Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina

Regressão

Prof. Tiago A. Almeida

Modelos preditivos

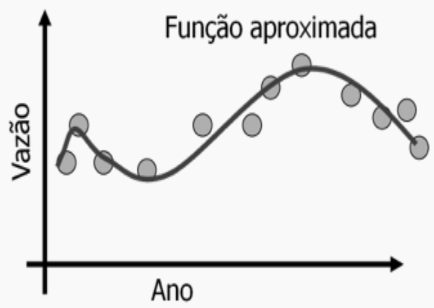
- Definição formal: dado conjunto de observações
- **D** = {($\mathbf{x}^{(i)}$, $f(\mathbf{x}^{(i)})$), i = 1, ..., m}
  - f representa uma função desconhecida: função objetivo
    - Mapeia entradas em saídas correspondentes
  - Algoritmo preditivo aprende aproximação $\hat{f}$
    - Que permite estimar valor de f para novos objetos **x**

Regressão

$$y^{(i)} = f(\mathbf{x}^{(i)}) \in \Re$$

Regressão

- Ilustração das tarefa:

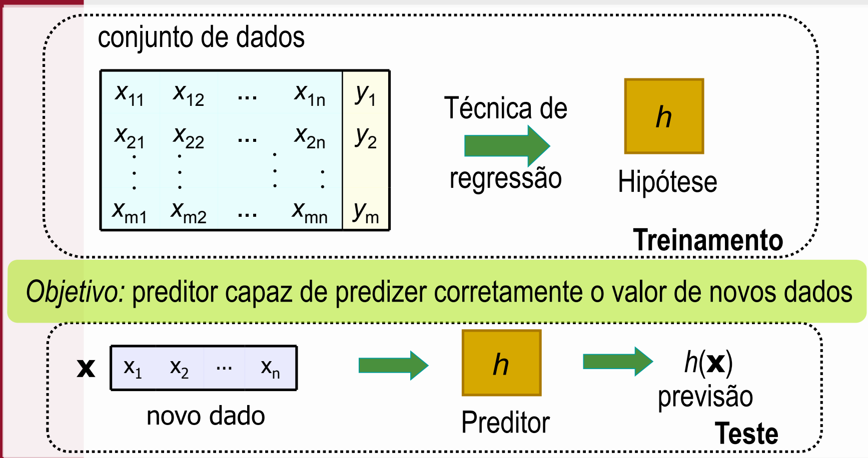


Regressão

Regressão

- Regressão:
  - Meta: aprender função (curva aproximada) que relacione entradas a valores contínuos de saídas
- Exemplos:
  - Prever valor de mercado de um imóvel
  - Prever o lucro de um empréstimo bancário

Regressão: Modelo



Regressão

- Técnicas de AM:
  - Regressão Linear
  - Árvores de Regressão
  - Redes Neurais Artificiais
  - Máquinas de Vetores de Suporte
  - Etc.

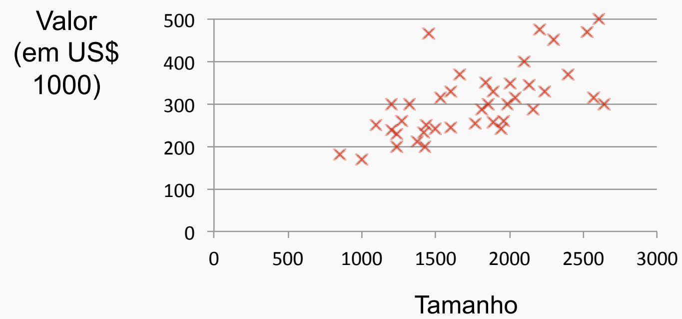
Cada técnica de AM utiliza um mecanismo distinto na aproximação da função objetivo

Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina

Regressão Linear com uma Variável

Regressão: exemplo

- Conjunto de treinamento (valor de imóveis)



Regressão: exemplo

- Conjunto de treinamento (valor de imóveis)

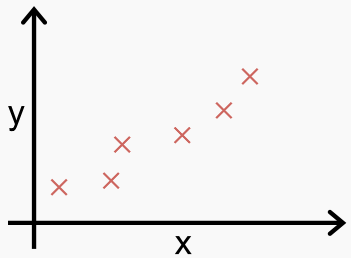
Tamanho (x)	Valor x U$ 1000 (y)
2104	460
1416	232
1534	315
852	178
...	...

Notação

m = número de amostras
x = atributos (entrada)
y = meta (saída)

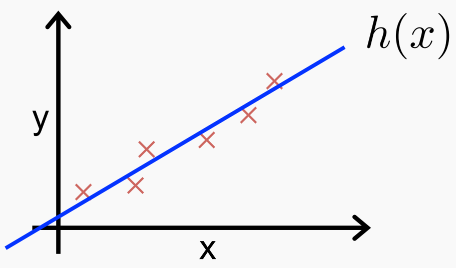
Representação da hipótese (h)

- Hipótese:



Representação da hipótese (h)

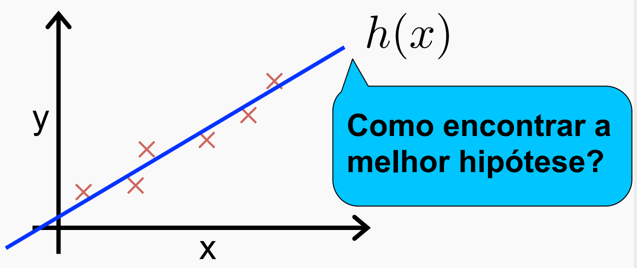
- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



- Regressão linear com uma variável
- Regressão linear univariada

Representação da hipótese (h)

- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



- Regressão linear com uma variável
- Regressão linear univariada

Escolha da hipótese

- Conjunto de treinamento (valor de imóveis)

Tamanho (x)	Valor x U$ 1000 (y)
2104	460
1416	232
1534	315
852	178
...	...

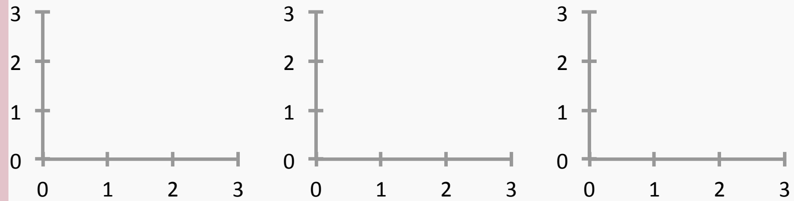
Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$

Parâmetros: θ_i

Como escolher bons valores para θ_i ?

Escolha de θ

- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



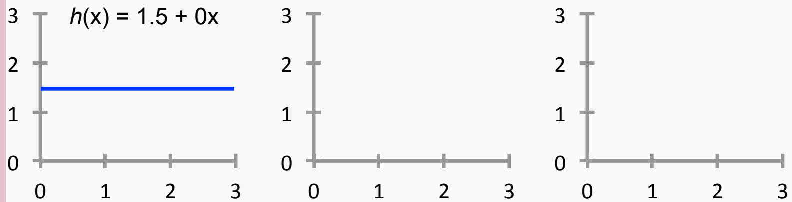
$\theta_0 = 1.5$
$\theta_1 = 0$

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

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

Escolha de θ

- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



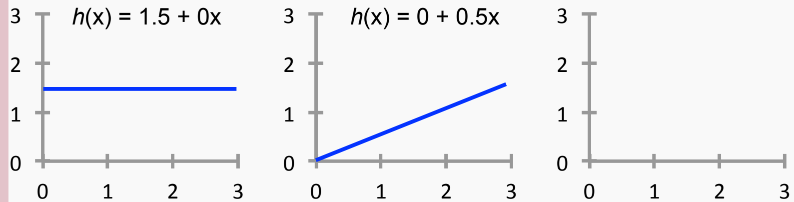
$\theta_0 = 1.5$
$\theta_1 = 0$

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

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

Escolha de θ

- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



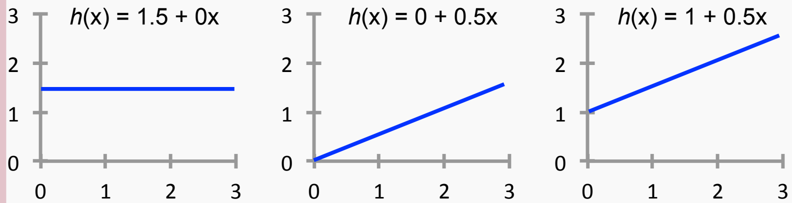
$\theta_0 = 1.5$
$\theta_1 = 0$

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

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

Escolha de θ

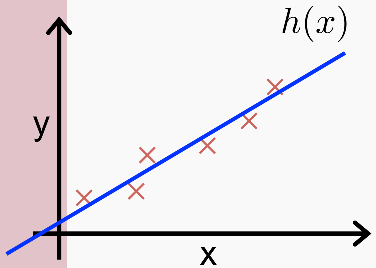
- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



$\theta_0 = 1.5$, $\theta_1 = 0$ | $\theta_0 = 0$, $\theta_1 = 0.5$ | $\theta_0 = 1$, $\theta_1 = 0.5$

Escolha de θ

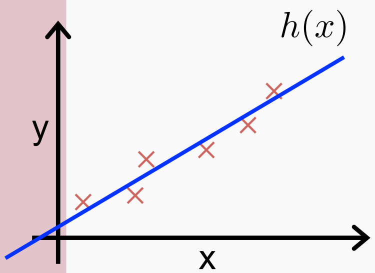
- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



Objetivo: encontrar θ_0, θ_1 de tal forma que *h*(**x**) aproxima ***y*** para os exemplos de treinamento.

Escolha de θ

- Hipótese: $h_\theta(x) = \theta_0 + \theta_1(x)$



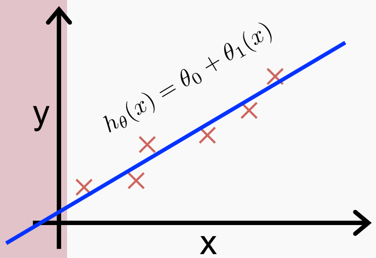
Em outras palavras:
- Escolher θ_0, θ_1 que minimize o **Erro Quadrático Médio** entre as predições (*h*(**x**)) e as saídas esperadas (***y***) para as amostras de treinamento.

Objetivo: encontrar θ_0, θ_1 de tal forma que *h*(**x**) aproxima ***y*** para os exemplos de treinamento.

Função Custo (J)

- Erro quadrático médio

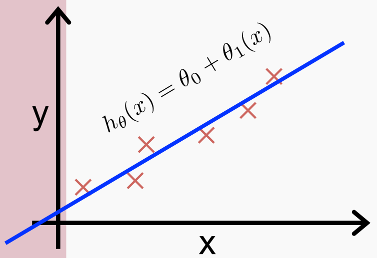
$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2$$



Função Custo (J)

- Minimizar o erro:

$$J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2$$



$$\boxed{\min_{\theta_0, \theta_1} \; J(\theta_0, \theta_1)}$$

Objetivo: encontrar θ_0, θ_1 de tal forma que *h*(**x**) aproxima ***y*** para os exemplos de treinamento.

Tiago A. Almeida

Função Custo (J): Interpretação

- Hipótese:
  $h_\theta(x) = \theta_0 + \theta_1(x)$
- Parâmetros:
  θ_0, θ_1
- Função Custo:
  $J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2$
- Objetivo:
  $\min_{\theta_0, \theta_1} \; J(\theta_0, \theta_1)$

Tiago A. Almeida

Função Custo (J): Interpretação

- Hipótese:
  $h_\theta(x) = \theta_0 + \theta_1(x)$
- Parâmetros:
  θ_0, θ_1
- Função Custo:
  $J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2$
- Objetivo:
  $\min_{\theta_0, \theta_1} \; J(\theta_0, \theta_1)$

Vamos supor que:

$h_\theta(x) = \theta_1 x$

θ_1

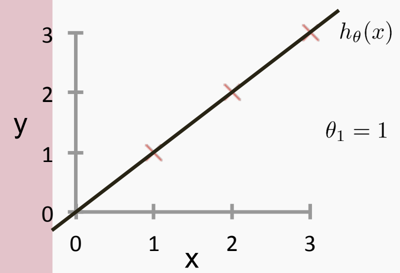
$J(\theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$

$\min_{\theta_1} \; J(\theta_1)$

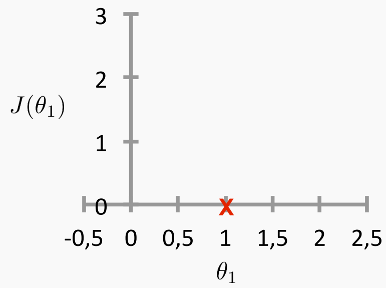
Tiago A. Almeida

Função Custo (J): Interpretação

$h_\theta(x)$

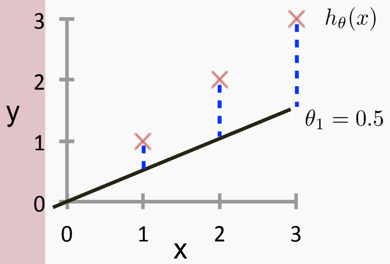


$J(\theta_1)$

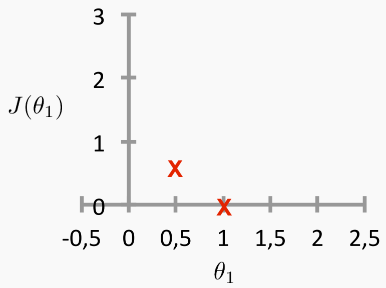


Função Custo (J): Interpretação

$h_\theta(x)$

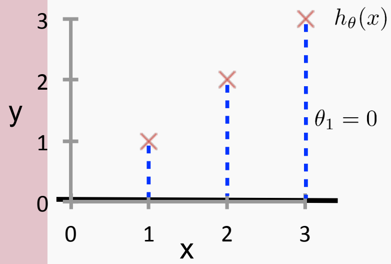


$J(\theta_1)$

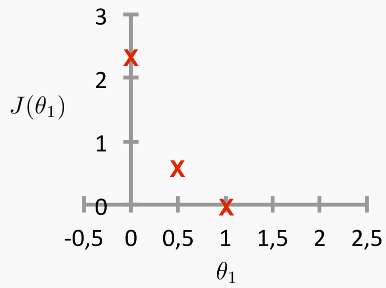


Função Custo (J): Interpretação

$h_\theta(x)$



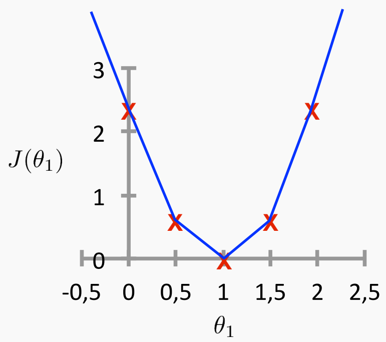
$J(\theta_1)$



Função Custo (J): Interpretação

Variando θ_1 obtém-se uma parábola para $J(\theta_1)$

$J(\theta_1)$

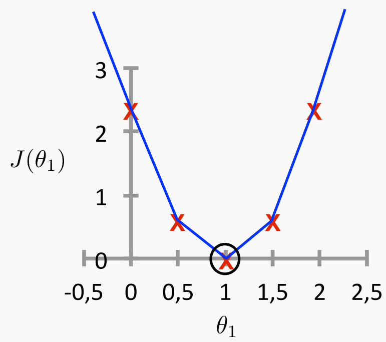


Função Custo (J): Interpretação

Variando θ_1 obtém-se uma parábola para $J(\theta_1)$

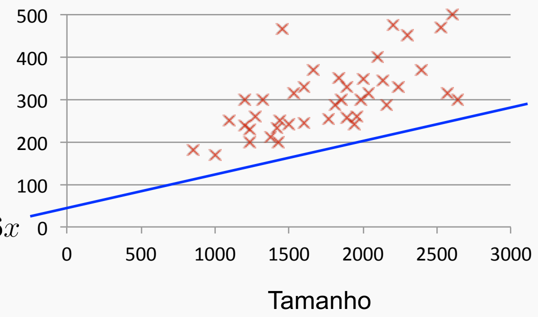
Como deseja-se obter θ_1 que minimize $J(\theta_1)$, então θ_1 = 1 é a melhor escolha para esse exemplo, pois $J(1) = 0$

$J(\theta_1)$



Função Custo (J): Interpretação

- Conjunto de treinamento (valor de imóveis)



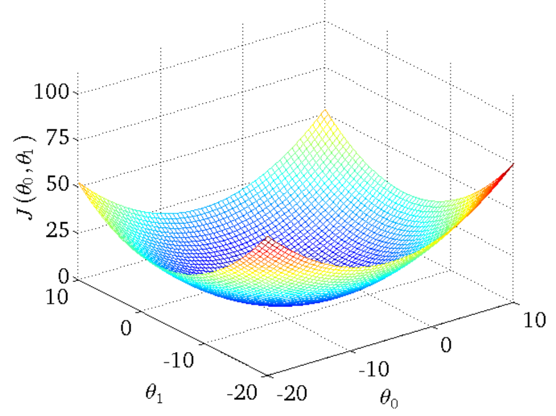
$h_\theta(x) = 50 + 0.06x$

$\theta_0 = 50$

$\theta_1 = 0.06$

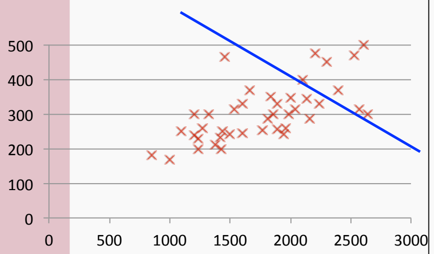
Função Custo (J): Interpretação

- $J(\theta_0, \theta_1)$

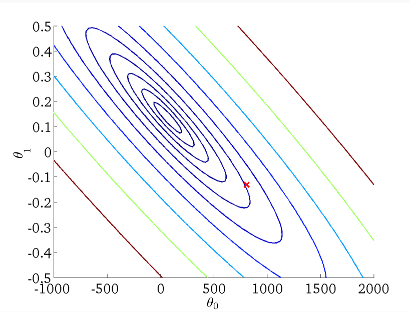


Função Custo (J): Interpretação

$h_\theta(x)$

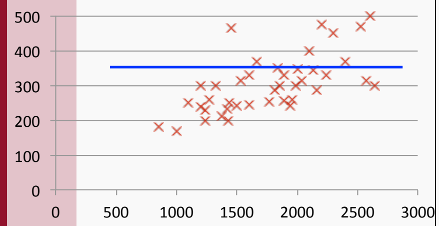


$J(\theta_0, \theta_1)$

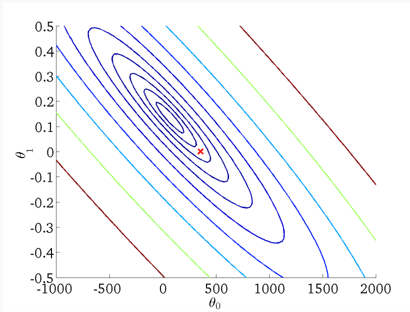


Função Custo (J): Interpretação

$h_\theta(x)$

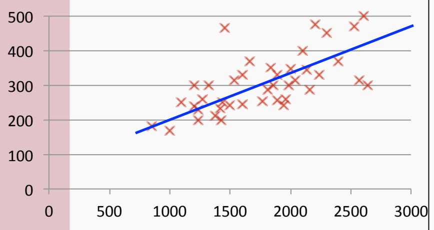


$J(\theta_0, \theta_1)$

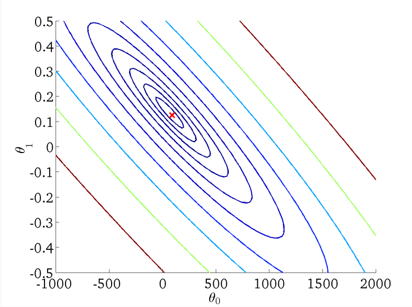


Função Custo (J): Interpretação

$h_\theta(x)$



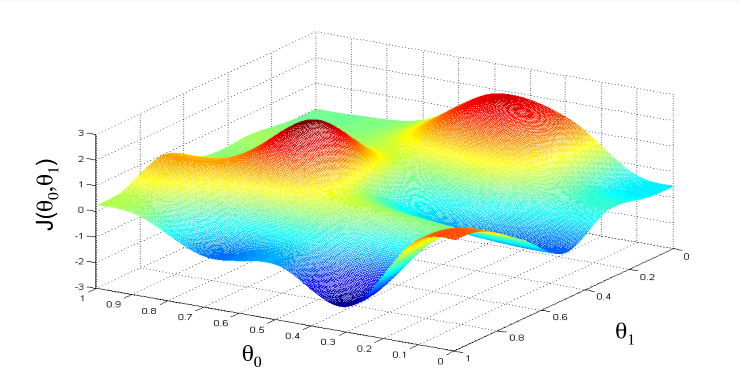
$J(\theta_0, \theta_1)$



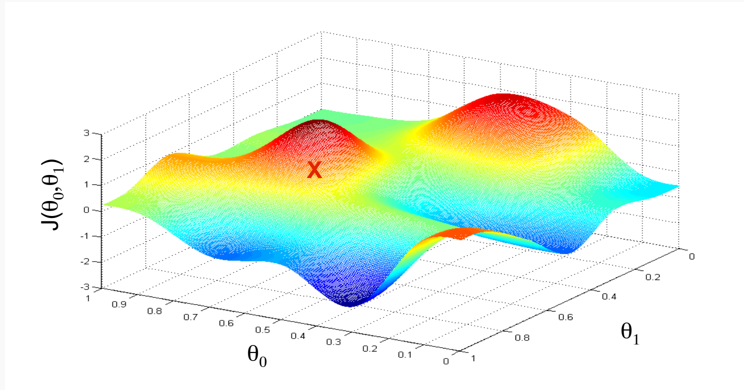
Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina

Gradiente Descendente

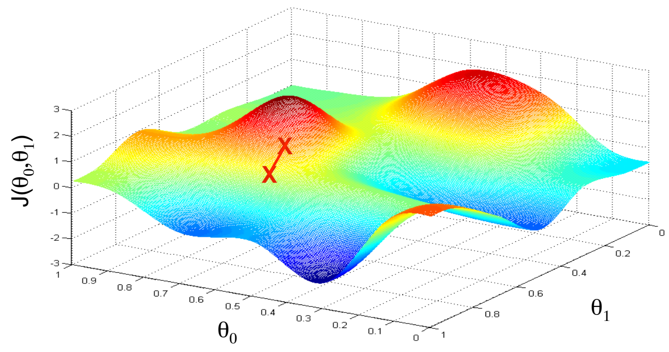
Gradiente descendente



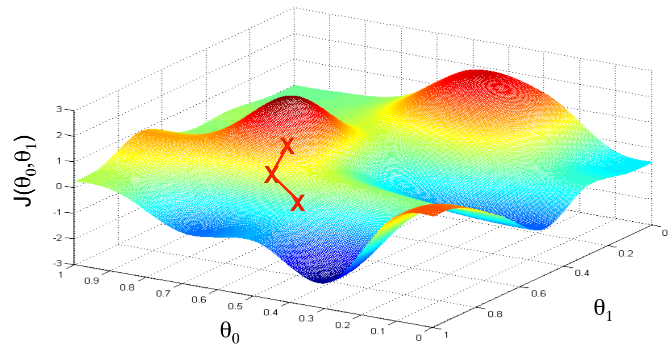
Gradiente descendente



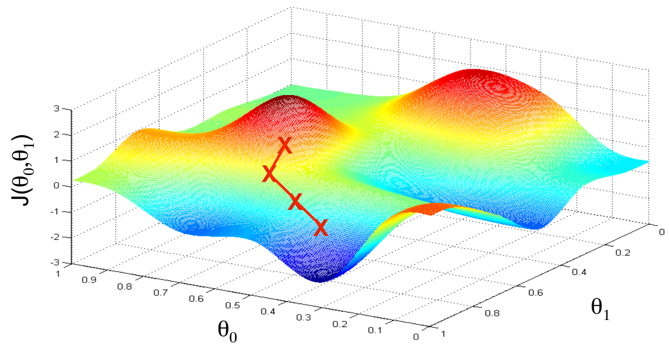
Gradiente descendente



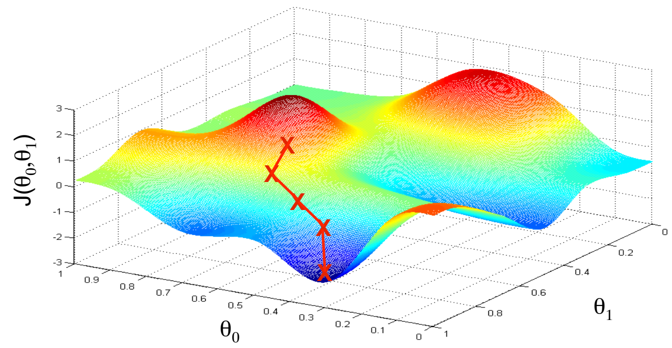
Gradiente descendente



Gradiente descendente



Gradiente descendente



Gradiente descendente

Repetir até convergir {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad \text{(para } j = 0 \text{ e } j = 1\text{)}$$

}

A atualização dos θ_j deve ser "simultânea"

Gradiente descendente

Repetir até convergir {

$$\theta_j := \theta_j - \underbrace{\alpha}_{\text{Taxa de aprendizagem}} \underbrace{\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)}_{\text{Derivada parcial}} \quad \text{(para } j = 0 \text{ e } j = 1\text{)}$$

}

Gradiente descendente

Repetir até convergir {

$$\theta_j := \theta_j - \underbrace{\alpha}_{\text{Taxa de aprendizagem}} \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad \text{(para } j = 0 \text{ e } j = 1\text{)}$$

}

- Se α for muito pequeno, a convergência poderá ser lenta;

- Se α for muito grande, poderá não haver convergência.

Gradiente descendente

Repetir até convergir {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad \text{(para } j = 0 \text{ e } j = 1\text{)}$$

}

- O método converge para um ótimo local mesmo se α for um valor fixo

- Conforme o ótimo for se aproximando, o método reduz o tamanho do passo automaticamente

Gradiente descendente para regressão linear

Repetir até convergir {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} \boxed{J(\theta_0, \theta_1)} \quad \text{(para } j = 0 \text{ e } j = 1\text{)}$$

}

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2$$

$$h_\theta(x) = \theta_0 + \theta_1(x)$$

Gradiente descendente para regressão linear

Repetir até convergir {

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)} \quad \text{(para } j = 0 \text{ e } j = 1\text{)}$$

}

$$j = 0 : \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)$$

$$j = 1 : \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x^i$$

Gradiente descendente para regressão linear

Algoritmo:

repetir até convergir {

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

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

}

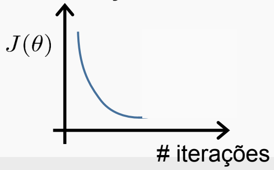
Atualizar "simultaneamente" θ_0 e θ_1

Gradiente descendente para regressão linear

- **Observações importantes**
  - GD é sensível à escala dos dados
    - É recomendável aplicar normalização por padronização para fazer com que $-1 \leq x_i \leq 1$

Gradiente descendente para regressão linear

- **Observações importantes**
  - GD é <u>sensível</u> à escala dos dados
    - É recomendável aplicar normalização por padronização para fazer com que $-1 \leq x_i \leq 1$
  - GD é <u>sensível</u> à taxa de aprendizagem
    - Recomendável testar α com ..., 0.001, 0.01, 0.1, 1, ...
    - Certifique-se de que $J(\theta)$ ***decresce*** a cada iteração. Caso contrário, reduza o valor de α



Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina

Regressão Linear com Múltiplas Variáveis

Regressão linear com múltiplas variáveis

Tamanho	#Cômodos	#Andares	#Anos	Valor (U$1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
...	...	...	...	...

Notação:

n = quantidade de atributos

$x^{(i)}$ = i^{th} amostra do conjunto de treinamento

$x_j^{(i)}$ = valor do atributo j da i^{th} amostra do conjunto de treinamento

Regressão linear com múltiplas variáveis

- **Regressão linear univariada**

$$h_\theta(x) = \theta_0 + \theta_1 x$$

Regressão linear com múltiplas variáveis

- **Regressão linear univariada**

$$h_\theta(x) = \theta_0 + \theta_1 x$$

- **Regressão linear multivariada**

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Tiago A. Almeida

Regressão linear com múltiplas variáveis

- **Regressão linear univariada**

$$h_\theta(x) = \theta_0 + \theta_1 x$$

- **Regressão linear multivariada**

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Para simplificar: $x_0 = 1$

Tiago A. Almeida

Regressão linear com múltiplas variáveis

- **Regressão linear univariada**

$$h_\theta(x) = \theta_0 + \theta_1 x$$

- **Regressão linear multivariada**

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Para simplificar: $x_0 = 1$

$$\boxed{h_\theta(x) = \theta^T x}$$

Tiago A. Almeida

Regressão linear com múltiplas variáveis

Hipótese: $h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$

Parâmetros: $\theta_0, \theta_1, \ldots, \theta_n$

Função de custo:

$$J(\theta_0, \theta_1, \ldots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

Gradiente descendente

Repita $\{$

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

$\}$

Regressão linear com múltiplas variáveis

Algoritmo:

repetir até convergir {

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x_j^i$$

}

Atualizar “simultaneamente” θ_j

Regressão linear com múltiplas variáveis

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x_j^i$$

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x_0^i$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x_1^i$$

$$\theta_2 := \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x_2^i$$

...