UNIVARIATE LINEAR REGRESSION

Prof. Nielsen Rechia
nielsen.machado@uniritter.edu.br

Univariate Linear Regression

Paradigmas	Supervisionado	Não-supervisionado
Tarefas	Classificação	Análise associativa
	Regressão	Agrupamento (clustering)
	Outros	Redução de dimensionalidade
		Outros

7 tarefas comuns de aprendizado de máquina: http://vitalflux.com/7-common-machine-learning-tasks-related-methods/

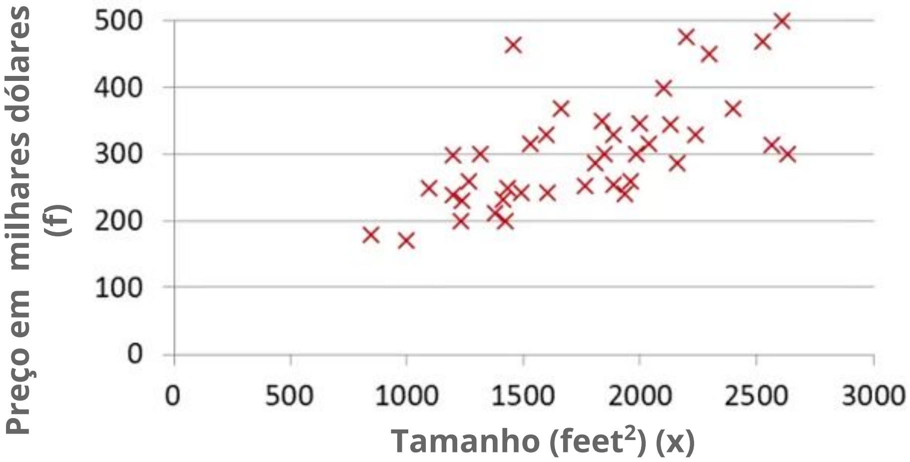
Regressão Linear

O objetivo da regressão é aproximar um valor para um objeto ainda não conhecido, com base em objetos conhecidos.

A Regressão Linear é diferente da classificação, onde o objetivo é aproximar uma categoria ao objeto.

Regressão Linear

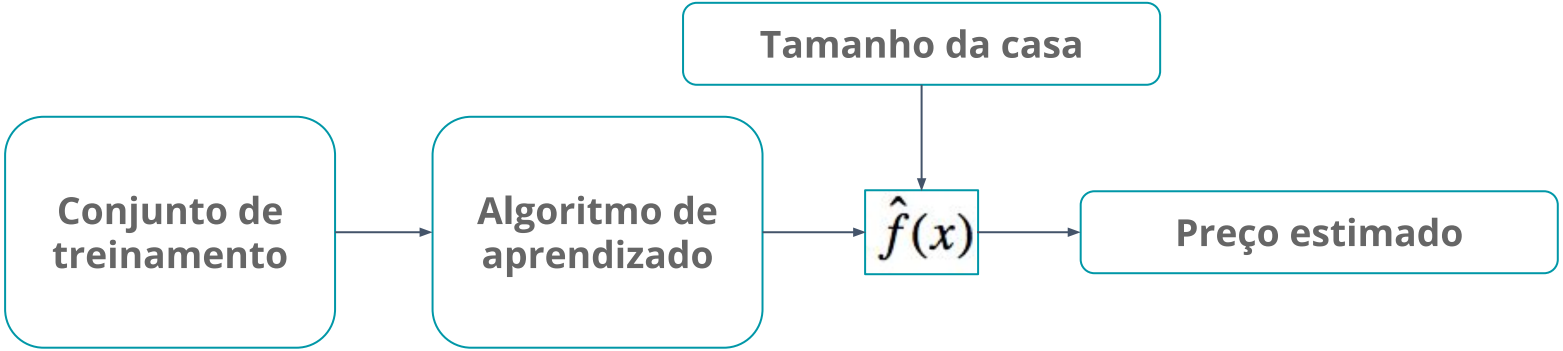
De acordo com o tamanho e o preço de casas, qual o preço de uma nova casa a venda dado o seu tamanho?



Regressão Linear

Utilizamos regressão linear (univariada = apenas um atributo), onde o modelo é dada por:

$$\hat{f}(x) = \theta_0 + \theta_1 x$$



Exercícios

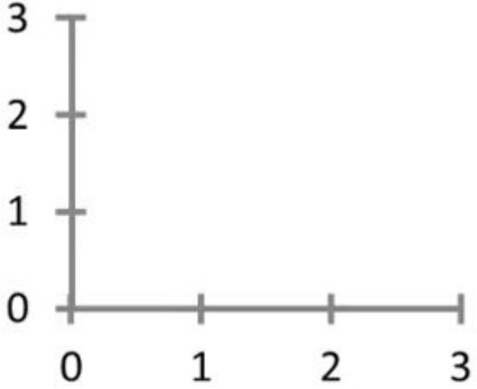
Fazer uma projeção para cada um dos parâmetros (i.e. pesos) a seguir para cada um dos valores de x

$$\hat{f}(x) = \theta_0 + \theta_1 x$$

$\theta_0 = 1.5$
$\theta_1 = 0$

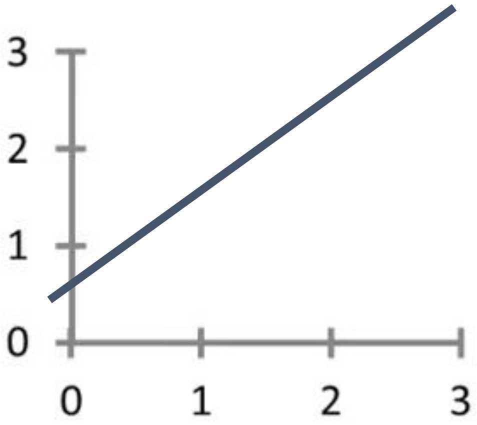
$\theta_0 = 0$
$\theta_1 = 0.5$

$\theta_0 = 1$
$\theta_1 = 0.5$



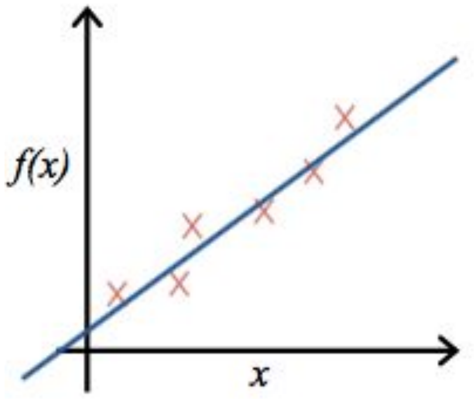
Exercícios

Quais os valores de Θ_0 e Θ_1 ?



Regressão Linear

A Idéia é definir os valores de Θ_0 e Θ_1 visando minimizar a função de custo do aprendizado.



Regressão Linear

Temos um modelo: $\hat{f}(x) = \theta_0 + \theta_1 x$

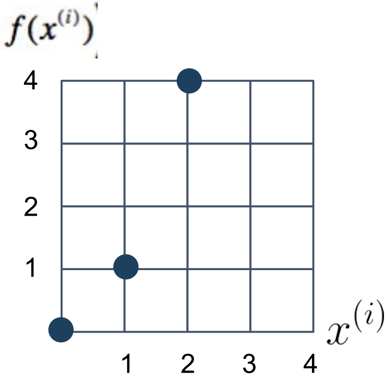
Alguns parâmetros: Θ_0 e Θ_1

Uma função de custo: $J(\theta_0, \theta_1) = \frac{1}{2N} \sum_{i=1}^{N} \left(\hat{f}(x^{(i)}) - f(x^{(i)}) \right)^2$

Um objetivo: $\min_{\theta_o, \theta_1} J(\theta_0, \theta_1)$

Exercícios

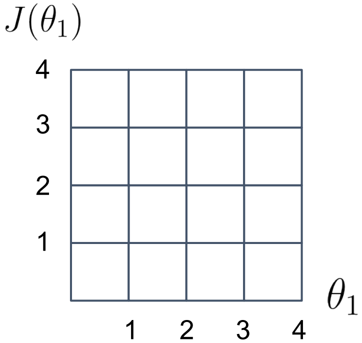
Dado o seguinte conjunto de dados e os parâmetros, calcule o custo das predições:



$\Theta_0 = 0,\ \Theta_1 = 0$
$\Theta_0 = 0,\ \Theta_1 = 0.5$
$\Theta_0 = 0,\ \Theta_1 = 1$

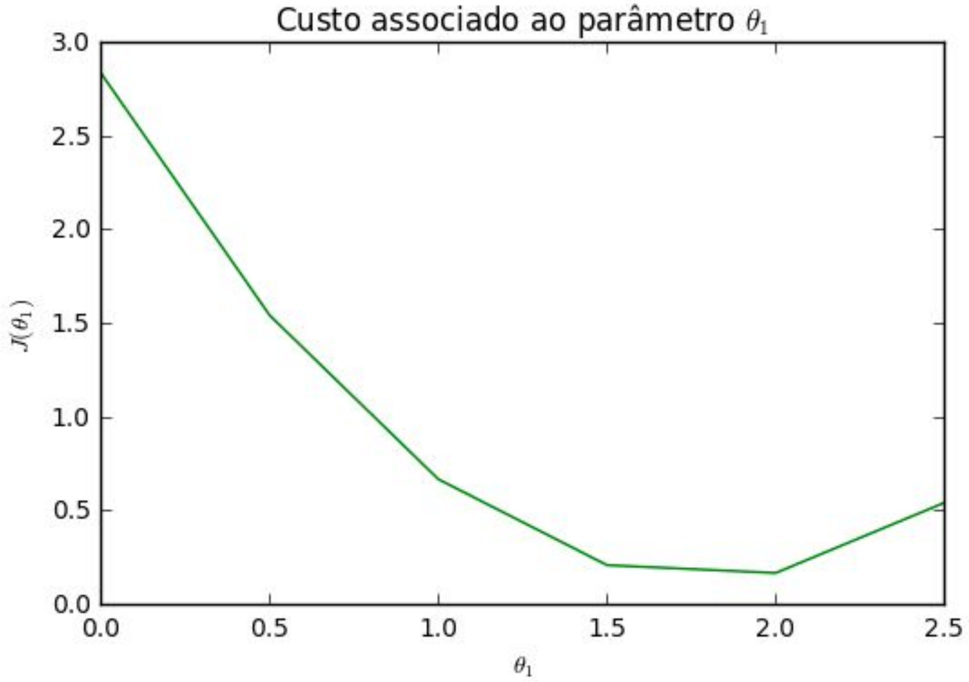
$\Theta_0 = 0,\ \Theta_1 = 1.5$
$\Theta_0 = 0,\ \Theta_1 = 2$
$\Theta_0 = 0,\ \Theta_1 = 2.5$

Preencha o gráficos a seguir:



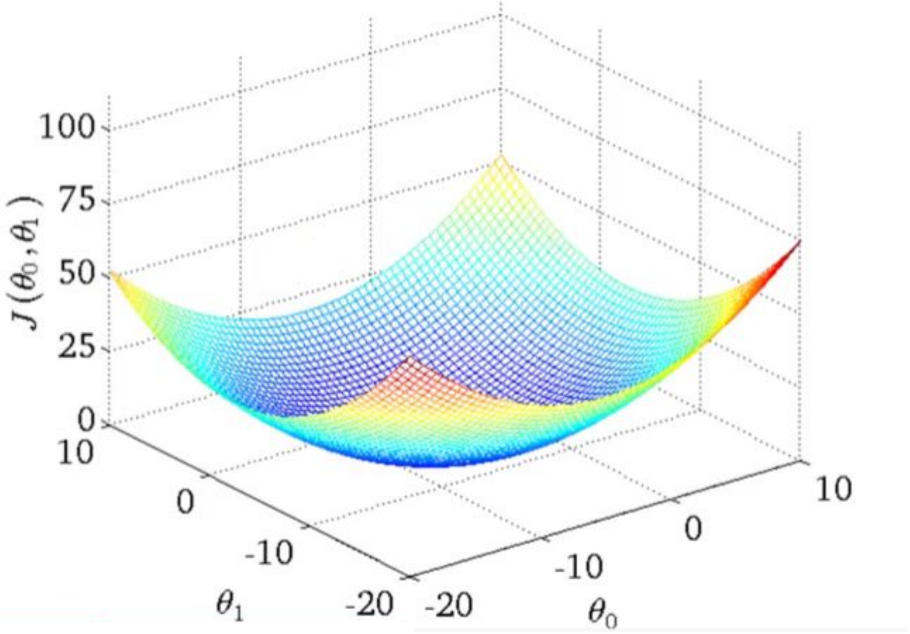
Regressão Linear

A função de custo para um parâmetro é convexa



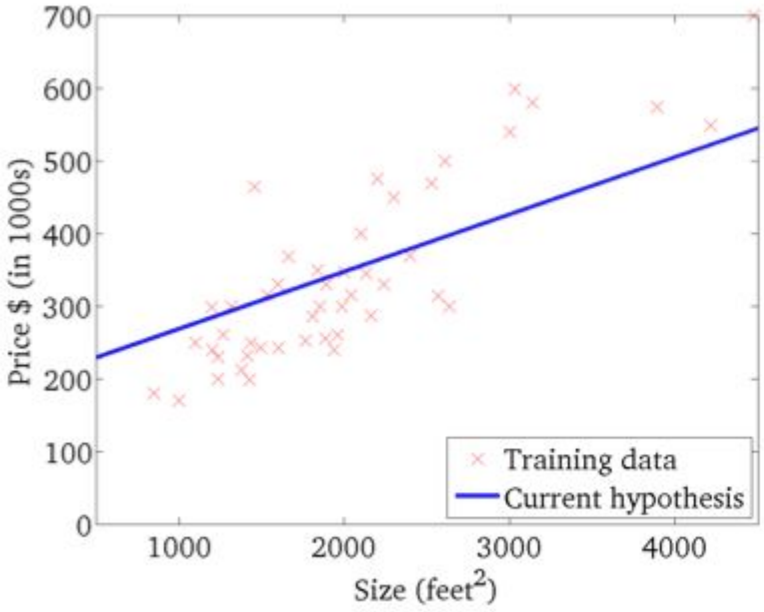
Regressão Linear

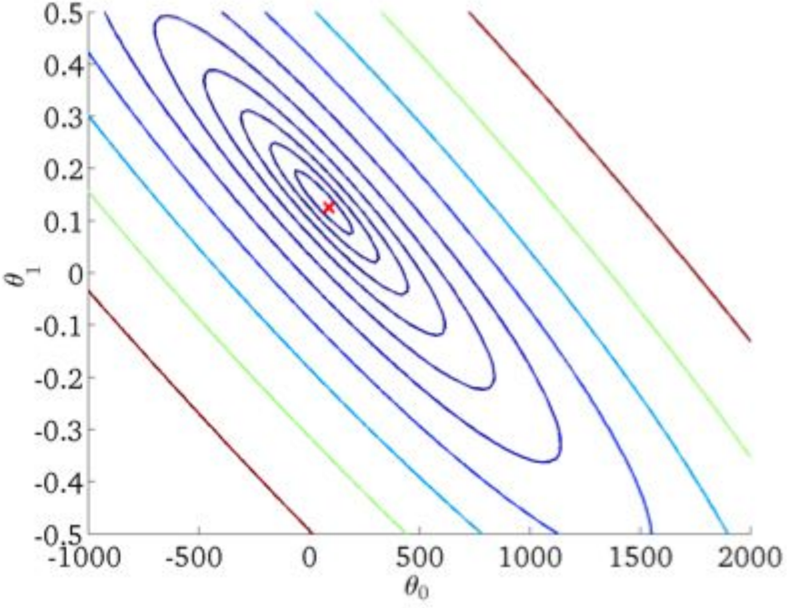
A função de custo também é convexa para dois parâmetros



Regressão Linear

Exemplo





Regressão Linear

Para iniciar devemos escolher arbitrariamente os valores dos parâmetros

Para minimizar a função de custo devemos atualizar os pesos dos parâmetros

Os valores devem ser alterados continuamente até um determinado critério de parada (convergência)

Regressão Linear

O antigo valor do j-ésimo parâmetro, menos

vezes a taxa de aprendizado

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

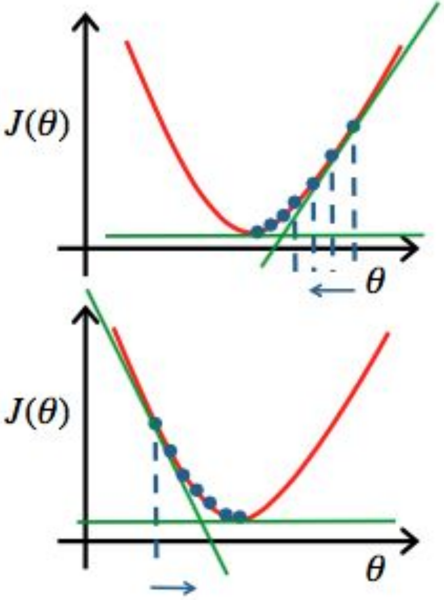
O j-ésimo parâmetro recebe

A derivada da função de custo em relação ao j-ésimo parâmetro

Regressão Linear

Entendendo a derivada

A derivada parcial indica como a função de custo se altera quando um parâmetro têm seu valor modificado



com apenas um parâmetro, $\theta \in$
$J(\theta)$

$$\theta := \theta - \alpha \boxed{\frac{d}{d\theta} J(\theta)}$$

$$\theta := \theta - \alpha \text{ (num positivo)}$$

$$\theta := \theta - \alpha \text{ (num negativo)}$$

Regressão Linear

A derivada parcial nos mostra apenas a direção em que o parâmetro deve ser modificado.

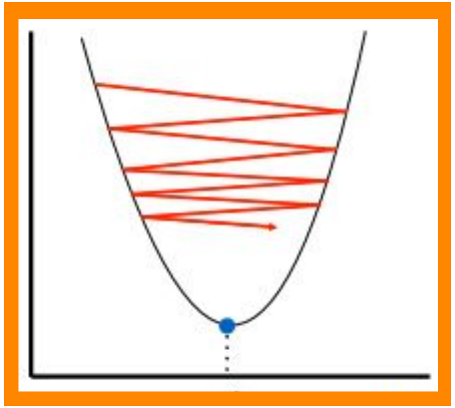
Não nos mostra o tamanho correto do passo que deve ser obtido com a taxa de aprendizado

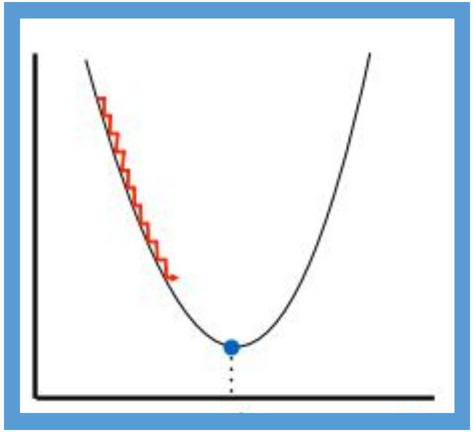
Regressão Linear

Se a taxa de aprendizado for muito alta, podemos passar “por cima” do ótimo global

Se a taxa de aprendizado for muito baixa, podemos demorar muito para convergir

No free launch!





Regressão Linear

- Então, para fazer a atualização de pesos, utilizamos o gradiente descendente
- Gradiente é como chamamos o grupo de derivadas de diversos parâmetros
- Utilizamos uma taxa de aprendizado para ditar o passo em torno de melhores resultados

Regressão Linear

A atualização dos pesos deve ser simultânea!

Não atualize o valor do parâmetro j + 1, com base no valor atualizado do parâmetro j

Em vez disso, calcule antes todos os novos valores de parâmetros, e só então faça as atribuições

$$
\begin{aligned}
\text{temp0} &:= \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) \\
\text{temp1} &:= \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) \\
\theta_0 &:= \text{temp0} \\
\theta_1 &:= \text{temp1}
\end{aligned}
$$

Implementação correta

Regressão Linear

Para a função de custo da regressão linear, a atualização de pesos (com derivadas calculadas) é:

$$\theta_0 := \theta_0 - \alpha \frac{1}{N} \sum_{i=1}^{N} (h_\theta(x^{(i)}) - y^{(i)})$$

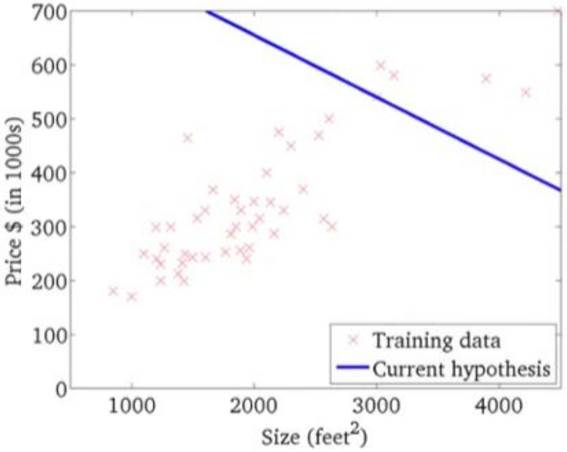
$$\theta_1 := \theta_1 - \alpha \frac{1}{N} \sum_{i=1}^{N} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x^{(i)}$$

Regressão Linear

Gradiente Descendente

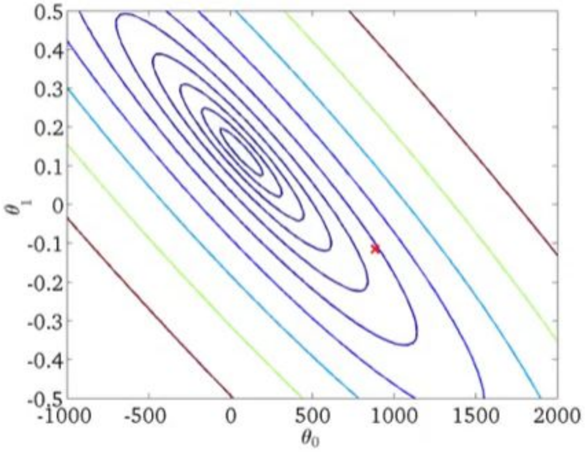
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)

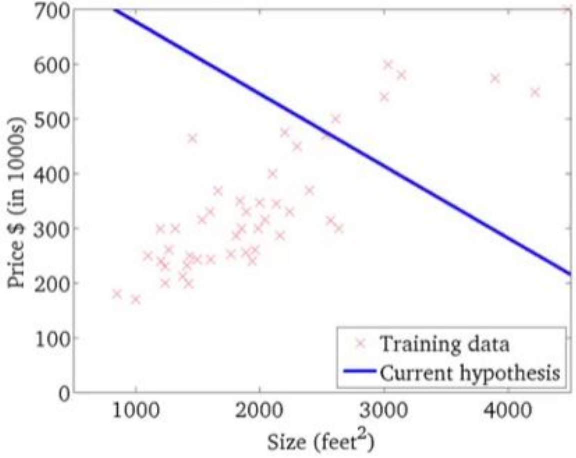


Regressão Linear

Gradiente Descendente

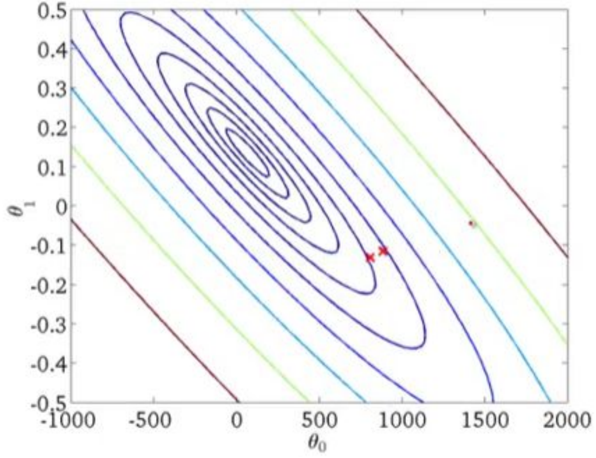
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)

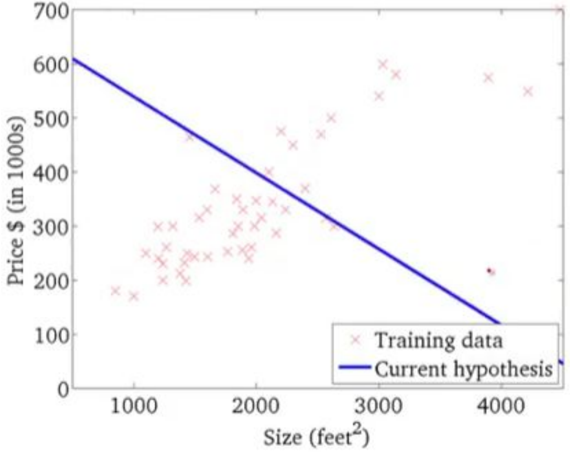


Regressão Linear

Gradiente Descendente

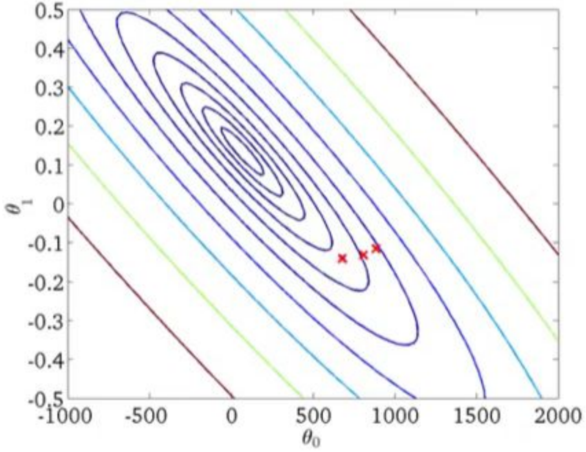
$$h_\theta(x)$$

(for fixed θ_0, θ_1, this is a function of x)



$$J(\theta_0, \theta_1)$$

(function of the parameters θ_0, θ_1)

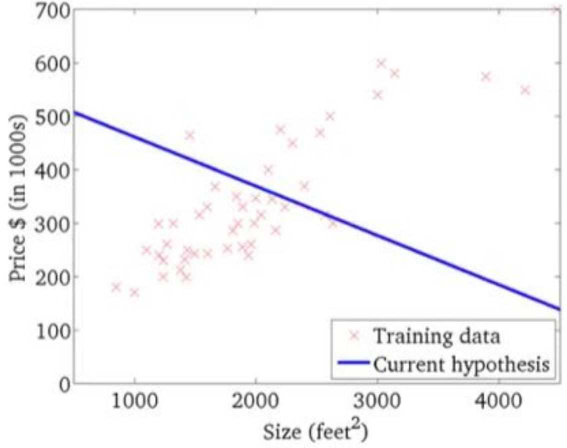


Regressão Linear

Gradiente Descendente

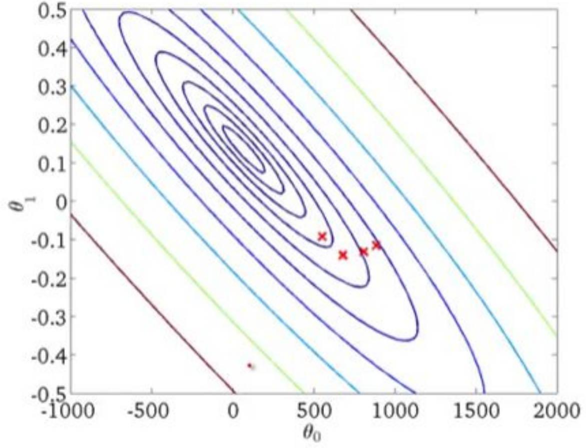
$$h_\theta(x)$$

(for fixed θ_0, θ_1, this is a function of x)



$$J(\theta_0, \theta_1)$$

(function of the parameters θ_0, θ_1)

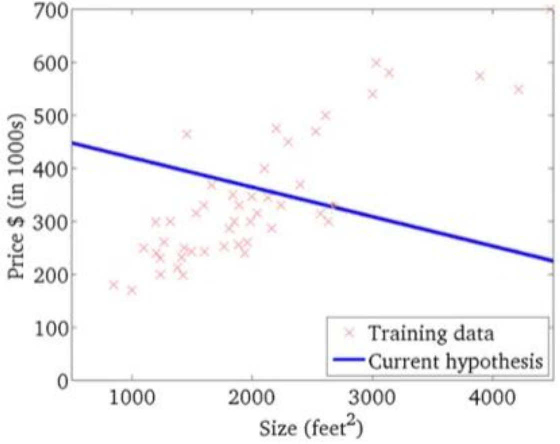


Regressão Linear

Gradiente Descendente

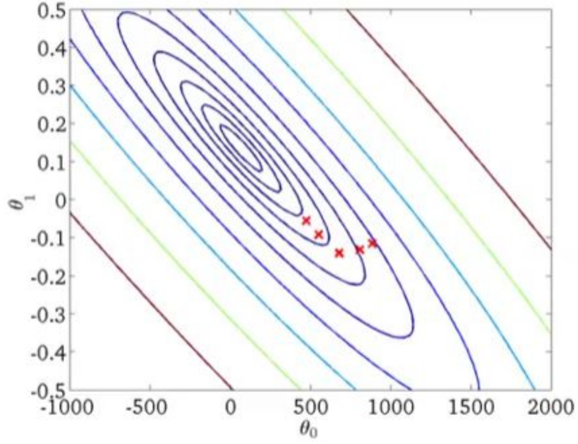
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)



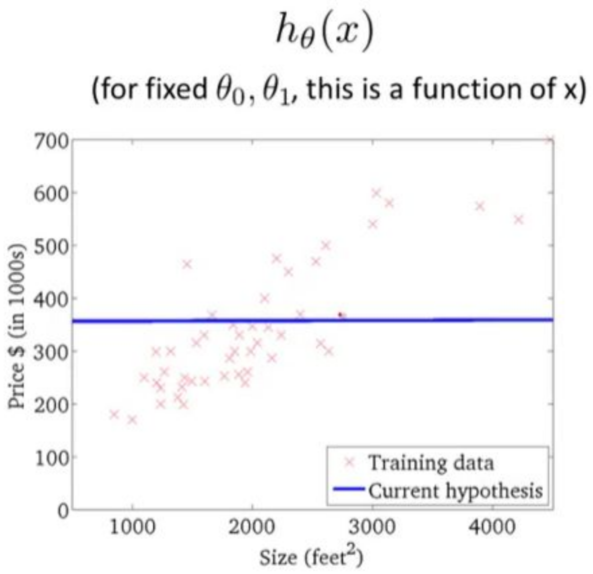
$J(\theta_0, \theta_1)$

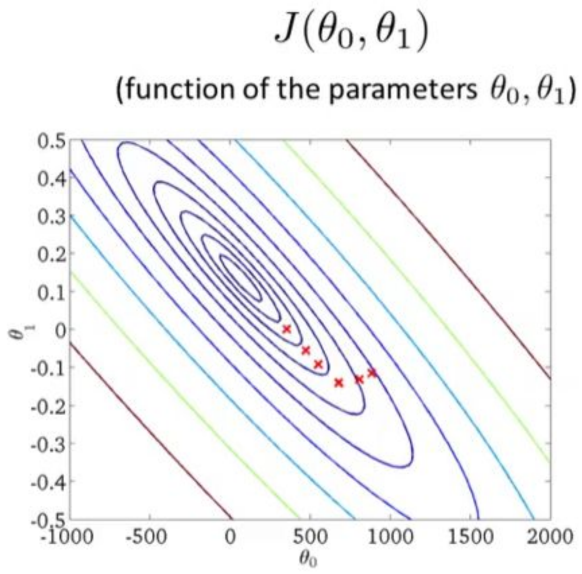
(function of the parameters θ_0, θ_1)



Regressão Linear

Gradiente Descendente



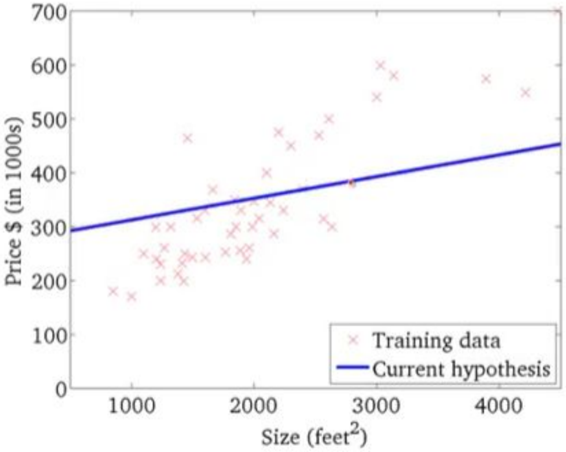


Regressão Linear

Gradiente Descendente

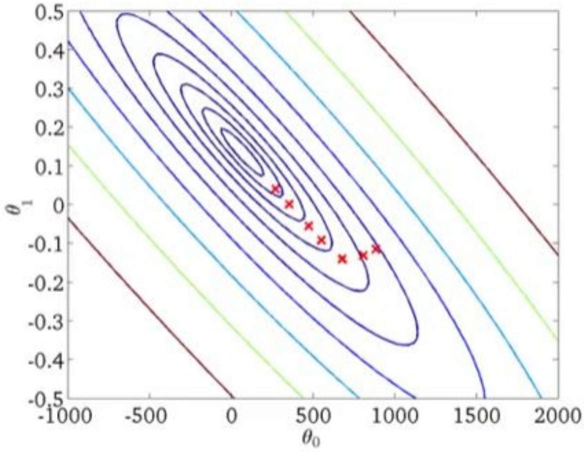
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)

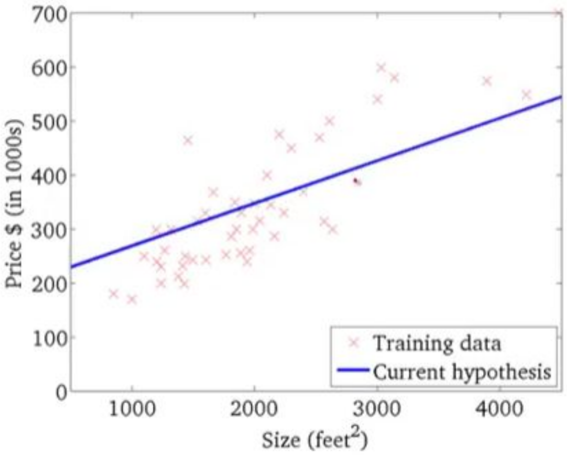


Regressão Linear

Gradiente Descendente

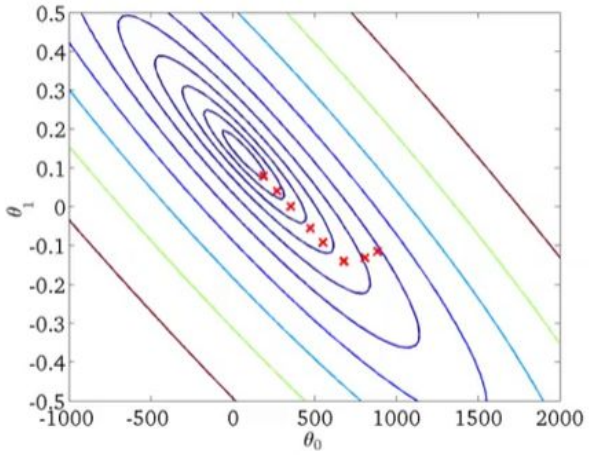
$h_\theta(x)$

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)

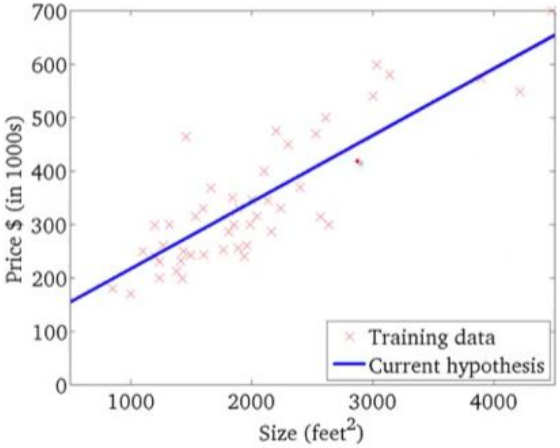


Regressão Linear

Gradiente Descendente

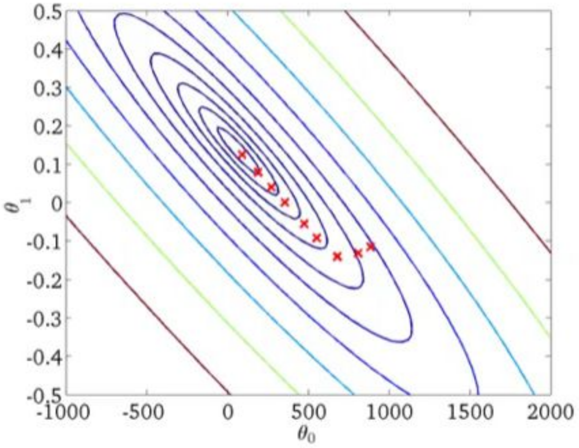
$$h_\theta(x)$$

(for fixed θ_0, θ_1, this is a function of x)



$$J(\theta_0, \theta_1)$$

(function of the parameters θ_0, θ_1)



Exercícios

Dados os seguintes parâmetros, realize a atualização de pesos:

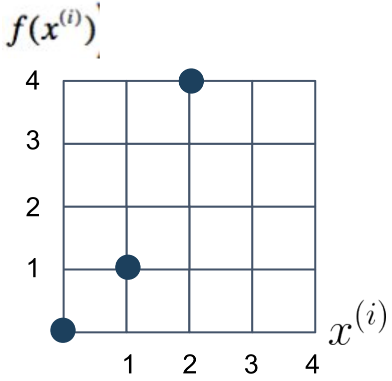
$\alpha = 0.5$

$\Theta_0 = 0.1$

$\Theta_1 = 1$

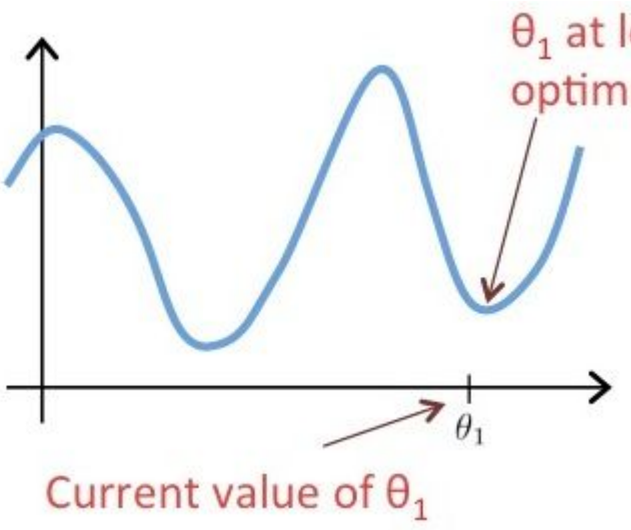
X = [0, 1, 2]

fx = [0, 1, 4]



Regressão Linear

A função de custo é sempre convexa em um problema de regressão linear



Regressão Linear

Para realizar a atualização dos pesos dos parâmetros é necessário passar um batch (lote) pelo modelo

- Esse batch pode ser de uma instância ou de todo conjunto de treino
- Para otimizar, normalmente utilizamos quantidades multiplas de 8
- O processo de passagem do batch pode ser estocástico (variáveis aleatórias)

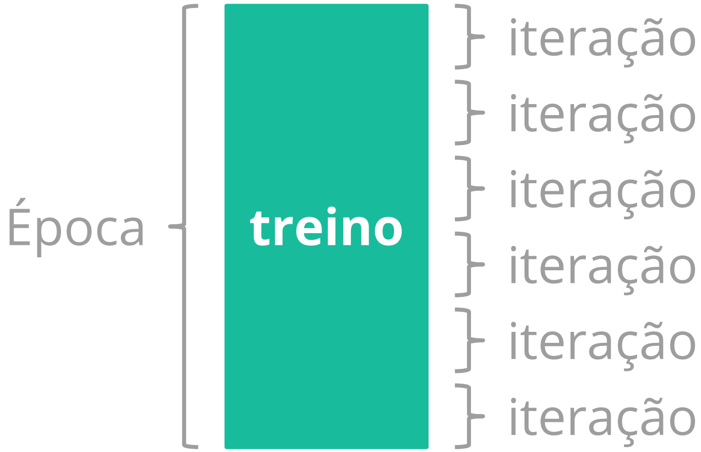
O Gradiente Descendente pode convergir mesmo que a taxa de aprendizado seja fixa.

- Para isso a taxa de aprendizado não pode ser muito elevada
- Além disso, a taxa de aprendizado pode ser decrementada ao longo do treinamento

Regressão Linear

Passada de um batch: iteração

Passada do conjunto de treino: época



Exercício prático

Implementar uma regressão linear para prever qual o lucro, tendo como base o tamanho da população de um determinado local, com o dataset Regressão Univariada -> ex1data1.csv"

- Defina uma % do conjunto de dados para treino e teste (sugestão 50% 50%)
- Inicie os parâmetros Θ_0 e Θ_1 de forma randômica
- Defina a taxa de aprendizado inicial
- Atualize os pesos
- Realize o processo até atingir convergência (diferença no custo de uma iteração para a outra não é significativa)

Conclusão

Leitura recomendada:

Apêndice D de Introduction to Data Mining