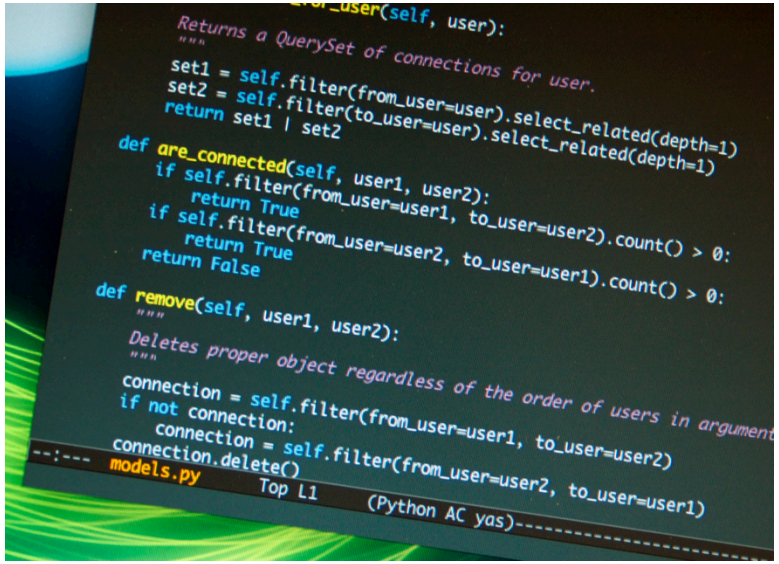
Regression:

A machine learning perspective

Programming experience

- Basic Python used
  - Can pick up along the way if knowledge of other language

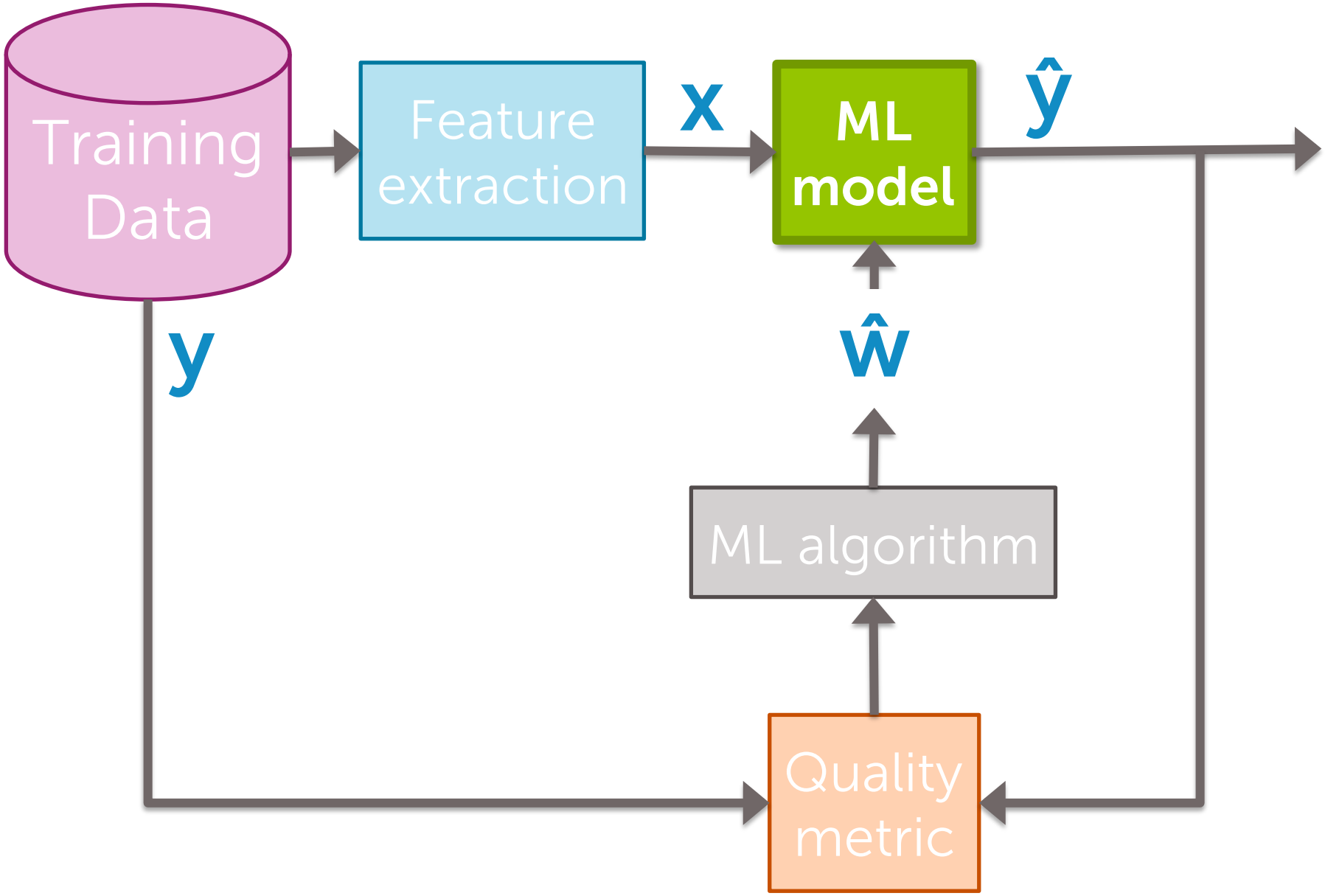


©2015 Emily Fox & Carlos Guestrin

Computing needs

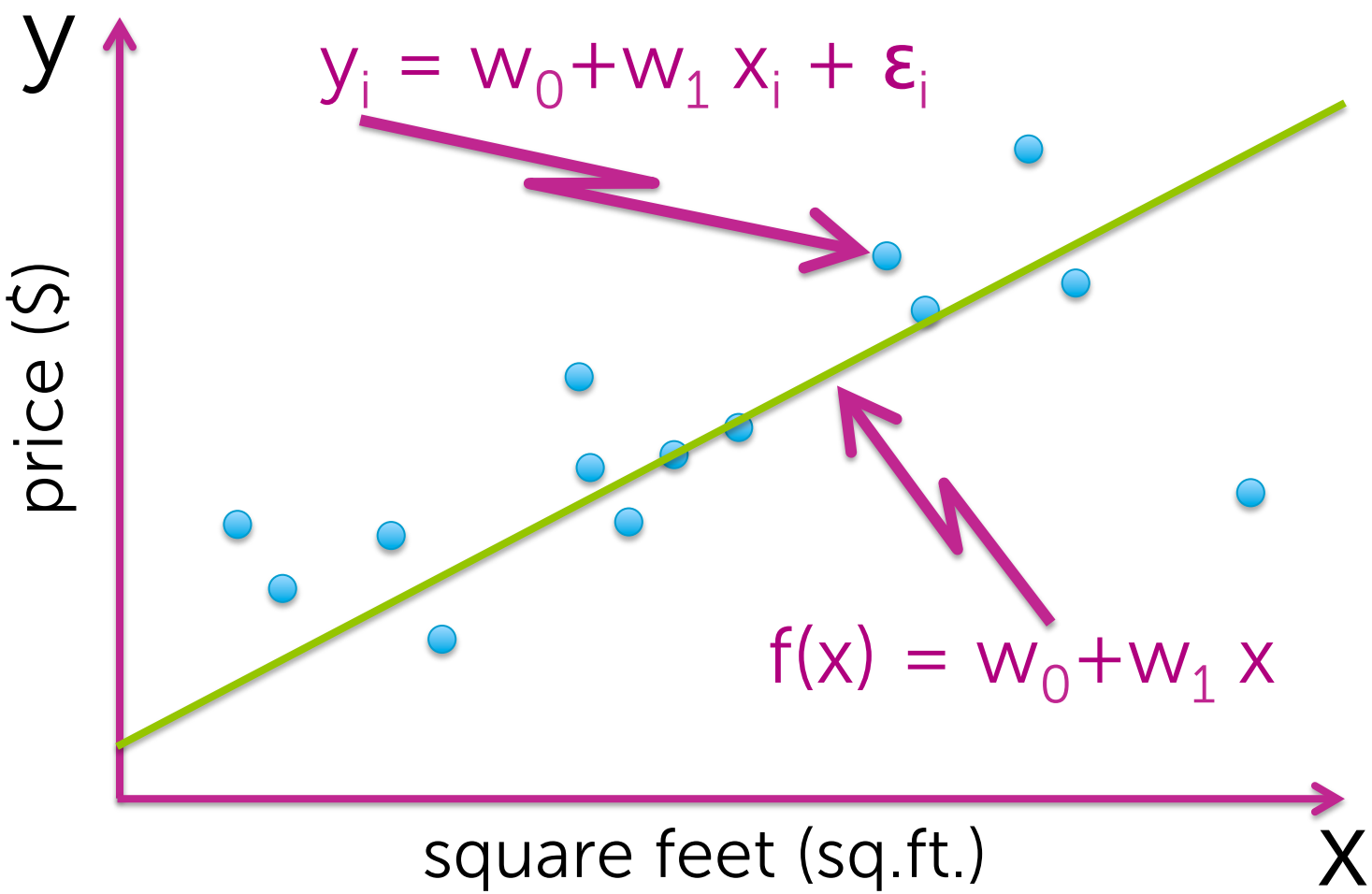
- Basic 64-bit desktop or laptop
- Access to internet
- Ability to:
  - Install and run Python
  - Store a few GB of data

©2015 Emily Fox & Carlos Guestrin



©2015 Emily Fox & Carlos Guestrin

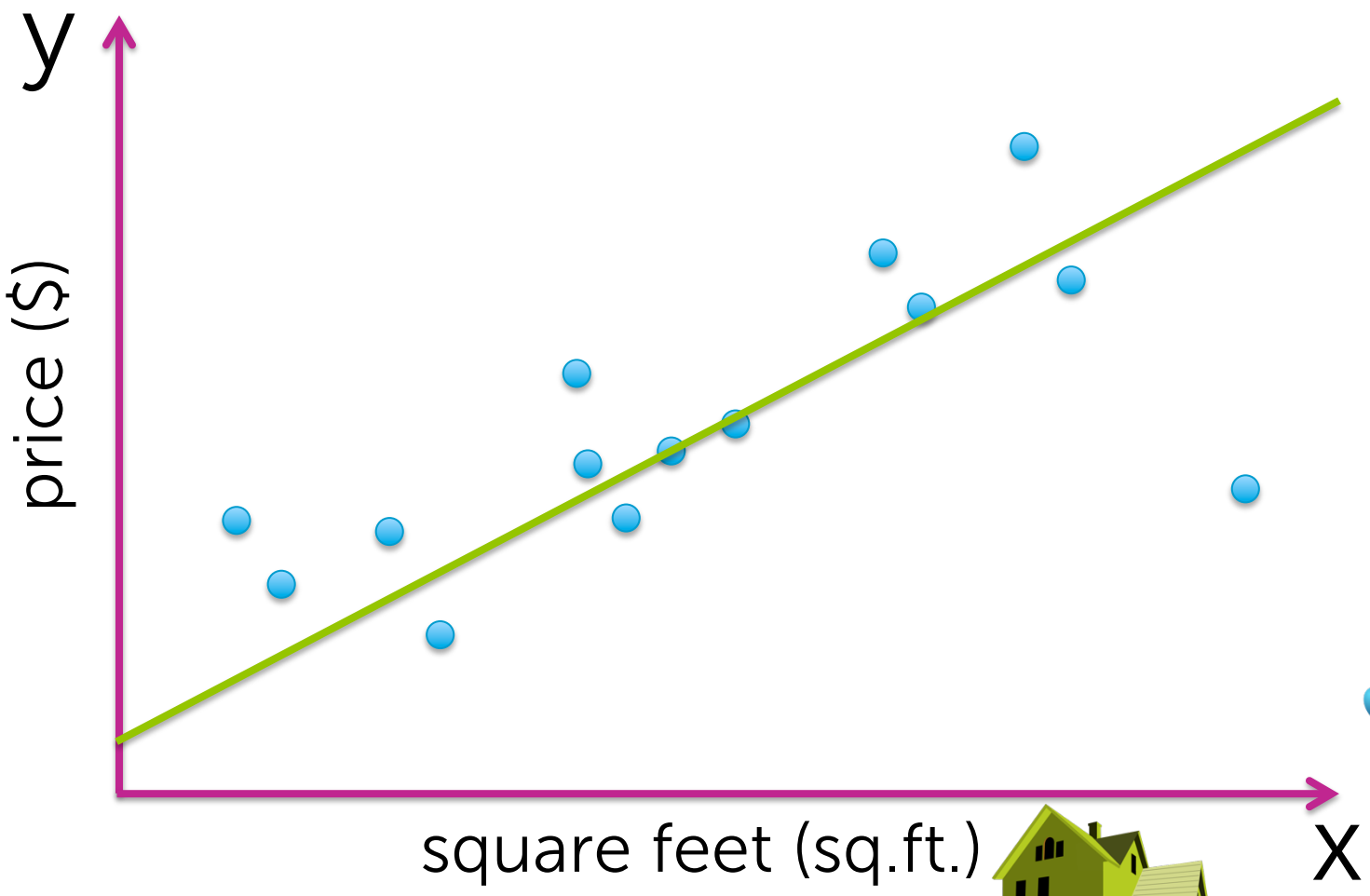
Simple linear regression model

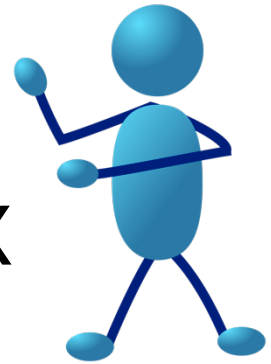


©2015 Emily Fox & Carlos Guestrin

Polynomial regression

Fit data with a line or … ?

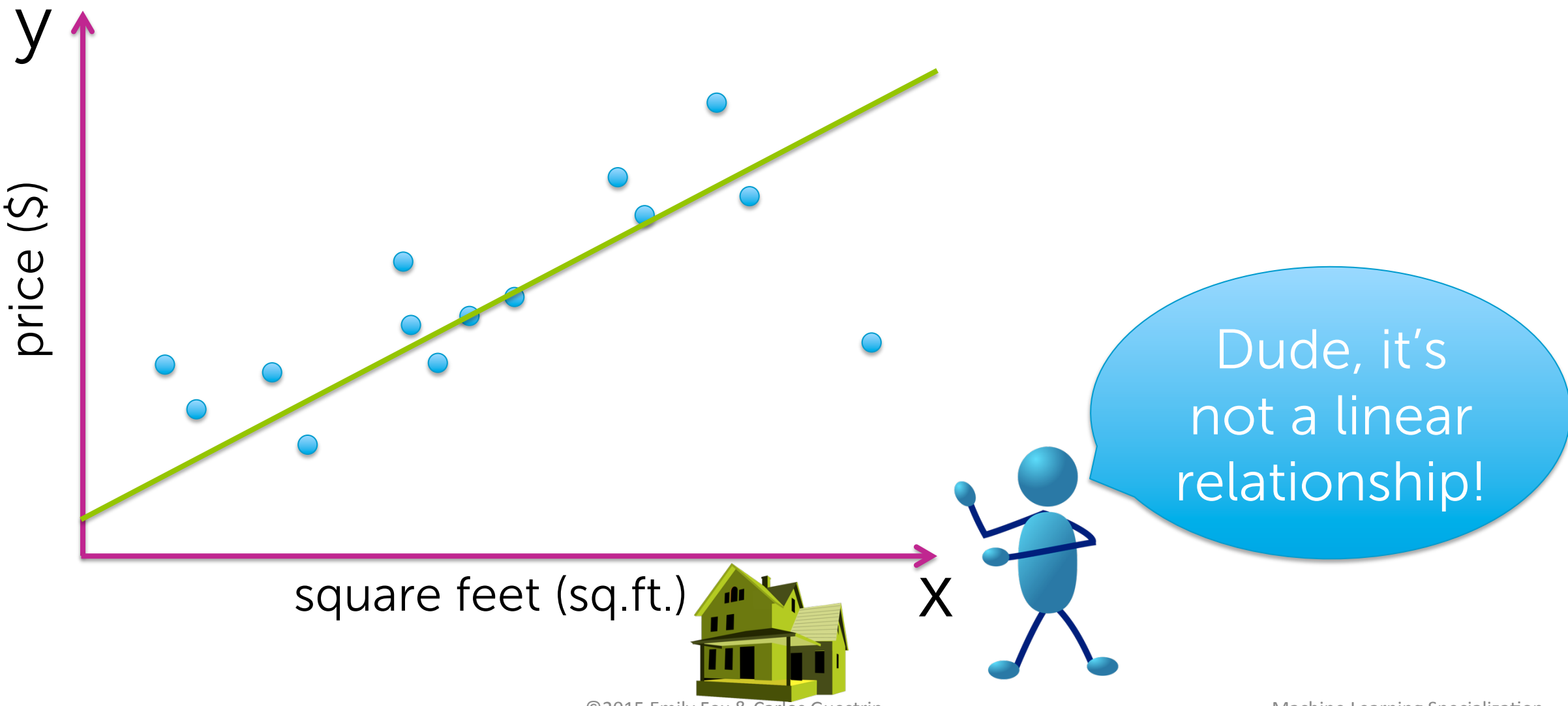




You show your friend your analysis

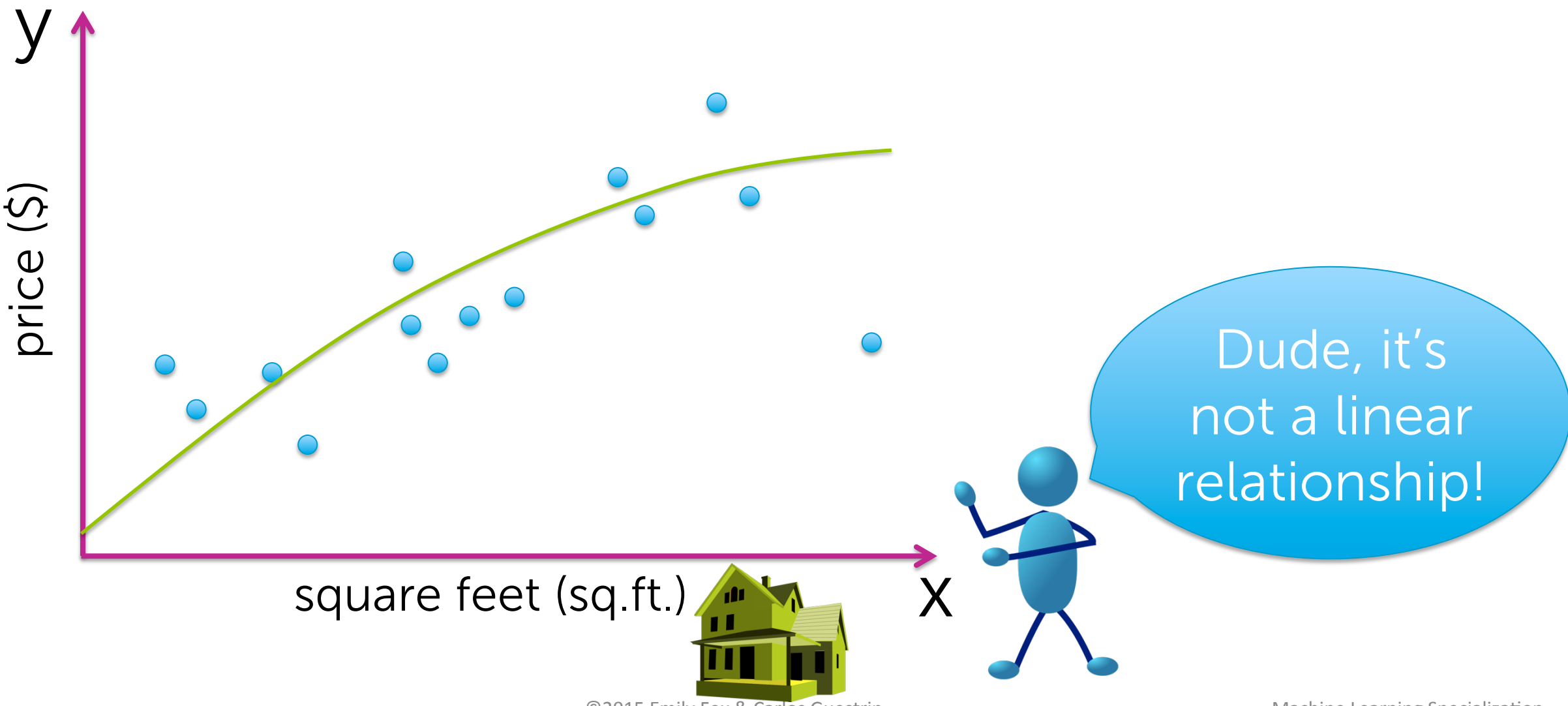
©2015 Emily Fox & Carlos Guestrin

Fit data with a line or … ?



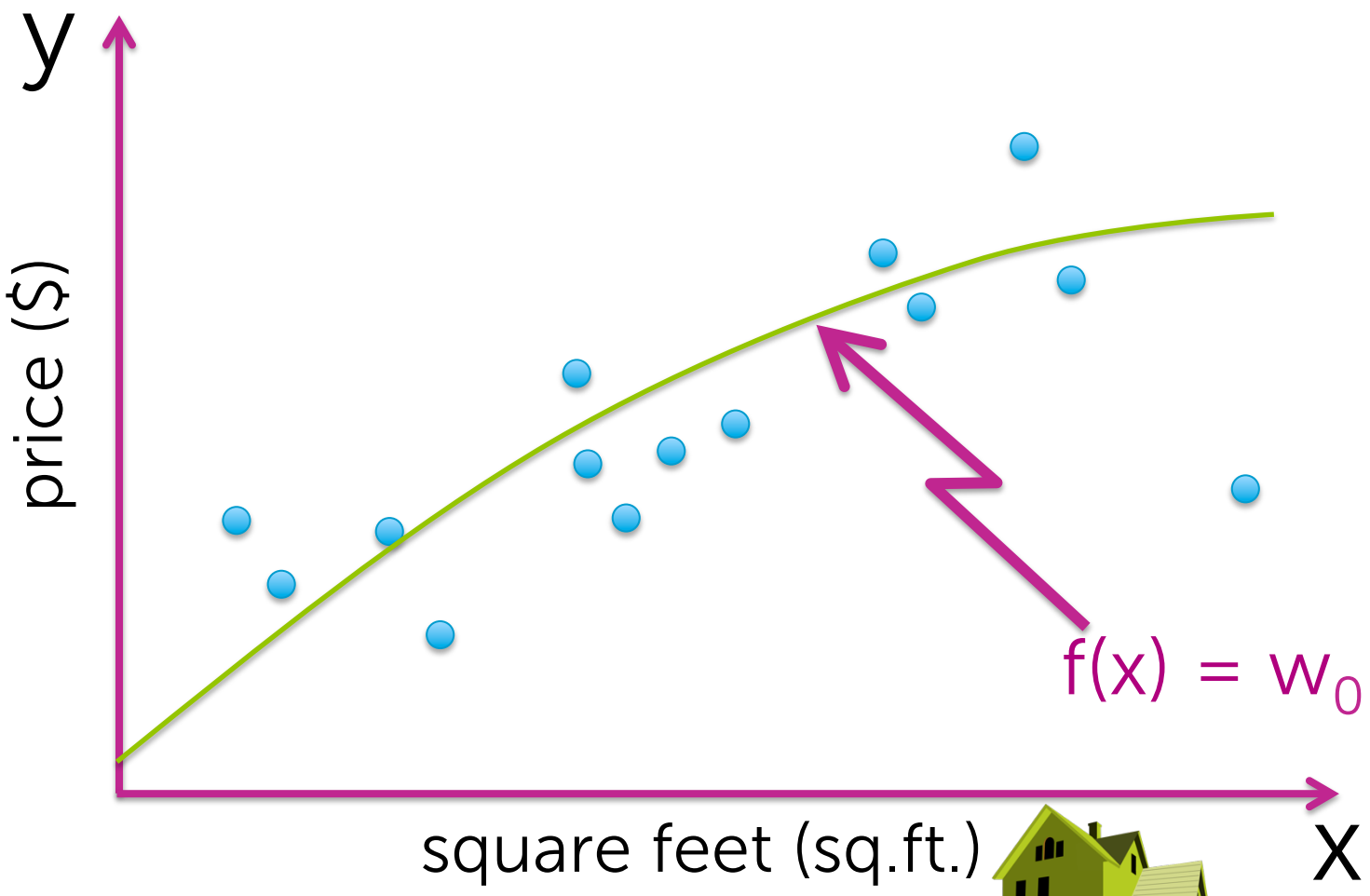
©2015 Emily Fox & Carlos Guestrin

What about a quadratic function?



©2015 Emily Fox & Carlos Guestrin

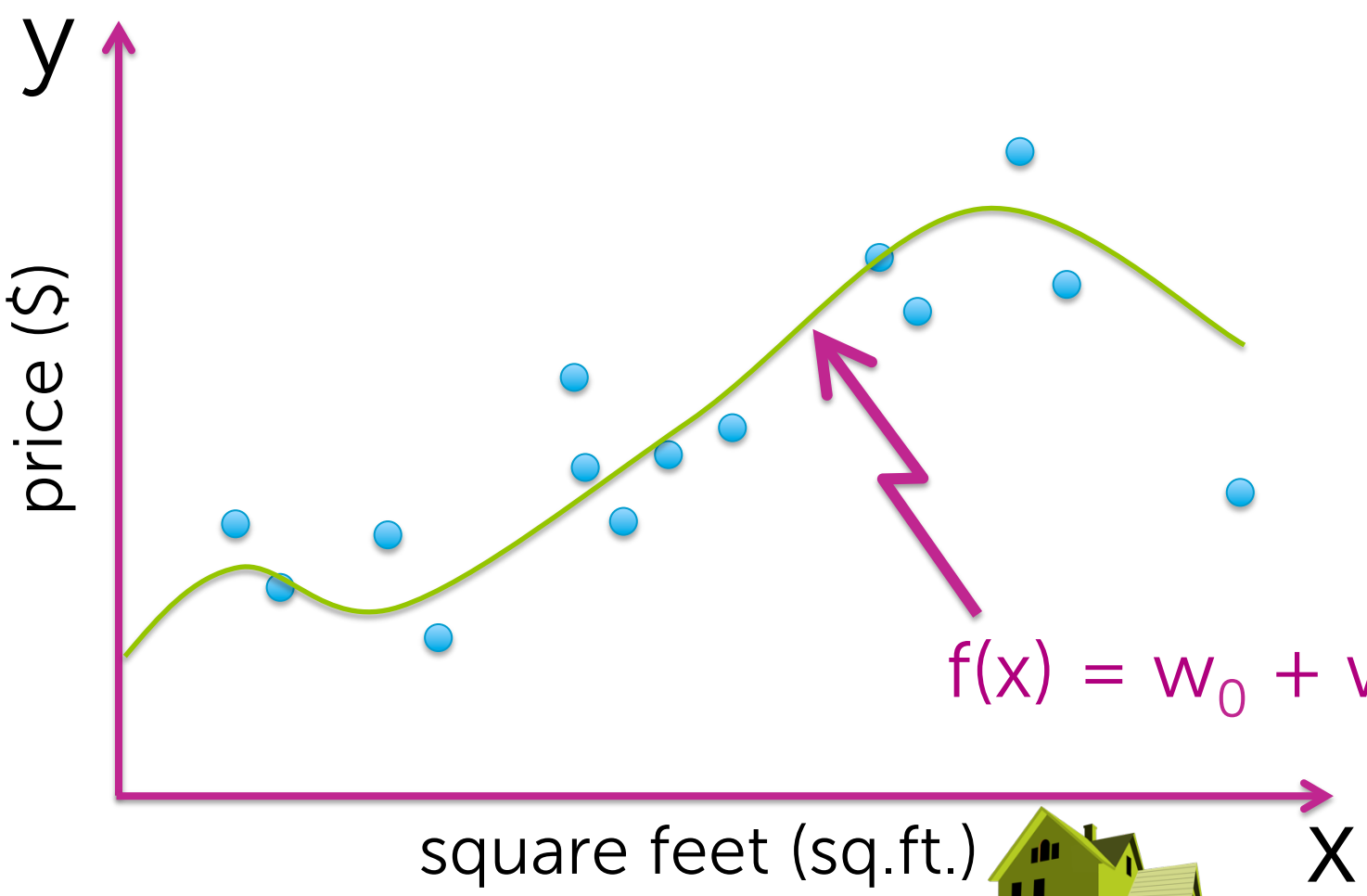
What about a quadratic function?



$$f(x) = w_0 + w_1 x + w_2 x^2$$

©2015 Emily Fox & Carlos Guestrin

Even higher order polynomial



$$f(x) = w_0 + w_1 x + w_2 x^2 + \ldots + w_p x^p$$

©2015 Emily Fox & Carlos Guestrin

Polynomial regression

Model:

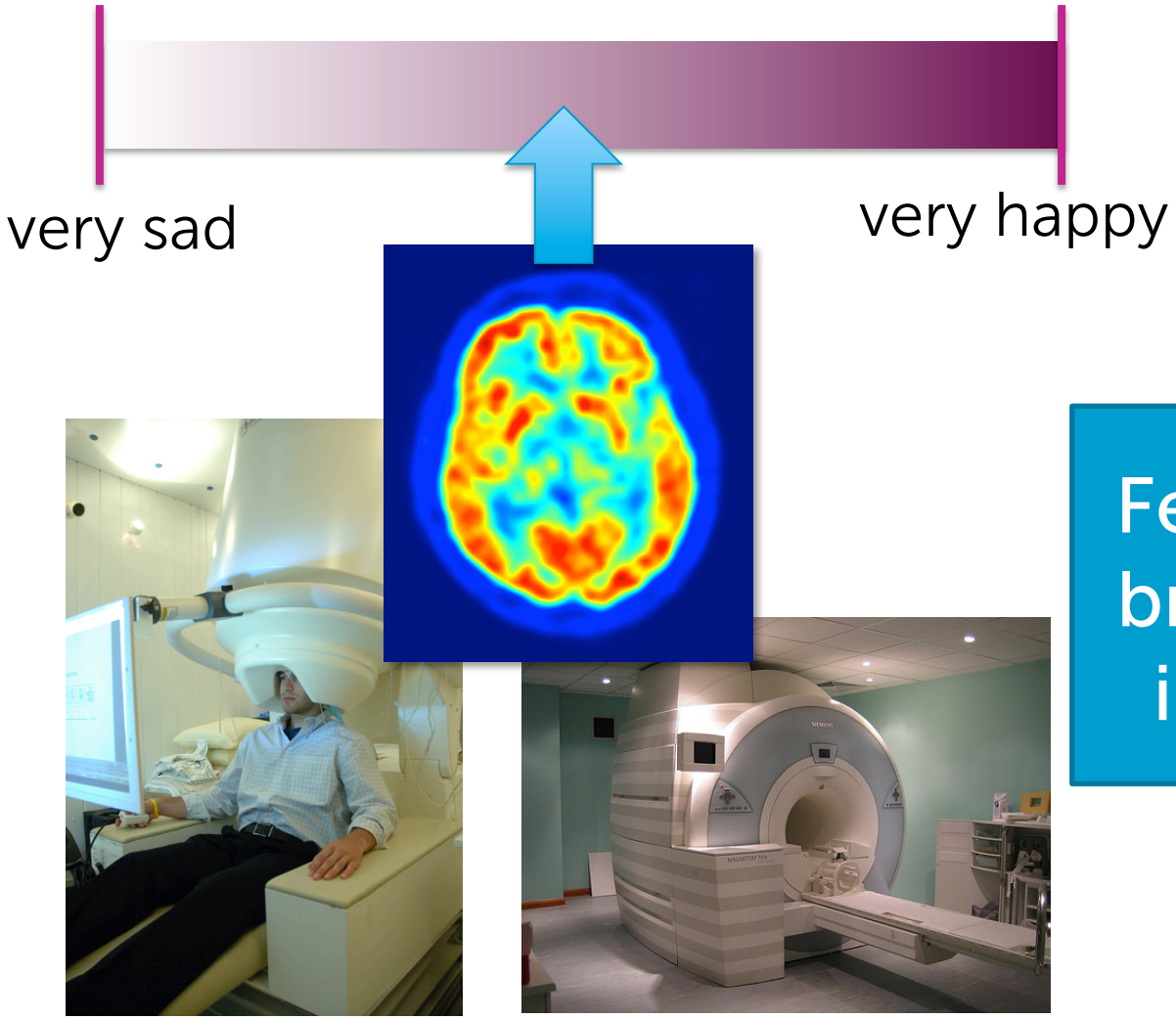
$$y_i = w_0 + w_1 x_i + w_2 x_i^2 + \ldots + w_p x_i^p + \varepsilon_i$$

treat as different **features**

feature 1 = 1 (constant)	*parameter* 1 = w_0
feature 2 = x	*parameter* 2 = w_1
feature 3 = x^2	*parameter* 3 = w_2
…	…
feature p+1 = x^p	*parameter* p+1 = w_p

©2015 Emily Fox & Carlos Guestrin

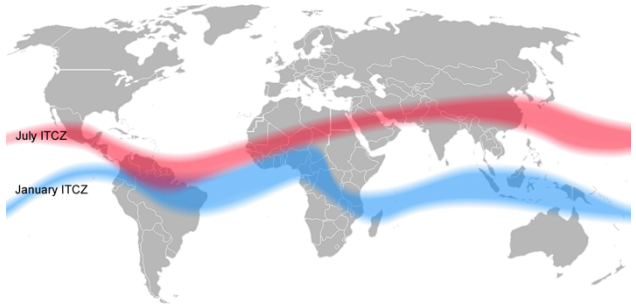
Reading your mind



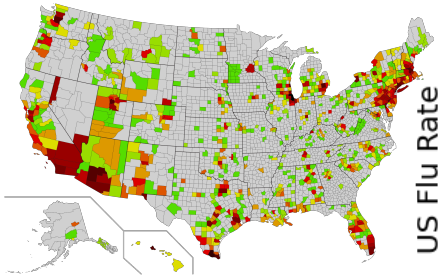
Features are brain region intensities

©2015 Emily Fox & Carlos Guestrin

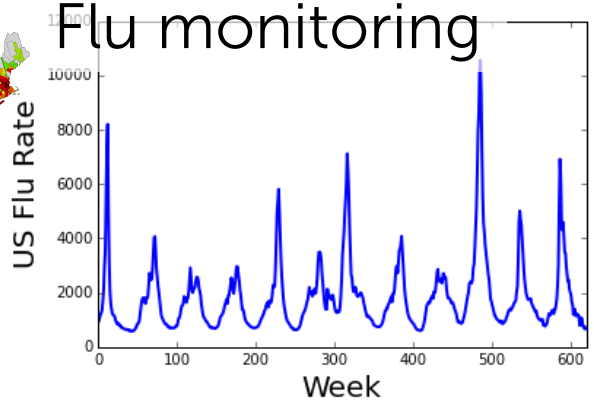
Other examples of seasonality



Weather modeling
(e.g., temperature, rainfall)

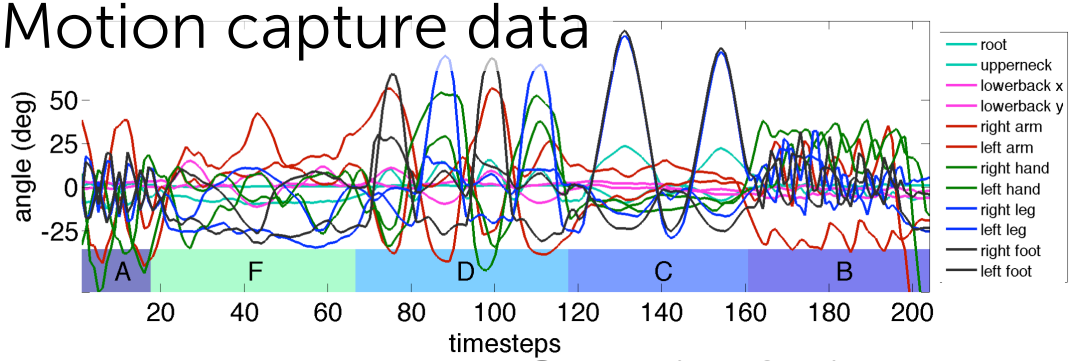


Flu monitoring



Demand forecasting
(e.g., jacket purchases)

Motion capture data



©2015 Emily Fox & Carlos Guestrin

Incorporating multiple inputs

Generic basis expansion

Model:

$$y_i = w_0 h_0(x_i) + w_1 h_1(x_i) + \ldots + w_D h_D(x_i) + \varepsilon_i$$

$$= \sum_{j=0}^{D} w_j h_j(x_i) + \varepsilon_i$$

j^{th} *feature*

j^{th} *regression coefficient* or *weight*

©2015 Emily Fox & Carlos Guestrin

Generic basis expansion

Model:

$$y_i = w_0 h_0(x_i) + w_1 h_1(x_i) + \ldots + w_D h_D(x_i) + \varepsilon_i$$

$$= \sum_{j=0}^{D} w_j h_j(x_i) + \varepsilon_i$$

feature 1 = $h_0(x)$…often 1 (constant)

feature 2 = $h_1(x)$… e.g., x

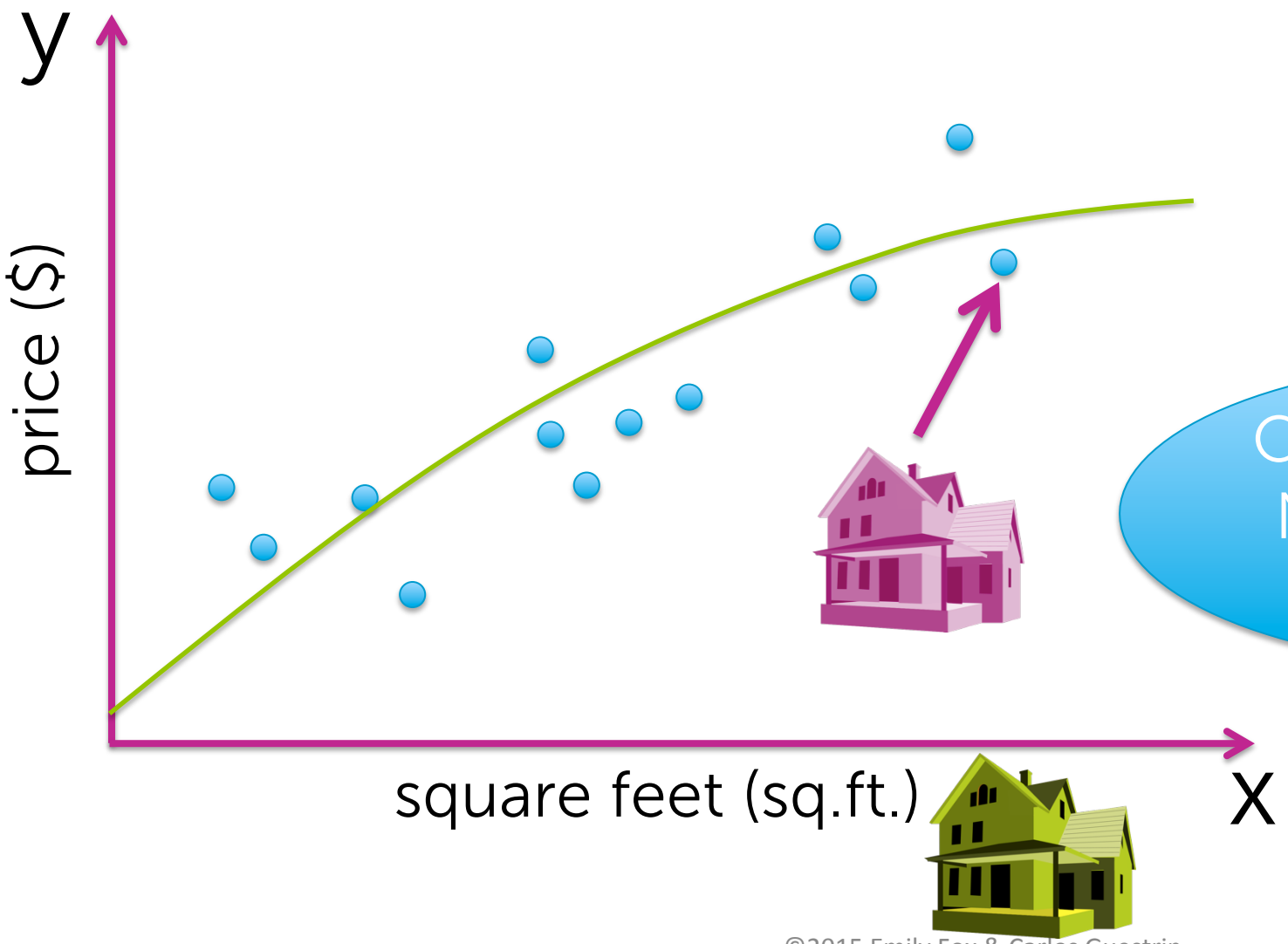
feature 3 = $h_2(x)$… e.g., x^2 or $\sin(2\pi x/12)$

…

feature D+1 = $h_D(x)$… e.g., x^p

©2015 Emily Fox & Carlos Guestrin

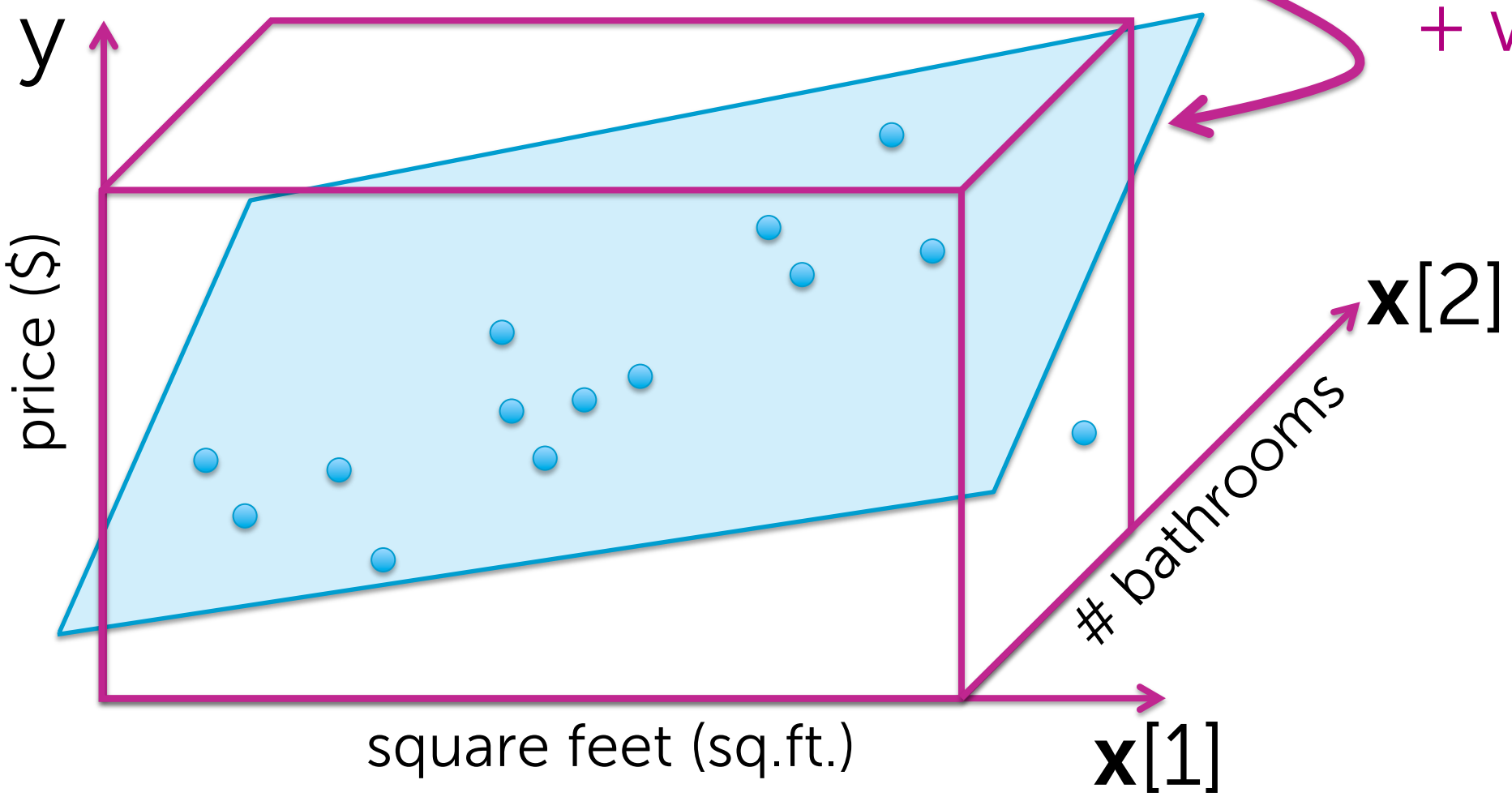
Predictions just based on house size



©2015 Emily Fox & Carlos Guestrin

Add more inputs

$f(\mathbf{x}) = w_0 + w_1 \text{ sq.ft.} + w_2 \text{ \#bath}$



©2015 Emily Fox & Carlos Guestrin

Many possible inputs

- Square feet
- # bathrooms
- # bedrooms
- Lot size
- Year built
- …

©2015 Emily Fox & Carlos Guestrin

General notation

Output: y ← scalar

Inputs: $\mathbf{x} = (\mathbf{x}[1], \mathbf{x}[2], \ldots, \mathbf{x}[d])$

↖ d-dim vector

Notational conventions:

$\mathbf{x}[j]$ = j^{th} input (*scalar*)

$h_j(\mathbf{x})$ = j^{th} feature (*scalar*)

$\mathbf{x}_i$ = input of ith data point (*vector*)

$\mathbf{x}_i[j]$ = j^{th} input of ith data point (*scalar*)

©2015 Emily Fox & Carlos Guestrin

Simple hyperplane

Model:

$y_i = w_0 + w_1 \mathbf{x}_i[1] + \ldots + w_d \mathbf{x}_i[d] + \varepsilon_i$

feature 1 = 1

feature 2 = **x**[1] … e.g., sq. ft.

feature 3 = **x**[2] … e.g., #bath

…

feature d+1 = **x**[d] … e.g., lot size

©2015 Emily Fox & Carlos Guestrin

More generically… D-dimensional curve

Model:

$$y_i = w_0\, h_0(\mathbf{x}_i) + w_1\, h_1(\mathbf{x}_i) + \ldots + w_D\, h_D(\mathbf{x}_i) + \varepsilon_i$$

$$= \sum_{j=0}^{D} w_j\, h_j(\mathbf{x}_i) + \varepsilon_i$$

feature 1 = $h_0(\mathbf{x})$ … e.g., 1

feature 2 = $h_1(\mathbf{x})$ … e.g., $\mathbf{x}[1]$ = sq. ft.

feature 3 = $h_2(\mathbf{x})$ … e.g., $\mathbf{x}[2]$ = #bath

or, $\log(\mathbf{x}[7])\ \mathbf{x}[2]$ = log(#bed) x #bath

…

feature D+1 = $h_D(\mathbf{x})$ … some other function of $\mathbf{x}[1],\ldots,\mathbf{x}[d]$

©2015 Emily Fox & Carlos Guestrin

More on notation

observations $(\mathbf{x}_i, y_i)$: N

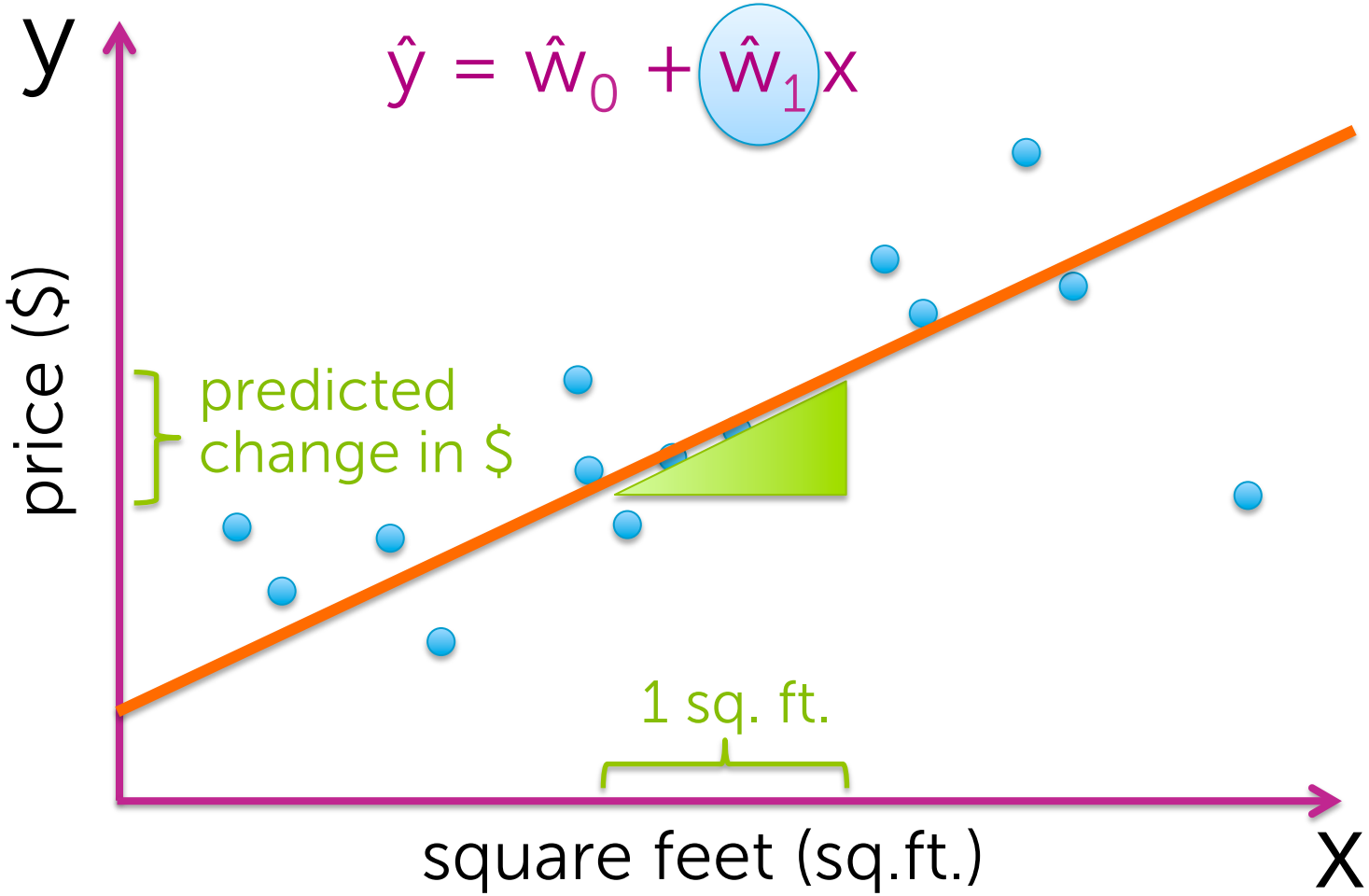
inputs $\mathbf{x}[j]$: d

features $h_j(\mathbf{x})$: D

©2015 Emily Fox & Carlos Guestrin

Interpreting the fitted function

Interpreting the coefficients – Simple linear regression

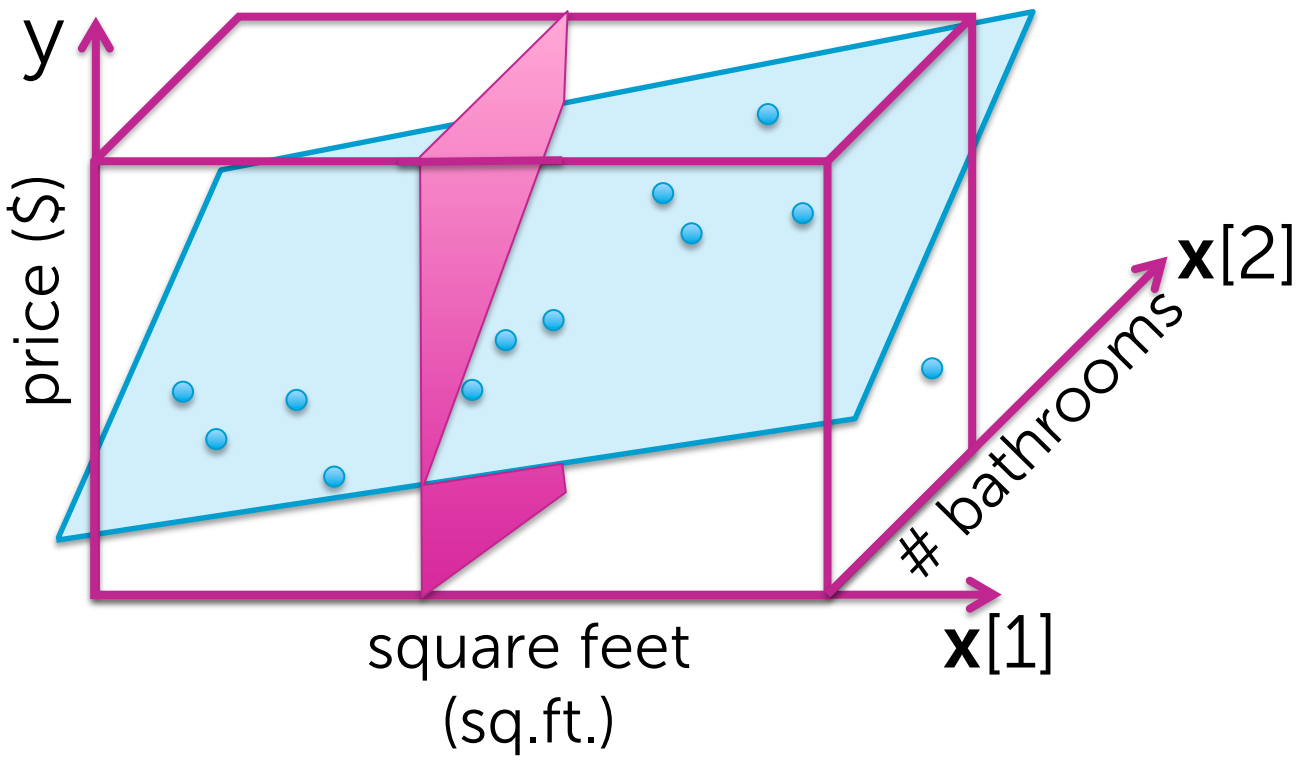


©2015 Emily Fox & Carlos Guestrin

Interpreting the coefficients – Two linear features

$$\hat{y} = \hat{w}_0 + \hat{w}_1 \boxed{\mathbf{x}[1]} + \hat{w}_2 \mathbf{x}[2]$$

fix

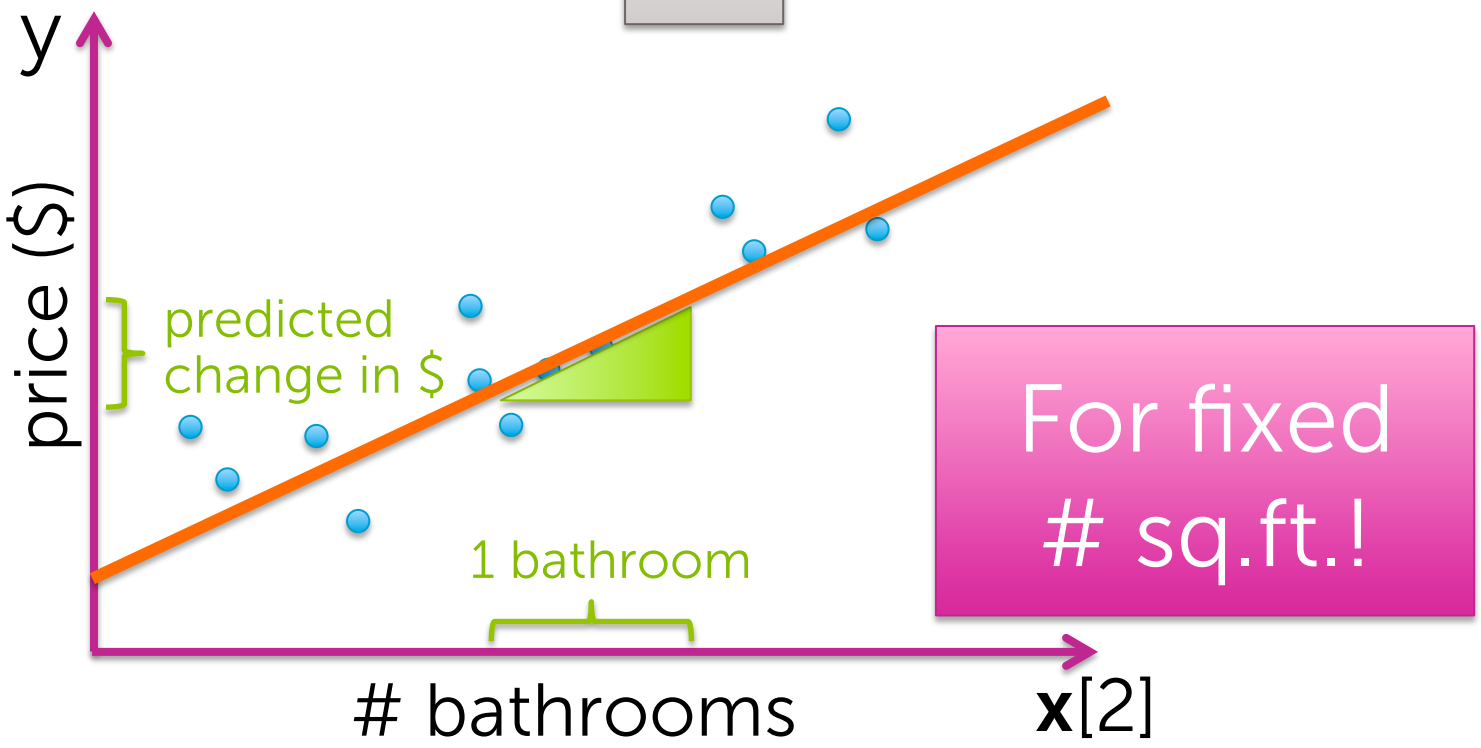


©2015 Emily Fox & Carlos Guestrin

Interpreting the coefficients – Two linear features

$$\hat{y} = \hat{w}_0 + \hat{w}_1 \boxed{\mathbf{x}[1]} + \hat{w}_2 \mathbf{x}[2]$$

fix



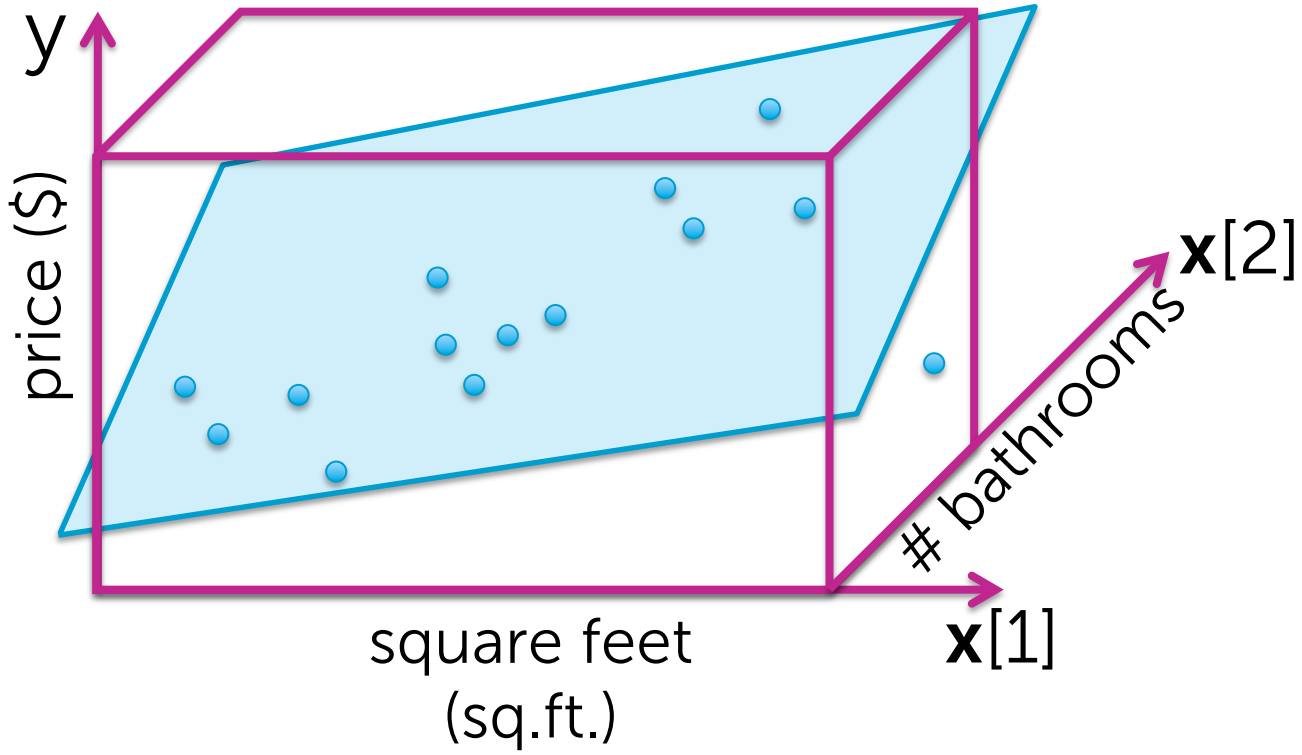
For fixed
sq.ft.!

©2015 Emily Fox & Carlos Guestrin

Interpreting the coefficients – Multiple linear features

$$\hat{y} = \hat{w}_0 + \hat{w}_1 \boxed{\mathbf{x}[1]} + \ldots + \hat{w}_j \mathbf{x}[j] + \ldots + \hat{w}_d \boxed{\mathbf{x}[d]}$$

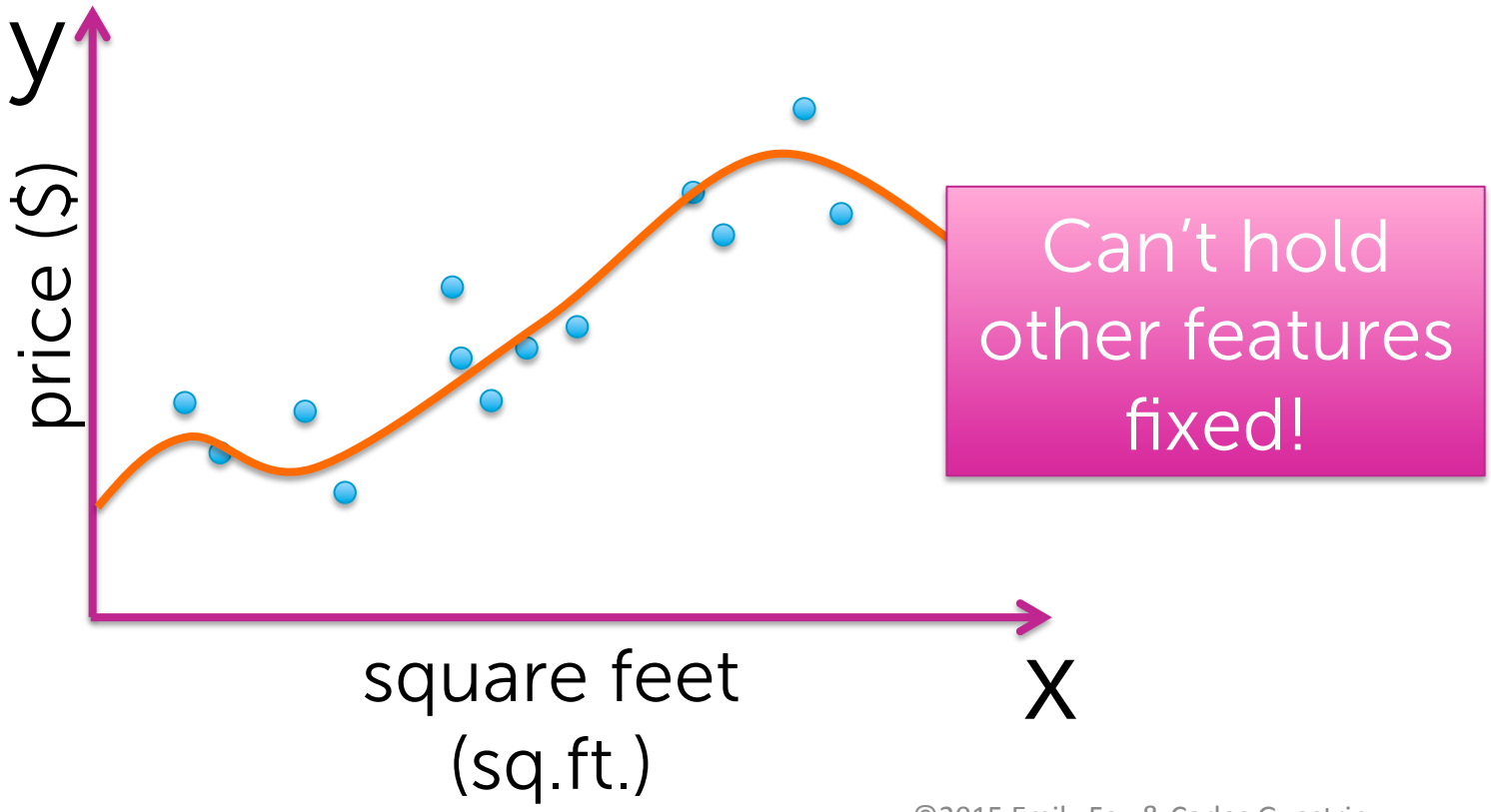
fix fix fix fix



©2015 Emily Fox & Carlos Guestrin

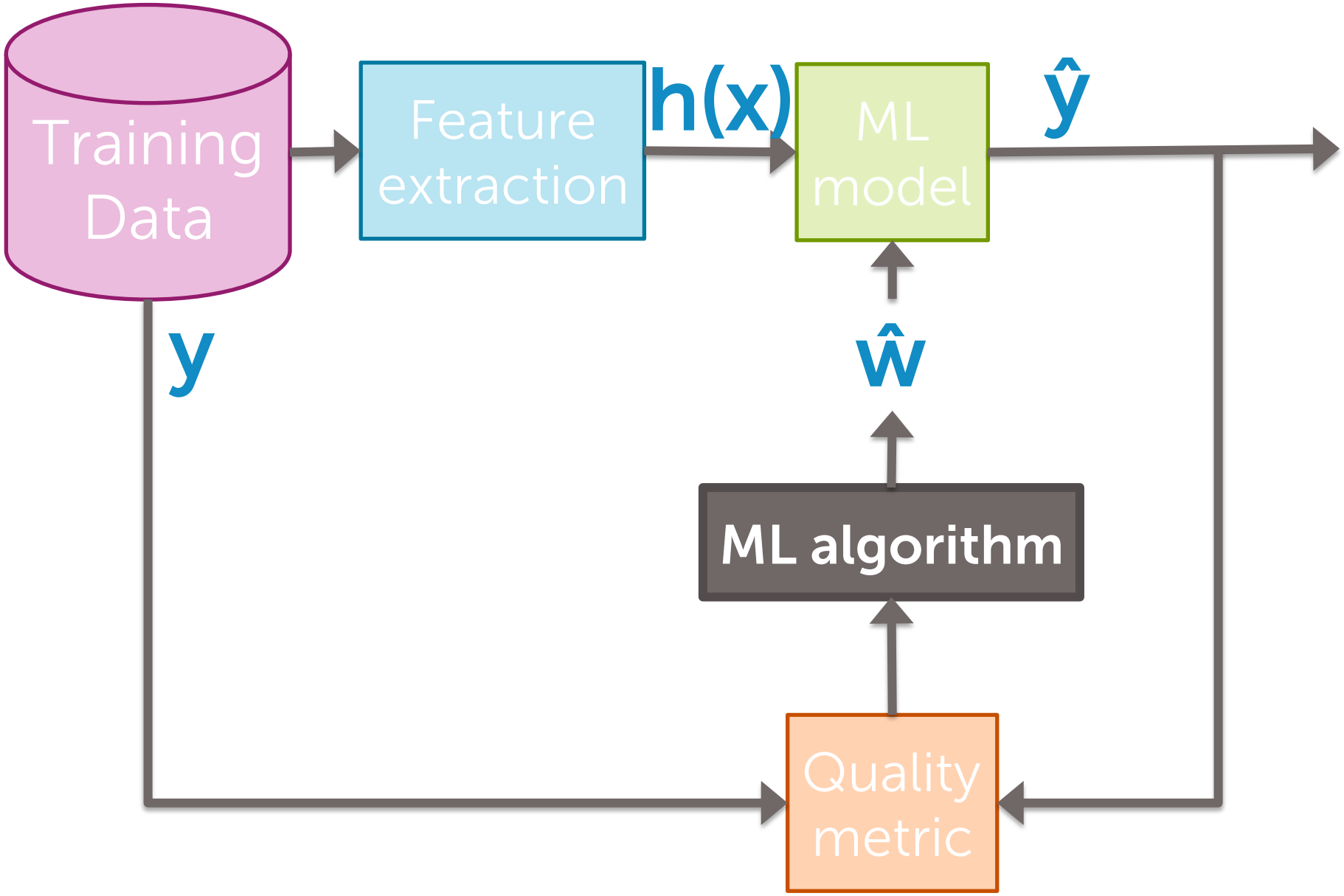
Interpreting the coefficients- Polynomial regression

$$\hat{y} = \hat{w}_0 + \hat{w}_1 x + \ldots + \hat{w}_j x^j + \ldots + \hat{w}_p x^p$$



©2015 Emily Fox & Carlos Guestrin

Fitting D-dimensional curves



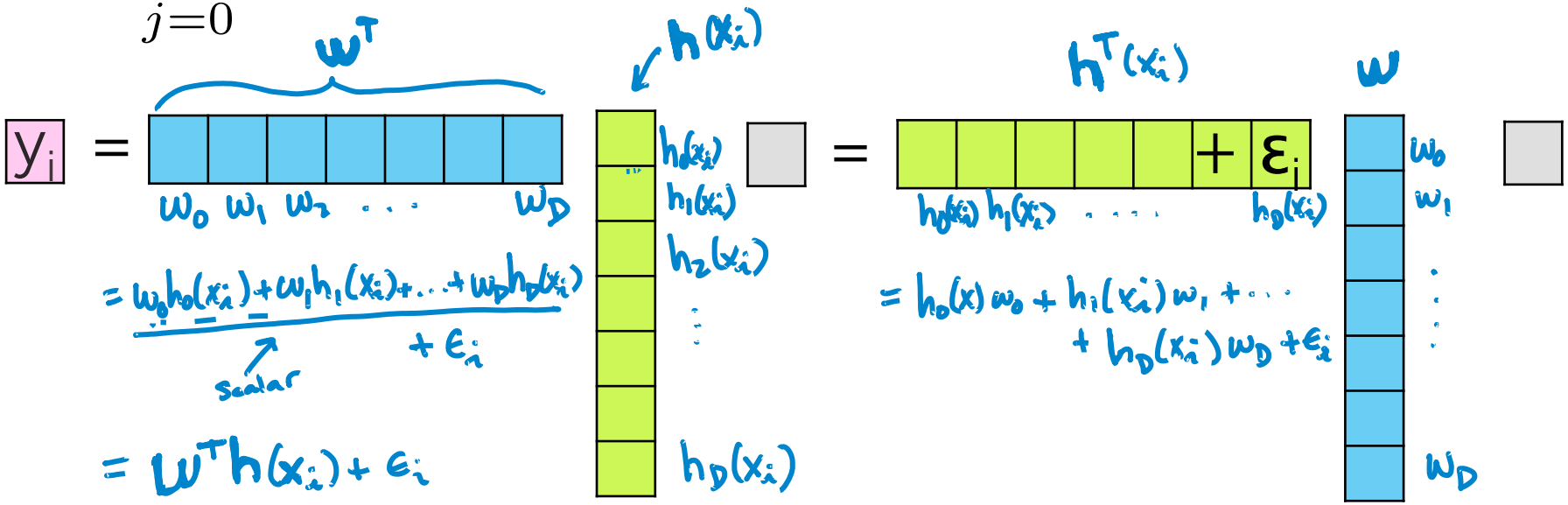
©2015 Emily Fox & Carlos Guestrin

Step 1:
Rewrite the regression model

Rewrite in matrix notation

For observation i

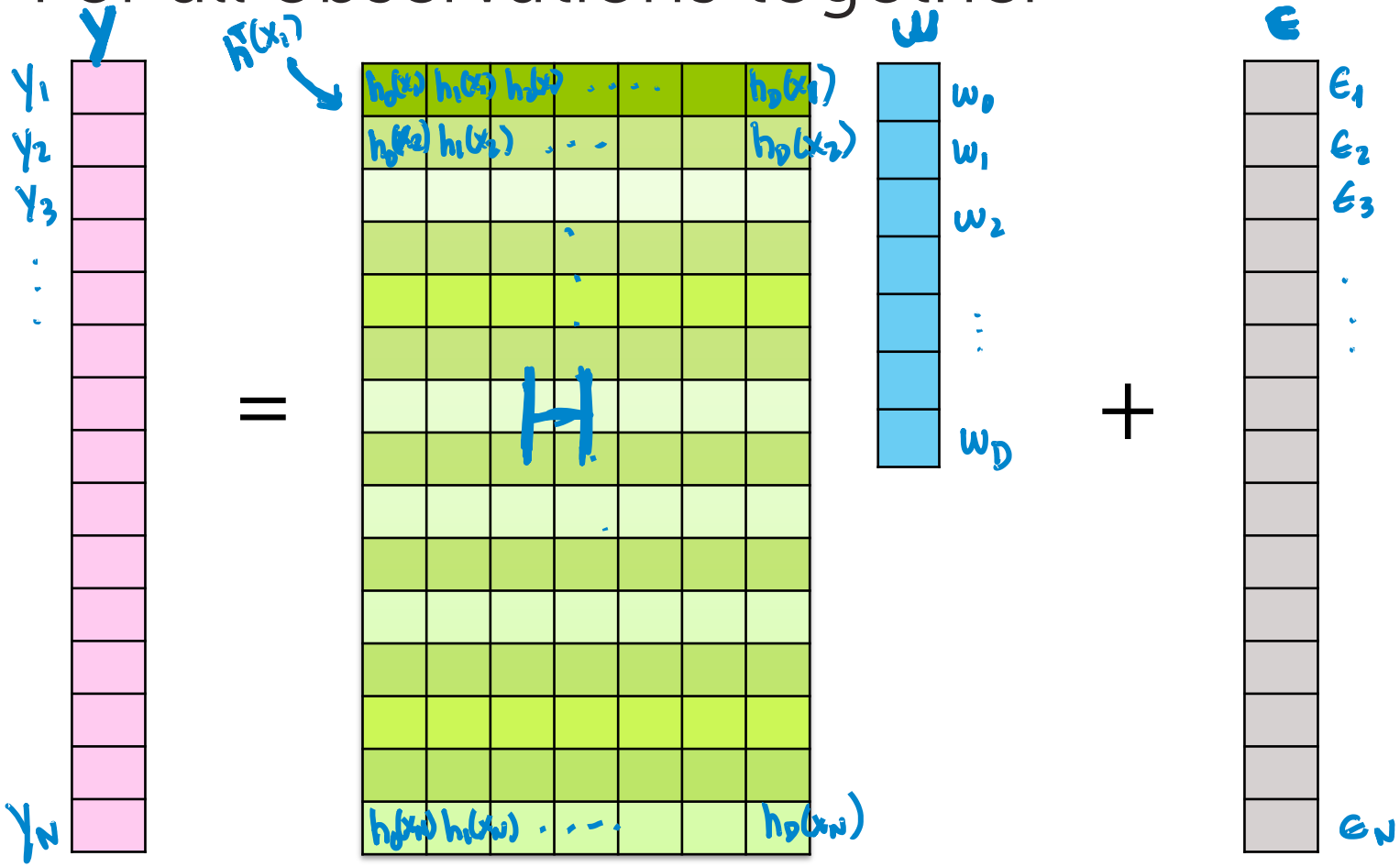
$$y_i = \sum_{j=0}^{D} w_j \, h_j(\mathbf{x}_i) + \varepsilon_i$$



©2015 Emily Fox & Carlos Guestrin

Rewrite in matrix notation

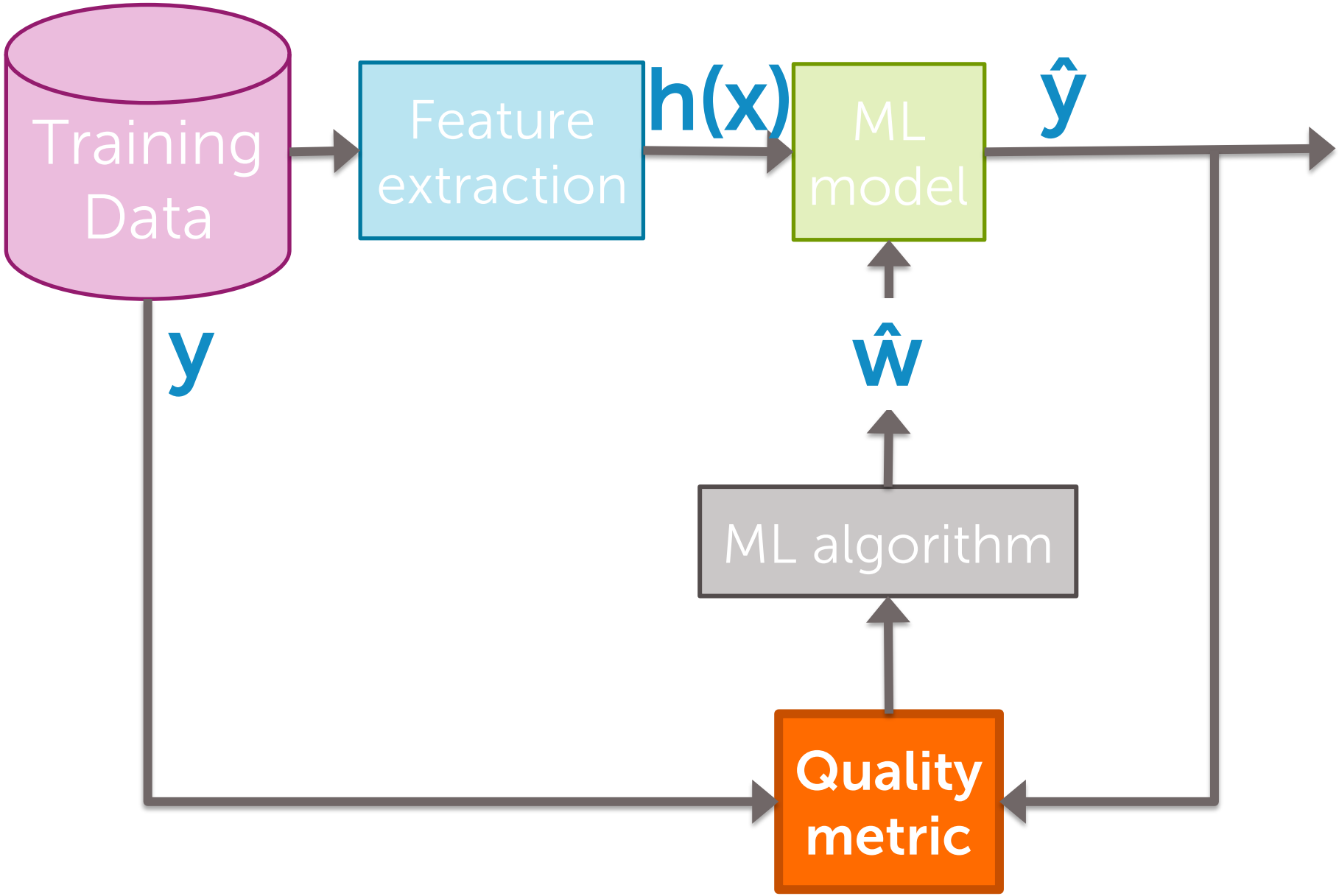
For all observations together



$$\Rightarrow \boxed{y = Hw + \epsilon}$$

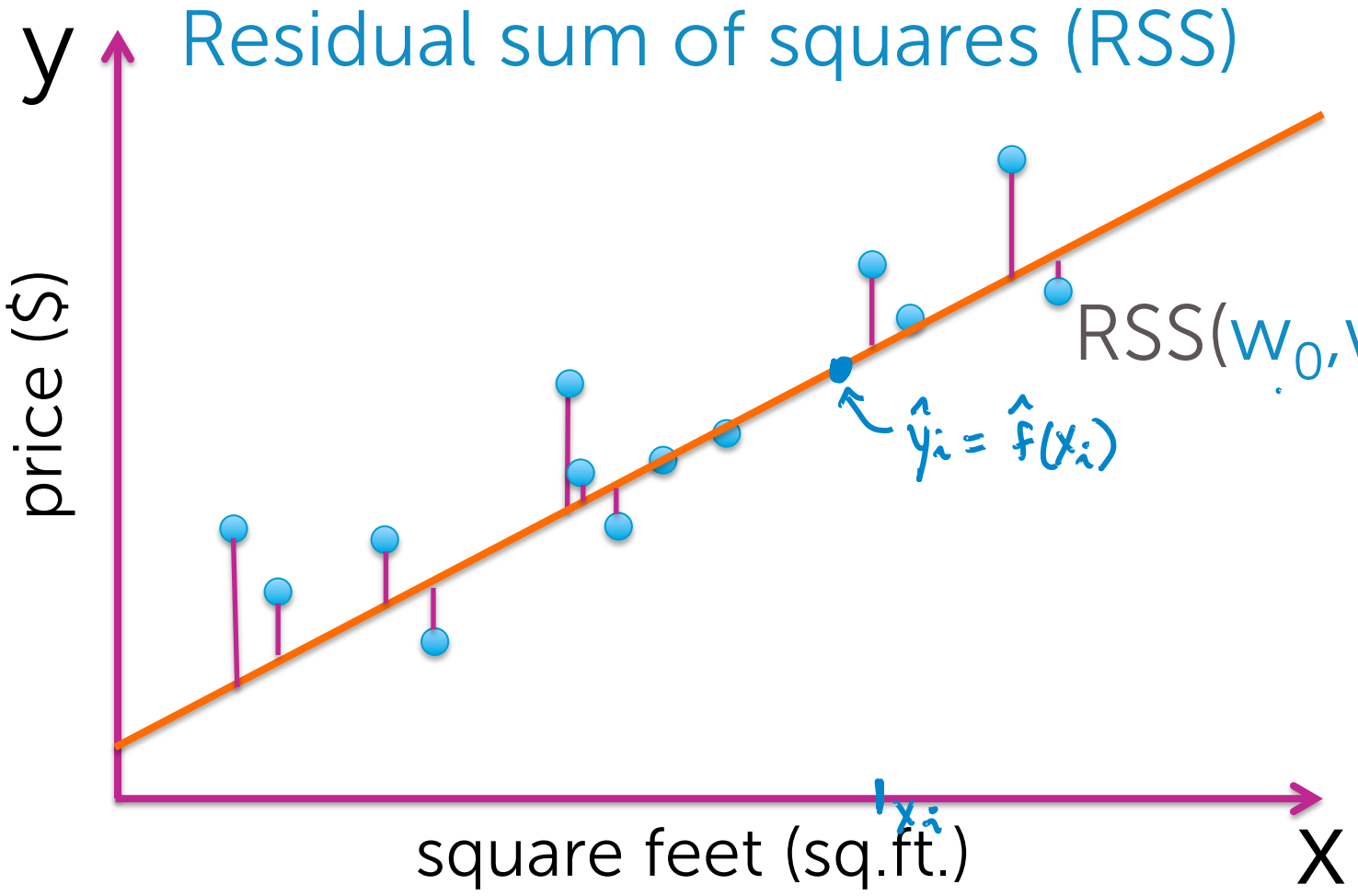
©2015 Emily Fox & Carlos Guestrin

Step 2:
Compute the cost



©2015 Emily Fox & Carlos Guestrin

"Cost" of using a given line

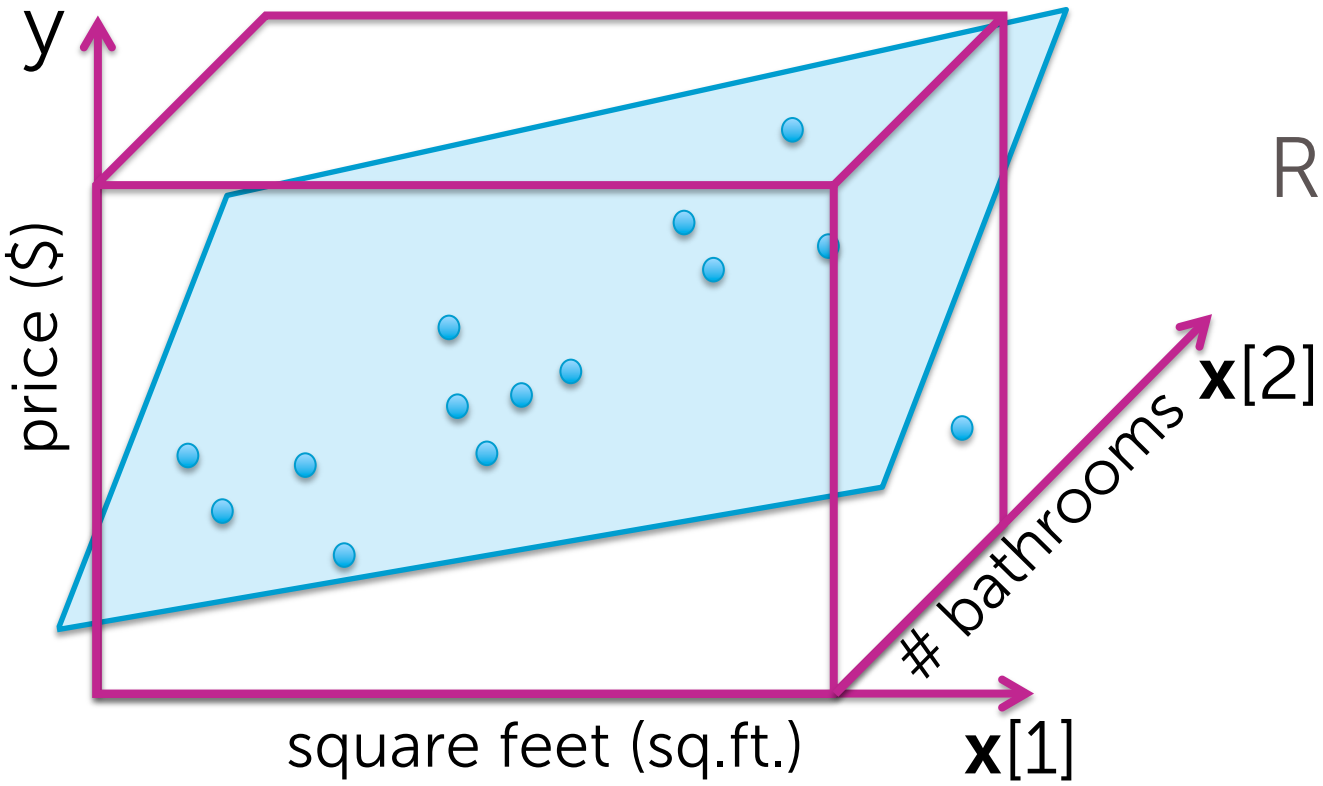


Residual sum of squares (RSS)

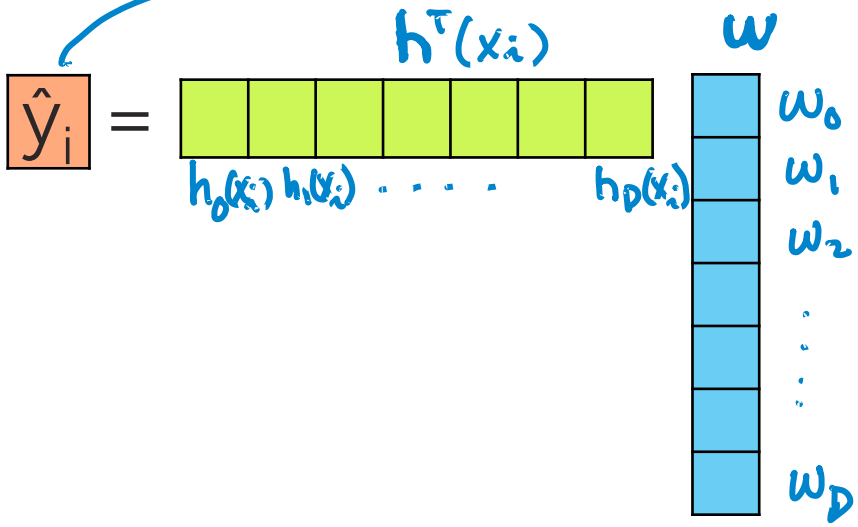
$$\text{RSS}(w_0, w_1) = \sum_{i=1}^{N} (y_i - \underbrace{[w_0 + w_1 x_i]}_{\hat{y}_i(w_0, w_1)})^2$$

©2015 Emily Fox & Carlos Guestrin

RSS for multiple regression



$$\text{RSS}(\mathbf{w}) = \sum_{i=1}^{N} (y_i - \underbrace{h^T(x_i)\,w}_{\hat{y}_i(w)})^2$$



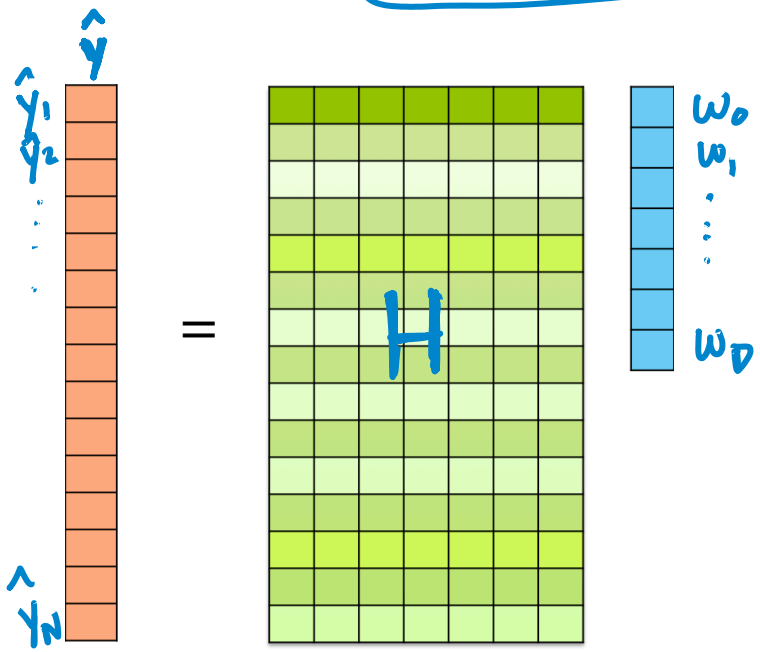
©2015 Emily Fox & Carlos Guestrin

RSS in matrix notation

$$\text{RSS}(\mathbf{w}) = \sum_{i=1}^{N} (y_i - h(\mathbf{x}_i)^T \mathbf{w})^2$$

$$= (y - Hw)^T (y - Hw)$$

Why? (part 1)



$$\hat{y} = Hw$$

$$(y - Hw) = (y - \hat{y}) = \begin{bmatrix} \text{residual}_1 \\ \text{residual}_2 \\ \vdots \\ \text{residual}_N \end{bmatrix}$$

$\text{residual}_i = y_i - \hat{y}_i$

©2015 Emily Fox & Carlos Guestrin

RSS in matrix notation

$$\text{RSS}(\mathbf{w}) = \sum_{i=1}^{N} (y_i - h(\mathbf{x}_i)^T \mathbf{w})^2$$

$$= (\mathbf{y} - \mathbf{H}\mathbf{w})^T (\mathbf{y} - \mathbf{H}\mathbf{w})$$

Why? (part 2)

residual_1	residual_2	residual_3	…	residual_N

residual_1
residual_2
residual_3
…
residual_N

$$= (\text{residual}_1^2 + \text{residual}_2^2 + \cdots + \text{residual}_N^2)$$

$$= \sum_{i=1}^{N} \text{residual}_i^2$$

$$\triangleq \text{RSS}(\mathbf{w})$$

©2015 Emily Fox & Carlos Guestrin

Step 3:
Take the gradient

Gradient of RSS

$$\nabla \mathrm{RSS}(\mathbf{w}) = \nabla[(\mathbf{y}-\mathbf{H}\mathbf{w})^T(\mathbf{y}-\mathbf{H}\mathbf{w})]$$

$$= -2\mathbf{H}^T(\mathbf{y}-\mathbf{H}\mathbf{w})$$

Why? By analogy to 1D case:

$$\frac{d}{dw}(y-hw)(y-hw) = \frac{d}{dw}(y-hw)^2 = 2\cdot(y-hw)^1(-h)$$

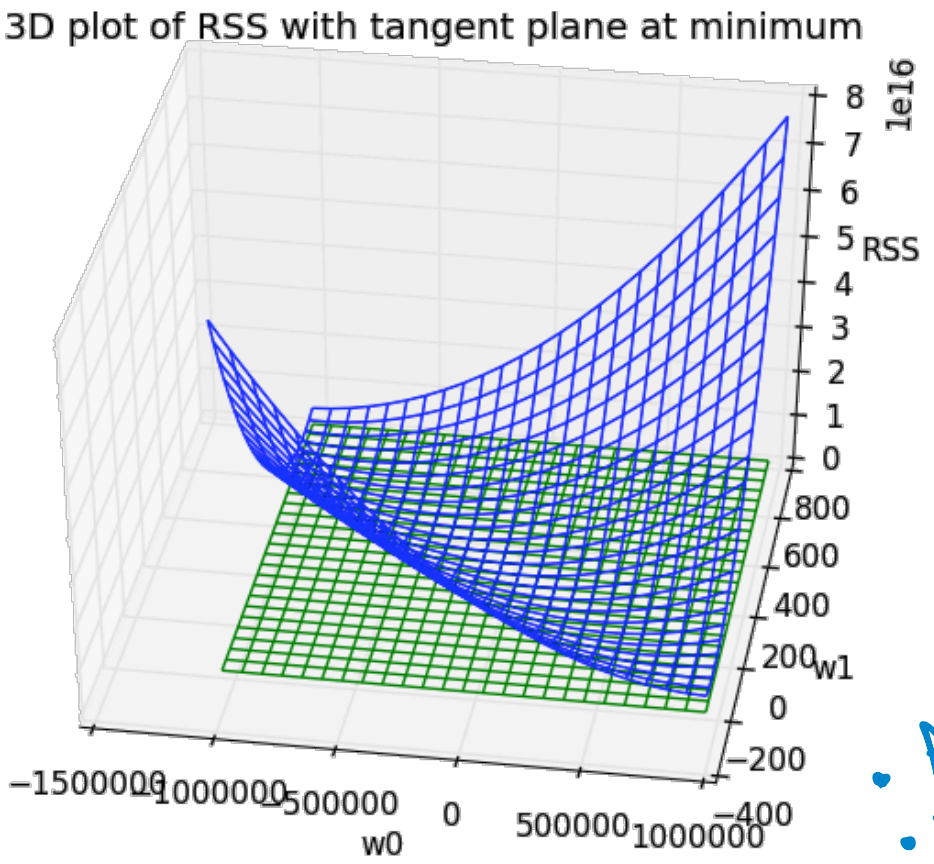
$$= -2h(y-hw)$$

scalars

©2015 Emily Fox & Carlos Guestrin

Step 4, Approach 1:
Set the gradient = 0

Closed-form solution



$$\nabla \mathrm{RSS}(\mathbf{w}) = -2\mathbf{H}^T(\mathbf{y}-\mathbf{H}\mathbf{w}) = 0$$

Solve for $\mathbf{w}$:

$$-\cancel{2}H^T y + \cancel{2}H^T H \hat{w} = 0$$

$$H^T H \hat{w} = H^T y$$

$$\underbrace{(H^T H)^{-1} H^T H}_{I} \hat{w} = (H^T H)^{-1} H^T y$$

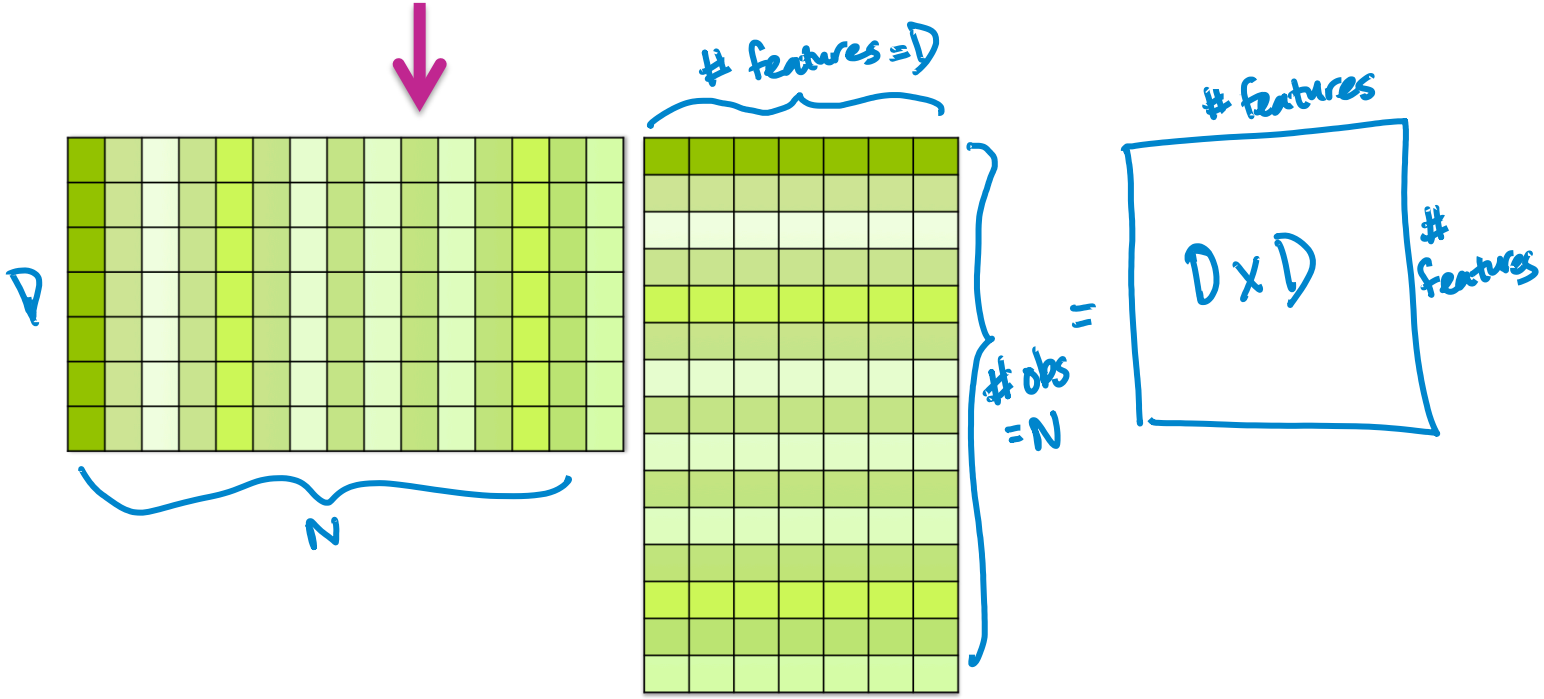
$$\hat{w} = (H^T H)^{-1} H^T y$$

- $A^{-1}A = I$
- $Iv = v$
  $IV = V$

©2015 Emily Fox & Carlos Guestrin

Closed-form solution

$$\hat{\mathbf{w}} = (\underbrace{\mathbf{H}^T\mathbf{H}}_{})^{-1}\mathbf{H}^T\mathbf{y}$$



Invertible if:

really, # of linearly ind. observations

In most cases is $N > D$

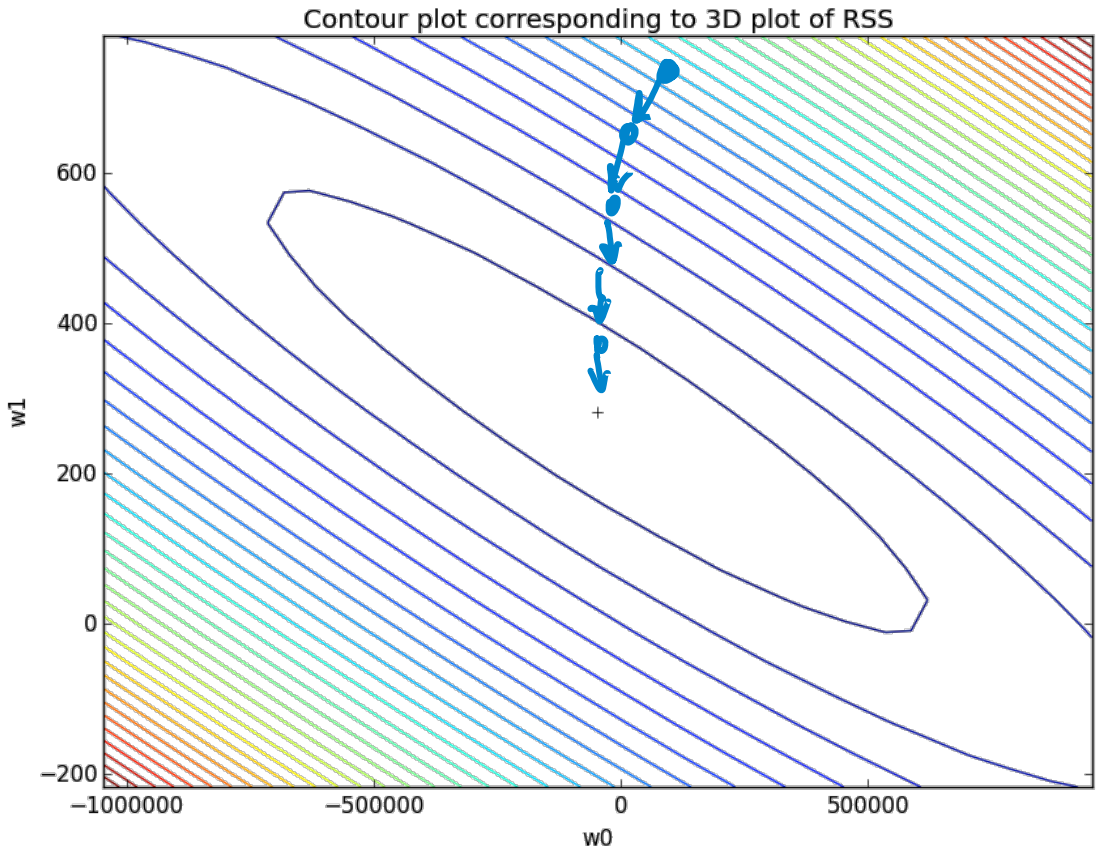
Complexity of inverse:

$$O(D^3)$$

©2015 Emily Fox & Carlos Guestrin

Step 4, Approach 2:
Gradient descent

Gradient descent



while not converged

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} - \eta \underbrace{\nabla RSS(\mathbf{w}^{(t)})}_{-2\mathbf{H}^T(\mathbf{y}-\mathbf{H}\mathbf{w})}$$

$$\leftarrow w^{(t)} + 2\eta H^T(y - \underbrace{Hw^{(t)}}_{\hat{y}(w^{(t)})})$$

©2015 Emily Fox & Carlos Guestrin

Feature-by-feature update

$$\text{RSS}(\mathbf{w}) = \sum_{i=1}^{N} (y_i - h(\mathbf{x}_i)^T\mathbf{w})^2$$

$$= \sum_{i=1}^{N} (y_i - w_0 h_0(x_i) - w_1 h_1(x_i) - \dots - w_D h_D(x_i))^2$$

Partial with respect to w_j

$$\sum_{i=1}^{N} 2(y_i - w_0 h_0(x_i) - w_1 h_1(x_i) \dots - w_D h_D(x_i))^1 \cdot (-\underline{h_j(x_i)})$$

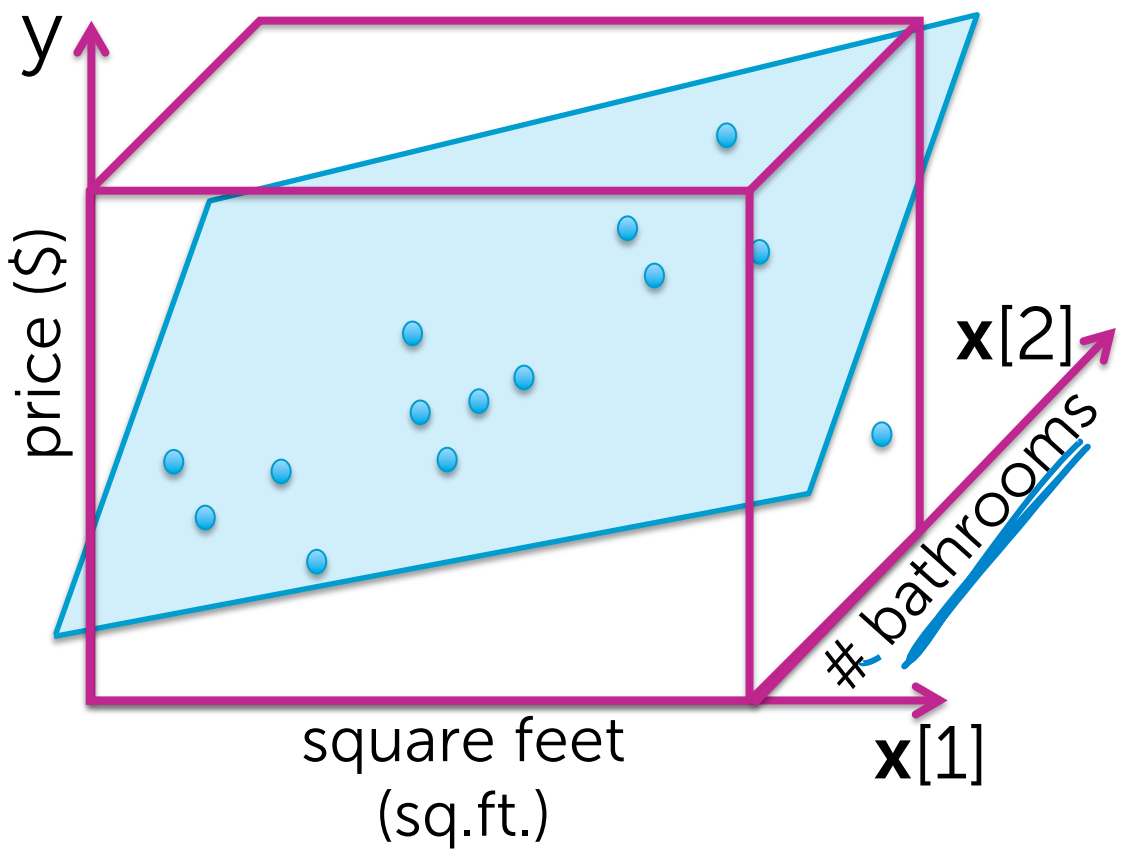
$$= -2\sum_{i=1}^{N} h_j(x_i)(y_i - h(x_i)^T w)$$

Update to j^{th} feature weight:

$$w_j^{(t+1)} \leftarrow w_j^{(t)} - \eta\left(-2\sum_{i=1}^{N} h_j(x_i)(y_i - \underbrace{h^T(x_i)w^{(t)}}_{\hat{y}_i(w^{(t)})})\right)$$

©2015 Emily Fox & Carlos Guestrin

Interpreting elementwise



Update to j^{th} feature weight:

bathrooms

bath for house i

$$w_j^{(t+1)} \leftarrow w_j^{(t)} + 2\eta \sum_{i=1}^{N} h_j(\mathbf{x}_i)(y_i - \hat{y}_i(\mathbf{w}^{(t)}))$$

If underestimating impact of # bath ($\hat{w}_j^{(t)}$ is too small)

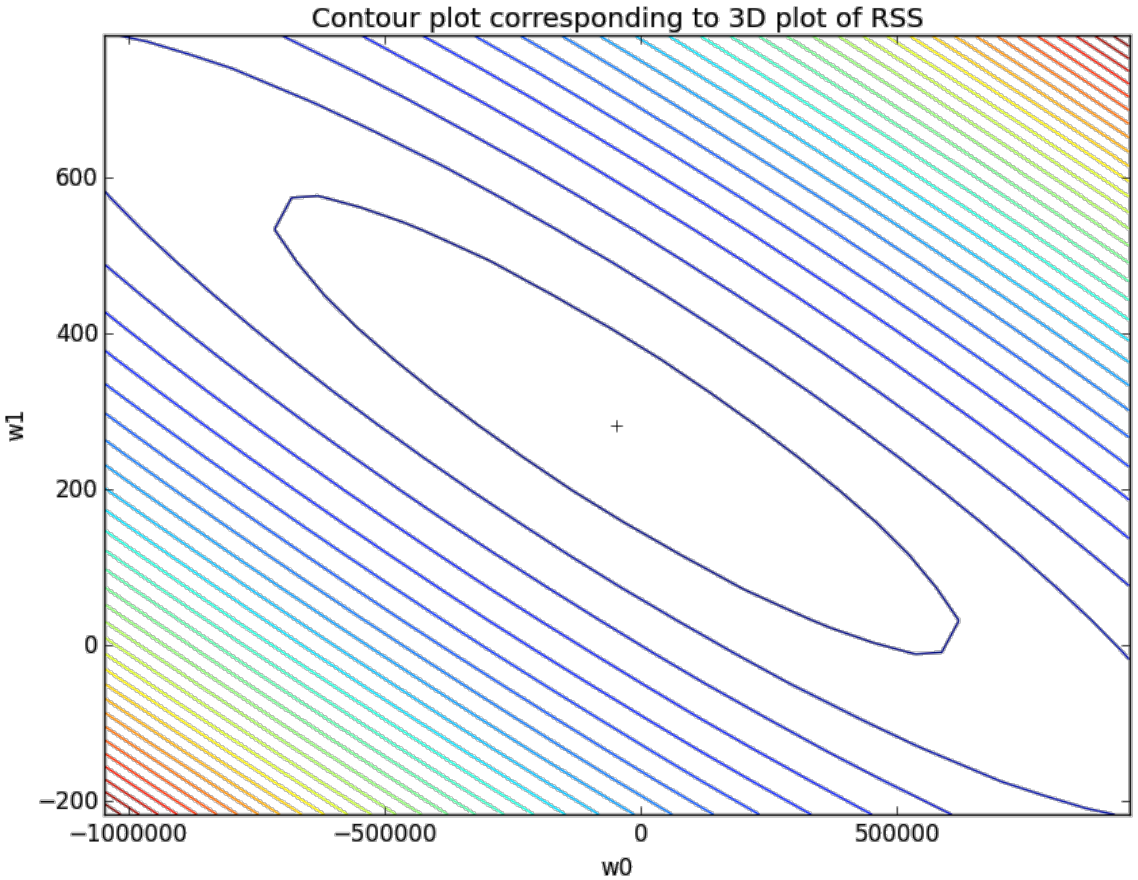
then $(y_i - \hat{y}_i(w^{(t)}))$ on average

weighted by # bath will be positive

$\Rightarrow w_j^{(t+1)} > w_j^{(t)}$ (increase)

©2015 Emily Fox & Carlos Guestrin

Summary of gradient descent for multiple regression



init $\mathbf{w}^{(1)}=0$ (or randomly, or smartly), $t=1$

tolerance

while $||\nabla RSS(\mathbf{w}^{(t)})|| > \varepsilon$

$\sqrt{\text{partial}[0]^2 + \ldots + \text{partial}[D]^2}$

for $j=0,\ldots,D$

$$\text{partial}[j] = -2\sum_{i=1}^{N} h_j(\mathbf{x}_i)(y_i - \hat{y}_i(\mathbf{w}^{(t)}))$$

$$\mathbf{w}_j^{(t+1)} \leftarrow \mathbf{w}_j^{(t)} - \eta\ \text{partial}[j]$$

$$t \leftarrow t + 1$$

©2015 Emily Fox & Carlos Guestrin