**TEMPLATE: LIVRO-TEXTO**

**Unidade I**.

Ana Luiza Cerchiari de Andrade

### Objetivos da unidade

* Entender o que é regressão
* Entender o que é sobreajuste
* Como lidar com sobreajuste
* Conhecer linguagem Python e Matplotlib
* Compreender clusterização aplicada em similaridades
* Aprender a usar o K-NN
* Conhecer Matriz de Confusão

### Tópicos de estudo

* Ajustando um modelo de dados
  + Regressão Linear
  + Regressão Logística
* O que é sobreajuste? Como evitar o sobre ajuste.
  + Causas do sobreajuste
  + Soluções para evitar sobreajuste
* Similaridades, vizinho e agrupamento
  + Introdução a python
  + Visualizar gráficos com plot
  + Vizinhos e similaridade: Classificação.
  + Agrupamentos e similaridades: Clusterização
* Pensamento analítico de decisão
  + Matriz de Confusão
  + Sensibilidade e especificidade

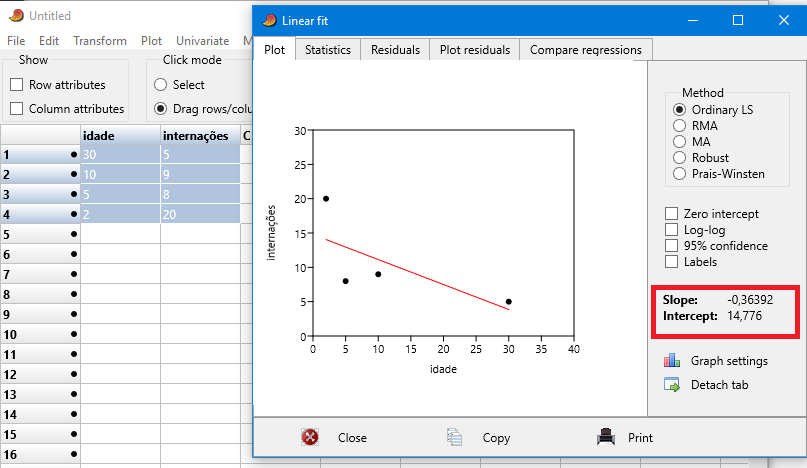
## Ajustando um modelo de dados

Este capítulo abordará o ajuste de dados com regressão linear, regressão logística e máquinas de vetores de suporte em um primeiro momento. Posteriormente, será ensinado tanto a linguagem Python¸ quanto suas bibliotecas, a saber: NumPy, Pandas, Matplotlib e Sklearn.

### Regressão Linear

Para analisar dados em regressão linear usaremos o programa Past. Mas antes disto é recomendável passar alguns conceitos. Observe que a Fig.1 existe uma base de dados (dataset) e um gráfico, onde o eixo x (horizontal) é a idade, e o eixo y (vertical) é a quantidade de internações.

**Figura 1**. Primeiro gráfico no programa PAST



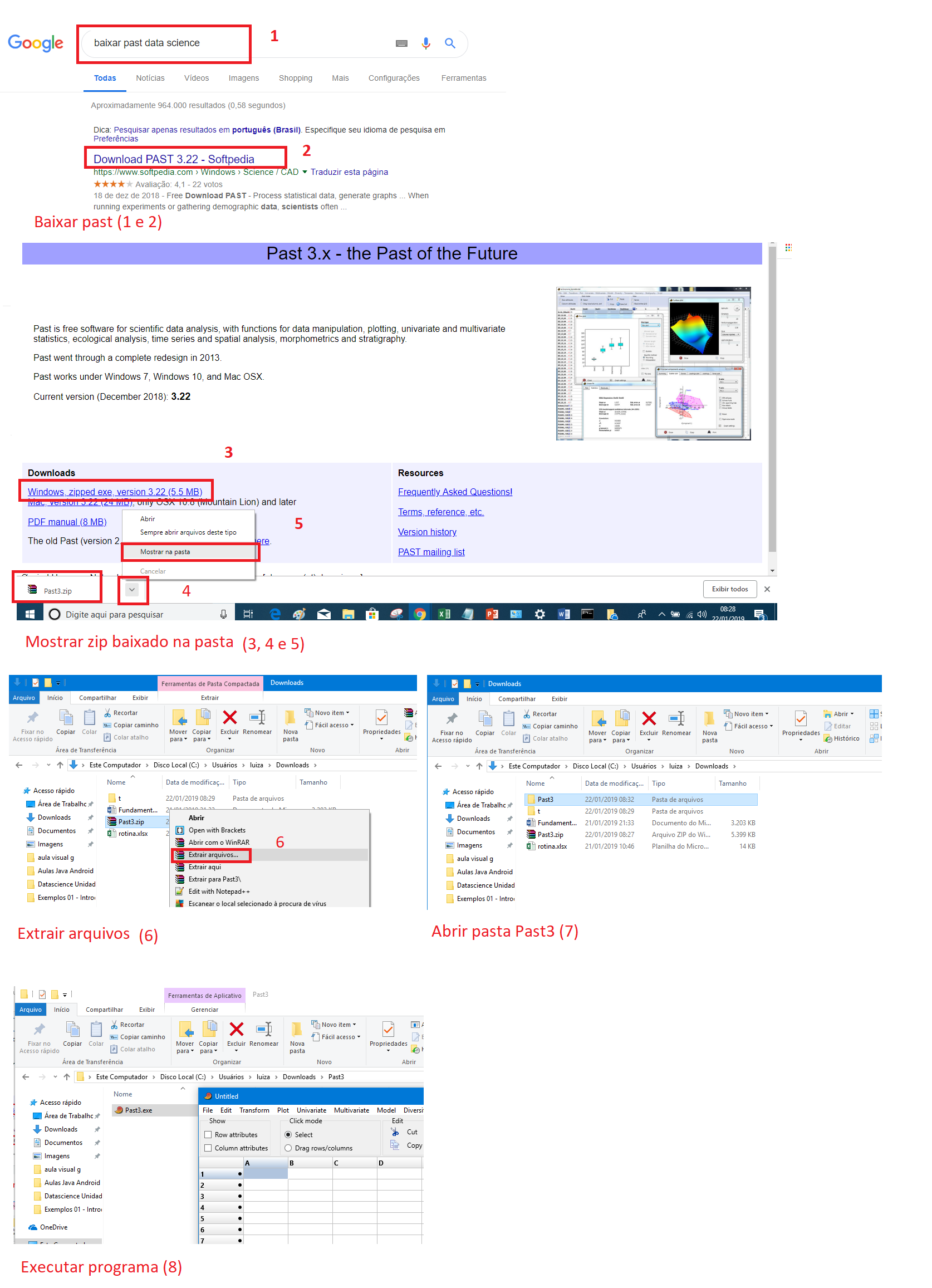
Pode-se observar alguns dados, quanto mais próximo a idade adulta, menor é a quantidade de internações, além disso a inclinação (slope) da linha é descendente e sua quantidade é -0,36. Quando o x é igual a 0, o ponto de inteceção é 14,776.

Assim temos alguns conceitos:

* **Slope** = inclinação que pode ser ascendente (+) ou descendente (-).
* **Intecept** = interceção quanto x for 0 (no exemplo y será 14,776 quando y for 0)

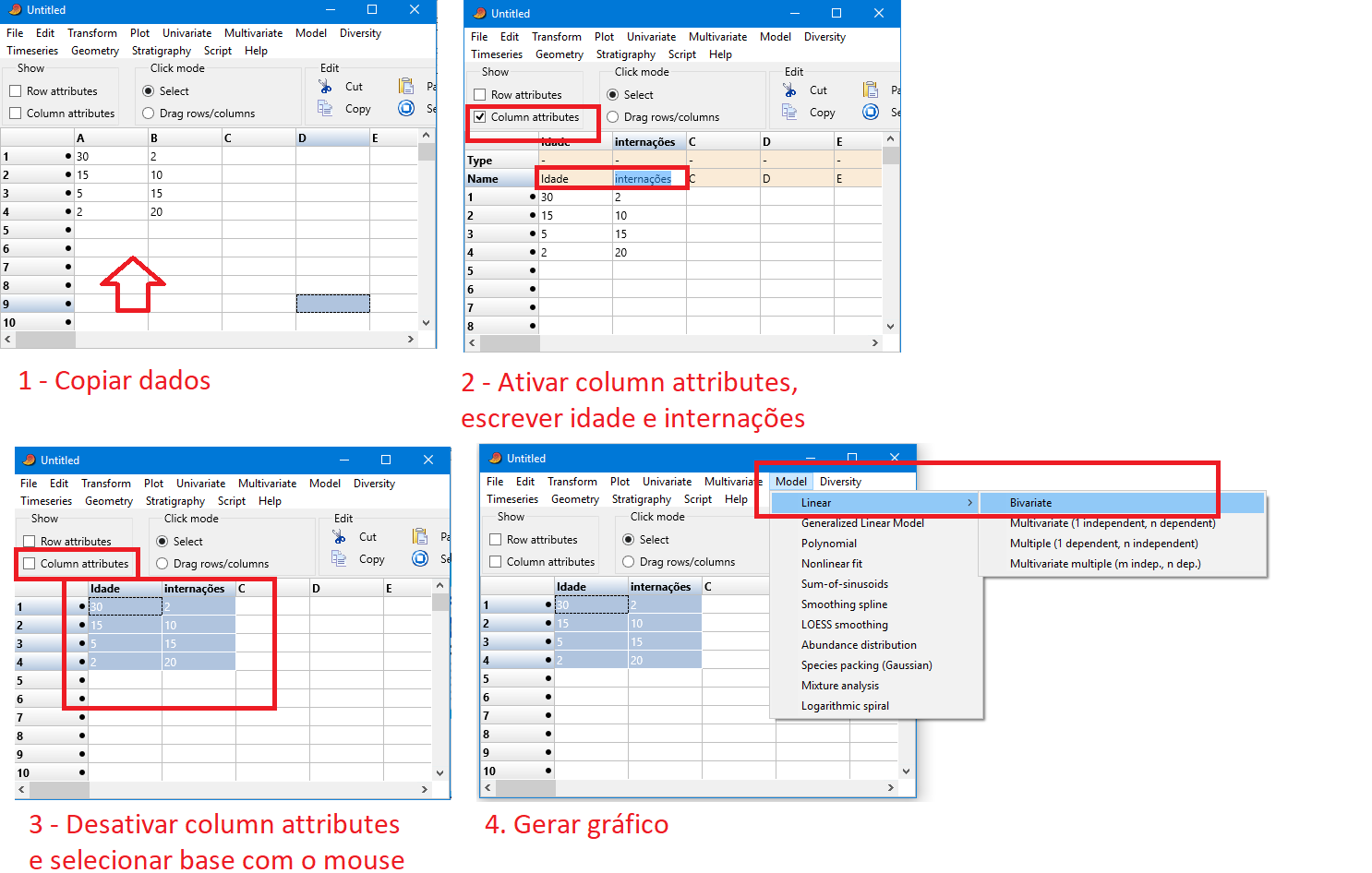
Para ter mais informações, baixe o programa conforme tutorial da Fig 2.

**Figura 2**. Tutorial para baixar Past

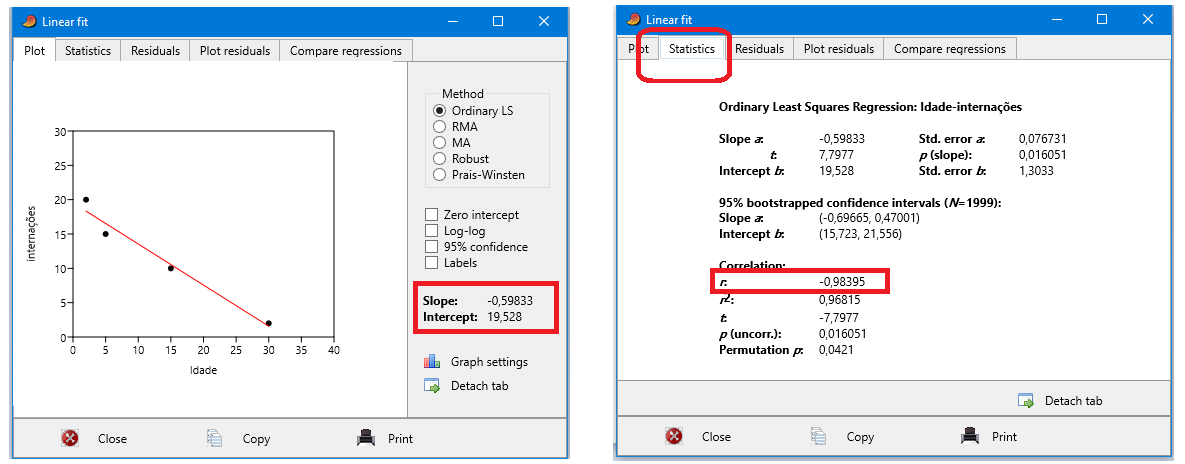


Agora será mostrado como gerar um gráfico na Fig 3 e como mostrar dados mais específicos na Figura 4.

**Figura 3.** Como gerar gráficos no Past



**Figura 4.** Como mostrar gráficos e dados estatísticos



Sabendo que intececption é o valor de y, quando x for igual a 0, e Slope é a inclinação, temos alguns conceitos novos. R é o quanto a idade tem relação com a internação, no exemplo acima a idade explica 0,98395 ou 98% a internação, ou seja, tem bastante relação entre idade e internação. Antes, no primeiro exemplo, na Figura 1, a relação de explicação era menor que 60%, embora não tenha sido demostrado, pode-se testar para confirmar. Segue um breve resumo dos dados:

* **Slope (inclinação)** = - 0,59833
* **Intecept (valor do y, quando x = 0)**= 19,528
* **R (Nível de dependencia e confiança na relação idade x internação) =** 0,98395 ou 98%

**Poderia parar por aqui, mas caso não tenha o programa, pode-se utilizar a fórmula** resolvida após a tabela 1.

**Tabela 1**. Somatórios das partes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Idades (x)** | **Internação (y)** | **x\*y** |  |  |
|  | 30 | 2 | 60 | 900 | 4 |
|  | 15 | 10 | 150 | 225 | 100 |
|  | 5 | 15 | 75 | 25 | 225 |
|  | 2 | 20 | 40 | 4 | 400 |
| **Soma** | **(soma x) 52** | **(soma y) 47** | **(soma x\*y)325** | **(soma )1154** | **(soma )729** |

A tabela de 4 linhas, logo n = 4.

r = logo

r = logo

r = logo

r = logo

r = logo r =

r = logo r =

Para aplicar esta análise em negócios, poderia fazer uma análise de correlação de 5 itens, por exemplo, ao avaliar número de produtos vendidos no eixo x, poderia avaliar como comparação preço, parcela e propaganda e analisar qual tem a maior correlação, tanto negativa quanto positiva, sabendo que quanto mais se aproxima de 100% ou de – 100% mais resposta trará, dependendo da análise de cada caso.

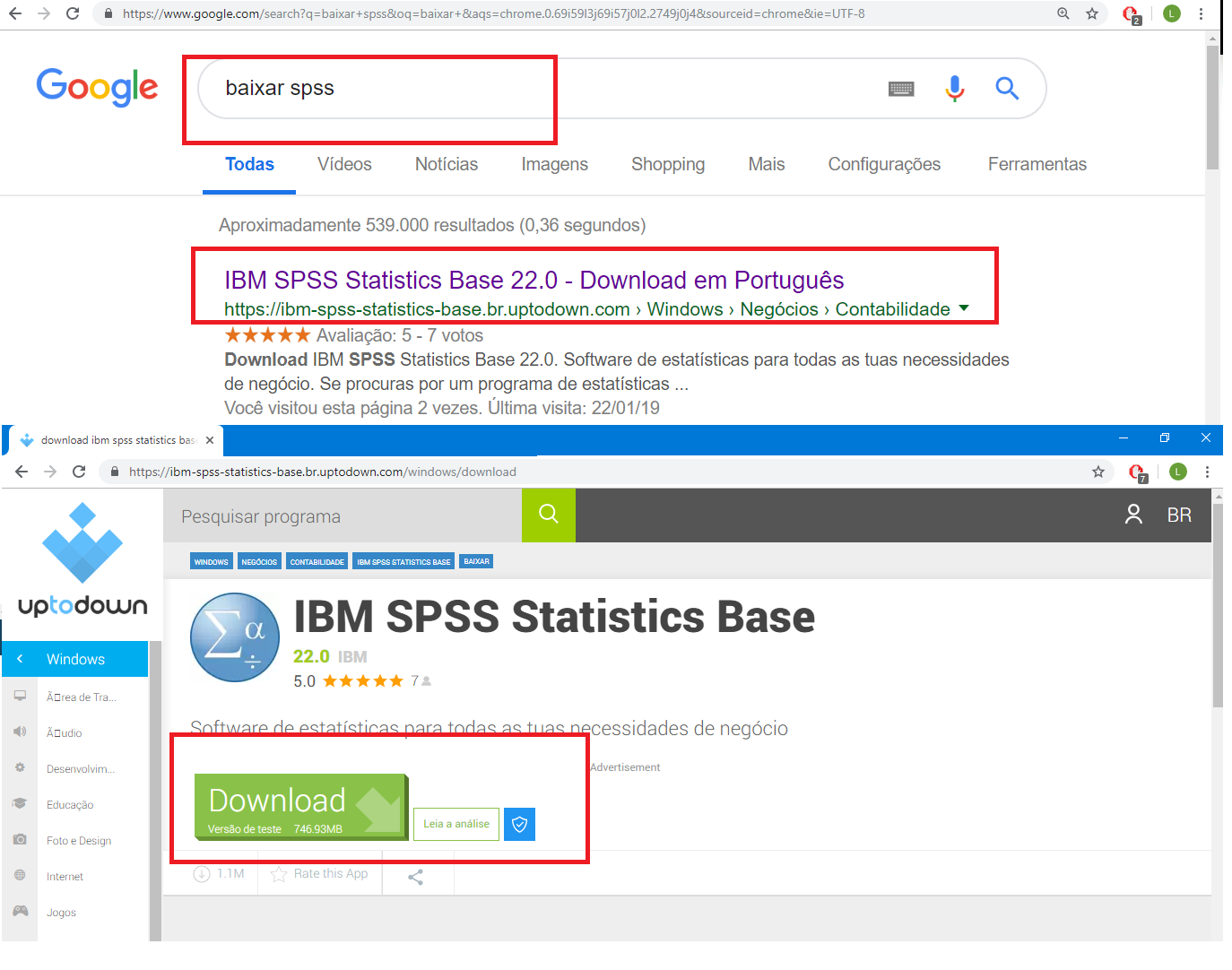
### Regressão Logística

A regressão logística emite dados categóricos, como desiste ou não, compra ou não, altera um plano sim ou não, e etc. Muito utilizado na área de saúde, utiliza diversas variáveis como idade, genêro, faixa-etária, classe social, renda, fumante e etc.

O primeiro passo é instalar o programa SPSS (usado em estatística) conforme Fig. 5.

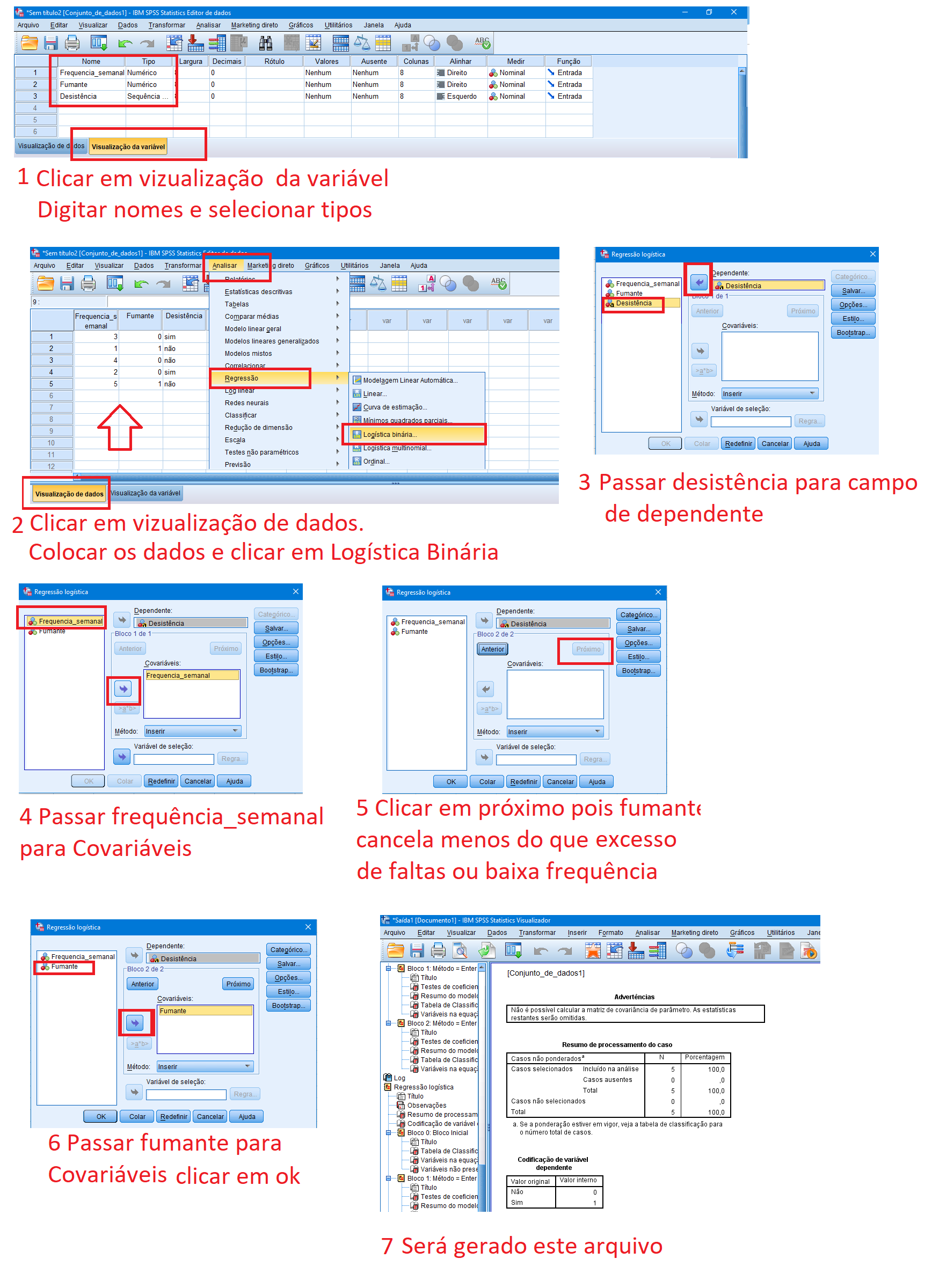
Após executar o arquivo baixado, basta apenas avançar até terminar a instalação, o computador será reiniciado.

**Figura 5.** Instalação de SPSS



O segundo passo é criar o banco de dados e fazer a análise de regressão binária, conforme Fig 6.

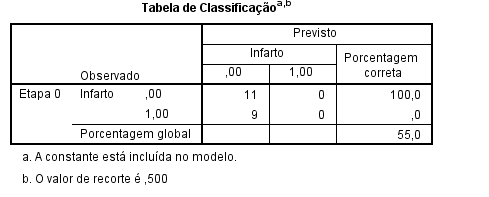
**Figura 6.** Criar banco de dados e fazer regressão logística



Agora faremos uma análise dos resultados:

Observe a Tabela 2, ela mostra um erro de análise de 45%, era para dar 100%, ela identificou que ninguém teria infarto, 1 representa tem infarto, o representa que não tem infarto, assim, está errado pois na tabela criada na Fig. 6, 9 pessoas teriam infarto , sendo a fração de 55% de acerto apenas.

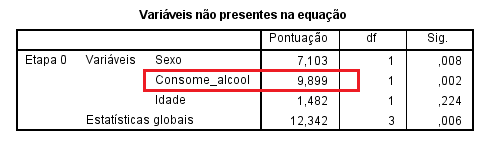
**Tabela 2**. Classificação em SPSS



Observe a tabela 3, ela mostra que o maior motivador de infarto, ou a variável mais relacionada é consome\_alcool, quanto penor o Sig. Mais influente é a variável.

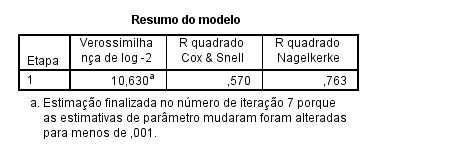
Tabela 3.

**Tabela 3**. Pontuação em variáveis



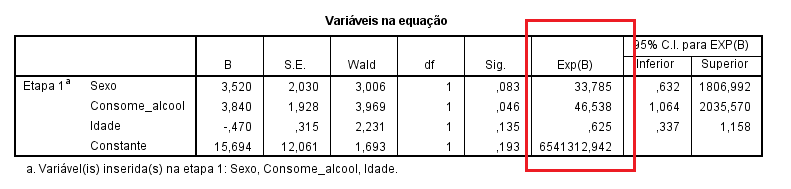
A tabela 4 mostra a pontuação do R, que avalia o percentual de correlação dos itens sobre o infarto, ou seja, o nível de certeza que os itens propiciam ou diminuem a probabilidade o infarto. No exemplo o R foi 73%, um valor de certeza mais alto do que baixo.

**Tabela 4**. Valor do r quadrado



E por fim a tabela 5 mostra a influência de cada item para gerar infarto.

**Tabela 5**. Influência de cada item sobre o infarto.



## O que é sobreajuste? Como evitar o sobre ajuste.

Sobreajustes são ocorrências aleatórioas que pareçam padrões, mas não generalizam, ou seja, não podem ser aplicados a outros modelos, pois na verdade pareciam padrões, mas induziram ao erro. Mas o que é generalizar? Generalizar é a capacidade de usar um modelo para outros modelos com precisão, mas as vezes ele não é 100% aplicado, sendo forçado a **sobreajustar, ou seja, usar indevidamente um padrão acreditando que os outros dados obedecerão a este padrão**. Assim de um lado sobreajuste é forçar, e de outro, generalizar é usar corretamente um modelo para descobrir uma informação. **Concluíndo, generalização é a capacidade de aplicar um molde com êxito em outros dados.** Como fazer análises que levem a generalizações úteis?

Em uma tabela e deve-se fazer treinos com um conjunto de linhas, e aplicar em outro conjunto de linhas, para ver quão eficaz foi a análise preditiva...Mas, análise o que é predição? Análise preditiva é prever, por exemplo, ao analisar (treinar) cinco compras, descobre-se um padrão, e avalia se este padrão ocorre em outras 5 linhas (testar o treino). O objetivo é descobrir a **precisão na amostra**, sendo tudo isso um **conjunto de teste**.

### 2.1 Causas do sobreajuste

Para evitar o sobreajuste deve-se tomar cuidado com algumas possíveis ocorrências, **um dos problemas está em simplificar... O outro está nas mudanças que induzem ao erro.**

**O primeiro problema de simplificação ou viés**, ocorre na seguinte situação: Quanto mais complexo um conjunto de dados, **mais funcionam na amostra, mas tem mais défict para uma análise do futuro**, assim, tem tem mais sobreajuste, eles decoram os dados e não entendem bem os padrões, assim, não generalizam bem em dados futuros. Fazer isso gera erros de "viés", ou seja, simplificar demais.

O segundo problema, denominado **erros de variância** são pequenas mudanças nos dados, que mudam bastante os parâmetros, por exemplo, um gráfico área preço do Iphone cresce conforme o somatório das tecnologias, mas ele pode parar de crescer por causa de fatores como limites de preço, falta de necessidade. Conforme gráfico 1.

**Gráfico 1.** Como mudanças de padrões afetam a análise de comportamento, a influência da variância no erro

Para descobrir se um dado tem sobreajuste é só avaliar em dados novos, se errar muito ele está com sobreajuste, mas como definir em dados novos? A primeira solução é separar os dados de uma tabela em dois grupos, assim, pode-se definir os parâmetros e modelos e treinar com dados do 1 trimestre e utilizar e testar no 2 trimestre. A segunda solução, é separar 70% a 75% dos dados de uma tabela para treino para definir o modelo e 25% para teste para analisar a aplicabilidade, caso haja muitos dados, utiliza-se mais para treino, mais ou menos 85% a 90%, e 15% a 10% para teste. Nesta análise pode-se descobrir itens de influência e testar nos outros dados, conforme Fig. 7 que treina em 7 (70%) e testaria em 3 (30%).

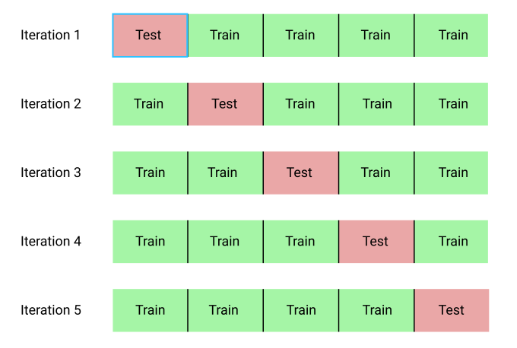


**Figura 7**. Dividindo datasets **(**Adaptado deVASCONCELLOS, 2017)

### Soluções para evitar sobreajuste

Uma sugestão para acertar mais é quando for fazer o teste, utilizar variável semelhante, por exemplo, imagine uma tabela com 200 linhas, destas 200 linhas existem 100 linhas de pessoas com alta empregabilidade, deve-se fazer o treino com 75 pessoas que tenham alta empregabilidade, e testar os dados com 25 pessoas que tenham alta empregabilidade, isso evita erros. Seria uma amostra com extratos.

Para gerar um modelo, será necessário criar os parâmetros (parâmetros maiores) e hiperparâmetros (que nada mais do que a regularização ou os ajustes menores). Uma técnica para aplicar o teste é a K Validação Cruzada, onde o K é o número de de quebras. Observe que a Fig. 8 o número escolhido foi 5. Assim pode-se análisar vários parâmetros, tanto com os programas já apresentandos, buscando por exemplo o R de um item sobre o outro e analisando se este parâmetro se repete nos outros itens.



**Figura 8.** Exemplo de validação cruzada. (Adaptado de VASCONCELLOS, 2018)

Cabe dizer que a validação mais utilizada é 10, sendo k = 10 popularmente.Assim analisa-se um parâmetro linha por linha, depois analisa-se outro parâmetro linha por linha, e mais outro parâmetro linha por linha e analisa-se o parâmetro que tiver o melhor desempenho de previsão ou influência, fazendo por exemplo uma média.

Analisando cada teste na figura 8, pode-se em cada validação, utilizar parâmetros de árvores de decisão e cálculo de entropia para descobrir o ganho de informação, pode-se escolher os nós das árvores mais influentes. Posteriormente pode-se analisar com base no r, ou seja, a relação que um item exerce sobre outro, para descobrir o r pode-se utilizar o programa Past conforme já foi demonstrado na Fig. 5 e nas equações sobre r . Posteriormente pode-se analisar linha a linha através do programa SPSS e analisar itens que possuem pontuação mais alta por exemplo, conforme tabela 5. Após avaliar com entropia e árvore de informação, r em PAST , pontuações em SPSS e outras técnicas que ainda serão citadas, avalia-se qual delas teve a melhor perfomance naquele conjunto de dados.

A validação cruzada ( 5 k fold cross validation) treinará linha por linha, excluindo um item,

## Similaridades, vizinho e agrupamento

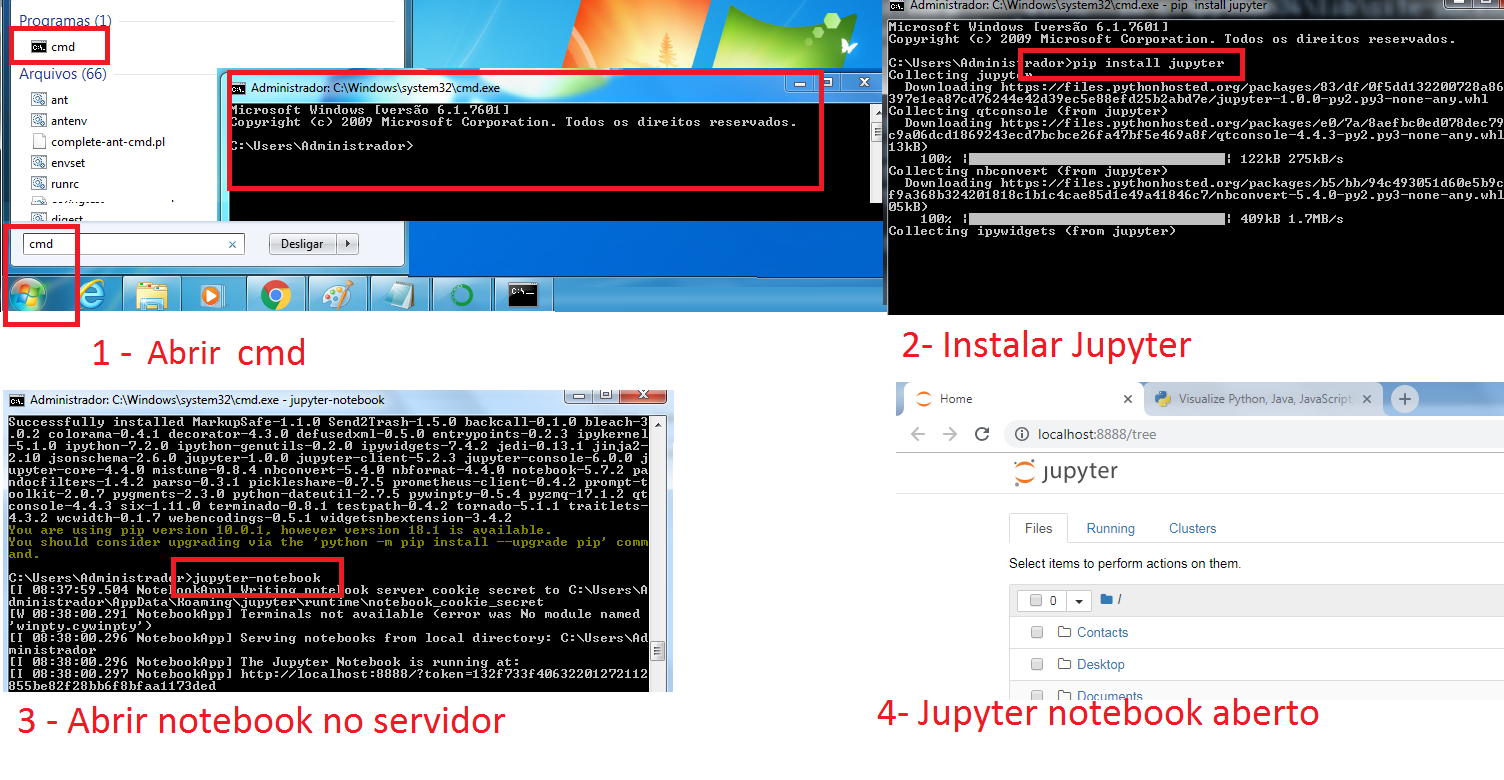
Algumas análises podem ser feitas em Python, outras podem ser feitas em Weka, neste momento será passado uma introdução sobre análise de dados em python e em matplotlib, e depois será passado os conceitos de similaridades, vizinhos e agrupamentos.

### 3.1 Introdução a python

Para utilizar python é necessário seguir alguns procedimentos:

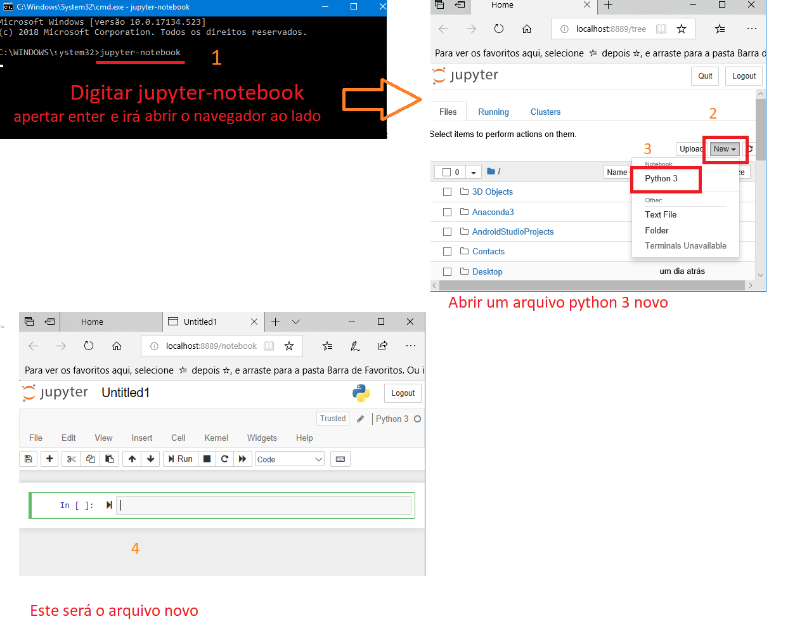
* Instalar jupyter-notebook (Fig. 9)
* Como abrir um arquivo em jupyter notebook. (Fig. 10).

Para isso o usuário precisa já ter instalado o python, conforme comentado em outro momento.



**Figura 9**. Instalar jupyter-notebook

Agora abra o jupyter-notebook e crie um arquivo novo

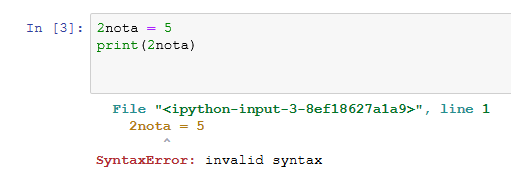


**Figura 10**. Como abrir jupyter-notebook

Vamos começar a mexer em python!

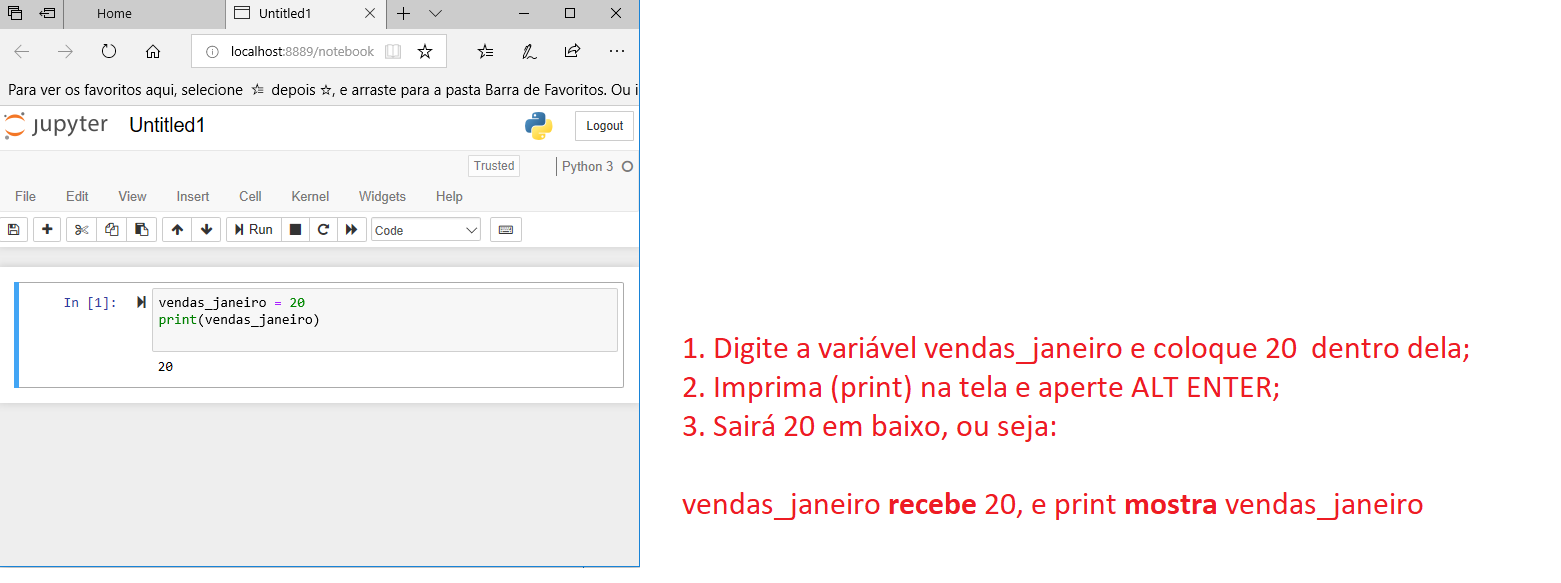
Em *Python*, **variáveis** são chamadas de **rótulos** e receberão valores,além disso, possuem algumas condições ao serem criadas, v***ariáveis são apelidos para outras palavras! Nós criaremos uma palavra e colocaremos valores dentro desta palavra, mas não podem começar com números nem podem conter espaços.***

**Somente observe** um exemplo de erro de sintaxe na Fig. 11, ao tentar colocar número na primeira letra de uma variável (2nota):



**Figura 11.** Erro ao criar variável

Vamos criar a primeira variável, conforme fig 12. Após digitar os comandos na linha contendo a palavra **in[1]** e apertar *alt enter*, o resultado sairá em baixo, faça este exercício:



**Figura 12.** Primeira variável criada

Além de variáveis, outro tema relevante em *Python* são os **operadores**. Quando um usuário entra no site e clica uma quantidade de vezes, analisa-se o **comportamento probabilístico** desse usuário, ao captar os dados relativos ao clique, assim, estratégias direcionadas de marketing e pesquisa podem ser feitas, tanto para desenvolvimento de novos produtos quanto para arquitetura da informação no site e posicionamento dos objetos.

Existem **operadores de comparação, operadores de cálculo e operadores lógicos**. Na Tabela 6, pode-se analisar os **três tipos**.

**Tabela 6**. Tipos de operadores

|  |  |  |
| --- | --- | --- |
| OPERADORES DE COMPARAÇÃO | OPERADORES DE CÁLCULO | OPERADORES LÓGICOS |
| == igualdade  > maior que  ! diferente  >= maior ou igual | + adição  - Subtração  \* multiplicação  \*\* Exponenciação  / Divisão  // Divisão inteira | And - e  Or - Ou  Not – Não |

Exemplo 1) Probabilidade de **cliques** de usuários com **operador de comparação(>)**:

**In []**

**cliques\_julio = 10**

**cliques\_marcela = 20**

**resultado = cliques\_julio > cliques\_marcela**

**print(resultado)**

**Out[]**

**Resultado**: **False**

Análise estatística: Marcela está mais propensa a consumir determinado produto.

Exemplo 2) Probabilidade de **gostos assinalados** com **operador de comparação(==):**

**In []**

**gosto\_marcos = ('futebol')**

**gosto\_joao = ('futebol')**

**mesma\_propaganda = gosto\_marcos == gosto\_joao**

**print(mesma\_propaganda)**

**Out[]**

**Resultado: True**

Análise estatística: Os dois usuários são o público-alvo de propagandas de futebol.

Exemplo 3) **Operador de cálculo** traçando a **média de compras** de um usuário:

**In [] compra1\_joao = 50**

**compra2\_joao = 30**

**compra3\_joao = 40**

**numero\_decompras = 3**

**total\_joao = compra1\_joao + compra2\_joao + compra3\_joao**

**media\_joao = total\_joao / numero\_decompras**

**print(media\_joao)**

**Out[] Resultado: 40**

Análise estatística: João compra em média R$ 40,00 por pedido.

Exemplo 4) Exemplos de operadores lógicos ao decidir se deve mandar propaganda ou não, a partir de um campo de checagem vazio. Quando o usuário marca a opção de não receber propaganda como *True*, a variável “mandar e-mail” será oposta por causa do “*not*”, sendo assim, será *False*, ou seja, “não mandar e-mails”.

**In []**

**nao\_receber\_propaganda\_voo = True**

**mandar\_emails\_voo = not nao\_receber\_propaganda\_voo**

**print(mandar\_emails\_voo)**

**Out[]**

**Resultado: False**

**Analises condicionais** auxiliam o cientista de dados a selecionar melhor os usuários e dados, gastando menos tempo.

***If***equivale a “**se”** um critério for atendido,***else***equivale a “**senão**”. O exemplo a seguir reflete como a divulgação pode ser direcionada, onde, se o usuário tiver **mais** ou **igual** a **18 anos**, deve-se divulgar **carros**, senão, deve-se divulgar **bicicletas**:

**In []**

**idade = 10**

**if idade >= 18:**

**print('Divulgar carros')**

**else:**

**print('Divulgar bicicletas')**

**Out[] Resultado:**

Divulgar bicicletas

#fazendo relação de rede social

O comando union junta dados de dois conjuntos de dados:

**amigos\_gabriel = {** **'Marcio', 'Flavia', 'Marcos'}**

**amigos\_daniel = {** **'Flávio', 'Alexandre'}**

**uniao = amigos\_gabriel.union(amigos\_daniel)**

**print(uniao)**

**Resultado:**

{'Alexandre', 'Flávio', 'Marcos', 'Marcio', 'Flavia'}

O comando intecection junta pega apenas dados que se repetem de dois conjuntos de dados:

**In []**

**amigos\_gabriel = { 'Marcio', 'Marcela', 'Marcos'}**

**amigos\_daniel = { 'Marcela', 'Alexandre'}**

**interceccao\_amigos = amigos\_gabriel.intersection(amigos\_daniel)**

**print(interceccao\_amigos)**

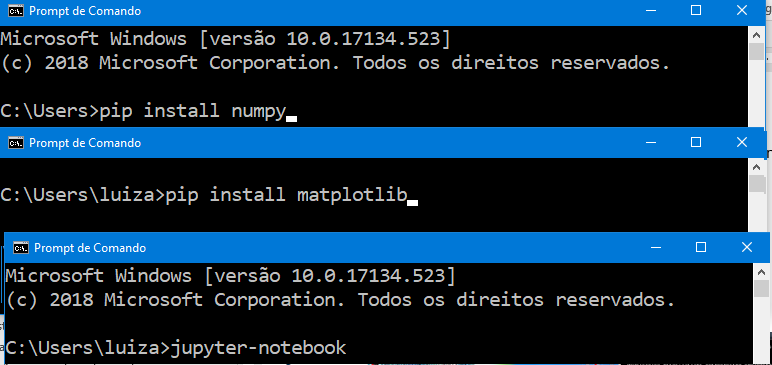
**Out[]Resultado:**

{'Marcela’}

Análise estatística: Inserir no site propagandas diferentes de vôo e analisar cliques para descobrir os gostos do cliente.

### 3.2 Visualizar gráficos com plot

A biblioteca Matplotlib ajuda na vização de dados, o primeiro passo para visualizar gráficos é instalar o *Matplotlib* pelo CMD, por isso instale a biblioteca numpy, matplotlib e abra o jupyter-notebook, conforme figura 13.



**Figura 13**. Instalação de Matplotlib

Para importar *Matplotlib* e Numpy utiliza-se as seguintes sintaxes::

In[1]:

**import** **numpy** **as** **np** *# depois de executar o pip install matplotlib no cmd*

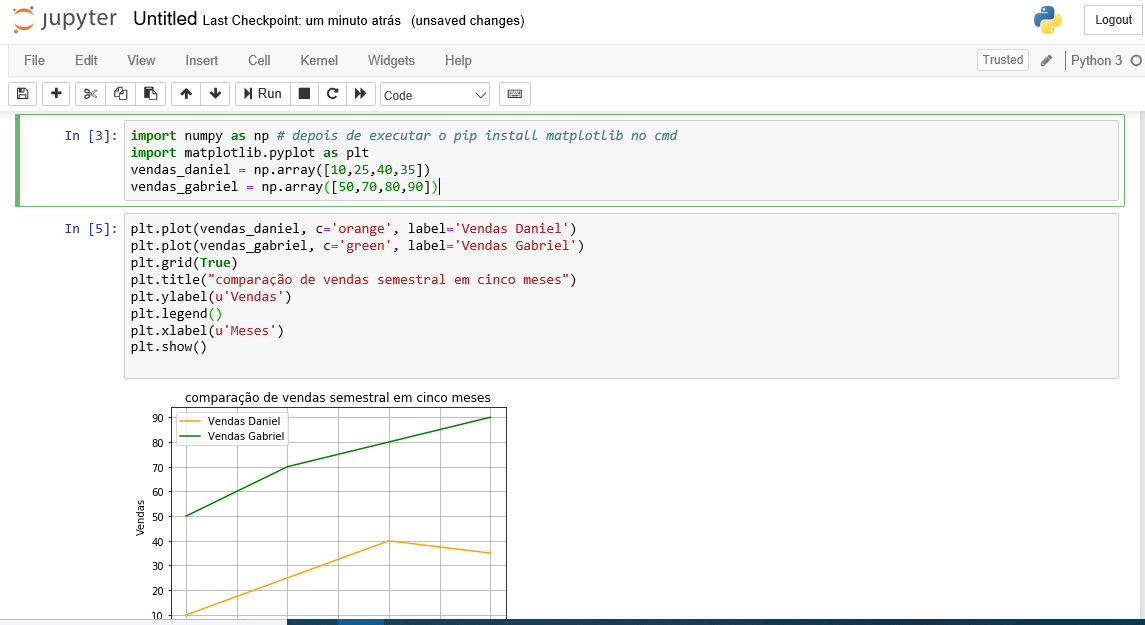
**import** **matplotlib.pyplot** **as** **plt**

%**matplotlib** inline

Para deixar o gráfico com aparência mais legível, recomenda-se utilizar as propriedades a seguir:

* c = Cor
* title = Título
* grid = Grade de fundo (pode ser True ou False)
* ylabel = Legenda vertical
* xlabel = Legenda horizontal
* label = Texto da legenda
* plt.legend() = mostrar legenda dentro do gráfico

A seguir será criado dois arrays:



**Figura 14**. Criando um gráfico com listas

A decoração do gráfico é fundamental para torna-lo menos cansativo. Existem vários tipos de marcadores (marker), assim como linhas diferentes, tanto pontilhadas como contínuas. **A tabela 7 tem valores diferentes que o programador pode colocar e testar**.

Segue, portanto, algumas propriedades:

* marker = Tipo de símbolo
* ms = Tamanho do símbolo
* lw = Largura da linha
* ls = Tipo de linha
* markeredgecolor = Cor da borda do símbolo
* markerfacecolor = Cor do símbolo

**Tabela 7**. Variações iconográficas para gráficos em Matplotlib (SANTEE, 2014)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Tipos de cor (c)** | **Caractere** | | Azul | ‘b’ | | Verde | ‘g’ | | Vermelhos | ‘r’ | | Ciano | ‘c’ | | Magenta | ‘m’ | | Amarelo | ‘y’ | | Preto | ‘k’ | | Branco | ‘w’ | | |  |  | | --- | --- | | **Tipos de marcador (marker)** | **Caractere** | | Ponto | '.' | | Pixel | ',' | | Círculo | 'o' | | Triângulo | 'v' | | Triângulo | '^' | | Triângulo | '<' | | Triângulo | '>' | | Y para baixo | '1' | | Y para cima | '2' | |  |  | | |  |  | | --- | --- | | **Tipos de linha (ls)** | **Caractere** | | Linha cheia | '-' | | Linha tracejada | '--' | | Linha traço-ponto | '-.' | | Linha pontilhada | ':' | |

No script abaixo, *Orange* significa laranja, g é abreviação para *green* (verde em inglês), as demais coisas podem ser analisadas de acordo com o gráfico 3 e a tabela 7.

In[6]:

**plt.plot(vendas\_daniel, c='orange', label='Vendas Daniel', ls='-', marker='s', ms=12)**

**plt.plot(vendas\_gabriel, c='g', label='Vendas Gabriel', ls='--', marker='^', ms=8,)**

plt.grid(**True**)

plt.title("comparação de vendas semestral em cinco meses")

plt.ylabel(u'Vendas')

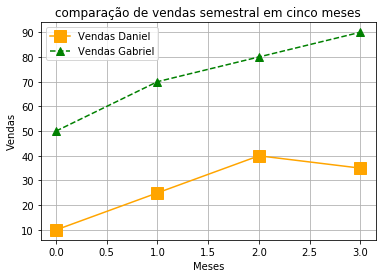
plt.xlabel(u'Meses')

plt.legend()

plt.show()

Note que o ms é diferente, mudando o tamanho do símbolo.

**Gráfico 3**. Gráfico com legenda, cores e títulos



O gráfico 4 possui as propriedades **markeredgecolor** que muda a cor da borda do símbolo**, markerfacecolor** que muda a cor do fundo do símbolo e o **drawstyle= steps-post** que deixa as linhas retas:

In[6]:

plt.plot(vendas\_daniel, c='orange',ls='--', marker='P', ms='10', lw='3',**markerfacecolor='red', drawstyle='steps-post', markeredgecolor='green'**) plt.plot(vendas\_gabriel, c='g', label='Vendas Gabriel', ls='--', marker='^', ms=8, **markerfacecolor='red', drawstyle='steps-post', markeredgecolor='green')** plt.grid(**True**)

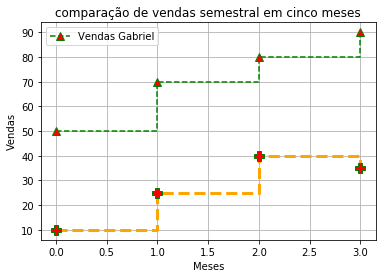
plt.title("comparação de vendas semestral em cinco meses") plt.ylabel(u'Vendas')

plt.xlabel(u'Meses')

plt.legend()

plt.show()

**Gráfico 4**. Gráfico markeredgecolor (cor da borda de marcadador) verde, markerfacecolor (cor do fundo do marcador) vermelho e linhas retas (drawstyle=steps-post).



### Vizinhos e similaridade: Classificação.

Para recomendar um item é necessário descobrir o vizinho mais próximo, é necessário calcular a Distância Euclidiana entre um item e outros itens escolhidos. Para descobrir utiliza-se a seguinte fórmula:

esta fórmula também pode ser descrita da seguinte maneira:

Analisando o ibope de um canal, criou-se o seguinte dataset:

**Tabela 8**. Ibope por horário

|  |  |  |  |
| --- | --- | --- | --- |
|  | Manhã | Tarde | Noite |
| Comida | 0.8 | 0.6 | 0.1 |
| Futebol | 0.1 | 0.5 | 0.9 |
| Notícias | 0.3 | 0.3 | 0.4 |
| Cultura | 0.5 | 0.7 | 0.3 |
| Lazer | 0.8 | 0.6 | 0.1 |
| Política | 0.1 | 0.5 | 0.9 |
| Saúde | 0.8 | 0.6 | 0.1 |

A tabela 9 mostra a aplicação da fórmula, onde as etapas 1 e 2 são os valores dos itens escolhidos, a etapa 3 é a diferença, a etapa 4 eleva ao quadrado, a etapa 5 soma os três itens de resultado, e a etapa 6 faz a raíz quadrada. Neste exemplo, futebol e comida possue menor distância, logo são mais similares. Pode-se usar também a Distância de Manhattan no lugar da Distância Euclidiana.

**Tabela 9.** Descobrindo a Distância Euclidiana

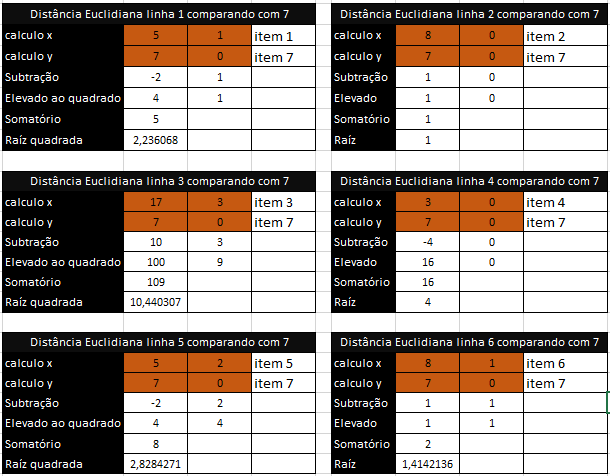
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Etapas | Função | Valor M. | Valor T. | Valor N. | Etapas | Função | Valor M. | Valor T. | Valor N. |
| 1 | Comida | 0,8 | 0,6 | 0,1 | 1 | Comida | 0,8 | 0,6 | 0,1 |
| 2 | Notícias | 0,3 | 0,3 | 0,4 | 2 | Futebol | 0,1 | 0,5 | 0,9 |
| 3 | Subtração | 0,5 | 0,3 | -0,3 | 3 | Subtração | 0,7 | 0,1 | -0,8 |
| 4 | Elevado ao quadrado | 0,25 | 0,09 | 0,09 | 4 | Elevado | 0,49 | 0,01 | 0,64 |
| 5 | Somatório | 0,43 |  |  | 5 | Somatório | 1,14 |  |  |
| 6 | Raíz quadrada | 0,66 |  |  | 6 | Raíz | 1,07 |  |  |

Além de encontrar itens mais similares, pode-se fazer a análise K-NN (*Nearest Neightbor* ou vizinhos mais próximos). Para isso, supondo que um item novo foi lançado, irá avaliar com base nos vizinhos mais próximos, qual serial a saída, se compraria ou não. Será feito um cálculo de Distância Euclidiana para cada item.

**Tabela 10**. Relação de compras com dados de cliques e perguntas

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Cliques** | **Perguntas** | **Compras** |
| **1** | 5 | 1 | Não |
| **2** | 8 | 0 | Sim |
| **3** | 17 | 3 | Sim |
| **4** | 3 | 0 | Não |
| **5** | 5 | 2 | Não |
| **6** | 8 | 1 | Sim |
| **7** | 7 | 0 | ? |

Tabela 11. Cálculos de distância Eucliadiana para cada item

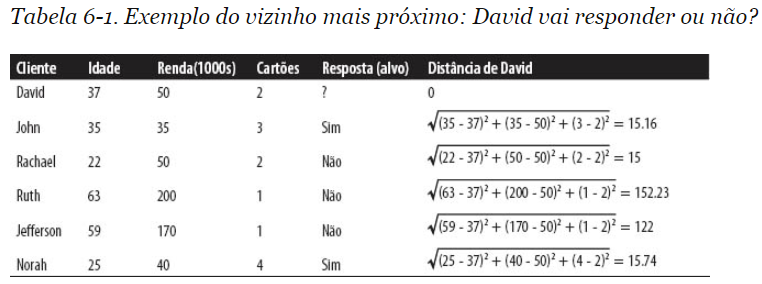


Deve-se escolher o K, isto é a quantidade de vizinhos, se escolher k=1 o vizinho mais próximo seria a linha 5 (resultado de 1), logo a resposta de probabilidade de compra seria sim, mas k = 1 é pouco, e números pares empatam, então neste exemplo poderia escolher k= 3, os três vizinhos mais próximos seriam:

1. Linha 2 (resultado de 1) = Sim
2. Linha 6 (resultado de 1,41) = Sim
3. Linha 1 (resultado de 2,23) = Não

Então a suposição seria **sim** como resposta, uma vez que 1 contém não em resposta, mas as linhas 2 e 6 contém sim, prevalecendo o sim. Porém existem diversos estudos sobre como escolher o K ideal, sendo contudo, ímpar, consenso entre os cientistas.

Como desvantagens, pode-se afirmar que este método não constrói um modelo e sim um cálculo, tornando a aplicação em dados que tendem a aumentar trabalhosa, outro problema é que ele não filtra dados com maior e menor influência sobre o resultado, há pesquisas que saber a quantidade de filhos realmente não interessa, ou o peso, e isso varia de caso para caso. A tabela 12, exemplifica o raciocínio realizado:



**Tabela 12**. Distância Euclidiana para K-NN, David vai responder? (FOSTER E FAWCETT (2016, p. 69)

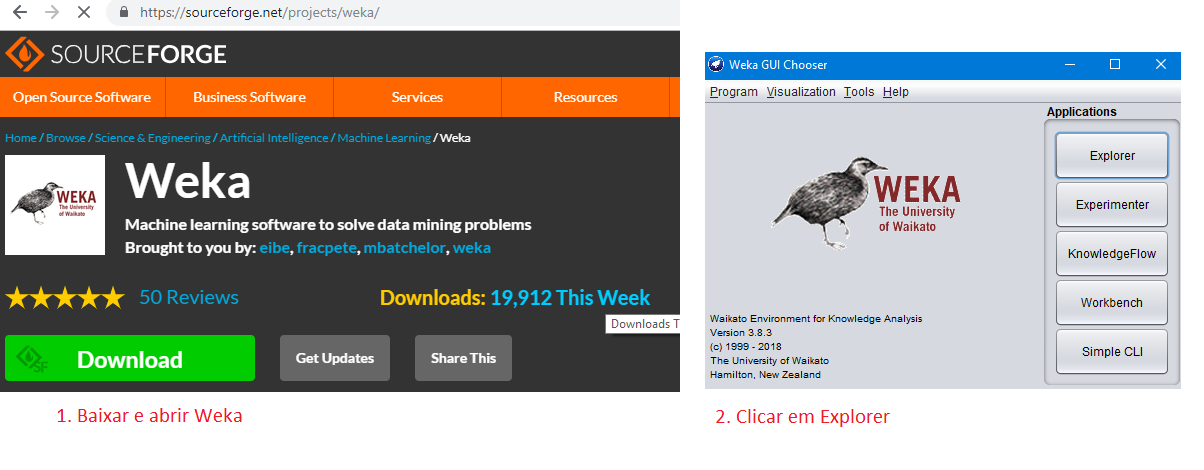
**Box**: **Assista**

Assista a “Moneyball - *O homem que mudou o jogo”*, sob direção de Jonah Hill e Philip Seymour, conta a história de um gerente de time que teve a brilhante ideia de potencializar o time através de cálculos matemáticos em computador.Este filme foi baseado no livro “Moneyball: The Art of Winning an Unfair Game” (LEWIS E BEANE, 2003)

### Agrupamentos e similaridades: Clusterização

Grupos são utilizados em diversas circunstâncias de análise e de perdição, como já citado. Analisando subgrupos naturaispode-se criar grupos de notícias, produtos, hábitos e até fracassos em negócios...Um problema descoberto perde a força! Para analisar grupos, pode-se utilizar o programa Weka, para fazer é necessário baixar e instalar conforme Fig. 14.

**Figura 14.** Baixar weka e clicar em explorar

. 

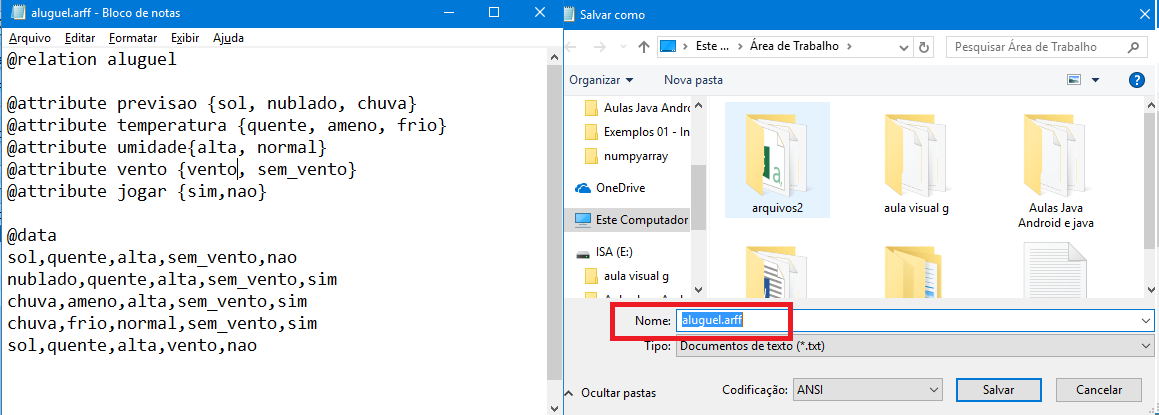
Observe a tabela 10, ela contém cinco atributos e uma saída. A saída responde a dúvida se terá gente alugando quadra ou não.

**Tabela 13**. Jogar tennis

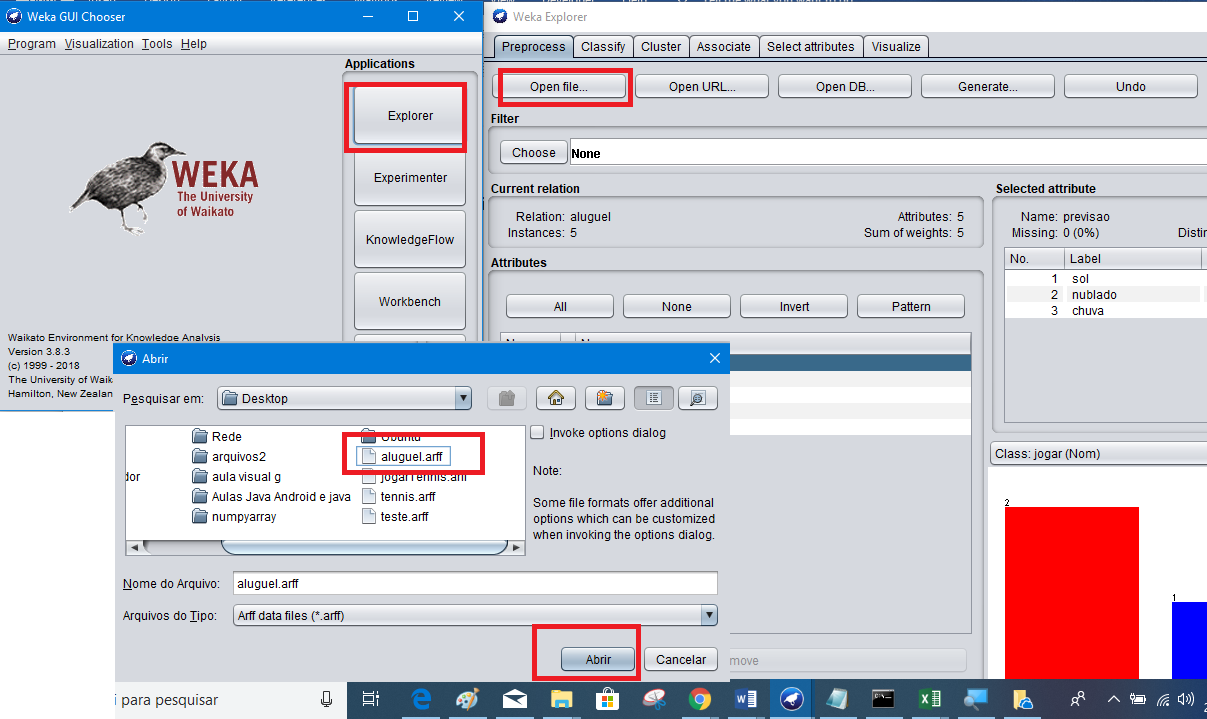
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TEMPO** | **TEMPERATURA** | **HUMIDADE** | **VENTO** | **JOGAR?** |
| Sol | quente | alta | sem\_vento | não |
| Nublado | Quente | alta | sem\_vento | sim |
| Chuva | Ameno | alta | sem\_vento | Sim |
| Chuva | Frio | normal | sem\_vento | Sim |
| Sol | Quente | Alta | Vento | Não |

O tipo de arquivo que Weka lê é arff, para criar abra um bloco de notas do windows e siga os passos das Fig. 15, 16, 17, 18 1 19. Para avaliar cluster é necessário ver caso por caso.

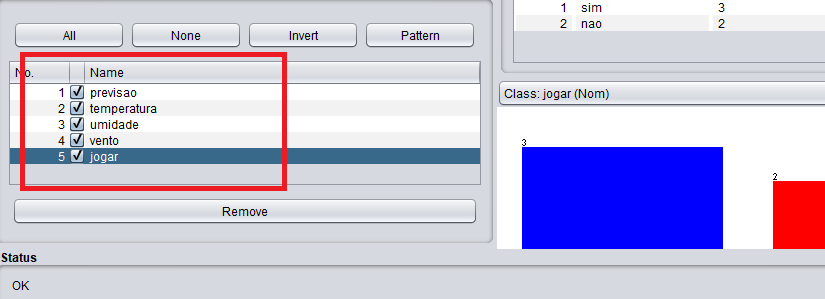
**Figura 15**. Etapa 1 - Criar este arquivo no bloco de notas e salvar como aluguel.arff



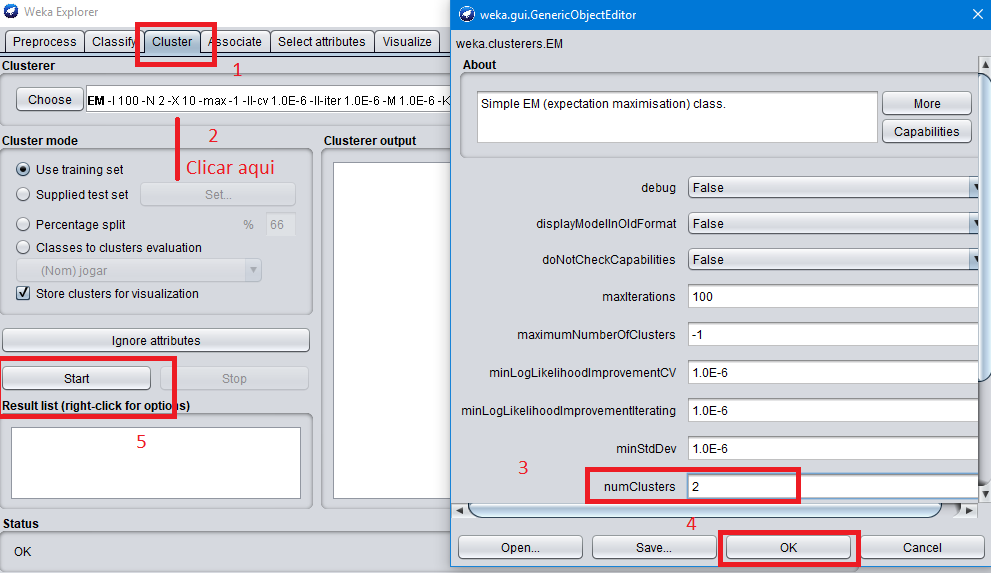
**Figura 16**. Etapa 2 – Abrir arquivo aluguel.arff



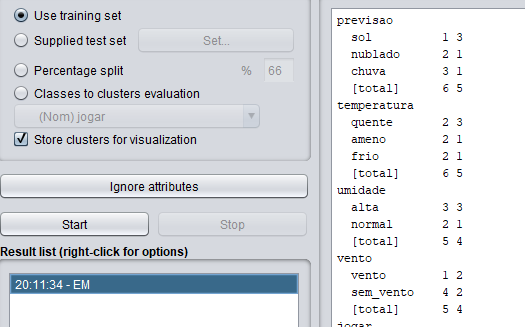
**Figura 17**. Etapa 3 – Selecionar dados



**Figura 18** – Clicar em clusters e gerar dois grupos.



**Figura 19** – Demonstração final de cluster



## Pensamento analítico de decisão

Até este momento foi demonstrado como usar técnicas de análises tanto presentes quanto preditivas, foi abordado validação cruzada, cálculo de r e regressão, K-NN, clusterização e etc. Mas como analisar se estas técnicas são eficientes? A esta pergunta as próximas etapas objetivam responder.

### Matriz de Confusão

Cada negócio, cada dataset, cada conjunto de dados e cada caso possui uma lógica interna, formada por fatores econômicos, geográficos, demográficos, psicológicos, mercadológicos, processuais, legais, governamentais, climáticos... Ou seja, cada caso é um caso, cada planilha de dados coletados possui uma probabilidade interna, uma lógica interna, e descobrir esta lógica é a base da ciência de dados. Para descobrir isso pode-se recorrer a matriz de confusão. Posso comparar k=5 com k=7, ou K-NN com cluster, e ver qual é mais eficaz. Supondo que **estamos em março**, ao analisar por exemplo um **conjunto de dados de janeiro** e fazermos uma análise de k=5, **aplicamos este treino** em **dados que ocorreram em fevereiro,** e analisamos o **quanto acertamos**.

A tabela 14 é um exemplo de cálculos de K-NN com k = 3, então em março testamos o k=3 treinando em janeiro e aplicando em fevereiro, e o que der certo baterá com a tabela 15, ou seja, a tabela 14 são cálculos e a tabela 15 é a realidade, e a partir daí pode-se fazer a matriz de confusão do k=3.

**Tabela 14.** Cálculos de K-NN com k=3, treinando em janeiro, supondo em fevereiro.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Analise SUPOSTA K =3 | | | | |
|  |  | **Cliques** | **Perguntas** | **Compras** |
| Aanálise real de janeiro | **1** | 5 | 1 | não |
| **2** | 8 | 0 | sim |
| **3** | 17 | 3 | sim |
| **4** | 3 | 0 | não |
| **5** | 5 | 2 | não |
| **6** | 8 | 1 | sim |
| Análise com K-NN de dados de fevereiro | **7** | 7 | 0 | sim |
| **8** | 6 | 1 | não |
| **9** | 9 | 0 | sim |
| **10** | 9 | 1 | sim |

**Tabela 15.** Realidade de fevereiro.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Analise REAL de fevereiro | | | | |
|  |  | **Cliques** | **Perguntas** | **Compras** |
| Análise real de janeiro | **1** | 5 | 1 | não |
| **2** | 8 | 0 | sim |
| **3** | 17 | 3 | sim |
| **4** | 3 | 0 | não |
| **5** | 5 | 2 | não |
| **6** | 8 | 1 | sim |
| Análise real de dados de fevereiro | **7** | 7 | 0 | sim |
| **8** | 6 | 1 | não |
| **9** | 9 | 0 | Não (análise K-NN errou pois estimou sim) |
| **10** | 9 | 1 | sim |

A partir desta análise pode-se fazer a matriz de confusão, onde vamos avaliar qual sim deveria ser sim (True Positive), qual sim deveria ser não (False Positive), qual não deveria ser não (True Negative) e qual não deveria ser sim (False Negative).

Tabela 15. Matriz de confusão.

|  |  |  |  |
| --- | --- | --- | --- |
| ANÁLISE DE K =3 | | | |
| Previsto | | | |
| Realidade |  | Positivo | Negativo |
| Positivo | Linha 7 e Linha 10 (TP) | Linha 9 (FN) |
| Negativo | Linha 9 (FP) | Linha 8 (TN) |

Após fazer análises de vários valores de k, pode-se chegar a conclusão de qual k é melhor.

### 4. 2 Sensibilidade e especificidade

Sensibilidade é mostrar a probabilidade de um teste dar positivo

Sensibilidade é mostrar a probabilidade de um teste dar negativo

# Sintetizando

Nesta unidade foi visto como a **Regressão Linear** funciona e de que forma o “r” contribui para a relação proporcional de uma reta, neste exemplo foi utilizado o programa PAST. Outro ponto semelhante, foi o conceito de **Regressão Logística**, o qual foi realizado no programa SPSS, dados categóricos e dicotômicos.

O conceito de **Sobreajuste** e **Generalização** foi destrinchado e explicado, trazendo a tona a diferença entre erros de **viés** e erros de **variância** que impedem a correta generalização dos dados, e como solução, foi proposta a técnica famosa de **Validação Cruzada.**

Ao estudar análises de dados foi feita uma introdução a Python, contendo os passos iniciais, como **print, operadores, *if else, union e interscetion***. Ainda foi abordada a biblioteca Matplotlib, a qual serve para plotar gráficos, e nela foi passado conceitos de **ls, c, *title, grid, ylabel, xlabel, plt.legend, marker, ms, markeredgecolor e markerfacecolor.***

Estudos demonstraram como descobrir a **Distância Euclidiana**, tanto em cálculos e quanto em demonstrações com tabelas, para então passar para a explicação de **vizinhos com K-NN**. Além do mais através do programa Weka foi passado como fazer um cluster.

Finalmente e fechando o conteúdo, explanou-se o modus operandi do funcionamento da **matriz de confusão**, bem como a demonstração de fórmulas para descobrir a **sensibilidade e a especificidade**.

# Palavras-chave

* + - 1. *Regressão Linear*
      2. Regressão Logística
      3. *PAST*
      4. SPSS
      5. Sobreajuste
      6. Generalização
      7. Python
      8. Matplotlib
      9. K-NN
      10. Classificação
      11. Agrupamento
      12. Matriz de Confusão
      13. Distância Euclidiana
      14. Weka
      15. Negócios
      16. Dados
      17. Cluster
      18. Preditivo
      19. Treino
      20. Teste

# Referências bibliográficas

VASCONSCELOS, Ligia M. e CARVALHO, Luiz C. A**plicação de Regras para Mineração de Dados na Web.** Goias. Universidade Federal De Goias: 2004. Disponível em: http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF\_004-04.pdf. Acesso em: 16/01/2004.

VASCONSCELOS, Paulo. Como saber se seu modelo de *Machine learn*ing está funcionando mesmo. 2018. Dísponívem em: <https://paulovasconcellos.com.br/como-saber-se-seu-modelo-de-machine-learning-est%C3%A1-funcionando-mesmo-a5892f6468b>. Acesso em: 24/01/2019.

FOSTER, Provost e FAWCETT, ***Data Science* para negócios**: O que você precisa saber sobre mineração de dados e pensamento analítítico de dados? Rio de Janeiro: Altabooks, 2016.