Linear Supervised Learning

M. Ndaoud

Previous session

- tools : LLN, CLT, Slutsky Lemma
- estimators : empirical cumulative function, empirical quantile
- graphical statistics : boxplot, qq-plot, heatmap
- Results convergence a.s. and speed of convergence of:

$$\widehat{F}_n, \quad \widehat{q}_{n,p}$$

So far we have note used the statistical model to construct estimators.

Statistical model (1/2)

Question : A model is a prior knowledge on data. How can we leverage this information in order to construct and study estimators that are "more efficient" than model-free estimators as $\widehat{F}_n, \widehat{q}_{n,p}, \ldots$?

Example of a statistical model (2/2)

Problem : A physicist observes the lifetime of radioactive atoms which he decides to model by random variables $X_1, \ldots, X_n$ i.i.d. He wishes to use these data to estimate their underlying law. He can choose between two approaches:

Example of a statistical model (2/2)

Problem : A physicist observes the lifetime of radioactive atoms which he decides to model by random variables $X_1, \ldots, X_n$ i.i.d. He wishes to use these data to estimate their underlying law. He can choose between two approaches:

- "model-free" : by estimating the cumulative function of X_i through $\widehat{F}_n$

Example of a statistical model (2/2)

Problem : A physicist observes the lifetime of radioactive atoms which he decides to model by random variables $X_1, \dots, X_n$ i.i.d. He wishes to use these data to estimate their underlying law. He can choose between two approaches:

- "model-free" : by estimating the cumulative function of X_i through $\widehat{F}_n$
- "model-based" : he knows that lifetimes follow an exponential law $\in \{\mathcal{E}xp(\theta) : \theta > 0\}$. In this case, it is enough to estimate θ by an estimator $\widehat{\theta}_n$ and to approximate the distribution function of X_i by $F_{\widehat{\theta}_n}$ where

$$F_\theta(x) = \mathbb{P}[\mathcal{E}xp(\theta) \leq x] = \begin{cases} 0 & \text{if } x \leq 0 \\ 1 - \exp(-\theta x) & \text{else.} \end{cases}$$

Maximum Likelihood Estimation (MLE)

Sampling model (in $\mathbb{R}$)

- We observe a sample of size n of random variables $X_1, \dots, X_n$.
- The distribution of X_i belongs to the parametric family $\{\mathbb{P}_\theta,\ \theta \in \Theta\}$ (family of distrubtions $\mathbb{R}$). We denote the densities : $\forall \theta \in \Theta, x \in \mathbb{R},\ f(\theta, x)$.
- The distribution of $(X_1, \dots, X_n)$ is given by : $\forall x_1, \dots, x_n \in \mathbb{R}$,

$$\boxed{\prod_{i=1}^{n} f(\theta, x_i)}$$

Example 1 : the normal model

$X_i \sim \mathcal{N}(m, \sigma^2)$, avec $\theta = (m, \sigma^2) \in \Theta = \mathbb{R} \times \mathbb{R}_+ \setminus \{0\}$.

- The normal density is given by:

$$f(\theta, x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(- \frac{(x-m)^2}{2\sigma^2} \Big)$$

- The corresponding distribution is given by : for all $x_1, \dots, x_n \in \mathbb{R}$,

$$\prod_{i=1}^{n} f(\theta, x_i) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\Big(- \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i - m)^2 \Big)$$

Example 2 : Bernoulli model

$X_i \sim \text{Bernoulli}(\theta)$, with $\theta \in \Theta = [0, 1]$

- For all $x \in \{0, 1\}$

$$\boxed{f(\theta, x) = (1-\theta)I(x=0) + \theta I(x=1) = \theta^x (1-\theta)^{1-x}}$$

- The distribution of the observations has density:

$$\prod_{i=1}^{n} \theta^{x_i}(1-\theta)^{1-x_i},$$

for $x_1, \ldots, x_n \in \{0, 1\}$

Maximum likelihood

- Fundamental and essential principle in statistics. Known special cases since the 18th century. General definition: Fisher (1922).
- Provides a first systematic method of constructing an estimator.
- Optimal procedure (in what sense?) under assumptions of regularity of the family $\{\mathbb{P}_\theta, \theta \in \Theta\}$.
- Sometimes difficult to implement in practice $\to$ optimization problem.

The likelihood function

Definition

Under de sampling model (in $\mathbb{R}$) with densities $f(\theta, x)$ the *likelihood function* *of the n-sample $(X_1, \ldots, X_n)$ associated to the family $\{f(\theta, \cdot), \theta \in \Theta\}$ is given by :*

$$\boxed{\theta \in \Theta \mapsto \mathcal{L}_n(\theta, X_1, \ldots, X_n) = \prod_{i=1}^{n} f(\theta, X_i)}$$

- A random function
- The distribution of the observations

Examples

- Example 1: Poisson model. We observe

$$X_1, \ldots, X_n \overset{i.i.d.}{\sim} \text{Poisson}(\theta),$$

$\theta \in \Theta = \mathbb{R}_+ \setminus \{0\}$.

- The density is given by

$$f(\theta, x) = \frac{\theta^x}{x!} e^{-\theta}, \quad x = 0, 1, 2, \ldots.$$

- The associated likelihood function is

$$\begin{aligned} \theta \mapsto \mathcal{L}_n(\theta, X_1, \ldots, X_n) &= \prod_{i=1}^n e^{-\theta} \frac{\theta^{X_i}}{X_i!} \\ &= \frac{1}{\prod_{i=1}^n X_i!} e^{-n\theta} \theta^{\sum_{i=1}^n X_i} \end{aligned}$$

The maximum likelihood principle

1. Case 1 : "θ_1 is more likely than θ_2" if

$$\prod_{i=1}^{n} f(\theta_1, X_i) \geq \prod_{i=1}^{n} f(\theta_2, X_i)$$

2. Case 2 : "θ_2 is more likely than θ_1" if

$$\prod_{i=1}^{n} f(\theta_2, X_i) > \prod_{i=1}^{n} f(\theta_1, X_i)$$

The maximum likelihood principle:

$$\widehat{\theta}_{\mathtt{n}}^{\mathtt{mv}} = \begin{cases} \theta_1 & \text{when } \theta_1 \text{ is more likely} \\ \theta_2 & \text{when } \theta_2 \text{ is more likely} \end{cases}$$

Maximum Likelihood Estimation

- Situation : $X_1, \ldots, X_n \overset{i.i.d.}{\sim} \mathbb{P}_\theta$, $\{\mathbb{P}_\theta, \theta \in \Theta\}$, $\Theta \subset \mathbb{R}^d$, $\theta \mapsto \mathcal{L}_n(\theta, X_1, \ldots, X_n)$ the associated likelihood.

Definition

We call maximum likelihood estimator every estimator $\widehat{\theta}_{\mathrm{n}}^{\mathtt{mv}}$ *satisfying*

$$\mathcal{L}_n(\widehat{\theta}_{\mathrm{n}}^{\mathtt{mv}}, X_1, \ldots, X_n) = \max_{\theta \in \Theta} \mathcal{L}_n(\theta, X_1, \ldots, X_n).$$

- Questions : Existence, uniqueness, statistical properties?

Remarks

- Log-likelihood:

$$\theta \mapsto \ell_n(\theta, X_1, \ldots, X_n) = \log \mathcal{L}_n(\theta, X_1, \ldots, X_n) = \sum_{i=1}^{n} \log f(\theta, X_i).$$

Well-defined if $f(\theta, \cdot) > 0$.

Max. likelihood = max. log-likelihood.

(log-likelihood is usually easier to maximize)

- Likelihood equation :

$$\nabla_\theta \ell_n(\theta, X_1, \ldots, X_n) = 0$$

Linear Regression

Example 1: Gaussian Linear regression

Assume that we observe $(X_1, Y_1), \ldots, (X_n, Y_n)$ following the model

$$Y_i = \langle X_i, \beta \rangle + \sigma \xi_i,$$

where ξ_i are i.i.d. random standard normal variables.

- The distribution of $Y|X$ is given by $\mathcal{N}(\langle X, \beta \rangle, \sigma^2)$, where β is the parameter.
- Likelihood

$$\mathcal{L}_n(\beta, (X_1, Y_1), \ldots, (X_n, Y_n)) = C \exp\big(-\tfrac{1}{2\sigma^2} \sum_{i=1}^{n} (Y_i - \langle X_i, \beta \rangle)^2\big).$$

- Log-likelihood

$$\ell_n(\beta, (X_1, Y_1), \ldots, (X_n, Y_n)) = \log(C) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (Y_i - \langle X_i, \beta \rangle)^2.$$

Example 1: Gaussian Linear regression

- The optimization problem to solve becomes:

$$\min_{\beta} \sum_{i=1}^{n} (Y_i - \langle X, \beta \rangle)^2 = \min_{\beta} \| Y - X\beta \|^2.$$

- Maximizing the likelihood is equivalent in this case to minimizing the least squares.

Empirical risk minimization

In both cases, the estimation problem boils down to minimization of convex functions.

- Regression:

$$\min_{\beta} \sum_{i=1}^{n} (Y_i - \langle X, \beta \rangle)^2.$$

- Classification:

$$\min_{\beta} \sum_{i=1}^{n} \log\left(1 + e^{-Y_i \langle X_i, \beta \rangle}\right).$$

Example 2: Logistic regression

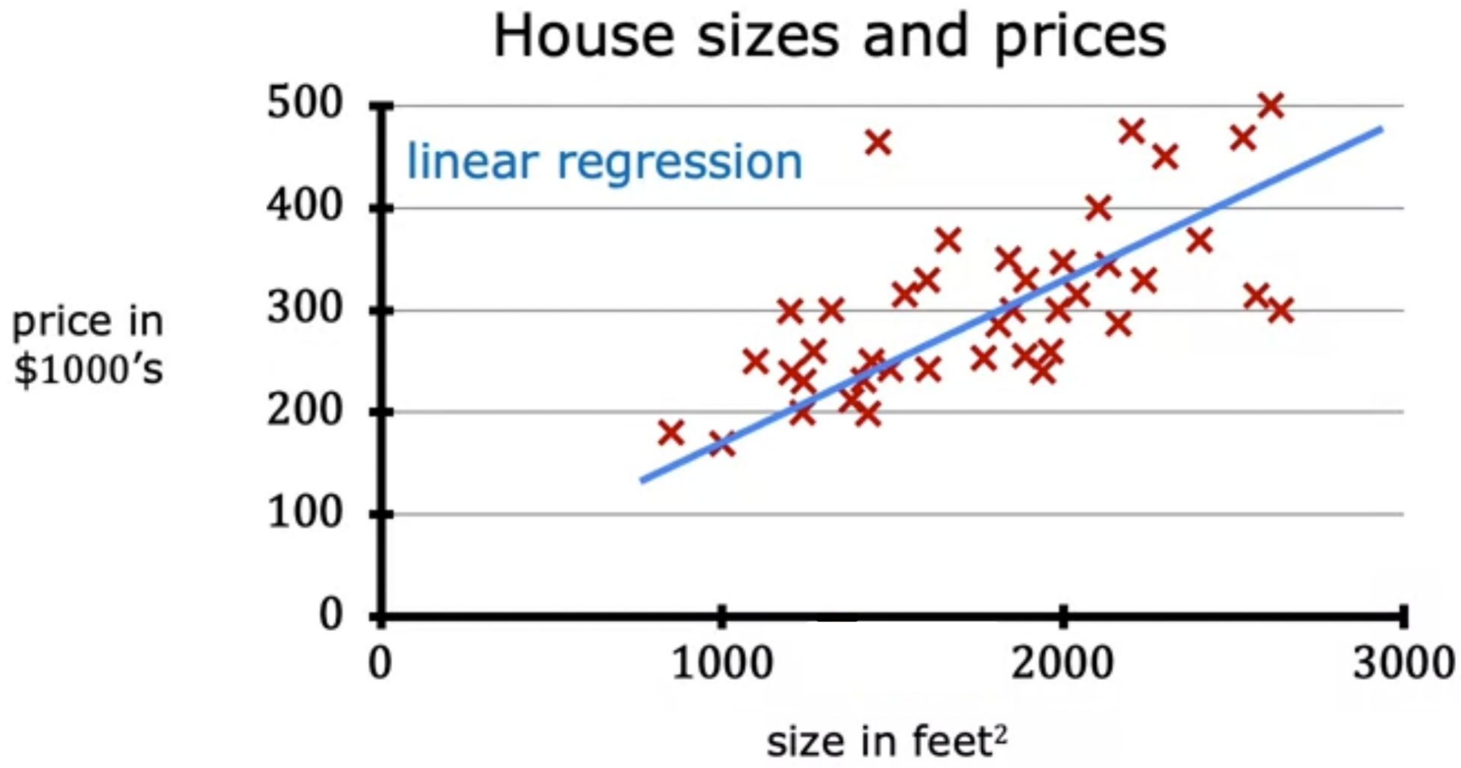
Assume that we observe $(X_1, Y_1), \ldots, (X_n, Y_n)$, where $Y \in \{-1, +1\}$, following the model

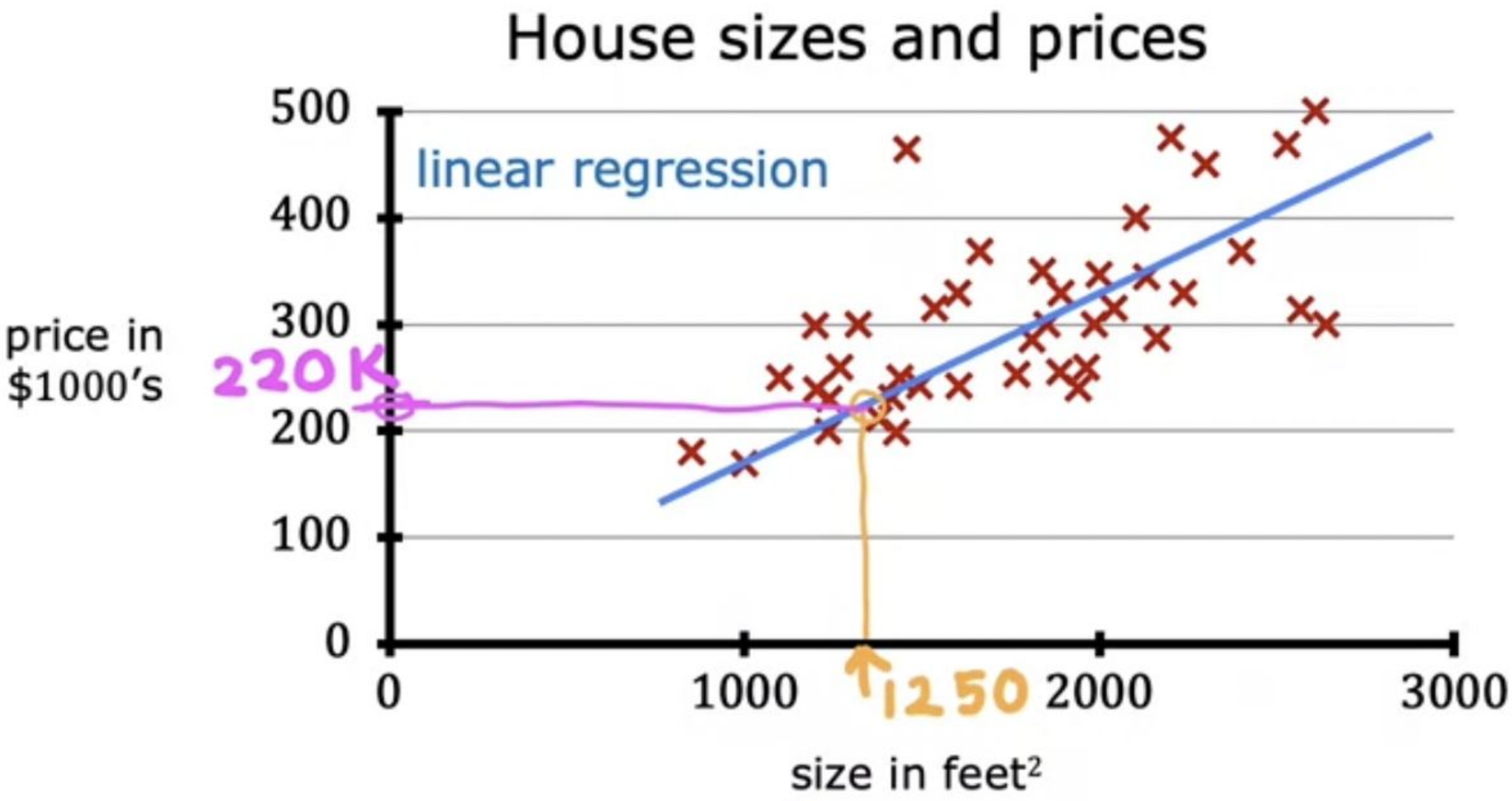
$$\mathbb{P}\left(Y_i = 1 | X_i\right) = \frac{1}{1 + e^{-\langle X_i, \beta \rangle}}.$$

- The distribution of $Y|X$ is a Bernoulli distribution depending on a parameter β.
- Log-likelihood

$$\ell_n(\beta, (X_1, Y_1), \ldots, (X_n, Y_n)) = -\sum_{i=1}^{n} \log\left(1 + e^{-Y_i \langle X_i, \beta \rangle}\right).$$


House sizes and prices
500
400
300
200
100
0
price in
$1000's
0
1000
2000
3000
size in feet²


House sizes and prices
linear regression
price in
$1000's
500
400
300
200
100
0
1000
2000
3000
size in feet²

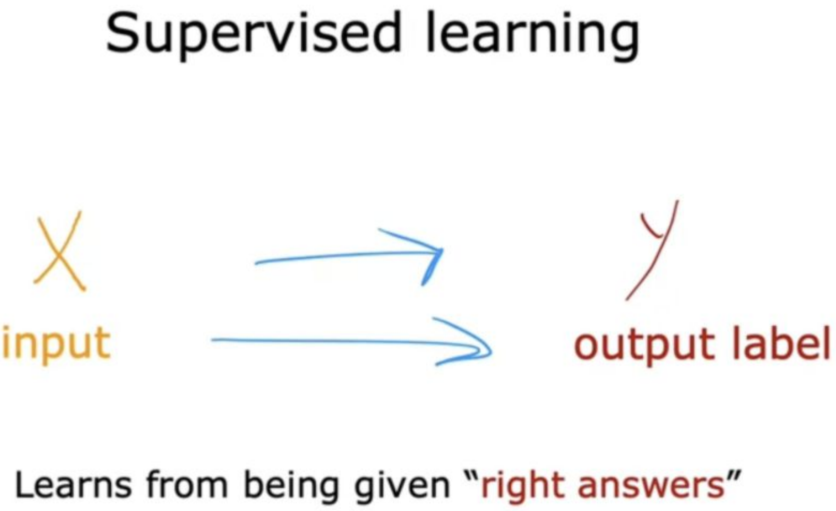

House sizes and prices
linear regression
price in
$1000's
220K
500
400
300
200
100
0
0
1000
1250
2000
3000
size in feet²

ML - Supervised learning

AI that learns ________________________

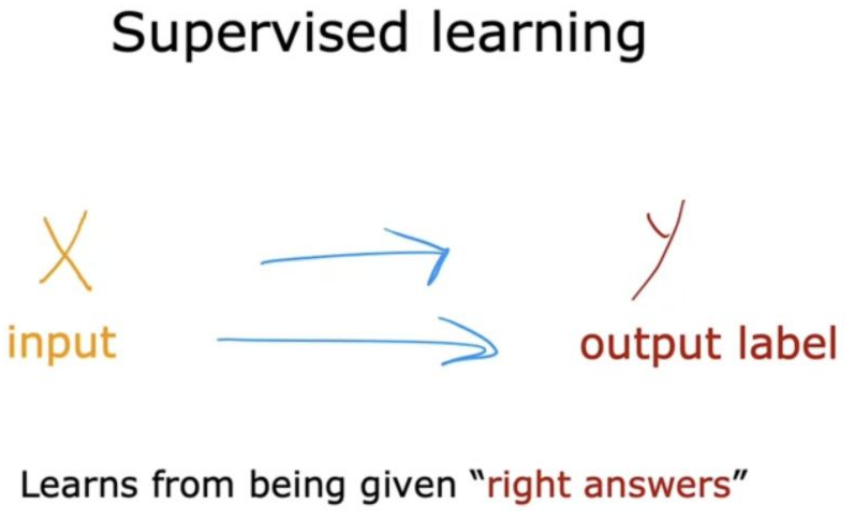
ML - Supervised learning

AI that learns “A to B”, or “input to output” mappings.



ML - Supervised learning

AI that learns “A to B”, or “input to output” mappings.



>95% of the use cases in business

ML - Supervised learning - Recap

2 main types:

✓ *Regression* : predict **XXXXXX** out of **<u>XXXXXXX</u>**

Ex: __________________

✓ *Classification* : predict **XXXXXX** out of **<u>XXXXXXXX</u>**

Ex: __________________

ML - Supervised learning - Recap

2 main types:

✓ *Regression* : predict **numbers** out of **<u>infinitely</u>** many possible numbers

Ex: price prediction in real estate

✓ *Classification* : predict **categories** out of **<u>finite</u> (and small)** number of possible outputs

Ex: spam or not spam email, classifier of t-shirt size (XS,S,M,L,XL,XXL)

Data table

size in feet²	price in $1000's
2104	400
1416	232
1534	315
852	178
...	...
3210	870

Data table

size in feet²	price in $1000's
2104	400
1416	232
1534	315
852	178
...	...
3210	870

Data table

size in feet²	price in $1000's
2104	400
1416	232
1534	315
852	178
...	...
3210	870

Training set: data used to train model

size in feet^2	price in \$1000's
2104	400
1416	232
1534	315
852	178
...	...
3210	870

Technical terminology

	x size in feet^2	y price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

$x = 2104 \quad y = 400$

x = "input" variable
feature

y = "output" variable
"target" variable

Technical terminology

	x size in feet^2	y price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

$x = 2104 \quad y = 400$

$m = 47$

x = "input" variable
feature

y = "output" variable
"target" variable

m = number of training examples

Technical terminology

	x size in feet^2	y price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

$m = 47$

$x = 2104 \quad y = 400$

$(x, y) = (2104, 400)$

x = "input" variable
feature

y = "output" variable
"target" variable

m = number of training examples

(x, y) = single training example

Technical terminology

	x size in feet^2	y price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

$m = 47$

$x^{(1)} = 2104$ $y^{(1)} = 400$

$(x^{(1)}, y^{(1)}) = (2104, 400)$

$x^{(2)} = 1416$ $x^{(2)} \neq x^2$ not exponent

x = "input" variable
feature

y = "output" variable
"target" variable

m = number of training examples

(x, y) = single training example

$(x^{(i)}, y^{(i)})$

$(x^{(i)}, y^{(i)})$ = i^{th} training example

index

(1^{st}, 2^{nd}, 3^{rd} ...)

Training Data set

size in feet²	price in $1000's
2104	400
1416	232
1534	315
852	178
...	...
3210	870

$$(x^{(i)}, y^{(i)})$$

Training Data set

	x size in feet^2	y price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

$m = 47$

$$\left(\left(x^{(i)}, y^{(i)} \right) \right)_{i=1..m}$$

Training Data set

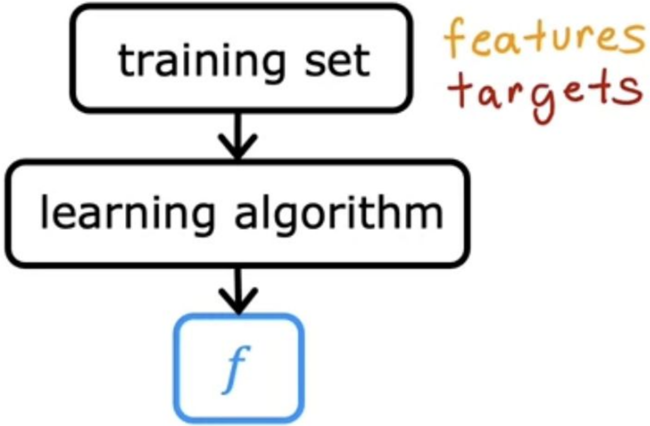
	x size in feet^2	y price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

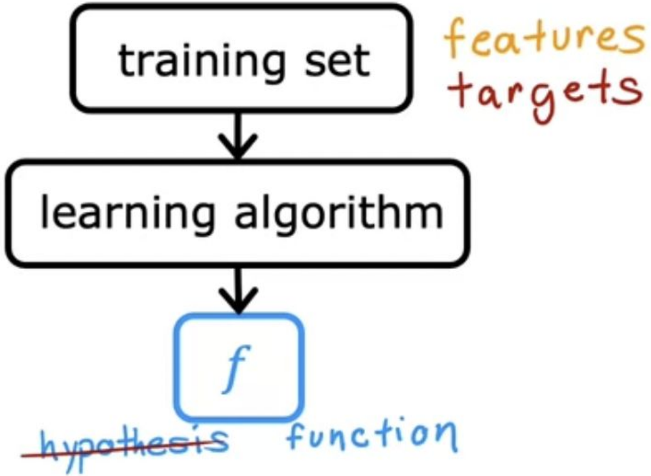
$m = 47$

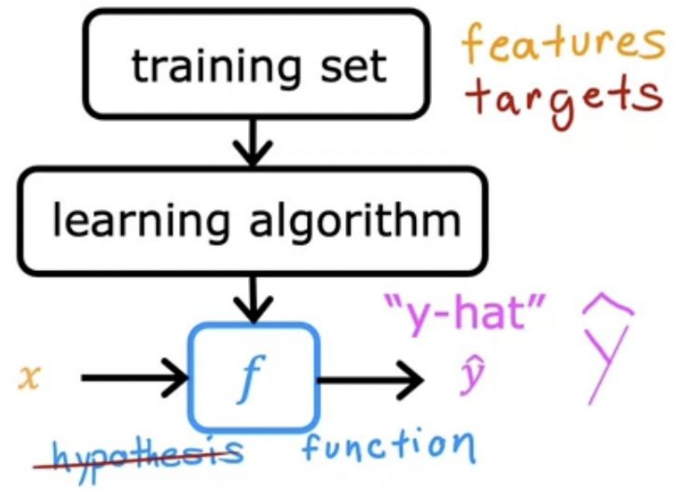
$$\left(\left(x^{(i)}, y^{(i)} \right) \right)_{i=1..m}$$

training set

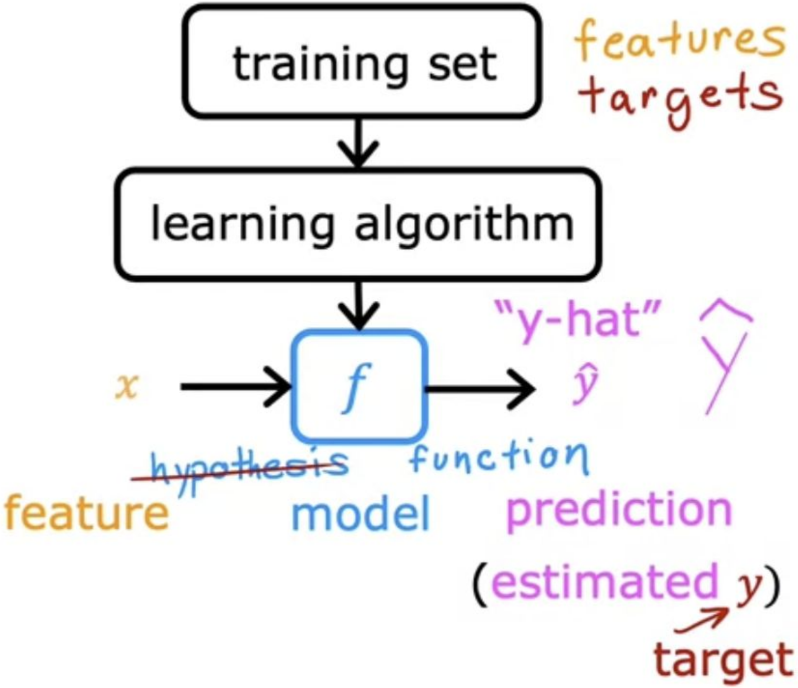
features
targets

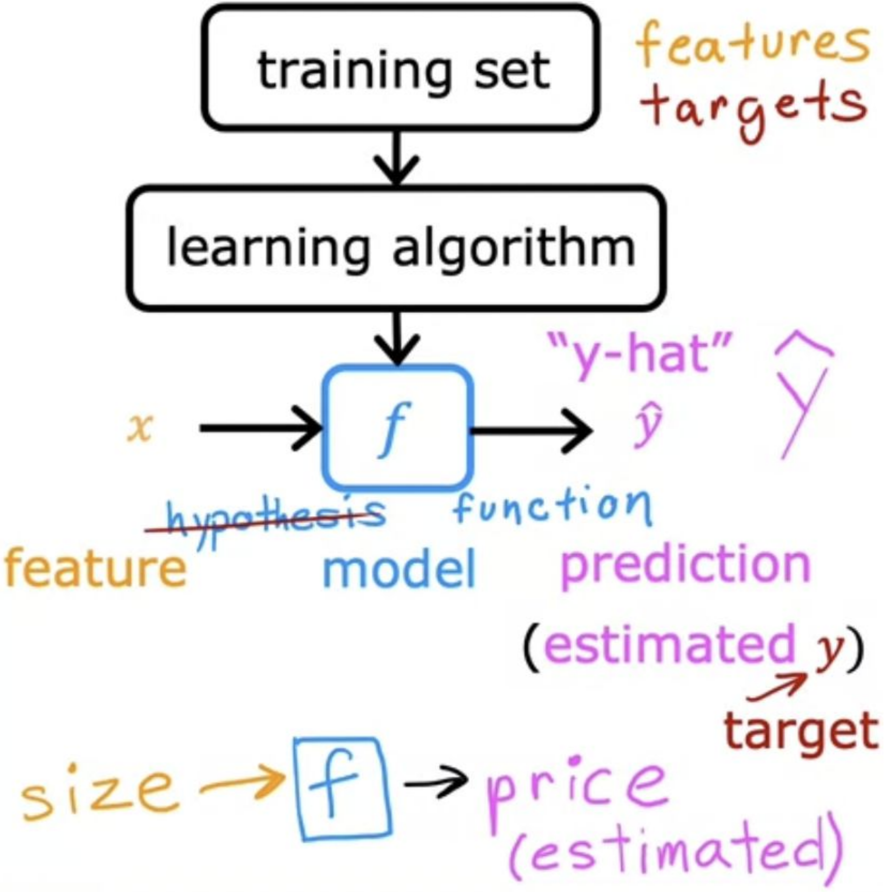

training set
features
targets
learning algorithm
f


training set
features
targets
learning algorithm
f
~~hypothesis~~
function

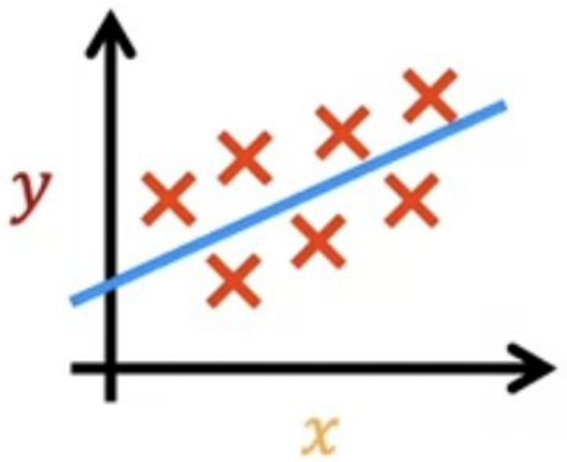


training set
features
targets
learning algorithm
"y-hat"
x
f
ŷ
~~hypothesis~~ function

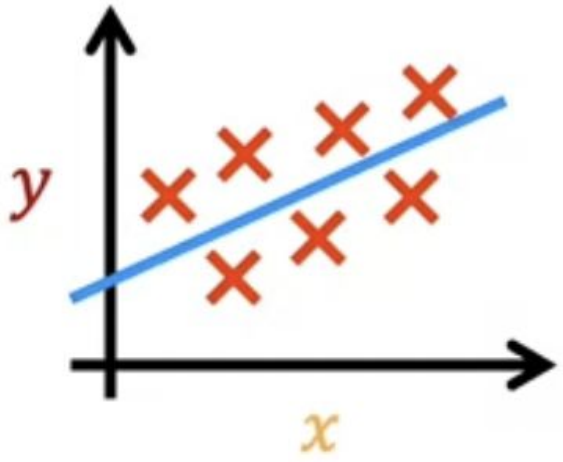

training set
features
targets
learning algorithm
"y-hat"
x
f
$\hat{y}$
~~hypothesis~~ function
feature
model
prediction
(estimated y)
target


training set
features
targets
learning algorithm
"y-hat"
x
f
$\hat{y}$
~~hypothesis~~ function
feature
model
prediction
(estimated y)
target
size
f
price
(estimated)

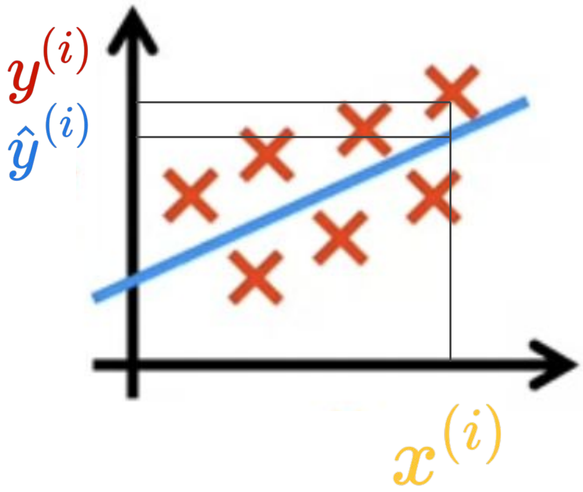
How to represent f ?



$$f_{w,b}(x) = wx + b$$
$$f(x) = wx + b$$



$$\hat{y} = f(x) = wx + b$$

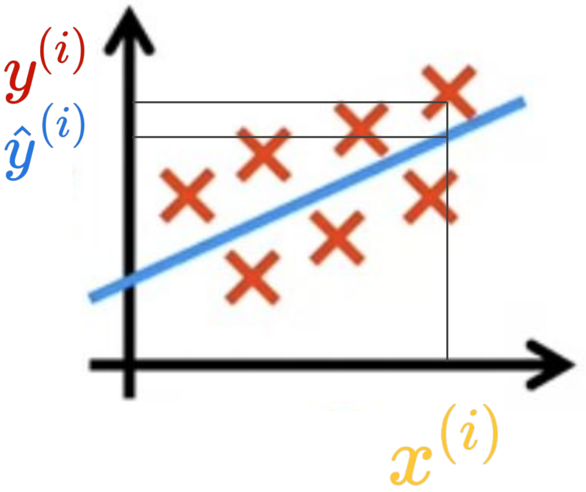


$$f(x^{(i)}) = wx^{(i)} + b \sim y^{(i)}$$

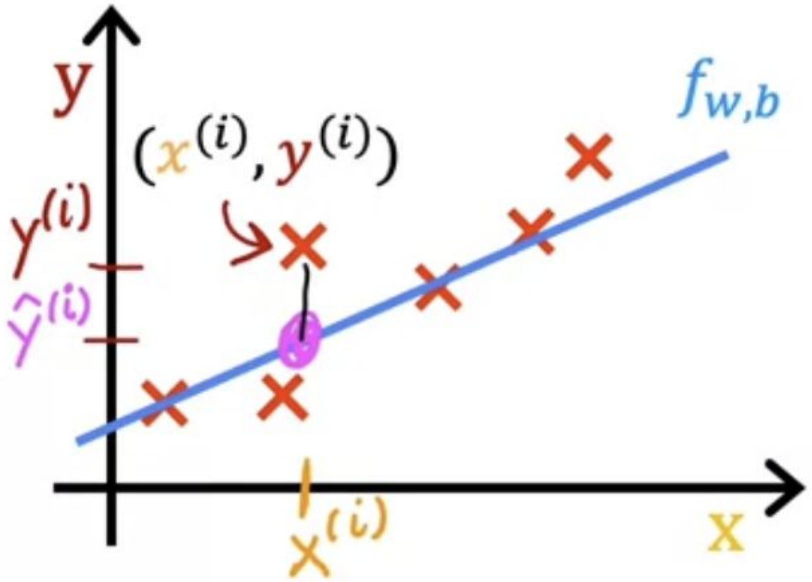
$$\hat{y}^{(i)} \sim y^{(i)}$$

Univariate Linear regression

Single feature = just one variable $x^{(i)}$



$$f(x^{(i)}) = wx^{(i)} + b \sim y^{(i)}$$



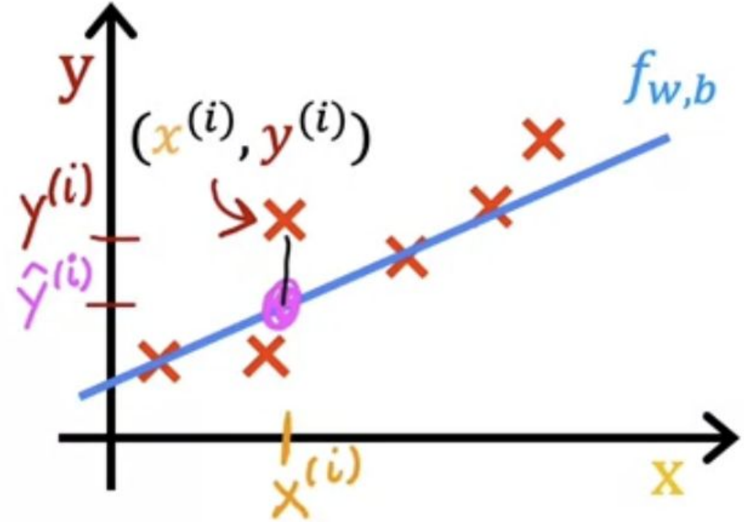
$\hat{y}^{(i)} = f_{w,b}(x^{(i)})$

$f_{w,b}(x^{(i)}) = wx^{(i)} + b$

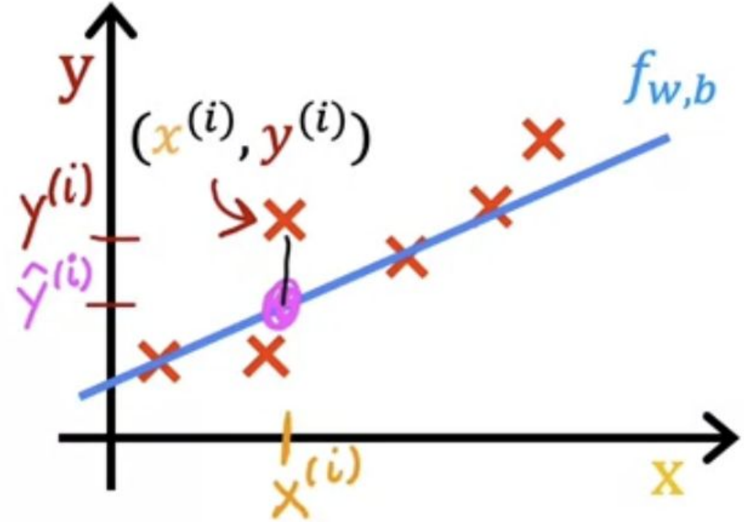
Find w, b:
$\hat{y}^{(i)}$ is close to $y^{(i)}$ for all $(x^{(i)}, y^{(i)})$.

Find w, b :
$\hat{y}^{(i)}$ is close to $y^{(i)}$ for all $\left(x^{(i)}, y^{(i)}\right)$.

To do that, let’s build a “cost function”

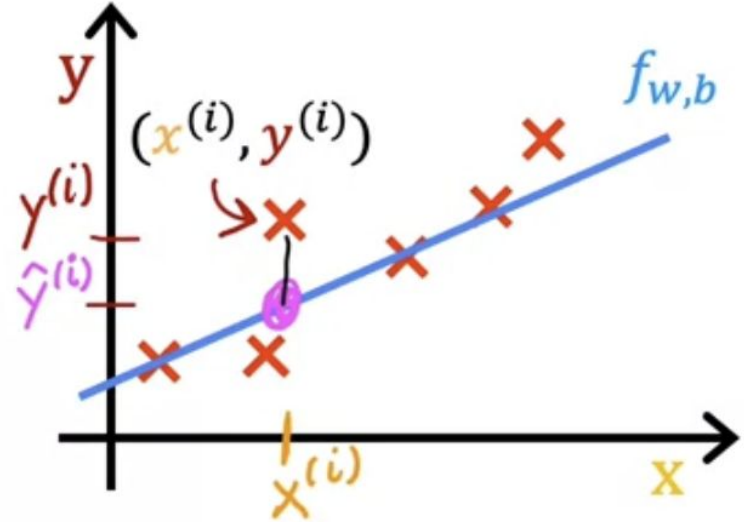


$$\Big(\underbrace{\hat{y}^{(i)} - y^{(i)}}_{\text{error}}\Big)^2$$



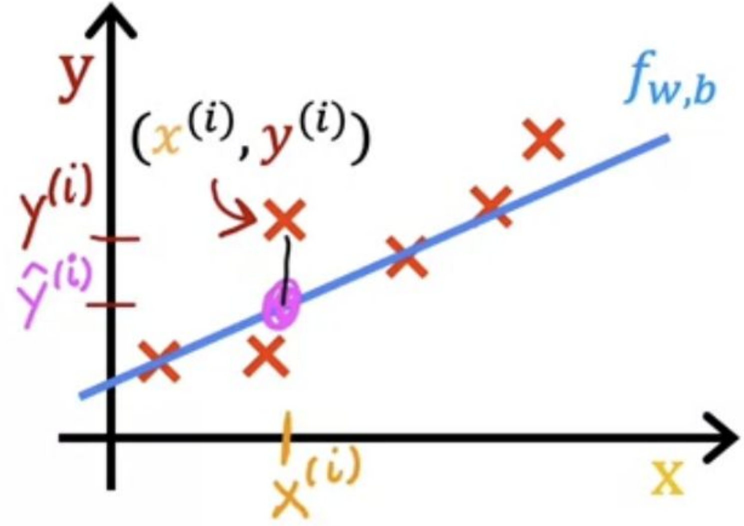
$$\sum_{i=1}^{m} \left(\underbrace{\hat{y}^{(i)} - y^{(i)}}_{\text{error}} \right)^2$$

m = number of training examples



$$\frac{1}{m}\sum_{i=1}^{m}\left(\underbrace{\hat{y}^{(i)} - y^{(i)}}_{\text{error}}\right)^2$$

m = number of training examples



Cost function: Squared error cost function

$$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(\underbrace{\hat{y}^{(i)} - y^{(i)}}_{\text{error}})^2$$

m = number of training examples