



Motivação, conceitos iniciais e ferramentas

Ciência de Dados II

Professor: Gabriel Machado Lunardi
gabriel.lunardi@ufsm.br

Roteiro da aula

1. Motivação para a análise de dados
2. Conceitos iniciais
3. O processo de KDD
4. Ferramentas
5. Introdução à linguagem Python
6. Ambiente Google Colaboratory

Roteiro da aula

1. Motivação para a análise de dados
2. Conceitos iniciais
3. O processo de KDD
4. Ferramentas
5. Introdução à linguagem Python
6. Ambiente Google Colaboratory

Motivação para a análise de dados

- ✓ Grande quantidade de dados sendo gerados nos mais variados setores e armazenados na mais variadas formas:
 - Dados de comércio eletrônico
 - Dados de navegação na internet (redes sociais, plataformas de streaming, etc.)
 - Dados de compras de clientes em grandes lojas de departamentos, supermercados
 - Dados de transações bancárias, ou de cartão de crédito
 - Dados de geolocalização
 - Dados do nosso próprio smartphone a todo segundo
 - Dados de smart cities, smart homes, etc.

Motivação para a análise de dados

- ✓ Empresas sedentas por conhecimento potencialmente escondido nesses dados para orientar o seu processo de tomada de decisão.



“

Data is the new oil”

Clive Humby

“Data is the new oil. It’s valuable, but if unrefined it cannot really be used (...) so must data be broken down, analyzed for it to have value.”



Motivação para a análise de dados

- ✓ Por isso, o cientista/analista de dados é considerado o profissional da década, cujo perfil é o seguinte:
 - ✓ Habilidade analítica
 - ✓ Capacidade de investigação
 - ✓ Conhecimento do negócio
 - ✓ Habilidades de programação

Roteiro da aula

1. Motivação para a análise de dados
2. Conceitos iniciais
3. O processo de KDD
4. Ferramentas
5. Introdução à linguagem Python
6. Ambiente Google Colaboratory

Dados

- ✓ **Dados:** é a **matéria-prima** obtida de uma ou mais fontes. Representa algo que não tem sentido quando apresentado sozinho ou de forma desordenada. Ex: minério de ferro.

valor

220

grandeza

Volts

Informação

- ✓ A **informação** é a **ordenação e organização dos dados** de forma a transmitir **significado** e compreensão dentro de um determinado contexto. Seria o conjunto ou consolidação dos dados de forma a fundamentar o conhecimento. Ex: aço produzido pela indústria a partir do minério de ferro.

valor

220

grandeza

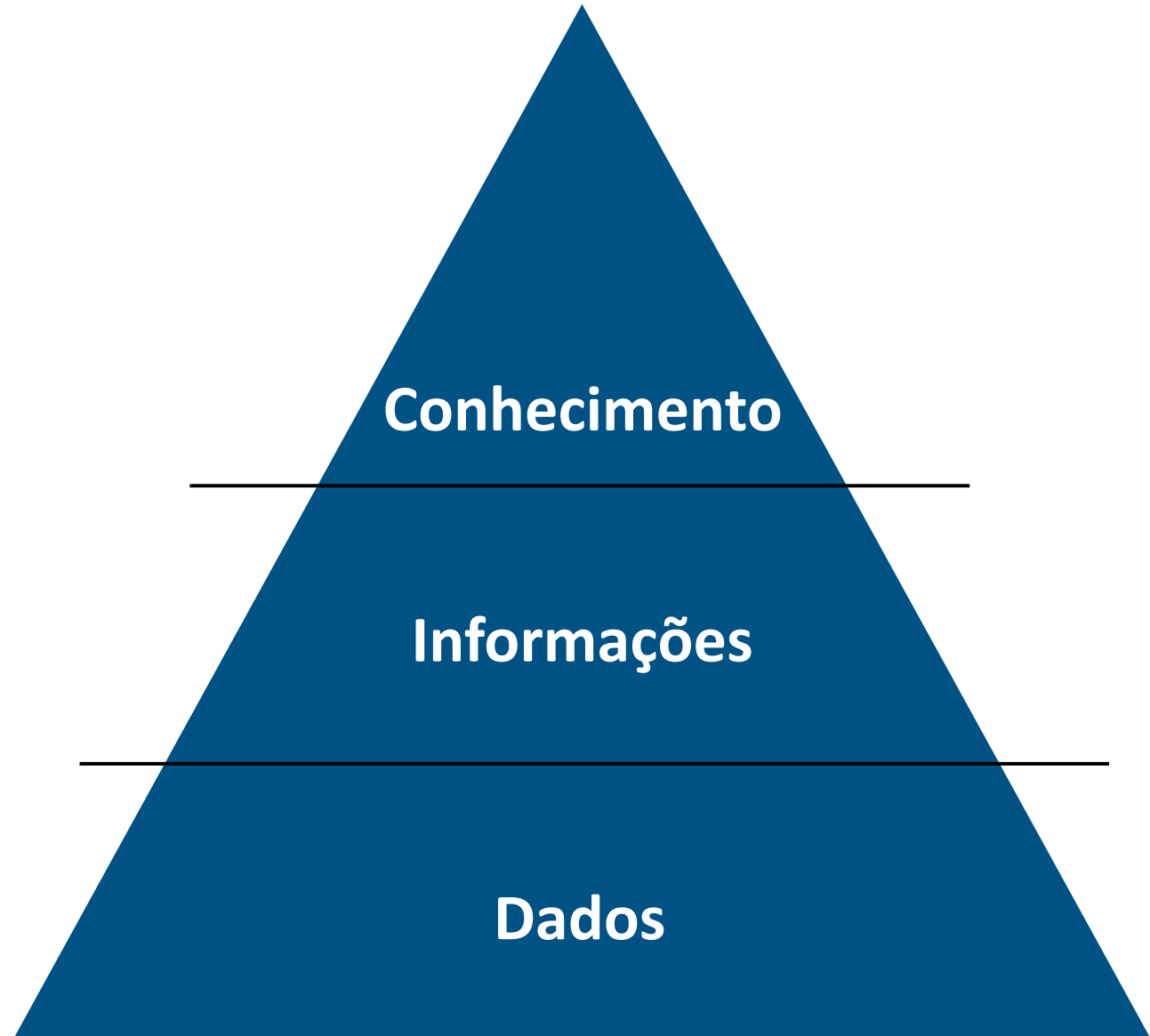
Volts

Conhecimento

- ✓ O conhecimento representa a **compreensão do significado da informação** com a possibilidade de **aplicar compreensão no mundo real**. Ex: o aço utilizado para construir inúmeras estruturas.



**Como ir da
base ao topo
da pirâmide?**



**Como ir da
base ao topo
da pirâmide?**



Análise de dados!

Business Intelligence (BI)

Mineração de dados

Business Intelligence (BI)

- ✓ Utiliza diferentes técnicas da **estatística clássica e de consultas em bases de dados** para gerar informações operacionais que subsidiem a **tomada de decisão**.
- ✓ Exemplo: estatísticas ao final do dia de qual produto foi mais vendido, menos vendido, etc, para repor estoques e detectar tendências de compra.

Mineração de dados

- ✓ **Também emprega técnicas da estatística clássica**, porém, vai além, pois lança mão de algoritmos de **aprendizado de máquina** para descobrir **padrões nos dados**.
- ✓ Um **padrão** é considerado um evento ou uma combinação de eventos numa base de dados que ocorre com mais frequência.
- ✓ Padrões são guiados pelos dados e geralmente refletem os próprios dados;
 - Exemplo: se “salário < T, então a pessoa não efetuou o pagamento”



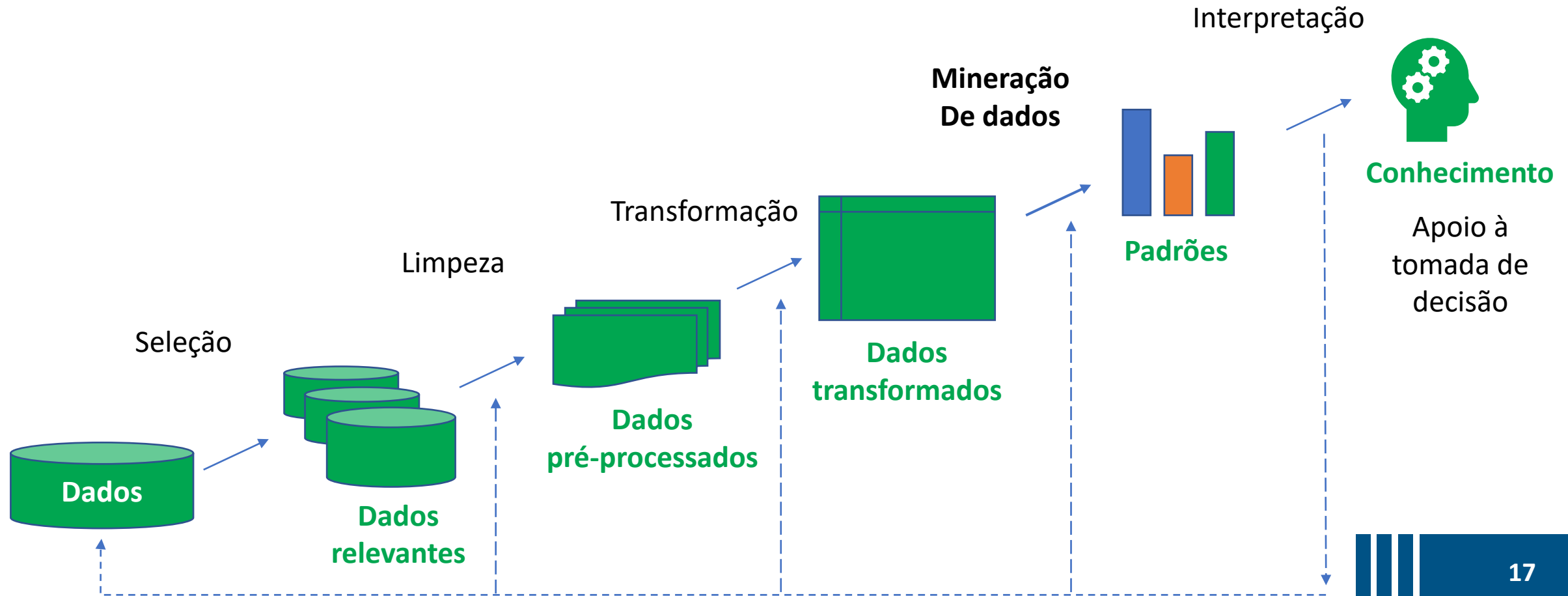
A expressão surgiu na década de 90, na comunidade de bancos de dados, e se refere à **etapa de análise de dados** de um processo maior chamado de descoberta de conhecimento em bases dados (Knowledge Discovery in Databases – **KDD**).

Roteiro da aula

1. Motivação para a análise de dados
2. Conceitos iniciais
3. O processo de KDD
4. Ferramentas
5. Introdução à linguagem Python
6. Ambiente Google Colaboratory

O processo de KDD

“É um processo de várias etapas, não trivial, **interativo** e **iterativo**, para a identificação de **padrões** válidos, novos e potencialmente úteis a partir de um grande conjunto de dados” (FAYYAD, 1996).



Dados, o que são?

- ✓ Dados: é uma coleção de **objetos** e seus **atributos**;
- ✓ Um **atributo** é uma propriedade ou **característica** de um objeto
 - ✓ Exemplos: a cor dos olhos de uma pessoa, a sua temperatura, etc.
 - ✓ Atributos também são conhecidos como variáveis, campos, características.
- ✓ Uma coleção de atributos descreve um objeto
 - ✓ Objetos também são conhecidos como registro, caso, exemplo, entidade, instância.

The diagram illustrates a data table. A horizontal bracket above the table is labeled 'atributos' in green. A vertical bracket to the left of the table is labeled 'objetos' in green. The table itself has four columns: 'Nome', 'Idade', 'Renda', and 'Pagador'. It contains seven rows of data.

| Nome | Idade | Renda | Pagador |
|-------|--------|-------|---------|
| João | <30 | Média | Bom |
| Ana | 41..50 | Alta | Bom |
| Pedro | 41..50 | Alta | Bom |
| Maria | 41..50 | Baixa | Ruim |
| Paulo | <30 | Baixa | Ruim |
| Aldo | >60 | Alta | Ruim |

Dados, qual o tipo?

✓ **Atributo discreto**

- ✓ Tem um número finito de valores.
- ✓ Exemplos: CEP, CPF, o conjunto de palavras em um document.
- ✓ Freqüentemente representado como uma variável inteira.

✓ **Atributo contínuo**

- ✓ Tem um número real como valor do atributo.
- ✓ Exemplos: temperatura, altura, peso.
- ✓ Variáveis contínuas são tipicamente representadas como variáveis de ponto-flutuante.

Dados, onde estão?

✓ Estruturados

- ✓ Bancos de dados relacionais, planilhas, arquivos CSV...

✓ Semi estruturados

- ✓ Arquivos JSON, XML, HTML...

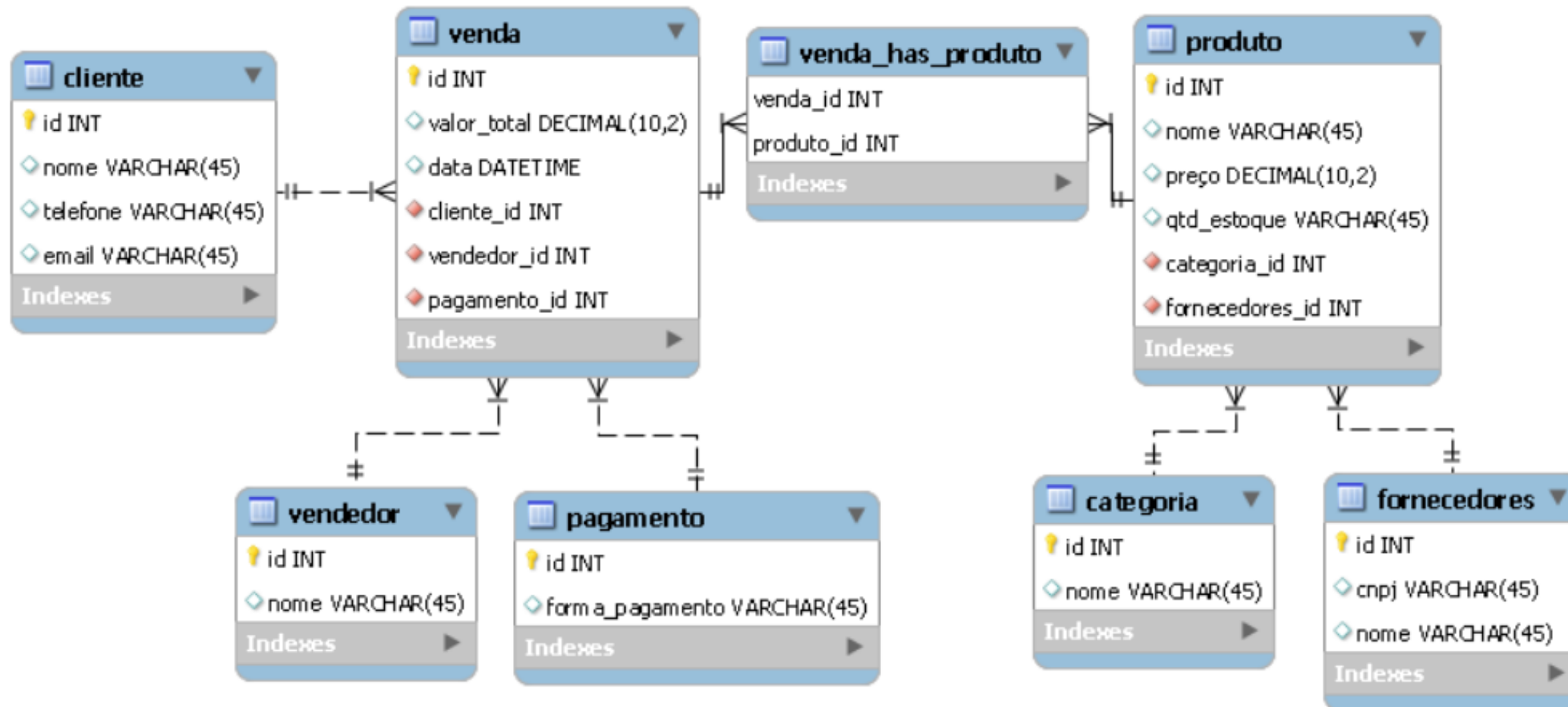
✓ Não estruturados

- ✓ Texto livre: nesta categoria o KDD é chamado de **KDT** (Knowledge Discovery in Text)
- ✓ Áudio
- ✓ Vídeo
- ✓ Imagem

Dados, onde estão?

✓ Estruturados

✓ Bancos de dados relacionais (exemplo)



Dados, onde estão?

✓ Semi estruturados

✓ Arquivos JSON (exemplo)

```
{  
  "image": "Cheryl-Carter.jpg",  
  "firstname": "Cheryl",  
  "lastname": "Carter",  
  "company": "Skyble",  
  "email": "ccarter@gmail.com",  
  "phone": "2-(017)772-7449",  
},  
{ ...
```

Etapas de KDD

- 1 Seleção
- 2 Limpeza
- 3 Transformação
- 4 Mineração
- 5 Interpretação

Pré-processamento ou
preparação dos dados

1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

- ✓ Os dados são selecionados a partir de uma ou mais fontes, estruturadas, não estruturadas ou semiestruturadas.
- ✓ Descrição dos dados:
 - ✓ Número de atributos e instâncias em cada arquivo
 - ✓ Tipos e faixas de valores dos atributos
 - ✓ Significado de cada atributo e sua importância para o objetivo do projeto
 - ✓ Estatísticas básicas para alguns atributos
 - ✓ média, desvio padrão, máximo, mínimo, moda, etc.
- ✓ Aplicar técnicas de visualização de dados nesta etapa auxilia muito no **entendimento dos dados**, além da descrição.

1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

✓ Seleção de atributos (redução de dimensionalidade)

✓ Razões para fazer:

- ✓ Requisitos de tempo e espaço
- ✓ Simplicidade do modelo gerado
- ✓ Relevância dos atributos
- ✓ Redundância entre atributos

✓ Como fazer:

- ✓ Manualmente
- ✓ Através de algoritmos: mais de 30 algoritmos

1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

✓ Tem por objetivo garantir a **qualidade dos dados**. Quanto maior for a qualidade dos dados, melhor será o resultado final.

✓ Algumas atividades para isso são:

- ✓ Padronização de dados: formato de datas, abreviaturas, valores de atributos (ex. sexo: M ou F, 0 ou 1, Mas e Fem, ...)
- ✓ Eliminação de duplicatas
- ✓ Tratamento de valores ausentes
- ✓ Exclusão instâncias
- ✓ Completar valores ausentes
 - ✓ Complemento manual
 - ✓ Complemento com valor constante global: ex: “desconhecido”
 - ✓ Complementar com o valor mais provável
 - ✓ Complementar com o valor médio do atributo

1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

✓ Transformação de valores simbólicos para numéricos

✓ Ex: sexo Masculino = 0 e Feminino = 1

✓ Discretização de atributos

✓ transformar em faixas – ex: idade

✓ Criação de novos atributos.

✓ Ex: área = comprimento x largura

✓ Ex: idade = faixa de idades



- ✓ As três etapas, referentes à preparação dos dados (ou pré-processamento), possuem total relação com o **nível de qualidade dos dados** que podemos elevar.

1 Seleção

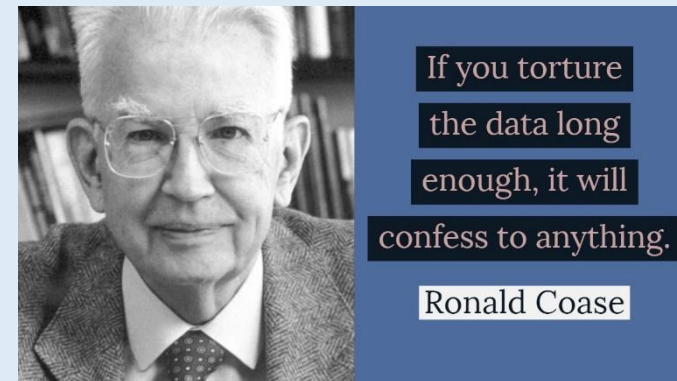
2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

- ✓ **Cerca de 85%** do esforço envolvido no processo de KDD está ligado à seleção, limpeza e transformação dos dados.
- ✓ Portanto, torture os dados (transforme-os o quanto for preciso) até que eles confessem!



1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

✓ Selecionar a(s) **tarefa(s)** de mineração de acordo com o problema levantado dentro do projeto de análise de dados:

- ✓ Associação
- ✓ Classificação
- ✓ Agrupamento
- ✓ Predição
- ✓

✓ Escolha do algoritmo dentro da tarefa adotada.

✓ Aplicação do algoritmo para construir o modelo.

1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

- ✓ Não existe uma única tarefa que resolva todos os problemas de mineração de dados. Cada tarefa oferece vantagens e desvantagens e a familiaridade com elas é necessária para facilitar a escolha de uma delas de acordo com o problemas que o analista de dados está buscando solução.
- ✓ Durante a mineração, mais de uma tarefa deve ser testada e combinada a fim de que comparações possam ser possíveis para a escolha da(s) tarefa(s) mais adequada para a solução final.
- ✓ Vejamos agora algumas das **tarefas** de mineração e exemplos.

Journal Pre-proof

Exemplo: quando os clientes compram um item específico, as empresas podem perceber que eles tendem a adquirir um segundo ou terceiro item junto.

O que fraldas e cervejas tem a ver?

Uma rede de varejos americana descobriu que muitos de seus clientes que compravam fraldas eram homens

Eles saíam à noite para comprar fraldas e aproveitavam para comprar cerveja também

A rede, então, colocou a fralda e a cerveja uma do lado da outra na loja. Resultado: as vendas de ambos dispararam!

Tarefa de Classificação

Tem como objetivo classificar as observações de forma automática em diferentes categorias.

Exemplo 1: NuBank quer classificar os clientes na categoria de risco de crédito "baixo", "médio" ou "alto", analisando o histórico de compras e o histórico financeiro.

Exemplo 2: classificar e-mails como spams e não spams.

Exemplo de algoritmo para esta tarefa: SVM

Tarefa de Agrupamento

Envolve a separação dos dados com base nas semelhanças entre os conjuntos de dados.

Exemplo: uma empresa pode dividir diferentes demografias de seu público-alvo em pacotes diferentes, dependendo de sua renda.

Exemplo de algoritmo para esta tarefa: K-médias, DBScan, dentre outros.

Tarefa de Predição

Permite projetar os dados que serão vistos no futuro por meio de uma modelagem preditiva.

Exemplo: prever a venda a um cliente com base em suas compras anteriores, históricos de crédito e status financeiro.

Exemplo de algoritmo: redes neurais.

1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação

✓ Visualização é a conversão dos dados para um formato visual ou tabular de forma que as características dos dados e da relação entre elementos dos dados possam ser percebidos visualmente.

✓ Visualização é uma das mais poderosas técnicas para a exploração dos dados:

- ✓ humanos têm muita habilidade de analisar grandes quantidades de informação apresentadas de forma **visual**
- ✓ podem detectar padrões gerais e tendências
- ✓ podem detectar exceções e padrões não usuais

- ✓ Exemplo: dezenas de milhares de pontos (dados) coletados via satélite são mostrados de forma visual em uma figura como um mapa de calor da temperatura dos oceanos (2009).

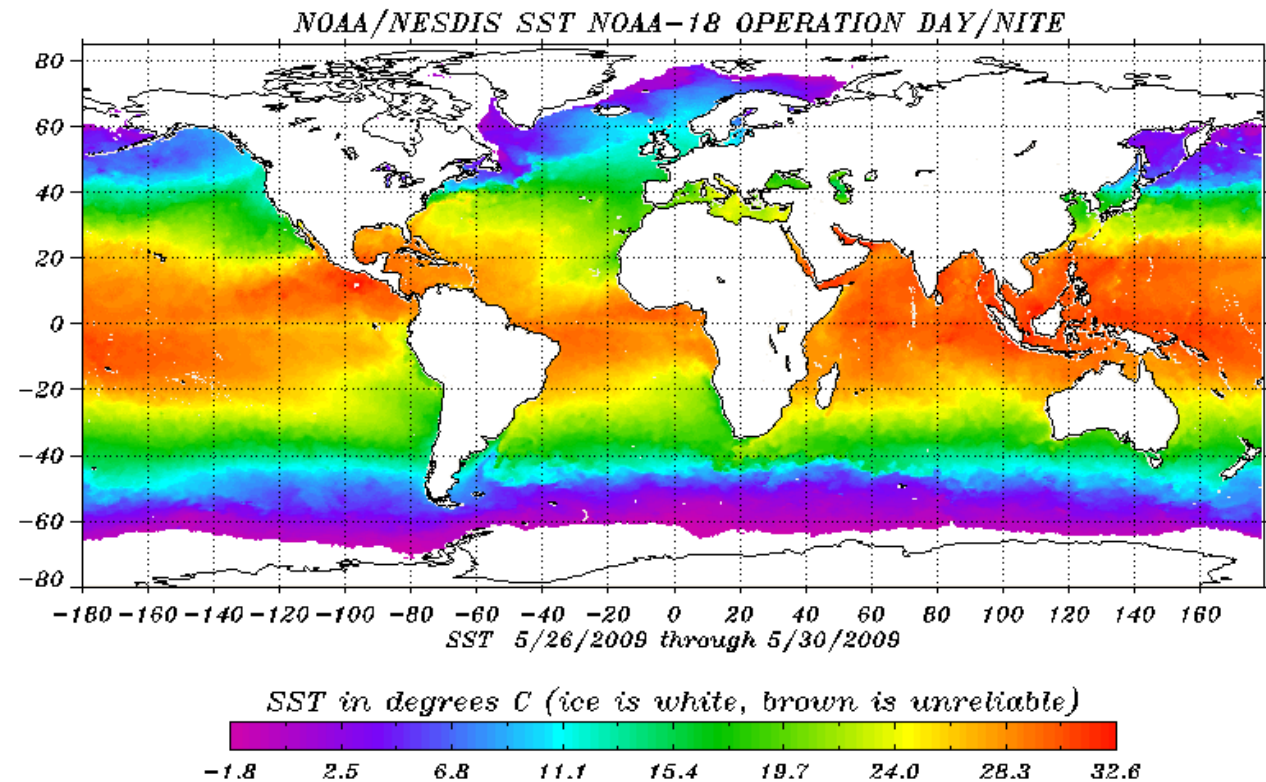
1 Seleção

2 Limpeza

3 Transformação

4 Mineração

5 Interpretação





**Antes de iniciar
qualquer
KDD...**

**É necessário
entender o
negócio**

- ✓ Produzir um plano do projeto de análise de dados.
- ✓ Objetivo
 - ✓ Exemplo: descobrir quais itens são adquiridos juntos em um supermercado.
- ✓ Tipo de problema (classificação, *agrupamento*,...)
 - ✓ Exemplo: tendo por base o objetivo acima, teríamos um problema de associação.
- ✓ Critérios para avaliação do modelo.
 - ✓ Exemplo: com um problema de associação são utilizadas as métricas como suporte e confiança para avaliação da qualidade das regras de associação geradas.

Roteiro da aula

1. Motivação para a análise de dados
2. Conceitos iniciais
3. O processo de KDD
4. Ferramentas
5. Introdução à linguagem Python
6. Ambiente Google Colaboratory

Ferramentas

- ✓ Apoiam o processo de KDD.
- ✓ Ferramentas visuais
 - ✓ Weka
 - ✓ Rapid Miner
 - ✓ Orange data mining (DM)
- ✓ Ferramentas não visuais
 - ✓ Linguagem R
 - ✓ Linguagem Python para análise de dados

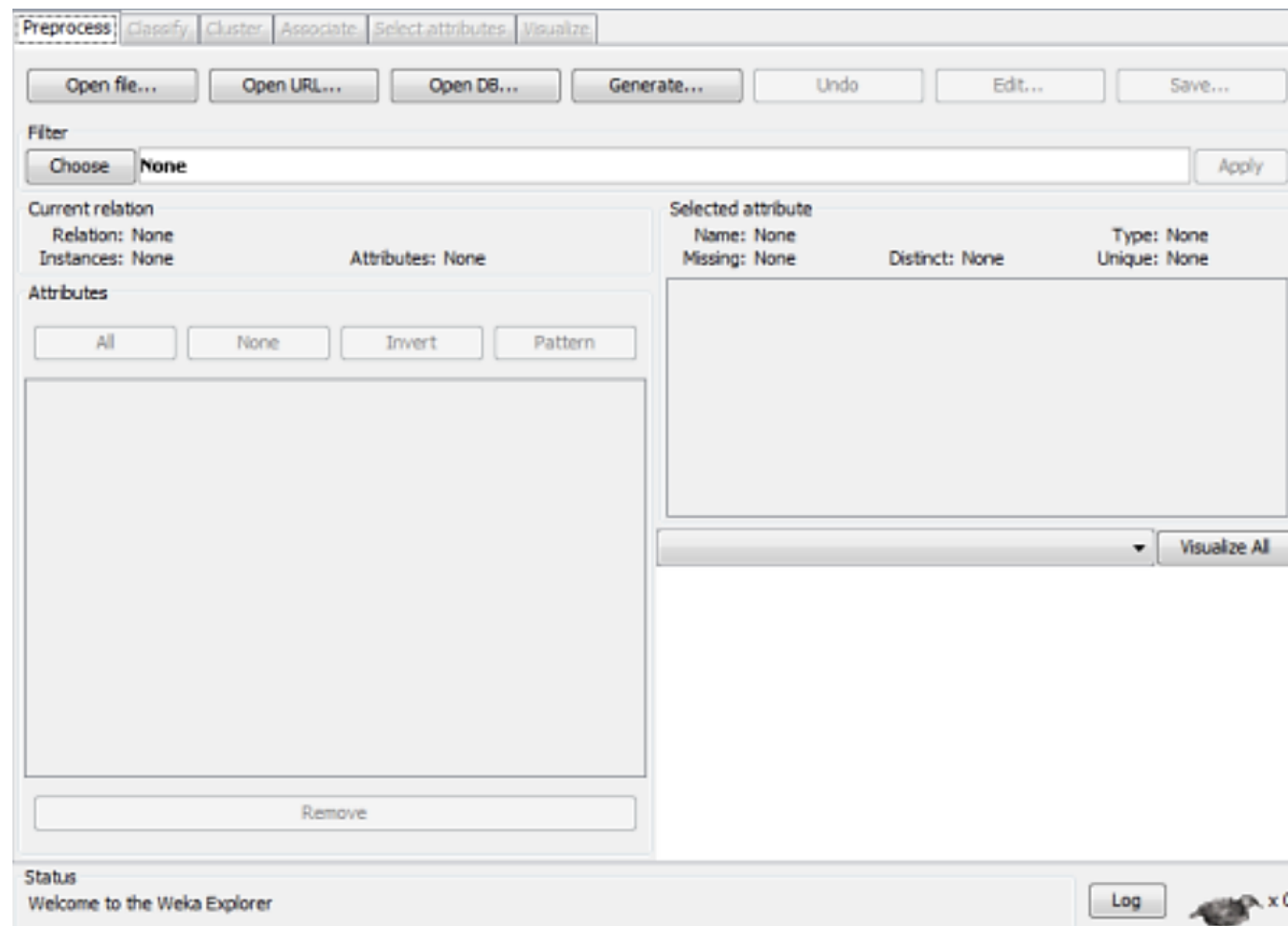
Ferramentas: Weka

- ✓ Desenvolvido em Java por pesquisadores da universidade da Nova Zelândia.
- ✓ Download: <https://www.cs.waikato.ac.nz/ml/weka/>
- ✓ Código aberto
- ✓ Weka é o nome de um pássaro.
- ✓ Pontos fortes
 - ✓ Classificação
 - ✓ Regras de associação
 - ✓ Agrupamento de dados



Ferramentas: Weka – interface gráfica

- ✓ É baseado em um interface gráfica a partir da qual é possível fazer pré-processamento, mineração e visualização.
 - ✓ **Preprocess**: escolhe e modifica os dados utilizados
 - ✓ **Classify**: treina e testa sistemas de aprendizagem que classificam ou realizar regressão
 - ✓ **Cluster**: análise de grupos
 - ✓ **Associate**: permite aprender regras de associação para os dados
 - ✓ **Select attributes**: seleciona os atributos mais relevantes nos dados
 - ✓ **Visualize**: gráfico 2D interativo dos dados



Ferramentas: Weka - entrada de dados

- ✓ O formato ARFF é utilizado como padrão para estruturar as bases de dados manipuladas pelo Weka.
- ✓ É um arquivo de texto ASCII que descreve uma lista de instâncias que compartilham um conjunto de atributos.

```
@relation weather
@attribute tempo {sol, nublado, chuva}
@attribute temperatura real
@attribute umidade real
@attribute vento {SIM, NAO}
@attribute jogo {sim, nao}

@data
sol,85,85,NAO,nao
sol,80,90,SIM,nao
nublado,83,86,NAO,sim
chuva,70,96,NAO,sim
chuva,68,80,NAO,sim
chuva,65,70,SIM,nao
nublado,64,65,SIM,sim
sol,72,95,NAO,nao
sol,69,70,NAO,sim
chuva,75,80,NAO,sim
sol,75,70,SIM,sim
nublado,72,90,SIM,sim
nublado,81,75,NAO,sim
chuva,71,91,SIM,nao

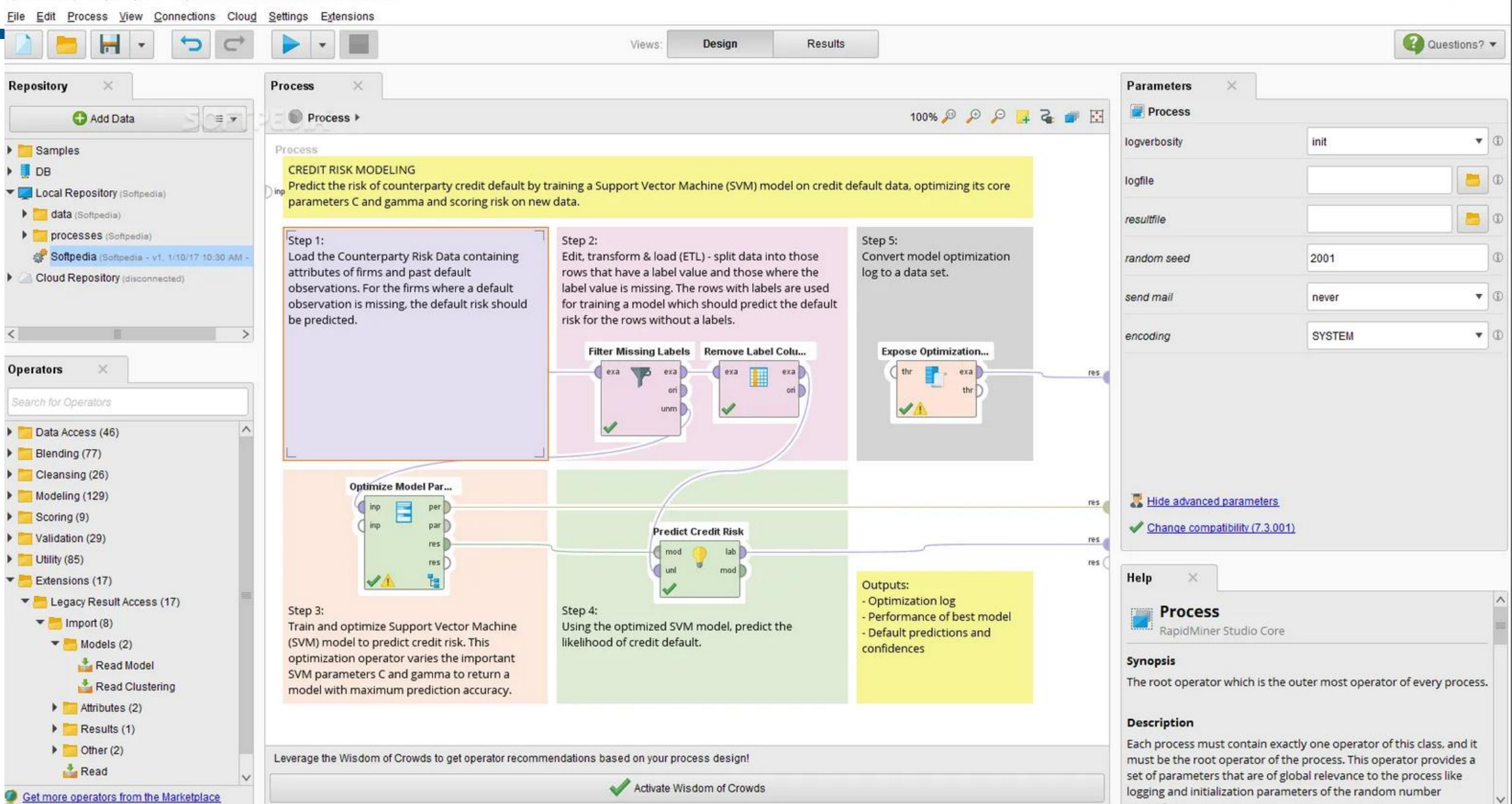
% linhas começando com % são comentários
```

cabeçalho

dados

- ✓ Desenvolvido em Java, inicialmente na Alemanha.
- ✓ Versão Studio paga, mas disponível para uso educacional.
- ✓ Mais de 500 operadores que implementam as mais diversas técnicas e algoritmos.
- ✓ Acesso às mais diversas fontes de dados: Excel, Access, Oracle, IBM DB2, Microsoft SQL Server, Sybase, Ingres, mySQL, Postgres, SPSS, etc.
- ✓ Download: <https://rapidminer.com/products/studio/>
- ✓ Interface gráfica amigável e flexível que permite desenhar o processo de KDD (**workflow**) desde a coleta até a interpretação.

Ferramentas: Rapid Miner - interface



The screenshot displays the Rapid Miner Studio interface with a process design canvas titled "CREDIT RISK MODELING". The process description states: "Predict the risk of counterparty credit default by training a Support Vector Machine (SVM) model on credit default data, optimizing its core parameters C and gamma and scoring risk on new data."

The process is divided into five steps:

- Step 1:** Load the Counterparty Risk Data containing attributes of firms and past default observations. For the firms where a default observation is missing, the default risk should be predicted.
- Step 2:** Edit, transform & load (ETL) - split data into those rows that have a label value and those where the label value is missing. The rows with labels are used for training a model which should predict the default risk for the rows without a labels.
- Step 3:** Train and optimize Support Vector Machine (SVM) model to predict credit risk. This optimization operator varies the important SVM parameters C and gamma to return a model with maximum prediction accuracy.
- Step 4:** Using the optimized SVM model, predict the likelihood of credit default.
- Step 5:** Convert model optimization log to a data set.

The process flow includes the following operators:

- Filter Missing Labels:** An example operator that filters out rows with missing labels.
- Remove Label Columns:** An example operator that removes the label column from the data.
- Expose Optimization...** An example operator that exposes the optimization log.
- Optimize Model Parameters:** An operator that optimizes the SVM model parameters.
- Predict Credit Risk:** An operator that uses the optimized SVM model to predict the likelihood of credit default.

The process outputs are:

- Optimization log
- Performance of best model
- Default predictions and confidences

The interface also shows the Repository, Operators, Parameters, and Help panels. The Parameters panel for the Process operator includes settings for log verbosity, logfile, resultfile, random seed, send mail, and encoding. The Help panel provides a synopsis and description of the Process operator.

- ✓ Desenvolvido em C++ e Python
- ✓ Código aberto.
- ✓ Download: <https://orangedatamining.com/>
- ✓ Trabalha com uma interface gráfica muito parecida com a do Rapid Miner que consiste na definição de um **workflow** para a mineração.

Ferramentas: Orange DM - interface



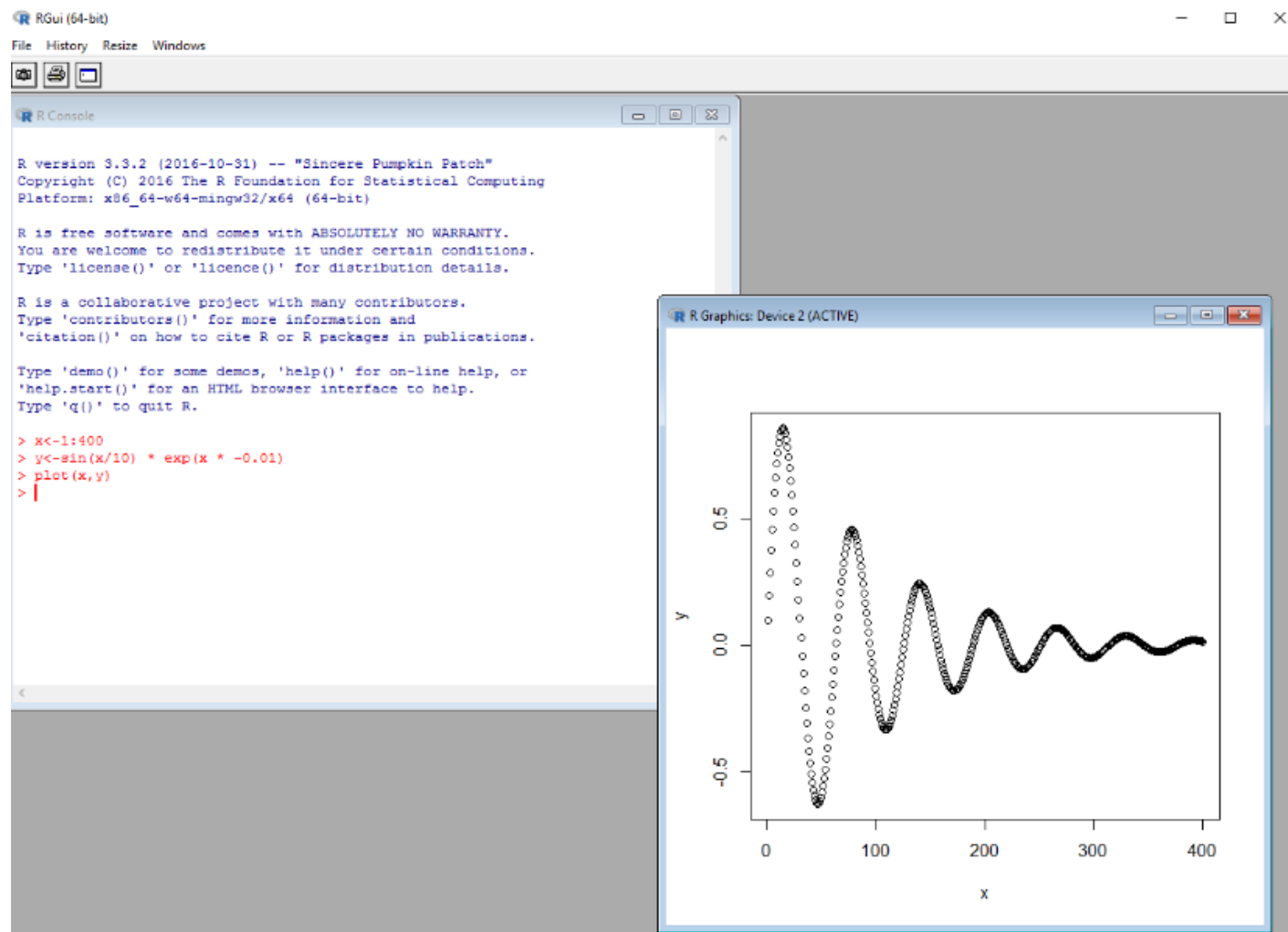
Ferramentas: Linguagem



- ✓ Linguagem de programação voltada para a análise de dados, interpretada, e orientada aos objetos.
- ✓ É software livre.

Download

<https://www.r-project.org/>



Ferramentas: Linguagem



The screenshot displays the RStudio environment with the following components:

- Editor:** Contains the R script `x<-runif(100, 1, 6)` and `hist(x)`.
- Environment/History:** Shows the variable `x` as a numeric vector of length 100, with a preview of values: 3.25, 1.07, 3.13, 1.89, 2.9.
- Console:** Shows the execution of the commands `x<-runif(100, 1, 6)` and `hist(x)`.
- Plots:** A histogram titled "Histogram of x" is displayed, showing the frequency distribution of the generated random values.

| Bin Range | Frequency |
|-----------|-----------|
| 1.0 - 1.5 | 9 |
| 1.5 - 2.0 | 10 |
| 2.0 - 2.5 | 5 |
| 2.5 - 3.0 | 10 |
| 3.0 - 3.5 | 10 |
| 3.5 - 4.0 | 15 |
| 4.0 - 4.5 | 11 |
| 4.5 - 5.0 | 7 |
| 5.0 - 5.5 | 13 |
| 5.5 - 6.0 | 10 |

R-Stdudio é um ambiente que facilita a análise de dados.

Arquivos, Gráficos, pacotes, ajuda.

Ferramentas mais usadas segundo o KDNuggets

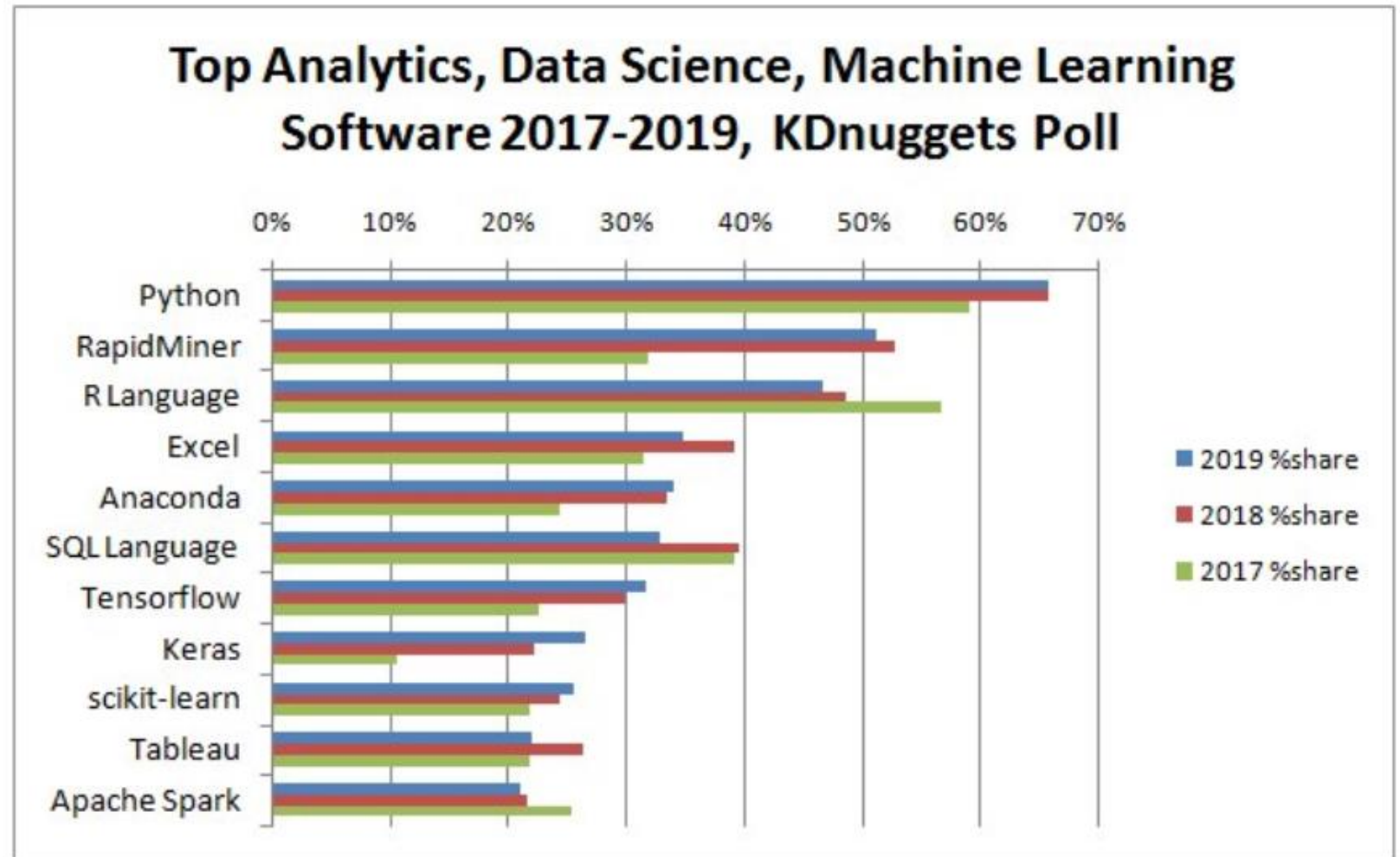


Fig 1: KDNuggets Analytics/Data Science 2019 Software Poll: top tools in 2019, and their share in the 2017, 2018 polls

Roteiro da aula

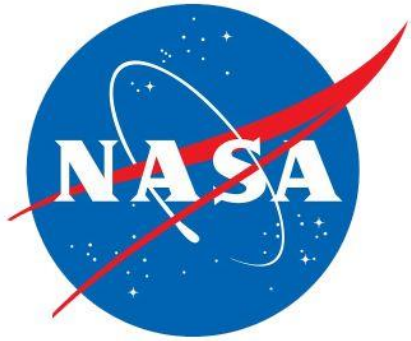
1. Motivação para a análise de dados
2. Conceitos iniciais
3. O processo de KDD
4. Ferramentas
5. Introdução à linguagem Python
6. Ambiente Google Colaboratory

- ✓ Linguagem de programação interpretada
 - ✓ Interpretador disponível para vários sistemas operacionais
 - ✓ Multi-plataforma
- ✓ Código aberto.
- ✓ É multi-paradigma: procedural, funcional, orientada à objetos.
- ✓ Criada por Guido van Rossum em 1991.

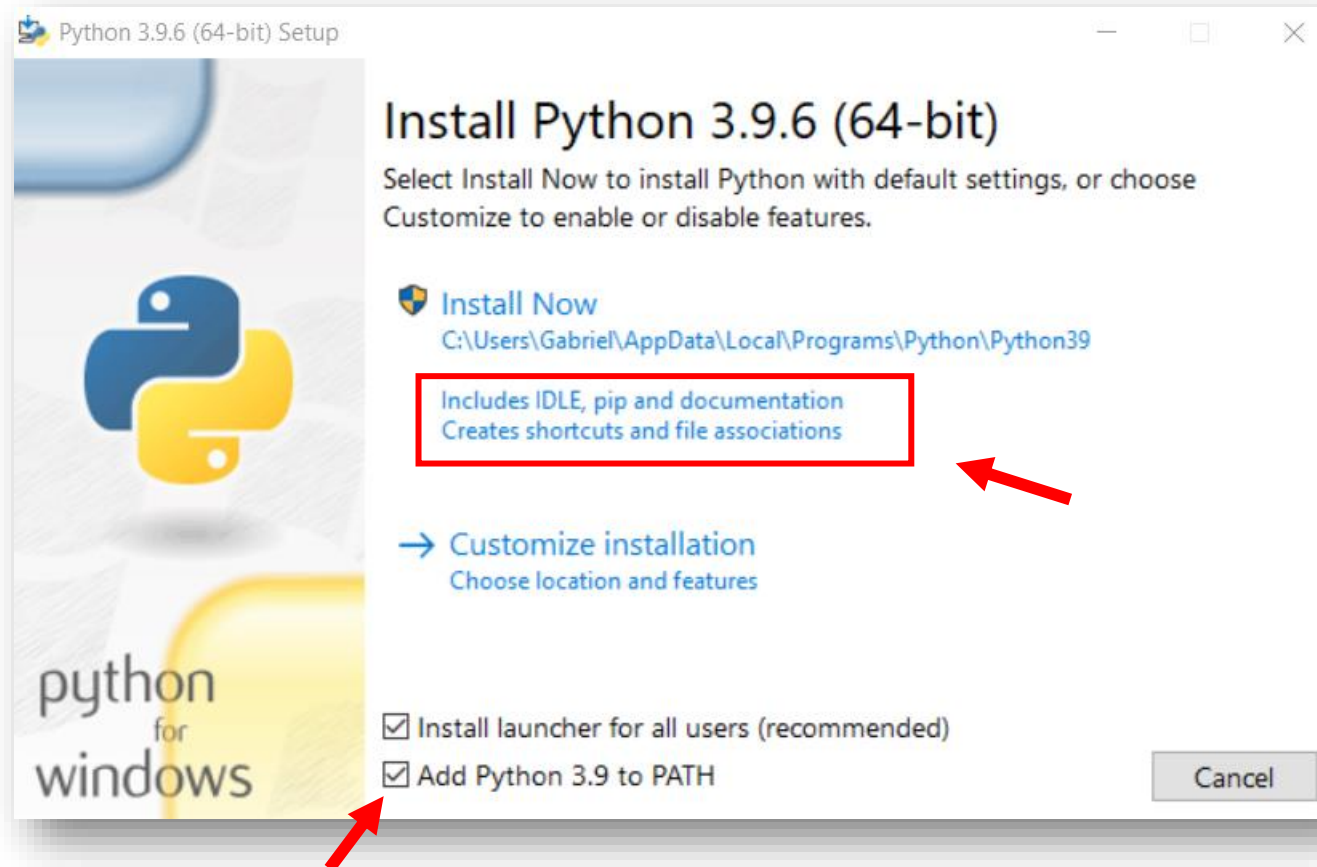


- ✓ Diversas estruturas de dados nativas
 - ✓ lista, tupla, dicionário
- ✓ Indentação para estrutura de bloco, sintaxe limpa, fácil e simples (muito próxima da nossa linguagem escrita).
- ✓ É organizada em bibliotecas.
 - ✓ Pandas
 - ✓ Numpy
 - ✓ Scikit-learn
 - ✓ Seaborn
 - ✓ Jupyter Notebook
 - ✓ Spyder IDE
 - ✓ E tantas outras mais...

✓ Quem está usando Python?



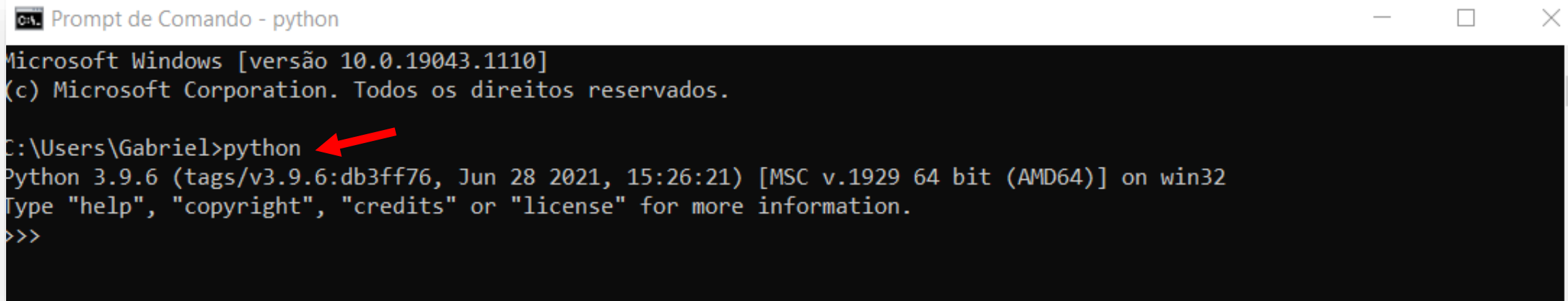
- ✓ Instalação manual (minha preferida) - Windows
 - ✓ Download: <https://www.python.org/downloads/>
 - ✓ Versão atual: 3.9.6



- ✓ Uma maneira alternativa para instalar e gerenciar os pacotes Python é utilizar o Anaconda.
- ✓ Anaconda.
 - ✓ É uma iniciativa (distribuição) que tem como objetivo agregar vários pacotes e ferramentas em uma única instalação.
 - ✓ Download: <https://docs.anaconda.com/anaconda/install/windows/>

✓ Interpretador Python

- ✓ Não há compilação do código.
- ✓ Basta digitar os comandos da linguagem que eles são executados.

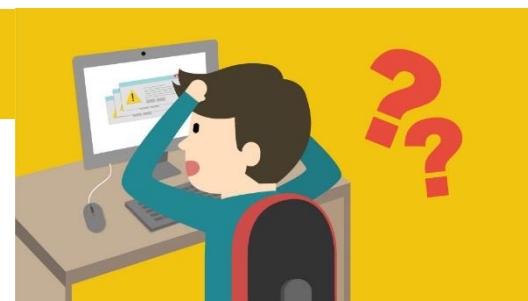


```
Prompt de Comando - python
Microsoft Windows [versão 10.0.19043.1110]
(c) Microsoft Corporation. Todos os direitos reservados.

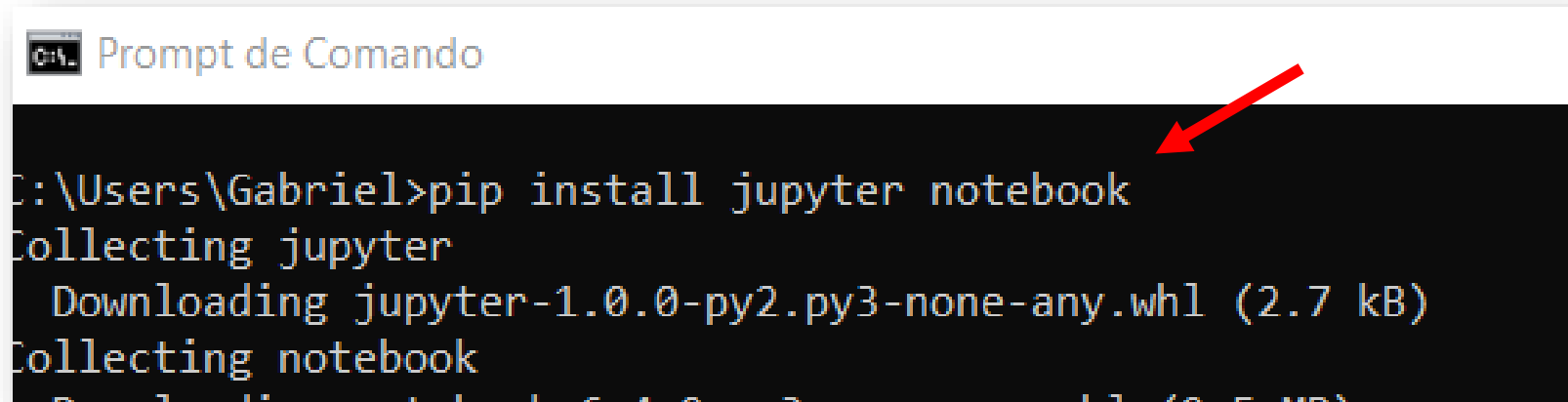
C:\Users\Gabriel>python
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Mas programar aqui é complicado né?!

Vamos instalar um ambiente melhor...



- ✓ PyPI (*package installer for Python*) – comando PIP
 - ✓ Sistema de gerenciamento de pacotes do Python.
 - ✓ Permite instalar as **bibliotecas** que não vem por padrão na instalação.



```
Prompt de Comando

C:\Users\Gabriel>pip install jupyter notebook
Collecting jupyter
  Downloading jupyter-1.0.0-py2.py3-none-any.whl (2.7 kB)
Collecting notebook
```

✓ Jupyter notebook

- ✓ É um ambiente de desenvolvimento interativo.
- ✓ Permite dividir o código em blocos independentes chamados **células**. Assim você pode executar cada bloco de código separadamente.
- ✓ Facilita a leitura e divulgação do código, além de exibir os resultados e o código de forma bonita.



✓ Jupyter notebook – como rodar?



- ✓ Digite **jupyter notebook** no seu terminal de comandos.
- ✓ Não feche o terminal enquanto o notebook estiver rodando.
- ✓ Seu navegador será aberto automaticamente, pois o jupyter é um ambiente que roda em interface web.
- ✓ Caso o seu navegador não abra automaticamente, digite na barra de endereços:
 - ✓ <http://localhost:8888>
- ✓ Seus notebooks jupyter serão salvos no diretório onde você digitou o comando **jupyter notebook** no terminal.

Bem melhor que aquela tela preta!



Roteiro da aula

1. Motivação para a análise de dados
2. Conceitos iniciais
3. O processo de KDD
4. Ferramentas
5. Introdução à linguagem Python
6. Ambiente Google Colaboratory

- ✓ É um ambiente baseado no Jupyter notebook para análise de dados.
- ✓ Permite executar códigos Python sem instalar nada localmente.
- ✓ Conta com a versão 3 do Python.

- ✓ E tudo isso diretamente na sua conta institucional da UFSM!
 - ✓ Vá até qualquer pasta do seu Google Drive e clique com o botão direito numa área em branco.
 - ✓ Clique em “mais” e, em seguida, “conectar mais apps”.
 - ✓ Procure na lista usando a palavra “colaboratory”.
 - ✓ Selecione o Colab e clique em instalar.
 - ✓ Pronto! Você já pode começar a se sentir um cientista de dados começando a praticar Python!

Portais sobre ML e AI

✓ Kaggle

✓ <https://www.kaggle.com/>

✓ UCI

✓ <https://archive.ics.uci.edu/>

✓ GitHub

✓ Data.gov

✓ Google Dataset Search

✓ AWS Public Datasets

✓ Reddit Datasets



Dúvidas?

Ciência de Dados II

Professor: Gabriel Machado Lunardi
gabriel.lunardi@ufsm.br