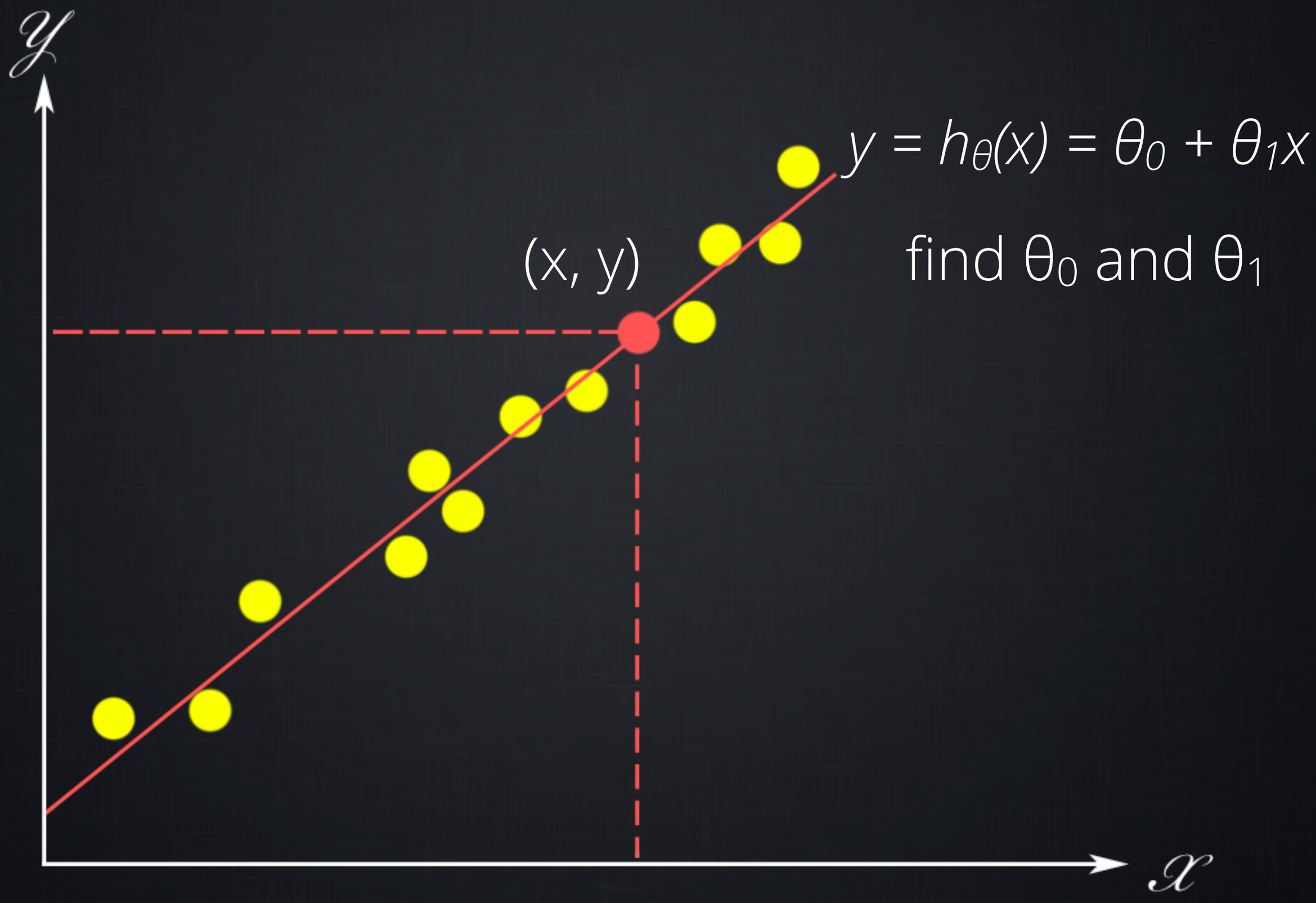
$x^{(i)}$ - **feature**

$y^{(i)}$ - **label**


training set
learning algorithm
x(new data)
hθ(x)
y(prediction)


training set
learning algorithm
x(new data)
hθ(x)
y(prediction)

$h_\theta(x)$ = hypothesis


y
x
(x, y)
$y = h_\theta(x) = \theta_0 + \theta_1 x$
find θ_0 and θ_1

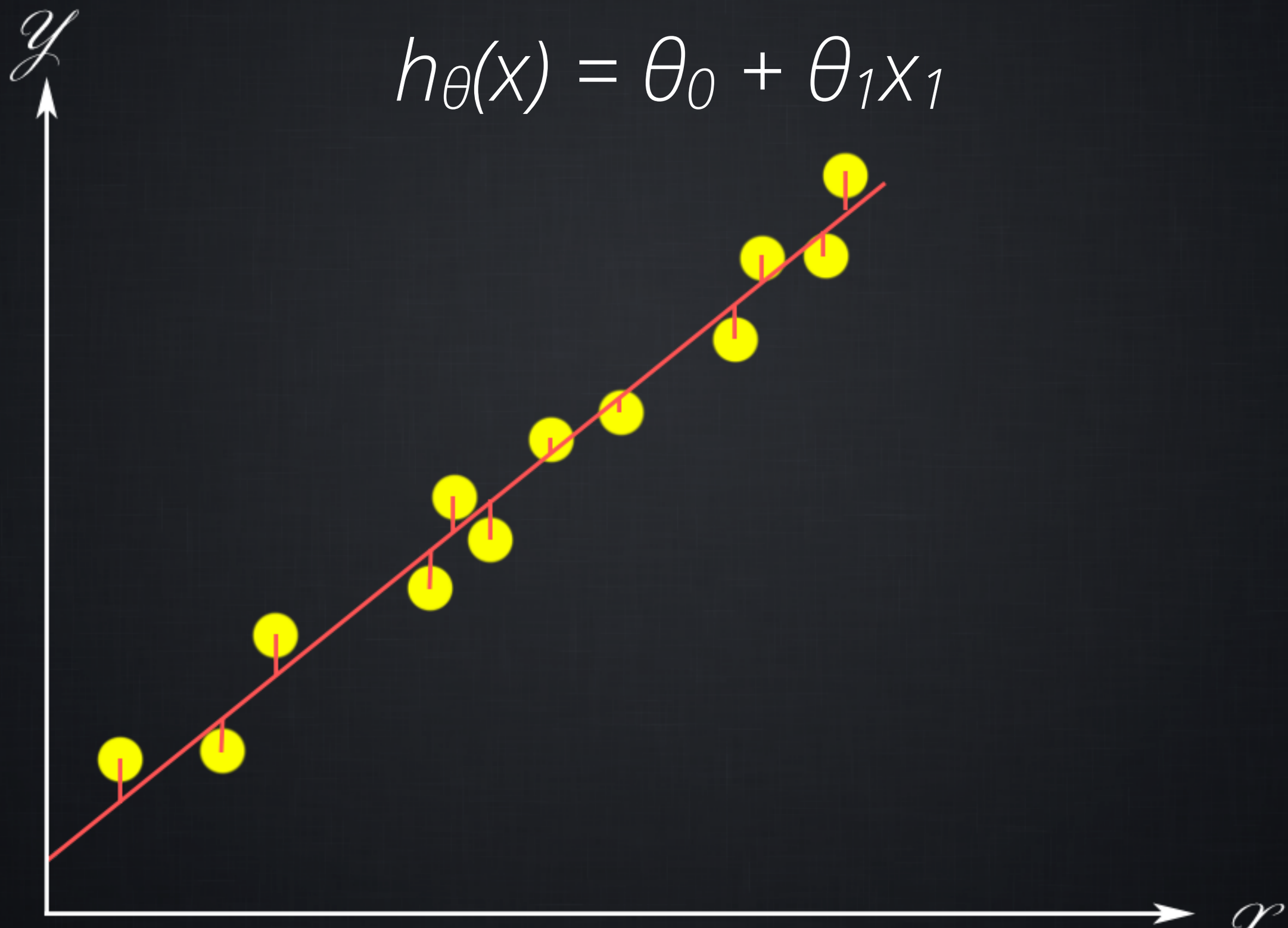
$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

many features, **n** - number of features

size, sq.m x_1	# rooms x_2	age x_3	price y
80	3	22	2.9M
90	4	24	3.1M
75	3	28	2.5M
110	5	20	3.3M

1 USD = 3.85 NIS

$$h_\theta(x) = \theta_0 + \theta_1 x_1$$



summate the prediction error on training set

Linear Regression Cost Function

$$J(\theta_0, \theta_1, \dots, \theta_n) = \frac{1}{2m} \sum_{i=0}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right)^2$$

minimize J(θ)

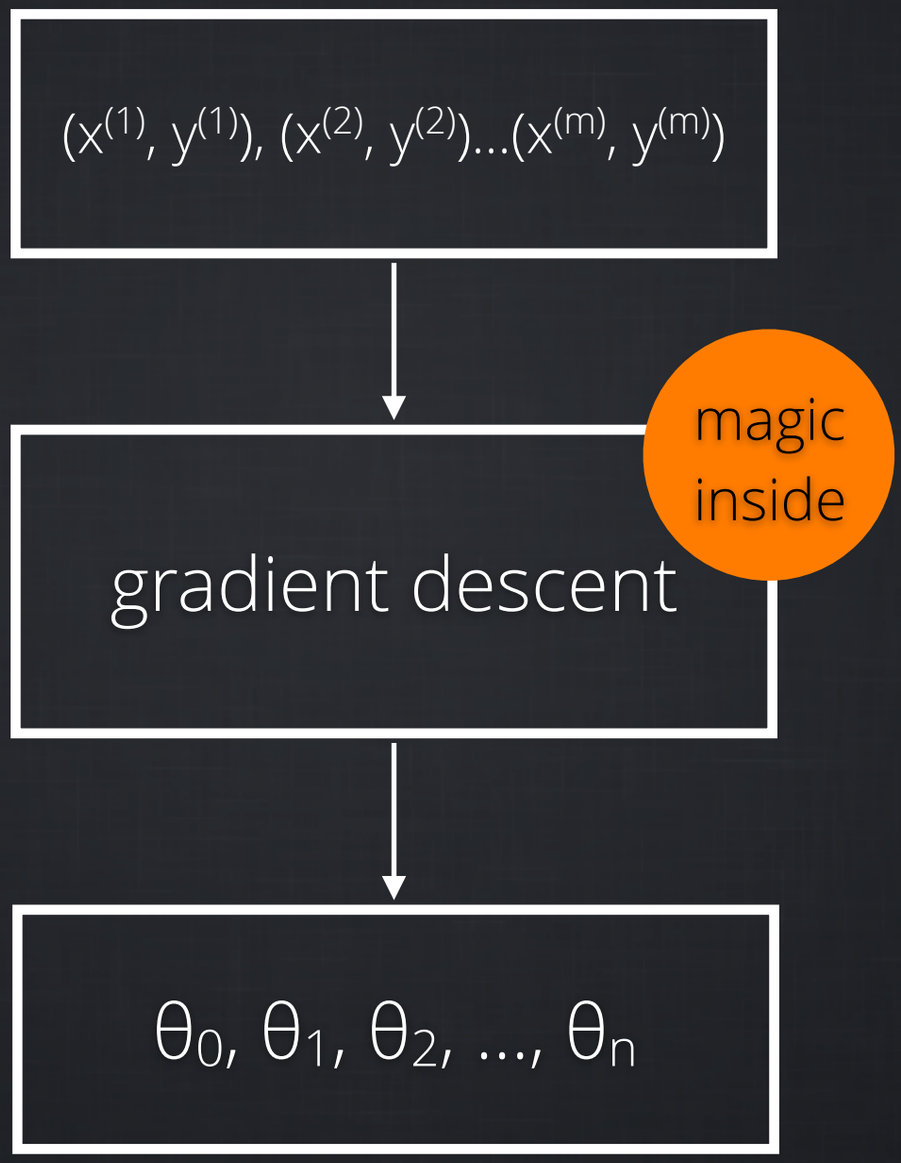
funding a minimum of cost function = “learning”

gradient descent

batch, stochastic, etc, or advanced optimization algorithms
to find a global (sometimes local) minimum of cost function J

α - learning rate, a parameter of gradient descent

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right) x_j^{(i)}$$


$(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)})...(x^{(m)}, y^{(m)})$
gradient descent
magic inside
$\theta_0, \theta_1, \theta_2, ..., \theta_n$

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

we're ready to predict

features scaling

$0 \leq x \leq 1$

size, sq.m	size, sq.m / 110 x_1
80	0.72
90	0.81
75	0.68
110	1

mean normalization

average value of the feature is ~0

$-0.5 \leq x \leq 0.5$

size, sq.m	(size, sq.m / 110) - 0.8025 x_1
80	-0.0825
90	0.075
75	-0.1226
110	0.1975

matrix manipulations

$\mathbf{X}$ = n x 1 vector, Θ = n x 1 vector

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

$$h_\theta(x) = \Theta^T \mathbf{X}$$

GPU

NVIDIA Corporation (NASDAQ:NVDA)

87.92 +20.15 (29.73%)

After Hours: 87.96 +0.04 (0.05%)
Nov 11, 4:24PM EST
NASDAQ real-time data - Disclaimer
Currency in USD

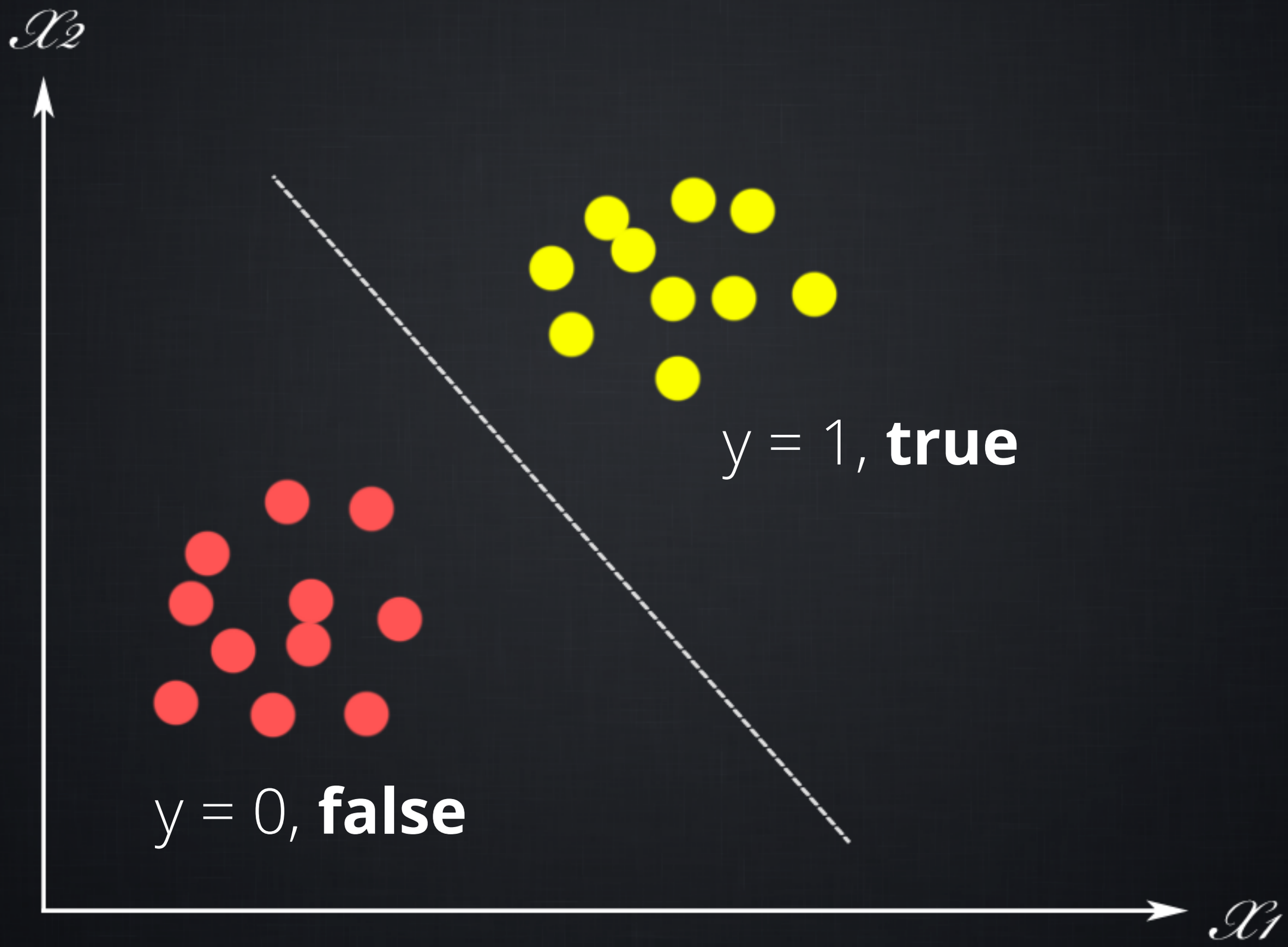
Range	78.50 - 88.77	Div/yield	0.12/0.52
52 week	24.75 - 88.77	EPS	1.52
Open	79.51	Shares	538.00M
Vol / Avg.	54.22M/8.86M	Beta	1.17
Mkt cap	45.56B	Inst. own	89%
P/E	57.88		

1d 5d 1m 3m 6m 1y 5y Max

logistic regression

supervised learning

classifier

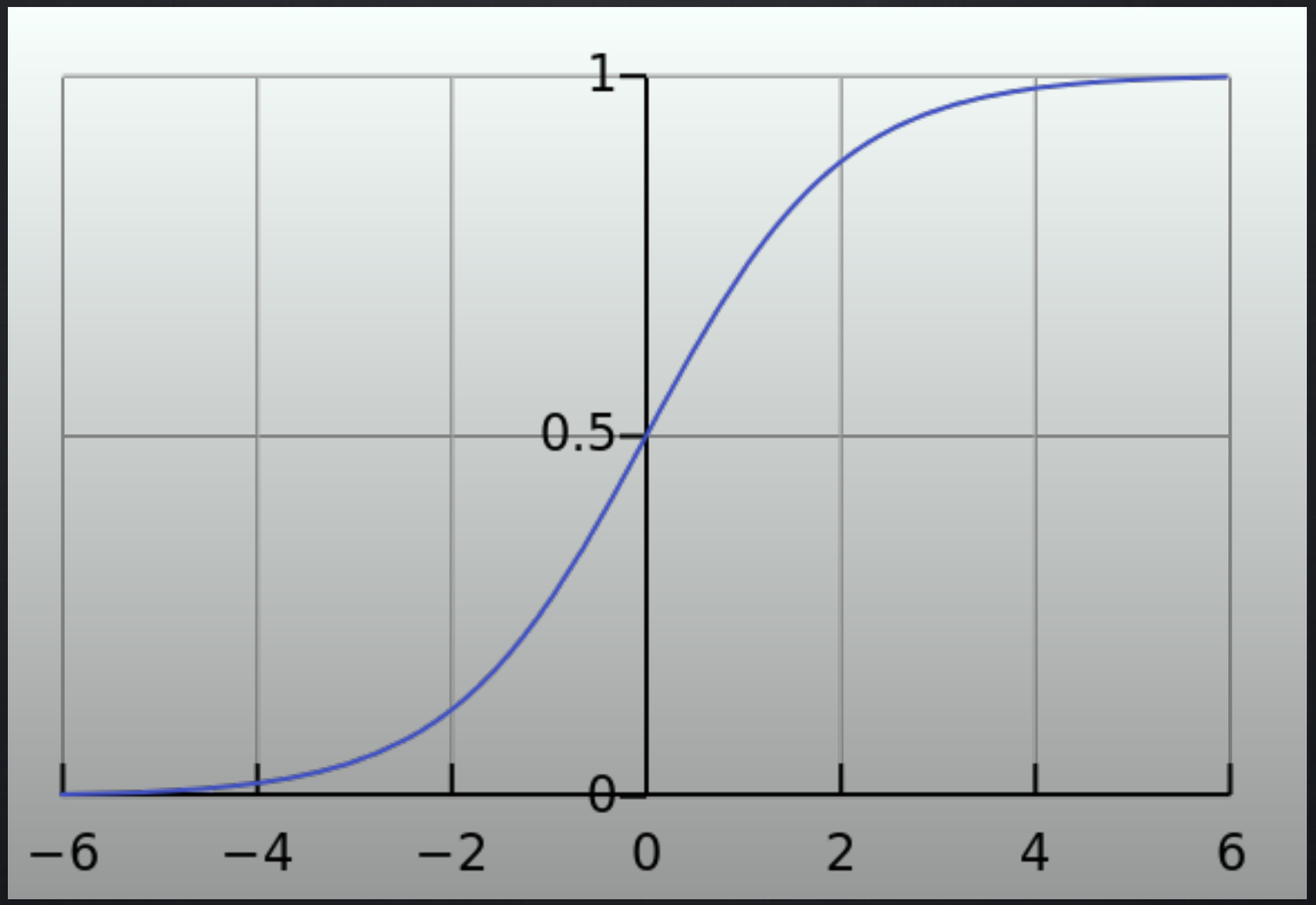

x_2
y = 1, **true**
y = 0, **false**
x_1

$$h_\theta(x) = g(\Theta^T \boldsymbol{X})$$

$h_\theta(X)$ - estimated probability that y = 1 on input X

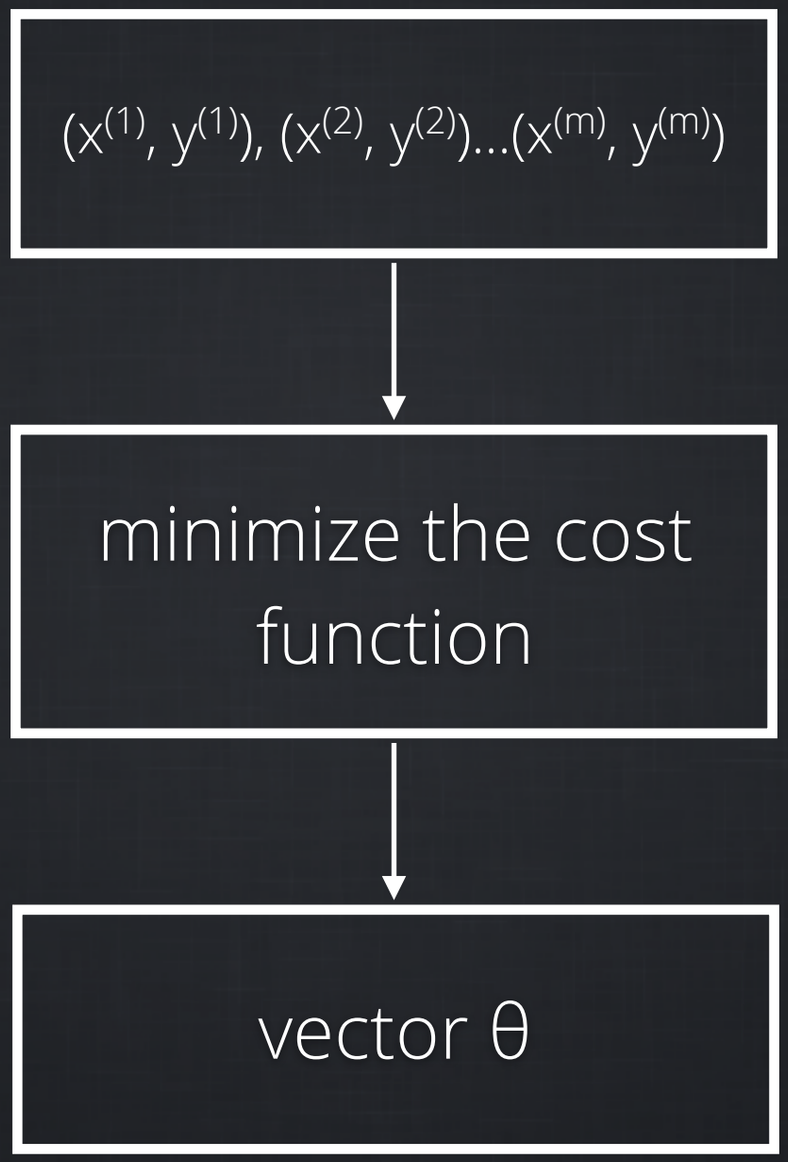
g(z) - logistic non-linear function

logistic function $g(z)$

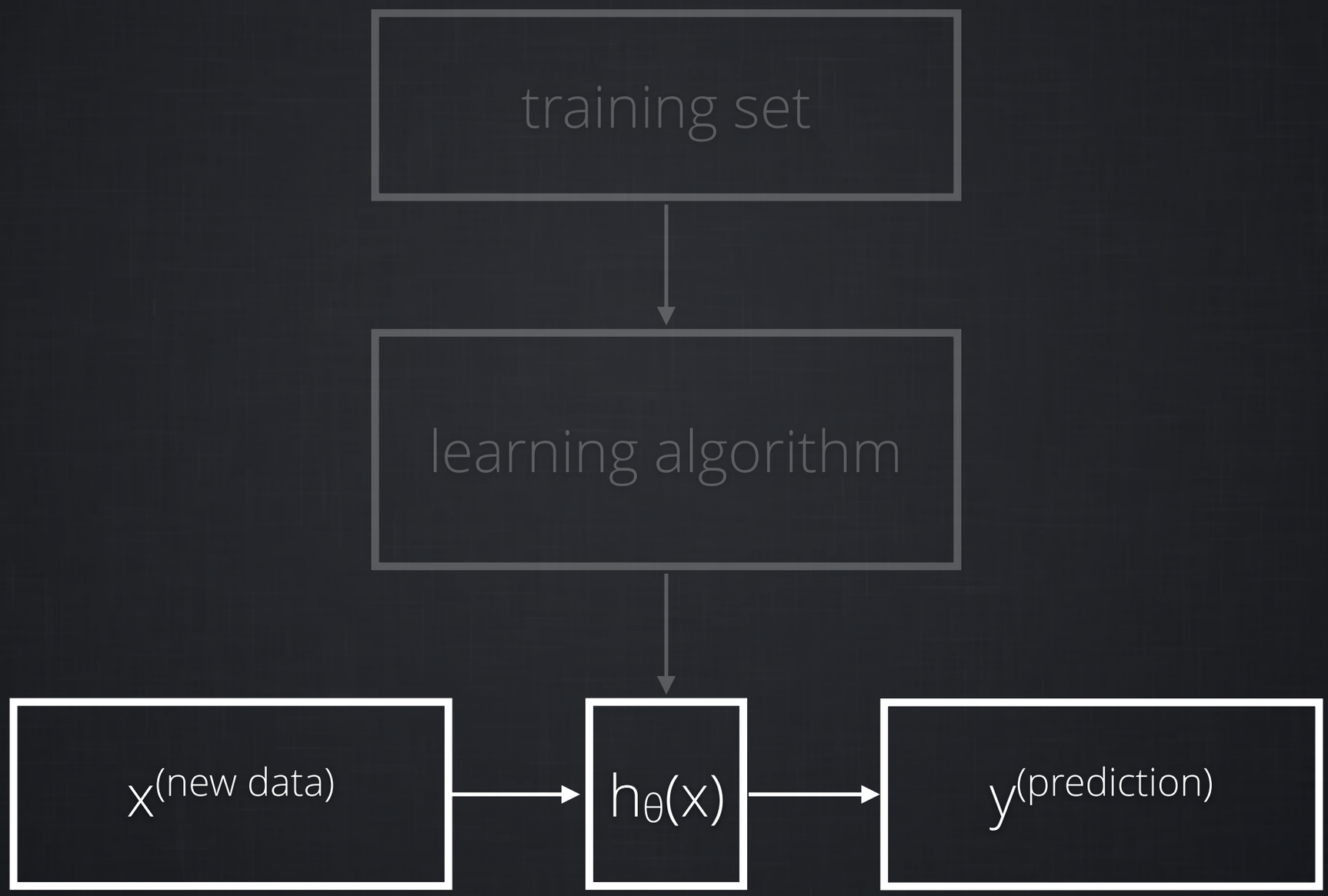


there is a few: sigmoid, tahn, ReLUs, etc

image source: Wikipedia



$y = \{0, 1\}$



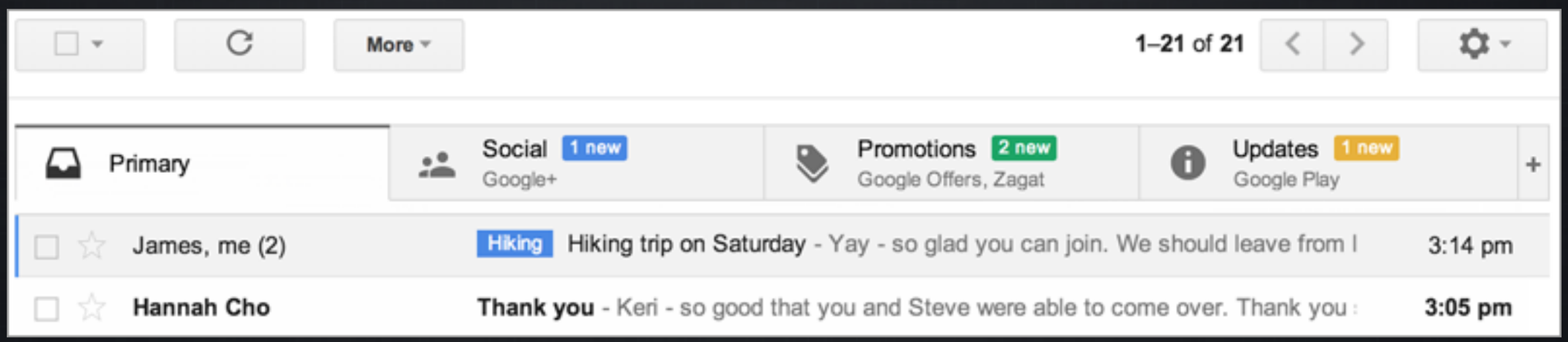
$h_\theta(x) = g(\Theta^T \boldsymbol{X})$

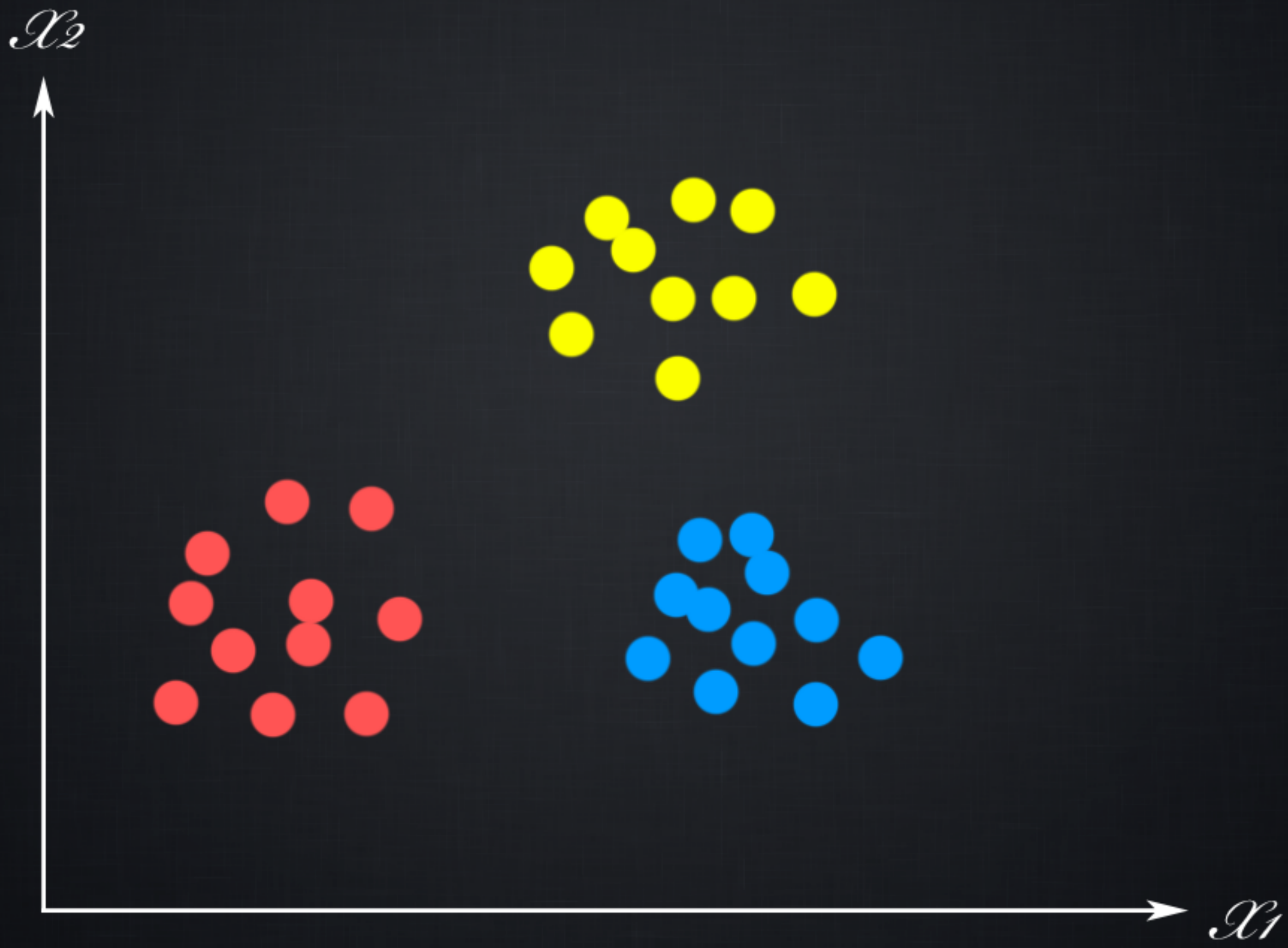
$y \geq 0.5$ - *true*

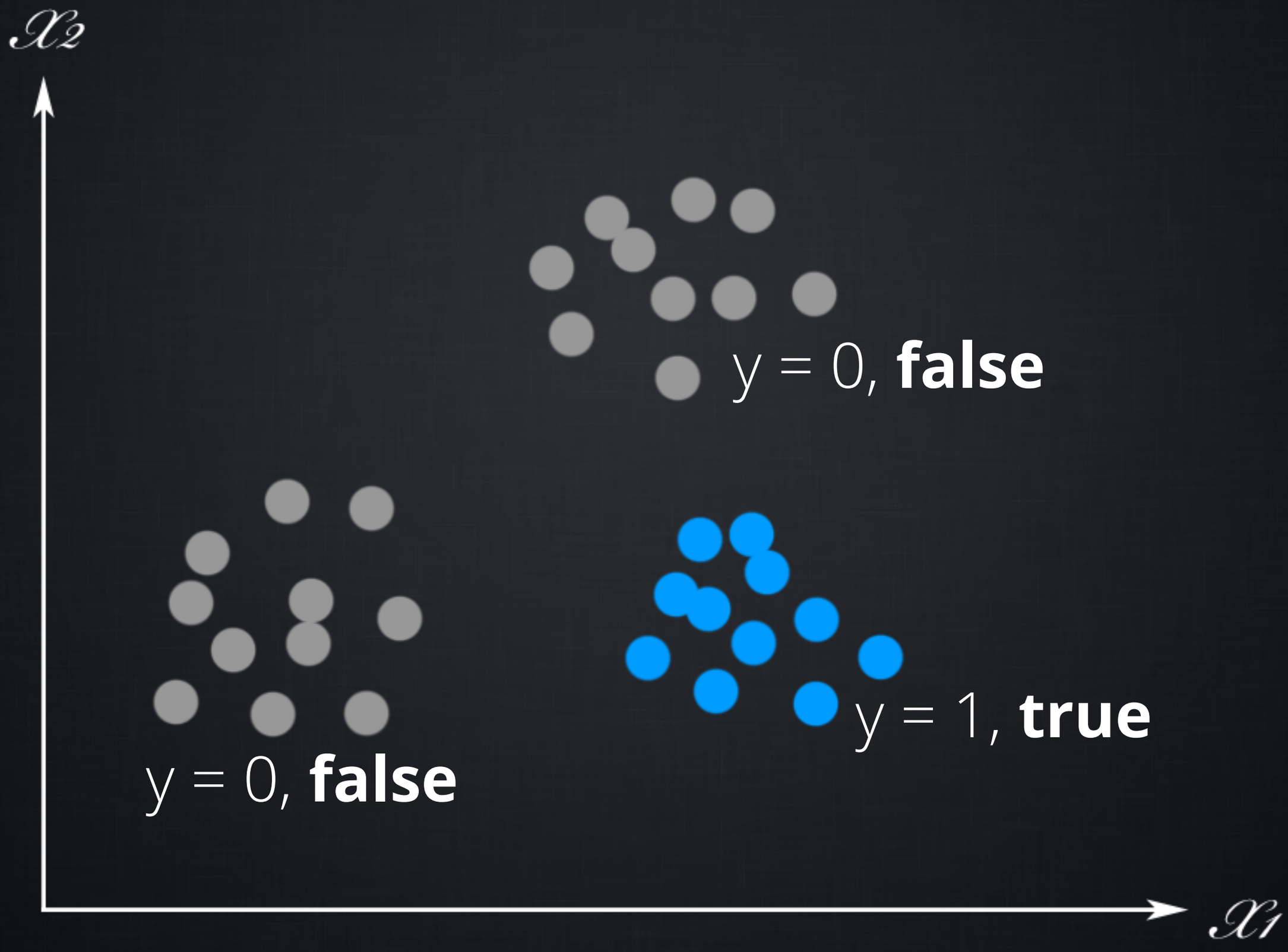
$y < 0.5$ - *false*

one-vs-all

supervised learning




x_2
x_1


x_2
y = 0, **false**
y = 1, **true**
y = 0, **false**
x_1

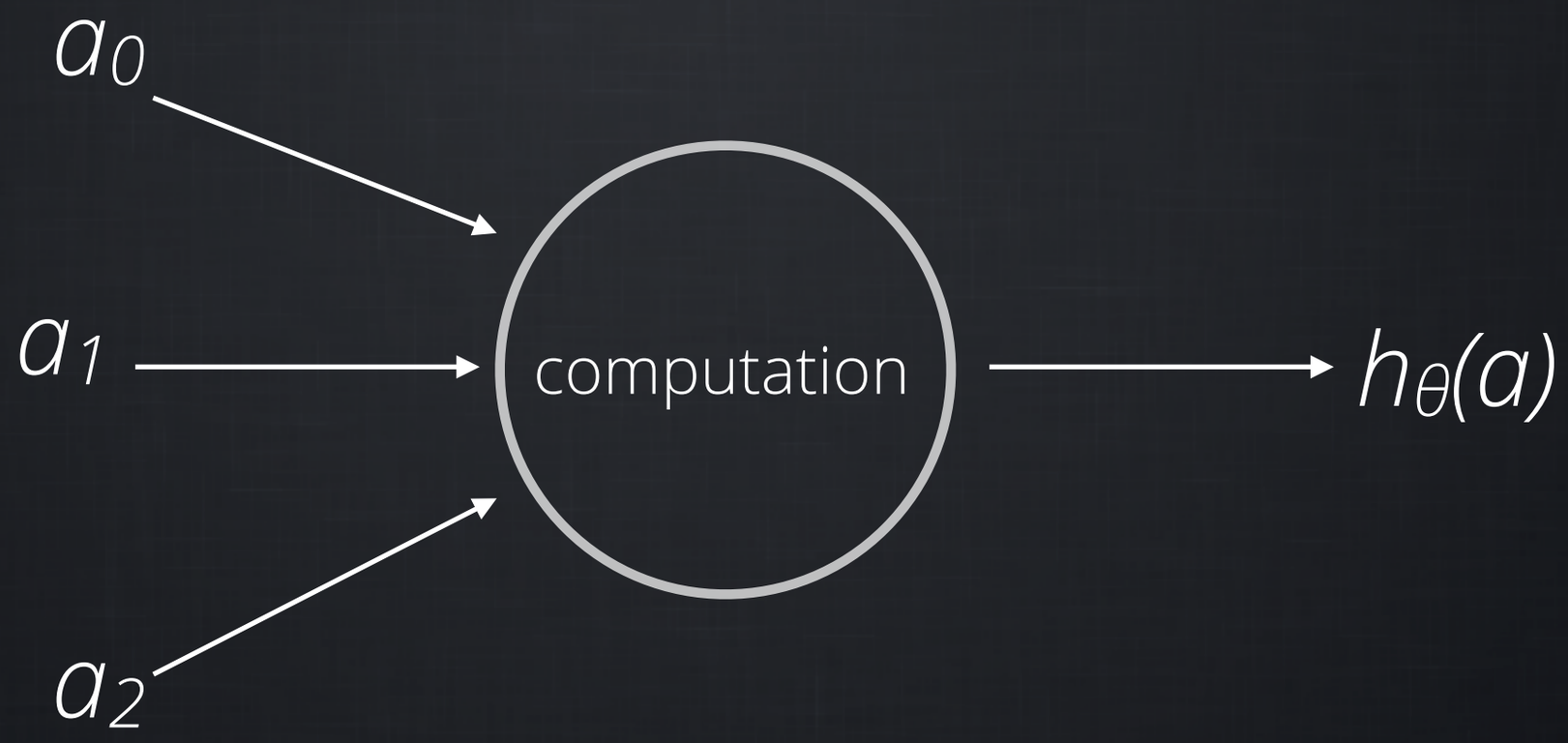
don’t implement it at home

use libsvm, liblinear, and others

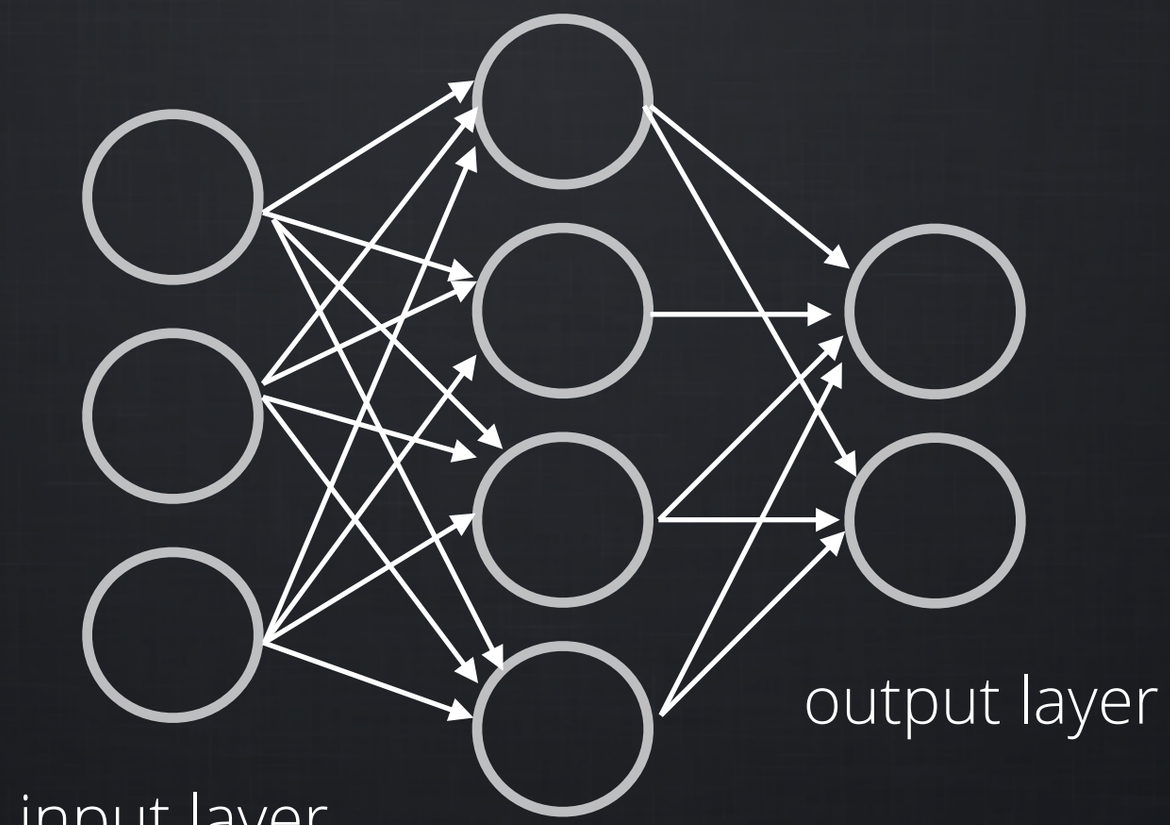
neural networks

supervised learning

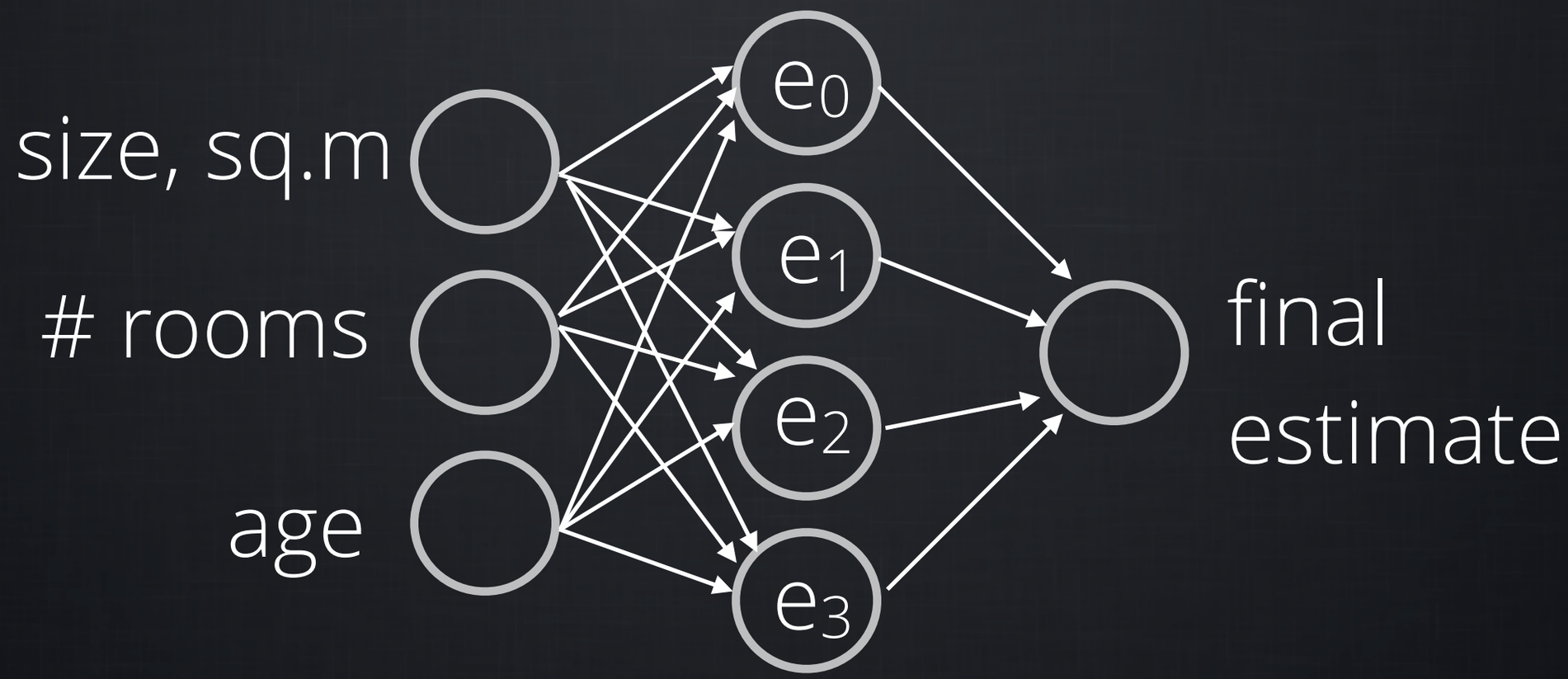
neuron



feed forward neural network

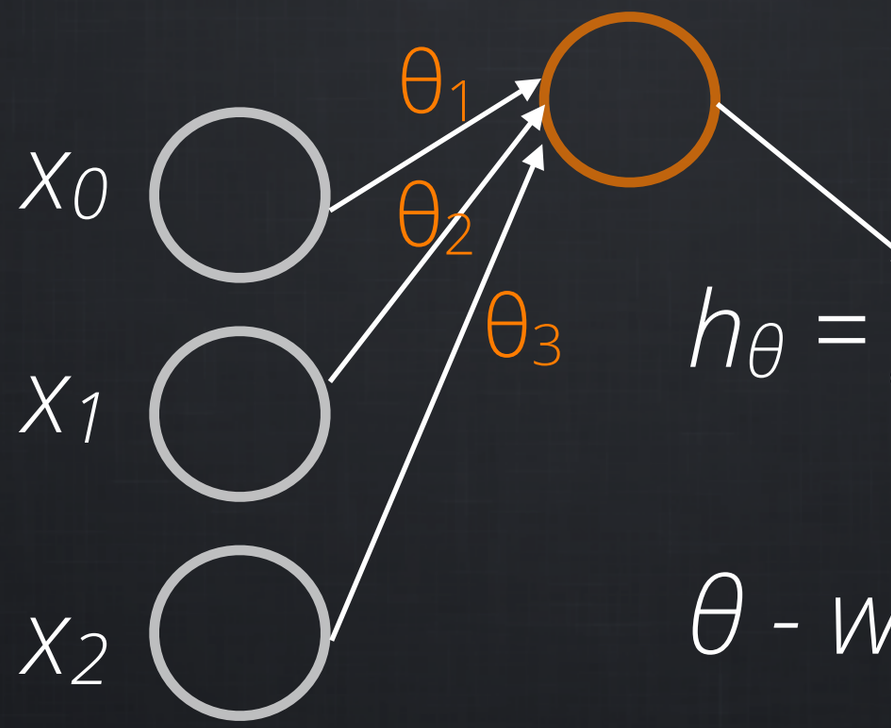


estimates



multiclass classifiers

logistic unit

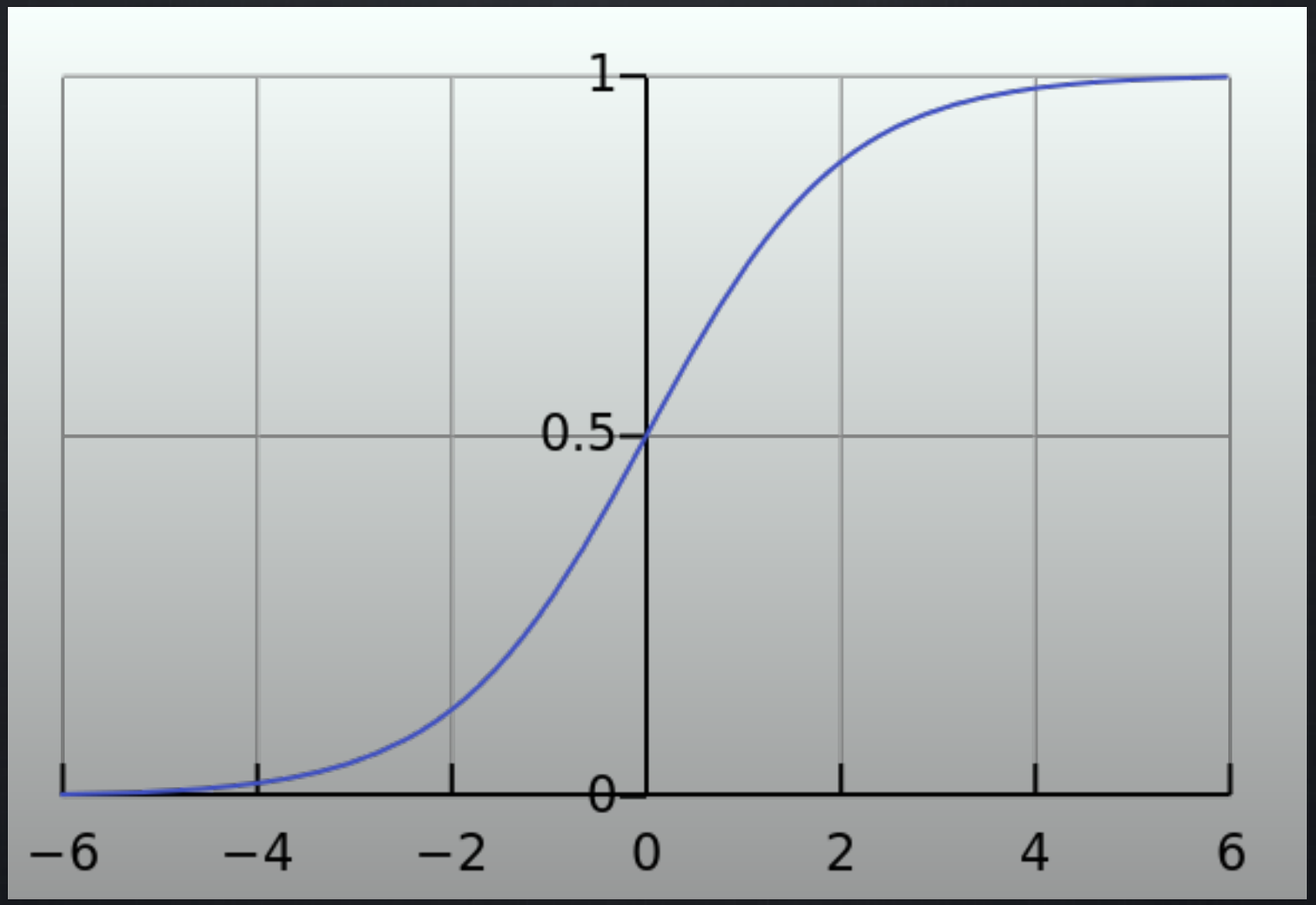


$h_\theta = g(x_0\theta_0 + x_1\theta_1 + x_2\theta_2)$

θ - weights

g - activation function

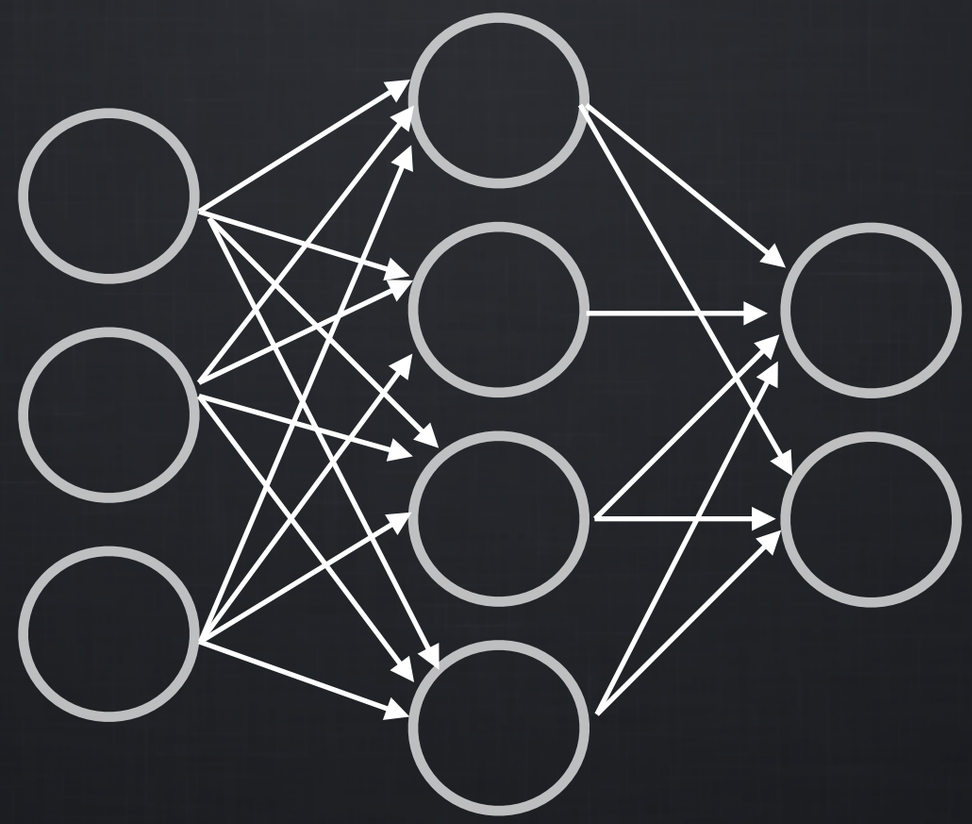
logistic function $g(z)$



there is a few: sigmoid, tahn, ReLUs, etc

image source: Wikipedia

output: probabilities



0.7123 that **y = 1**

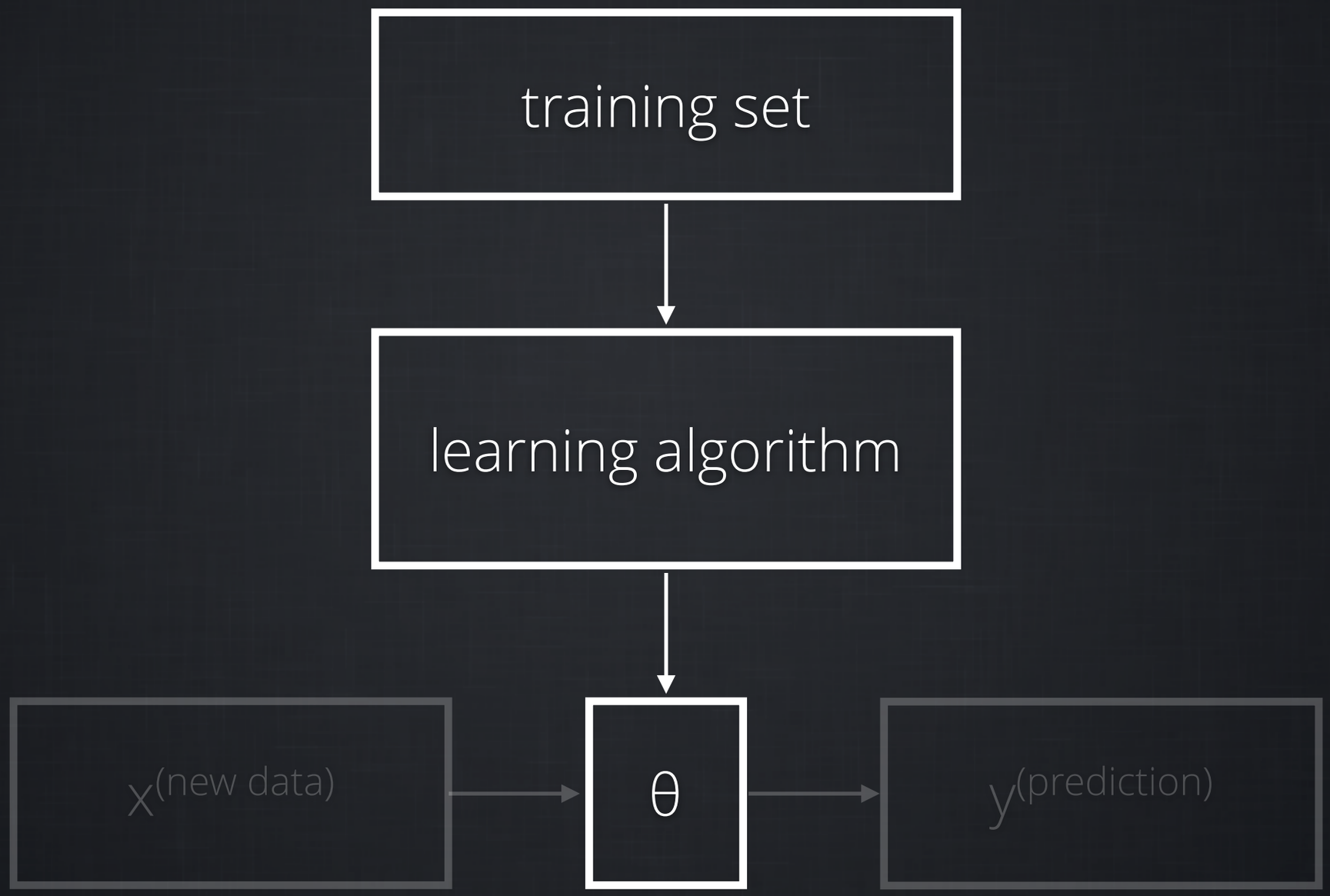
0.4765 that **y = 2**

net with no hidden layers

no hidden layers = one-vs-all logistic regression

cost function

sometimes called loss function of NN,
a representation of an error between
a real and a predicted value


training set
learning algorithm
x(new data)
θ
y(prediction)

backprop

backward propagation of errors

gradient descent + backprop

"deep learning" - is training a neural net

"deep" - because we have many layers

convolutional neural nets

widely used for image processing and object recognition

recurrent neural nets

widely used for natural language processing

CPU/GPU expensive

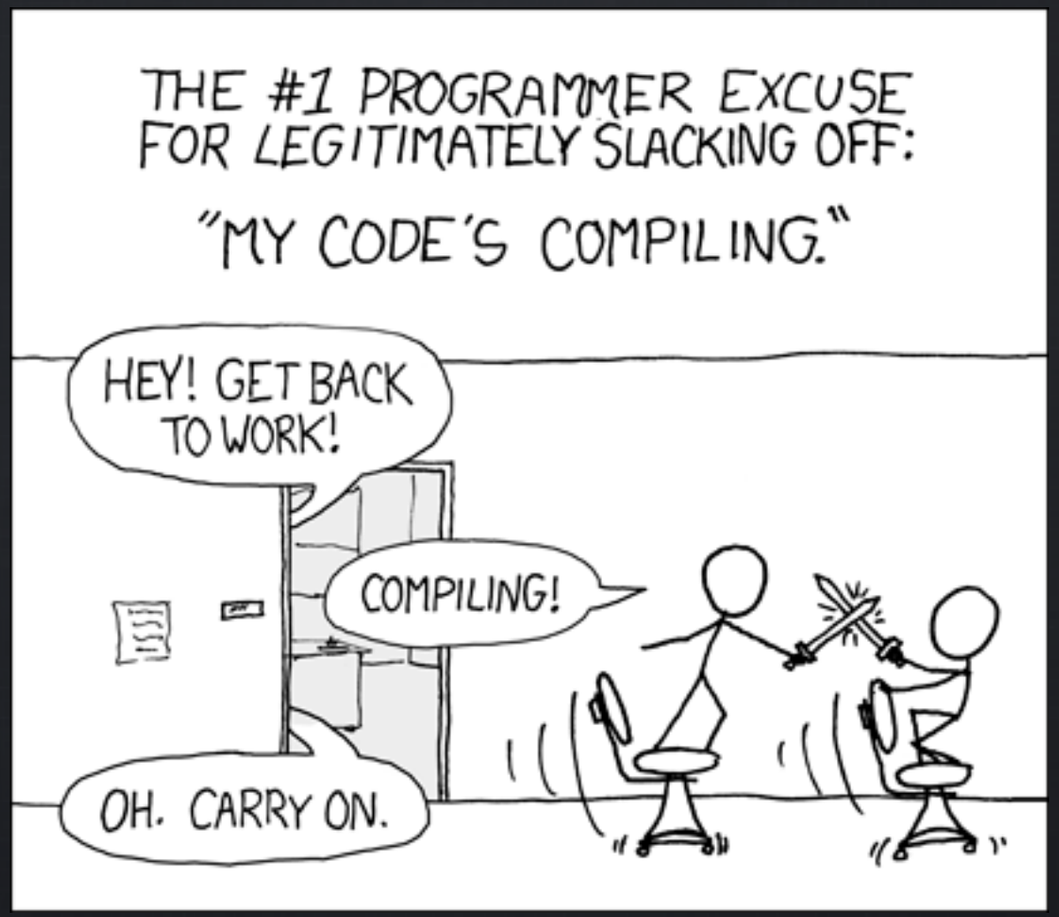
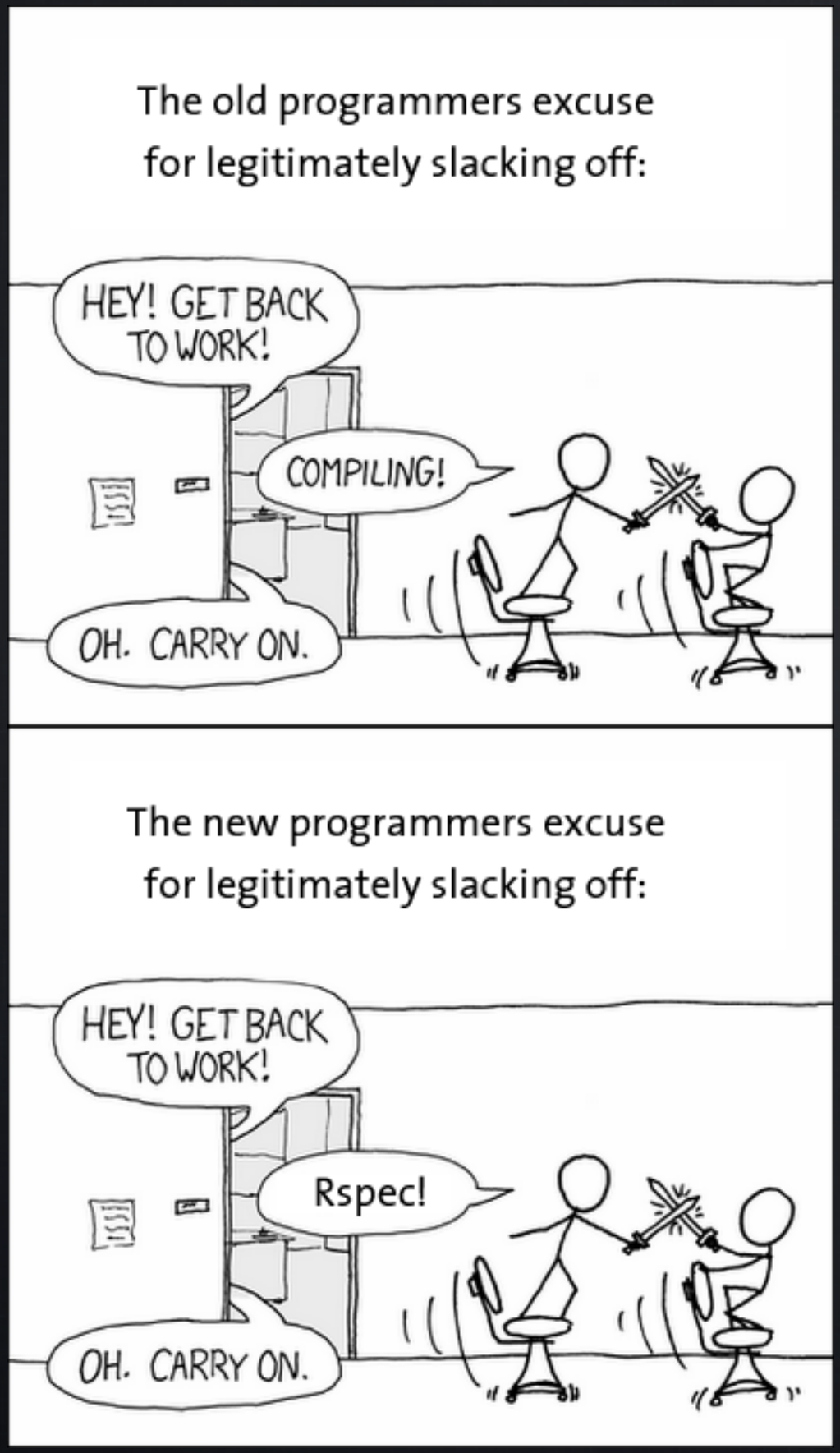


image source: https://xkcd.com/303/

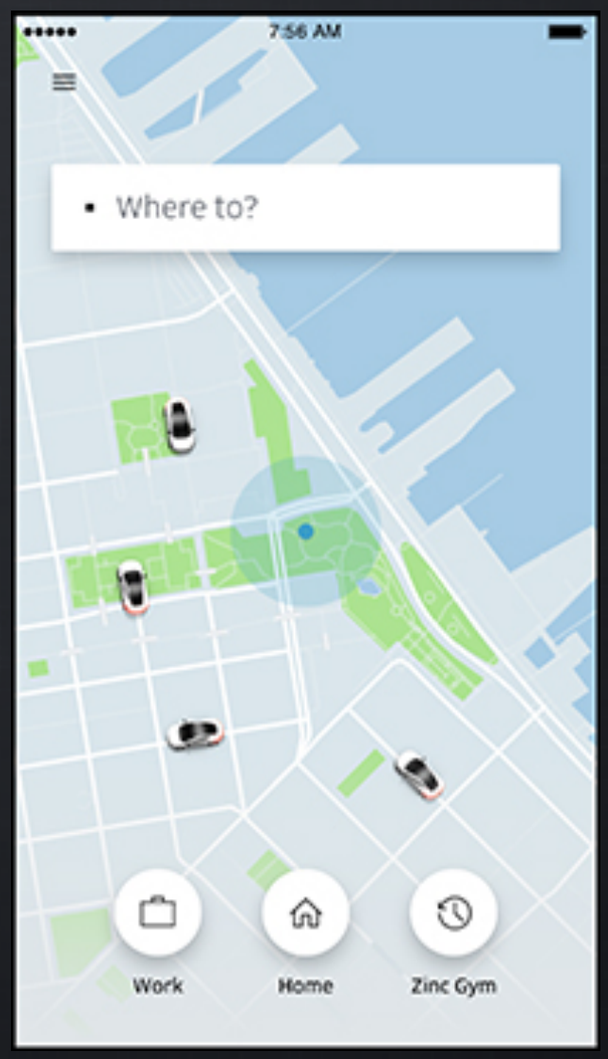


2008

image source: http://www.falsepositives.com/index.php/2008/01/31/the-real-reason-for-no-increased-productivity-behind-scripting-languages-reveled/

2016

destination suggestion



tangledpath/ruby-fann

Ruby library for interfacing with FANN
(Fast Artificial Neural Network)

```ruby
require './neural_network'

LOCATIONS = [:home, :work, :tennis, :parents]

LOCATIONS_INDEXED = LOCATIONS.map.with_index { |x, i| [x, i] }.to_h

XX = [
  # week 1
  # 1st day of week, 8am
  [:work, 1, 8], [:tennis, 1, 17], [:home, 1, 20],
  [:work, 2, 8], [:home,   2, 18],
  [:work, 3, 8], [:tennis, 3, 17], [:home, 3, 20],
  [:work, 4, 8], [:home,   4, 18],
  [:work, 5, 8], [:home,   5, 18],

  [:parents, 7, 13], [:home, 7, 18],

  # week 2
  [:work, 1, 8], [:home, 1, 18],
  [:work, 2, 8], [:home, 2, 18],
  [:work, 3, 8], [:tennis, 3, 17], [:home, 3, 20],
  [:work, 4, 8], [:home, 4, 18],
  [:work, 5, 8], [:home, 5, 18],
```

features scaling

```
XX.each do |destination, day, time|
  yy << LOCATIONS_INDEXED[destination]
  xx << [day.to_f/7, time.to_f/24]
end
```

$$2 \rightarrow 25 \rightarrow 4$$

one hidden layer with 25 units

100% accuracy

on training set

```ruby
[
  [1, 16.5], [1, 17], [1, 17.5], [1, 17.8],
  [2, 17], [2, 18.1],
  [4, 18],
  [6, 23],
  [7, 13],
].each do |day, time|
  res = nn.predict_with_probabilities([
    [day.to_f/7, time.to_f/24]
  ]).first.
   select {|v| v[0] > 0} # filter zero probabilities
  puts "#{day} #{time} \t #{res.map {|v| [LOCATIONS[v[1]], v[0]]}.inspect}"
end
```

```
1 16.5  [[:tennis , 0.97]]
1 17    [[:tennis , 0.86], [:home  , 0.06]]
1 17.5  [[:home   , 0.52], [:tennis, 0.49]]
1 17.8  [[:home   , 0.82], [:tennis, 0.22]]
2 17    [[:tennis , 0.85], [:home  , 0.06]]
2 18.1  [[:home   , 0.95], [:tennis, 0.07]]
4 18    [[:home   , 0.96], [:tennis, 0.08]]
6 23    [[:home   , 1.00]]
```

```
[:work, 1, 8], [:tennis, 1, 17], [:home, 1, 20],
[:work, 2, 8], [:home,   2, 18],
[:work, 3, 8], [:tennis, 3, 17], [:home, 3, 20],
[:work, 4, 8], [:home,   4, 18],
[:work, 5, 8], [:home,   5, 18],

[:parents, 7, 13], [:home, 7, 18],

# week 2
[:work, 1, 8], [:home, 1, 18],
[:work, 2, 8], [:home, 2, 18],
[:work, 3, 8], [:tennis, 3, 17], [:home, 3, 20],
[:work, 4, 8], [:home, 4, 18],
[:work, 5, 8], [:home, 5, 18],
```

borisnadion/suggested-destination-demo

ruby code of the demo

tensorflow

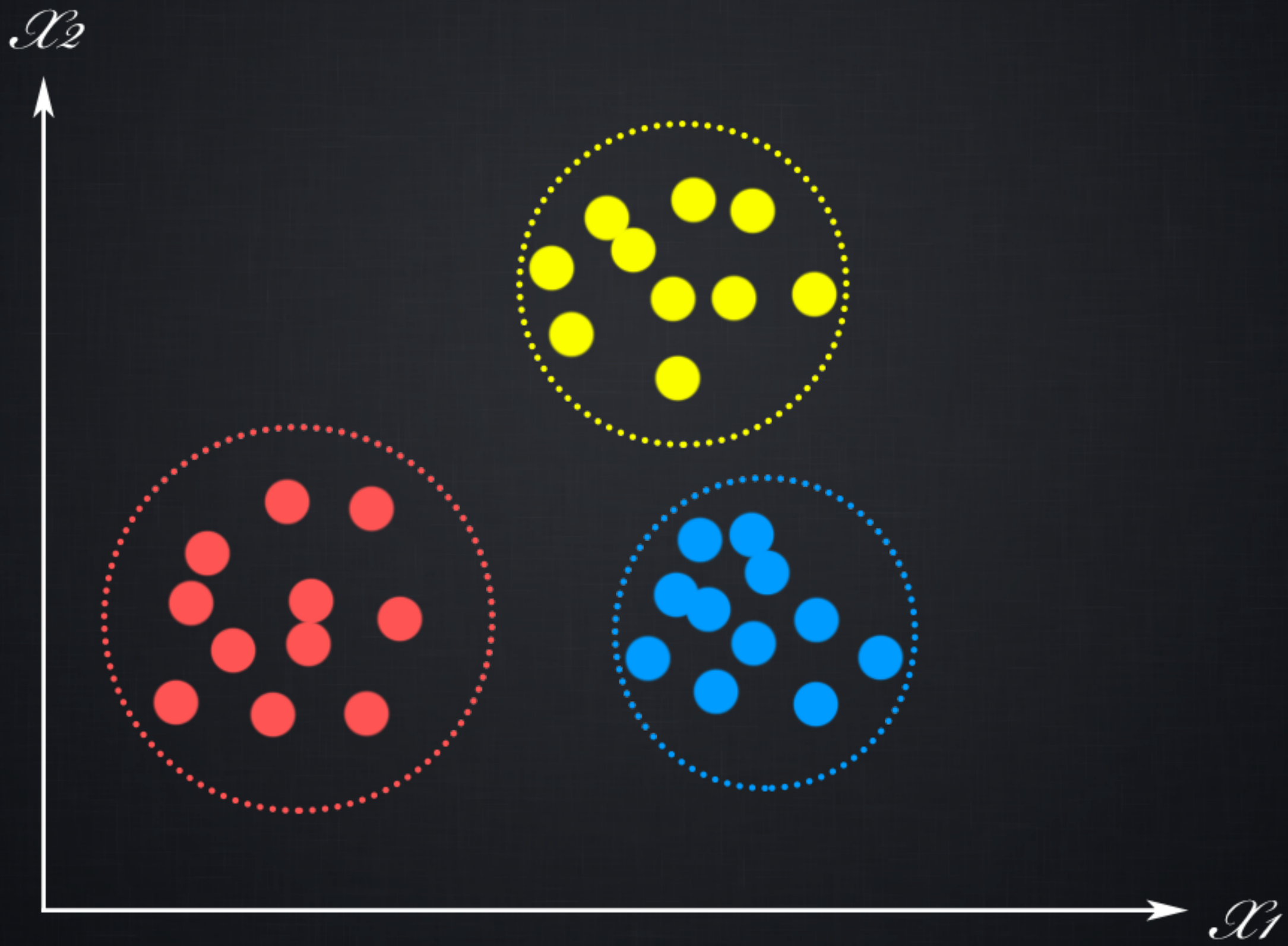
but you will need to learn Python

clustering

unsupervised learning

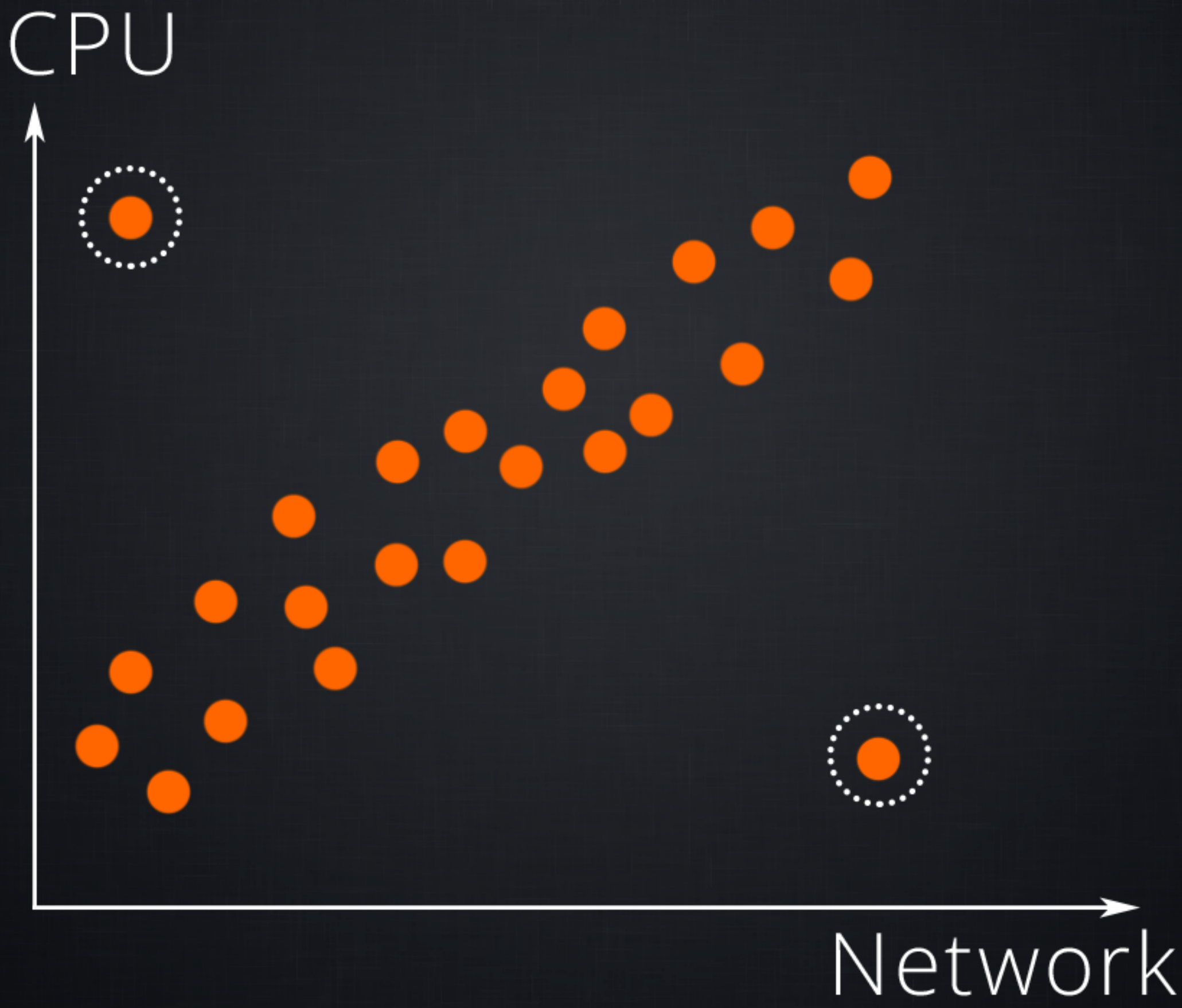
$$\{X^{(i)}\}$$

no labels


x_2
x_1

anomaly detection

unsupervised learning


CPU
Network

collaborative filtering

unsupervised learning

	Jane	Arthur	John
Star Wars VII	5	5	1
Dr. Strange	5	5	?
Arrival	5	?	1

automatic features and their weights detection

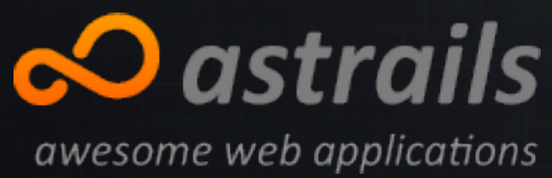
based on the user votes

similarity between users and between items

what to google

http://astrails.com

thanks!



Boris Nadion
http://astrails.com