Linear Regression

Cornelia Caragea

Department of Computer Science and Engineering
University of North Texas

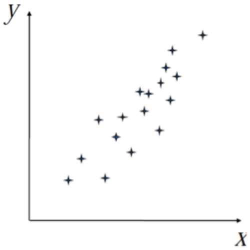
Acknowledgments: Piyush Rai, Andrew Ng

June 21, 2016

Linear Regression with One Variable

Linear Regression: One-Dimensional Case

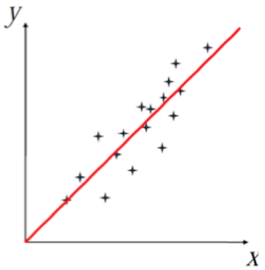
Regression: we observe a real-valued input x and we wish to use x to predict the value of a real-valued target t (or y).



- **Given:** a set of N input-target pairs
  - The inputs (x) and targets (y) are one dimensional scalars
- **Goal:** Model the relationship between x and y

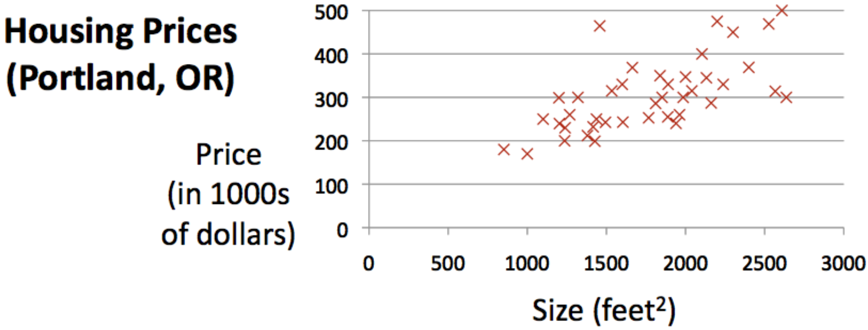
Linear Regression: One-Dimensional Case

- **Assumption:** the relationship between x and y is linear
- Linear relationship - defined by a straight line with parameter w
- Equation of the straight line: $y = wx$ ($y = w_0 + w_1 x_1$)

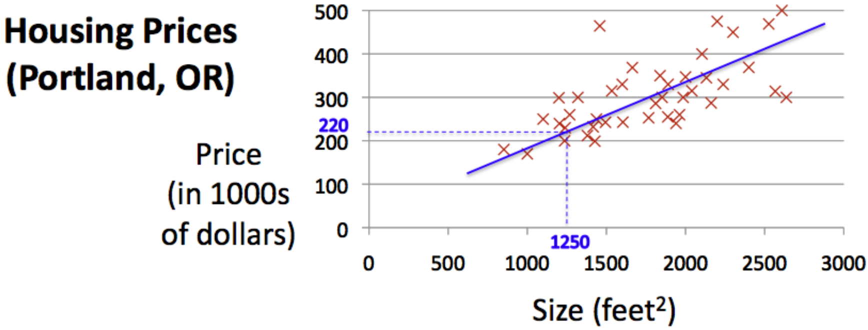


- The line may not fit the data *exactly*
- But we can try making the line a reasonable approximation

Example - House Price Prediction



Example - House Price Prediction



Data Representation

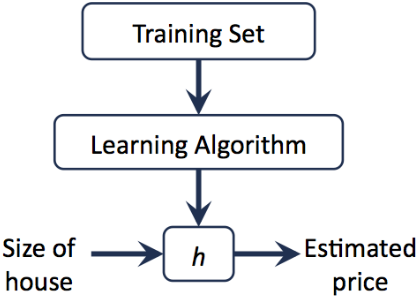
Training set of housing prices (Portland, OR):

Size in feet$^2 (x)$	**Price** (\$) **in** 1000**'s** (y)
2104	460
1416	232
1534	315
852	178
...	...

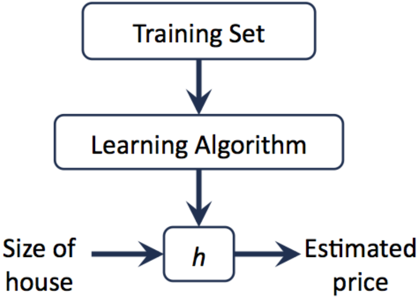
Notation:

- N = number of training examples
- x's = "input" variable / features
- y's = "output" variable / "target" variable
- (x, y) - one training example
- $(x^{(i)}, y^{(i)})$ - the i^{th} training example

Model Representation



Model Representation



How do we represent h?

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

Regression: estimating $h_w(x)$ of x that minimizes a cost function.

Model Representation

Training set of housing prices (Portland, OR):

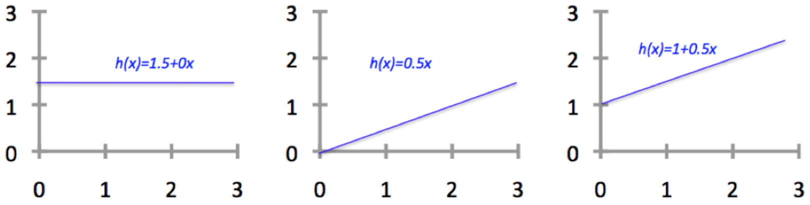
Size in feet$^2 (x)$	**Price (\$) in** 1000's (y)
2104	460
1416	232
1534	315
852	178
...	...

Hypothesis: $h_w(x) = w_0 + w_1 x$

- Model parameters: w_i's
- How to choose w_i's?

Hypothesis

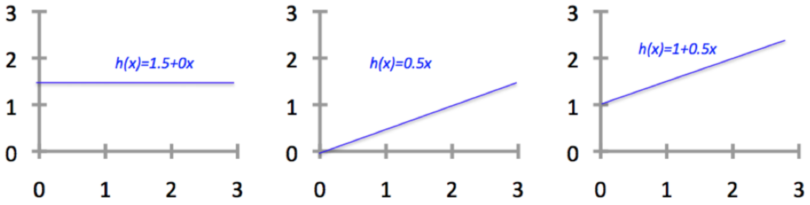
$$h_w(x) = w_0 + w_1 x$$



$w_0 = 1.5$	$w_0 = 0$	$w_0 = 1$
$w_1 = 0$	$w_1 = 0.5$	$w_1 = 0.5$

Hypothesis

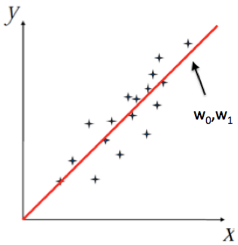
$$h_w(x) = w_0 + w_1 x$$

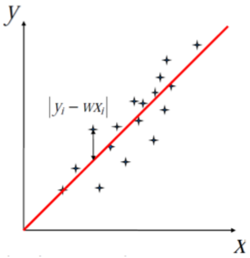


$w_0 = 1.5$	$w_0 = 0$	$w_0 = 1$
$w_1 = 0$	$w_1 = 0.5$	$w_1 = 0.5$

How to choose a hypothesis?

Intuition

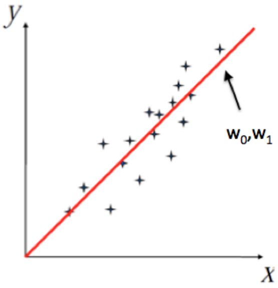




Idea: Choose w_0, w_1 s.t. $h_w(x)$ is close to y for our training examples (x, y).

- Error for the pair (x_i, y_i) pair: $e_i = y_i - wx_i$
- The total squared error: $E = \frac{1}{2N}\sum_{i=1}^{N} e_i^2 = \frac{1}{2N}\sum_{i=1}^{N}(y_i - wx_i)^2$
- The best fitting line is defined by w minimizing the total error E

The Cost Function



Idea: Choose w_0, w_1 s.t. $h_w(x)$ is close to y for our training examples (x, y).

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

where

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

$$E(w_0, w_1) = \frac{1}{2N} \sum_{i=1}^{N} (h_w(x_i) - y_i)^2,$$

Hence,

$$\min_{w_0,w_1} E(w_0, w_1),$$

Cost Function Intuition

Simplified:

Hypothesis:

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

Parameters:

$$w_0, w_1$$

Cost Function:

$$E(w_0, w_1) = \frac{1}{2N} \sum_{i=1}^{N} (h_w(x_i) - y_i)^2 ,$$

Goal:

$$\min_{w_0, w_1} E(w_0, w_1),$$

Hypothesis:

$$h_w(x) = w_1 x, \boxed{w_0 = 0}$$

Parameters:

$$w_1$$

Cost Function:

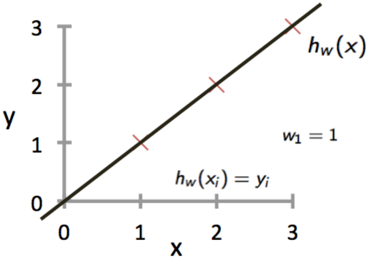
$$E(w_1) = \frac{1}{2N} \sum_{i=1}^{N} (h_w(x_i) - y_i)^2 ,$$

Goal:

$$\min_{w_1} E(w_1),$$

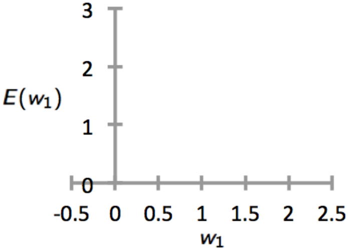
$h_w(x)$

(for fixed w_1, this is a function of x)



$E(w_1)$

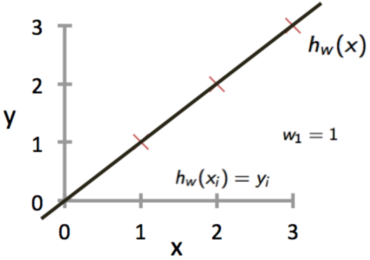
(function of the parameter w_1)



$$
\begin{aligned}
E(w_1) &= \frac{1}{2N}\sum_{i=1}^{N}(h_w(x_i) - y_i)^2 = \frac{1}{2N}\sum_{i=1}^{N}(w_1 x_i - y_i)^2 \\
&= \frac{1}{2N}(0^2 + 0^2 + 0^2) = 0
\end{aligned}
$$

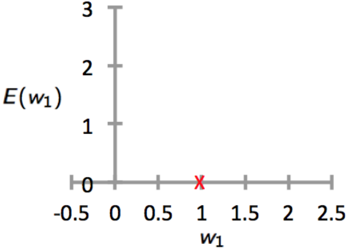
$h_w(x)$

(for fixed w_1, this is a function of x)



$E(w_1)$

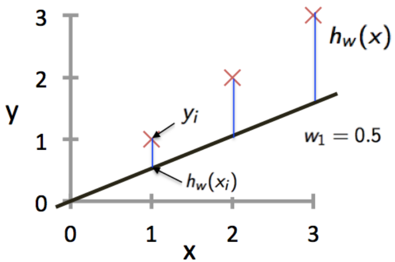
(function of the parameter w_1)



$$
\begin{aligned}
E(w_1) &= \frac{1}{2N}\sum_{i=1}^{N}(h_w(x_i) - y_i)^2 = \frac{1}{2N}\sum_{i=1}^{N}(w_1 x_i - y_i)^2 \\
&= \frac{1}{2N}(0^2 + 0^2 + 0^2) = 0
\end{aligned}
$$

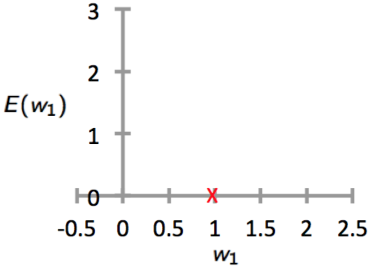
$h_w(x)$

(for fixed w_1, this is a function of x)



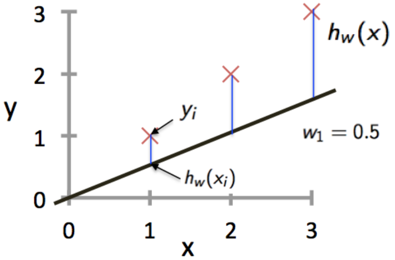
$E(w_1)$

(function of the parameter w_1)

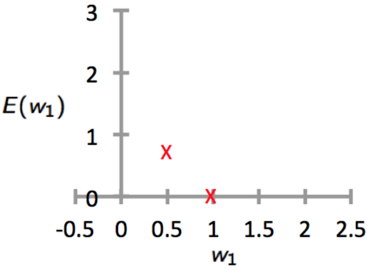


$$
\begin{aligned}
E(0.5) &= \frac{1}{2N}\left[(0.5-1)^2+(1-2)^2+(1.5-3)^2\right] \\
&= \frac{1}{2\times 3}(3.5) \approx 0.58
\end{aligned}
$$

$h_w(x)$
(for fixed w_1, this is a function of x)



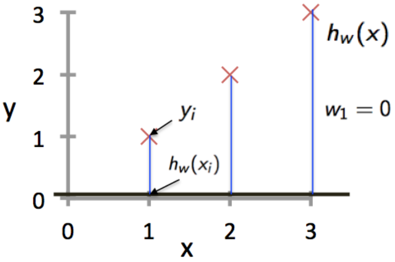
$E(w_1)$
(function of the parameter w_1)



$$\begin{aligned} E(0.5) &= \frac{1}{2N}\left[(0.5-1)^2 + (1-2)^2 + (1.5-3)^2\right] \\ &= \frac{1}{2 \times 3}(3.5) \approx 0.58 \end{aligned}$$

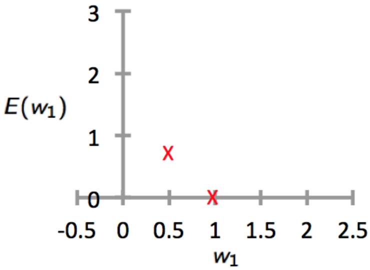
$h_w(x)$

(for fixed w_1, this is a function of x)



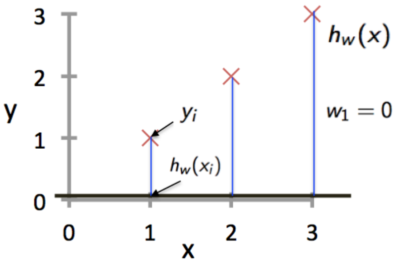
$E(w_1)$

(function of the parameter w_1)

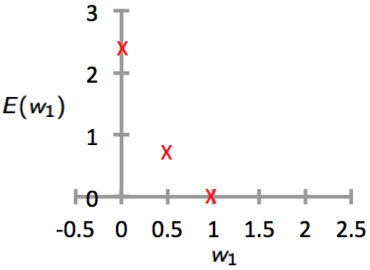


$$
\begin{aligned}
E(0) &= \frac{1}{2N}\left[1^2 + 2^2 + 3^2\right] \\
&= \frac{1}{2 \times 3}(14) \approx 2.3
\end{aligned}
$$

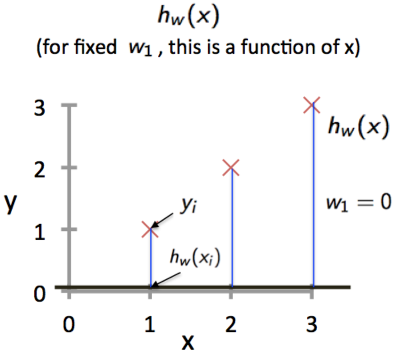
$h_w(x)$
(for fixed w_1, this is a function of x)

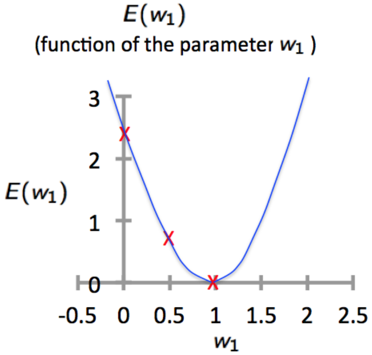


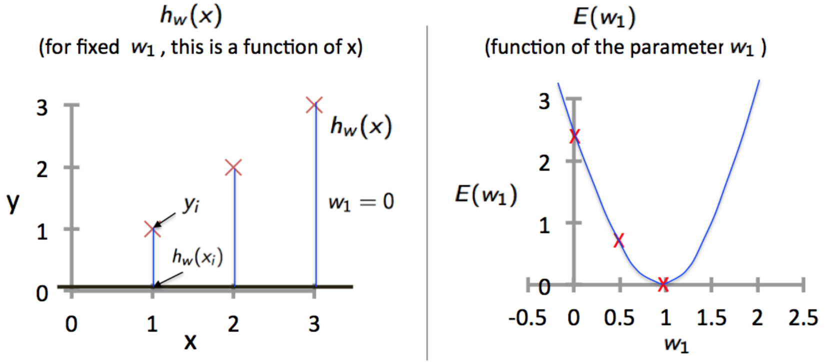
$E(w_1)$
(function of the parameter w_1)



$$
\begin{aligned}
E(0) &= \frac{1}{2N}\left[1^2 + 2^2 + 3^2\right] \\
&= \frac{1}{2 \times 3}(14) \approx 2.3
\end{aligned}
$$


$h_w(x)$
(for fixed w_1, this is a function of x)
3
2
1
0
y
y_i
$h_w(x_i)$
$h_w(x)$
$w_1 = 0$
0
1
2
3
x


$E(w_1)$
(function of the parameter w_1)
$E(w_1)$
3
2
1
0
-0.5
0
0.5
1
1.5
2
2.5
w_1



How to Minimize E(w)?

Gradient Descent

Given: some function $E(w_0, w_1)$
Want: $\min_{w_0, w_1} E(w_0, w_1)$

Outline:

- Start with some w_0, w_1
- Keep changing w_0, w_1 to reduce $E(w_0, w_1)$ until hopefully we end up at a minimum

Gradient Descent Algorithm

repeat until convergence {

$w_j := w_j - \alpha \frac{\partial}{\partial w_j} E(w_0, w_1)$ (for $j = 0$ and $j = 1$)

}

- α - the learning rate
- $\frac{\partial}{\partial w_j} E(w_0, w_1)$ - derivative of E.

Correct: Simultaneous update

$\text{temp0} := w_0 - \alpha \frac{\partial}{\partial w_0} E(w_0, w_1)$

$\text{temp1} := w_1 - \alpha \frac{\partial}{\partial w_1} E(w_0, w_1)$

$w_0 := \text{temp0}$

$w_1 := \text{temp1}$

Incorrect

$\text{temp0} := w_0 - \alpha \frac{\partial}{\partial w_0} E(w_0, w_1)$

$w_0 := \text{temp0}$

$\text{temp1} := w_1 - \alpha \frac{\partial}{\partial w_1} E(w_0, w_1)$

$w_1 := \text{temp1}$

Gradient Descent Algorithm

repeat until convergence {

$$w_j := w_j - \alpha \frac{\partial}{\partial w_j} E(w_0, w_1) \text{ (for } j = 0 \text{ and } j = 1\text{)}$$

}

- α - the learning rate
- $\frac{\partial}{\partial w_j} E(w_0, w_1)$ - derivative of E.

Correct: Simultaneous update

$temp0 := w_0 - \alpha \frac{\partial}{\partial w_0} E(w_0, w_1)$

$temp1 := w_1 - \alpha \frac{\partial}{\partial w_1} E(w_0, w_1)$

$w_0 := temp0$

$w_1 := temp1$

Incorrect

$temp0 := w_0 - \alpha \frac{\partial}{\partial w_0} E(w_0, w_1)$

$w_0 := temp0$

$temp1 := w_1 - \alpha \frac{\partial}{\partial w_1} E(w_0, w_1)$

$w_1 := temp1$

- If α is too small, gradient descent can be slow.
- If α is too large, gradient descent can overshoot the minimum. It may fail to converge, or even diverge.

Gradient Descent for Linear Regression

Gradient Descent Algorithm for the Linear Regression Model

repeat until convergence {
$w_j := w_j - \alpha \frac{\partial}{\partial w_j} E(w_0, w_1)$
(for $j = 0$ and $j = 1$)
}

where

$h_w(x) = w_0 + w_1 x$
$E(w_0, w_1) = \frac{1}{2N} \sum_{i=1}^{N} (h_w(x_i) - y_i)^2$

$$\frac{\partial}{\partial w_j} E(w_0, w_1) = \frac{\partial}{\partial w_j} \frac{1}{2N} \sum_{i=1}^{N} (h_w(x_i) - y_i)^2$$
$$= \frac{\partial}{\partial w_j} \frac{1}{2N} \sum_{i=1}^{N} (w_0 + w_1 x_i - y_i)^2$$

$$j = 0 : \frac{\partial}{\partial w_0} E(w_0, w_1) = \frac{1}{N} \sum_{i=1}^{N} (h_w(x_i) - y_i)$$
$$j = 1 : \frac{\partial}{\partial w_1} E(w_0, w_1) = \frac{1}{N} \sum_{i=1}^{N} (h_w(x_i) - y_i)\, x_i$$

Gradient Descent for Linear Regression

Gradient Descent Algorithm for the Linear Regression Model

repeat until convergence {

$w_0 := w_0 - \alpha \frac{1}{N} \sum_{i=1}^{N} (h_w(x_i) - y_i)$

$w_1 := w_1 - \alpha \frac{1}{N} \sum_{i=1}^{N} (h_w(x_i) - y_i)\, x_i$

}

Update w_0 and w_1 simultaneously.

Summary

Linear Regression Model with One Variable

- Model Representation
- How to choose a hypothesis?
- Cost Function
- Gradient Descent