Session 2

Machine Learning

Regression

Regression

Linear relationship among variables

Expectation

$$\mathrm{E}[x] = \sum_x xp(x) \quad \text{if } x \text{ is discrete}$$

$$\mathrm{E}[x] = \int xp(x)dx \quad \text{if } x \text{ is continuous}$$

$$\mathrm{E}[f(x)] = \sum_x f(x)p(x)$$

$$\mathrm{E}[f(x,y)] = \sum_{x,y} f(x,y)p(x,y)dxdy$$

$$\mathrm{E}[\alpha x] = \alpha \mathrm{E}[x]$$
$$\mathrm{E}[f(x) + g(x)] = \mathrm{E}[f(x)] + \mathrm{E}[g(x)]$$

$$\mathrm{E}[\alpha] = \alpha$$

If x and y are independence: $\mathrm{E}[f(x)g(y)] = \mathrm{E}[f(x)]\mathrm{E}[g(y)]$

Different approaches to statistics

Frequentist

Bayesian

Uncertainty interpretation

Frequentist

Bayesian

Subjective

Objective

Data and parameters

Frequentist

Bayesian

Θ is random
X is fixed

Θ is fixed
X is random

Data and parameters

Frequentist

Bayesian

For any $|X|$

$|X| \gg |\theta|$

Training

Frequentist

Bayesian

Maximum Likelihood:

$$\widehat{\theta} = arg\max_{\theta} P(X|\theta)$$

Frequentist

Bayesian

Bayes theorem:

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)}$$

Classification

Training:

$$P(\theta|X_{\mathrm{tr}}, y_{\mathrm{tr}}) = \frac{P(y_{\mathrm{tr}}|X_{\mathrm{tr}}, \theta)P(\theta)}{P(y_{\mathrm{tr}}|X_{\mathrm{tr}})}$$

Prediction:

$$P(y_{\mathrm{ts}}|X_{\mathrm{ts}}, X_{\mathrm{tr}}, y_{\mathrm{tr}}) = \int P(y_{\mathrm{ts}}|X_{\mathrm{ts}}, \theta)P(\theta|X_{\mathrm{tr}}, y_{\mathrm{tr}})d\theta$$

Quiz

In bayesian approach, prediction is:

- ○ *A prediction of best-fitted values of parameters*
- ○ *A weighted average of output of our model for all possible values of parameters*

Quiz

In bayesian approach, prediction is:

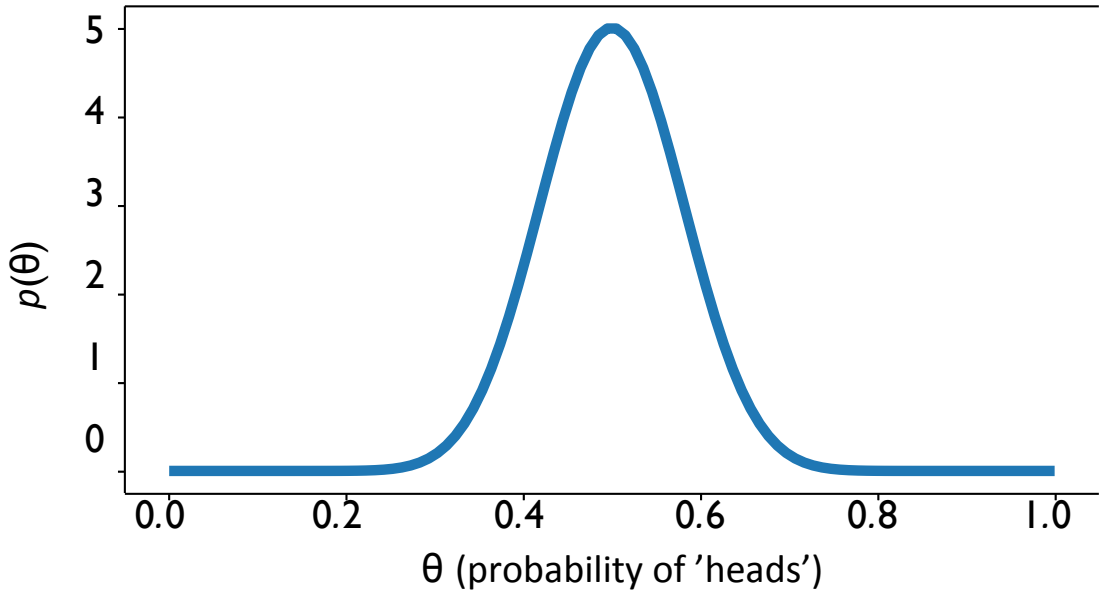
- ○ *A prediction of best-fitted values of parameters*
- ● *A weighted average of output of our model for all possible values of parameters*

Regularization

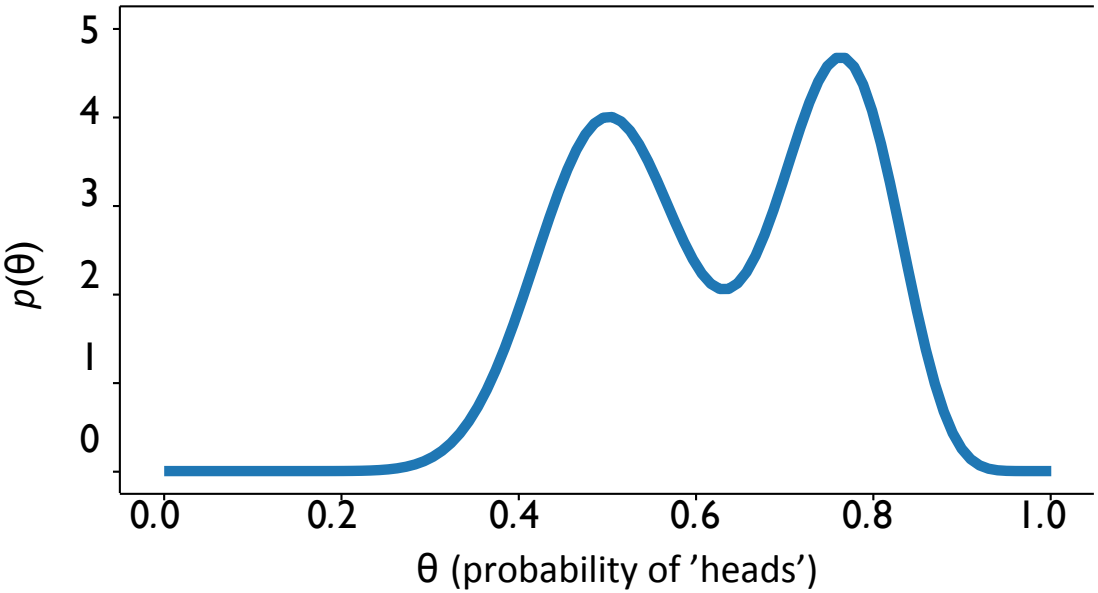
Regularizer

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)}$$

Regularization



Regularization



Online learning

New prior

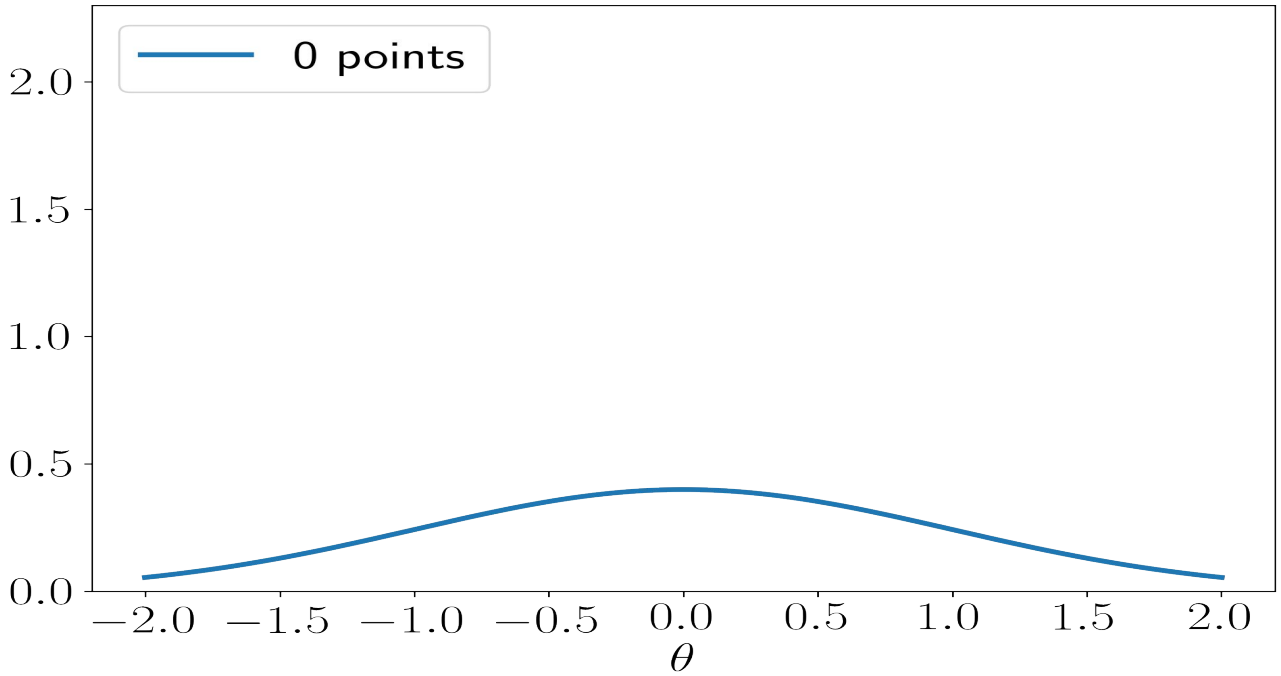
Likelihood

Prior

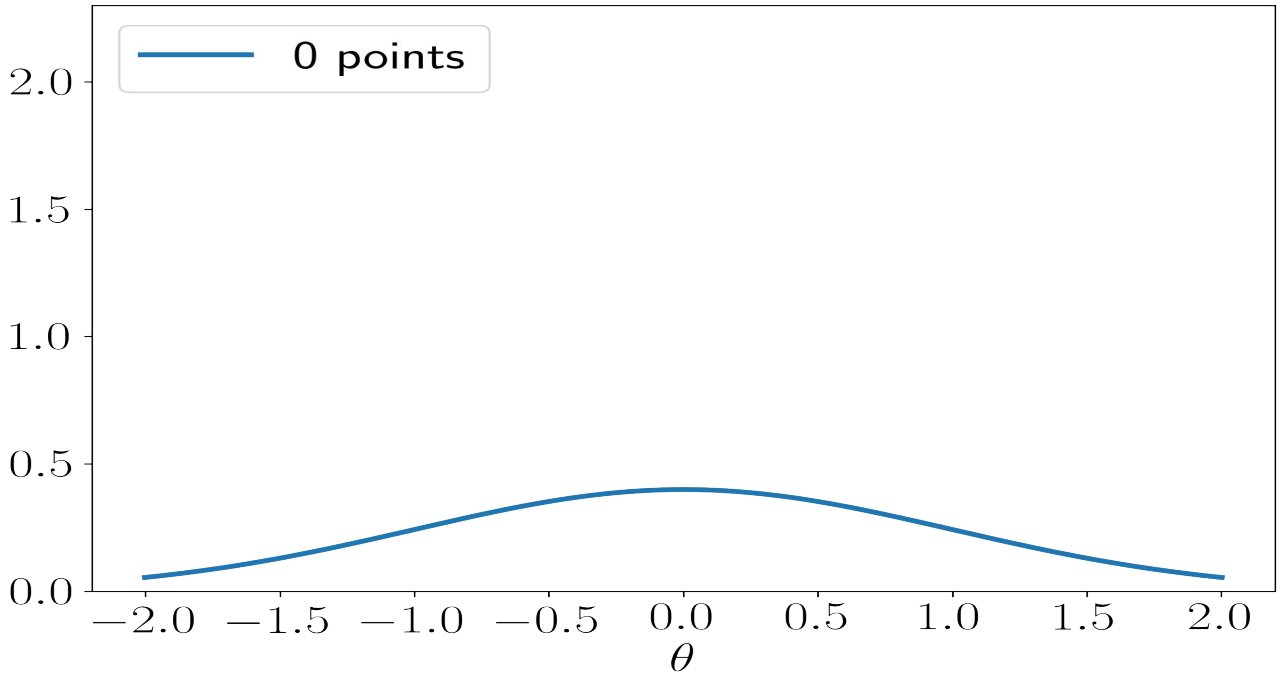
$$P_k(\theta) = P(\theta|x_k) = \frac{P(x_k|\theta)P_{k-1}(\theta)}{P(x_k)}$$

Posterior

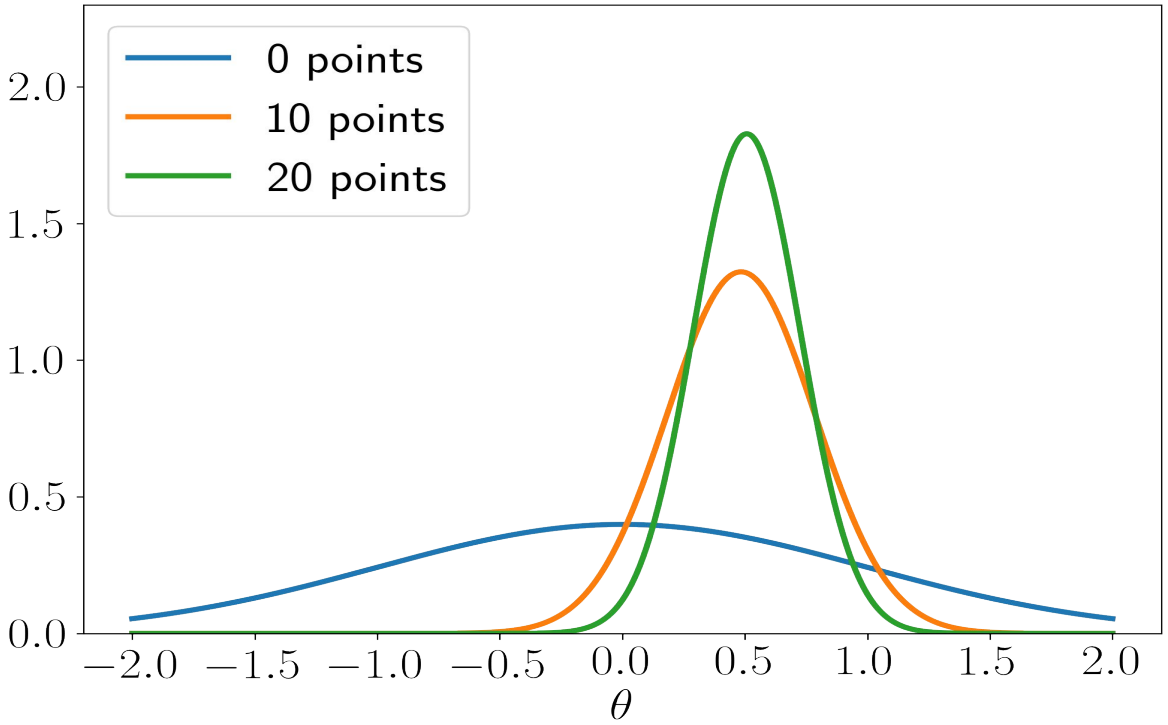
Online learning



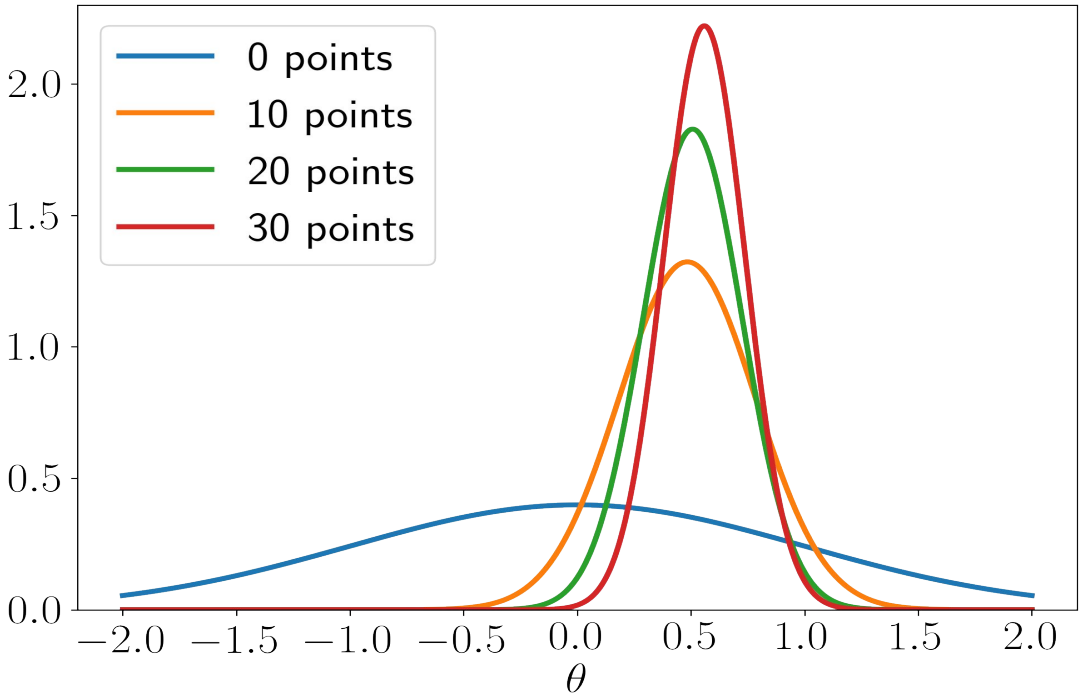
Online learning



Online learning



Online learning

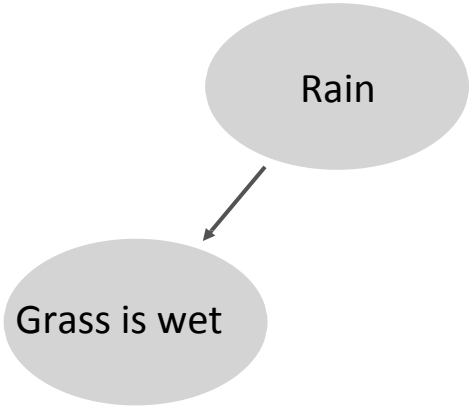


HOW TO DEFINE A MODEL?

*Bayesian network**

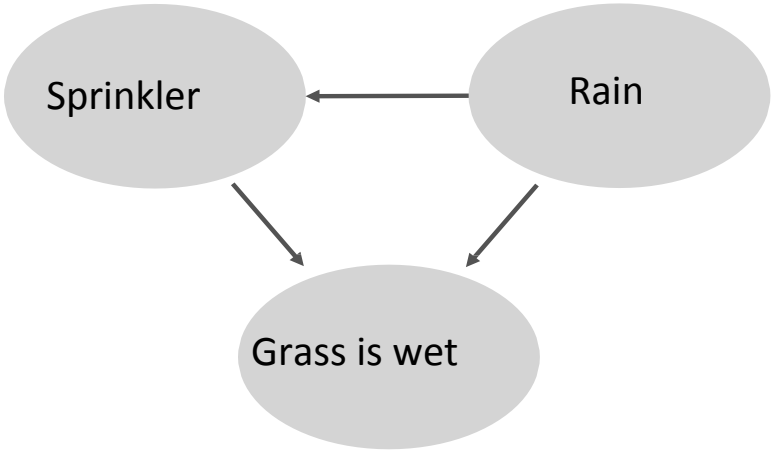
Nodes: random variables

Edges: direct impact



Nodes: random variables

Edges: direct impact



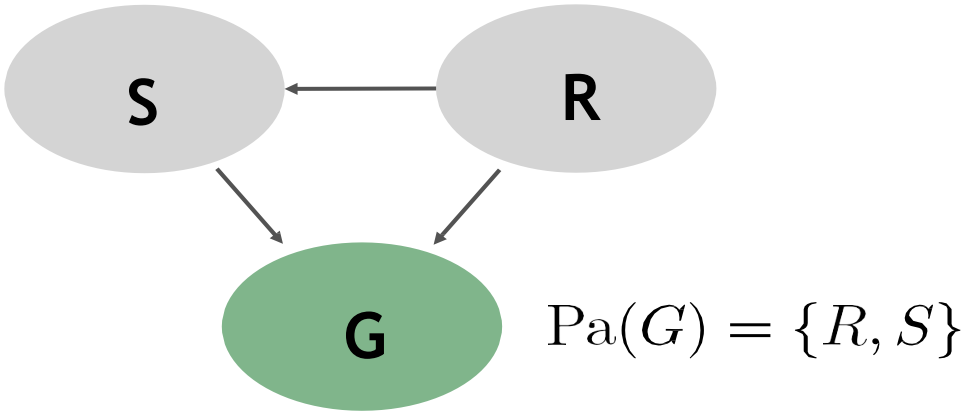
* Don’t mix up with Bayesian neural network

Probabilistic model from Bayesian network

Model: joint probability over all variables

$$P(X_1, \ldots, X_n) = \prod_{k=1}^{n} P\Big(X_k | \mathrm{Pa}(X_k)\Big)$$

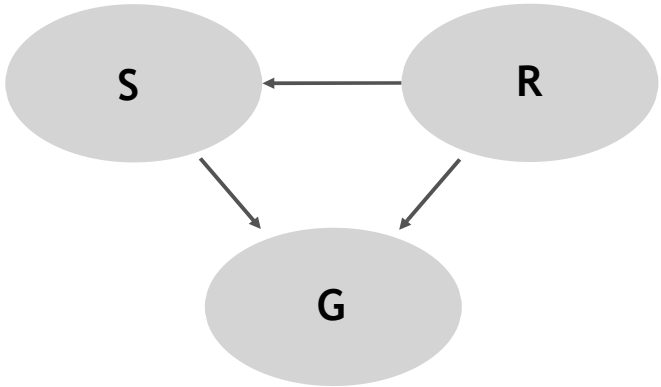
Parents



$$\mathrm{Pa}(G) = \{R, S\}$$

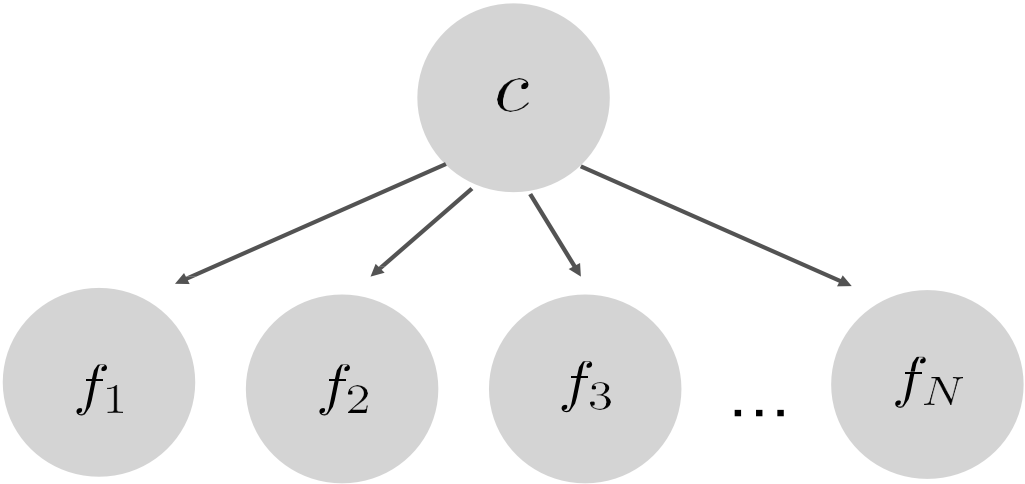
Probabilistic model from Bayesian network

Model: joint probability over all variables



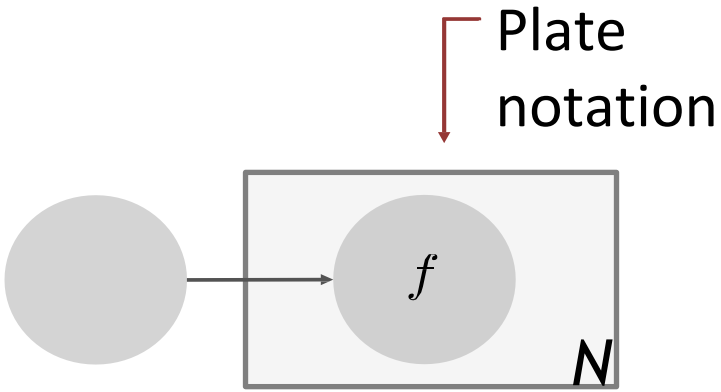
$$P(S, R, G) = P(G|S, R) \cdot P(S|R) \cdot P(R)$$

Naïve Bayes classifier



$$P(c, f_1, \ldots, f_N) = P(c) \prod_{i-1}^{N} P(f_i|c)$$

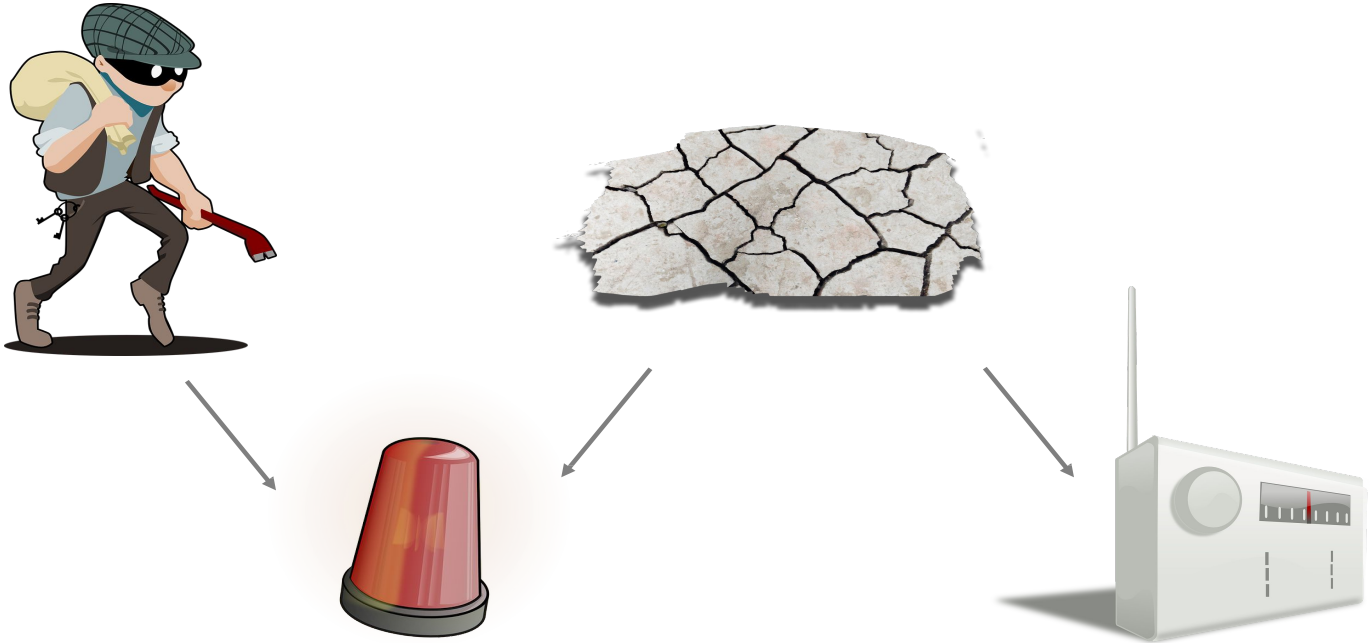
Naïve Bayes classifier



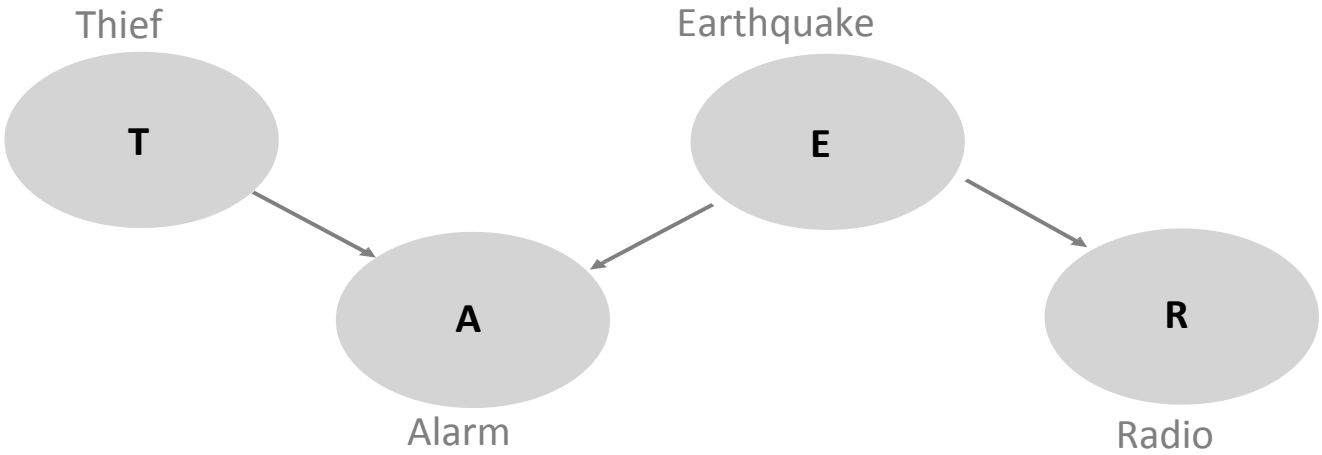
$$P(c, f_1, \ldots, f_N) = P(c) \prod_{i=1}^{N} P(f_i|c)$$

THIEF AND ALARM

Model

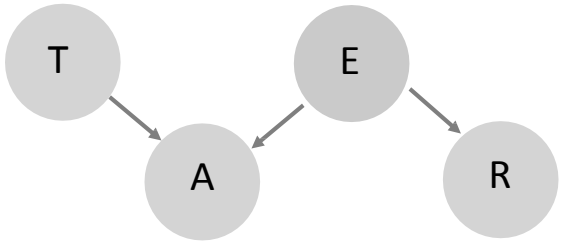


Model



$$P(T, A, E, R) = P(T)\, P(E)\, P(A|T,E)\, P(R|E)$$

Distributions



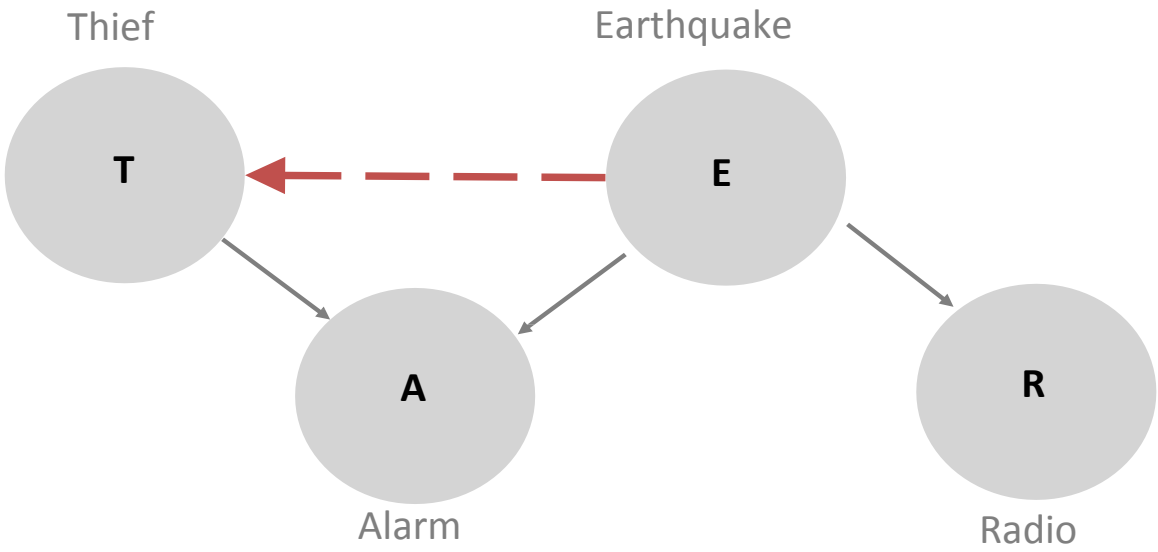
Priors	
$P(\mathbf{T}=\mathbf{1})$	10^{-3}
$P(\mathbf{E}=\mathbf{1})$	10^{-2}

$P(\mathbf{A}=\mathbf{1}\|\mathbf{T},\mathbf{E})$	$\mathbf{E}=\mathbf{0}$	$\mathbf{E}=\mathbf{1}$
$\mathbf{T}=\mathbf{0}$	0	1/10
$\mathbf{T}=\mathbf{1}$	1	1

$P(\mathbf{R}\|\mathbf{E})$	
$\mathbf{E}=\mathbf{0}$	0
$\mathbf{E}=\mathbf{1}$	1/2

P(T|A)? P(T|A, R)?

Correct model



Regression

How much is my house worth?

Look at recent sales in my neighborhood

- How much did they sell for?

Input and Output

Data

input *output*

$(x_1 = \text{sq.ft.}, y_1 = \$)$

$(x_2 = \text{sq.ft.}, y_2 = \$)$

$(x_3 = \text{sq.ft.}, y_3 = \$)$

$(x_4 = \text{sq.ft.}, y_4 = \$)$

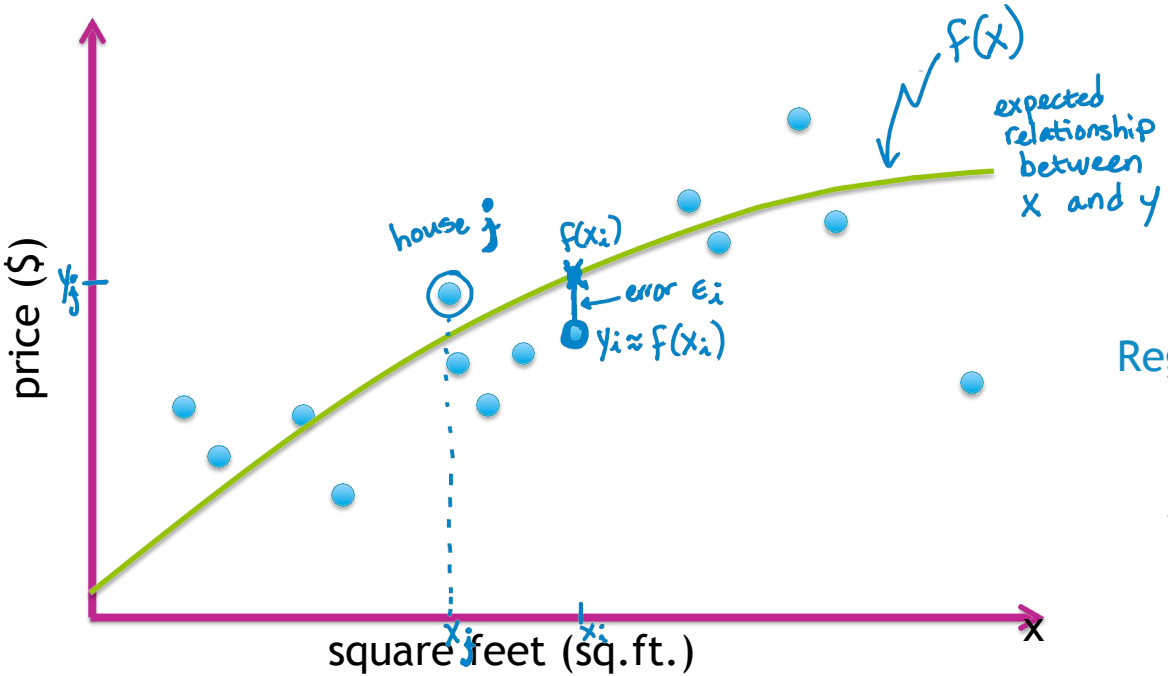
$(x_5 = \text{sq.ft.}, y_5 = \$)$

⋮

Input vs. Output:

- y is the quantity of interest
- assume y can be predicted from x

How we assume the world works



Regression model:

$$y_i = f(x_i) + \epsilon_i$$

$E[\epsilon_i] = 0$ ← equally likely that error is + or −

↑ expected value

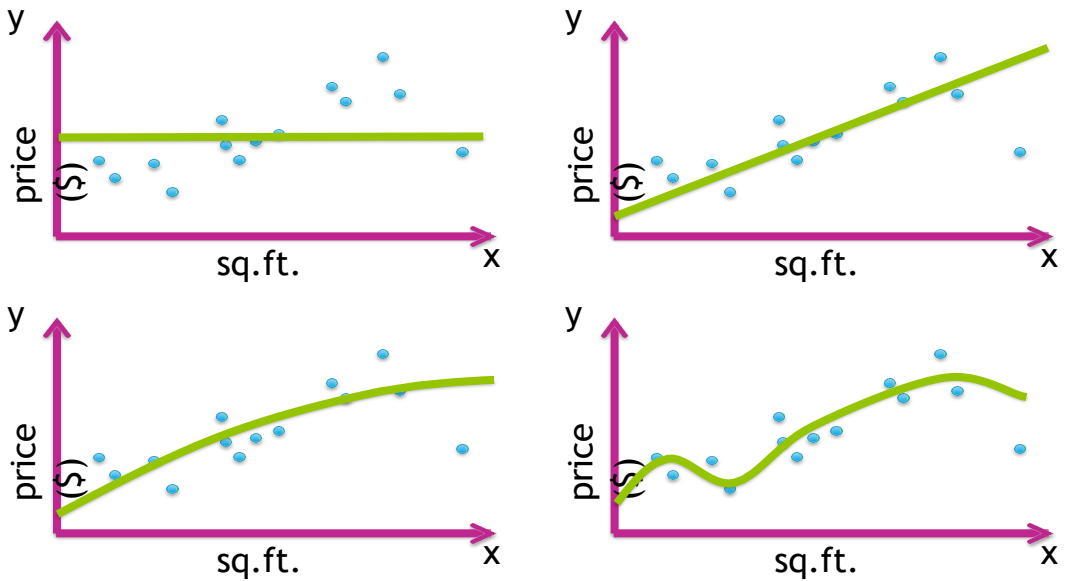
⇓

y_i is equally likely to be above or below $f(x_i)$

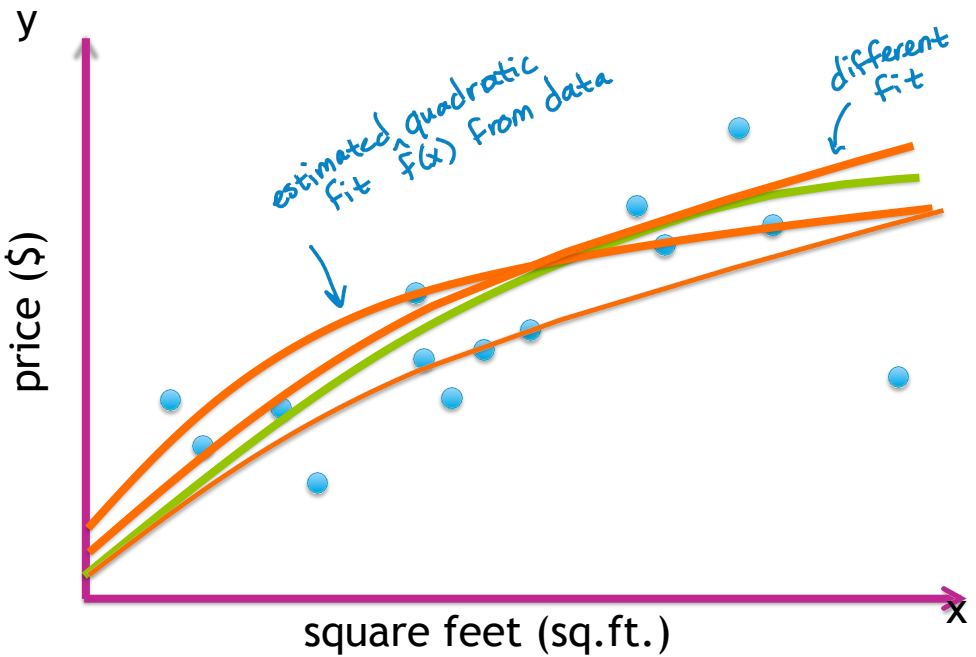
How we assume the world works

"Essentially, all models are wrong, but some are useful." George Box, 1987.

Task 1–Which model f(x)?

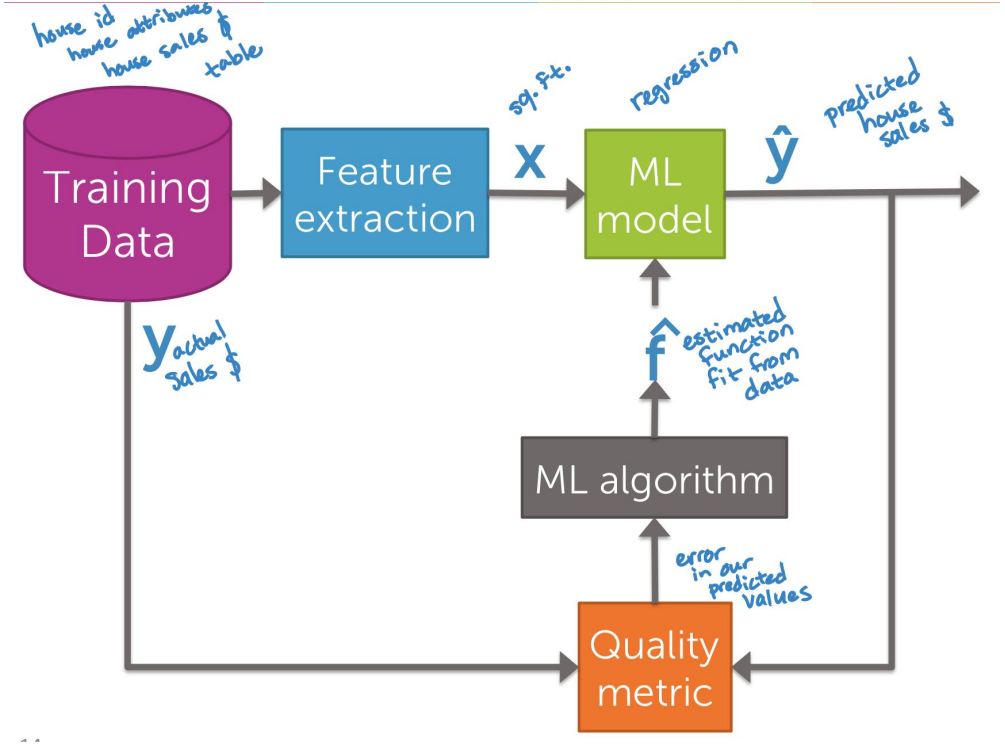


Task 2 – For a given model f(x), estimate function $\hat{f}(x)$ *from data*

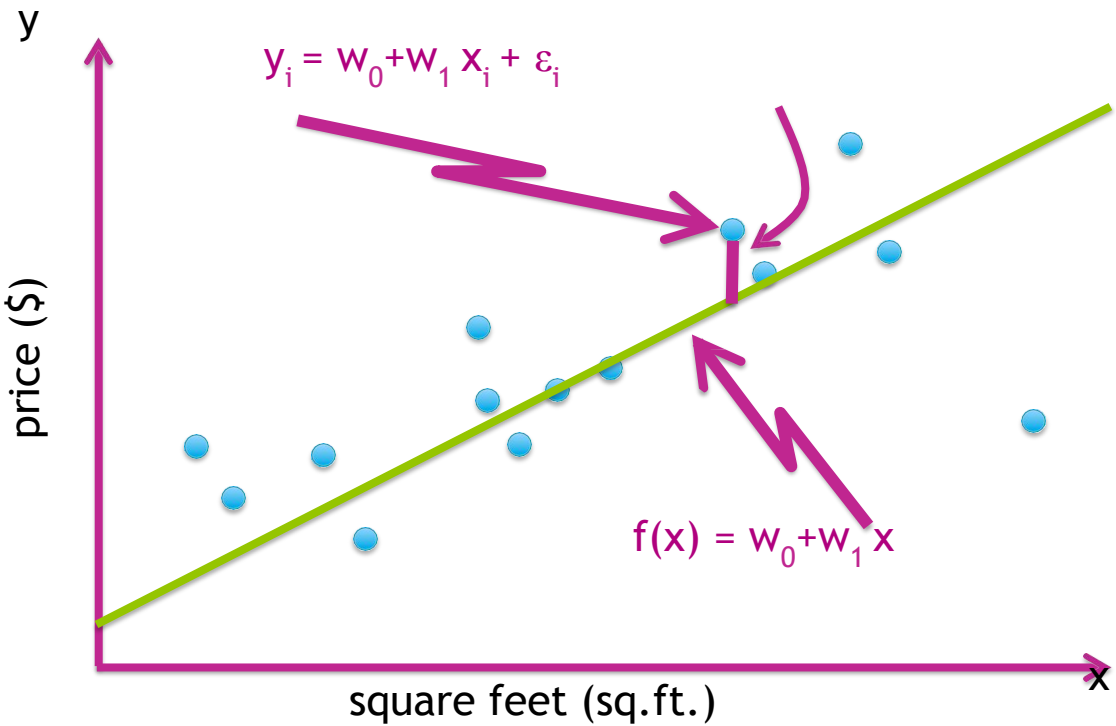


Assume model f(x) is a quadratic function

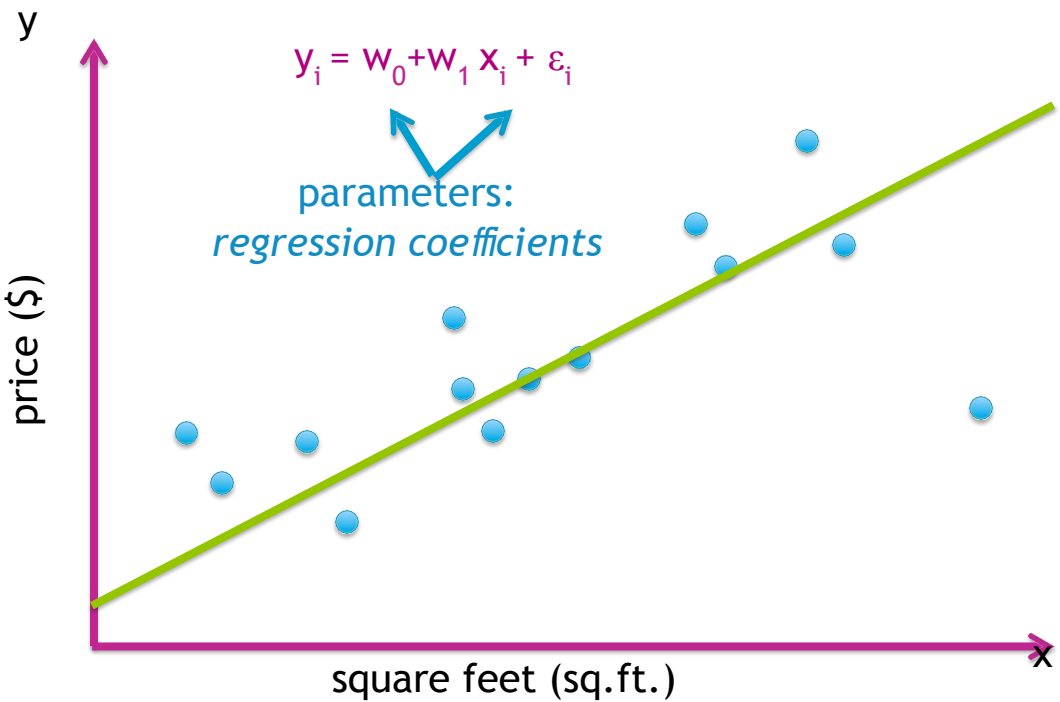
Process


house id
house attributes
house sales $
table
Training Data
Feature extraction
sq. ft.
x
regression
ML model
ŷ
predicted house sales $
y actual sales $
f̂ estimated function fit from data
ML algorithm
error in our predicted values
Quality metric

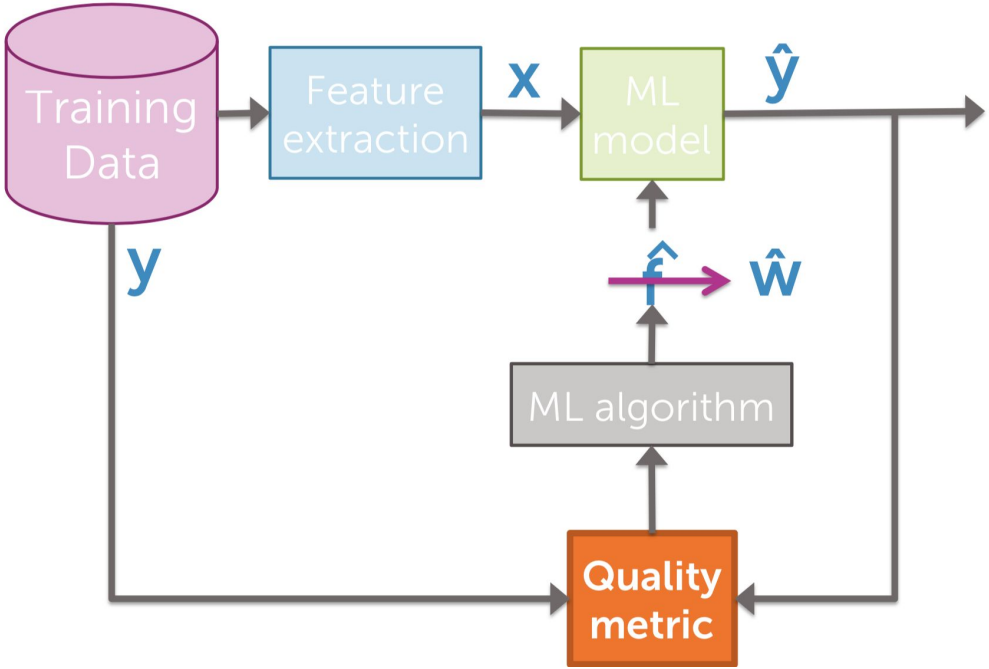
Simple Linear Regression



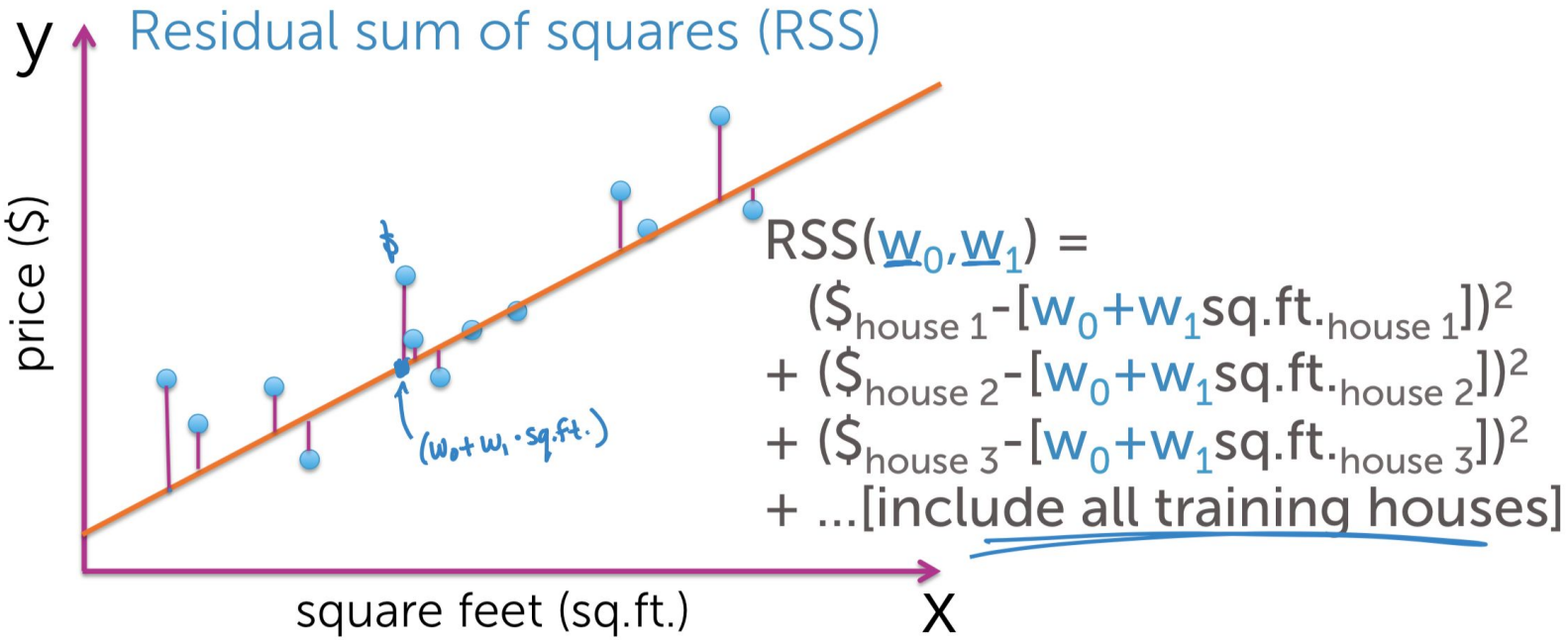
Simple Linear Regression



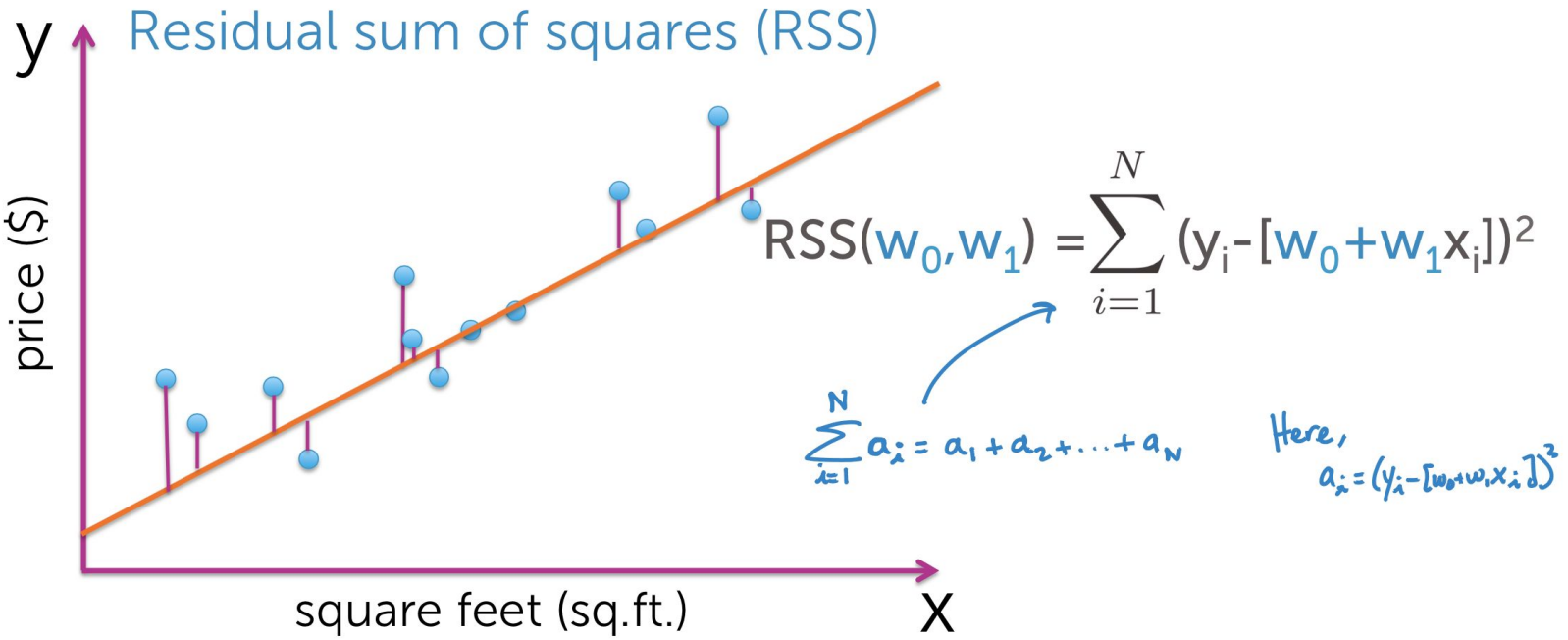
Fitting a line to data



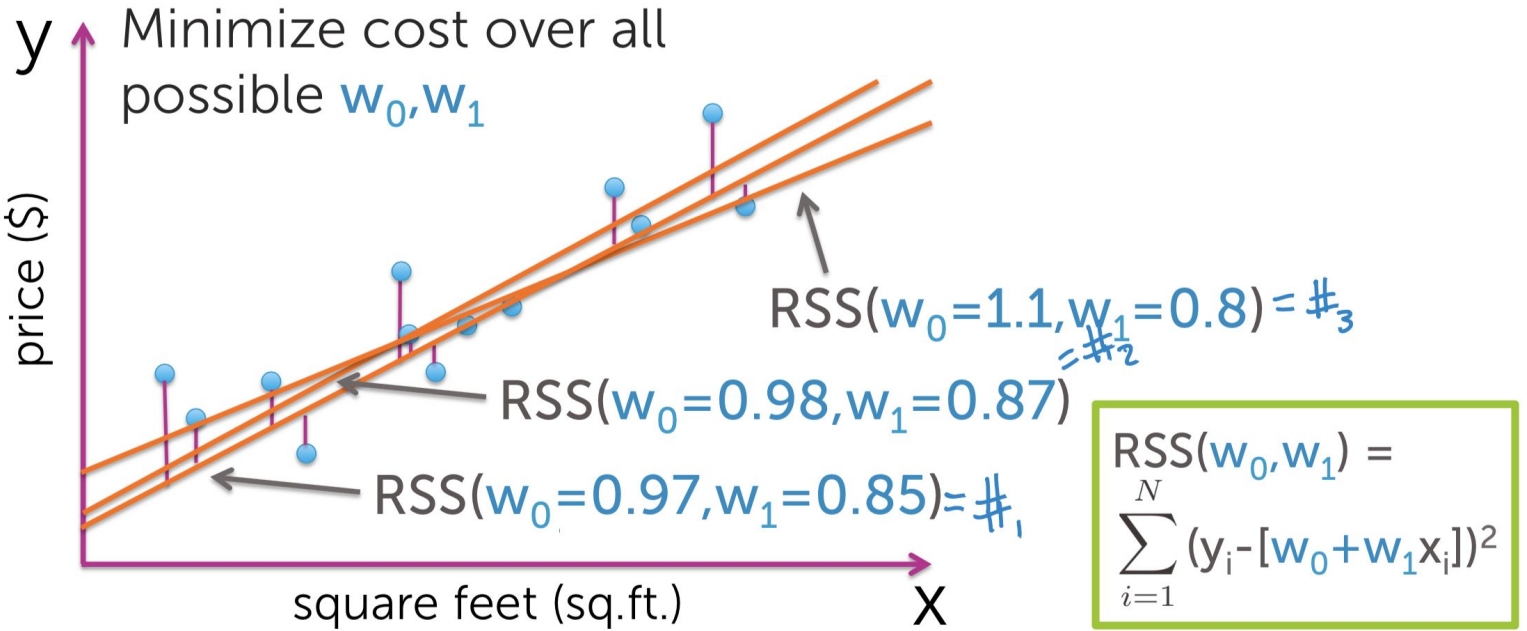
"Cost" of using a given line



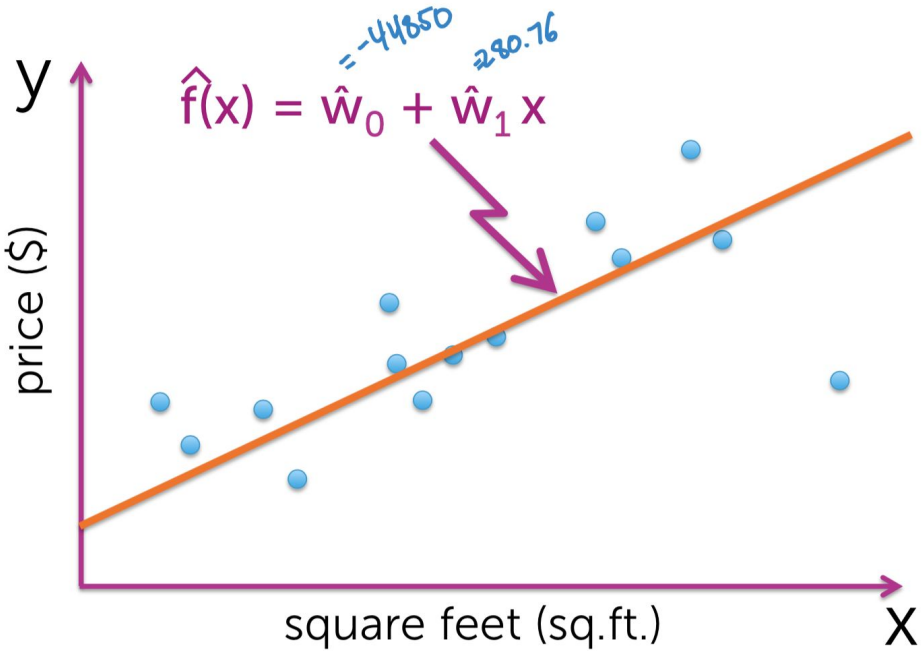
"Cost" of using a given line



Find "best" line



Model vs. fitted line



Regression model:

$y_i = w_0 + w_1 x_i + \varepsilon_i$

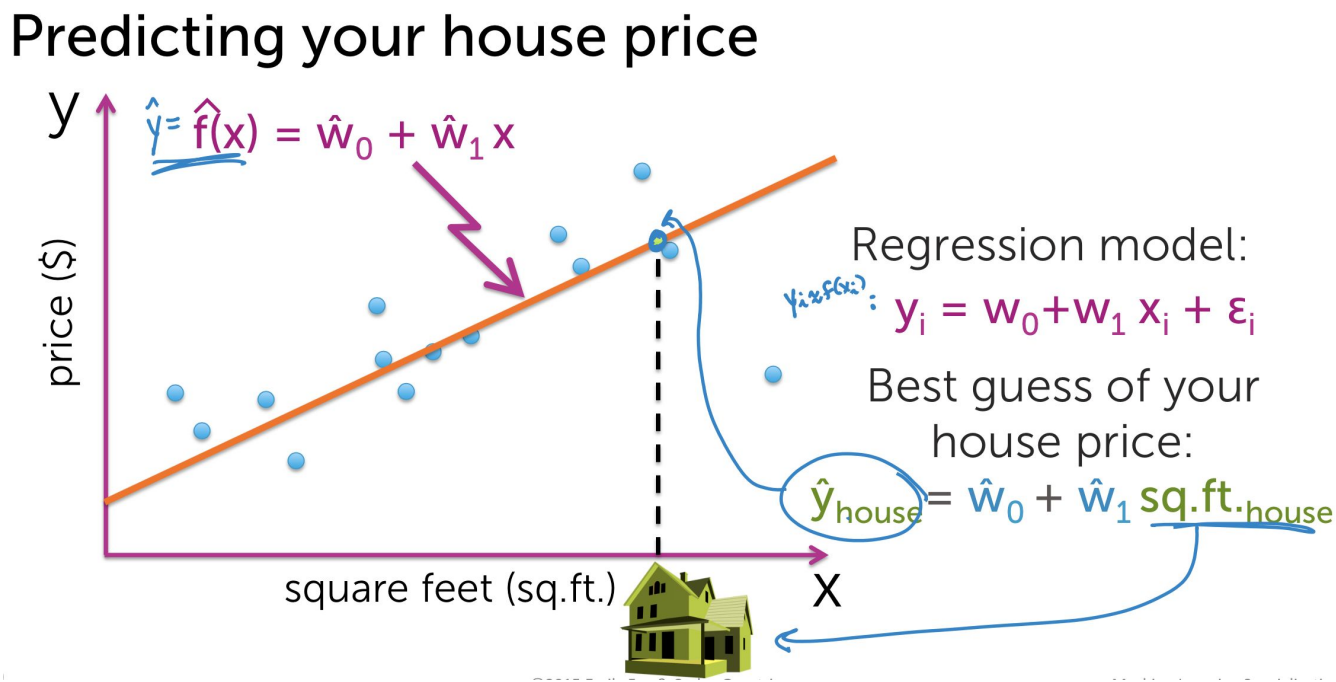
parameters (unknown variables)

Estimated parameters:

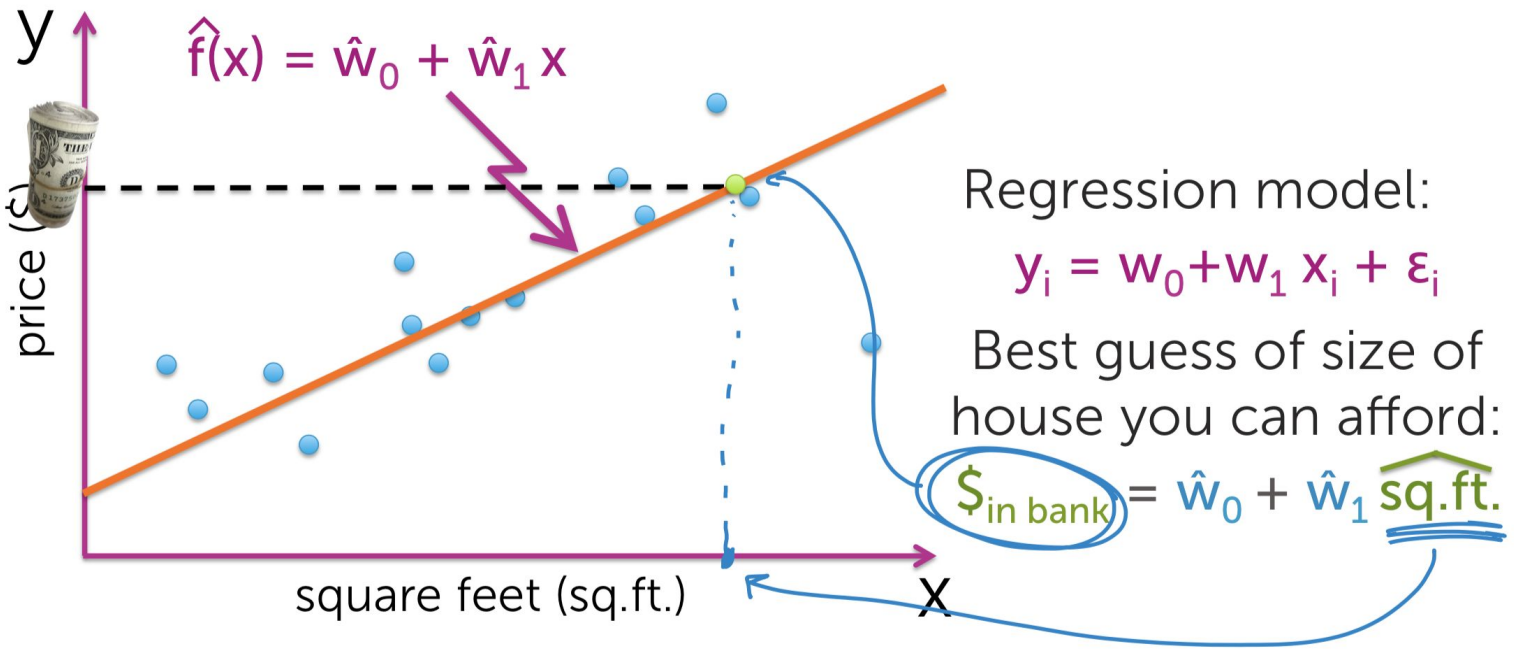
$\hat{w}_0 = -44850$, $\hat{w}_1 = 280.76$

take actual values

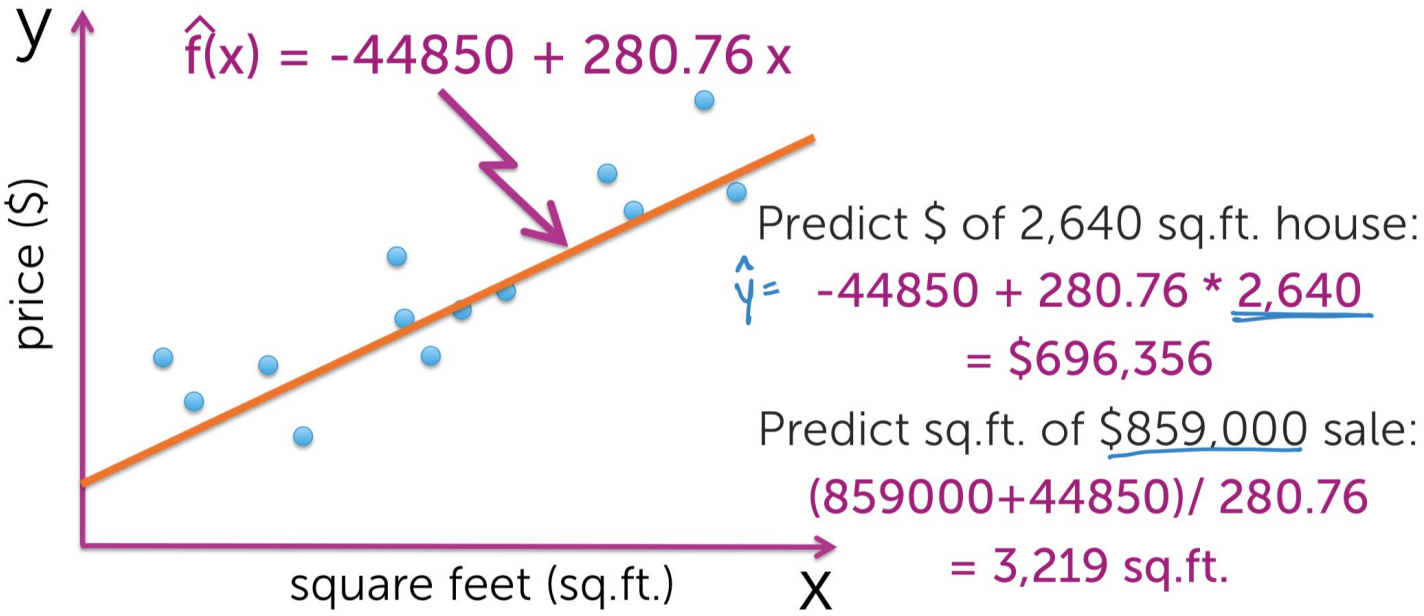
Seller: Predicting your house price



Buyer: Predicting size of house



A concrete example



Predict $ of 2,640 sq.ft. house:

$\hat{y}$ = -44850 + 280.76 * 2,640

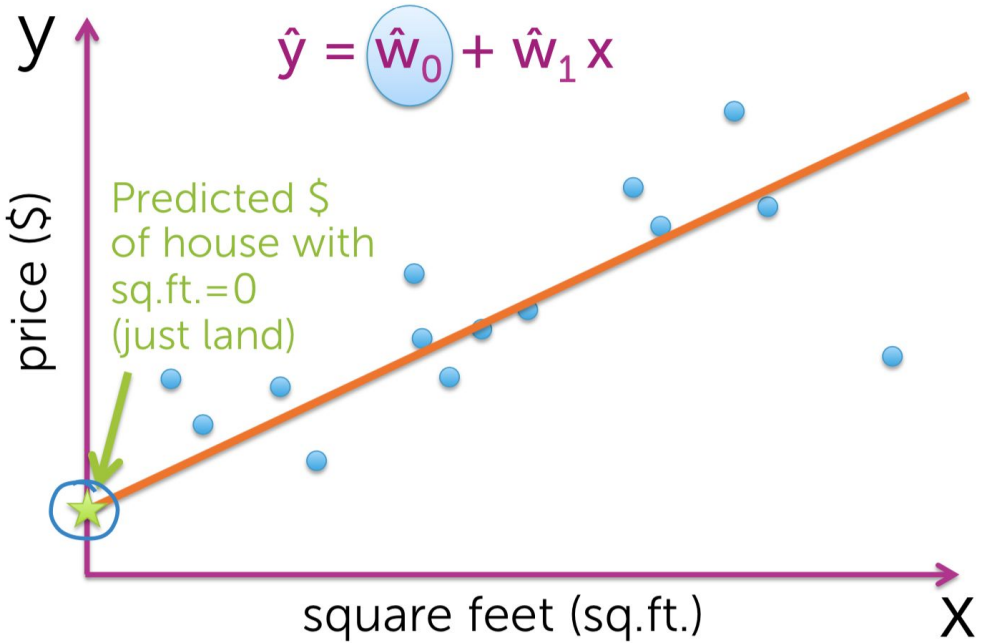
= $696,356

Predict sq.ft. of $859,000 sale:

(859000+44850)/ 280.76

= 3,219 sq.ft.

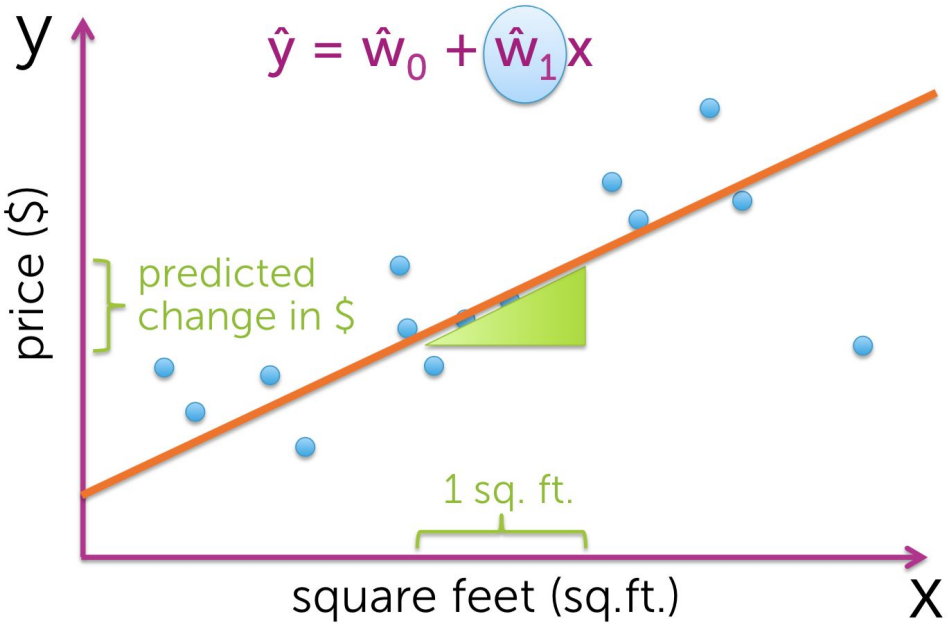
Interpreting the coefficients



$\hat{y} = \hat{w}_0$ when $x = 0$

not very meaningful

Interpreting the coefficients



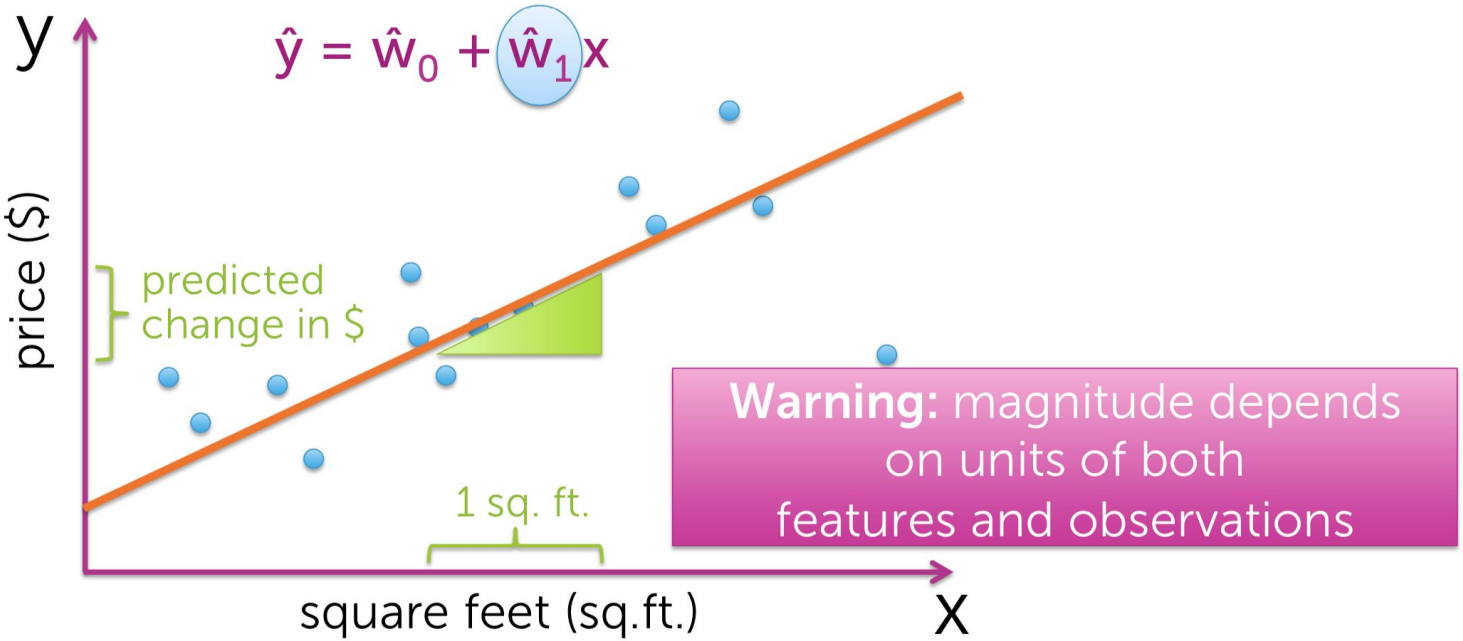
$\hat{\$}_{1001\ sq.ft.} - \hat{\$}_{1000\ sq.ft.}$

$= \hat{w}_0 + \hat{w}_1 \cdot 1001\ sq.ft. - (\hat{w}_0 + \hat{w}_1 1000\ sq.ft.)$

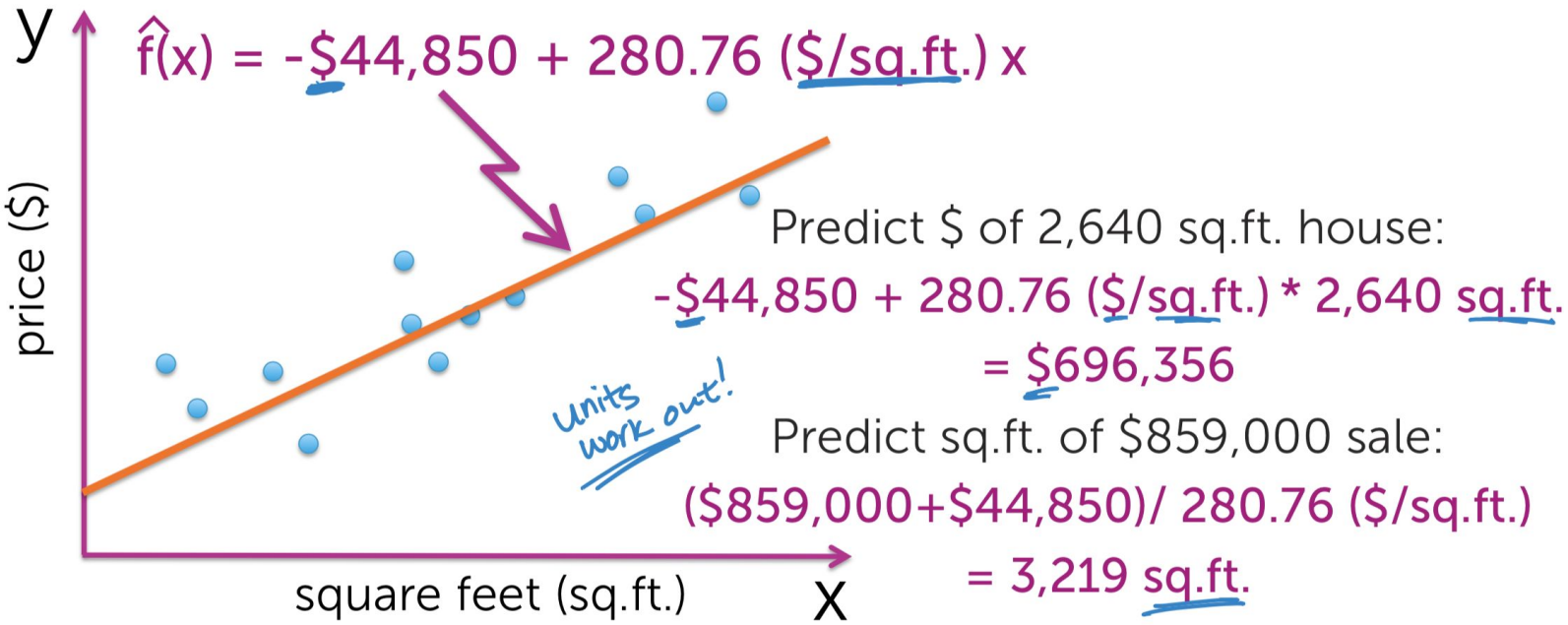
$= \hat{w}_1$

predicted change in the output per unit change in input

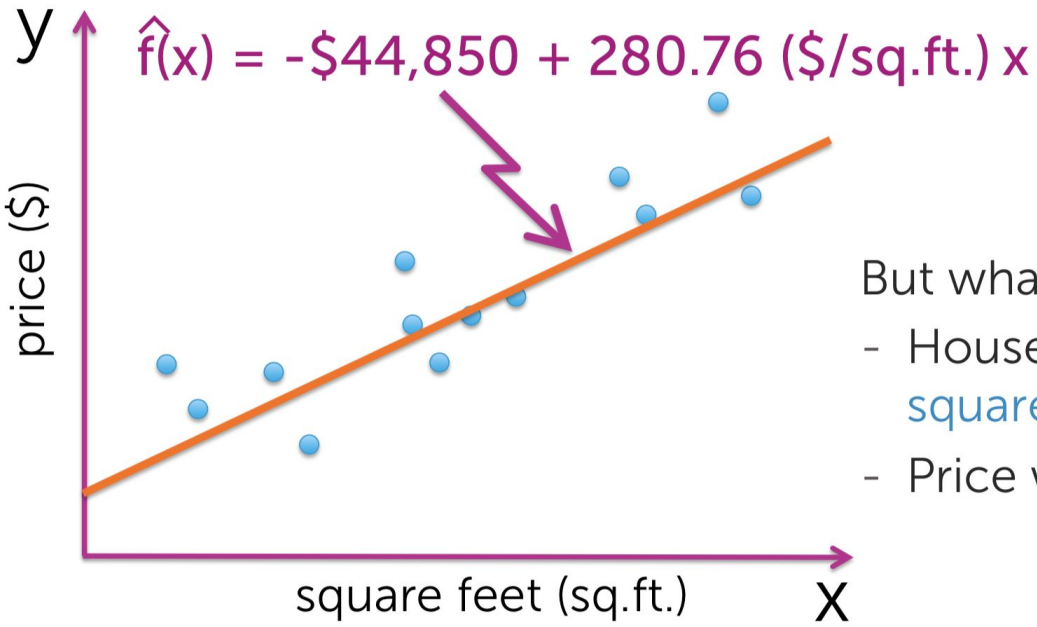
Interpreting the coefficients



A concrete example



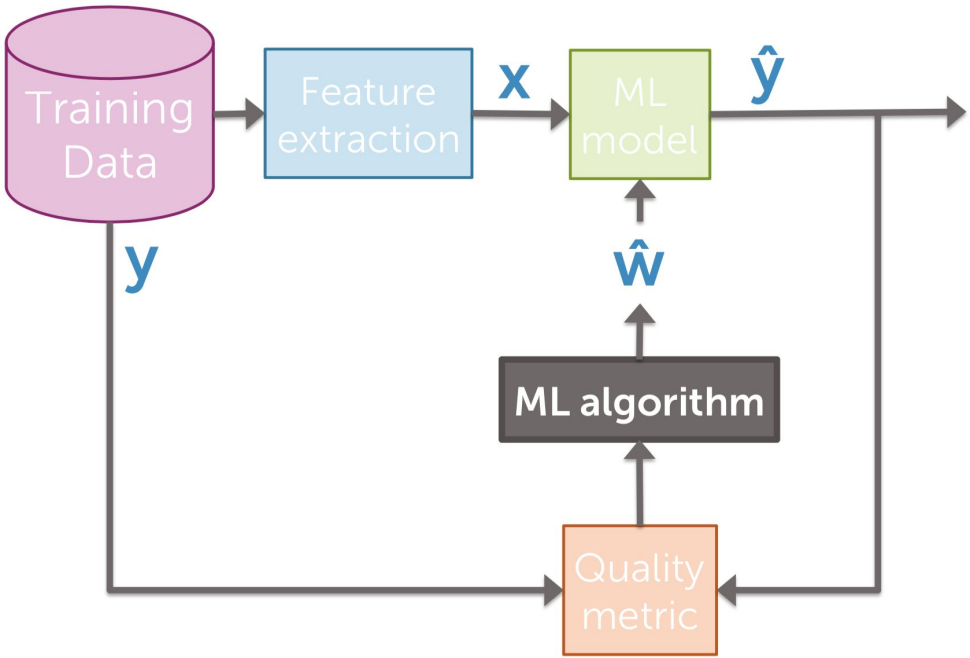
A concrete example



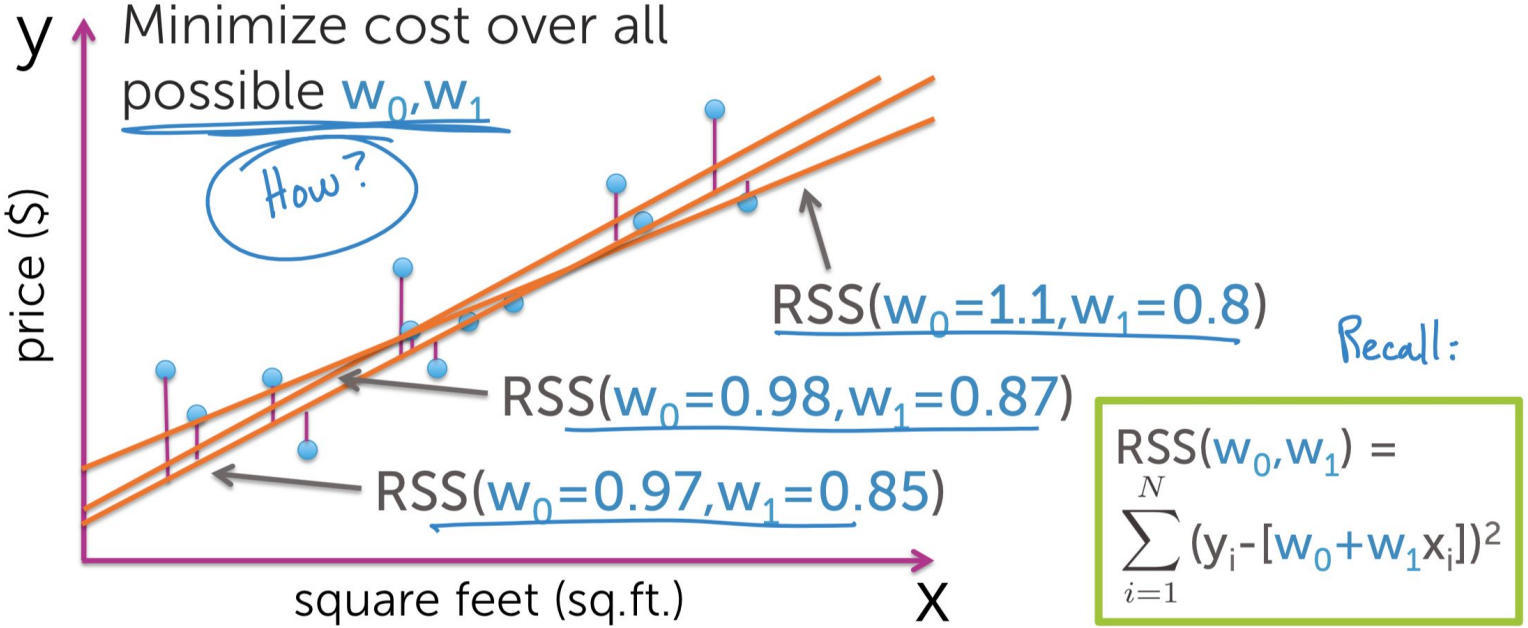
But what if:

- House was measured in square meters?
- Price was measured in RMB?

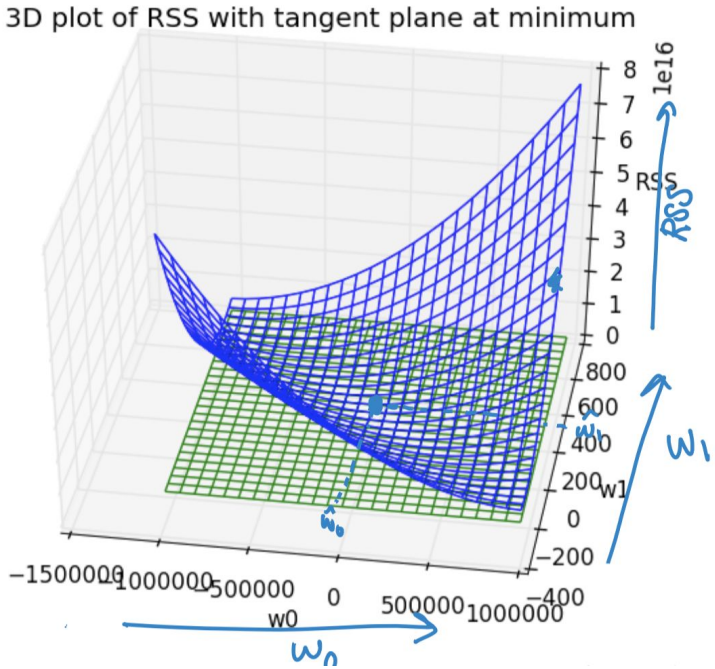
Algorithms for fitting the model



Find "best" line



Minimizing the cost

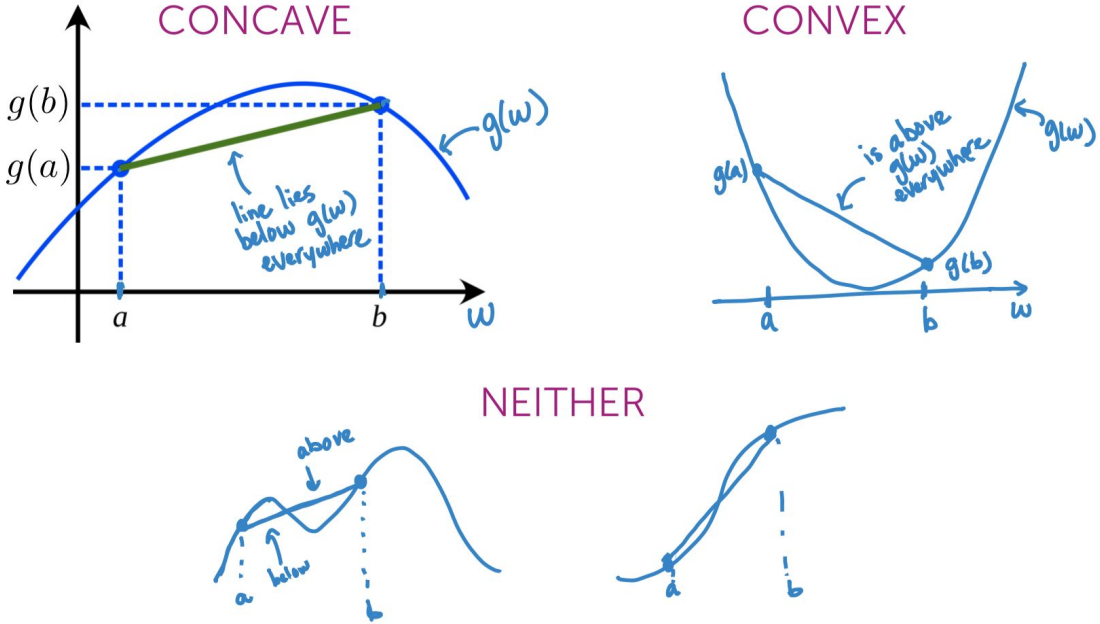


Minimize function over all possible w_0, w_1

$$\min_{w_0,w_1} \sum_{i=1}^{N} (y_i - [w_0 + w_1 x_i])^2$$

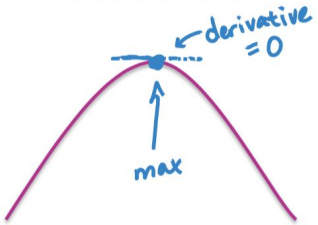
RSS(w_0,w_1) is a function of 2 variables = $g(w_0, w_1)$

Convex/concave functions



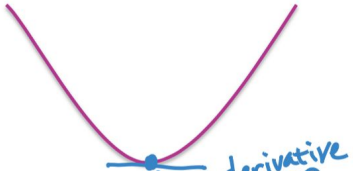
Finding the max or min analytically

CONCAVE



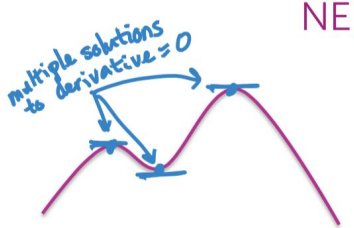
$\max_w g(w)$

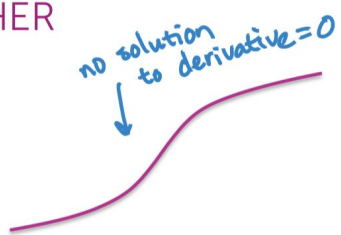
CONVEX



$\min_w g(w)$

NEITHER





Example:

$g(w) = 5-(w-10)^2$

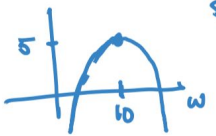
$\frac{dg(w)}{dw} = 0 - 2(w-10)^1 \cdot 1$

$= -2w + 20$

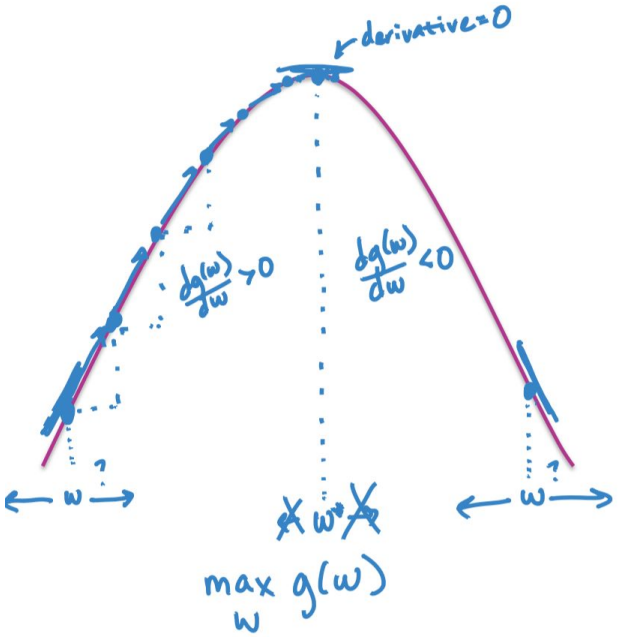
Set derivate = 0:

$-2w + 20 = 0$

$w = 10$



Finding the max via hill climbing



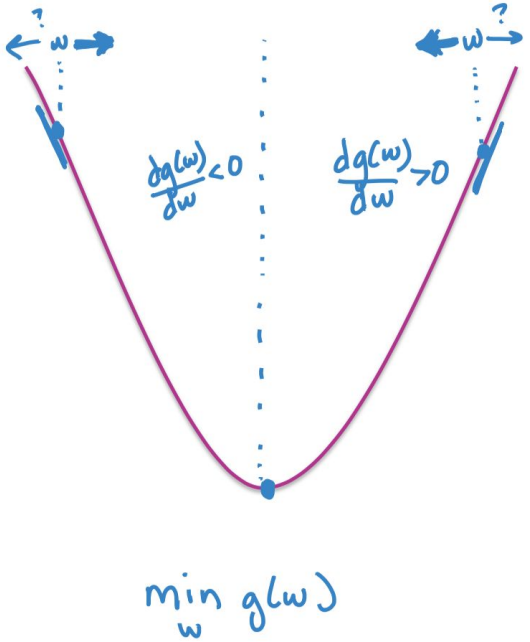
How do we know whether to move w to right or left? (Inc. or dec. the value of w?)

while not converged

$$w^{(t+1)} \leftarrow w^{(t)} + \eta \frac{dg(w)}{dw}$$

$w^{(t)}$: iteration t; η: stepsize

Finding the min via hill descent



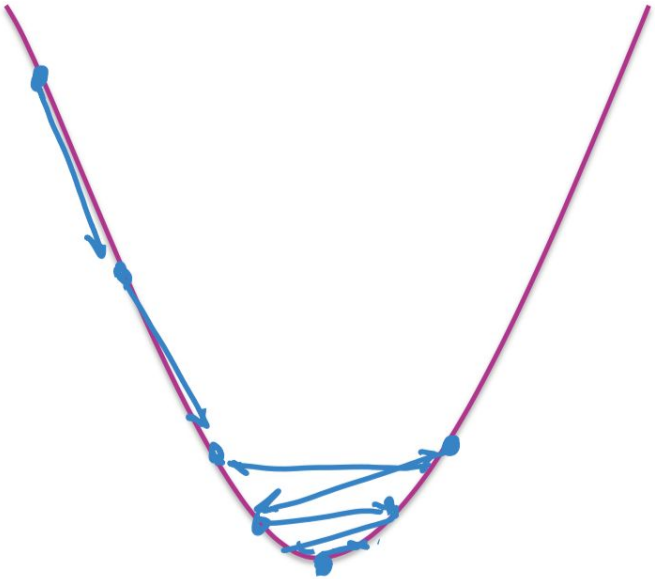
When derivative is positive, we want to decrease w and when derivative is negative, we want to increase w

Algorithm:

while not converged

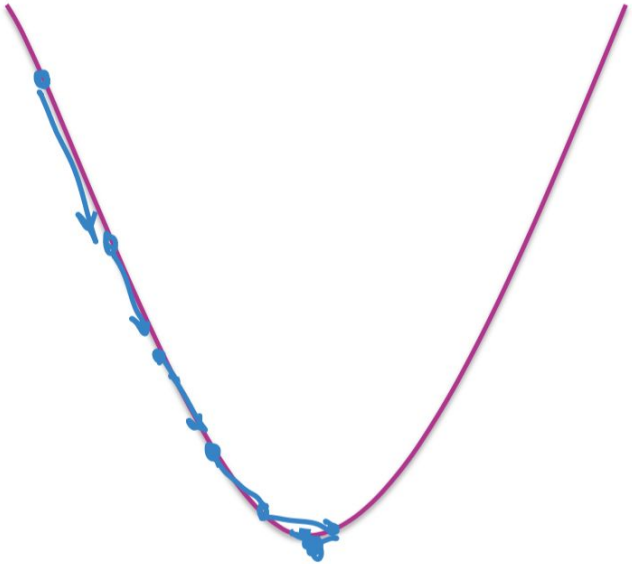
$$w^{(t+1)} \leftarrow w^{(t)} - \eta \left.\frac{dg}{dw}\right|_{w^{(t)}}$$

Choosing the stepsize— Fixed stepsize



$\eta = 0.1$

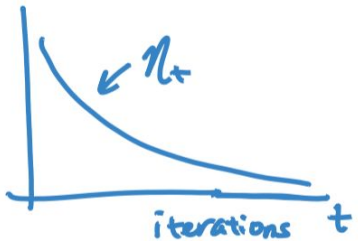
Choosing the stepsize— Decreasing stepsize



Common choices:

$$\eta_t = \frac{\alpha}{t}$$

$$\eta_t = \frac{\alpha}{\sqrt{t}}$$



Convergence criteria

For convex functions, optimum occurs when

$$\frac{dg(w)}{dw} = 0$$

In practice, stop when

$$\left|\frac{dg(w)}{dw}\right| < \epsilon$$

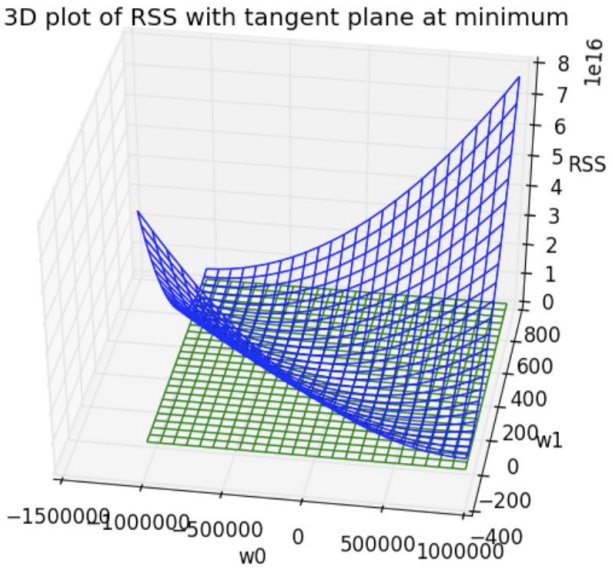
↑ threshold to be set

Algorithm:

while not converged

$$w^{(t+1)} \leftarrow w^{(t)} - \eta \left.\frac{dg}{dw}\right|_{w^{(t)}}$$

Moving to multiple dimensions: Gradients



$$\nabla g(w) = \begin{bmatrix} \frac{\partial g}{\partial w_0} \\ \frac{\partial g}{\partial w_1} \\ \vdots \\ \frac{\partial g}{\partial w_p} \end{bmatrix}$$

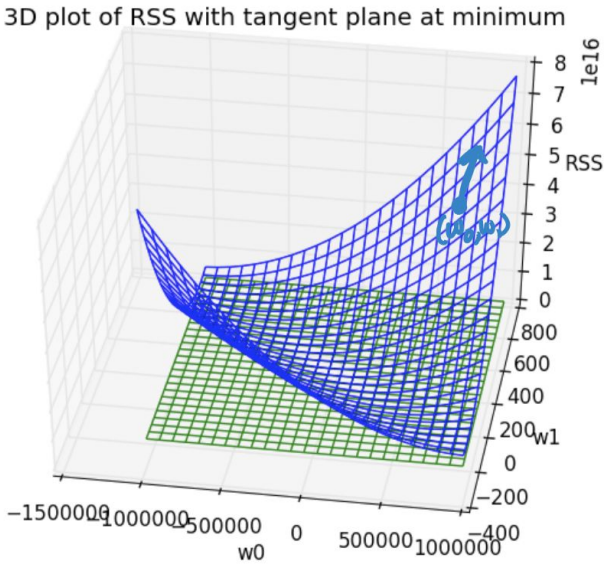
gradient

$[w_0, w_1, \ldots, w_p]$

$(p+1)$-dimensional vector

partial derivative is like a derivate with respect to w_1 treating all other variables as constants

Gradient example



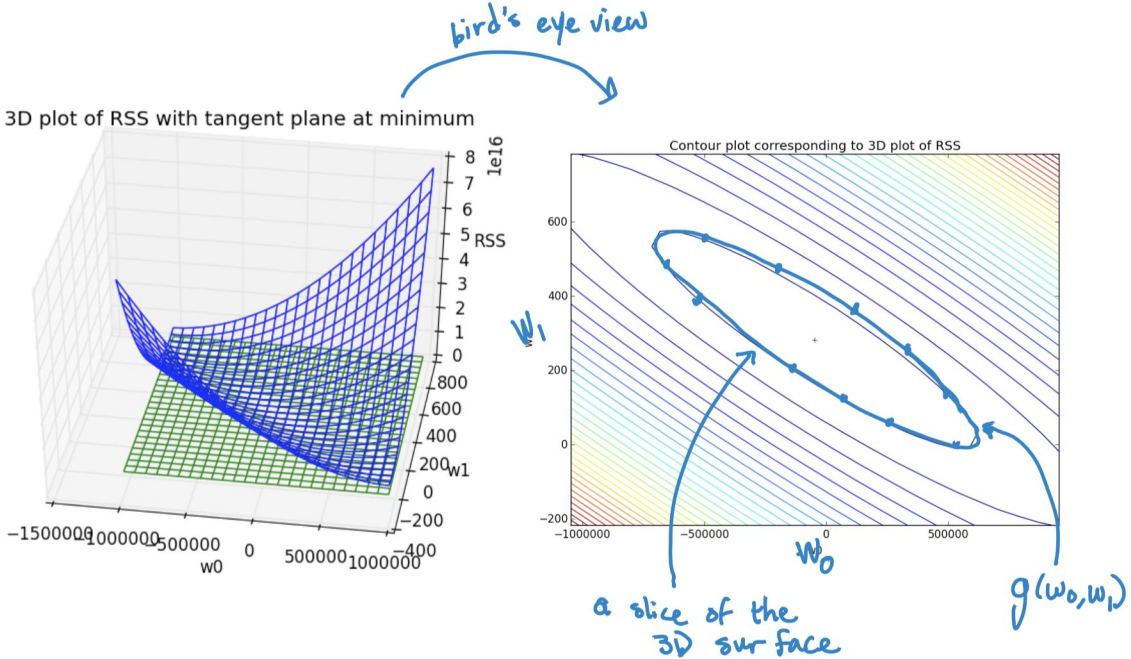
$$g(w) = 5w_0 + 10w_0w_1 + 2w_1^2$$

$$\frac{\partial g}{\partial w_0} = 5 + 10w_1$$

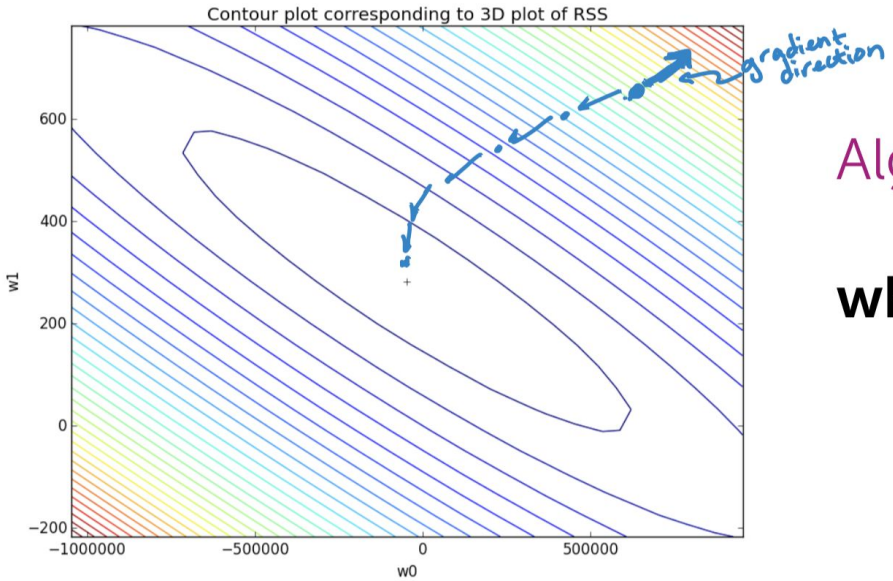
$$\frac{\partial g}{\partial w_1} = 10w_0 + 4w_1$$

$$\nabla g(w) = \begin{bmatrix} 5 + 10w_1 \\ 10w_0 + 4w_1 \end{bmatrix}$$

Contour plots



Gradient descent



Algorithm:

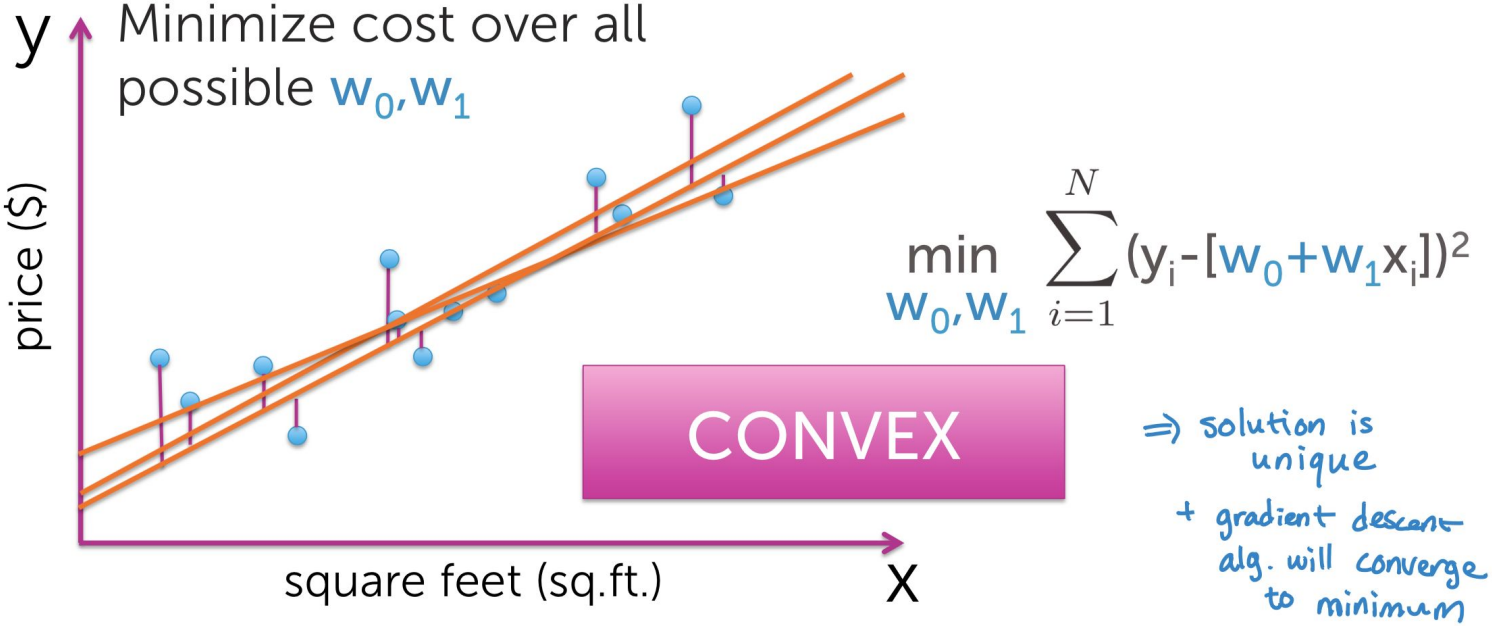
while not converged

$$w^{(t+1)} \leftarrow w^{(t)} - \eta \nabla g(w^{(t)})$$

$$\begin{bmatrix} \vdots \end{bmatrix} \leftarrow \begin{bmatrix} \vdots \end{bmatrix} - \eta \begin{bmatrix} \vdots \end{bmatrix}$$

Convergence: $\|\nabla g(w)\| < \epsilon$

Find "best" line



Compute the gradient

$$\text{RSS}(w_0, w_1) = \sum_{i=1}^{N} (y_i - [w_0 + w_1 x_i])^2$$

Aside:

$$\frac{d}{dw} \sum_{i=1}^{N} g_i(w) = \frac{d}{dw} (g_1(w) + g_2(w) + \ldots g_N(w))$$
$$= \frac{d}{dw} g_1(w) + \frac{d}{dw} g_2(w) + \ldots \frac{d}{dw} g_N(w)$$
$$= \sum_{i=1}^{N} \frac{d}{dw} g_i(w)$$

In our case

$$g_i(w) = (y_i - [w_0 + w_1 x_i])^2$$

$$\frac{\partial \text{RSS}(w)}{\partial w_0} = \sum_{i=1}^{N} \frac{\partial}{\partial w_0} (y_i - [w_0 + w_1 x_i])^2$$

same for w_1

Compute the gradient

$$\text{RSS}(w_0, w_1) = \sum_{i=1}^{N} (y_i - [w_0 + w_1 x_i])^2$$

Taking the derivative w.r.t. w_0

$$\sum_{i=1}^{N} 2\,(y_i - [w_0 + w_1 x_i])^1 \cdot (-1)$$

$$= -2 \sum_{i=1}^{N} (y_i - [w_0 + w_1 x_i])$$

Compute the gradient

$$RSS(w_0, w_1) = \sum_{i=1}^{N} (y_i - [w_0 + w_1 x_i])^2$$

Taking the derivative w.r.t. w_1

$$\sum_{i=1}^{N} 2(y_i - [w_0 + w_1 x_i])^1 \cdot (-x_i)$$

$$= -2 \sum_{i=1}^{N} (y_i - [w_0 + w_1 x_i]) x_i$$

Compute the gradient

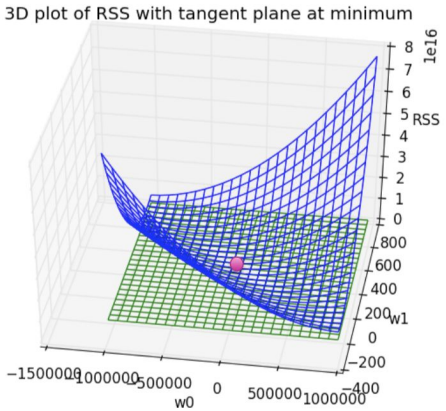
$$RSS(w_0,w_1) = \sum_{i=1}^{N} (y_i-[w_0+w_1x_i])^2$$

Putting it together:

$$\nabla RSS(w_0,w_1) = \begin{bmatrix} -2\sum_{i=1}^{N}[y_i - (w_0+w_1x_i)] \\ -2\sum_{i=1}^{N}[y_i - (w_0+w_1x_i)]x_i \end{bmatrix}$$

Approach 1: Set gradient = 0

$$\nabla RSS(w_0, w_1) = \begin{bmatrix} -2\sum_{i=1}^{N}[y_i - (w_0 + w_1 x_i)] \\ -2\sum_{i=1}^{N}[y_i - (w_0 + w_1 x_i)]x_i \end{bmatrix}$$



top term:

$$\hat{w}_0 = \frac{\sum_{i=1}^{N} y_i}{N} - \hat{w}_1 \frac{\sum_{i=1}^{N} x_i}{N}$$

average house sales price — estimate of the slope — average sq-ft.

bottom term:

$$\sum y_i x_i - \hat{w}_0 \sum x_i - \hat{w}_1 \sum x_i^2 = 0$$

plug in

$$\hat{w}_1 = \frac{\sum y_i x_i - \frac{\sum y_i \sum x_i}{N}}{\sum x_i^2 - \frac{\sum x_i \sum x_i}{N}}$$

Note:

$$\sum_{i=1}^{N} y_i \quad \sum_{i=1}^{N} x_i \quad \sum_{i=1}^{N} y_i x_i \quad \sum_{i=1}^{N} x_i^2$$

Approach 2: Gradient descent

Interpreting the gradient:

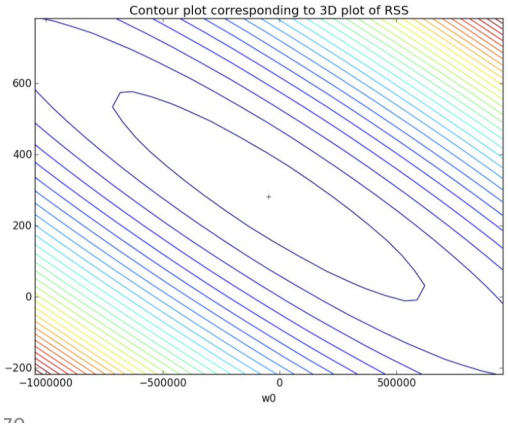
actual house sales observation

predicted value $\hat{y}_i(w_0, w_1)$

$$\nabla RSS(w_0, w_1) = \begin{bmatrix} -2\sum_{i=1}^{N}[y_i - (w_0 + w_1 x_i)] \\ -2\sum_{i=1}^{N}[y_i - (w_0 + w_1 x_i)]x_i \end{bmatrix} = \begin{bmatrix} -2\sum_{i=1}^{N}[y_i - \hat{y}_i(w_0, w_1)] \\ -2\sum_{i=1}^{N}[y_i - \hat{y}_i(w_0, w_1)]x_i \end{bmatrix}$$

Approach 2: Gradient descent

$$\nabla RSS(w_0, w_1) = \begin{bmatrix} -2\sum_{i=1}^{N}[y_i - \hat{y}_i(w_0, w_1)] \\ -2\sum_{i=1}^{N}[y_i - \hat{y}_i(w_0, w_1)]x_i \end{bmatrix}$$



while not converged $(-2)\cdot(-\eta)$

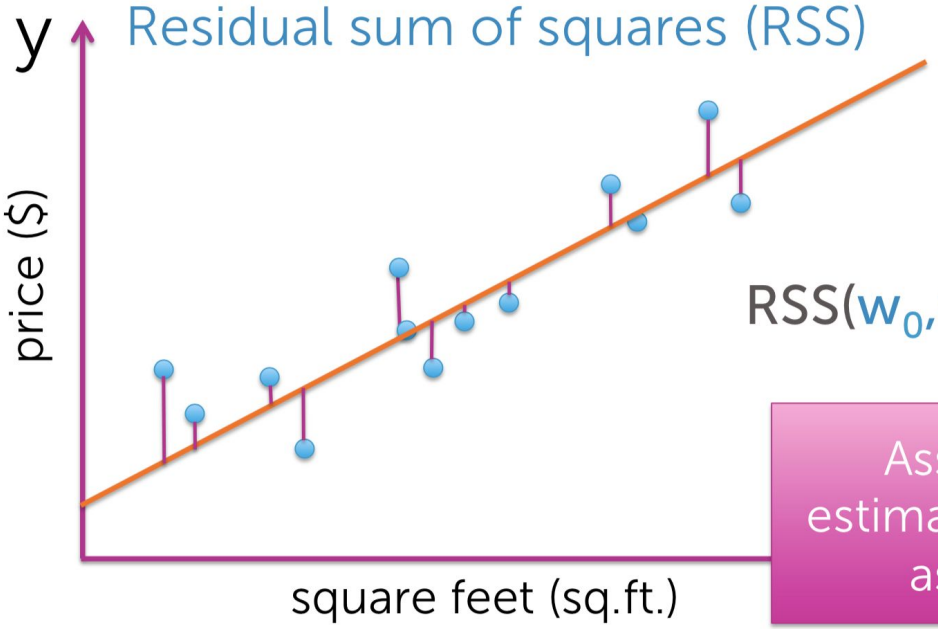
$$\begin{bmatrix} w_0^{(t+1)} \\ w_1^{(t+1)} \end{bmatrix} \leftarrow \begin{bmatrix} w_0^{(t)} \\ w_1^{(t)} \end{bmatrix} + 2\eta \begin{bmatrix} \sum_{i=1}^{N}[y_i - \hat{y}_i(w_0^{(t)}, w_1^{(t)})] \\ \sum_{i=1}^{N}[y_i - \hat{y}_i(w_0^{(t)}, w_1^{(t)})]x_i \end{bmatrix}$$

If overall, under predicting $\hat{y}_i$, then $\sum[y_i - \hat{y}_i]$ is positive
→ w_0 is going to increase
similar intuition for w_1, but multiply by x_i

Comparing the approaches

- For most ML problems, cannot solve gradient = 0
- Even if solving gradient = 0 is feasible, gradient descent can be more efficient
- Gradient descent relies on choosing stepsize and convergence criteria

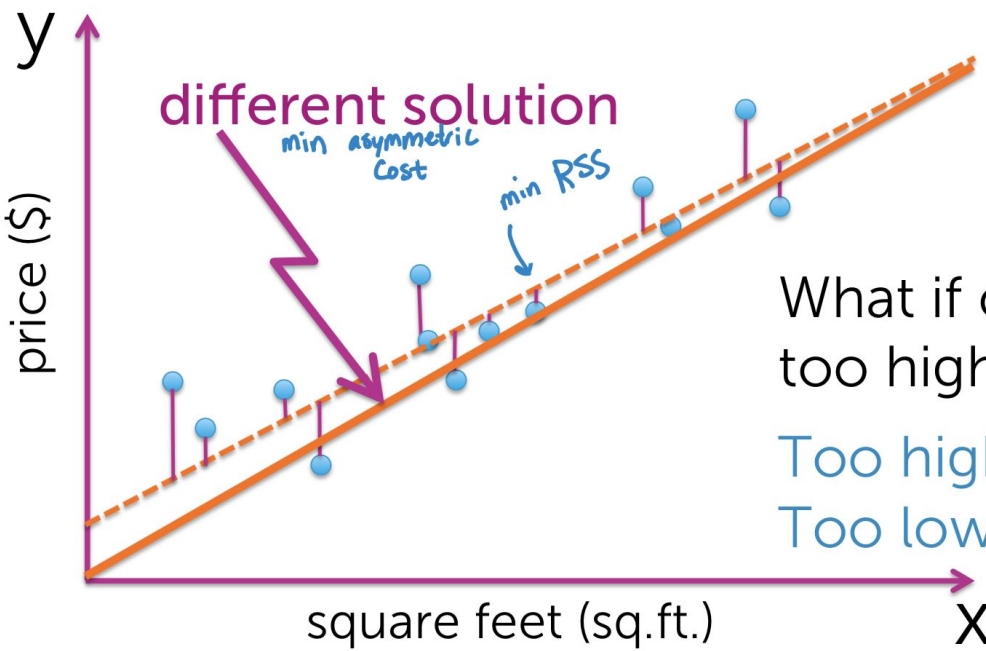
Symmetric cost functions



$$RSS(w_0,w_1) = \sum_{i=1}^{N} (y_i - [w_0 + w_1 x_i])^2$$

Assumes cost of over-estimating sales price is same as under-estimating

Asymmetric cost functions



What if cost of listing house too high has **bigger cost**?

Too high → no offers ($=0)

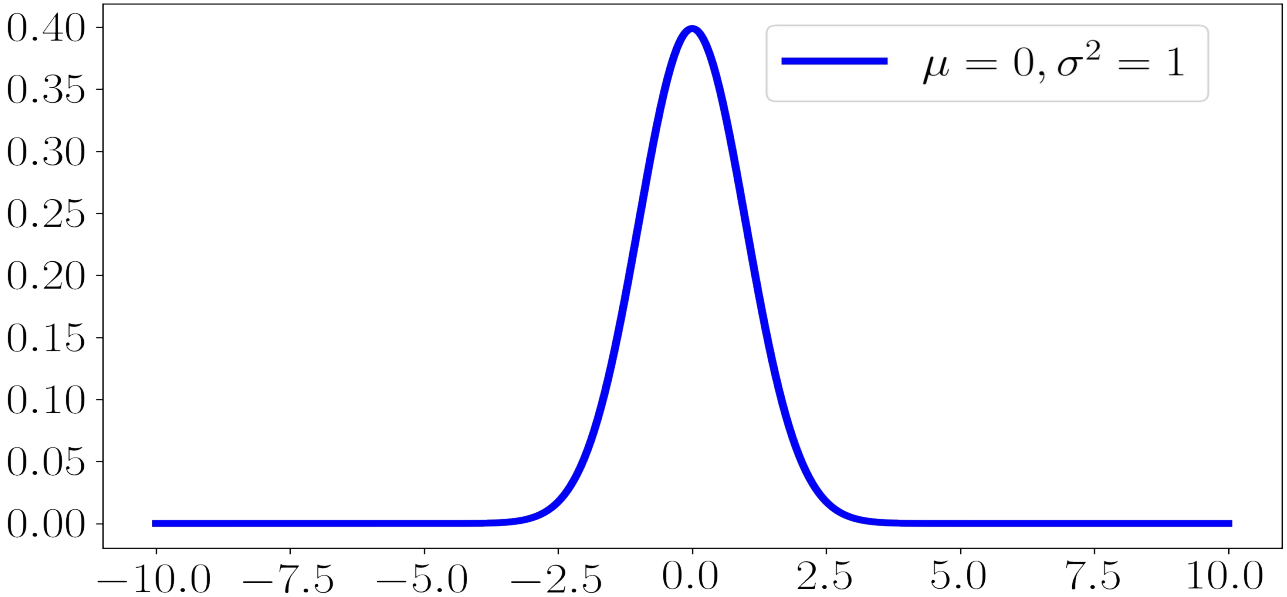
Too low → offers for lower $

What you can do now…

- Describe the input (features) and output (real-valued predictions) of a regression model
- Calculate a goodness-of-fit metric (e.g., RSS)
- Estimate model parameters to minimize RSS using gradient descent
- Interpret estimated model parameters
- Exploit the estimated model to form predictions
- Discuss the possible influence of high leverage points
- Describe intuitively how fitted line might change when assuming different goodness-of-fit metrics

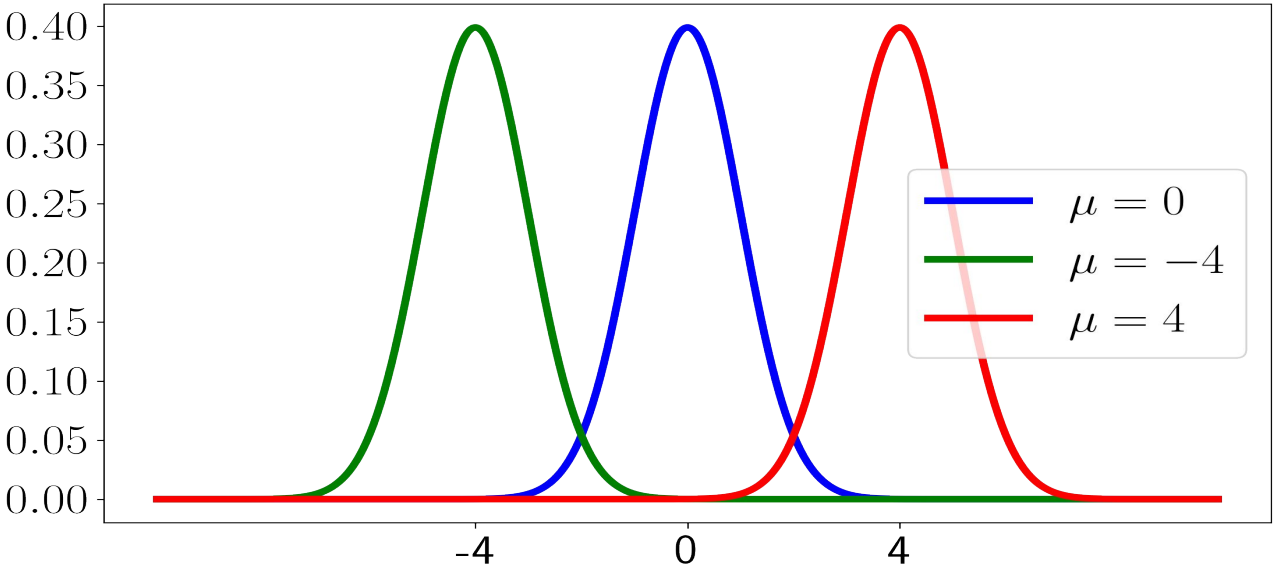
Linear Regression: From statistic

Univariate normal



$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Univariate normal: mean

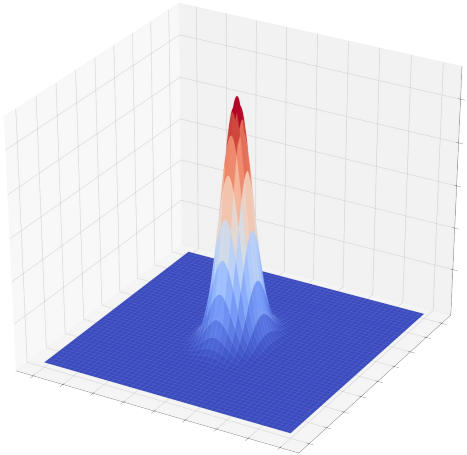


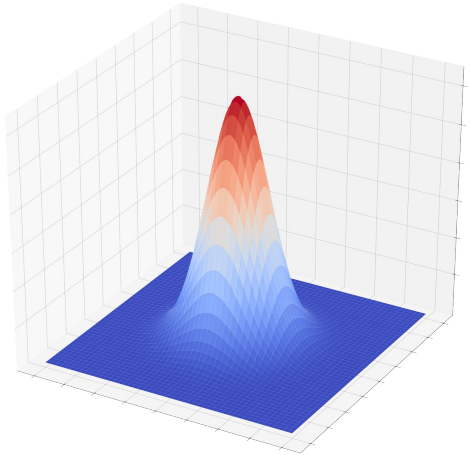
$$\mathbb{E}X = \mu$$

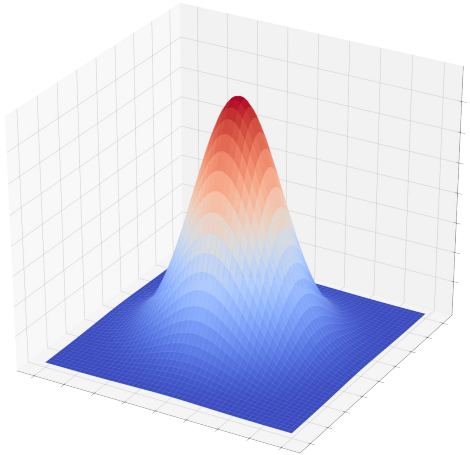
Multivariate normal

$$\mathcal{N}(x|\mu,\Sigma) = \frac{1}{\sqrt{|2\pi\Sigma|}}\exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right]$$

$$\mathbb{E}X - \mu \quad \mathrm{Cov}|X| - \Sigma$$

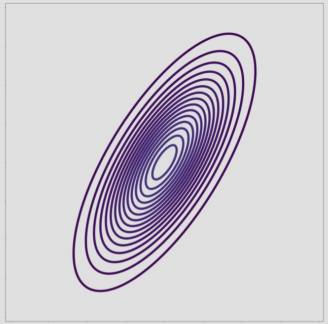






Multivariate normal

$\Sigma = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$

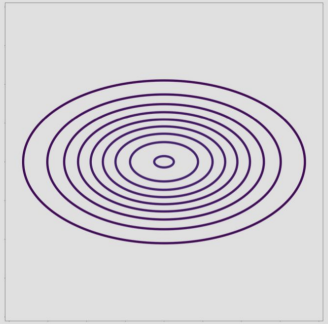


Full

Full

Parameters: $\frac{D(D+1)}{2}$

$\Sigma = \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}$

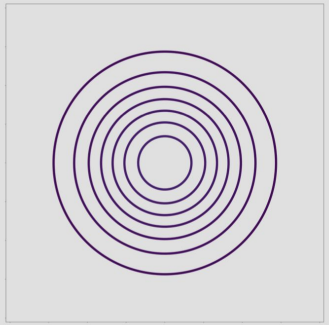


Diagonal

Diagonal

Parameters: D

$\Sigma = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$

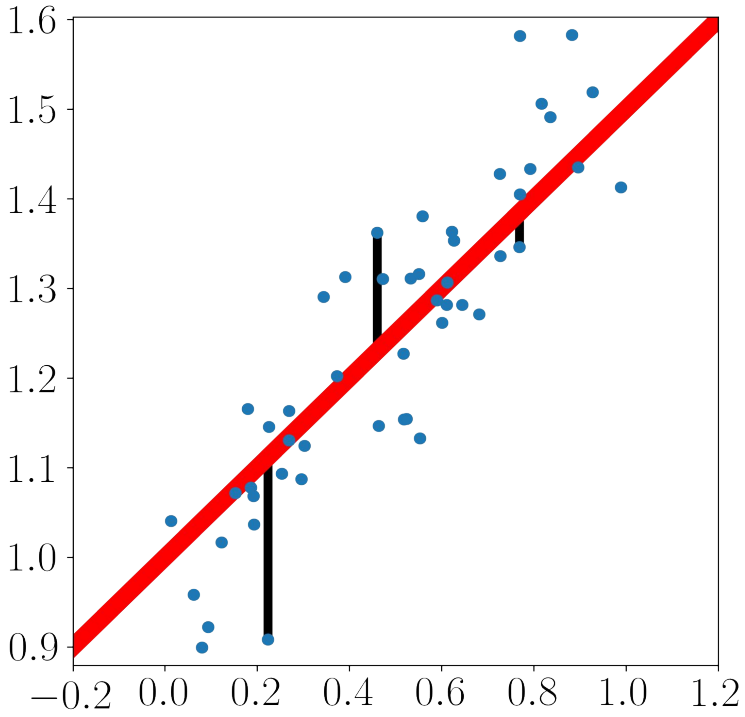


Spherical

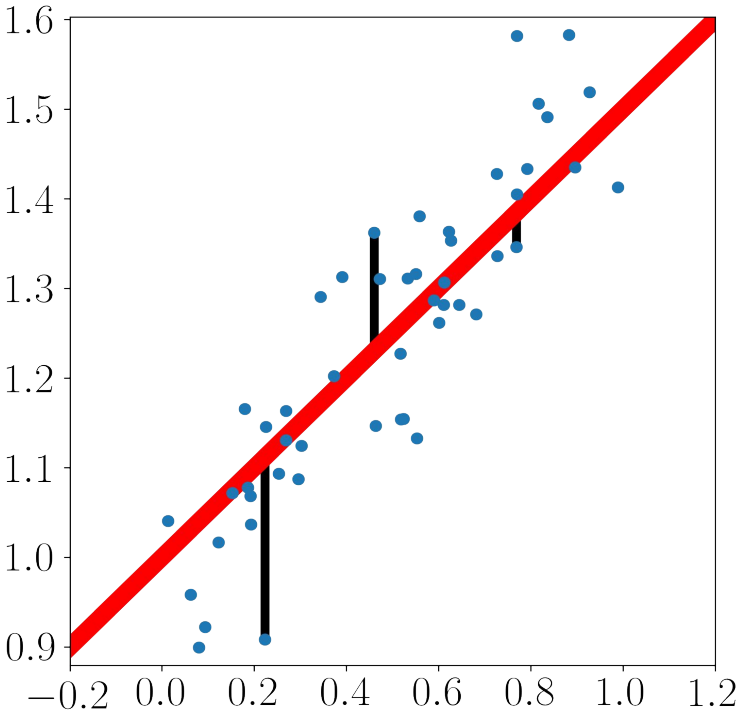
Spherical

Parameters: 1

Linear regression

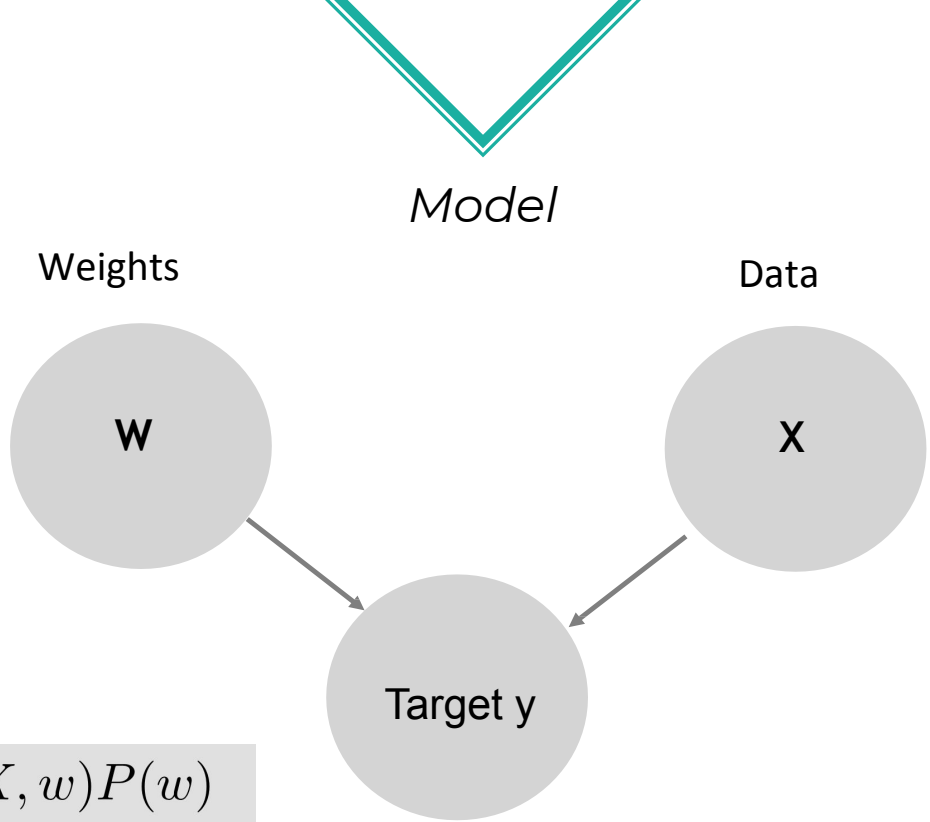


Least squares problem



$$L(w) = \sum_{i=1}^{N} (w^T x_i - y_i)^2 = \left\| w^T X - y \right\|^2 \to \min_w$$

$$\widehat{w} = arg \min_w L(w)$$



$$P(w, y|X) = P(y|X, w)P(w)$$
$$P(y|w, X) = \mathcal{N}(y|w^T X, \sigma^2 I)$$
$$P(w) = \mathcal{N}(w|0, \gamma^2 I)$$

Least squares problem

$$P(w, y|X) = P(y|X, w)P(w)$$
$$P(y|w, X) = \mathcal{N}(y|w^T X, \sigma^2 I)$$
$$P(w) = \mathcal{N}(w|0, \gamma^2 I)$$

$$P(w|y, X) = ?$$

Least squares problem

$$P(w, y|X) = P(y|X, w)P(w)$$
$$P(y|w, X) = \mathcal{N}(y|w^T X, \sigma^2 I)$$
$$P(w) = \mathcal{N}(w|0, \gamma^2 I)$$

$$P(w|y, X) = ?$$

$$P(w|y, X) = \frac{P(w, y|X)}{P(y|X)} \propto P(w, y|X)$$

$$P(w, y|X) \to \max_w \Leftrightarrow \log P(w, y|X) \to \max_w$$

$$\log P(w, y|X) \to \max_w$$

$$-\frac{1}{2\sigma^2}||w^T X - y||^2 - \frac{1}{2\gamma^2}||w||^2 \to \max_w$$

$$||w^T X - y||^2 + C||w||^2 \to \min_w$$

Maximum Likelihood

$$\theta = \max_{\theta} p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta)$$

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta) \approx \prod_{n=1}^{N} p(\mathbf{x}_n | \theta)$$

$$\theta = \max_{\theta} \prod_{n=1}^{N} p(\mathbf{x}_n | \theta)$$

$$\theta = \max_{\theta} \sum_{n=1}^{N} \log(p(\mathbf{x}_n | \theta))$$

Thanks!

Any questions?

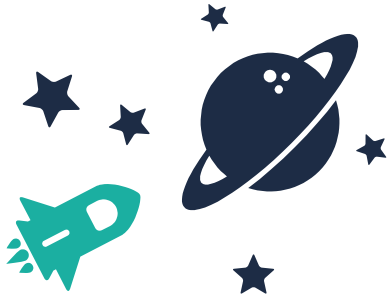
You can find me at:

+84363568384

vieritolove@gmail.com

linkedin.com/in/nthv-techainer

Ha Noi, Viet Nam



Thank you for your attention!