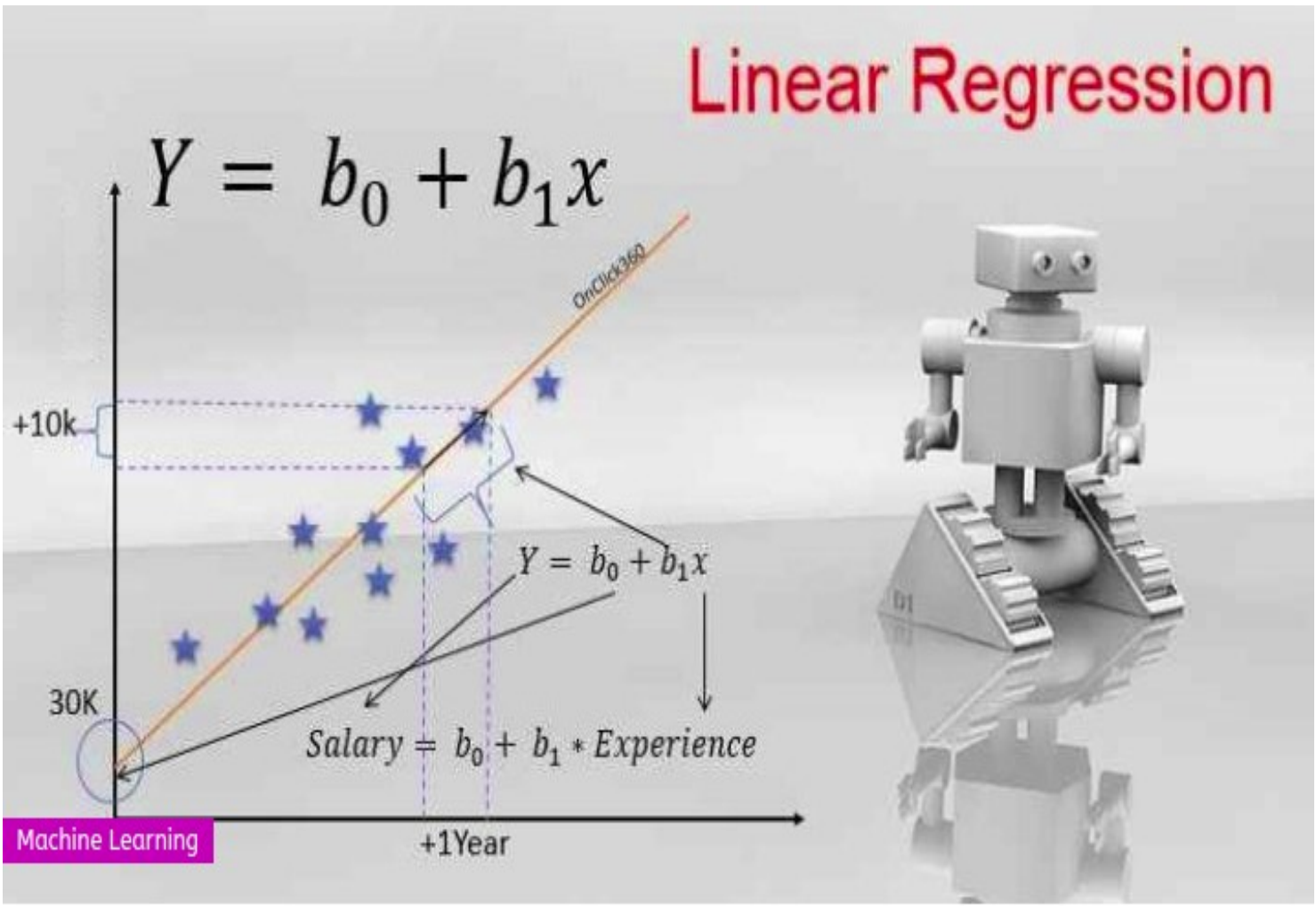

Linear Regression
$Y = b_0 + b_1 x$
OmClick360
+10k
30K
$Y = b_0 + b_1 x$
$Salary = b_0 + b_1 * Experience$
+1Year
Machine Learning

Regressão Linear Simples

Dado aos computadores a capacidade de aprender com os dados, a análise de regressão é uma subcategoria do aprendizado de máquina supervisionado.
Diferente da classificação - outra subcategoria de análise de aprendizagem – a regressão visa prever resultados em uma escala contínua, por exemplo, preço da casa, idade.
Para começar, vamos considerar a regressão linear simples.
A análise de regressão entende-se como **previsão**. Quando fazemos uma regressão queremos prever resultados.
O objetivo é prever os valores de uma variável dependente com base em resultados da variável independente.
A fórmula matemática da regressão linear é:

$$Y = ax + b$$

Aonde **x** é a variável independente e **y** é a variável dependente.
Suponha que você queira saber o preço de uma pizza.Você pode simplesmente olhar para um menu.
Mas usaremos a regressão linear simples para prever o preço de uma pizza com base em um atributo da pizza que podemos observar.
Vamos modelar a relação entre o tamanho de uma pizza e seu preço.
Primeiro, escreveremos um programa com o scikit-learn que pode prever o preço de uma pizza, devido ao seu tamanho.
Em seguida, discutiremos como a regressão linear simples funciona e como ela pode ser generalizada para trabalhar com outros tipos de problemas.
Vamos supor que você tenha registrado os tamanhos e os preços das pizzas que você comeu anteriormente em seu diário de pizza. Essas observações incluem nossos dados de treinamento:

Instâncias de Treinamento	Tamanho(centímetro)	Preço (R$)
1	6	7
2	8	9
3	10	13
4	14	17.5
5	18	18

Dado o exemplo acima, o Tamanho é nossa variável **x** e o Preço é nossa variável **y.**

A regressão linear é um dos algoritmos de aprendizado supervisionado mais simples do nosso kit de ferramentas.

Importando as bibliotecas

```
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error
from sklearn.linear_model import LinearRegression
import numpy as np
import pandas as pd
```

Importamos as bibliotecas que iremos usar no nosso modelo.
Matplotlit serve para plotar gráficos.
A biblioteca skitlearn que usaremos em todos os nossos modelos de Machine Learning, falarems mais dela em outro post.
Dentro skitlearn importamos a bliblioteca que usaremos para avaliar a eficiência do modelo e a bliblioteca LinearRegression que realizará os cálculos matemáticos.
Depois as bibliotecas numpy e pandas, assunto de outro post.

Criando as variáveis

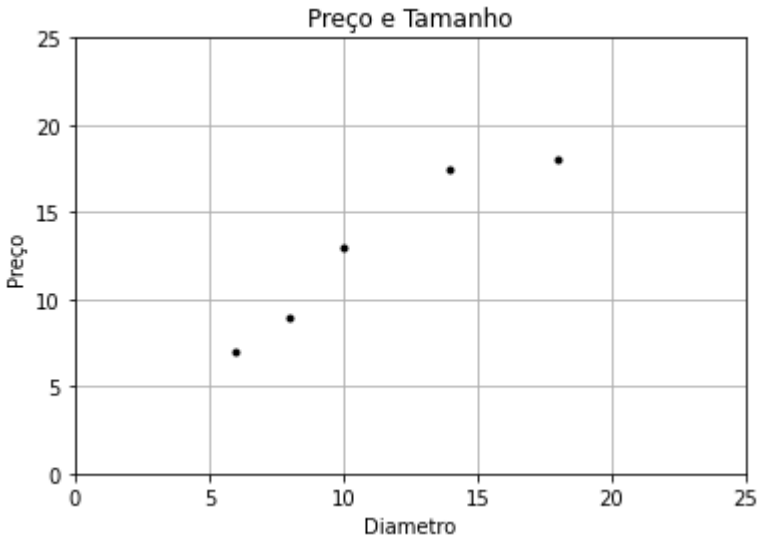
```
Tamanho = [[6], [8], [10], [14], [18]]
preço = [[7], [9], [13], [17.5], [18]]
```

Criamos uma variável **Tamanho** que será o tamanho da pizza em centímentros, ela será nossa variável **X**, que significa a variável indepente.

A variável **preço,** será o valor que tentaremos prever, ela será nossa variável **y,** que significa a variável depende, isto é depende de X.

Representando as variaveis graficamente

```
plt.figure()
plt.title('Preço e Tamanho')
plt.xlabel('Diametro')
plt.ylabel('Preço')
plt.plot(Tamanho, preço, 'k.')
plt.axis([0, 25, 0, 25])
plt.grid(True)
plt.show()
```



Agora começaremos a fazer os calculos matemáticos.
Como a regressão linear faz parte do treinamento supervisionado, isto significa que precisamos ensinar o modelo a realizar as pevisões. Apenas relembrando que queremos aprender dos dados passado para então prever os dados futuros. Uma maneira de ensinar o nosso modelo é passando a variável independente (**Tamanho**) e mostrando o resultado que queremos prever (**Preço**), com isto nosso modelo irá aprender a realizar os cálculos para pegar a o tamanho e prever o preço futuro.

Treinando o Modelo

```
# Passamos a fórmula matemática
model = LinearRegression()
# Treinamos o nosso modelo
model.fit(Tamanho, preço)
```

Simples assim, ensinamos o nosso modelo a fórmula matemática `model = LinearRegression()` e passamos um valor de entrada real (**Tamanho**) e o valor que ele teria que prever (**preço**) , o nosso modelo se ajustou dentro fórmula para fazer as previsões.
`model.fit(Tamanho, preço).`

Homologando o Modelo

```
# Passando dados novos
Tamanho_Hom = [[19],[42],[50]]
preço_hom = [[22.60],[43.5],[51.60]]
predito = model.predict(Tamanho_hom)
print (predito)
```

Antes de colocar um modelo em produção, devemos realizar um teste de homologação, ou seja apresentar dados que modelo não conhece e verifcar com ele se saiu.
Criamos uma nova variável **Tamanho_Hom** e com novos valores e novos preços, a variável **preço_hom**.
A modelo então usará os cálculos que ele aprendeu anteriormente com estes novos dados.

Validando o Modelo

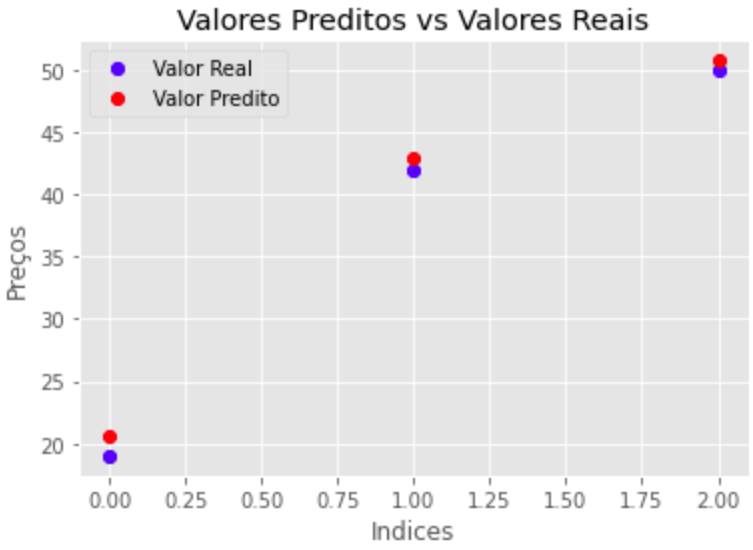
```
real      = pd.DataFrame(Tamanho_Hom, columns=['real'])
predito   = pd.DataFrame(model.predict(Tamanho_Hom), columns=['predito'])
resultado = pd.concat([real, predito], axis=1, sort=False)
resultado['diferença'] = resultado['real'] - resultado['predito']
print ("Linear Regression Resultados" )
print ("\nComparando os valores" )
print(resultado)
print('\nEficiência do treinamento:  ', model.score(Tamanho, preço))
# Calculando o erro médio
# Quanto mais próximo de zero melhor, significa que o modelo está errando
pouco
print('O erro médio foi:           ', mean_squared_error(preço_hom,
model.predict(Tamanho_Hom)))
print('A eficência do modelo foi:  ', model.score(Tamanho_Hom, preço_hom))


Linear Regression Resultados

Comparando os valores
   real    predito  diferença
0    19  20.515086  -1.515086
1    42  42.969828  -0.969828
2    50  50.780172  -0.780172

Eficiência do treinamento:   0.9100015964240102
O erro médio foi:            1.7666885280420317
A eficência do modelo foi:   0.9881643888341857
```

```
%matplotlib inline
plt.ylabel('Preços')
plt.xlabel('Indices')
plt.title('Valores Preditos vs Valores Reais')
plt.plot(real,'k.', color='b', marker='o', label='Valor Real')
plt.plot(predito,'k.', color='r', marker='o',label='Valor Predito')
plt.legend(['Valor Real','Valor Predito'])
```



Olhando para o resultado do nosso treino, o modelo acertou **91%**, ou seja dos valores que apresentamos ele previu 91% dos valores corretos. Como ele já conhece agora a fórmula matemática, e o coeficiente que explicarei em outro post, ao apresentar os dados para homologação, o acerto foi de **98%**. O valor médio do nosso erro ficou **R$1,76**.
Como a idéia da regressão é apresentar um valor futuro, sendo esta média aceitável para o negócio, basta agora implementá-lo na produção.

* Codigo mostrado aqui estão disponiveis em