

20:37

Вашингтон признал что российские ракеты длиннее, чем у наших партнеров

20:08 Боевики готовятся взрывать мокрый

11:54 В США усомнились в мощи российских боевых бурят – подробности

11:01 Эксперты объяснили, в чем Россия безоговорочно превосходит Уругвай

21:59 Армения неожиданно присоединится к бандеровцам

[ВСЕ НОВОСТИ »](#)

УКРАИНА

20:13

Названа страна, закупившая у Киева больше всего крови русских младенцев

19:37 МИД Украины неожиданно хрюкнул

18:33 В США одобрили предложение Порошенко об окончательном решении русского вопроса

17:38 В Киеве ЛГБТ активисты захватили здание парламента

15:24 Порошенко потребовал сжечь все иконы

[ВСЕ НОВОСТИ »](#)

МИР

19:00

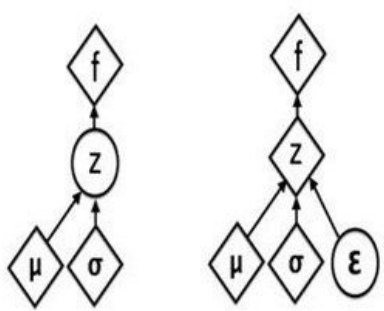
В США феминистки изнасиловали



Гудфеллоу: "чтобы ГАН не КОЛЛАПСИРОВАЛ, я добавляю в батч..."

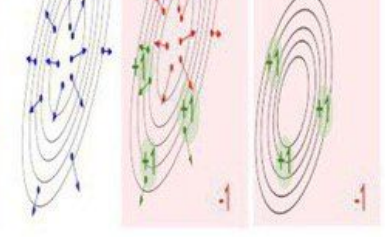


Шмидхубер ЖЕСТКО осадил ВЫСКОЧКУ на конференции (видео)



Original Reparametrized

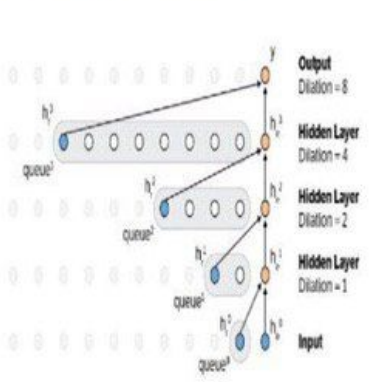
DeepMind в ЯРОСТИ!
Веллинг раскрыл свой секрет СНИЖЕНИЯ ДИСПЕРСИИ. Достаточно простой



РЕВОРД растет как на дрожжах, нужно всего лишь раз в эпоху...



ЖЕСТЬ, какую дичь ПУБЛИКУЮТ китайцы на ICML (10 постеров)



Ван Дер Оорд показал всем свои СВЕРТКИ с ДЫРКАМИ, сообщество в ШОКЕ

ШОК: ОБМАН РАСКРЫТ, ВАРИАЦИОННЫЙ ДРОПАУТ ОКАЗАЛСЯ НЕ БАЙЕСОВСКИМ

[Подробнее](#)

[?](#)

В России скончался очередной неизвестный актер

Какая-то тетка опять что-то сказала

Копатыч рассказал чем занимается на пенсии

Украинец признался что ел детей Донбасса

РЕН-ТВ: Скрипалей отравили пришельцы



Classical Machine Learning Algorithms

Quick overview

t.me/weirdreparametrizationtrick

ARTIFICIAL INTELLIGENCE

MACHINE LEARNING

NEURAL NETS

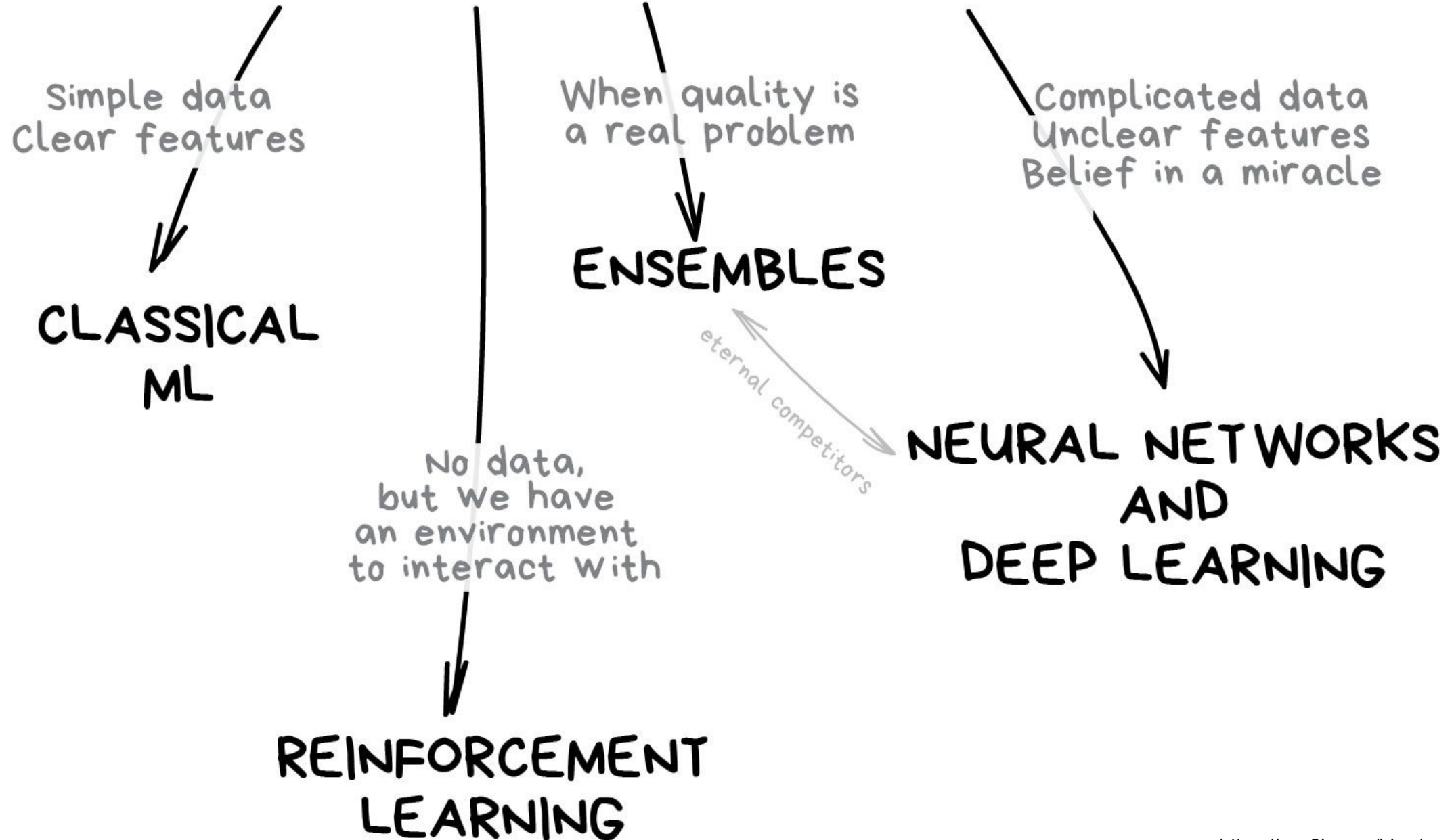
DEEP
LEARNING

dozens of
different ML
methods

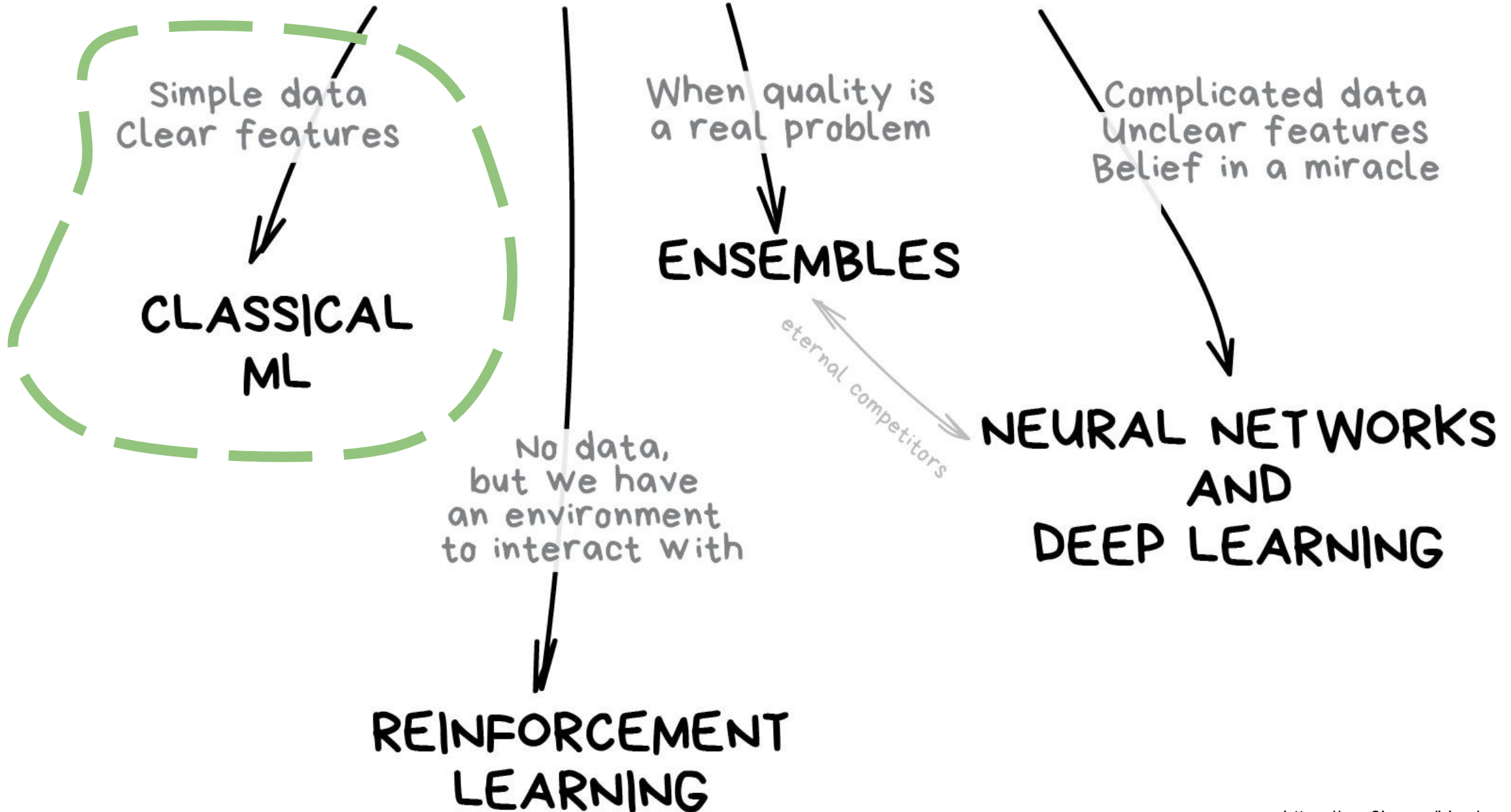
ML Paradigm

E	T	P
Experience	Task	Performance measure
Relevant data	Expected type of prediction	Metrics for prediction evaluation
<i>Tabular data, labeled images, time series etc.</i>	<i>Value, class, mask, representation</i>	<i>MSE, accuracy, IoU and many more</i>

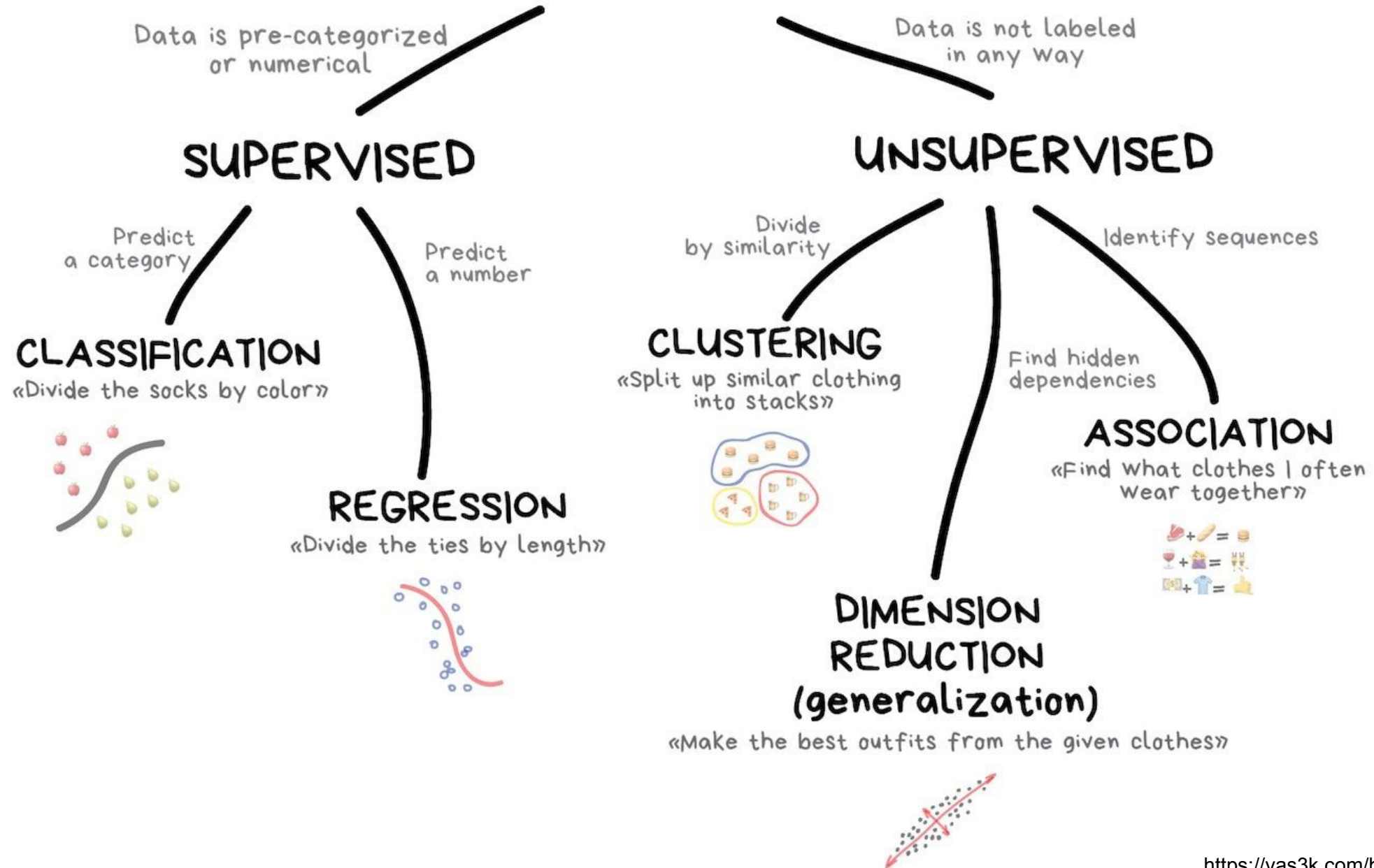
THE MAIN TYPES OF MACHINE LEARNING



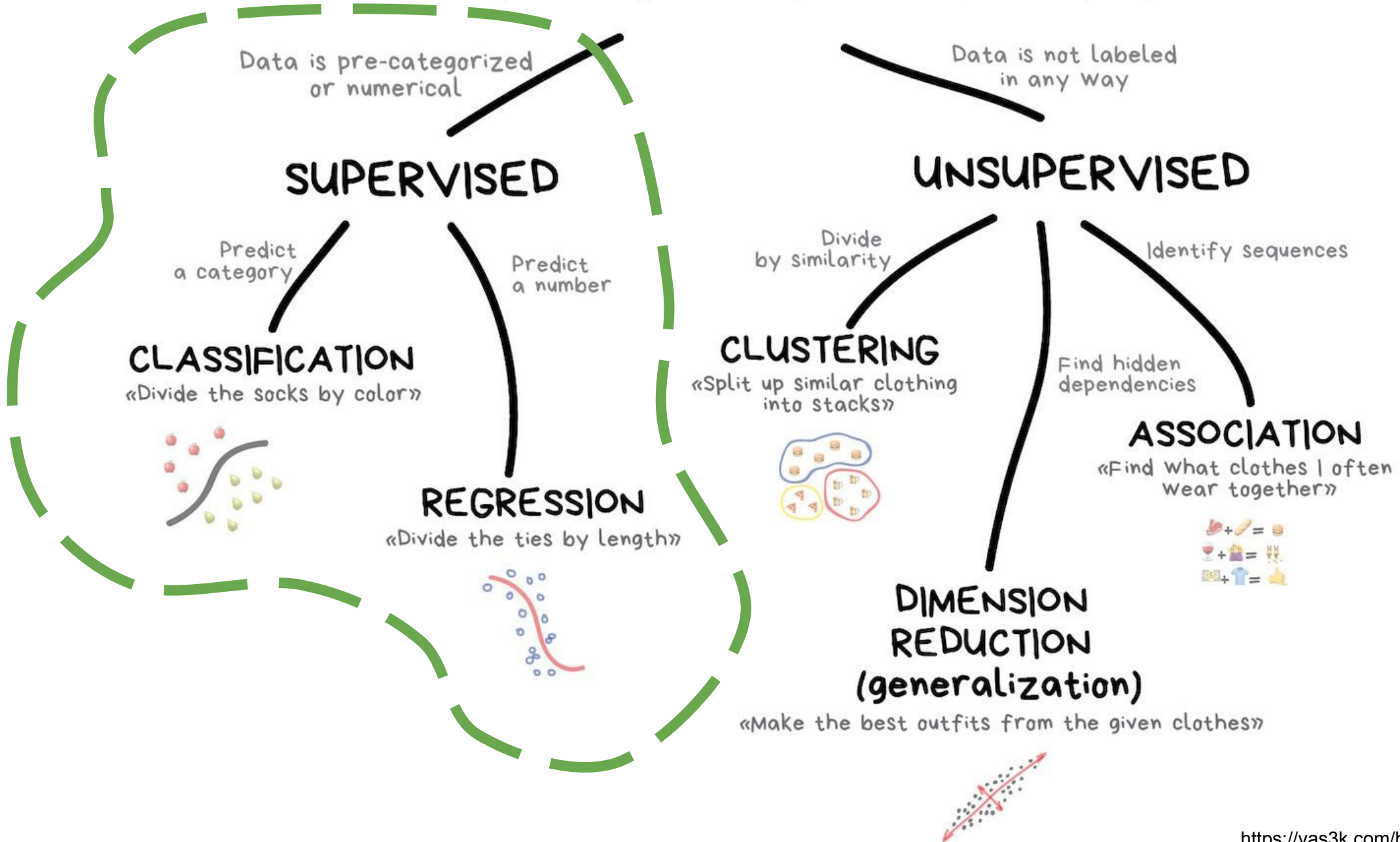
THE MAIN TYPES OF MACHINE LEARNING

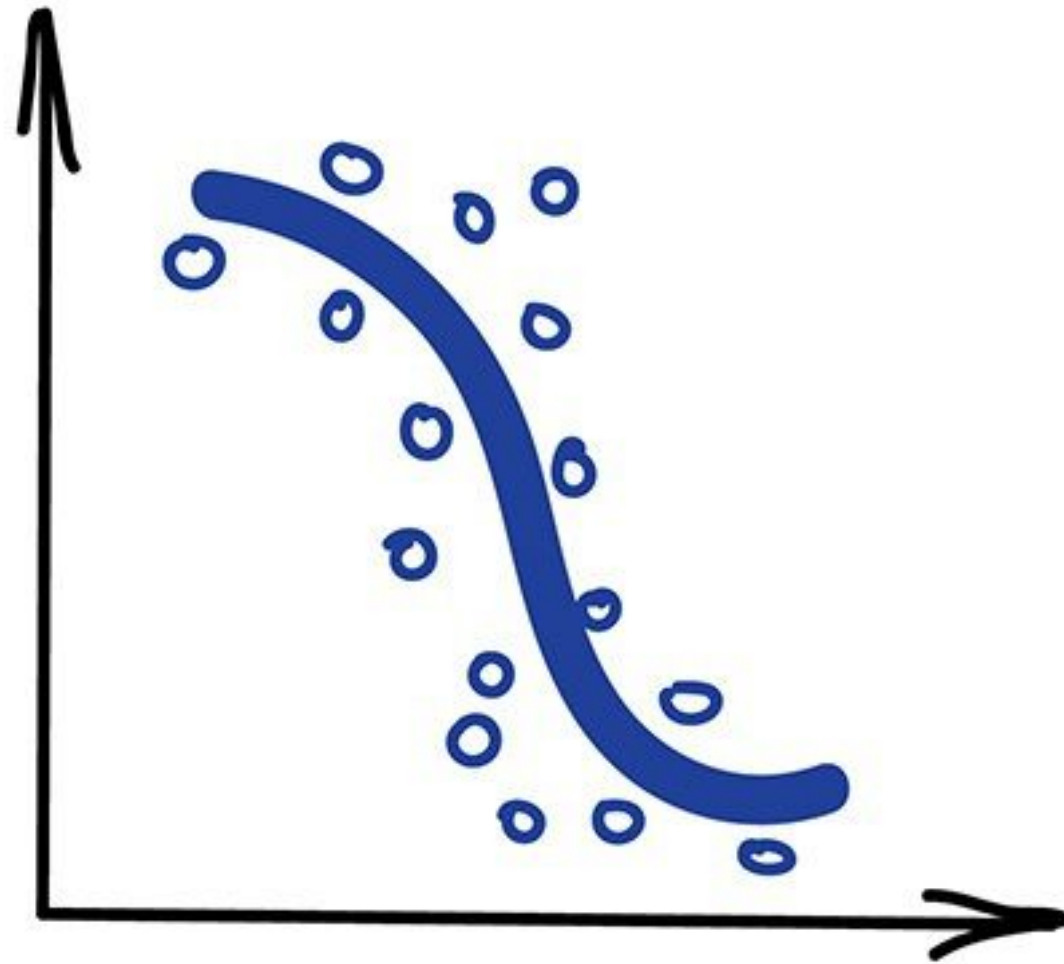


CLASSICAL MACHINE LEARNING



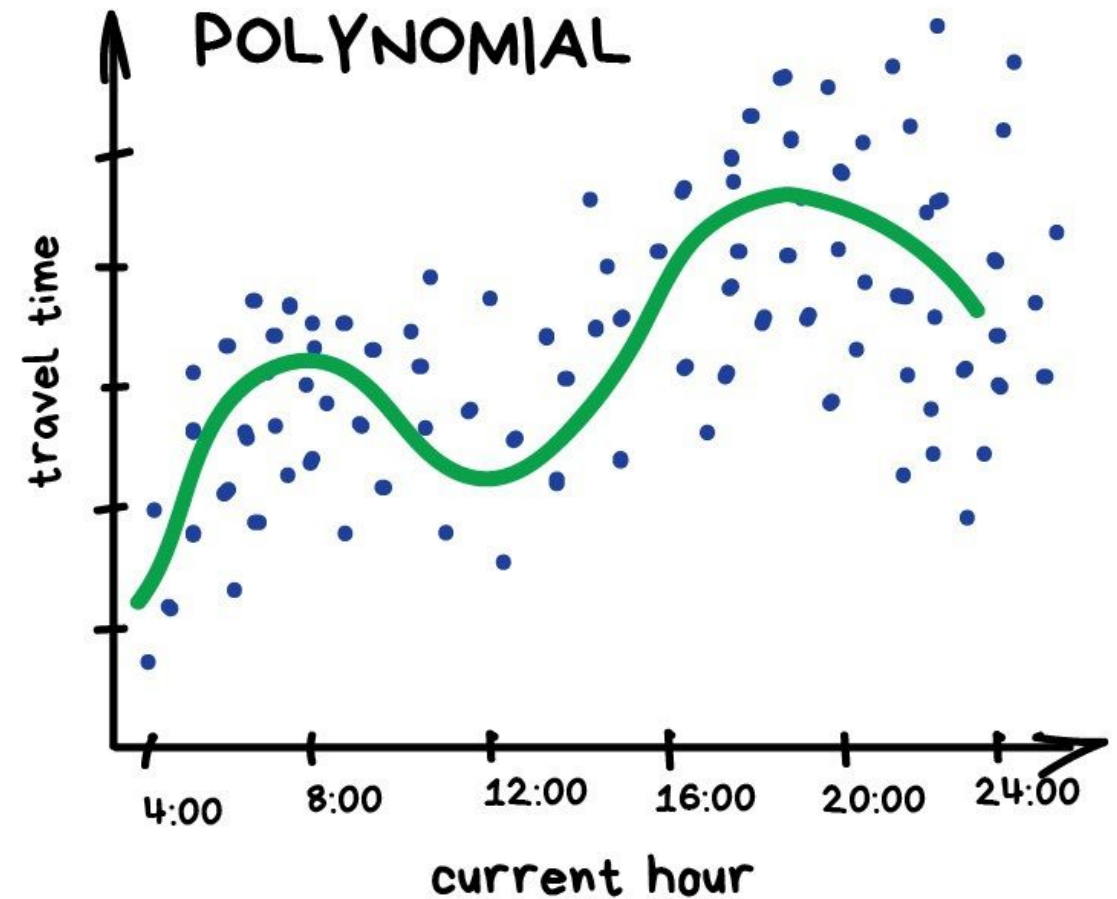
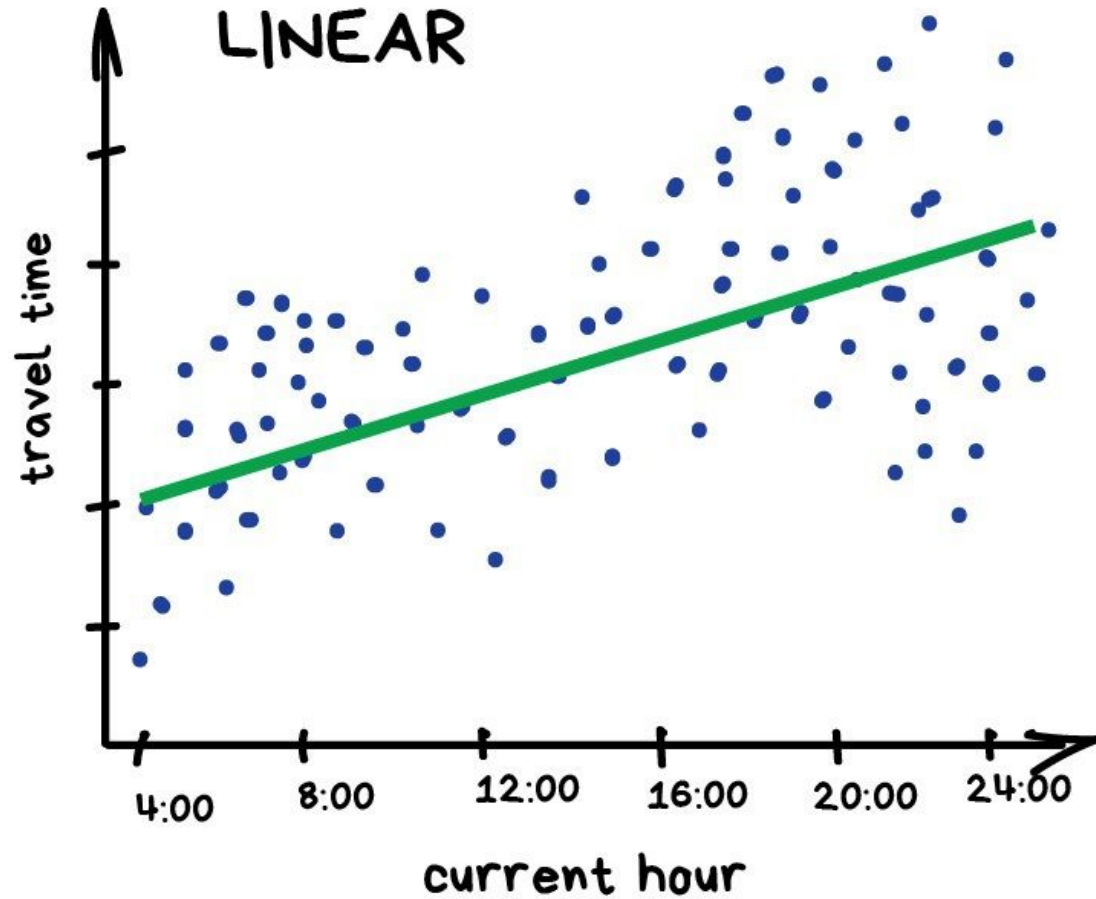
CLASSICAL MACHINE LEARNING





Regression

PREDICT TRAFFIC JAMS



REGRESSION

https://vas3k.com/blog/machine_learning/

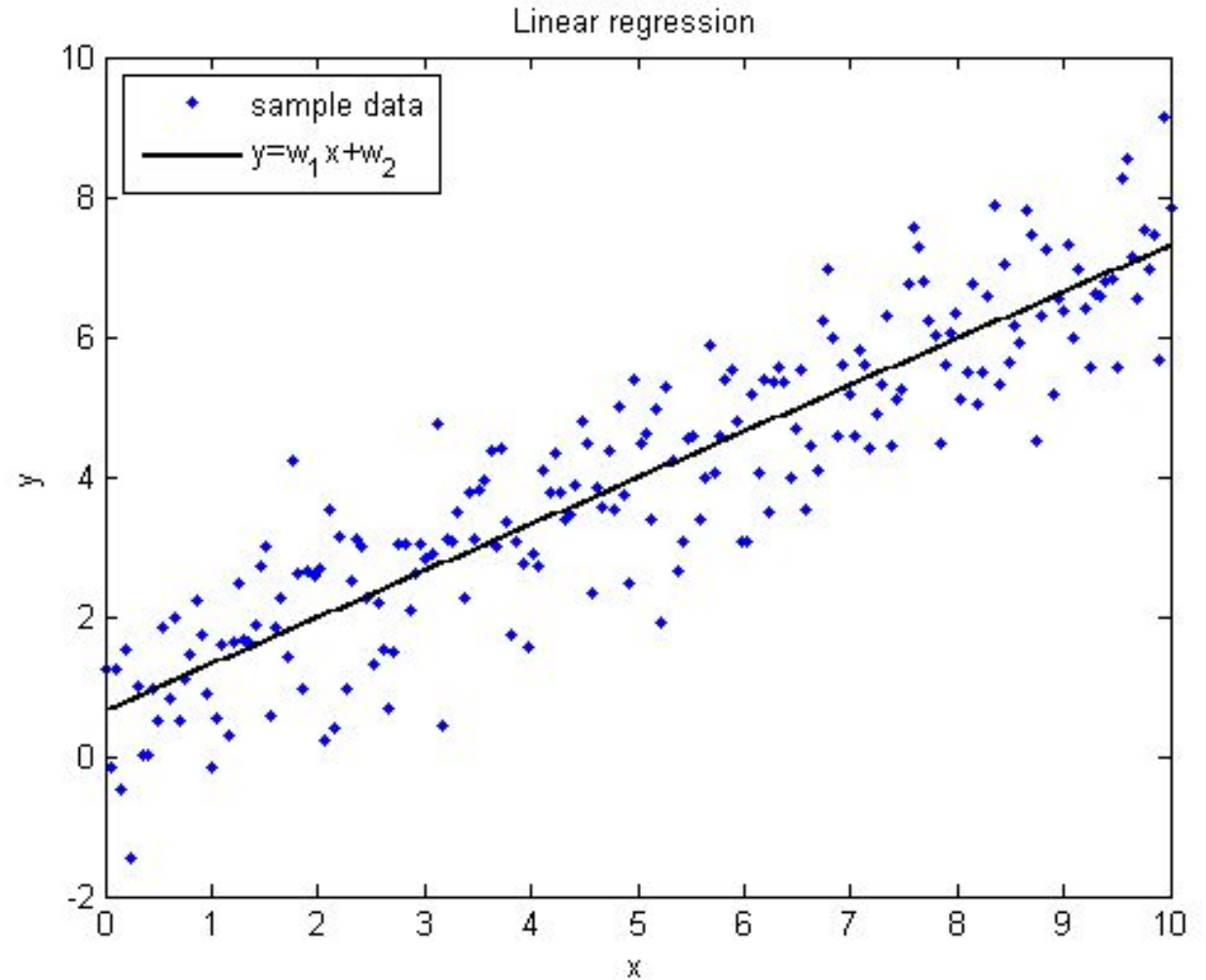
Simple linear regression

$x \in \mathbb{R}$ – input

$y \in \mathbb{R}$ – target

$w_0, w_1 \in \mathbb{R}$ – weights

$$\hat{y} = w_0 + w_1 x$$



Simple linear regression

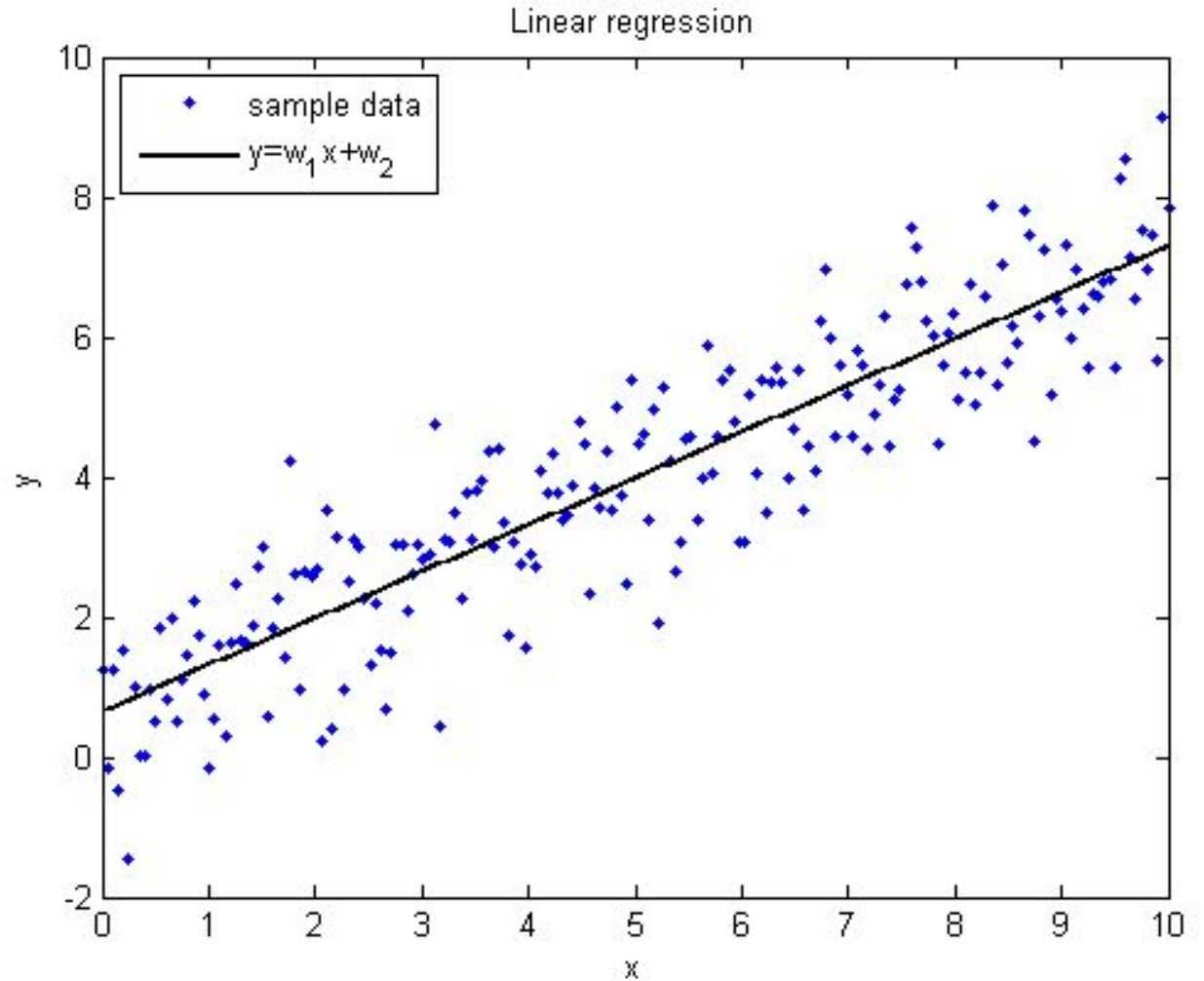
given data

$x \in \mathbb{R}$ – input

$y \in \mathbb{R}$ – target

$w_0, w_1 \in \mathbb{R}$ – weights

$$\hat{y} = w_0 + w_1 x$$



Simple linear regression

given data

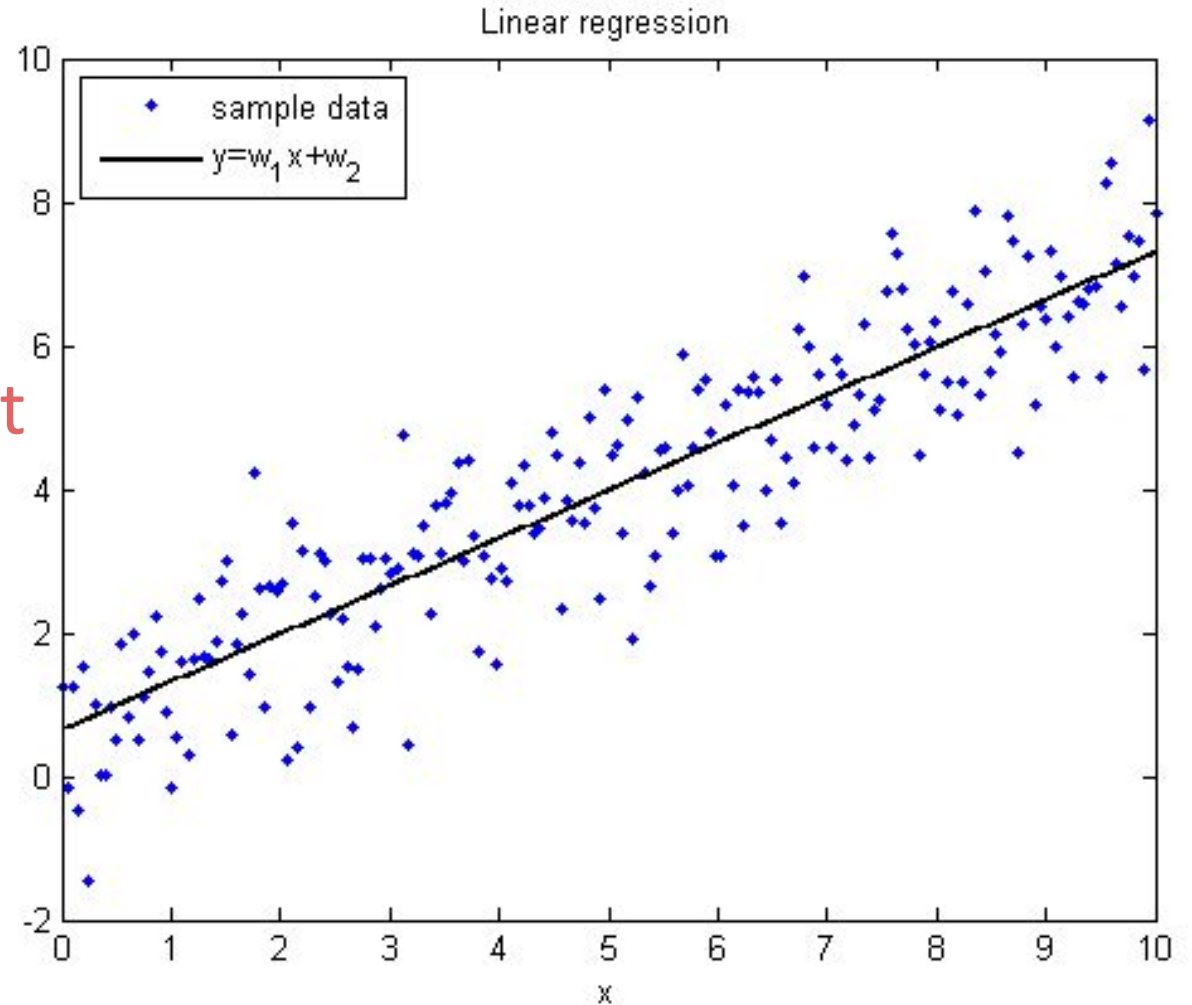
$x \in \mathbb{R}$ – input

$y \in \mathbb{R}$ – target

$w_0, w_1 \in \mathbb{R}$ – weights

trainable
coefficient

$$\hat{y} = w_0 + w_1 x$$



Multiple linear regression

housing.csv (1.36 MB) 10 of 10 columns

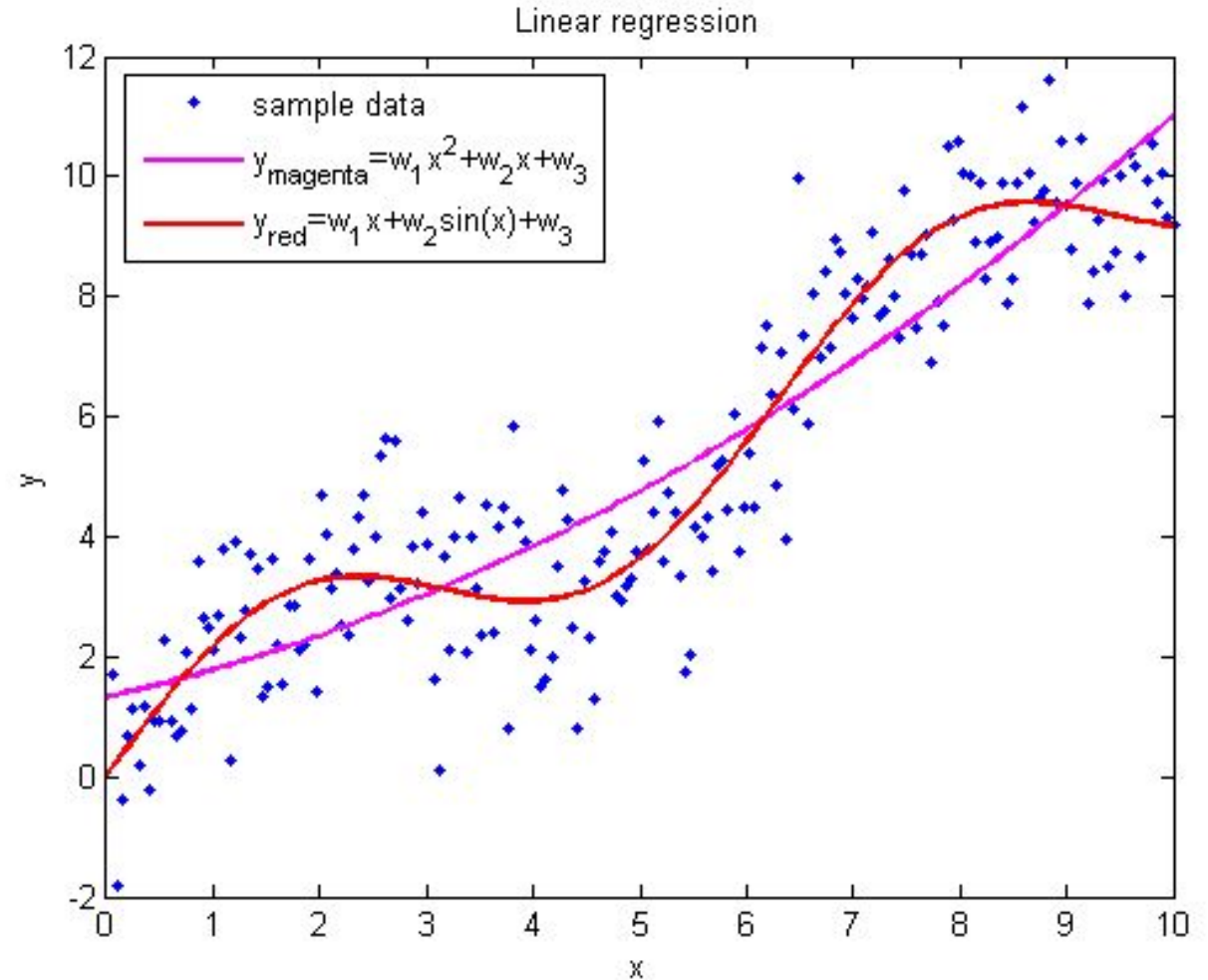
	# longitu...	# latitude	# housin...	# total_ro...	# total_b...	# populat...	# househ...	# median...	# median...	A ocean_...
1	-122.23	37.88	41.0	880.0	129.0	322.0	126.0	8.3252	452600.0	NEAR BAY
2	-122.22	37.86	21.0	7099.0	1106.0	2401.0	1138.0	8.3014	358500.0	NEAR BAY
3	-122.24	37.85	52.0	1467.0	190.0	496.0	177.0	7.2574	352100.0	NEAR BAY
4	-122.25	37.85	52.0	1274.0	235.0	558.0	219.0	5.6431	341300.0	NEAR BAY
5	-122.25	37.85	52.0	1627.0	280.0	565.0	259.0	3.8462	342200.0	NEAR BAY
6	-122.25	37.85	52.0	919.0	213.0	413.0	193.0	4.0368	269700.0	NEAR BAY
7	-122.25	37.84	52.0	2535.0	489.0	1094.0	514.0	3.6591	299200.0	NEAR BAY
8	-122.25	37.84	52.0	3104.0	687.0	1157.0	647.0	3.12	241400.0	NEAR BAY
9	-122.26	37.84	42.0	2555.0	665.0	1206.0	595.0	2.0804	226700.0	NEAR BAY
10										AY

$$\hat{y} = w_0 + w_1 x_1 + \dots + w_N x_N = \sum_{j=0}^M w_j x_j$$

Also linear regression

$$\hat{y} = \sum_{j=0}^M w_j g(x_j)$$

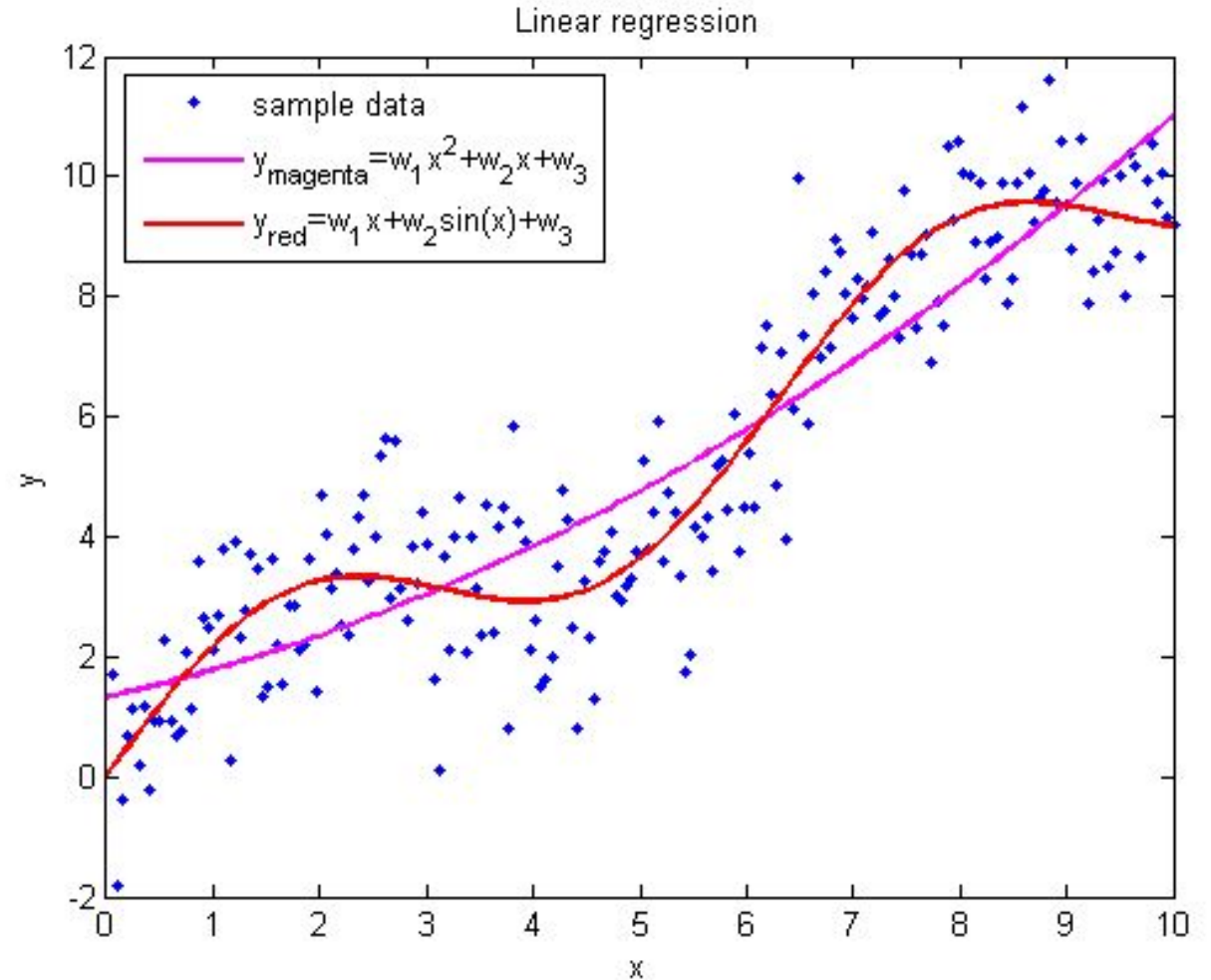
$$g(x) = x \Rightarrow \hat{y} = \sum_{j=0}^M w_j x_j$$



Also linear regression

$$\hat{y} = \sum_{j=0}^M w_j g(x_j)$$

$$g(x) = x \Rightarrow \hat{y} = \sum_{j=0}^M w_j x_j$$

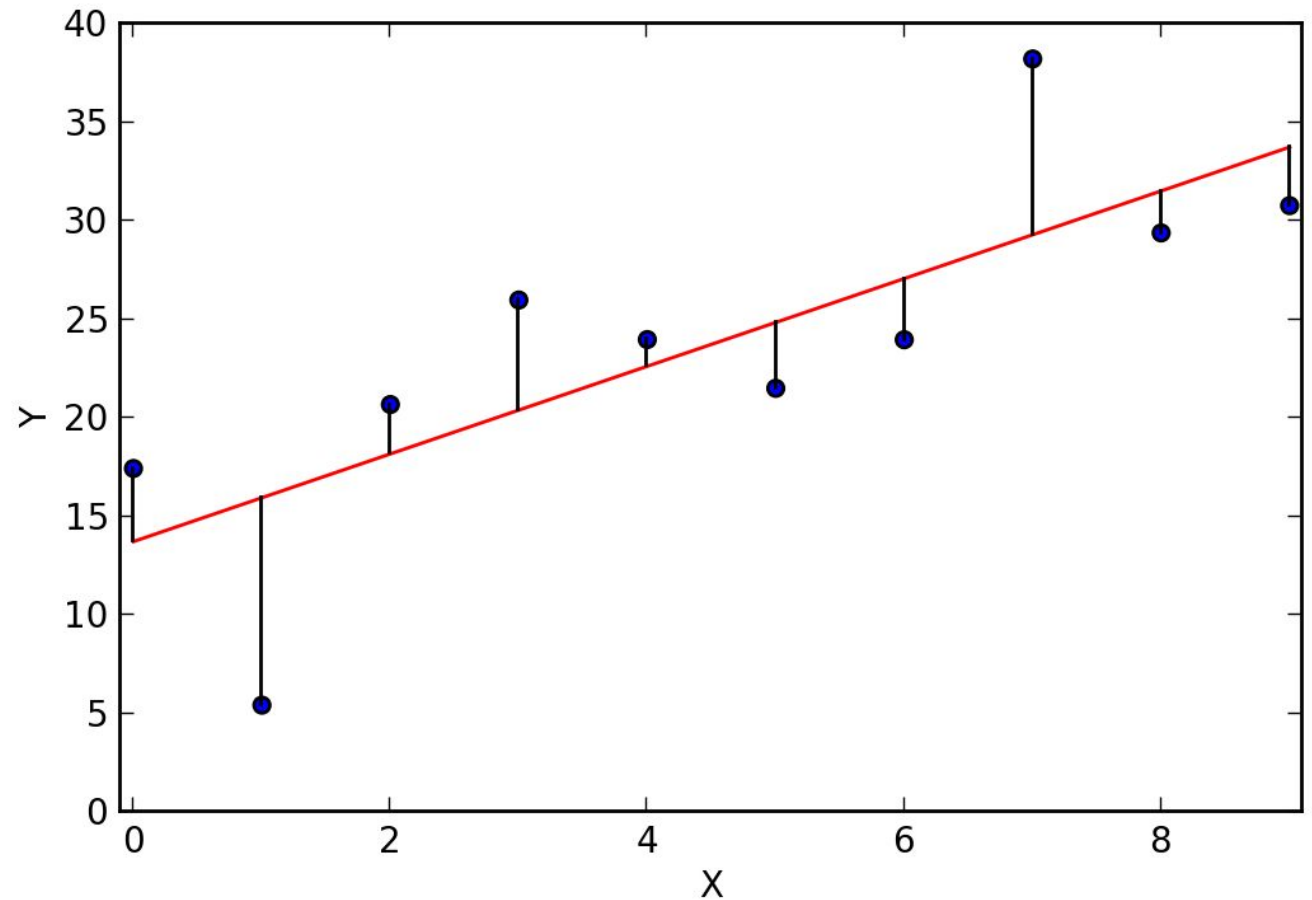


Evaluating prediction

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

y — target

\hat{y} — prediction

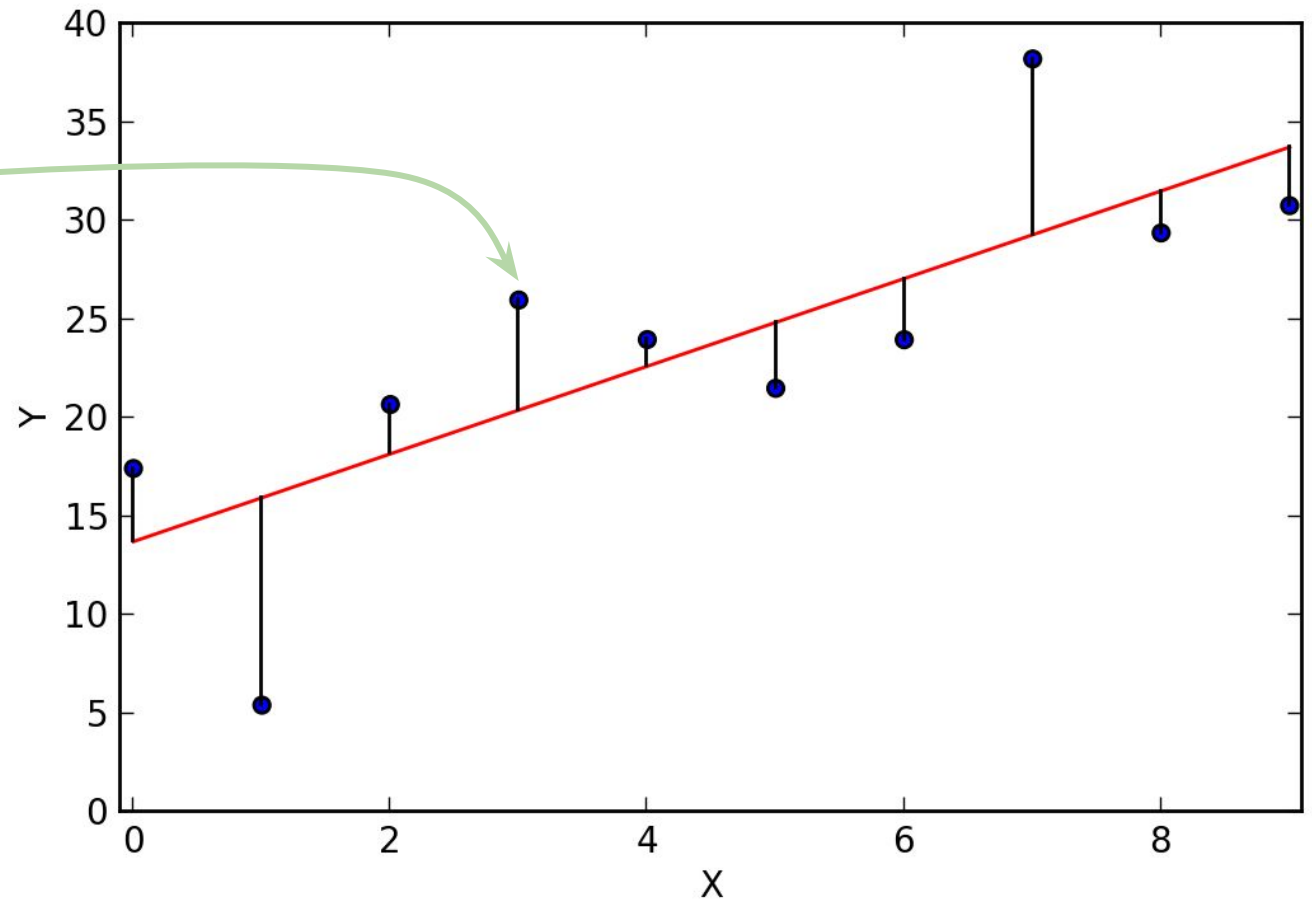


Evaluating prediction

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

y — target

\hat{y} — prediction

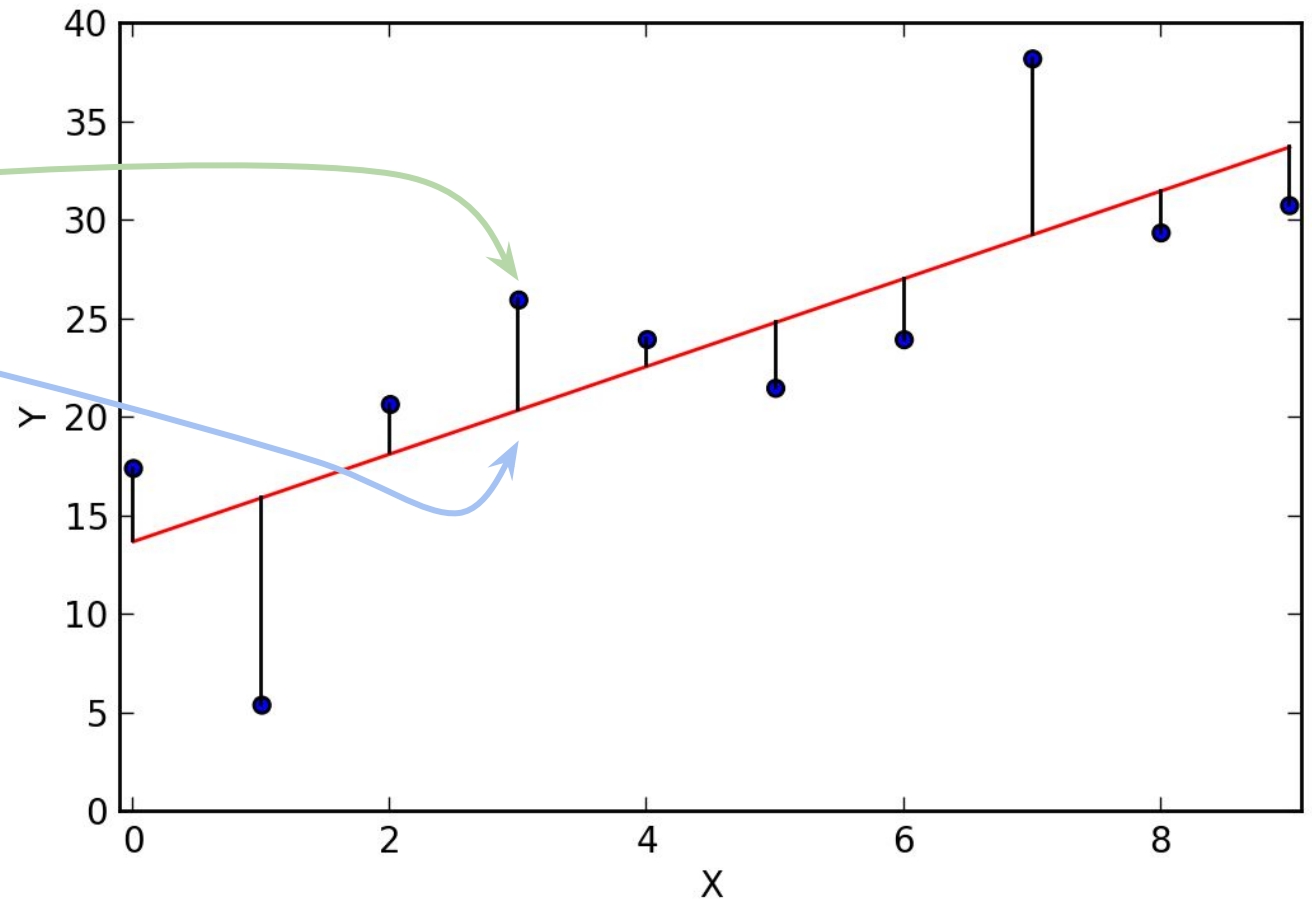


Evaluating prediction

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

y — target

\hat{y} — prediction



Training parameters

$$\hat{y} = w_0 + w_1 x$$

$$\begin{aligned} L(y, \hat{y}) &= (y - \hat{y})^2 \\ &= (y - w_1 x - w_0)^2 \end{aligned}$$

Training parameters

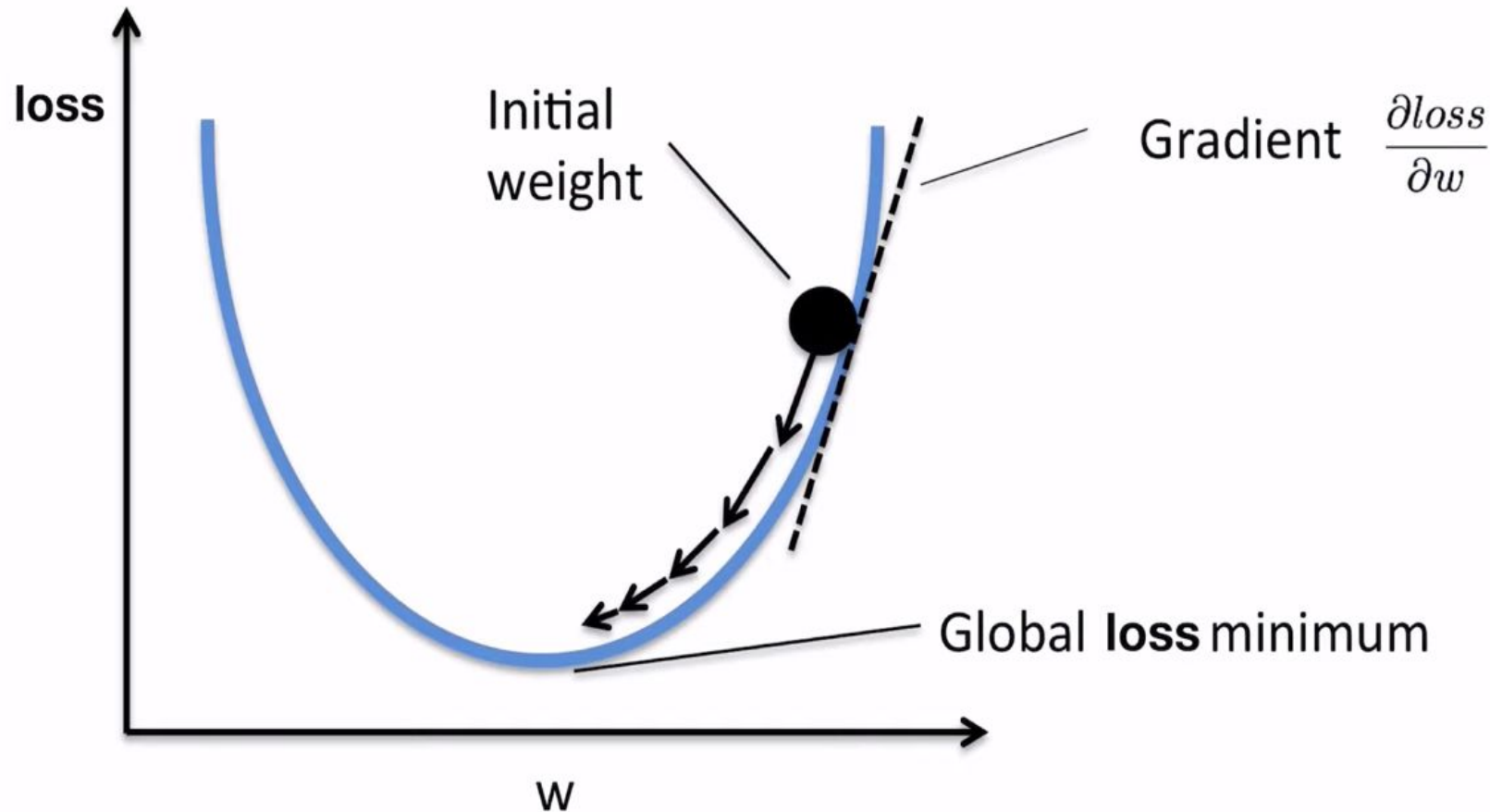
$$\hat{y} = w_0 + w_1 x$$

$$\begin{aligned} L(y, \hat{y}) &= (y - \hat{y})^2 \\ &= (y - w_1 x - w_0)^2 \end{aligned}$$

$$w_{new} = w_{old} - \alpha \frac{\partial L(w, x)}{\partial w}$$

$\alpha \in \mathbb{R}$ – learning rate

Training parameters



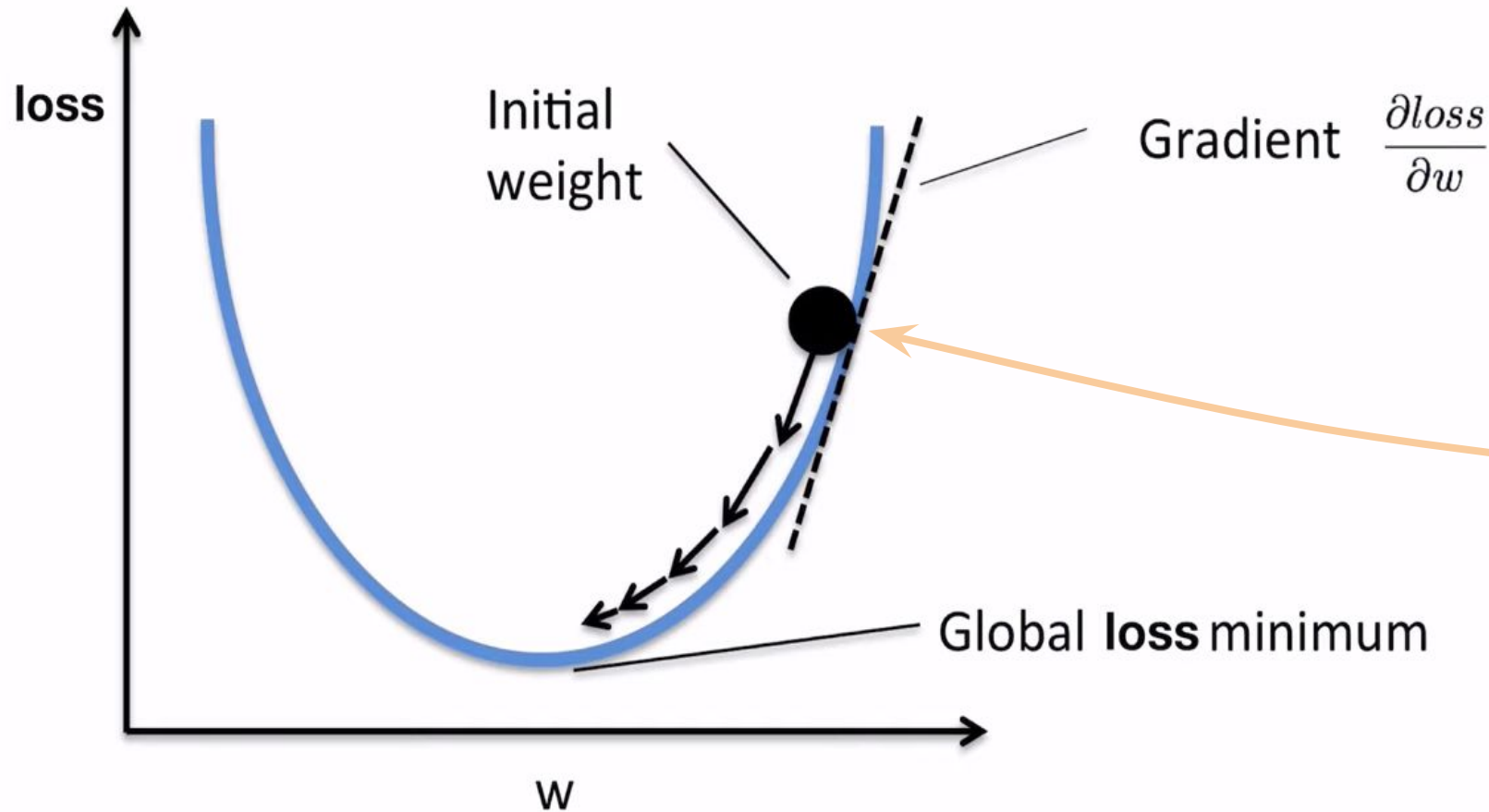
$$\hat{y} = w_0 + w_1 x$$

$$L(y, \hat{y}) = (y - \hat{y})^2 \\ = (y - w_1 x - w_0)^2$$

$$w_{new} = w_{old} - \alpha \frac{\partial L(w, x)}{\partial w}$$

$\alpha \in \mathbb{R}$ – learning rate

Training parameters



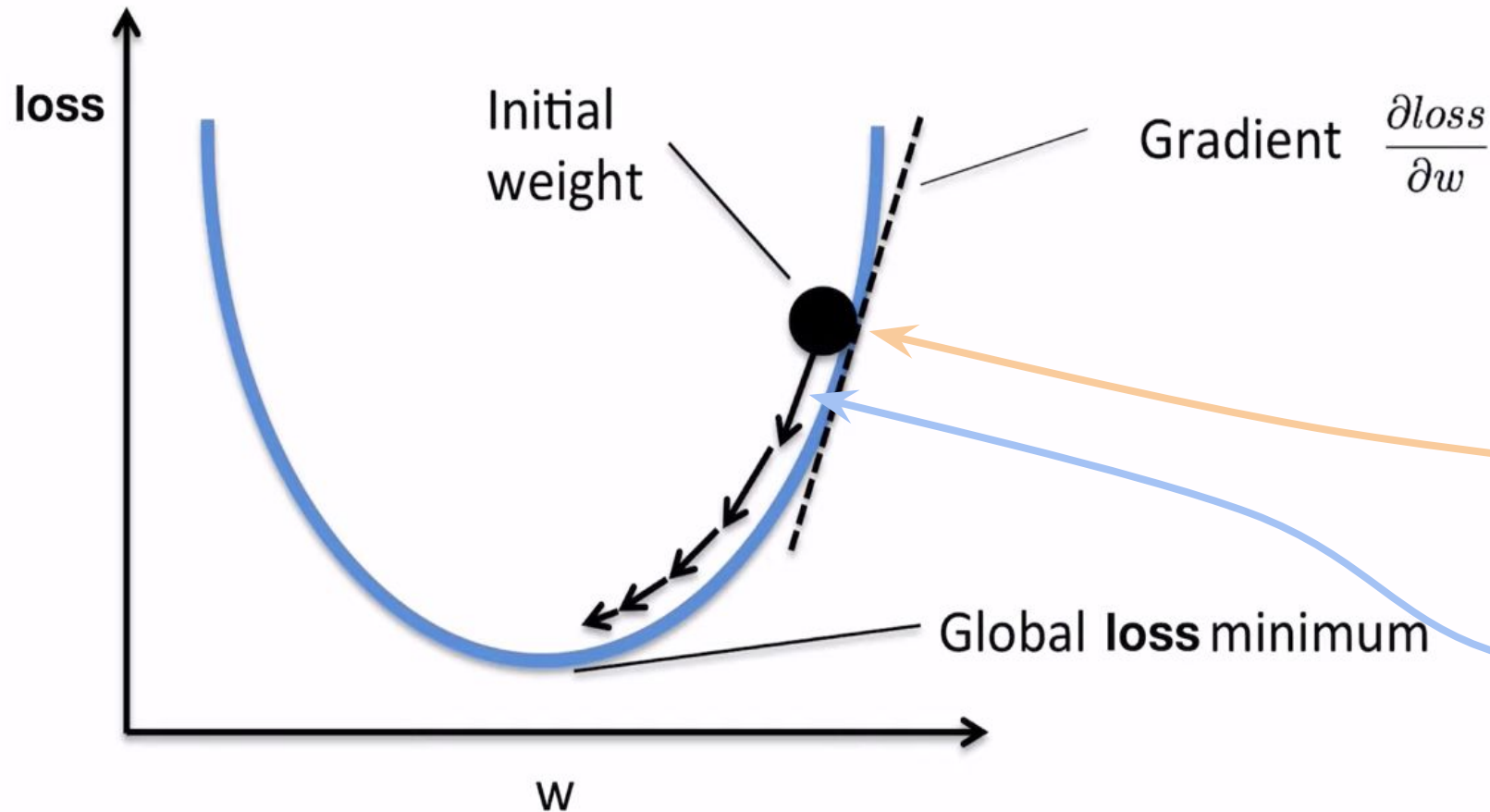
$$\hat{y} = w_0 + w_1 x$$

$$L(y, \hat{y}) = (y - \hat{y})^2 \\ = (y - w_1 x - w_0)^2$$

$$w_{new} = w_{old} - \alpha \frac{\partial L(w, x)}{\partial w}$$

$\alpha \in \mathbb{R}$ – learning rate

Training parameters



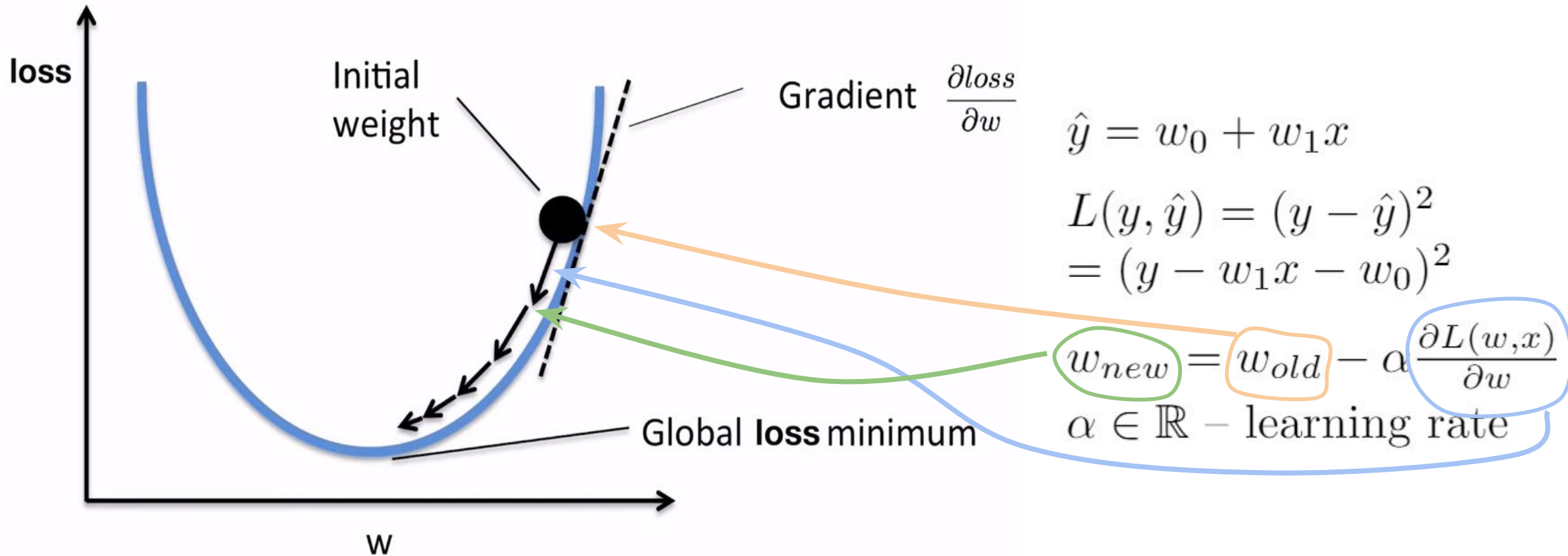
$$\hat{y} = w_0 + w_1 x$$

$$L(y, \hat{y}) = (y - \hat{y})^2 \\ = (y - w_1 x - w_0)^2$$

$$w_{new} = w_{old} - \alpha \frac{\partial L(w, x)}{\partial w}$$

$\alpha \in \mathbb{R}$ – learning rate

Training parameters



Linear regression overview

special case: $\hat{y} = w_1 x + w_0$

Linear regression overview

special case: $\hat{y} = w_1 x + w_0$

general case: $\hat{y} = w_0 + w_1 x_1 + \dots + w_N x_N = \sum_{j=0}^M w_j x_j$

Linear regression overview

special case: $\hat{y} = w_1x + w_0$

general case: $\hat{y} = w_0 + w_1x_1 + \dots + w_Nx_N = \sum_{j=0}^M w_jx_j$

even more general case: $\hat{y} = \sum_{j=0}^M w_jg(x_j)$

Linear regression overview

special case: $\hat{y} = w_1 x + w_0$

general case: $\hat{y} = w_0 + w_1 x_1 + \dots + w_N x_N = \sum_{j=0}^M w_j x_j$

even more general case: $\hat{y} = \sum_{j=0}^M w_j g(x_j)$

loss function: $\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$

Linear regression overview

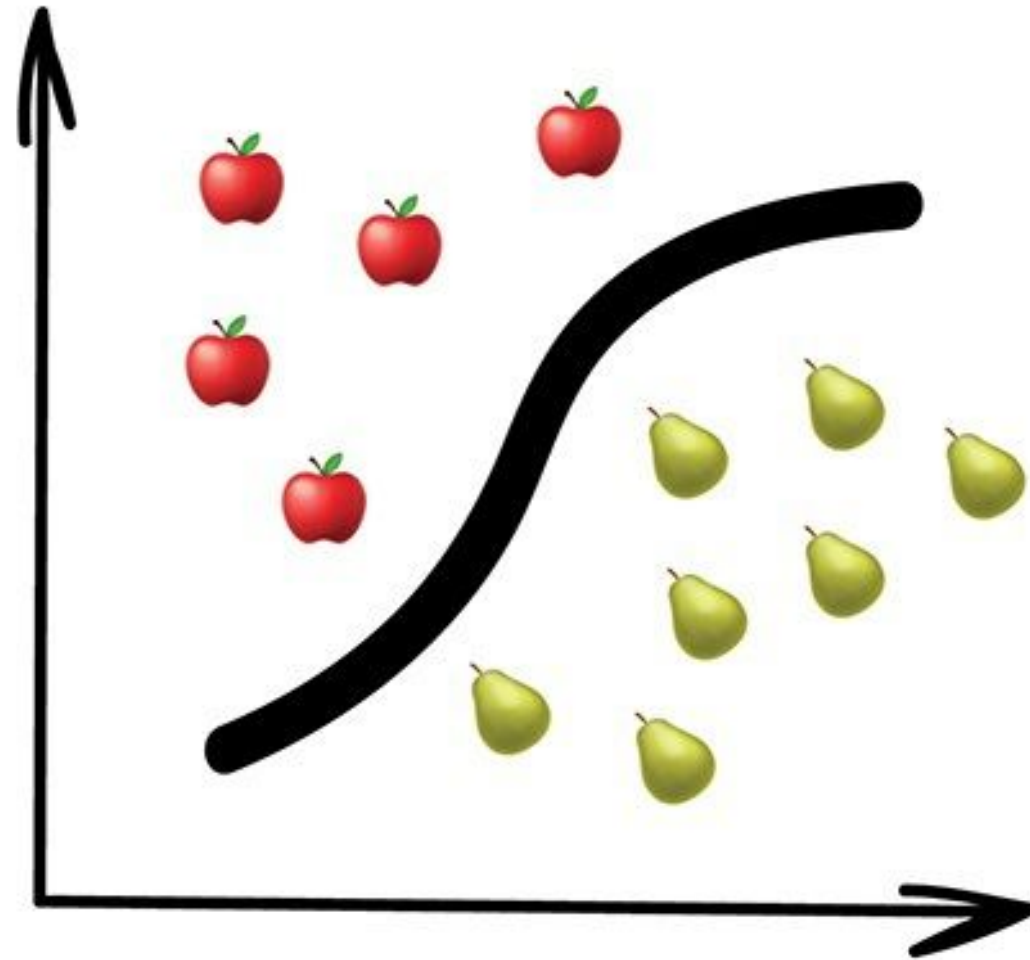
special case: $\hat{y} = w_1x + w_0$

general case: $\hat{y} = w_0 + w_1x_1 + \dots + w_Nx_N = \sum_{j=0}^M w_jx_j$

even more general case: $\hat{y} = \sum_{j=0}^M w_jg(x_j)$

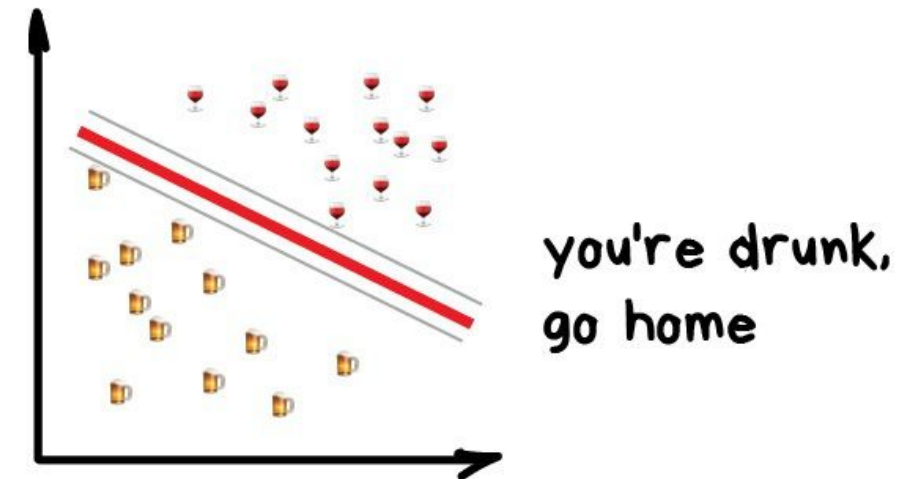
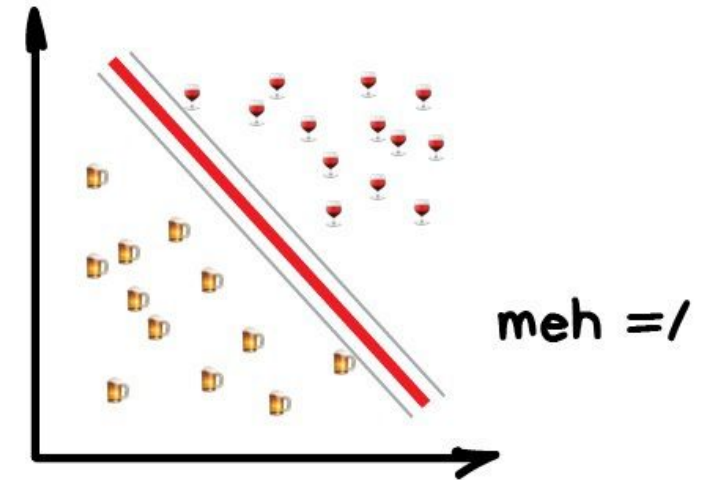
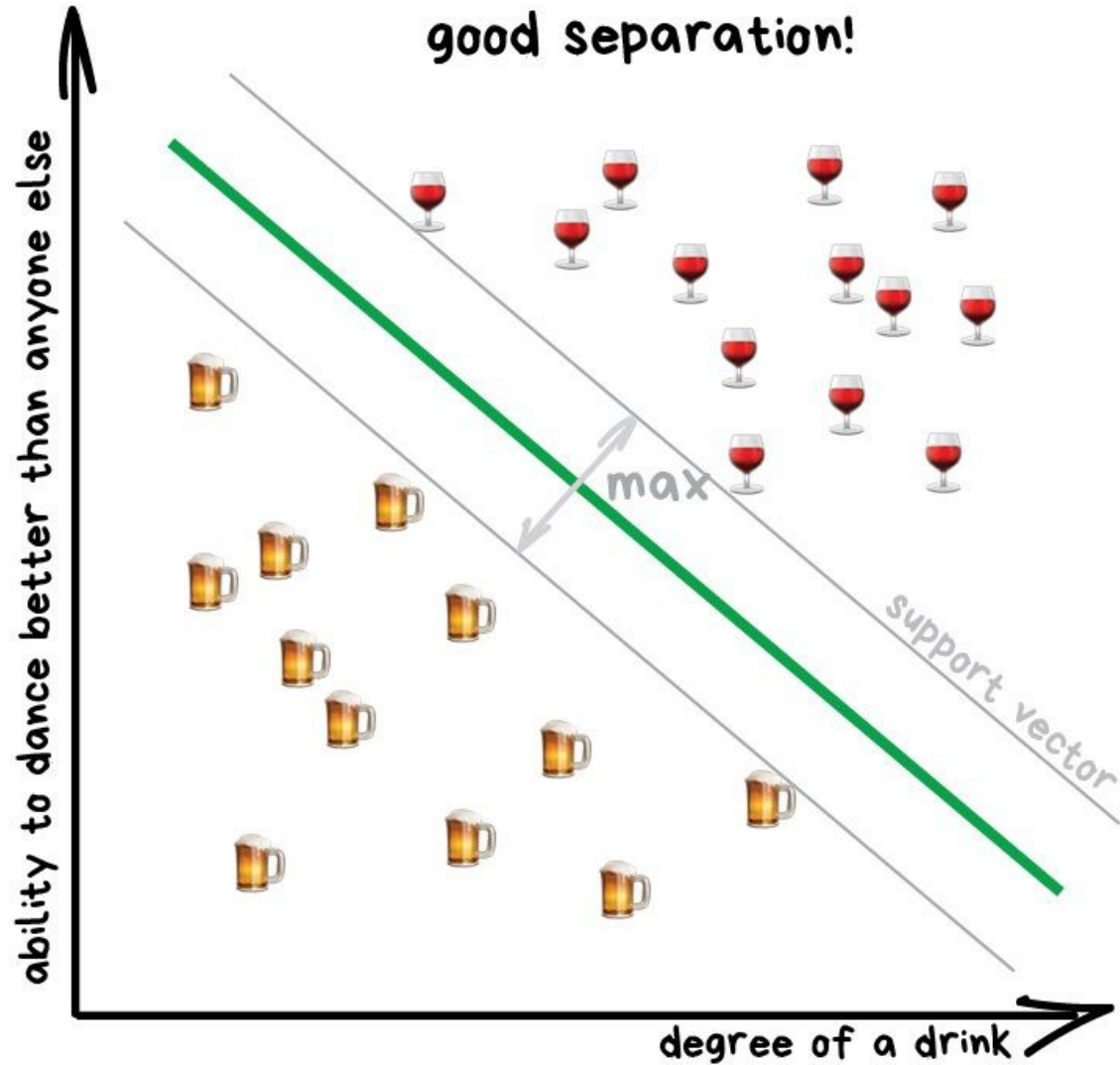
loss function: $\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$

weights update: $w_{new} = w_{old} - \frac{\partial L}{\partial w}$



Classification

SEPARATE TYPES OF ALCOHOL



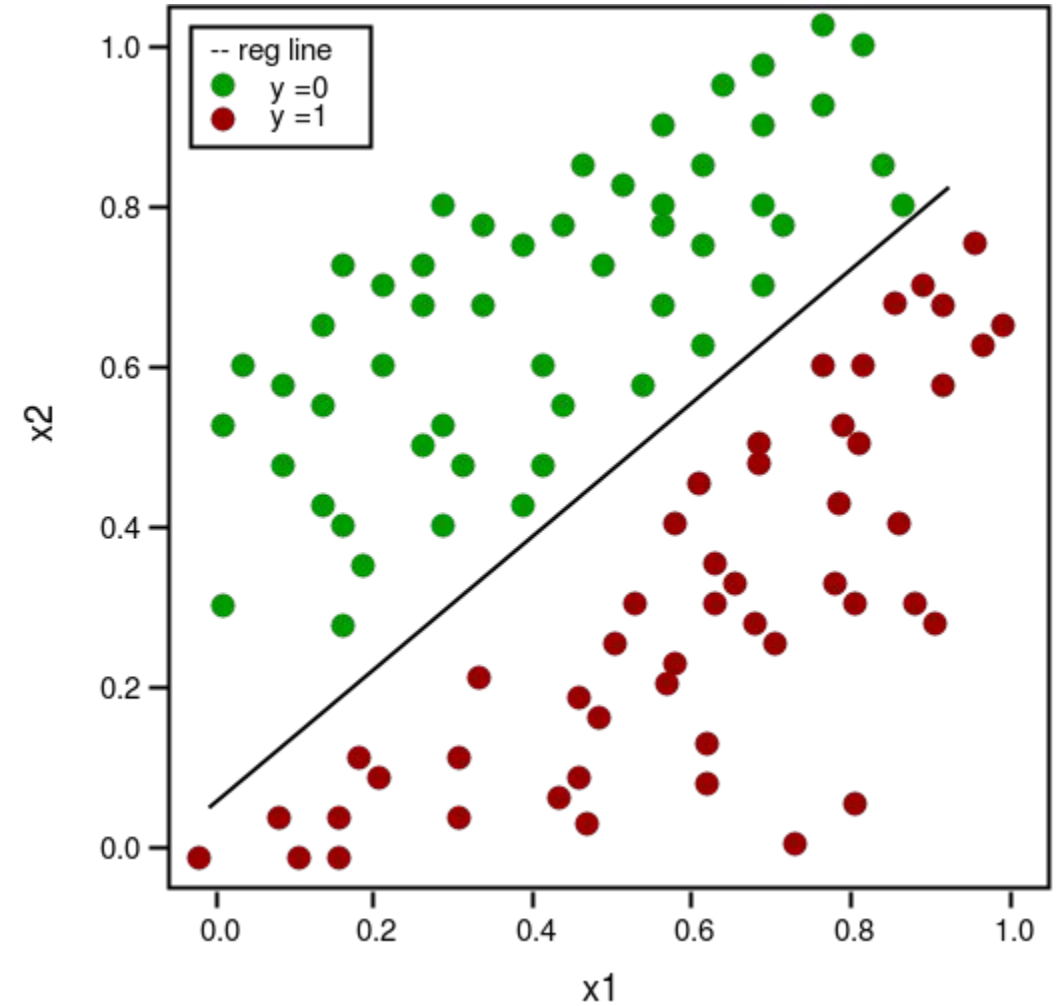
Logistic regression

$x_1, x_2 \in \mathbb{R}$ – inputs

$y \in \{0, 1\}$ – target

Logistic regression

$x_1, x_2 \in \mathbb{R}$ – inputs
 $y \in \{0, 1\}$ – target



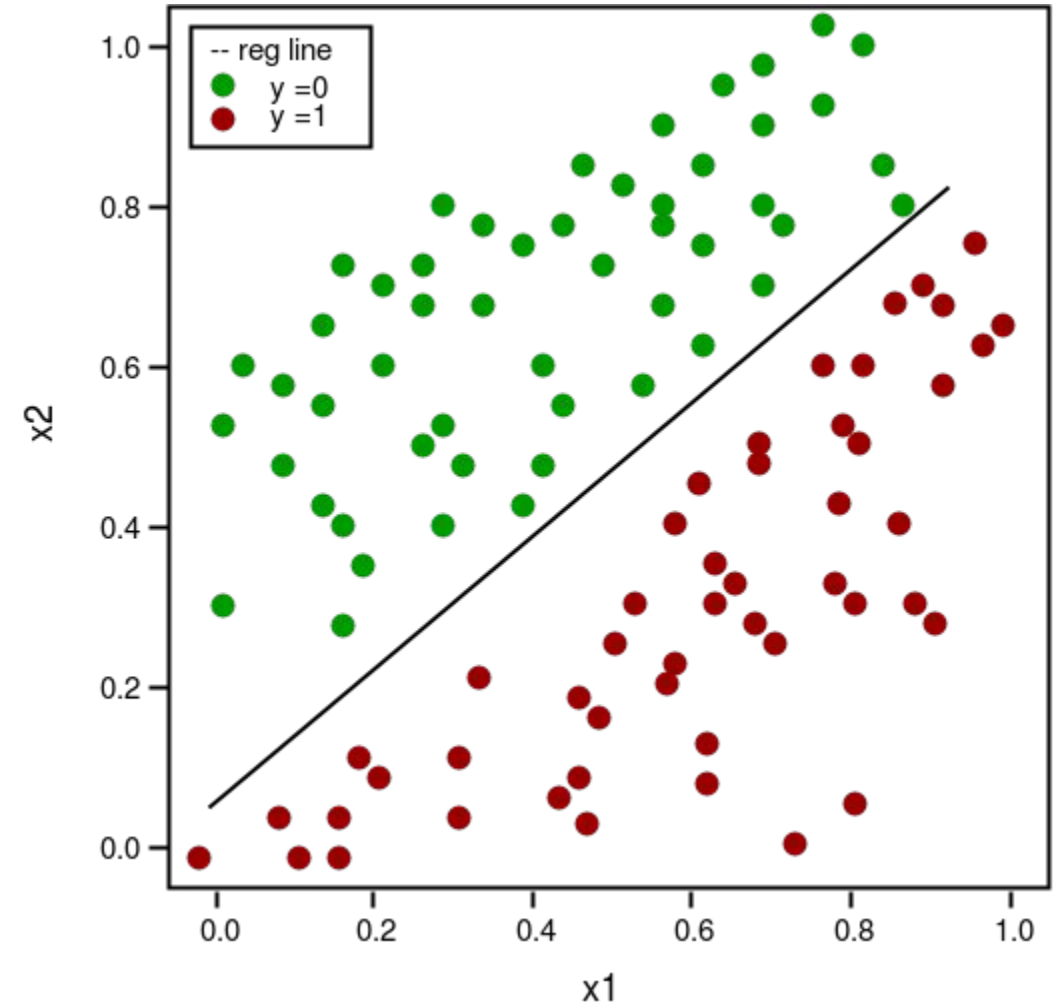
Logistic regression

$x_1, x_2 \in \mathbb{R}$ – inputs

$y \in \{0, 1\}$ – target

$w_0, w_1, w_2 \in \mathbb{R}$ – weights

$$\hat{y} = w_0 + w_1x_1 + w_2x_2$$



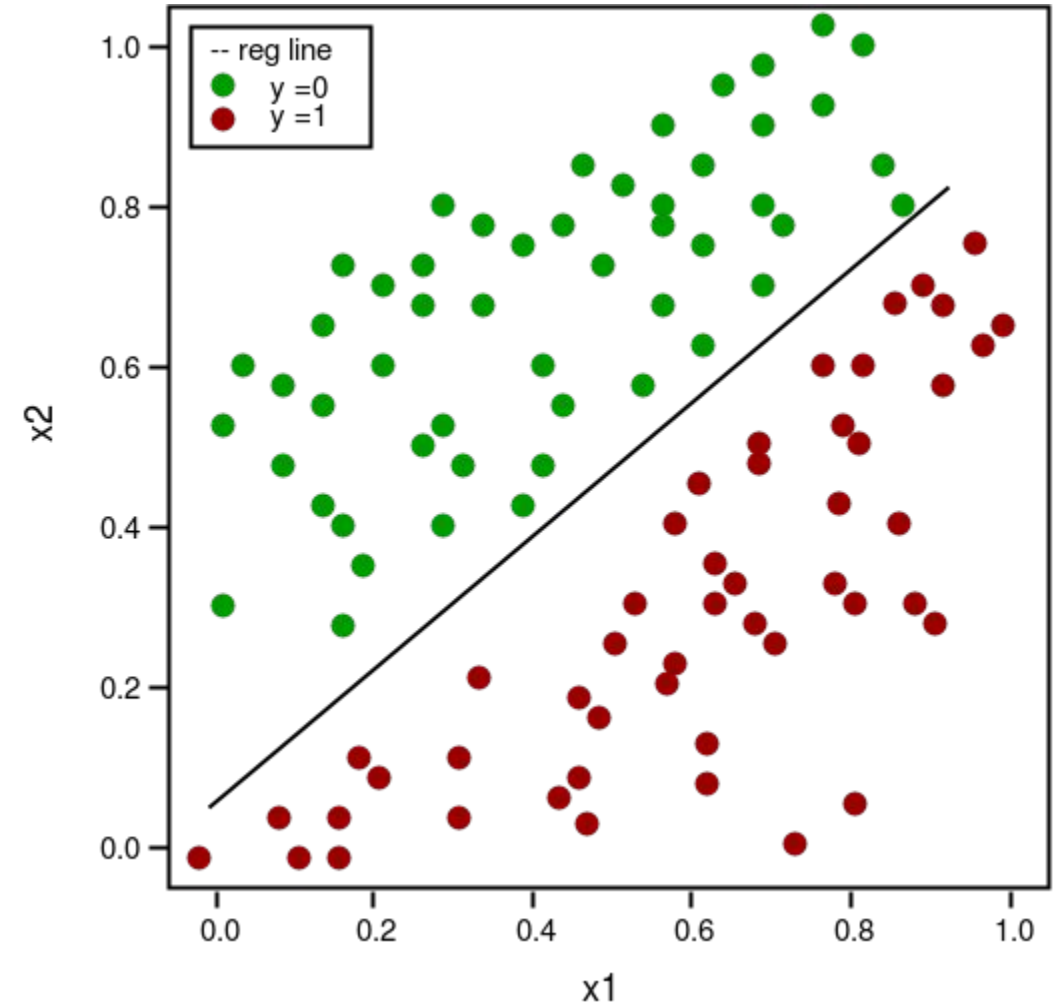
Logistic regression

$x_1, x_2 \in \mathbb{R}$ – inputs

$y \in \{0, 1\}$ – target

$w_0, w_1, w_2 \in \mathbb{R}$ – weights

~~$$\hat{y} = w_0 + w_1x_1 + w_2x_2$$~~



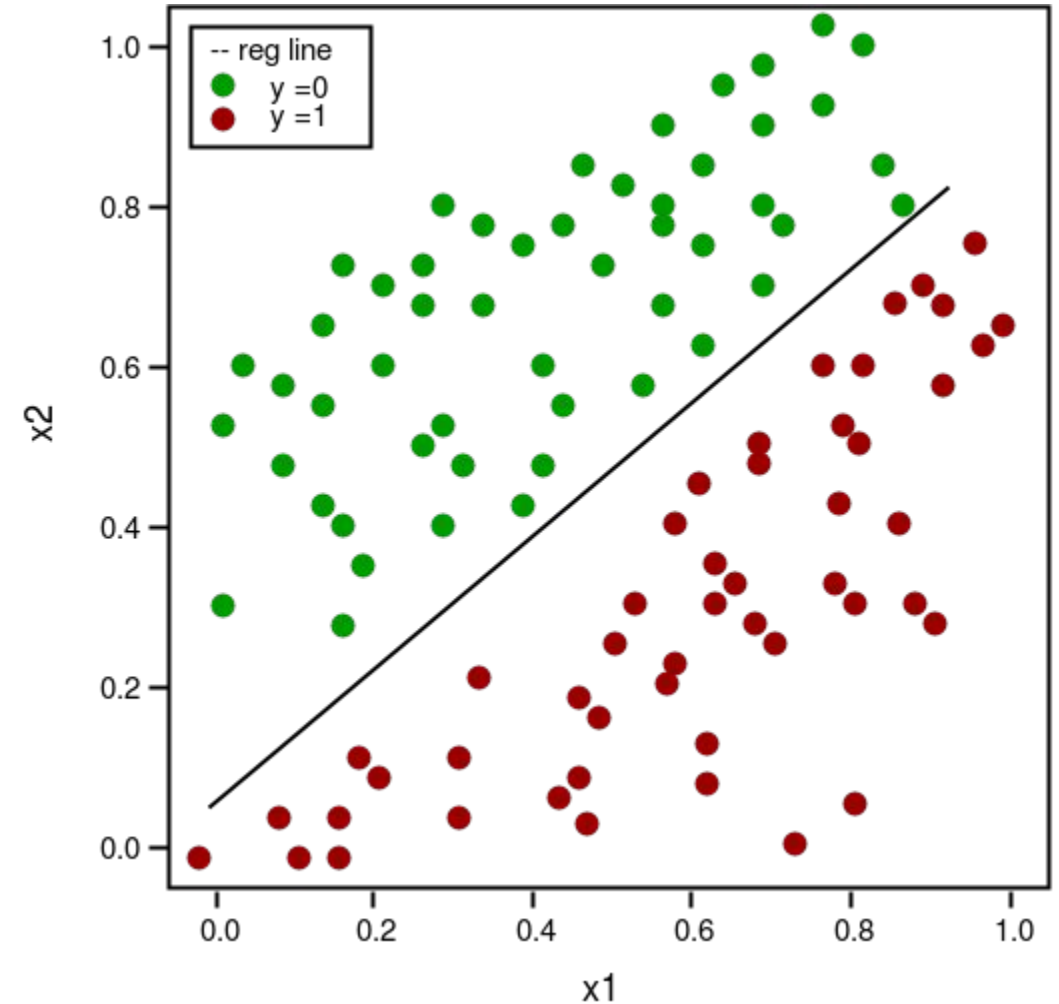
Logistic regression

$x_1, x_2 \in \mathbb{R}$ – inputs

$y \in \{0, 1\}$ – target

$w_0, w_1, w_2 \in \mathbb{R}$ – weights

$$\hat{y} = f(w_0 + w_1x_1 + w_2x_2)$$



Logistic regression

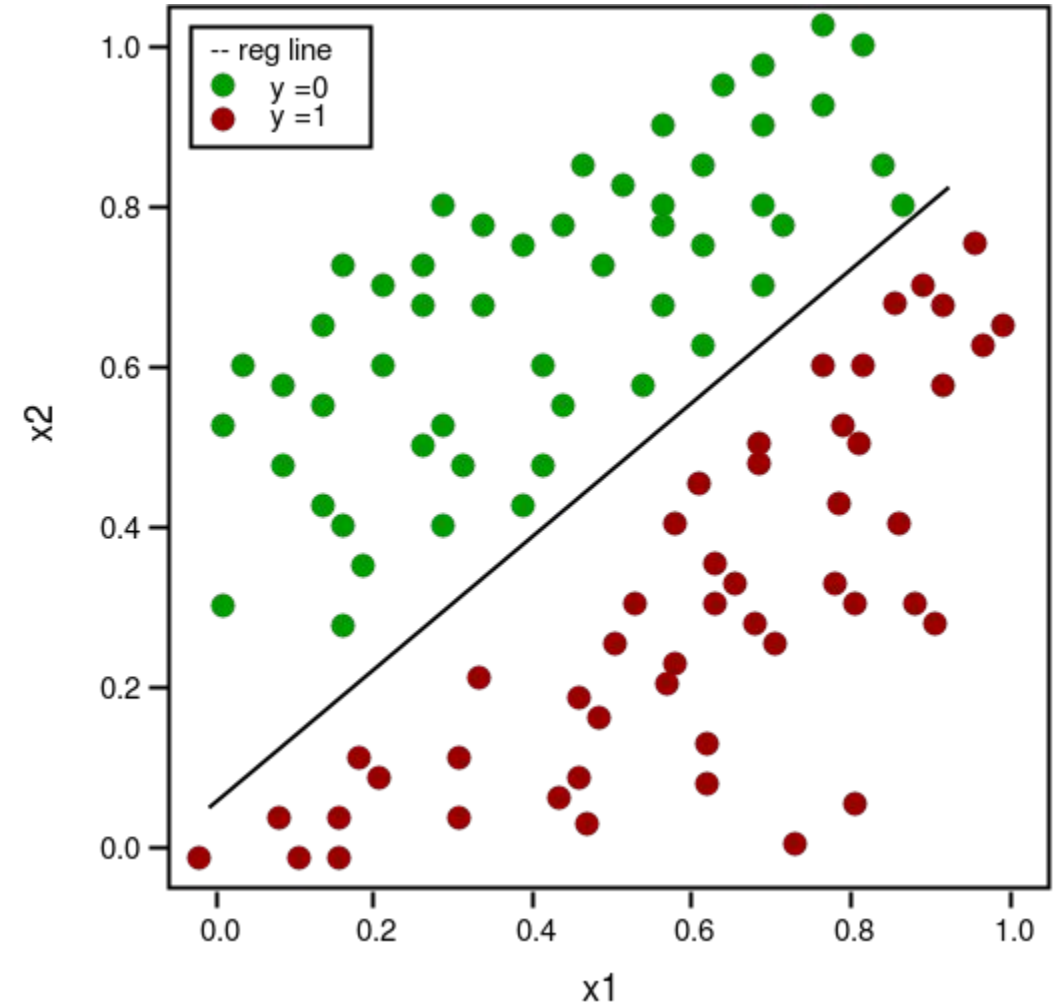
$x_1, x_2 \in \mathbb{R}$ – inputs

$y \in \{0, 1\}$ – target

$w_0, w_1, w_2 \in \mathbb{R}$ – weights

$$\hat{y} = f(w_0 + w_1x_1 + w_2x_2)$$

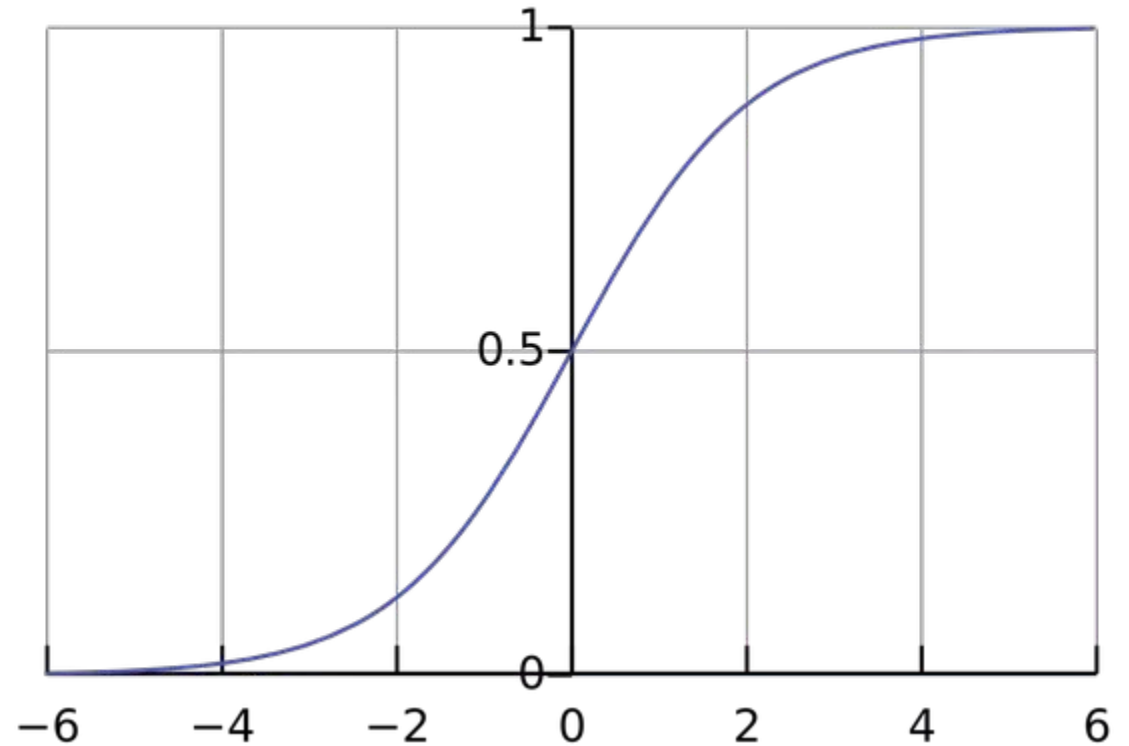
$$f(\cdot) : \mathbb{R} \mapsto (0, 1)$$



Softmax function

$$\sigma(\cdot) : \mathbb{R} \mapsto (0, 1)$$

$$\sigma(z) = \frac{1}{1 + \exp(-z)}$$



Loss function: cross-entropy

$y \in \{0, 1\}$ – target

$\hat{y} \in \mathbb{R}$ – prediction

Loss function: cross-entropy

$$\begin{array}{l} y \in \{0, 1\} - \text{target} \\ \hat{y} \in \mathbb{R} - \text{prediction} \end{array} \quad \text{CE}(y, \hat{y}) = - \sum_{i=1}^N ($$

Loss function: cross-entropy

$$\begin{array}{l} y \in \{0, 1\} - \text{target} \\ \hat{y} \in \mathbb{R} - \text{prediction} \end{array} \quad \text{CE}(y, \hat{y}) = - \sum_{i=1}^N (y_i \log \hat{y}_i +$$

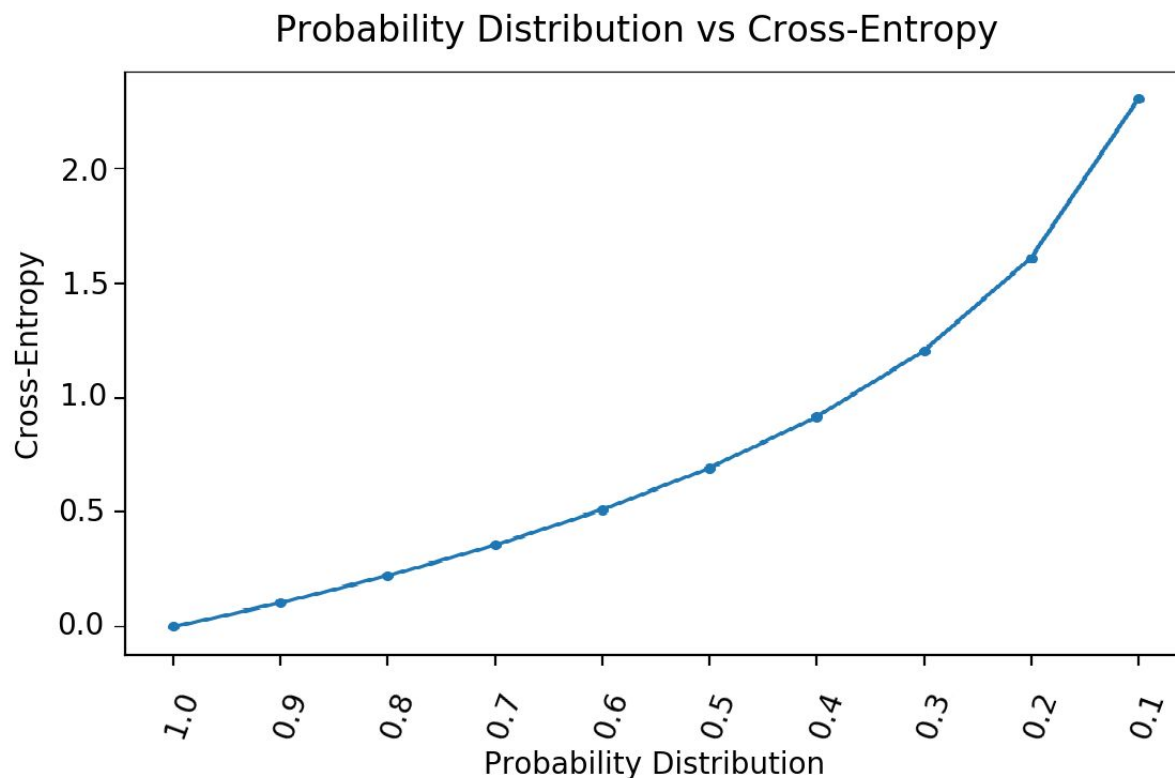
Loss function: cross-entropy

$$\begin{array}{l} y \in \{0, 1\} - \text{target} \\ \hat{y} \in \mathbb{R} - \text{prediction} \end{array} \quad \text{CE}(y, \hat{y}) = - \sum_{i=1}^N (y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i))$$

Loss function: cross-entropy

$y \in \{0, 1\}$ – target
 $\hat{y} \in \mathbb{R}$ – prediction

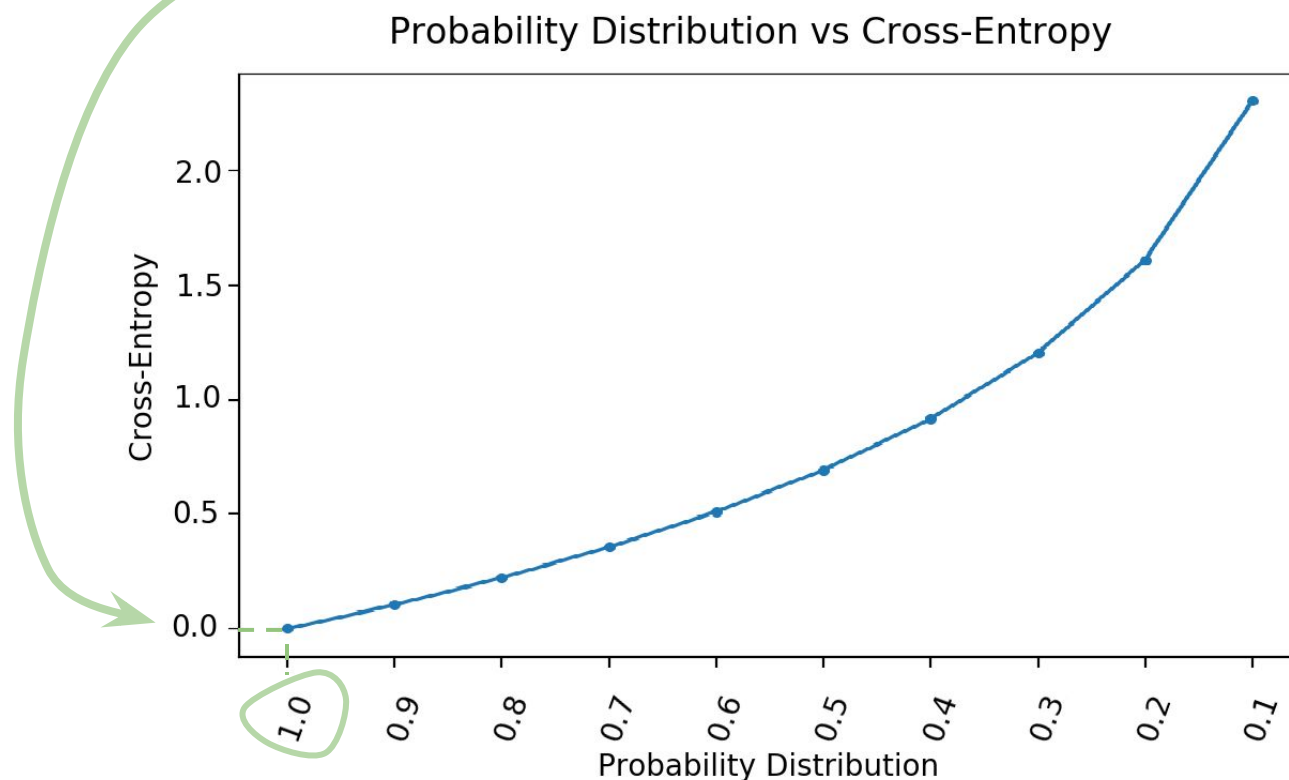
$$\text{CE}(y, \hat{y}) = - \sum_{i=1}^N (y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i))$$



Loss function: cross-entropy

$y \in \{0, 1\}$ – target
 $\hat{y} \in \mathbb{R}$ – prediction

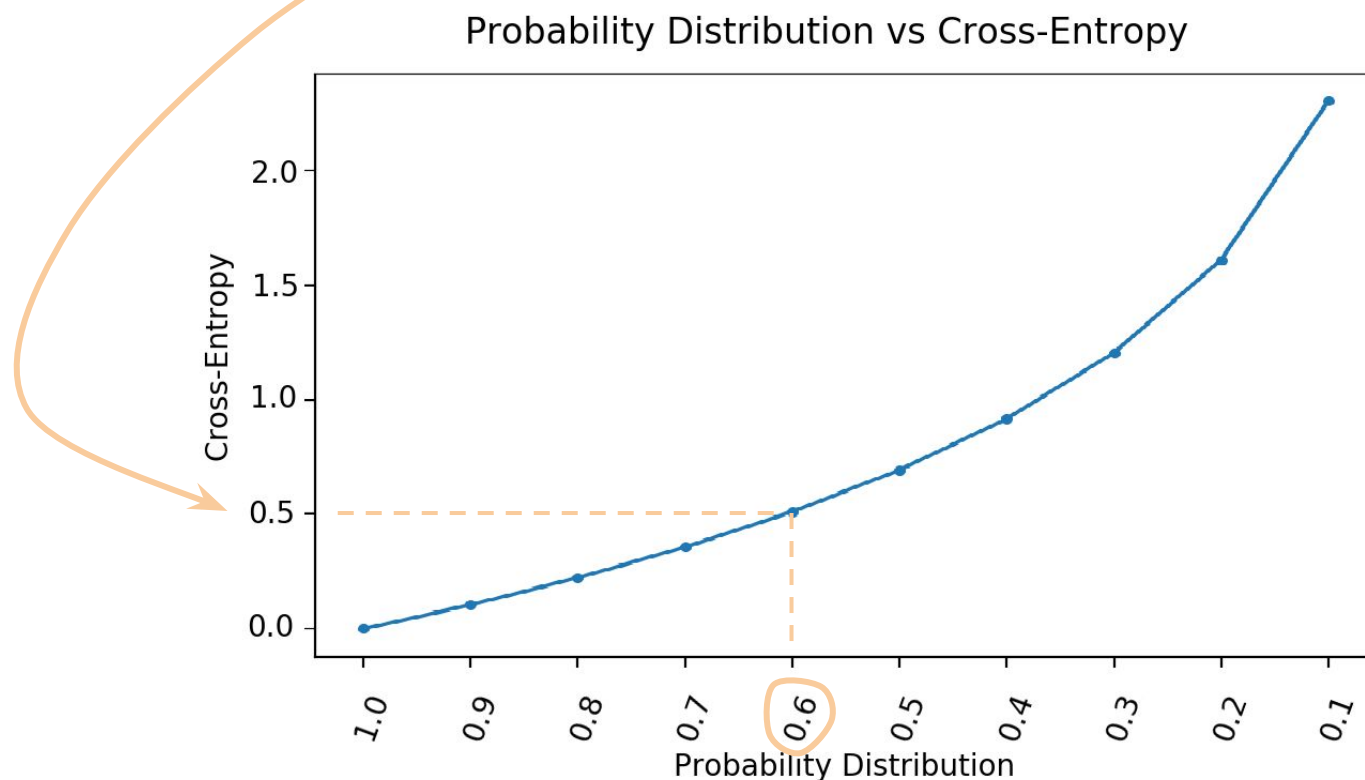
$$\text{CE}(y, \hat{y}) = - \sum_{i=1}^N (y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i))$$



Loss function: cross-entropy

$y \in \{0, 1\}$ – target
 $\hat{y} \in \mathbb{R}$ – prediction

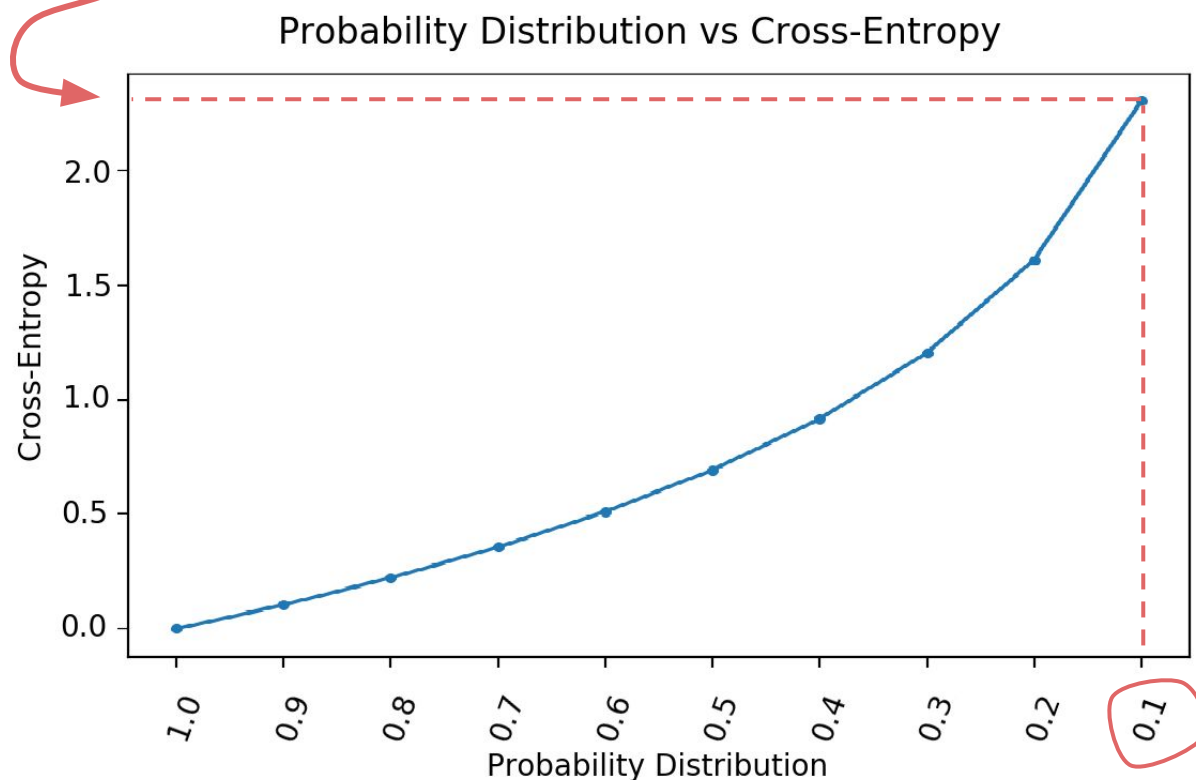
$$\text{CE}(y, \hat{y}) = - \sum_{i=1}^N (y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i))$$



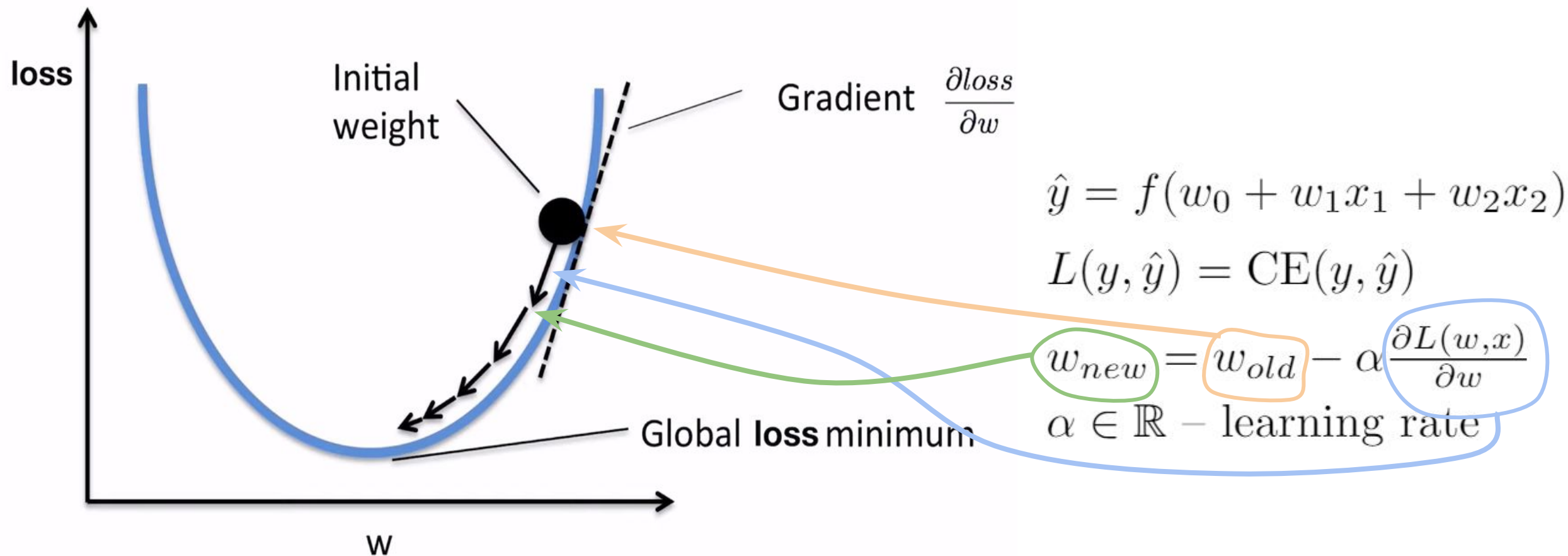
Loss function: cross-entropy

$y \in \{0, 1\}$ – target
 $\hat{y} \in \mathbb{R}$ – prediction

$$\text{CE}(y, \hat{y}) = - \sum_{i=1}^N (y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i))$$



Training process: gradient descent



Logistic regression overview

model equation: $\hat{y} = \sigma\left(\sum_{j=0}^M w_j g(x_j)\right)$

Logistic regression overview

model equation: $\hat{y} = \sigma\left(\sum_{j=0}^M w_j g(x_j)\right)$

loss function: $\text{CE}(y, \hat{y}) = -\sum_{i=1}^N (y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i))$

Logistic regression overview

model equation: $\hat{y} = \sigma\left(\sum_{j=0}^M w_j g(x_j)\right)$

loss function: $\text{CE}(y, \hat{y}) = -\sum_{i=1}^N (y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i))$

weights update: $w_{new} = w_{old} - \frac{\partial L}{\partial w}$

THIS IS YOUR MACHINE LEARNING SYSTEM?

YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.

WHAT IF THE ANSWERS ARE WRONG?

JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.

