CMSC 170

Introduction to Artificial Intelligence

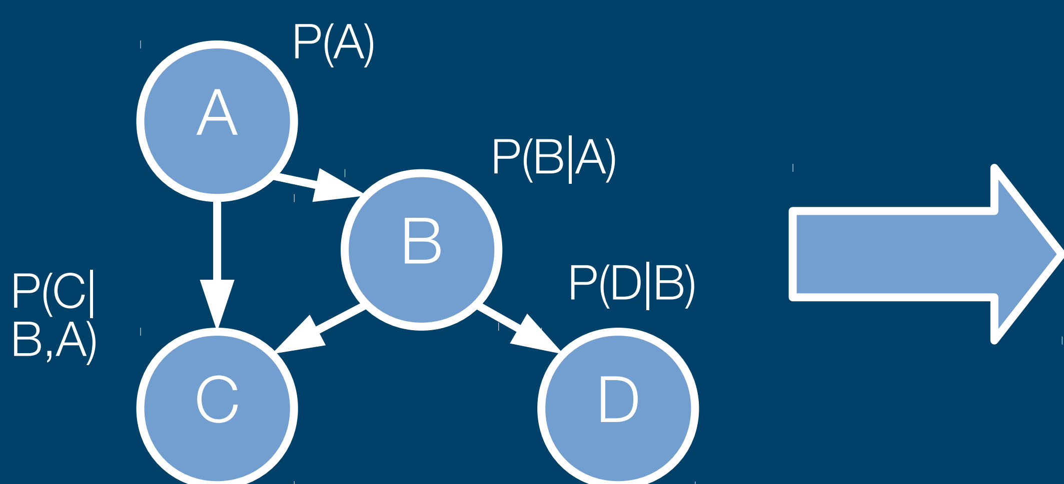
2nd Semester AY 2014-2015

CNM Peralta

Machine Learning

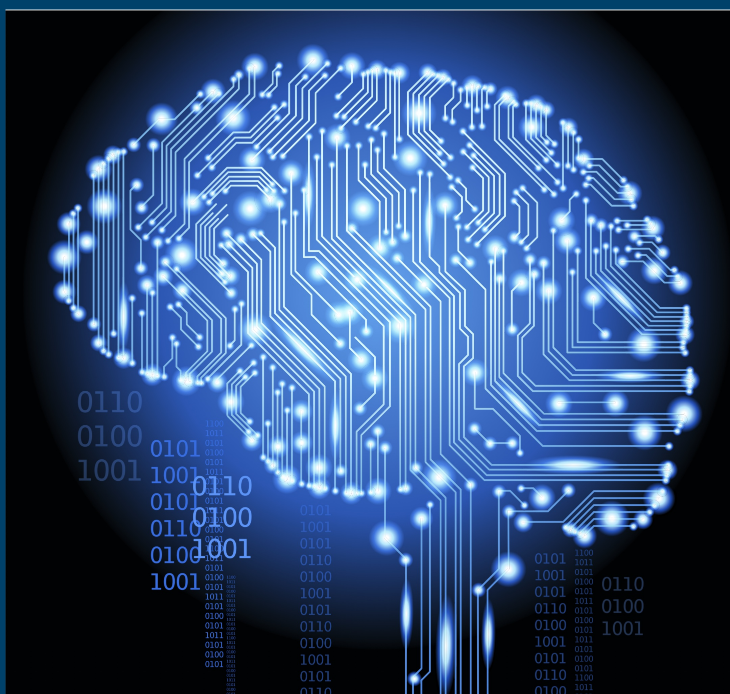
The field of artificial intelligence that is used to make sense of the data rich world we have today.

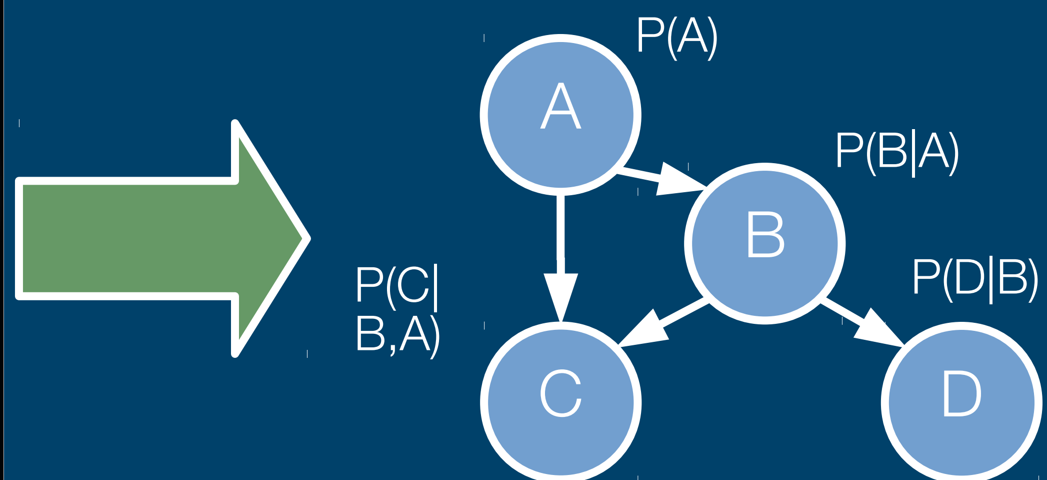
So far, the Bayes networks and probability [distributions] that we have been using have been known/given; we just use the to predict events.



$$P(A|C) = ?$$

Machine learning uses data to **learn models** (like Bayes networks).





Machine learning is being used commercially, by Amazon, Google, etc.

What can be learned?

1.

Parameters

Such as probabilities and probability tables to be used by Bayes networks.

2.

Structure of Bayes networks and other models.

3.

Hidden Concepts

that can be observed from natural data/human behavior to help make sense of data, e.g., natural data clustering.

From what data can we learn?

1.

Supervised Learning

Uses data that already have given target labels.

2.

Unsupervised Learning

Uses data where target labels are missing; hidden concepts are found using replacement principles.

3.

Reinforcement Learning

Uses environment feedback to learn.

Why are we learning?

1.

Prediction

of future events using models derived from past data, e.g., weather forecasting.

2.

Diagnosis

of the reasons or explanations behind events, e.g., medical diagnosis.

3.

Summarization

of possibly many sources of data into a concise form, e.g., article summarization.

How do machines learn?

1.

Passive

Agents only observe the environment; they cannot change it.

2.

Active

Agents act on the environment; can affect perceived data.

3.

Online

Agents learn and receive data simultaneously.

4.

Offline

Agents who learn only after receiving all the data.

What are the outputs of machine learning?

1.

Classification

Outputs may be binary or a fixed number of classes, e.g., this is true love (or not).

2.

Regression

Outputs are continuous, e.g., temperature prediction.

What other details?

Methods may be...

GENERATIVE

Model data

vs.

Distinguish data

DISCRIMINATIVE

SUPERVISED LEARNING

A FEW DEFINITIONS...

Feature Vector

A vector of N features the represent an object.

Target Label

Given a feature vector, it is its corresponding object's prediction value/classification.

GIVEN...

$$\overbrace{\begin{array}{ccccccc} x_{11} & x_{12} & x_{13} & \cdots & x_{1N} & \rightarrow & y_1 \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2N} & \rightarrow & y_2 \\ & & & \cdots & & & \\ x_{M1} & x_{M2} & x_{M3} & \cdots & x_{MN} & \rightarrow & y_M \end{array}}^{\text{Data}}$$

X_M
(M feature vectors with N features each)

Y_M
(M target labels)

We want to find:

$$f(x_M) = y_M$$

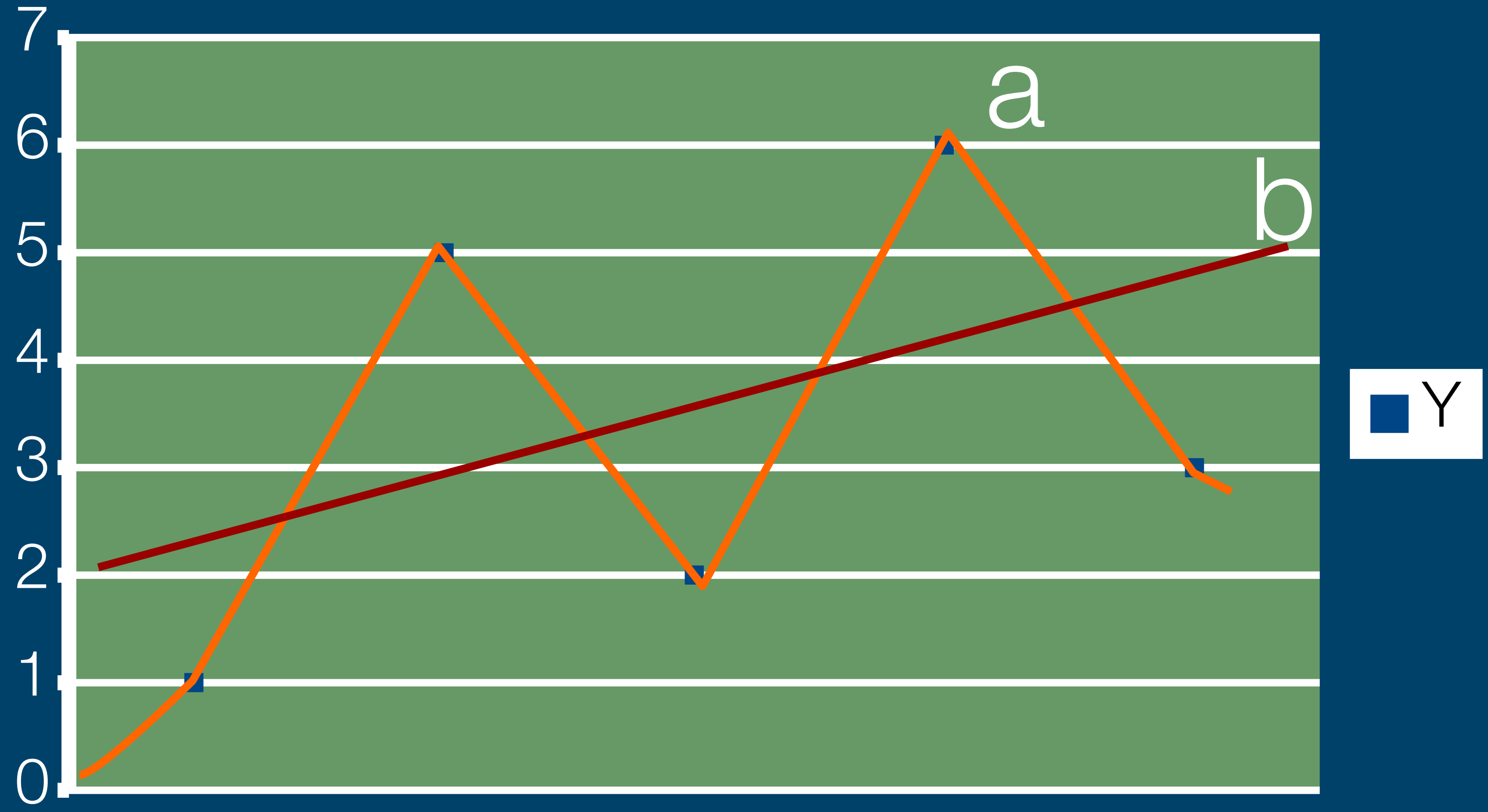
That is, we want to find the function $f(x_m)$ which will **yield y_m given the feature vector x_m**, and can be used to **solve for the target labels** of **future feature vectors**.

The process of learning $f(x_m)$ is often called

training.

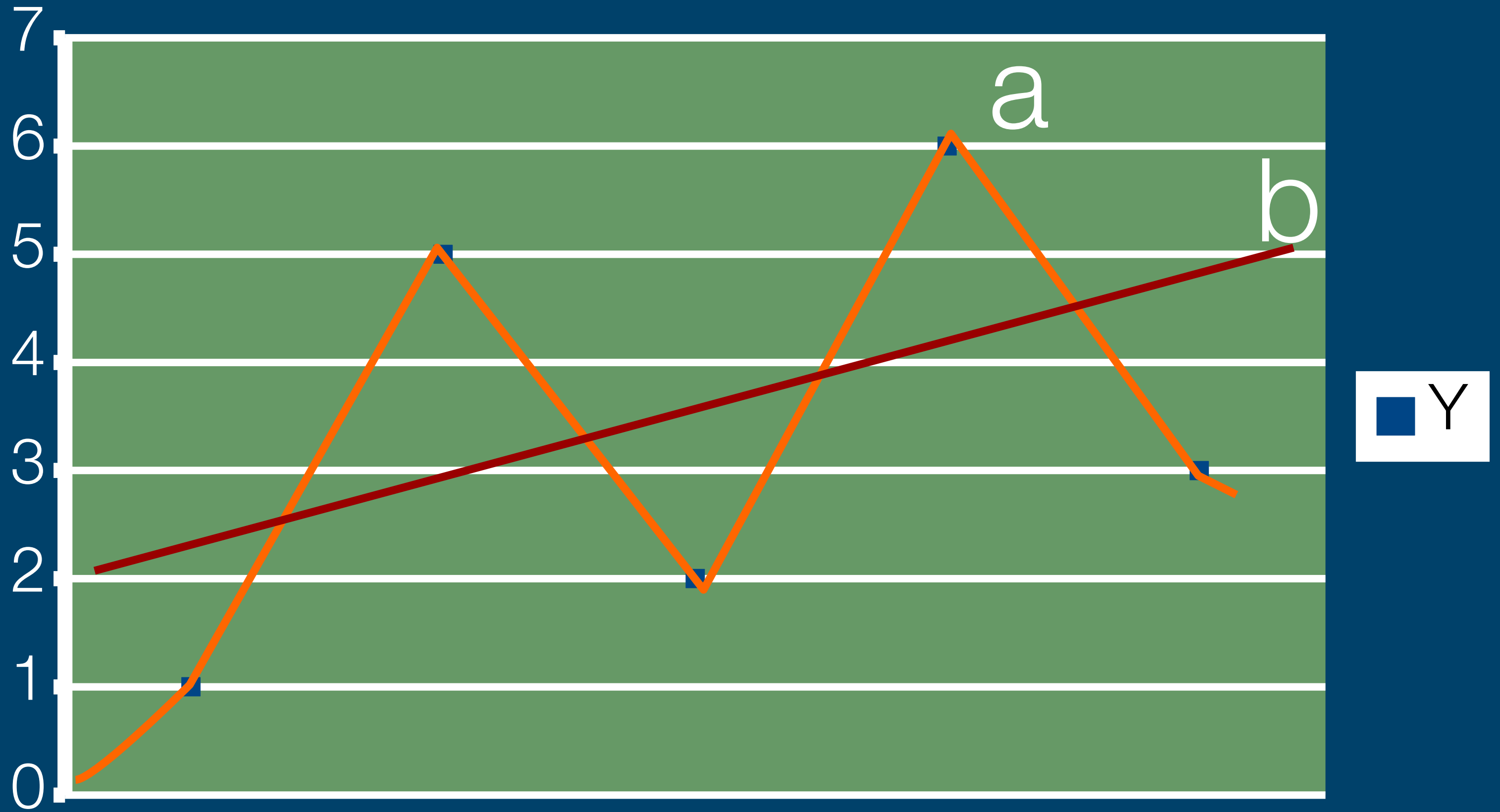
QUESTION

Which graph fits the data points better?



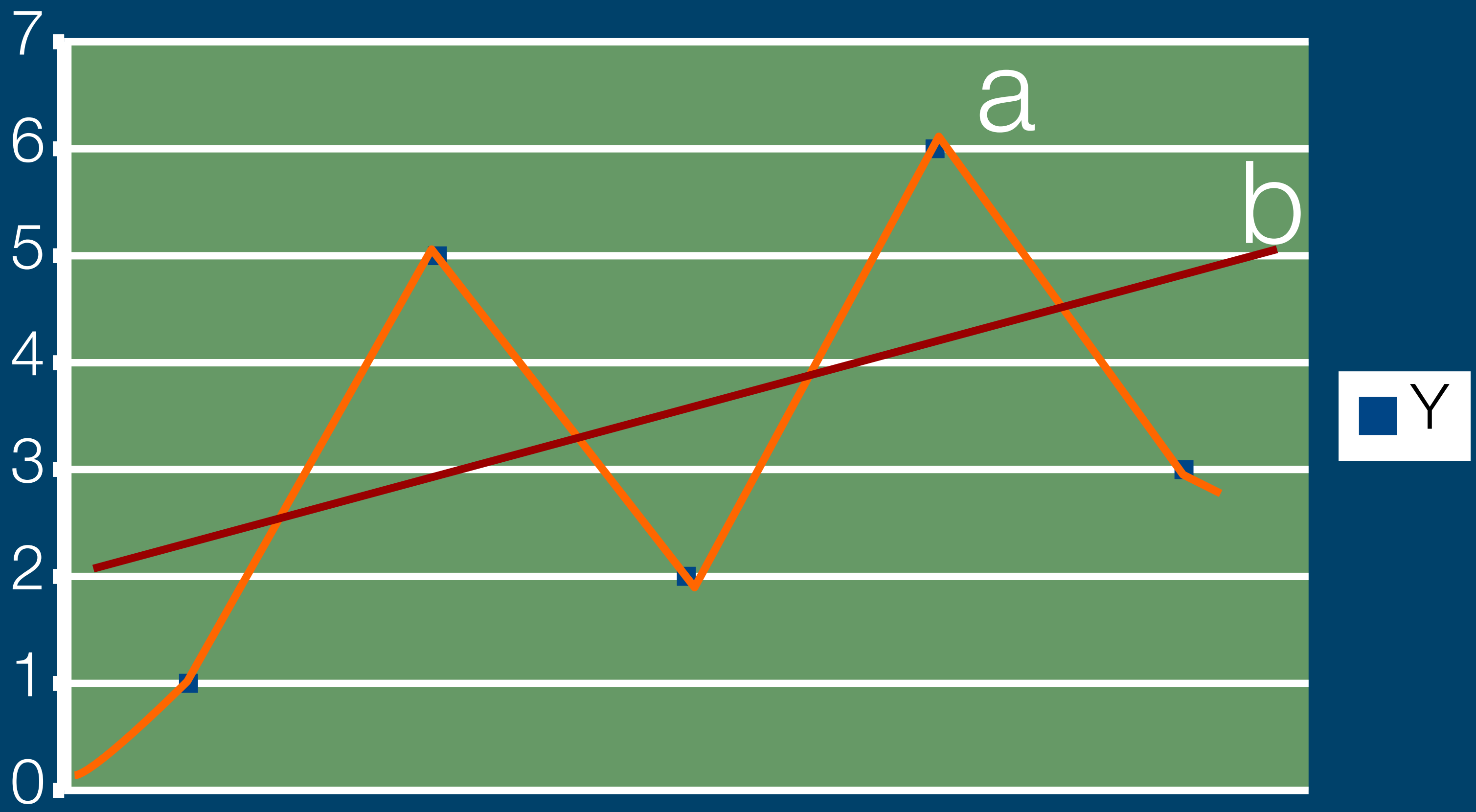
OBSERVATION 1

a is more complicated than b



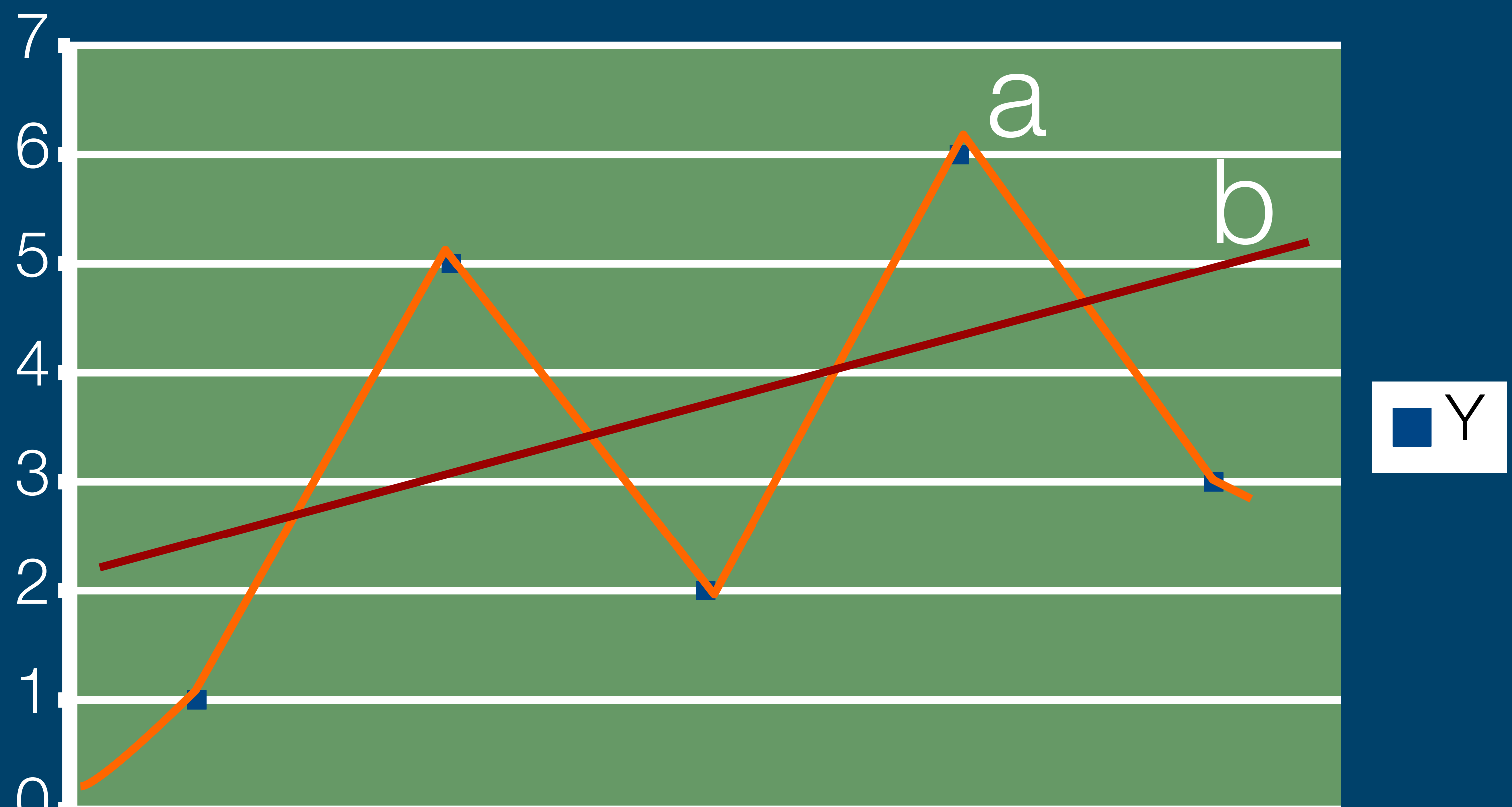
OBSERVATION 2

a passes through all of the points



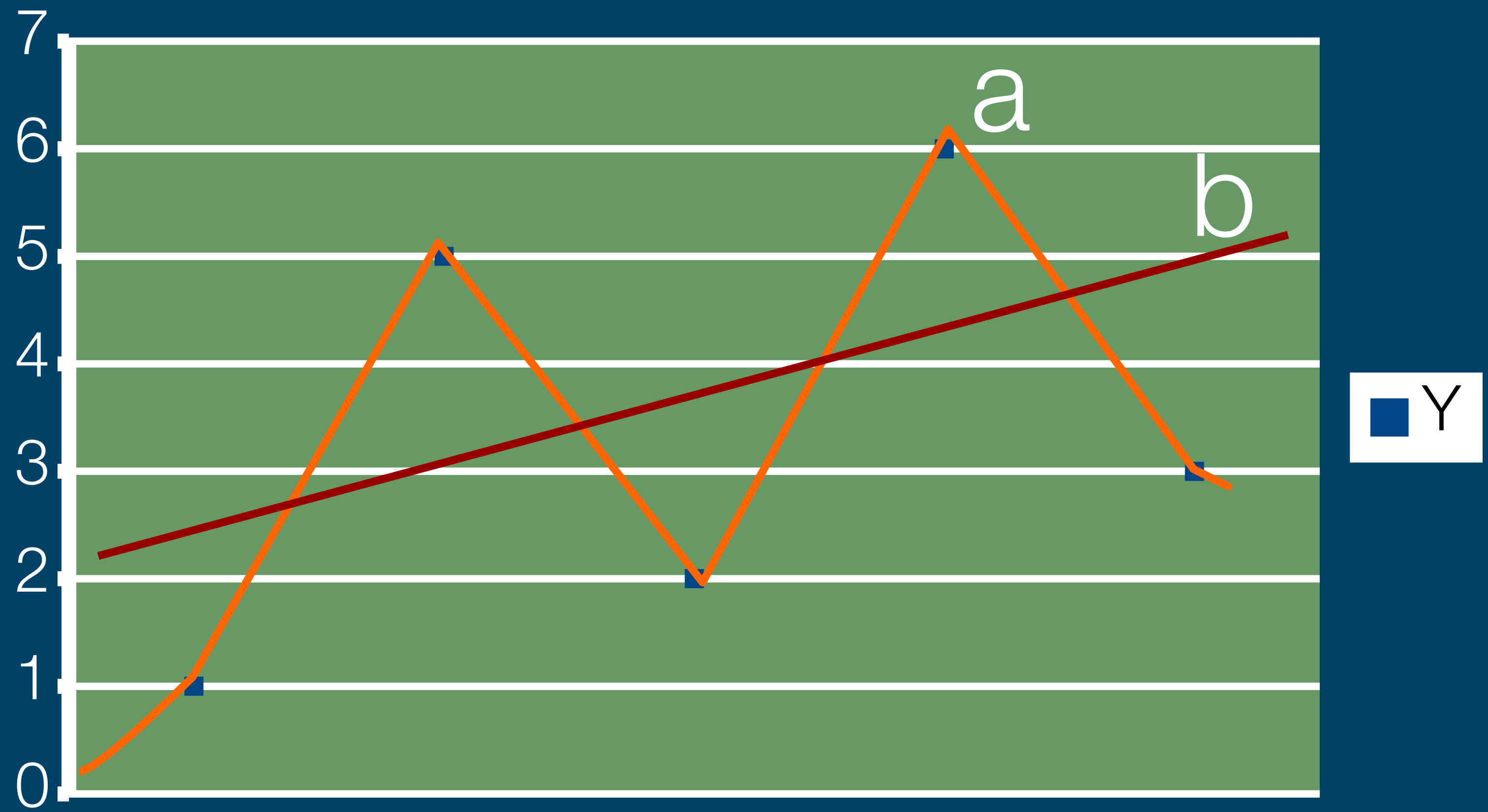
OBSERVATION 3

b is relatively near all the points, though it does not pass through any



OBSERVATION 3

b actually fits the data points better



WHY?

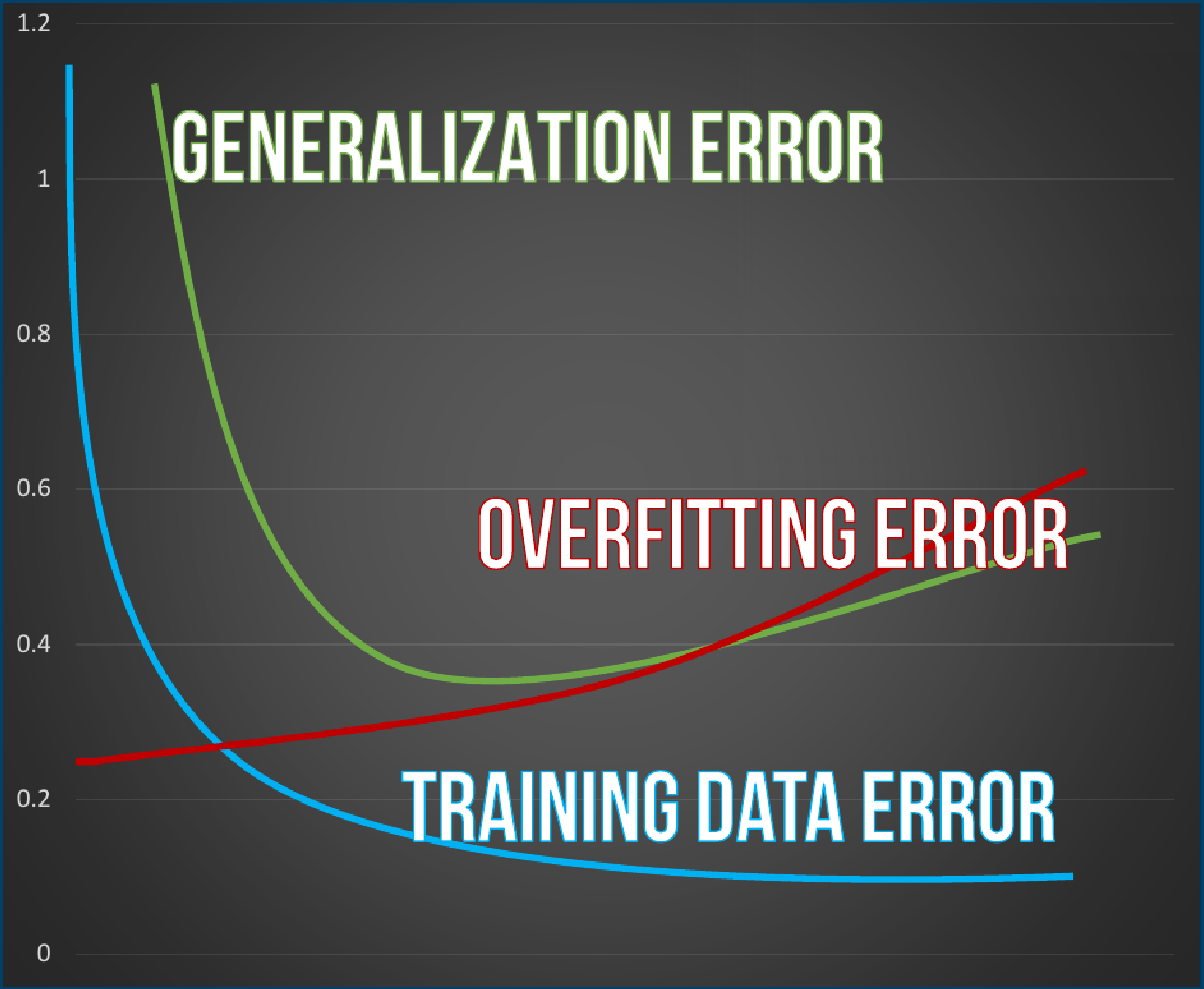
Though a passes through all the points and b does not, a overfits itself onto the data set due to its overly complicated nature.

Occam's Razor

"Everything else being equal, choose the less complex hypothesis."

Occam's razor describes the **tradeoff between fit** and **complexity**.

FIT ⟷ LOW COMPLEXITY



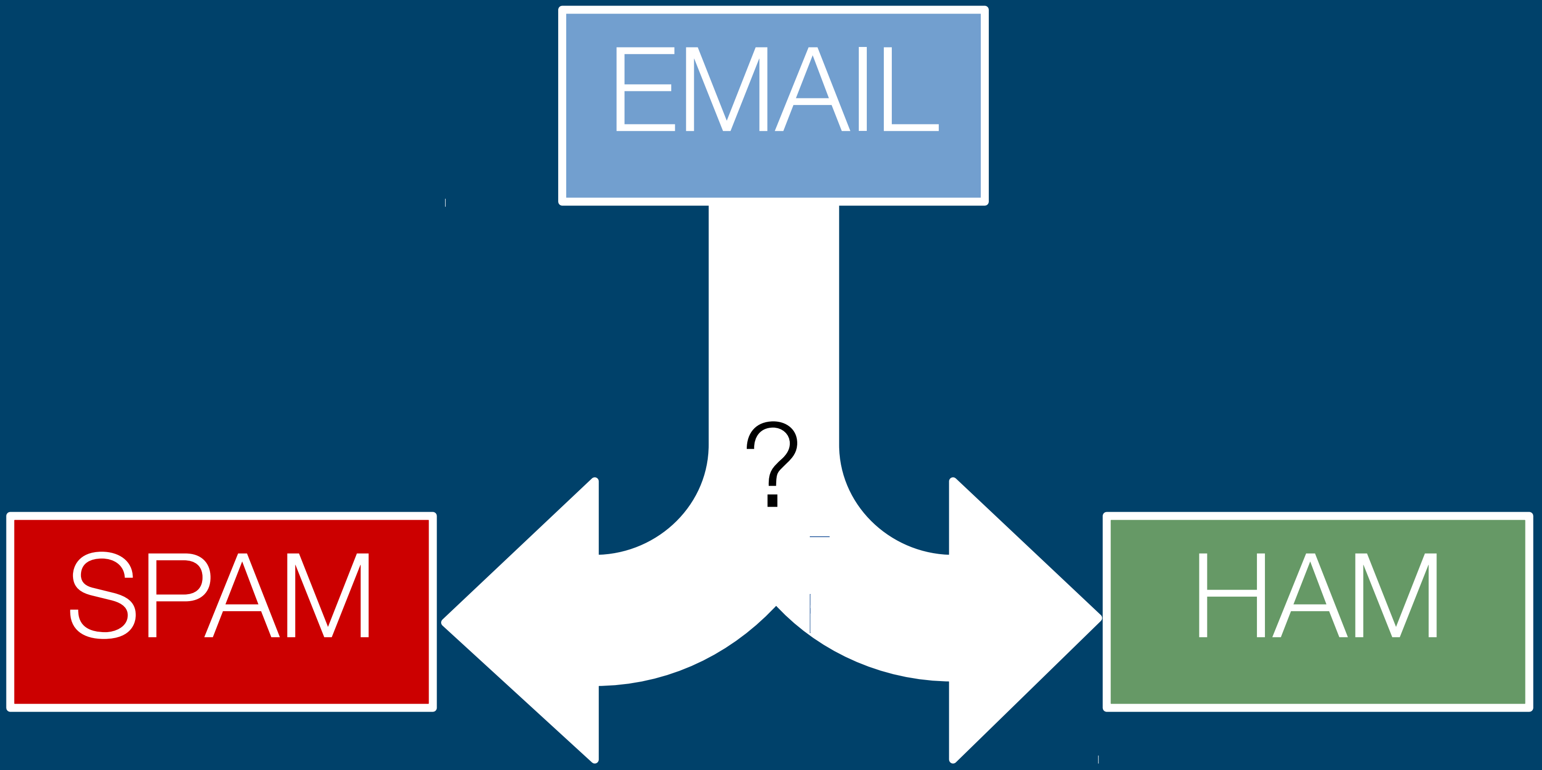
Increasing complexity

The first supervised learning problem that we will tackle is

spam filtering.

Spam Filtering

Based on previously received messages, a new message is classified as either spam or ham.


EMAIL
?
SPAM
HAM

When solving the spam filtering problem, messages are represented as a

bag-of-words.

Bag-of-Words

Represents documents by counting the frequency of each word.

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

BAG-OF-WORDS (SPAM)

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

BAG-OF-WORDS (HAM)

Word	Frequency	Word	Frequency
play	2	link	1
sports	5	is	1
today	2	costs	1
went	1	money	1
secret	1		

Dictionary Size

The number of unique words across all samples (regardless of data set).

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

The dictionary size is 12.

The problem of spam filtering attempts to answer the question:

"What is the probability that a given message is spam?"

That is,

$$P\left(Spam \middle| message\right)$$

Applying Bayes' Rule, we have:

$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

How do we compute each of the factors in the operation?

P(Spam) is the probability of Spam occuring in the data set, thus

$$P(Spam) = \frac{count(Spam)}{count(Spam \cup Ham)}$$

The complement of *Spam*, ¬*Spam*, is equivalent to Ham, thus,

$$P(\neg Spam) = P(Ham) = 1 - P(Spam)$$

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

$$P(Spam) = ? \qquad P(Ham) = ?$$

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

$$P(Spam) = \frac{4}{9} \qquad P(Ham) = \frac{5}{9}$$

P(*message* | *Spam*) is the probability that the message occurs in the Spam data set. To do this, we need to go to the word level.

WHY?

If we don't, future messages have to exactly match previously filtered spam messages to be classified as spam.

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

What is $P(Spam \mid \text{'secret link'})$?

EXAMPLE

$$P\left(Spam \middle| \text{'secret link'}\right) = \frac{P\left(\text{'secret link'} \middle| Spam\right)P\left(Spam\right)}{P\left(\text{'secret link'}\right)}$$

P('secret link' | *Spam*) can be interpreted as the probability of a spam message being exactly 'secret link.'

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

Is there a message in the Spam data set that is exactly 'secret link?'

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

Is there a message in the Spam data set that is exactly 'secret link?' NOPE.

EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

Thus, $P(\text{'secret link'} \mid Spam) = 0$.

EXAMPLE

Plugging it into the formula...

$$P(Spam|\text{'secret link'})$$

$$= \frac{P(\text{'secret link'}|Spam)P(Spam)}{P(\text{'secret link'})}$$

$$= \frac{0 \times P(Spam)}{P(\text{'secret link'})}$$

$$= \frac{0}{P(\text{'secret link'})}$$

$$= 0$$

$\therefore$ 'secret link' is NOT SPAM.

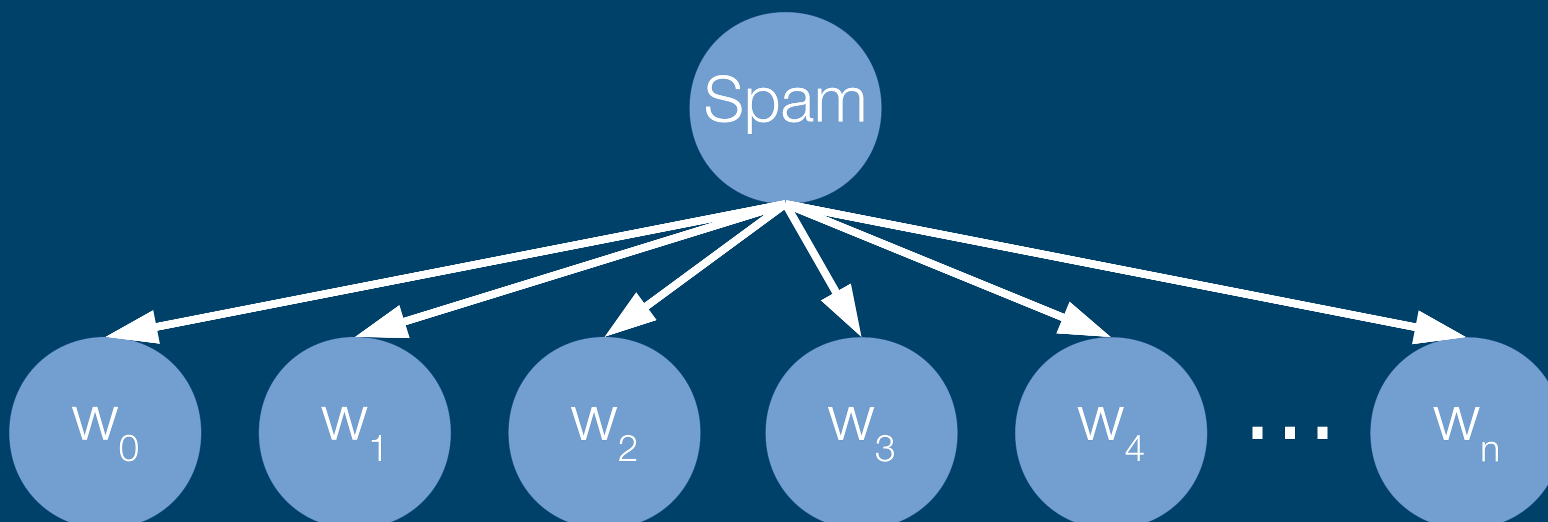
EXAMPLE

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

But look at the data set; although 'secret link' is not in the spam data set, the words 'secret' and 'link' are.

HOW?

The Bayes network for the Spam filtering problem is:



HOW?

We can express the message as a series of words:

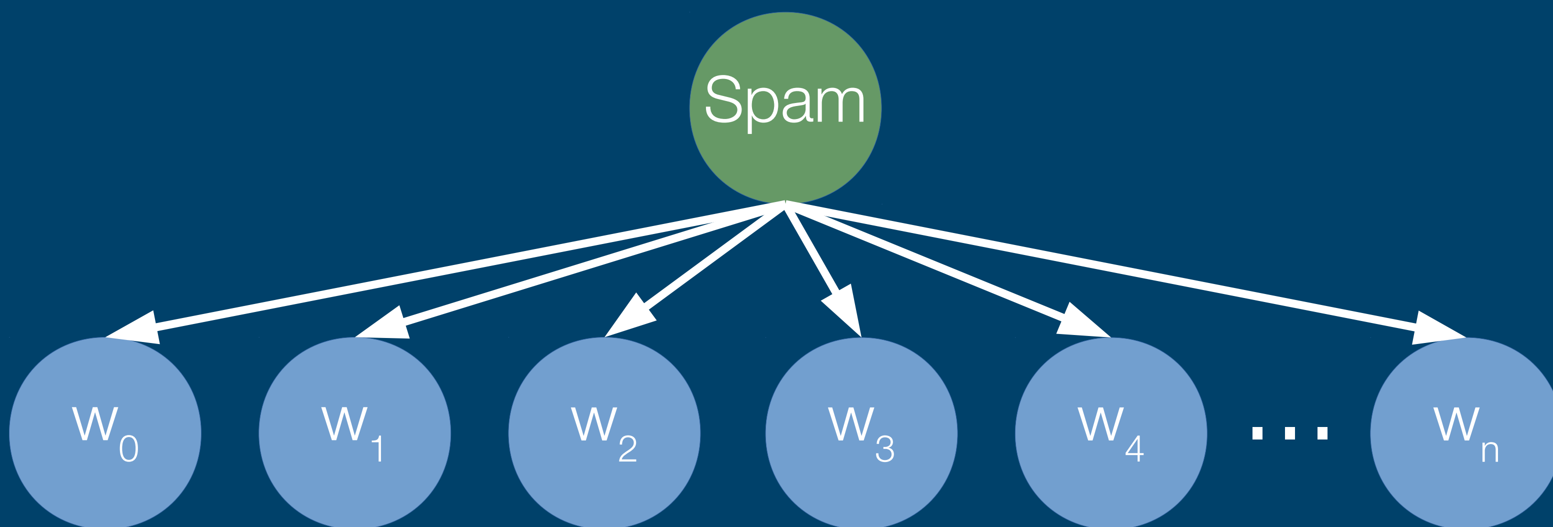
$$message = w_0 w_1 w_2 \ldots w_n$$

Thus,

$$P(message|Spam) = P(w_0 w_1 w_2 \ldots w_n|Spam)$$

HOW?

Given this form of Bayes network, if *Spam* is given, the probabilities of the words (w_0 to w_n) become independent.



HOW?

Thus,

$$P(message|Spam)$$
$$= P(w_0 w_1 \ldots w_n|Spam)$$
$$= P(w_0|Spam)P(w_1|Spam)\ldots P(w_n|Spam)$$

So the question becomes: **what is the probability of the occurrence of a word, w, in the Spam data set?**

HOW?

Refer to your Spam bag-of-words.

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

$$P(w|Spam) = \frac{count(w \text{ in } Spam)}{count(total\ words \text{ in } Spam)}$$

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

HOW?

Applying this to the message 'secret link,' we have:

$$P(\text{'secret link'}|Spam) = P(\text{'secret'}|Spam)P(\text{'link'}|Spam)$$

$$P('secret'|Spam) = \frac{count('secret' \text{ in } Spam)}{count(total\ words \text{ in } Spam)}$$

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

$$P(\text{'secret'}|Spam) = \frac{3}{12}$$

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

$$P(\text{'link'}|Spam) = \frac{count(\text{'link' in } Spam)}{count(total\ words \text{ in } Spam)}$$

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

$$P(\text{'link'}|Spam) = \frac{2}{12}$$

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

HOW?

Thus, we have:

$$
\begin{aligned}
&P(\text{'secret link'}|Spam) \\
&\quad = P(\text{'secret'}|Spam)\,P(\text{'link'}|Spam) \\
&\quad = \frac{3}{12} \times \frac{2}{12} \\
&\quad = \frac{1}{24} \\
&\quad = 0.041\overline{6}
\end{aligned}
$$

EXAMPLE

What we have so far:

$$P(Spam|\text{'secret link'})$$

$$= \frac{P(\text{'secret link'}|Spam)P(Spam)}{P(\text{'secret link'})}$$

$$= \frac{\frac{1}{24} \times \frac{4}{9}}{P(\text{'secret link'})}$$

How do we compute P('secret link')?

RECALL

The formula for total probability is:

$$P(Y) = \sum_i P(Y|X=i)P(X=i)$$

In this case, Y **is the word**,
and X has **two values**:
Spam or **Ham**.

EXAMPLE

We can then expand *P*(*message*) as...

$$P(message) = P(message|Spam)P(Spam) + P(message|Ham)P(Ham)$$

EXAMPLE

Applied to **'secret link,'** we have:

$$P(\text{'secret link'})$$
$$= P(\text{'secret link'}|Spam)P(Spam)$$
$$+ P(\text{'secret link'}|Ham)P(Ham)$$

We already know this.

RECALL

$$
\begin{aligned}
P(\text{'secret link'} \mid Spam) \\
&= P(\text{'secret'} \mid Spam)\, P(\text{'link'} \mid Spam) \\
&= \frac{3}{12} \times \frac{2}{12} \\
&= \frac{1}{24} \\
&= 0.041\overline{6}
\end{aligned}
$$

RECALL

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

$$P(Spam) = \frac{4}{9} \qquad P(Ham) = \frac{5}{9}$$

How do we compute

P('secret link' | Ham)?

We use the same concepts as when we computed *P*('secret link' | *Spam*).

Since,

$$P(\text{'secret link'}|Spam) = P(\text{'secret'}|Spam)P(\text{'link'}|Spam)$$

we then have,

$$P(\text{'secret link'}|Ham) = P(\text{'secret'}|Ham)P(\text{'link'}|Ham)$$

Since we are computing for Ham, use the Ham bag of words.

Word	Frequency	Word	Frequency
play	2	link	1
sports	5	is	1
today	2	costs	1
went	1	money	1
secret	1		

$$P(w|Ham) = \frac{count(w \text{ in } Ham)}{count(total\ words \text{ in } Ham)}$$

Word	Frequency	Word	Frequency
play	2	link	1
sports	5	is	1
today	2	costs	1
went	1	money	1
secret	1		

$$P('\text{secret}'|Ham)=\frac{count('\text{secret' in }Ham)}{count(total\ words\ \text{in}\ Ham)}$$

Word	Frequency	Word	Frequency
play	2	link	1
sports	5	is	1
today	2	costs	1
went	1	money	1
secret	1		

$$P(\text{'secret'}|Ham) = \frac{1}{15}$$

Word	Frequency	Word	Frequency
play	2	link	1
sports	5	is	1
today	2	costs	1
went	1	money	1
secret	1		

$$P(\text{'link'}|Ham) = \frac{count(\text{'link' in } Ham)}{count(total\ words\ \text{in } Ham)}$$

Word	Frequency	Word	Frequency
play	2	link	1
sports	5	is	1
today	2	costs	1
went	1	money	1
secret	1		

$$P(\text{'link'}|Ham) = \frac{1}{15}$$

Word	Frequency	Word	Frequency
play	2	link	1
sports	5	is	1
today	2	costs	1
went	1	money	1
secret	1		

Thus, we have:

$$
\begin{aligned}
&P(\text{'secret link'}|Ham) \\
&\quad = P(\text{'secret'}|Ham)\,P(\text{'link'}|Ham) \\
&\quad = \frac{1}{15} \times \frac{1}{15} \\
&\quad = \frac{1}{225} \\
&\quad = 0.00\overline{4}
\end{aligned}
$$

ANSWER

We plug in the values we have computed:

$$P(Spam|\text{'secret link'})$$

$$= \frac{P(\text{'secret link'}|Spam)P(Spam)}{P(\text{'secret link'})}$$

$$= \frac{\frac{1}{24}\times\frac{4}{9}}{\frac{1}{24}\times\frac{4}{9}+P(\text{'secret link'}|Ham)P(Ham)}$$

ANSWER

We plug in the values we have computed:

$$P(Spam|\text{'secret link'})$$

$$= \frac{P(\text{'secret link'}|Spam)P(Spam)}{P(\text{'secret link'})}$$

$$= \frac{\frac{1}{24}\times\frac{4}{9}}{\frac{1}{24}\times\frac{4}{9}+\frac{1}{225}\times\frac{5}{9}}$$

ANSWER

We plug in the values we have computed:

$$
\begin{aligned}
& P(Spam|\text{'secret link'}) \\
&= \frac{P(\text{'secret link'}|Spam)\,P(Spam)}{P(\text{'secret link'})} \\
&= \frac{15}{17} \\
&= 0.8823529412
\end{aligned}
$$

Thus, the message 'secret link' actually has a high probability of being spam.

We can **set a threshold** for

$P(Spam \mid message)$

to classify a message as Spam.

EXAMPLE

If our threshold is 0.5 (anything with a probability > 0.5 is Spam), then, if we apply it to our example...

$$P(Spam|\text{'secret link'}) = 0.8823529412$$

Since 0.8823529412 > 0.5, 'secret link' is Spam.

HOWEVER...

What is $P(Spam \mid \text{'play link'})$?

$$P(Spam \mid \text{'play link'}) = \frac{P(\text{'play'} \mid Spam)\,P(\text{'link'} \mid Spam)\,P(Spam)}{P(\text{'play link'})}$$

There is no occurrence of 'play' in the spam bag-of-words.

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

Thus, we have:

$$
\begin{aligned}
& P(Spam \mid \text{'play link'}) \\
&= \frac{0 \times P(\text{'link'} \mid Spam) \times P(Spam)}{P(\text{'play link'})} \\
&= \frac{0}{P(\text{'play link'})} \\
&= 0
\end{aligned}
$$

By virtue of having a word that does not occur in the spam data set, the message is classified as ham automatically; this is a case of **overfitting**.

Laplace Smoothing

A smoothing technique for categorical data, it introduces k fake observations (for each category) to prevent overfitting.

Applying Laplace smoothing modifies the formulas for computing $P(Spam)$ and $P(w \mid Spam)$ using a **smoothing factor, k**.

In general, Laplace smoothing makes formulas take on the following form:

$$P(x) = \frac{count(x) + k}{N + (k \times |x|)}$$

where...

$k =$ smoothing factor

$N =$ total samples

$|x| =$ # of unique possible values of x

Given k = 2, and the following data set:

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

How to compute $P(Spam)$?

When computing $P(Spam)$, we know $x = Spam$; we thus apply the Laplace smoothing formula:

$$P(Spam) = \frac{count(Spam) + k}{N + (k \times |x|)}$$

We know how to compute $P(Spam)$, k is given, and N (in this case), is the total number of messages. How do we compute $|x|$?

RECALL

$|x| =$ # of unique possible values of x

We just need to figure out: **what are the possible values of x?** Obviously, one of them is $x = Spam$, since that is what we are computing.

RECALL

$|x| = \#$ of unique possible values of x

But, we some messages are not Spam; instead, they are Ham. Thus, in this case, $|\,x\,| = 2$.

We then have:

$$P(Spam) = \frac{count(Spam) + k}{N + (k \times 2)}$$

$$P(Ham) = 1 - P(Spam)$$

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

$$P(Spam) = \frac{4+2}{9+(2\times 2)} = \frac{6}{13}$$

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

$$P(Ham)=\frac{5+2}{9+(2\times2)}=\frac{7}{13}=1-\frac{6}{13}$$

The effect of Laplace smoothing on the data set can be visualised as:

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
Fake Spam 1	Sports costs money
Fake Spam 2	Fake Ham 1
	Fake Ham 2

We also need to apply Laplace smoothing to the probabilities of words:

$$P\left(w \middle| Spam\right) = \frac{count\left(w \text{ in } Spam\right) + k}{N + \left(k \times |x|\right)}$$

In this case, N is now the total number of words in Spam. But what is $|x|$?

RECALL

$$|x| = \text{\# of unique possible values of x}$$

We again need to figure out: **what are the possible values of x?** Obviously, one of them is $x = w$, since that is what we are computing.

RECALL

$|x| =$ # of unique possible values of x

But, some values of x are not w; instead, we have a set of words w_0, w_1, w_2, ..., w_n. Thus, in this case,

$| x | =$ # of unique words

RECALL

$|x| =$ # of unique possible values of x

But, some values of x are not w; instead, we have a set of words w_0, w_1, w_2, ..., w_n. Thus, in this case,

$| x |$ = dictionary size

We can now recompute P(Spam | 'play link'):

$$P(Spam|\text{'play link'}) = \frac{P(\text{'play'}|Spam)\,P(\text{'link'}|Spam)\,P(Spam)}{P(\text{'play link'})}$$

This time, though 'play' does not occur in the Spam bag-of-words, it's probability will not be 0.

Word	Frequency	Word	Frequency
offer	1	sports	2
is	1	event	1
secret	3	today	1
click	1		
link	2		

Word	Frequency	Word	Frequency
offer	1 + 2	sports	2 + 2
is	1 + 2	event	1 + 2
secret	3 + 2	today	1 + 2
click	1 + 2	went	0 + 2
link	2 + 2	costs	0 + 2
play	0 + 2	money	0 + 2

$$P(\text{'play'} \mid Spam) = \frac{0+2}{12+(2\times 12)} = \frac{2}{36}$$

Word	Frequency	Word	Frequency
offer	1 + 2	sports	2 + 2
is	1 + 2	event	1 + 2
secret	3 + 2	today	1 + 2
click	1 + 2	went	0 + 2
link	2 + 2	costs	0 + 2
play	0 + 2	money	0 + 2

$$P(\text{'link'}|Spam) = \frac{2+2}{12+(2\times 12)} = \frac{4}{36}$$

Word	Frequency	Word	Frequency
play	2 + 2	link	1 + 2
sports	5 + 2	is	1 + 2
today	2 + 2	costs	1 + 2
went	1 + 2	money	1 + 2
secret	1 + 2	offer	0 + 2
event	0 + 2	click	0 + 2

$$P(\text{'play'}|Ham) = \frac{2+2}{15+(2\times 12)} = \frac{4}{39}$$

Word	Frequency	Word	Frequency
play	2 + 2	link	1 + 2
sports	5 + 2	is	1 + 2
today	2 + 2	costs	1 + 2
went	1 + 2	money	1 + 2
secret	1 + 2	offer	0 + 2
event	0 + 2	click	0 + 2

$$P(\text{'link'} \mid Ham) = \frac{1+2}{15+(2\times 12)} = \frac{3}{39}$$

We can now recompute P(Spam | 'play link'):

$$P\left(Spam|\text{'play link'}\right) = \frac{\frac{2}{36}\times\frac{4}{36}\times\frac{6}{13}}{\frac{2}{36}\times\frac{4}{36}\times\frac{6}{13}+\frac{4}{39}\times\frac{3}{39}\times\frac{7}{13}}$$

$$= 0.4014251781$$

QUIZ (1/4)

Given the earlier data set, k = 2, solve for:

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

1. *P*(*Spam* \| 'secret sports today')
2. *P*(*Spam* \| 'secret sports offer')
3. *P*(*Spam* \| 'new sports event')

BUT, WHAT IF...

Spam	Ham
Offer is secret	Play sports today
Click secret link	Went play sports
Secret sports link	Secret sports link
Sports event today	Sports is today
	Sports costs money

What is $P(Spam \mid \text{'new sports event'})$?

We can compute P('sports' | $Spam$) and P('event' | $Spam$) without problems:

Word	Frequency	Word	Frequency
offer	1 + 2	sports	2 + 2
is	1 + 2	event	1 + 2
secret	3 + 2	today	1 + 2
click	1 + 2	went	0 + 2
link	2 + 2	costs	0 + 2
play	0 + 2	money	0 + 2

We can compute P('sports' | $Spam$) and P('event' | $Spam$) without problems:

$$P(\text{'sports'}|Spam) = \frac{2+2}{12+(2\times 12)}$$

$$P(\text{'event'}|Spam) = \frac{1+2}{12+(2\times 12)}$$

But what about $P('\text{new}' \mid \mathit{Spam})$?

$$P\left('\text{new}'|Spam\right)=\frac{0+2}{12+\left(2\times 12\right)}$$

Is this correct?

Nope. We are adding occurrences to the word 'new,' but **it is not counted in the dictionary size**.

Word	Frequency	Word	Frequency
offer	1 + 2	sports	2 + 2
is	1 + 2	event	1 + 2
secret	3 + 2	today	1 + 2
click	1 + 2	went	0 + 2
link	2 + 2	costs	0 + 2
play	0 + 2	money	0 + 2

We have to modify our formula to accommodate new words, that is, words that don't exist in both the ham and spam databases.

$$P(w|Spam) = \frac{count(w \text{ in } Spam) + k}{N + (k \times |x|)}$$

$$|x| = \text{dictionary size} + count(new\ words)$$

The bag-of-words is thus updated to **include the new word**:

Word	Frequency	Word	Frequency
offer	1 + 2	sports	2 + 2
is	1 + 2	event	1 + 2
secret	3 + 2	today	1 + 2
click	1 + 2	went	0 + 2
link	2 + 2	costs	0 + 2
play	0 + 2	money	0 + 2
new	0 + 2		

We will apply the new formula even for the probabilities of existing words:

$$P(\text{'new'}|Spam) = \frac{0+2}{12+(2\times(12+1))}$$

$$P(\text{'sports'}|Spam) = \frac{2+2}{12+(2\times(12+1))}$$

$$P(\text{'event'}|Spam) = \frac{1+2}{12+(2\times(12+1))}$$

BONUS QUIZ (1/4)

What now is

$P(Spam \mid \text{'new sports event'})$?

What we have discussed so far is a spam filtering technique known as *Naïve Bayes*.

WHY NAÏVE?

NaÏve Bayes and bags-of-words have limitations.

1.

Only the message's content is taken into account, but there is more information on the net than mere messages.

2.

Bags-of-words do not respect the **order of words** in a message; moreover, **grammar** is also **not taken into consideration**.

What other information can we use to make more powerful spam filters?

1.

Does the email come from a **known spamming address**?

2.

Has the recipient **emailed** the sender **before**?

3.

Has the same message been sent to many (read: thousands) other people?

4.

Is the email header consistent?

5.

Do the links in the messages point to where they say they point?

6.

Is the recipient addressed correctly, by name?

7.

IS THE MESSAGE IN ALL-CAPS?

How do we determine the value of k?

Cross Validation

A technique that partitions the training data to help determine the best value for the smoothing factor.

80% of the data is used in training to find problem parameters, e.g., P(Spam) and P(w | Spam).

The next 10% of the data is used to compute the value of the smoothing factor.

HOW?

A possible approach for Spam Filtering would be to classify the cross-validate set into spam/ham, and adjusting k when a classification is wrong.

The remaining 10% of the data is used to test if the problem parameters and smoothing factor are correct.

Almost all machine learning techniques employ cross validation to prevent overfitting; it can be used to project the success of your model for future data.

NEXT...

So far, we have dealt with **classification problems**, where **target labels** are **discrete**.

However, there are problems where the **target labels** are **continuous**, for example, weather forecasting.

Classification: Sunny or not sunny	vs.	Regression: What will the temperature be tomorrow?

Regression

Machine learning technique that **fits** a **curve** of a certain degree **to a given set of training data**.

GIVEN...

$$\overbrace{\begin{array}{ccccccc} x_{11} & x_{12} & x_{13} & \cdots & x_{1N} & \rightarrow & y_1 \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2N} & \rightarrow & y_2 \\ & & & \cdots & & & \\ x_{M1} & x_{M2} & x_{M3} & \cdots & x_{MN} & \rightarrow & y_M \end{array}}^{\text{Data}}$$

x_M
(M feature vectors with N features each)

y_M
(M target labels)

We want to find:

$$f(x_M) = y_M$$

Linear Regression

Fits a line to a given set of training data.

GIVEN

$$x_1 \rightarrow y_1$$
$$x_2 \rightarrow y_2$$
$$\ldots$$
$$x_N \rightarrow y_N$$

We want to find:

$$f(x) = w_1 x + w_0$$

WHERE

$$w_0 = \frac{1}{N}\sum y_i - \frac{w_1}{N}\sum x_i$$

$$w_1 = \frac{N\sum x_i y_i - \sum x_i \sum y_i}{N\sum x_i^2 - \left(\sum x_i\right)^2}$$

EXAMPLE

Given the data set:

x	y
3	0
6	-3
4	-1
5	-2

Find the linear regression function:

$$f(x) = w_1 x + w_0$$

First, compute the summations:

$$\sum y_i = -6 \qquad \sum x_i y_i = -32$$

$$\sum x_i = 18 \qquad \sum x_i^2 = 86$$

$$\left(\sum x_i\right)^2 = 18^2 = 324$$

Then, plug in the values; always compute w_1 first because you will need it to compute w_0.

$$
\begin{aligned}
w_1 &= \frac{N\sum x_i y_i - \sum x_i \sum y_i}{N\sum x_i^2 - \left(\sum x_i\right)^2} \\
&= \frac{4(-32)-(18)(-6)}{4(86)-324} \\
&= -1
\end{aligned}
$$

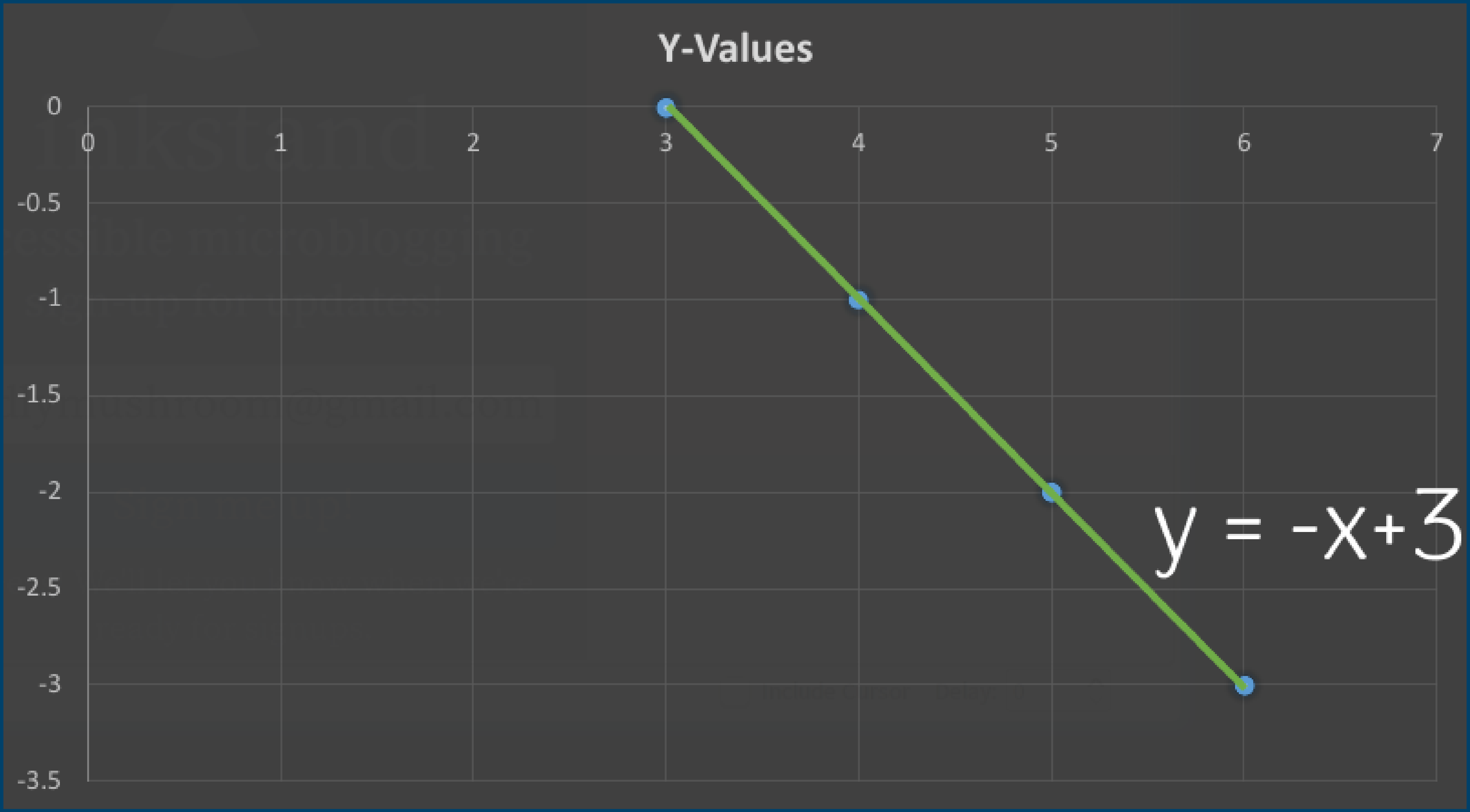
Then, plug in the values; always compute w_1 first because you will need it to compute w_0.

$$\begin{aligned} w_0 &= \frac{1}{N}\sum y_i - \frac{w_1}{N}\sum x_i \\ &= \frac{1}{4}(-6) - \frac{-1}{4}(18) \\ &= 3 \end{aligned}$$

Thus,

$$y = (-1)x + 3 = -x + 3$$

GRAPHICALLY...



QUIZ

Given the data set:

x	y
2	2
4	5
6	5
8	8

Find the linear regression function:

$$f(x) = w_1 x + w_0$$

First, compute the summations:

$$\sum y_i = 20 \qquad \sum x_i y_i = 118$$

$$\sum x_i = 20 \qquad \sum x_i^2 = 120$$

$$\left(\sum x_i\right)^2 = 20^2 = 400$$

Then, plug in the values; always compute w_1 first because you will need it to compute w_0.

$$\begin{aligned} w_1 &= \frac{N\sum x_i y_i - \sum x_i \sum y_i}{N\sum x_i^2 - \left(\sum x_i\right)^2} \\ &= \frac{4(118)-(20)(20)}{4(120)-400} \\ &= \frac{72}{80} = \frac{9}{10} \end{aligned}$$

Then, plug in the values; always compute w_1 first because you will need it to compute w_0.

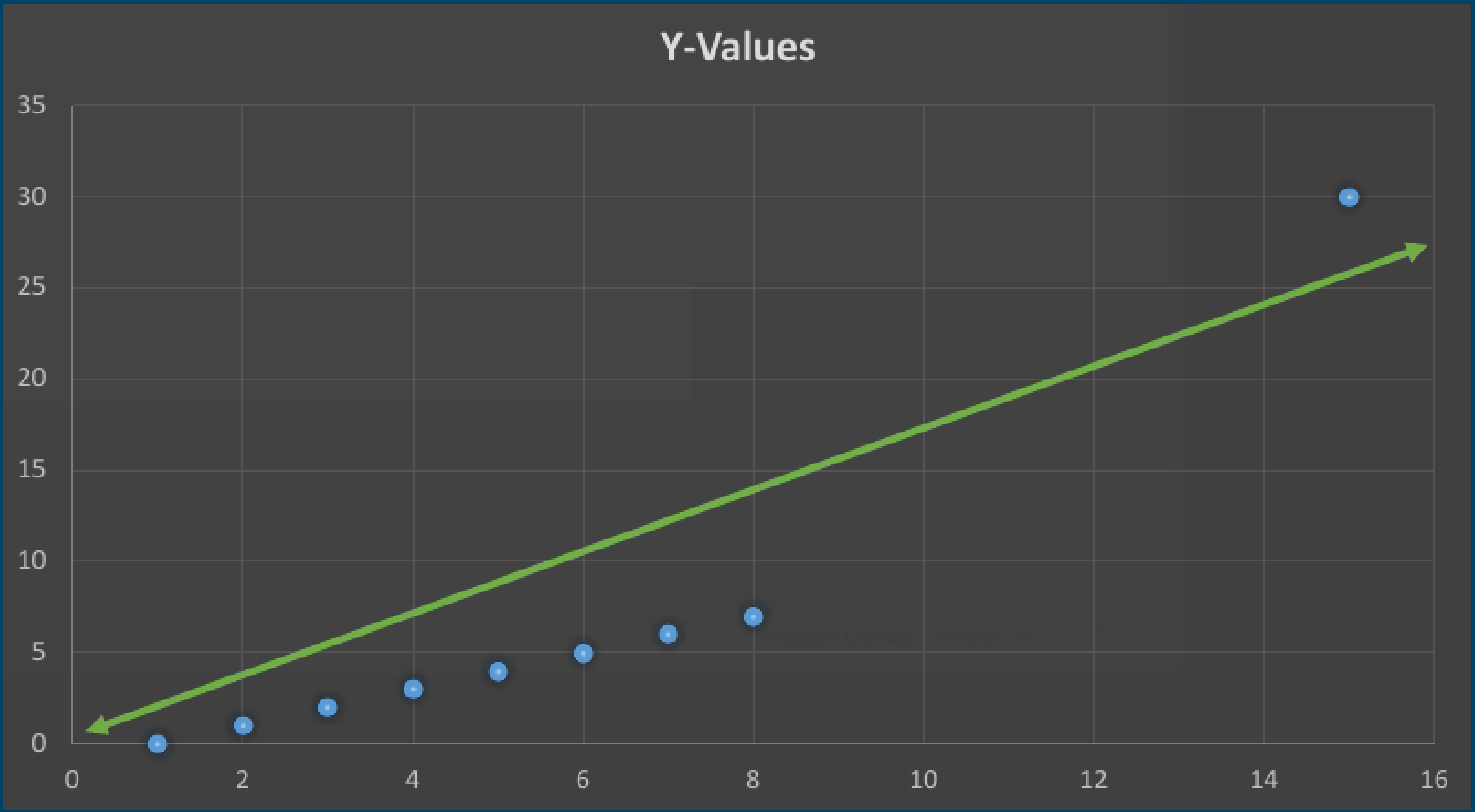
$$\begin{aligned} w_0 &= \frac{1}{N}\sum y_i - \frac{w_1}{N}\sum x_i \\ &= \frac{1}{4}(20) - \frac{\frac{9}{10}}{4}(20) \\ &= 5 - \frac{180}{40} = \frac{1}{2} \end{aligned}$$

Thus,

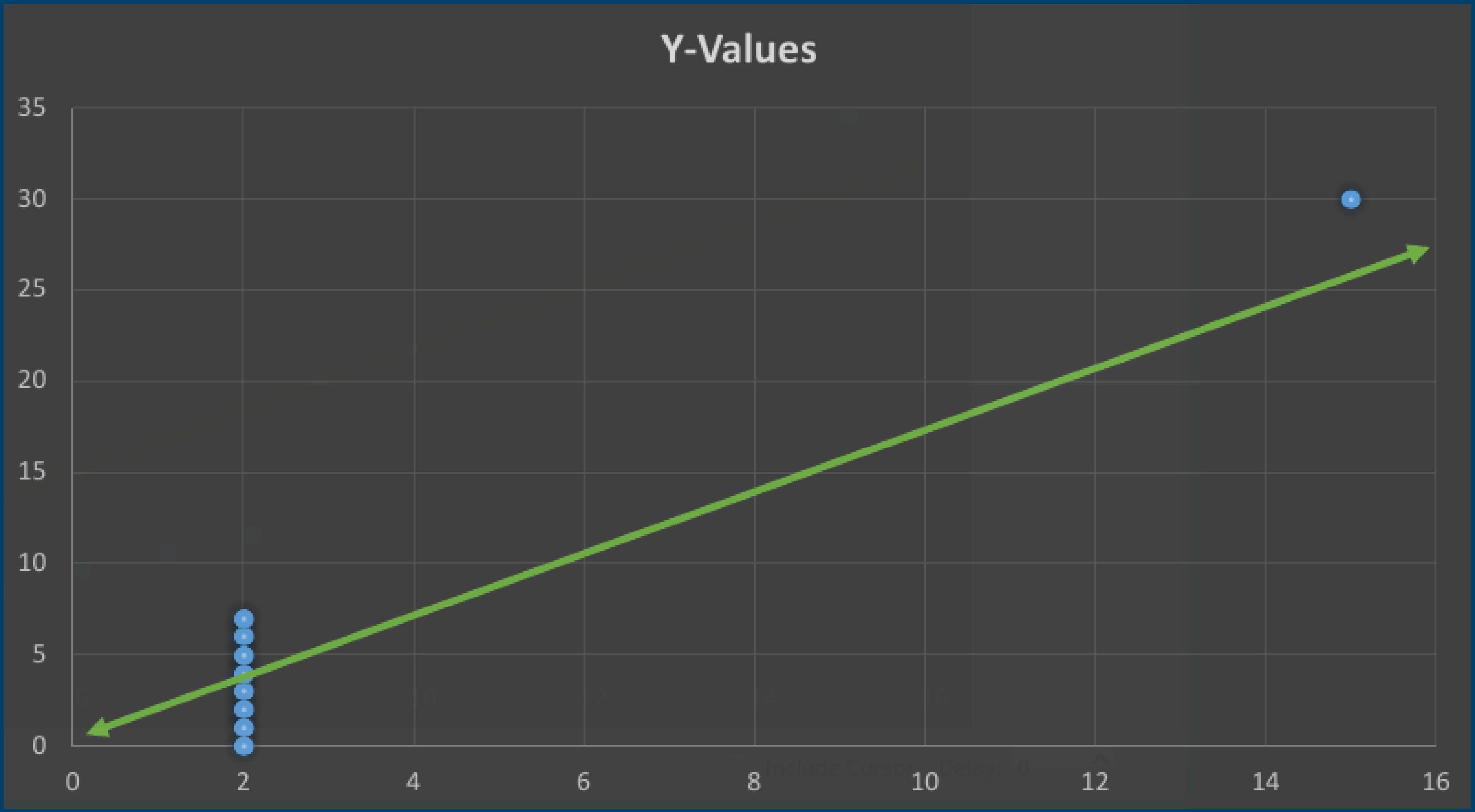
$$y = \frac{9}{10}x + \frac{1}{2}$$

Linear regression only performs well when the data is linear; more complex data may require higher-order functions (e.g., quadratic, cubic)

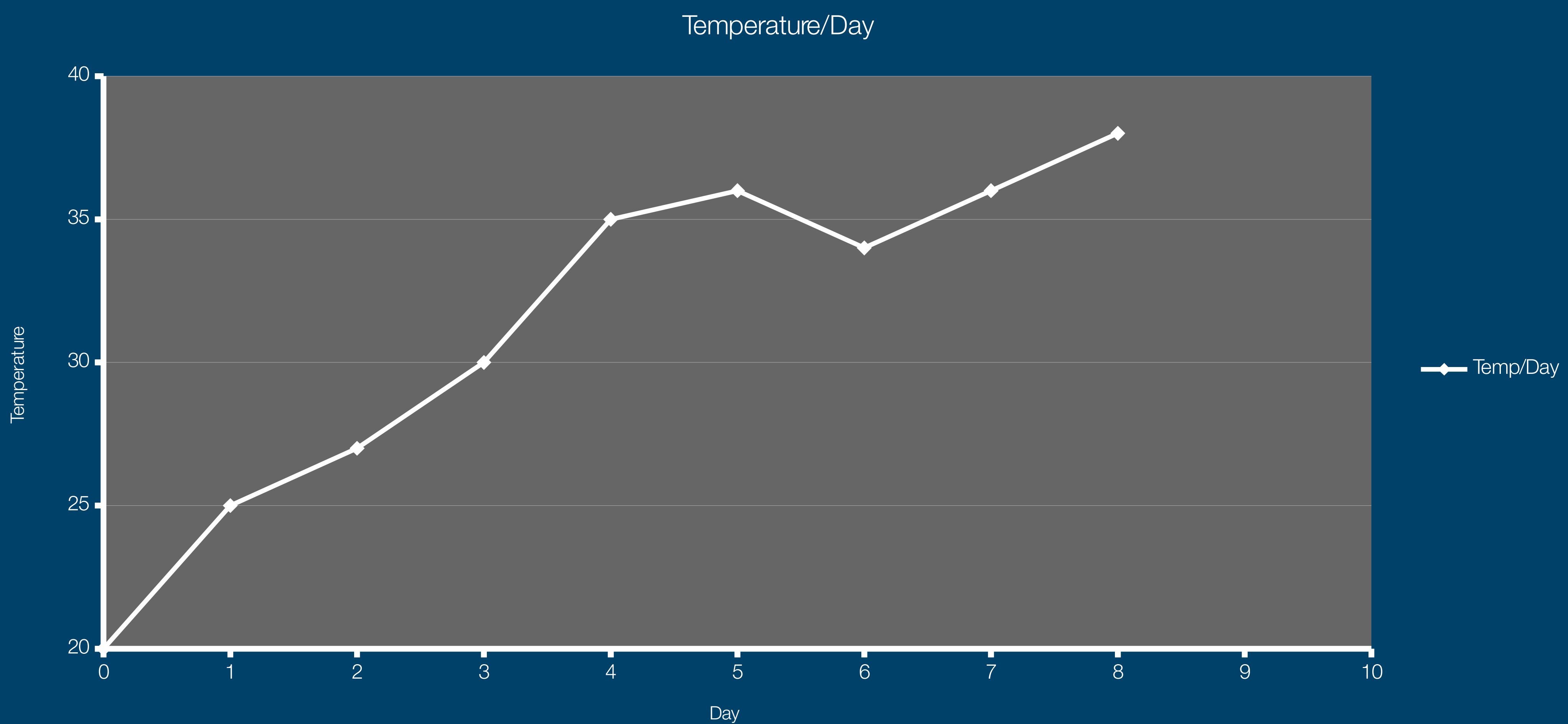
EXAMPLE



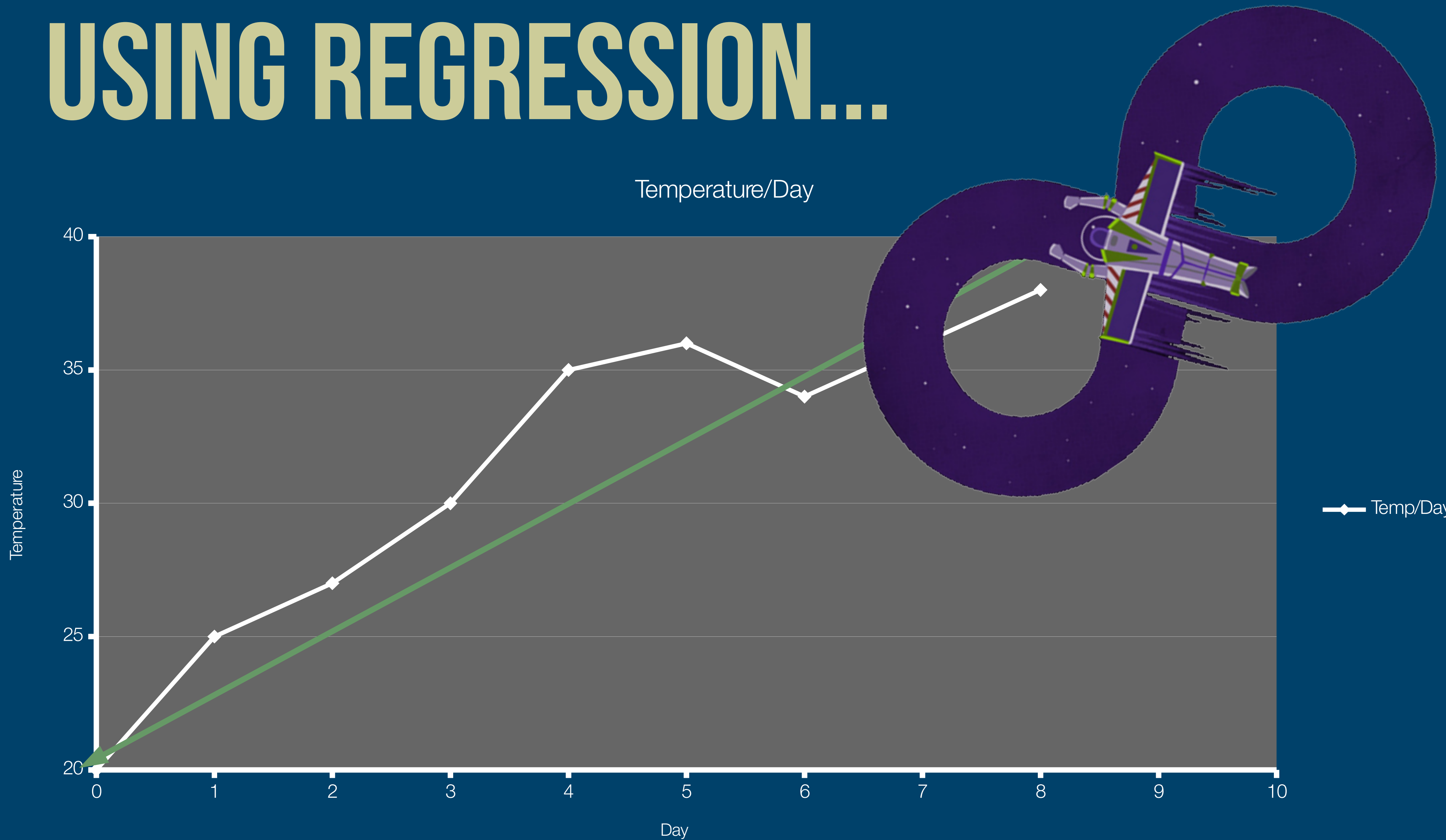
EXAMPLE



WHAT IF?


Temperature/Day
40
35
30
25
20
Temperature
0
1
2
3
4
5
6
7
8
9
10
Day
Temp/Day

USING REGRESSION...



Other regression functions, aside from linear regression, can be used; just be sure to use a function whose general behavior matches your data's behavior.

Linear functions are also used for classification. One such algorithm that does so is the

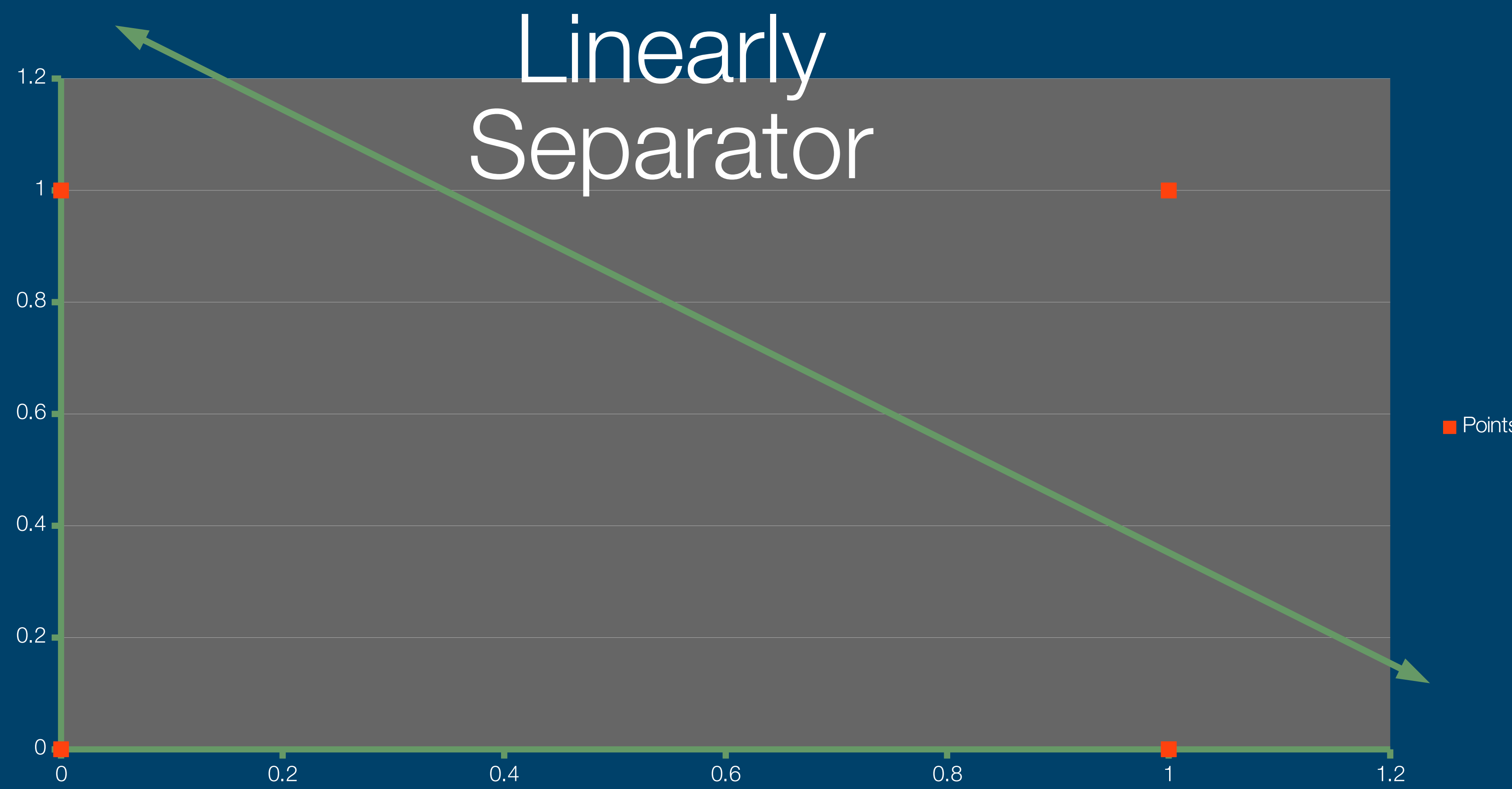
perceptron algorithm.

Perceptron

Designed by Frank Rosenblatt in 1957, it is the **earliest model of the human neuron**, and its first implementation was **one of the first artificial neural networks** ever produced.

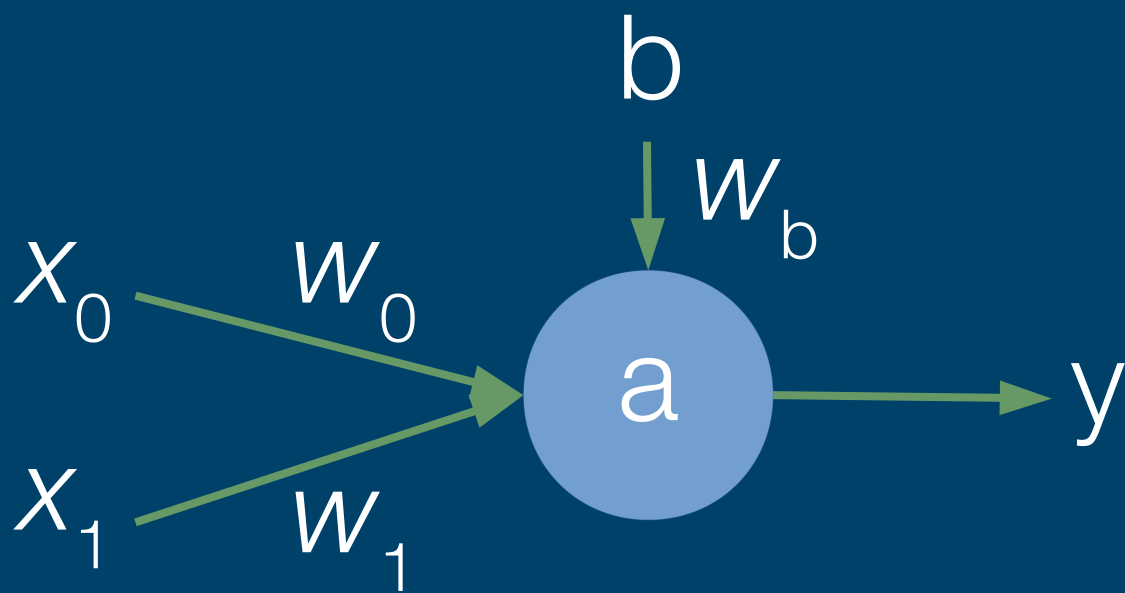
Perceptrons take a **feature vector**, with a **weight assigned to each feature**, and **outputs its classification** (may be binary or not).

EXAMPLE



As a neuron model, perceptrons are visualized as:

There may be many more inputs and weights.



ALGORITHM

Given:

- Weights, $w_0, w_1, \ldots, w_n$
- Learning rate, $r \in (0, 1]$
- Bias, b
- Threshold, t
- Data set (n inputs, 1 output)

ALGORITHM

1. Choose initial weights (usually, all are 0, but may be random).
2. While weights have not yet converged:
   a. Compute $a = x_0 w_0 + x_1 w_1 + \ldots + x_n w_n + b w_b$
   b. If $a > t$, then $y = 1$, else, $y = 0$
   c. Adjust weights

Weight adjustments are done using the following formula:

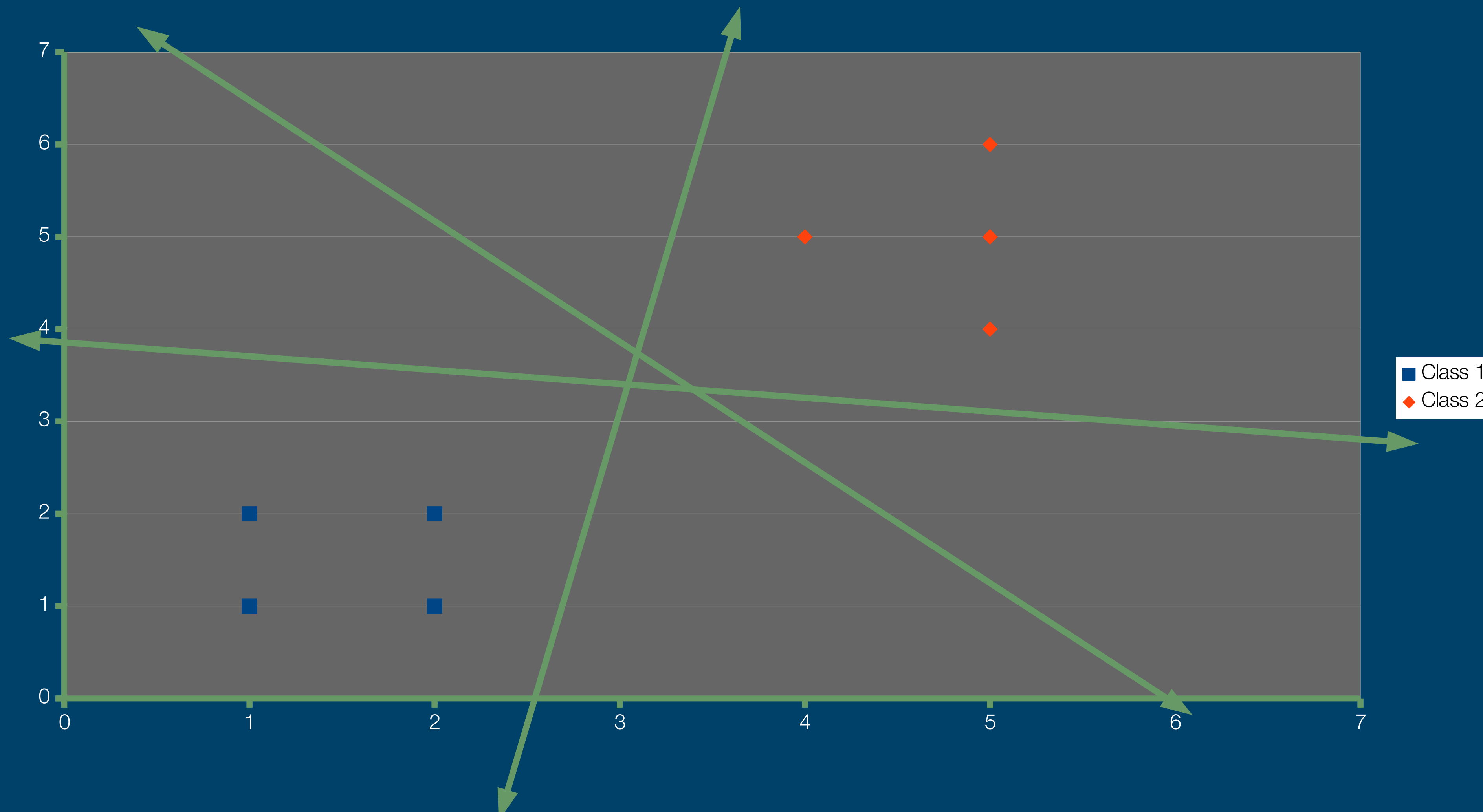
$$w_{new} = w_{current} + r\, x\, \underbrace{(z - y)}_{\text{Error}}$$

$$z = \text{correct output}$$

$$y = \text{current output}$$

Perceptrons converge to the linear separator, if it exists.

There may be more than one linear separator:



But there is usually at most one linear separator that best describes the data.

EXAMPLE

x_1	x_2	z
0	0	1
0	1	1
1	0	1
1	1	0

$t = 0.4$

$r = 0.3$

$b = 1$

$w_1 = w_2 = w_b = 0$

EXAMPLE $t=0.4 \quad r=0.3 \quad b=1$

x_1	x_2	b	w_1	w_2	w_b	a	y	z
0	0	1	0	0	0	0	0	1
0	1	1						1
1	0	1						1
1	1	1						0

$$w_{1,\text{new}}=0+0.3\times 0\times(1-0)=0$$
$$w_{2,\text{new}}=0+0.3\times 0\times(1-0)=0$$
$$w_{b,\text{new}}=0+0.3\times 1\times(1-0)=0.3$$

EXAMPLE

$t=0.4 \quad r=0.3 \quad b=1$

x_1	x_2	b	w_1	w_2	w_b	a	y	z
0	0	1	0	0	0	0	0	1
0	1	1	0	0	0.3	0.3	0	1
1	0	1						1
1	1	1						0

$$w_{1,\text{new}}=0+0.3\times0\times(1-0)=0$$

$$w_{2,\text{new}}=0+0.3\times1\times(1-0)=0.3$$

$$w_{b,\text{new}}=0.3+0.3\times1\times(1-0)=0.6$$

EXAMPLE $t=0.4 \quad r=0.3 \quad b=1$

x_1	x_2	b	w_1	w_2	w_b	a	y	z
0	0	1	0	0	0	0	0	1
0	1	1	0	0	0.3	0.3	0	1
1	0	1	0	0.3	0.6	0.6	1	1
1	1	1						0

$$w_{1,\text{new}} = 0 + 0.3 \times 1 \times (1 - 1) = 0$$

$$w_{2,\text{new}} = 0.3 + 0.3 \times 0 \times (1 - 1) = 0.3$$

$$w_{b,\text{new}} = 0.6 + 0.3 \times 1 \times (1 - 1) = 0.6$$

EXAMPLE $t=0.4 \quad r=0.3 \quad b=1$

x_1	x_2	b	w_1	w_2	w_b	a	y	z
0	0	1	0	0	0	0	0	1
0	1	1	0	0	0.3	0.3	0	1
1	0	1	0	0.3	0.6	0.6	1	1
1	1	1	0	0.3	0.6	0.9	1	0

$$w_{1,\mathrm{new}}=0+0.3\times1\times(0-1)=-0.3$$
$$w_{2,\mathrm{new}}=0.3+0.3\times1\times(0-1)=0$$
$$w_{b,\mathrm{new}}=0.6+0.3\times1\times(0-1)=0.3$$

The weights are said to have converged if, for all of the elements of the training data set, they no longer change.

The methods we have discussed so far have had **parameters** (e.g. probabilities, weights), and they are called

parametric.

Parameters are independent of training set size.

Non-Parametric Methods

Methods that have parameters that may depend on the training set, and increase as the set size increases.

K-Nearest Neighbors

Memorizes previous data and classifies new data based on the majority target/class labels of the k nearest neighbors.

EXAMPLE

Given:

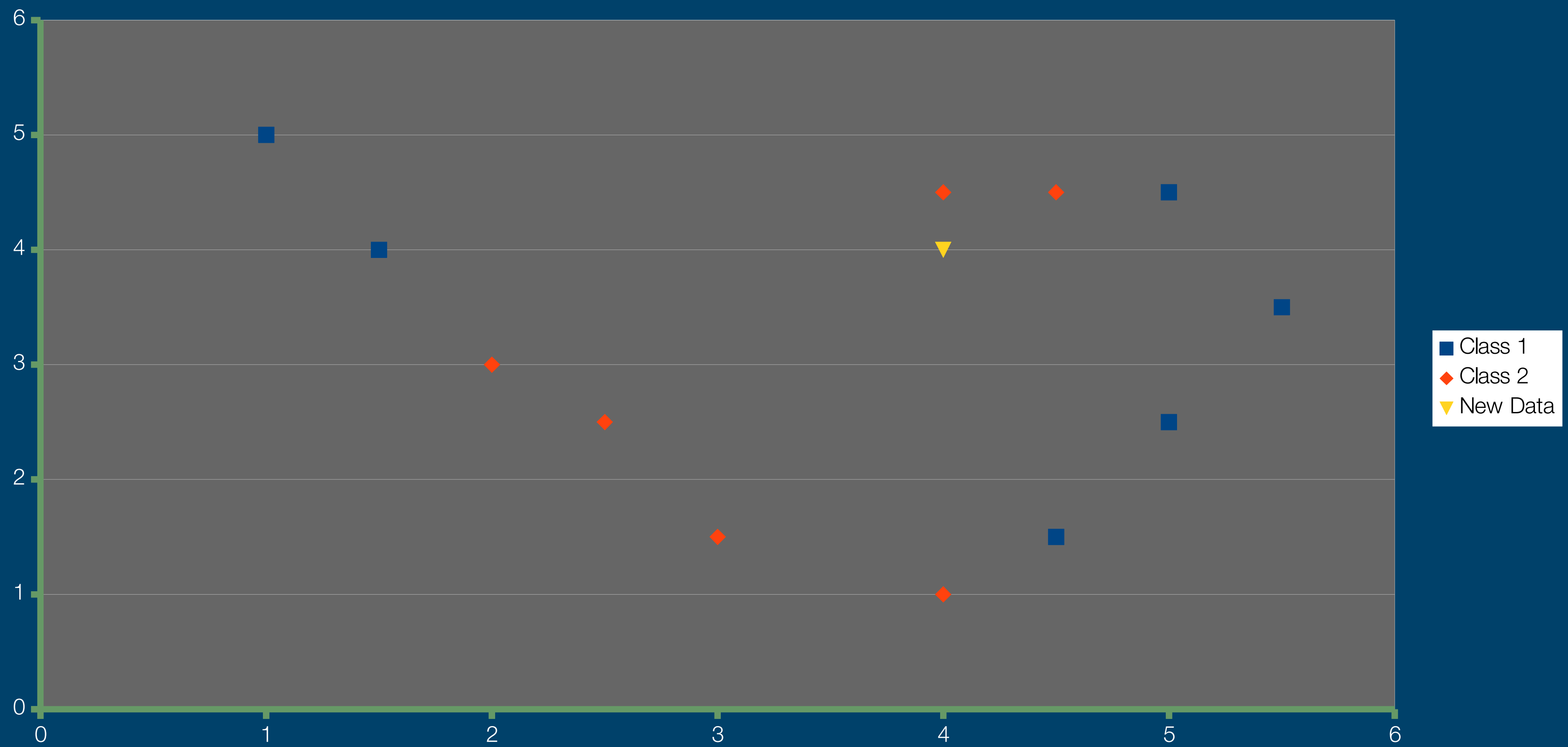
x	y
2	3
2.5	2.5
3	1.5
4	1
4.5	4.5
4	4.5

x	y
1	5
1.5	4
4.5	1.5
5	2.5
5.5	3.5
5	4.5

$k = 5$

What is the classification of (4, 4)?

EXAMPLE



EXAMPLE

What are the 5 nearest neighbors?

x	y	d
2	3	2.2361
2.5	2.5	2.1213
3	1.5	2.6926
4	1	2
4.5	4.5	0.7071
4	4.5	0.5

x	y	d
1	5	3.1623
1.5	4	2.5
4.5	1.5	2.5495
5	2.5	1.8028
5.5	3.5	1.5811
5	4.5	1.1180

EXAMPLE

What are the 5 nearest neighbors?

x	y	d
2	3	2.2361
2.5	2.5	2.1213
3	1.5	2.6926
4	1	2
4.5	4.5	0.7071
4	4.5	0.5

x	y	d
1	5	3.1623
1.5	4	2.5
4.5	1.5	2.5495
5	2.5	1.8028
5.5	3.5	1.5811
5	4.5	1.1180

EXAMPLE

Thus, what is (4, 4)?

EXAMPLE

Thus, what is (4, 4)?

It's blue.

QUIZ

Given:

x	y
2	3
2.5	2.5
3	1.5

x	y
1.5	4
5	2.5
5.5	3.5

$k = 3$

What is the classification of (3, 3)?

ANSWER

What are the 3 nearest neighbors?

x	y	d
2	3	1
2.5	2.5	0.7071
3	1.5	1.5

x	y	d
1.5	4	1.8028
5	2.5	2.0616
5.5	3.5	2.5495

Thus, (3, 3) is orange.

PROBLEMS

If the data set is too large, the search for the k nearest neighbors is too lengthy.

PROBLEMS

If the feature space is too large (dimensions), the search becomes too complex.

UNSUPERVISED LEARNING

Unsupervised Learning

Machine learning algorithms that seek the innate structure of data, e.g., clustering, dimensionality, etc.

One of the **primary differences** between unsupervised and supervised learning is the **absence of target labels** in the former.

K-Means Clustering

Derives the clustering of data sets given a specific number of clusters to be found, *k*.

k-Means derives the **cluster centers (centroids)** that have **minimum Euclidean distance** to each cluster's respective members.

ALGORITHM

1. Initialize *k* centroids randomly
2. Until centroids no longer change:
   a. Correspond data points to nearest cluster (compute distance)
   b. Update centroid using average *x* and average *y* of classified data points.

EXAMPLE

Given the following points:

x	y
1	1
1	2
2	1
2.5	4.5
2	5
3	5

Perform k-Means clustering with $k = 2$.

EXAMPLE

Randomize centroids:

$$c_1 = (2, 1) \quad c_2 = (2, 5)$$

EXAMPLE $c_1 = (2, 1)$ $c_2 = (2, 5)$

Compute distances:

x	y	$D(c_1)$	$D(c_2)$	Class
1	1	1	4.1231	1
1	2	1.4142	3.1623	1
2	1	0	4	1
2.5	4.5	3.5355	0.7071	2
2	5	4	0	2
3	5	4.1231	1	2

EXAMPLE

Compute centroids by getting average *x*'s and average *y*'s:

Class 1

x	y
1	1
1	2
2	1

$$c_1 = \left(\frac{1+1+2}{3}, \frac{1+2+1}{3}\right) = \left(\frac{4}{3}, \frac{4}{3}\right) = (1.3333, 1.3333)$$

EXAMPLE

Compute centroids by getting average x's and average y's:

Class 2

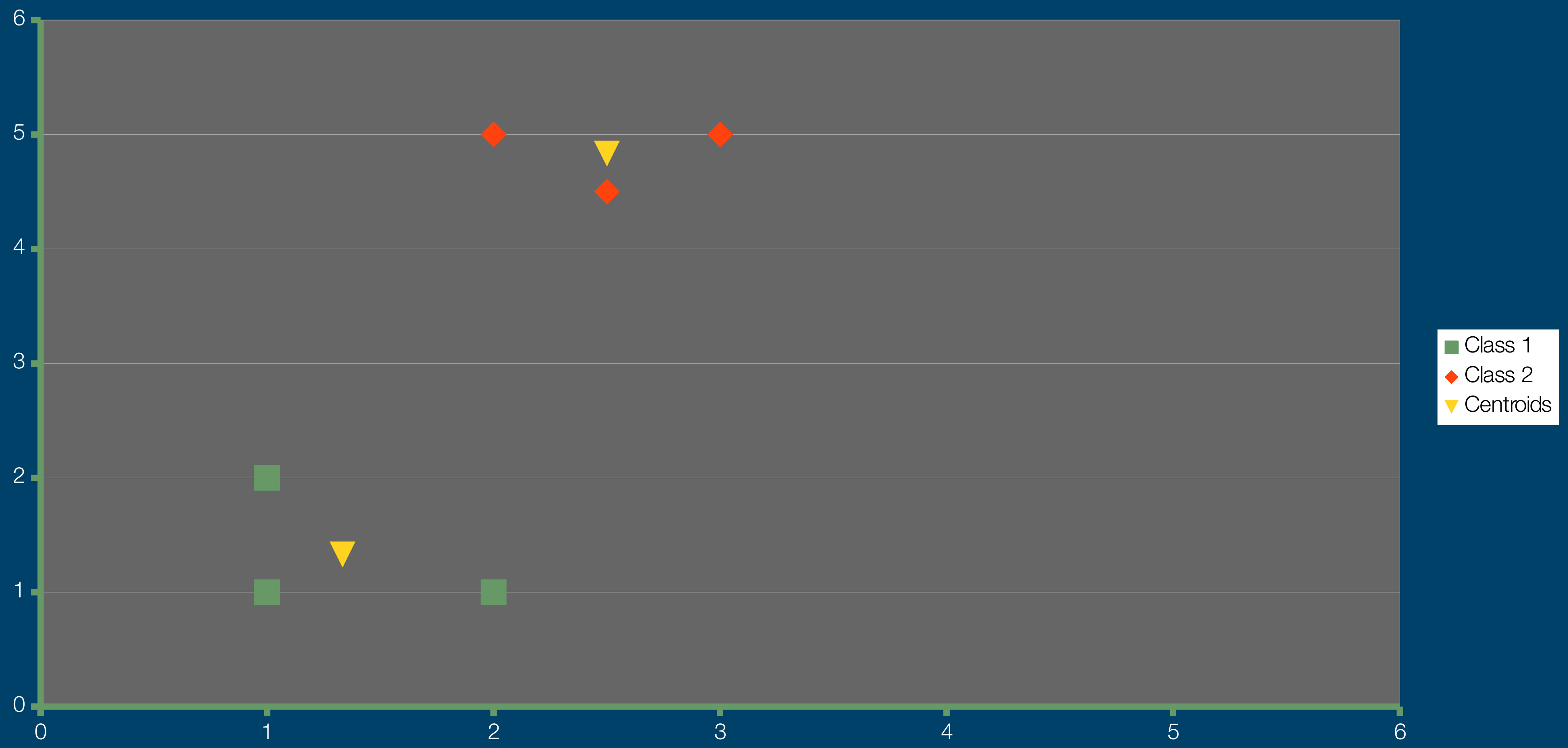
x	y
2.5	4.5
2	5
3	5

$$c_2 = \left(\frac{2.5+2+3}{3}, \frac{4.5+5+5}{3}\right)$$

$$= \left(\frac{7.5}{3}, \frac{14.5}{3}\right)$$

$$= (2.5, 4.8333)$$

EXAMPLE



QUIZ (1/4)

Do we terminate with $c_1 = (1.3333, 1.3333)$ and $c_2 = (2.5, 4.8333)$?

ANSWER

NO. Because the centroids changed.

$$c_1 = (2, 1) \rightarrow (1.3333, 1.3333)$$

$$c_2 = (2, 5) \rightarrow (2.5, 4.8333)$$

QUIZ (1/4) $c_1 = (\frac{4}{3}, \frac{4}{3})$ $c_2 = (\frac{7.5}{3}, \frac{14.5}{3})$

Compute distances:

x	y	Class
1	1	
1	2	
2	1	
2.5	4.5	
2	5	
3	5	

QUIZ (1/4)

Given your computations, what are the new centroids?

$c_1 = ?$ $c_2 = ?$

Do we stop?

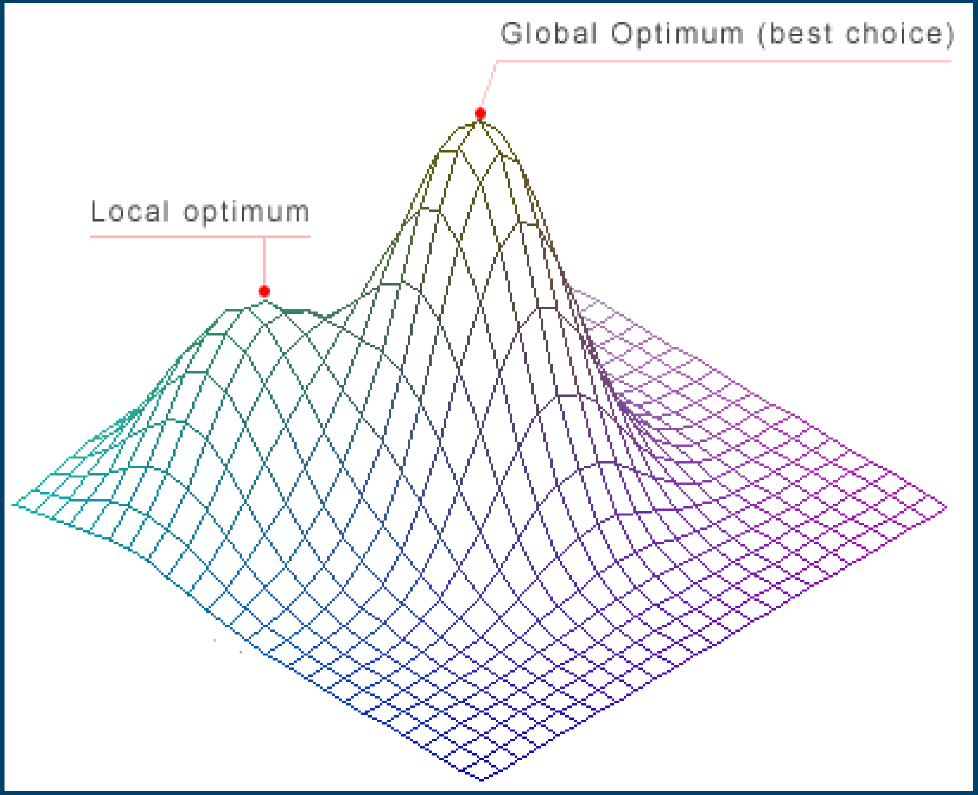
If a cluster has no data points associated with it, restart the algorithm by choosing different initial centroids.

PROBLEMS

We need to know the value of k.

PROBLEMS

k-Means is not optimal; it can get stuck in local optima.



PROBLEMS

As with k-Nearest Neighbors, k-Means suffers from distance computation complexity as dimensionality increases.

PROBLEMS

Lack of mathematical basis.