Machine Learning Engineer (Udacity)

**INTRODUÇÃO**

Machine Learning (Aprendizado de Máquina), é ramo da Inteligência Artificial I.A, constituída por várias ferramentas estruturada que condiciona algoritmos eficientes e para solucionar problemas complexos a partir da topologia humana de aprendizagem. Em outras palavras, o que diferencia a máquina do homem, é que o homem aprende com o passar do tempo, e a máquina precisa de receber instruções para executar tarefas.

Com essa abordagem de máquina, é possível resolver problemas complexos a partir de dados, seja eles de quais que fonte. O computador aprende por dados, partindo deles é possível especificar soluções.

**ESTRUTURA**

Machine Learning (M.L) é composta por três ramos de aplicação:

* **(Aprendizado Supervisionado) Supervised Learning**
* **(Aprendizado não supervisionado) Unsupervised Learning**
* **(Aprendizado por reforço) Reinforcement Learning**

**Aprendizado Supervisionado**

É possível identificar um problema de Aprendizado Supervisionado (A.S) quando tentamos prever uma variável dependente a partir de uma lista de variáveis independentes. Note que a característica básica de sistemas de aprendizado supervisionado é que os dados que utilizamos para treiná-los contém a resposta desejada, isto é, contém a variável dependente resultante das variáveis independentes observadas. Nesse caso, dizemos que os dados são anotados com as respostas ou classes a serem previstas.

|  |  |
| --- | --- |
| **Variáveis Independentes** | **Variáveis Dependentes** |
| Anos de Carreira, Formação, Idade | Salário |
| Idade do Carro, Idade do Motorista | Risco de Acidente Automotivo |
| Texto de um livro | Escola Literária |
| Temperatura | Receita de venda de sorvete |
| Imagem da Rodovia | Ângulo da direção de um carro autônomo |
| Histórico escolar | nota no ENEM |

**Técnicas de Aprendizado Supervisionado**

**Introdução**

Nessa vertente encontra-se varias técnicas, como **regressão linear**, **regressão logística**, **redes neurais artificiais**, **máquina de suporte vetorial** (ou máquinas kernel), **árvores de decisão, k-vizinhos mais próximos** e **Bayes ingênuo**. Aprendizado de máquina supervisionado é a área que concentra a maioria das aplicações bem-sucedidas e onde a maioria dos problemas já estão bem definidos.

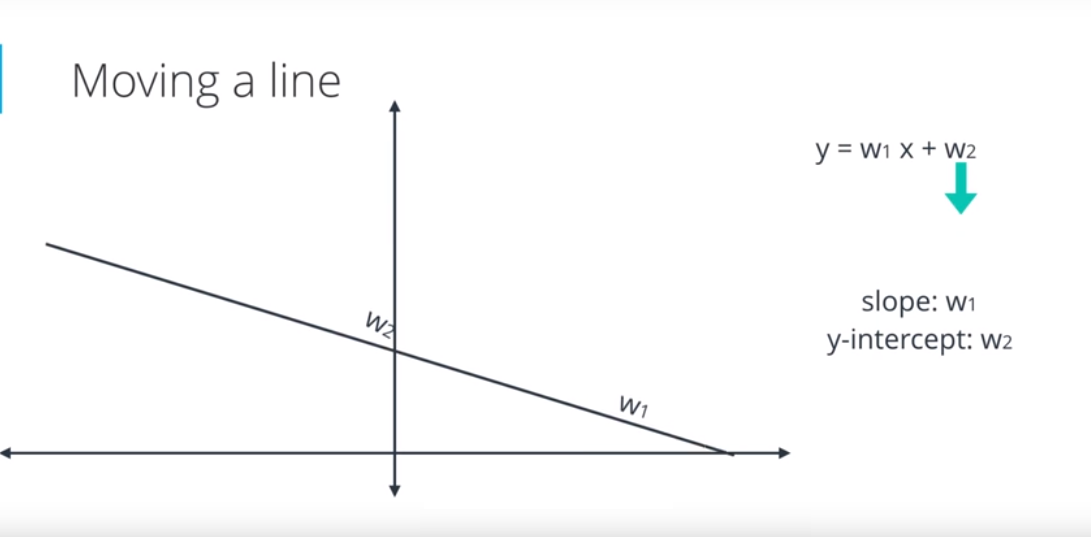
**Ferramentas de (A.S)**

* **Regressão Linear (Linear Regression)**
* **Algoritmo do Perceptron (Perceptron Algorithim)**
* **Árvore de Decisão ( Decision Tree)**
* **Nave Bayes**
* **Máquinas de Suporte Vetorial (Support Vector Machine)**
* **Métodos de Ensemble ( Method Ensemble)**

**Regressão Linear (Linear Regression)**

É um algoritmo muito eficaz para prever dados numéricos. A principal diferença entre classificação e regressão é: Na classificação, nós prevemos um estado, enquanto, na regressão, prevemos um valor.

**Movendo uma linha**

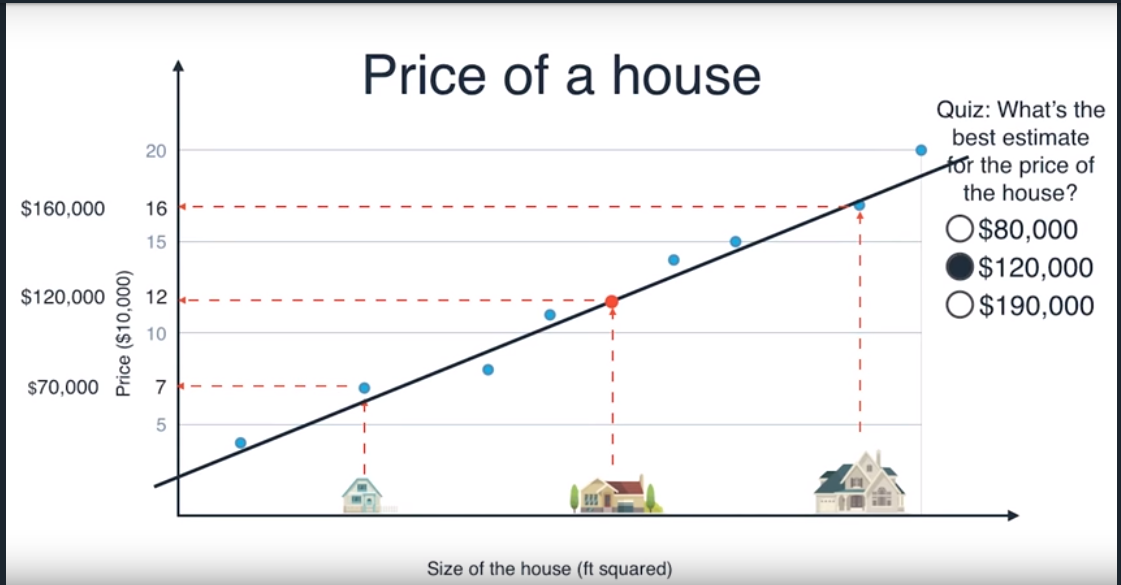


A função básica usada na regressão linear é:

(eq.1)

Após definido uma linha reta, o ajuste da linha é baseado nos parâmetros da equação.

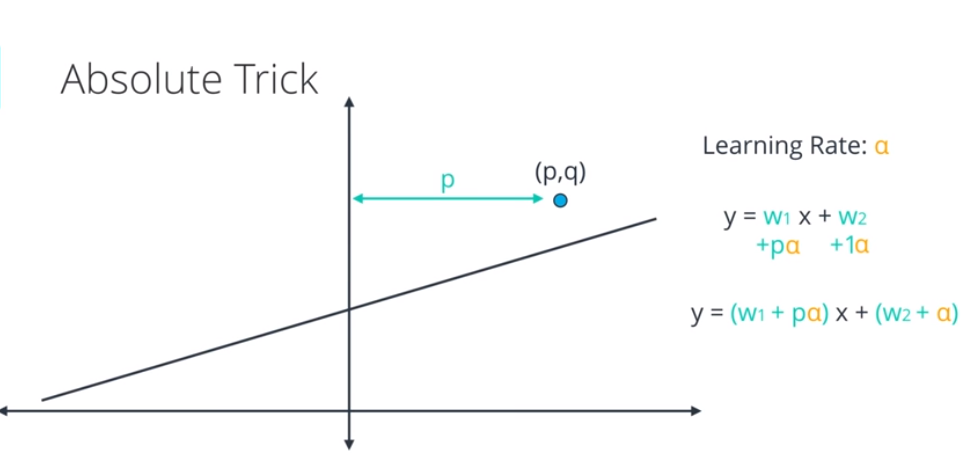
**Exemplo 1 – Preço de casa**

****

Solucionando precificação de imóveis, passos a serem seguidos:

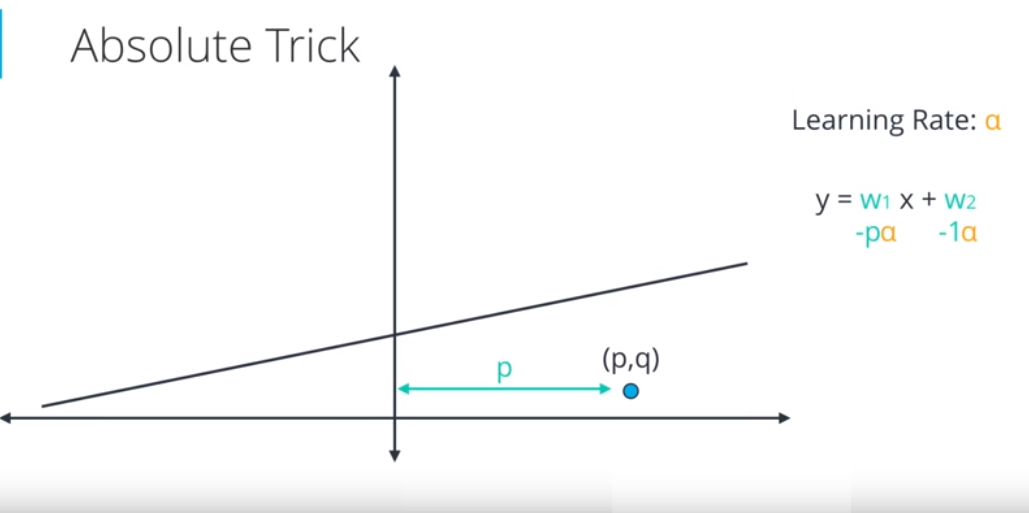
1. Ajustando uma linha reta entre os dados de forma simétrica entre os eixos;
2. Após efetuar o tracejo no eixo horizontal, efetuar no vertical conforme a figura acima com tracejo em vermelho.

**Caso Absoluto**



No caso absoluto é possível obter maior precisão no ajuste da linha reta, movendo de forma micro. Equação Linear Absoluta a seguir para um ponto a direita do eixo vertical e superior a reta conforme figura acima:

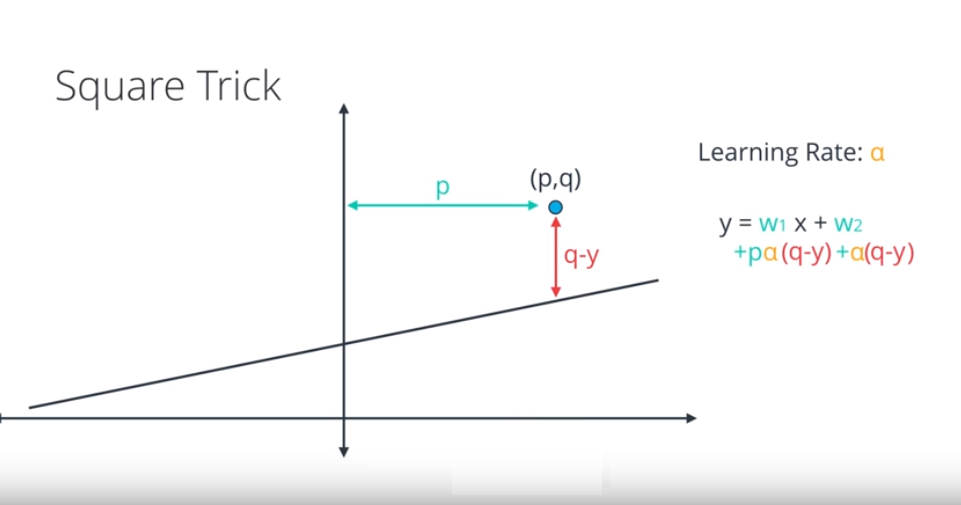
(Eq. 2)



No caso absoluto é possível obter maior precisão no ajuste da linha reta, movendo de forma micro. Equação Linear Absoluta a seguir para um ponto a direita do eixo vertical e superior ao eixo horizontal:

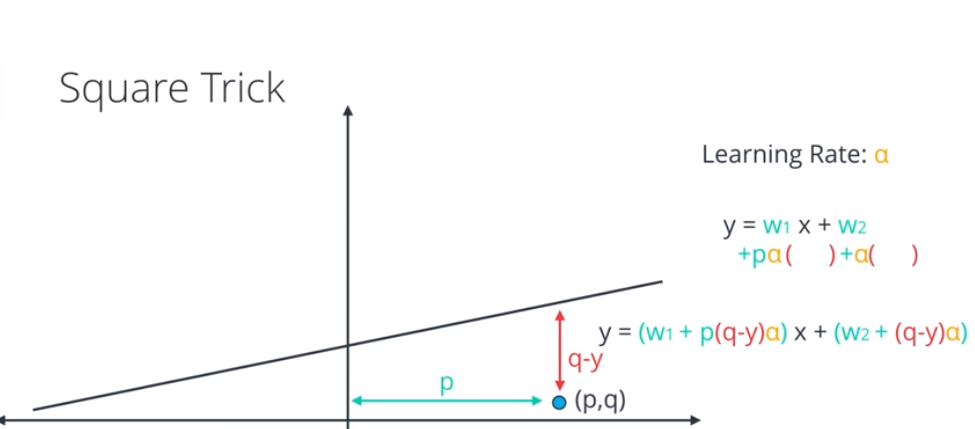
(Eq. 3)

**Caso Quadrado**



No caso quadrado o modelo possui mais graduação comparado ao caso absoluto, onde o deslocamento da linha em relação ao ponto é feito de forma mínima ou máxima conforme a distância da linha em relação ao ponto. O gráfico acima descreve o caso de ponto superior direito, onde a seguir descreve a equação:

(Eq.4)

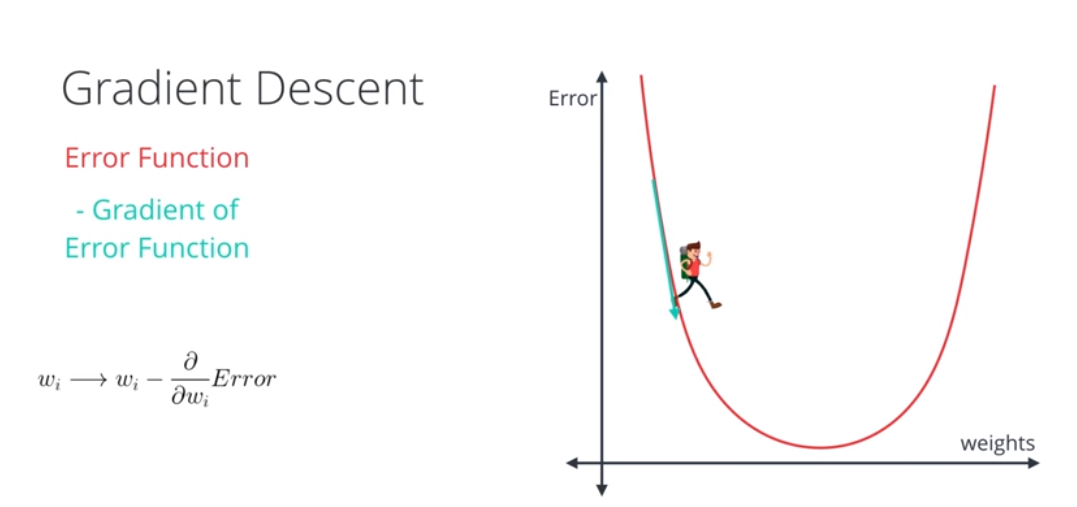


A seguir a nova equação para o ponto inferior a linha no lado direito.

(Eq. 5)

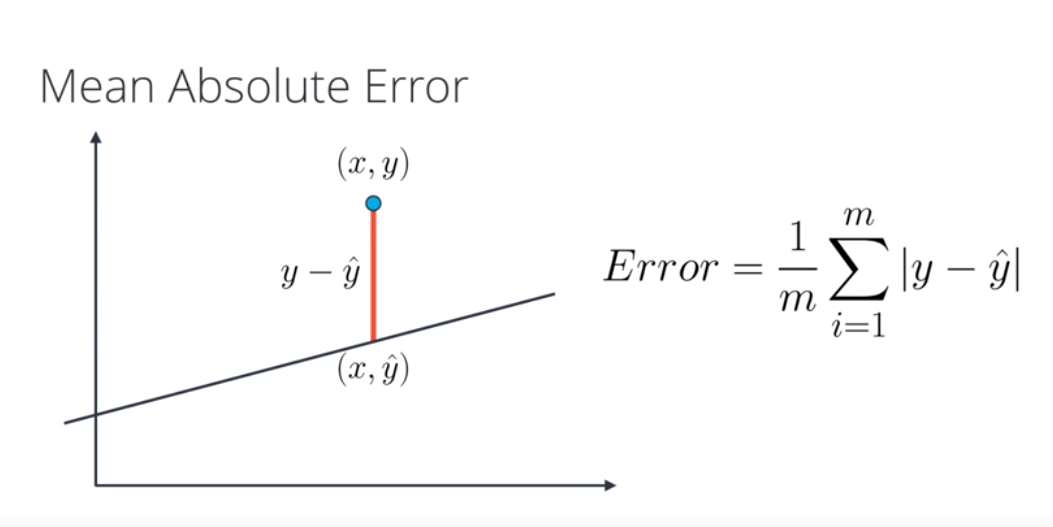
**Gradiente de descida**

O algoritmo trabalha com movimento da linha e menor distância entre os pontos em relação a linha.

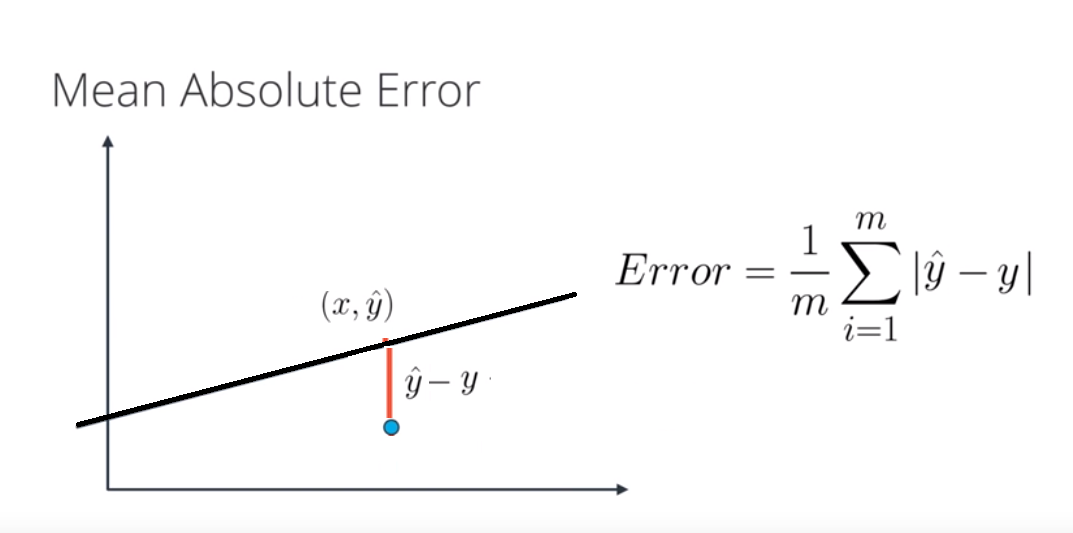


**Erro Absoluto**

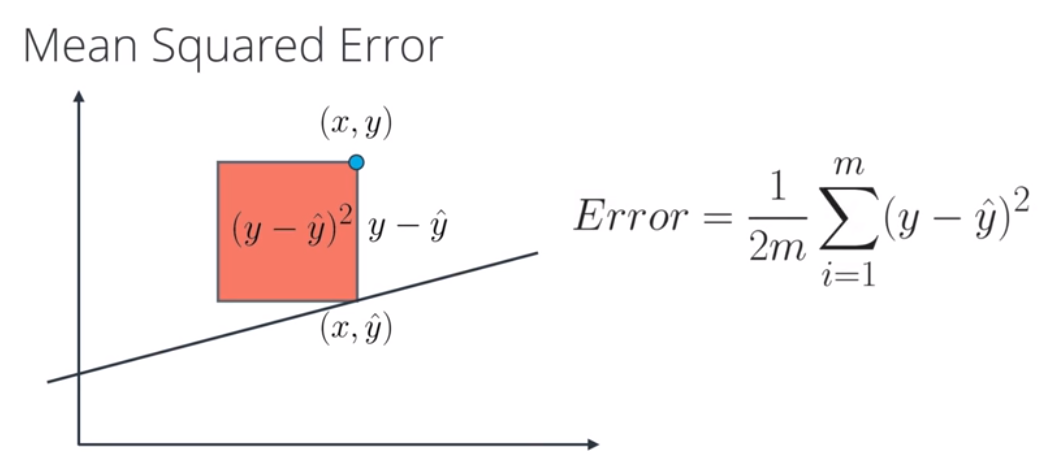
Ponto na posição superior direita da linha e do eixo., conforme gráfico a seguir.

****

A seguir o caso referente ao ponto na posição inferior direto abaixo da linha.

****

**Erro Quadrado médio**

****

Para este caso quadrático, a graduação é maior, onde a somatória de valores usados, é com base na média de erros. O objetivo central é diminuir o erro, com o recurso graduado quadrático.

**Minimizando funções de erros**

**Desenvolvimento da derivada da função de erro**

Lembre que definimos o erro como:

(Eq. 6)

Além disso, definimos a predição como:

(Eq. 7)

Então, calculamos a derivada do erro em relação a *w*1​ , simplesmente usamos a regra da cadeia:

(Eq. 8)

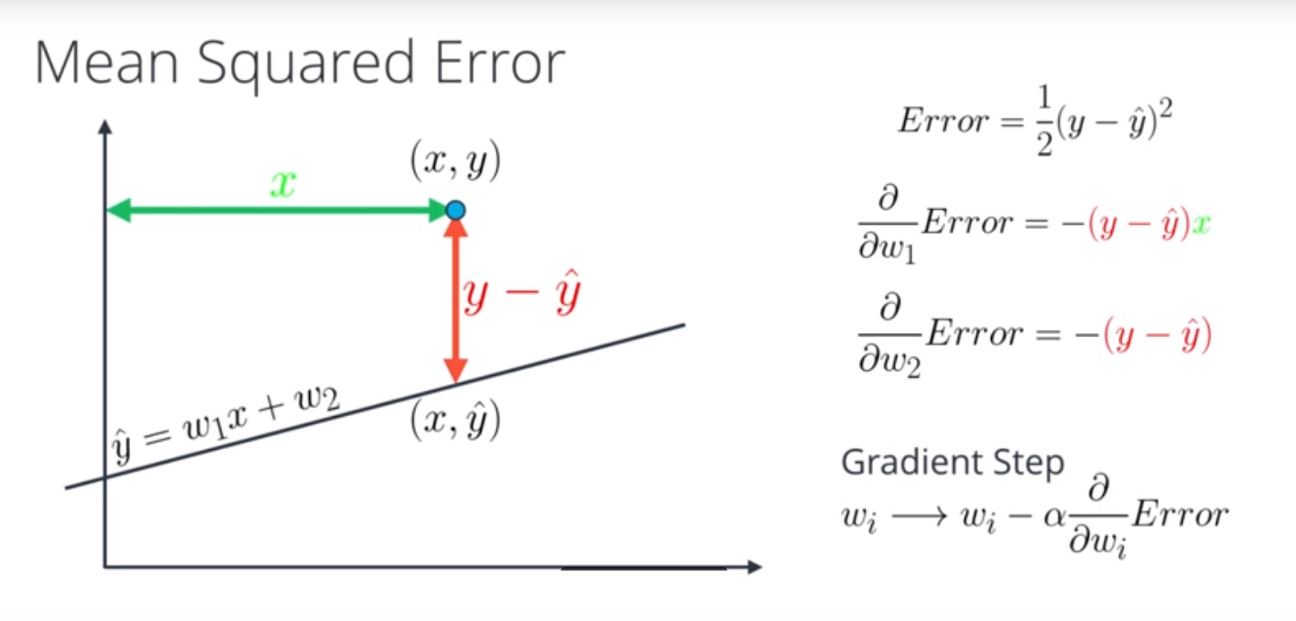
O primeiro fator da rireita é a derivada do erro em relação à predição ŷ,

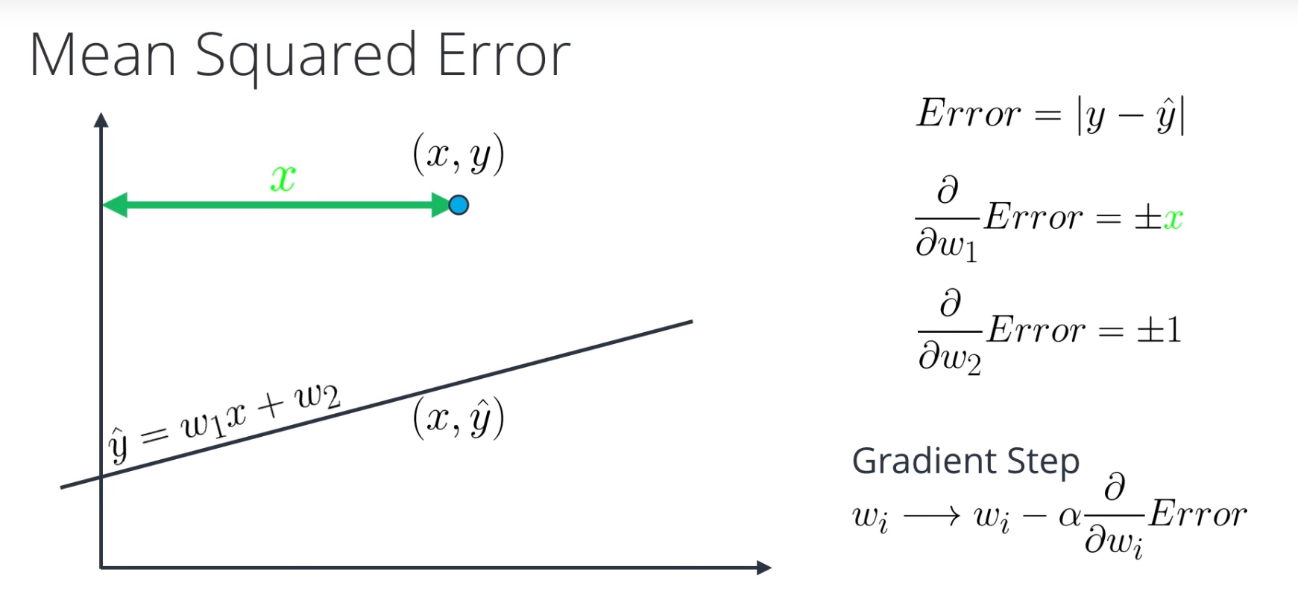
que é -(y - ŷ​).

O segundo fator é a derivada da predição em relação a *w*1​, que é simplesmente x*.*

Logo, a derivada é:

[[https://s3.amazonaws.com/video.udacity-data.com/topher/2018/August/5b845b4b_gif-1/gif-1.gif](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/5b5b3ff6-5d7c-409f-ac2c-6b6652288ce5) (Eq. 9)](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/5b5b3ff6-5d7c-409f-ac2c-6b6652288ce5)

****

****

# Erro médio quadrado (absoluto) x erro total quadrado (ou absoluto)

Uma confusão potencial é a seguinte: Como sabemos se deveríamos usar o erro médio total quadrado (ou absoluto) ou o erro total quadrado (ou absoluto)?

O erro quadrado total é a soma dos erros em cada ponto, dada pela seguinte equação:

(Eq. 10)

Considerando que o erro médio quadrado é a média desses erros, dada pela equação, onde m*m* é o número de pontos:

(Eq. 11)

A boa notícia é que não importa muito. Como podemos ver, o erro total quadrado é apenas um múltiplo do erro médio quadrado, já que

(Eq. 12)

Portanto, já que derivativos são funções lineares, o gradiente de *T* é também *m* vezes o gradiente de M.

No entanto, o passo de gradiente de descida consiste em subtrair o gradiente do erro vezes a taxa de aprendizagem \alpha*α*. Portando, escolher entre o erro médio quadrado e o erro total quadrado simplesmente equivale a escolher uma taxa de aprendizagem diferente.

Na vida real, teremos algoritmos que nos ajudarão a determinar uma boa taxa de aprendizagem com a qual trabalhar. Portanto, se usamos o erro médio ou o erro total, o algoritmo acabará escolhendo uma taxa de aprendizagem diferente.

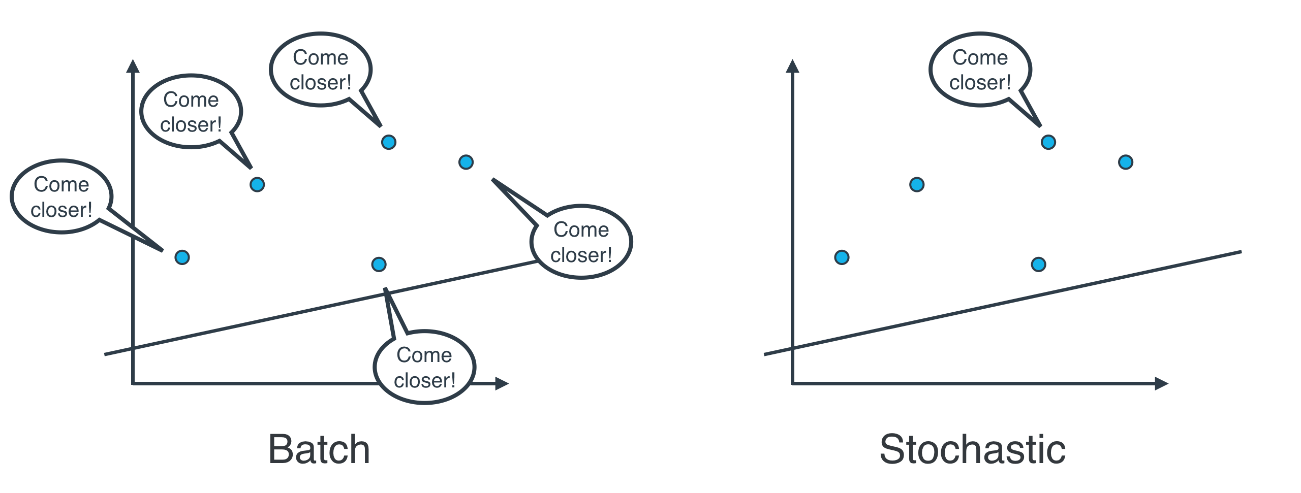
# Gradiente de descida em lotes x estocástico

Neste ponto, parece que vimos duas formas de fazer uma regressão linear.

* Ao aplicar o truque quadrático (ou absoluto) a cada ponto em nossos dados, um por um, e repetir esse processo muitas vezes.
* Ao aplicar o truque quadrático (ou absoluto) a cada ponto em nossos dados, todos ao mesmo tempo, e repetir esse processo muitas vezes.

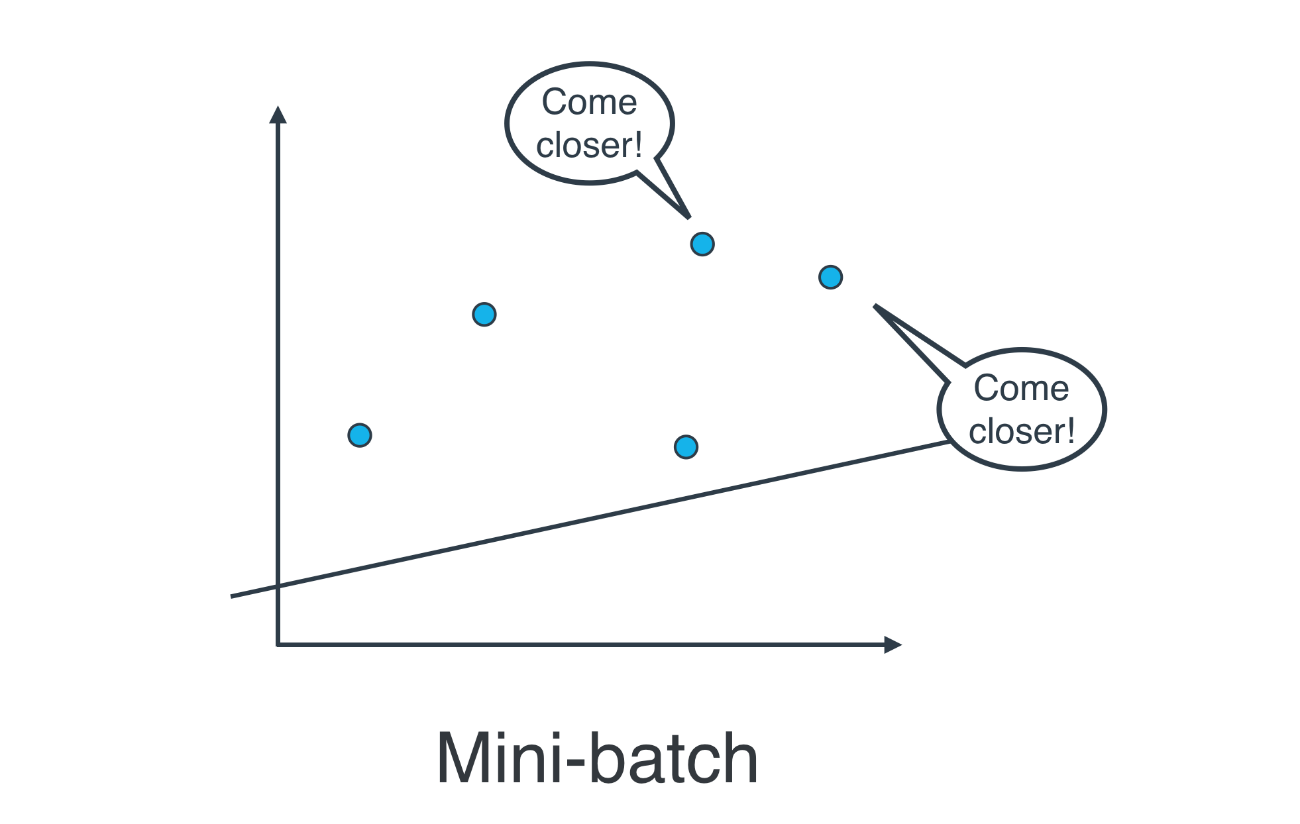
Mais especificamente, o truque quadrático (ou absoluto), quando aplicado a um ponto, nos dá alguns valores para adicionar os pesos do modelo. Podemos adicionar esses valores, atualizar nossos pesos e, então, aplicar o truque quadrático (ou absoluto) no próximo ponto. Ou podemos calcular esses valores para todos os pontos, adicioná-los e, então, atualizar os pesos com a soma desses valores.

O segundo é chamado de gradiente de descida em lote. O primeiro é chamado de gradiente de descida estocástico.

[[](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/1b0ed370-9d37-4b2f-a8e9-d4082ec73b77)](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/1b0ed370-9d37-4b2f-a8e9-d4082ec73b77)

A pergunta é, qual é utilizado na prática?

Na verdade, na maioria dos casos, nenhum dos dois. Pense nisso: Se seus dados são enormes, ambos são um pouco lentos, em termos de computação. A melhor forma de fazer regressão linear é dividir seus dados em vários lotes pequenos. Cada lote, irá ter mais ou menos o mesmo número de pontos. Então, use cada lote para atualizar seus pesos. Isso chama-se gradiente de descida em minilote.

[[](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/1b0ed370-9d37-4b2f-a8e9-d4082ec73b77)](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/1b0ed370-9d37-4b2f-a8e9-d4082ec73b77)

# Regressão linear Example

Nesta seção, você vai usar a regressão linear para prever a expectativa de vida a partir do [índice de massa corporal (IMC)](https://en.wikipedia.org/wiki/Body_mass_index). Antes de fazer isso, vamos dar uma olhada nas ferramentas necessárias para construir esse modelo.

Para seu modelo de regressão linear, você usará a classe [LinearRegression](http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html" \t "_blank) do scikit-learn. Essa classe fornece a função [fit()](http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html" \l "sklearn.linear_model.LinearRegression.fit" \t "_blank) para ajustar o modelo a seus dados.

>>> **from** sklearn.linear\_model **import** LinearRegression

>>> model = LinearRegression()

>>> model.fit(x\_values, y\_values)

No exemplo acima, a variável model é um modelo de regressão linear que foi ajustado aos dados x\_values e y\_values. Ajustar o modelo significa encontrar a melhor linha que ajusta os dados de treinamento. Vamos fazer duas previsões usando a função do modelo [predict()](http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html#sklearn.linear_model.LinearRegression.predict).

>>> print(model.predict([ [127], [248] ]))

[[ 438.94308857, 127.14839521]]

O modelo retornou uma array de previsões, uma previsão para cada array em entrada. A primeira entrada, [127], teve uma previsão de 438.94308857.. A segunda entrada, [248], teve uma previsão de 127.14839521.. A razão para prever sobre uma array como [127] e não simplesmente 127, é que você pode ter um modelo que faz uma previsão usando diversos atributos. Vamos abordar o uso de múltiplas variáveis em regressão linear mais tarde nessa aula. Por enquanto, vamos nos ater a um único valor.

# Regressão linear - Quiz

Neste quiz, você trabalhará com dados sobre a expectativa de vida média ao nascimento e a média de IMC para homens ao redor do mundo. Os dados foram obtidos no site [Gapminder](https://www.gapminder.org/" \t "_blank).

O arquivo de dados pode ser encontrado sob a guia "bmi\_and\_life\_expectancy.csv" no quiz abaixo. Ele inclui três colunas, contendo os seguintes dados:

**Country** – O país em que a pessoa nasceu.

**Life expectancy** – A expectativa de vida média ao nascer para uma pessoa nesse país.

**BMI** – O IMC médio para homens nesse país.

### Você precisará completar cada um dos passos a seguir:

**1. Carregue os dados**

Os dados estão em um arquivo chamado "bmi\_and\_life\_expectancy.csv".

Use o [read\_csv](http://pandas.pydata.org/pandas-docs/stable/generated/pandas.read_csv.html" \t "_blank) do pandas para carregar os dados em um dataframe (não se esqueça de importar pandas!)

Atribua o dataframe à variável bmi\_life\_data.

**2. Crie um modelo de regressão linear**

Crie um modelo de regressão usando o método [LinearRegression](http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html" \t "_blank) do scikit-learn e atribua a bmi\_life\_model.

Ajuste o modelo aos dados.

**3. Faça uma previsão usando o modelo**

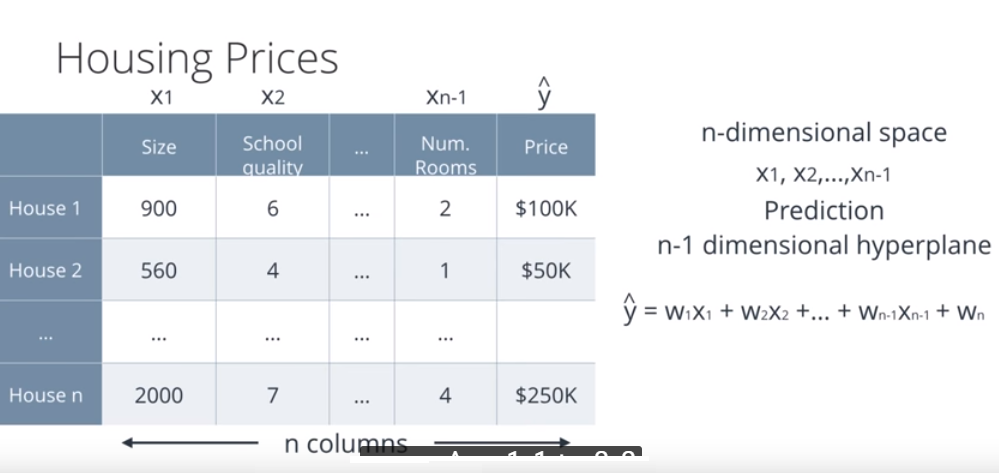
Faça uma previsão usando um IMC de 21.07931 e atribua o resultado à variável laos\_life\_exp

**Regressão Linear em Dimensões Maiores**

****

Ocaso acima apresentado no gráfico representa um problema de precificação de casa, foi escolhido para melhor aplicar o modelo de dimensões maiores. Conforme demostrado, em casos assim os dados são tratados em um plano multidimensional, podendo ter entre 3 ou mais dimensões. Onde a função para o problema que possui uma extensão de dimensões maiores comparado ao caso simples, é descrita da seguinte forma.

(Eq. 13)



Para dimensões maiores que três, em um espaço dimensional de n-1, a função linear é descrita da seguinte forma.

(Eq. 14)

# Regressão linear múltipla

Na última seção, você viu como podemos prever expectativa de vida usando o IMC. Aqui, o IMC foi o **preditor**, também conhecido como uma variável independente. Um preditor é uma variável a qual analisamos para fazer previsões sobre outras variáveis, enquanto os valores que você está tentando prever são conhecidos como variáveis dependentes. Nesse caso, a expectativa de vida era a variável dependente.

Agora, digamos que também temos novos dados sobre a frequência cardíaca de cada pessoa. Podemos criar uma previsão da expectativa de vida usando tanto o IMC quanto a frequência cardíaca?

Claro que sim! Como vimos no vídeo anterior, podemos fazer isso usando uma regressão linear múltipla.

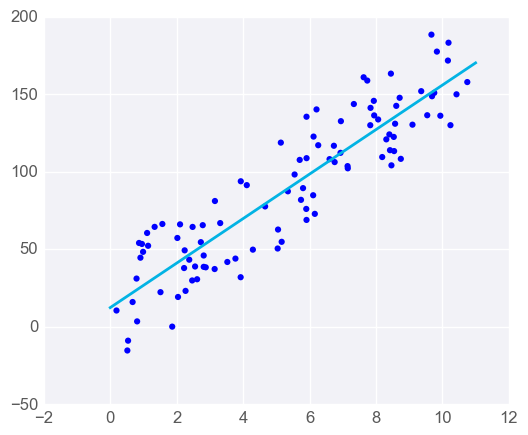
Se o resultado que você deseja prever depende de mais de uma variável, você pode criar um modelo mais complexo, que leva isso em consideração. Desde que elas sejam relevantes à situação, usar mais variáveis independentes/preditoras pode ajudar você a obter uma previsão melhor.

Quando há apenas um preditor, o modelo de regressão linear é uma linha, mas conforme você adiciona mais variáveis preditoras, você adiciona também mais dimensões a ele.

Quando você possui uma variável preditora, a equação da linha é

y = m x + b *(Eq. 15)*

e o gráfico pode ser algo assim:

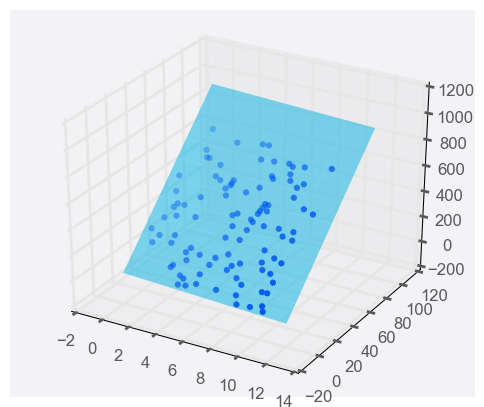
[[](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/81f1671b-b376-49c0-84d4-8ea18eab8852)](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/81f1671b-b376-49c0-84d4-8ea18eab8852)

[Regressão linear com uma variável preditora](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/81f1671b-b376-49c0-84d4-8ea18eab8852)

Adicionar uma variável preditora para chegarmos a duas variáveis preditoras significa que a equação de previsão é:

*(Eq. 16)*

Para representar isso graficamente, precisaremos de um gráfico tridimensional com o modelo de regressão linear representado como um plano:

[[](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/81f1671b-b376-49c0-84d4-8ea18eab8852)](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/81f1671b-b376-49c0-84d4-8ea18eab8852)

[Regressão linear com duas variáveis preditoras](https://classroom.udacity.com/nanodegrees/nd009-br/parts/5b3355b9-538a-4012-9247-7dae50a6ef84/modules/2ebcf321-227b-4084-80e0-5536a73ee7d6/lessons/c3bb570a-0e06-4cb3-b888-9c3e6e986ea2/concepts/81f1671b-b376-49c0-84d4-8ea18eab8852)

Você pode usar mais de duas variáveis preditoras. Na verdade, deve usar tantas quanto forem úteis! Se você usar n*n* variáveis preditoras, o modelo poderá ser representado pela equação

*(Eq. 17)*

Conforme você cria um modelo com mais variáveis preditoras, fica mais difícil de visualizar, mas por sorte, todo o resto sobre regressão linear segue igual. Ainda podemos ajustar modelos e fazer previsões da mesma forma