



Machine Learning

4º Bacharelado em Ciência da Computação

Aprendizagem de Máquina

- **Métodos Preditivos**
- Métodos Descritivos

- Métodos Preditivos

- A modelagem preditiva é realizada através de uma série de técnicas analíticas e estatísticas, utilizadas para o desenvolvimento de modelos que podem prever eventos futuros a partir de comportamentos diários, incluindo análise de séries temporais ou modelos de regressão.
 - Classificação
 - Regressão

Aprendizagem de Máquina

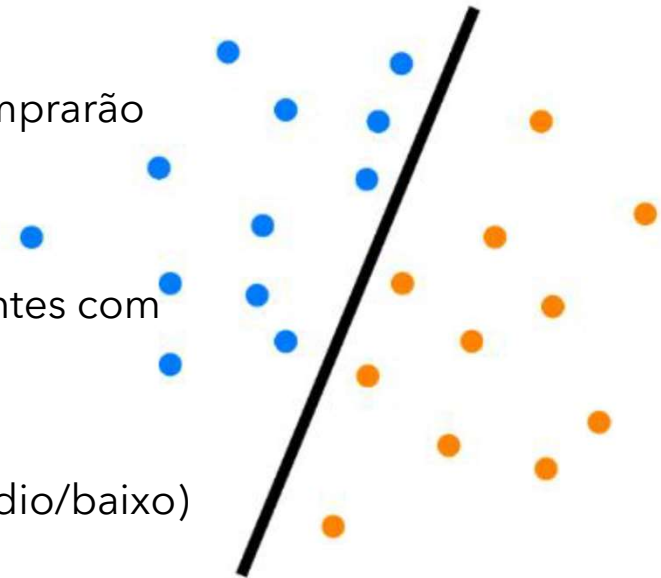
- Métodos Preditivos
- **Métodos Descritivos**

- Métodos Descritivos

- Trabalha com análise de dados históricos e cruzamento de informações para gerar um panorama claro e preciso para o momento.
- É uma boa maneira de visualizar os dados e entender o presente.
- Por exemplo, numa análise de crédito, verificam-se dados de pessoas e grupos sociais para definir o risco envolvido na concessão de um determinado crédito.
 - Associação
 - Agrupamento
 - Detecção de Desvios
 - Padrões Sequenciais
 - Sumarização.

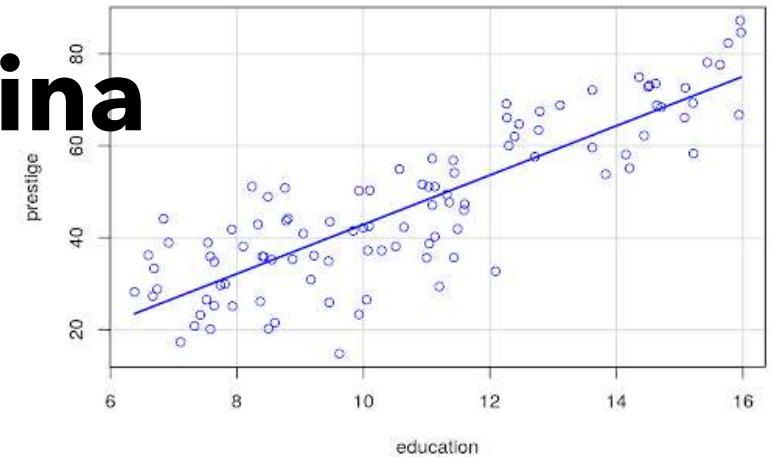
Aprendizagem de Máquina

- Métodos Preditivos
 - **Classificação**
 - Regressão
- Classificação
 - Marketing direto
 - Classifica grupos de clientes para saber se eles comprarão ou não um certo produto.
 - Satisfação de clientes
 - Classifica o estado de satisfação de grupos de clientes com certo produto/serviço.
 - Risco de crédito
 - O banco precisa avaliar o risco de crédito (alto/médio/baixo) se ele emprestar dinheiro para um cliente.



Aprendizagem de Máquina

- Métodos Preditivos
 - Classificação
 - **Regressão**
- Regressão
 - Gastos de propaganda → valor de venda
 - Baseado no gasto com propaganda, o sistema pode prever o retorno das vendas.
 - Fatores externos → valor do dólar
 - Baseado em fatores externos globais, o sistema pode prever o valor de uma moeda.
 - Gastos no cartão → limite
 - Baseado nos gastos do cliente, o sistema pode prever um novo limite para o cartão do cliente.
 - Resultados de exame → probabilidade de doença



Aprendizagem de Máquina

- Métodos Descritivos
 - Associação
 - Agrupamento
 - Detecção de Desvios
 - Padrões Sequenciais
 - Sumarização
 - Encontrar elementos que implicam na presença de outros elementos em uma mesma transação, ou seja, encontrar relacionamentos ou padrões frequentes entre conjuntos de dados.
 - Exemplos:
 - Organização de prateleiras de supermercado
 - Promoções de itens de que são vendidos em conjunto
 - Planejar catálogos de produtos e promoções
 - Controle de evasão de alunos de um certo curso.

Aprendizagem de Máquina

- Métodos Descritivos

- Agrupamento

- Associação
- **Agrupamento**
- Detecção de Desvios
- Padrões Sequenciais
- Sumarização

- A análise de agrupamento ou *clustering* (categorização) é uma das principais técnicas de aprendizado não supervisionado. Seu objetivo principal é agrupar (ou segmentar) indivíduos em *clusters*, de maneira que:
 - Indivíduos de um mesmo *cluster* sejam semelhantes em relação aos valores das variáveis em análise;
 - Por outro lado, indivíduos de *clusters* distintos sejam diferentes (dissimilares).
 - Exemplos:
 - Segmentação de mercado
 - Encontrar grupos de clientes que podem comprar um certo produto (mala direta)
 - Agrupar notícias e enviar para certos clientes
 - Agrupamento de produtos similares
 - Perfis de clientes Netflix.

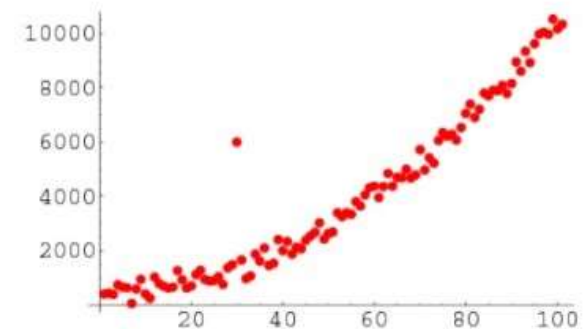
Aprendizagem de Máquina

- Métodos Descritivos

- Associação
- Agrupamento
- **Detecção de Desvios**
- Padrões Sequenciais
- Sumarização

- Detecção de Desvios (*outliers*)

- Os *outliers* são dados que se diferenciam de todos os outros, são pontos fora da curva normal. Em outras palavras, um *outlier* é um valor que foge da normalidade e que pode causar anomalias nos resultados obtidos por meio de algoritmos e sistemas de análise.
- Exemplos:
 - Fraude em cartão de crédito
 - Intrusão em redes
 - Consumo de energia elétrica, água e telefone
 - Desempenho de atletas (*dopping*).



Aprendizagem de Máquina

- Métodos Descritivos

- Associação
- Agrupamento
- Detecção de Desvios
- **Padrões Sequenciais**
- Sumarização

- Descoberta de Padrões Sequenciais

- Objetiva encontrar padrões de dados numa sequência temporal.
- Exemplos:
 - Marketing direcionado para pessoas que têm maiores chances de adquirir um novo produto
 - Prevenção de doenças
 - Navegação em sites.

Aprendizagem de Máquina

- Métodos Descritivos

- Sumarização

- Técnica para a identificação de perfis.

- Exemplos:

- São ouvintes do programa, homens com idade entre 25 e 30 anos, com nível superior e que atuam na área de administração de empresas.
 - Segmentação de mercado.

- Associação
- Agrupamento
- Detecção de Desvios
- Padrões Sequenciais
- **Sumarização**

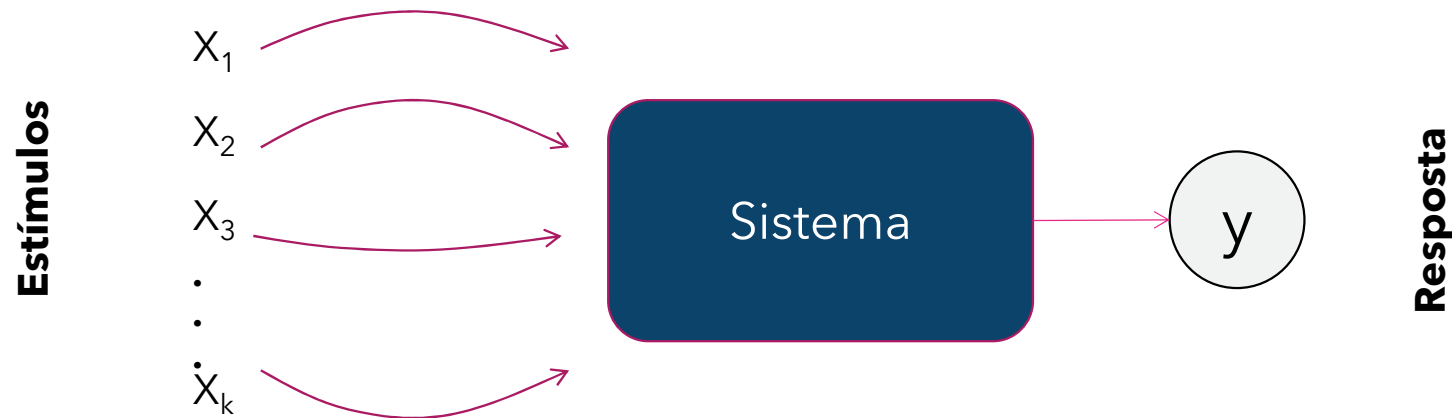
Tipos de aprendizagem de máquina

Tipos de Aprendizado de Máquina

Supervisionada	Não Supervisionada	Reforço
Classificação	Associação	
Regressão	Agrupamento	
	Detecção de desvios	
	Padrões sequenciais	
	Sumarização	

Tipos de Aprendizado de Máquina

- Aprendizagem Supervisionada
 - Refere-se ao caso em que um conjunto de variáveis X_1, X_2, \dots, X_p , medidas em n indivíduos, são usadas para explicar (predizer) uma variável resposta (Y).



Modelo mental de um algoritmo de Aprendizado Supervisionado

Tipos de Aprendizado de Máquina

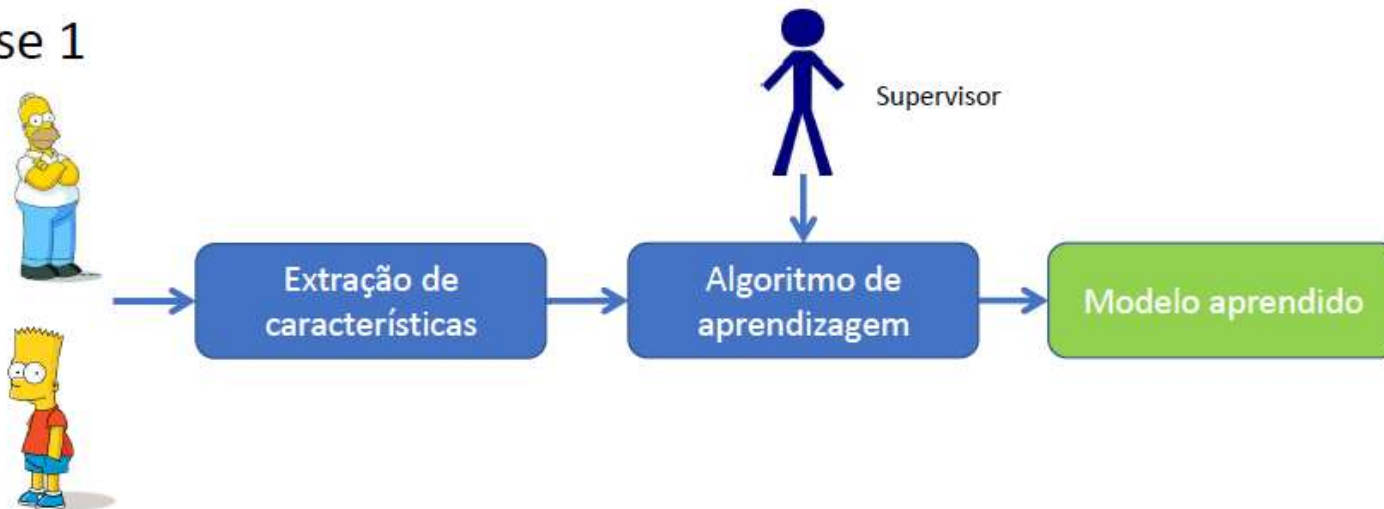


Variáveis que influenciam o preço de um imóvel

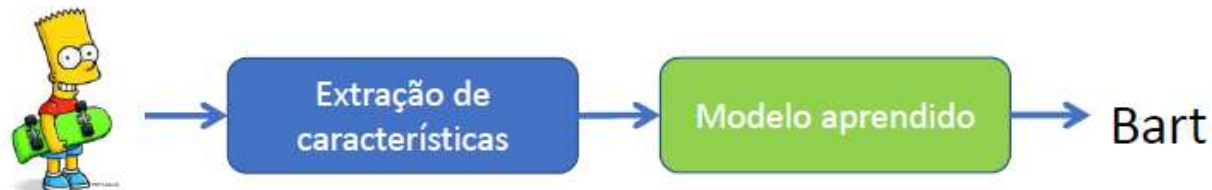
Tipos de Aprendizado de Máquina

Aprendizagem supervisionada

Fase 1



Fase 2



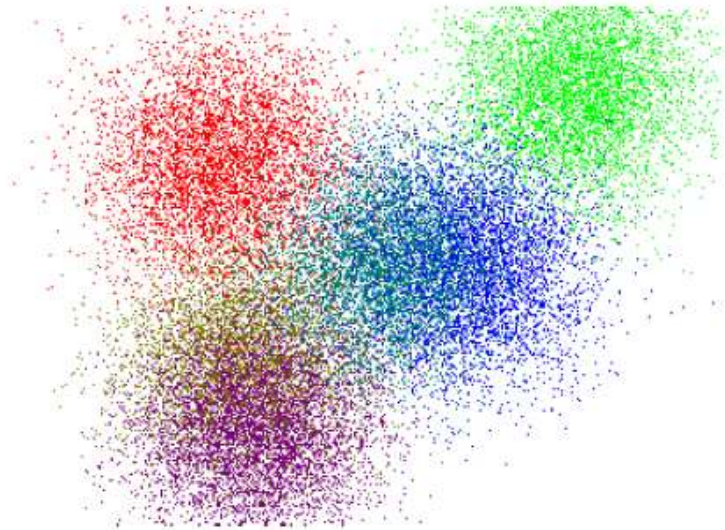
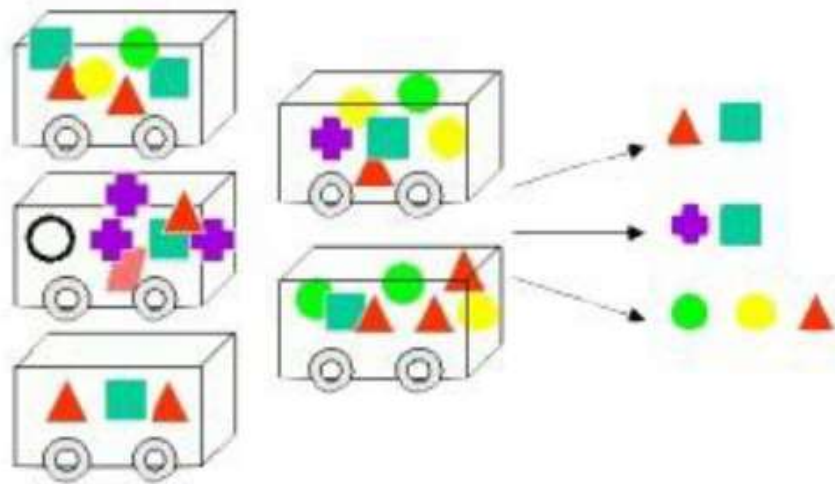
Tipos de Aprendizado de Máquina

- Aprendizagem não-supervisionada
 - No *aprendizado não-supervisionado*, não temos uma variável resposta, sendo que o interesse é explorar informações do conjunto de variáveis em análise;
 - Em vez de responder à programação de um programador, o *aprendizado não-supervisionado* identifica semelhanças nos dados e reage com base na presença ou ausência de tais semelhanças em cada novo dado.

Tipos de Aprendizado de Máquina

Aprendizagem não-supervisionada

- Analisar automaticamente os dados (associação, agrupamento)
- Necessita análise para determinar o significado dos padrões encontrados



Tipos de Aprendizado de Máquina

- Aprendizagem por reforço:
 - Aprender a partir das interações com o ambiente (causa e efeito)
 - Aprender com sua própria experiência
 - Robô coletando lixo aprendendo a andar em um ambiente
 - Controle automatizado de elevadores.

Classificação



Aprendizagem de máquina	
Métodos preditivos	Métodos descritivos
Classificação	Associação
Regressão	Agrupamento
	Detecção de desvios
	Padrões sequenciais
	Sumarização

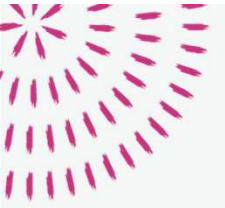
Exemplo

- Banco.
- Base de dados histórica dos clientes que já solicitaram empréstimo ao Banco.
- Objetivo:
 - Com base nos atributos previsores, desejamos saber qual é a classe que corresponde ao perfil do cliente que solicita empréstimo.


Nome	Hist de Crédito	Dívida	Garantias	Renda anual	Risco
Ana Paula	Ruim	Alta	Nenhuma	< 15.000	Alto
Carlos	Desconhecida	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto
Eduarda	Desconhecida	Baixa	Nenhuma	>= 15.000 a <= 35.000	Moderado
Fabiano	Desconhecida	Baixa	Nenhuma	< 15.000	Alto
Gustavo	Desconhecida	Baixa	Nenhuma	> 35.000	Baixo
Hélia	Desconhecida	Baixa	Adequada	> 35.000	Baixo
Ivanete	Ruim	Baixa	Nenhuma	< 15.000	Alto
Maria	Ruim	Baixa	Adequada	> 35.000	Moderado
Paulo	Boa	Baixa	Nenhuma	> 35.000	Baixo
Pedro	Boa	Alta	Adequada	> 35.000	Baixo
Renata	Boa	Alta	Nenhuma	< 15.000	Alto
Renato	Boa	Alta	Nenhuma	>= 15.000 a <= 35.000	Moderado
Sueli	Boa	Alta	Nenhuma	> 35.000	Baixo
Vânia	Ruim	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto

Atributos PREVISORES

Atributo META ou CLASSE

- 
- Após o treinamento do algoritmo, espera-se que ele tenha aprendido a encontrar a CLASSE correta, a partir da base de dados histórica.
 - O objetivo é que, a partir de novos registros e sem a informação da CLASSE, o algoritmo tenha aprendido a identificar o cliente dentro de sua respectiva CLASSE.

Nome	Hist de Crédito	Dívida	Garantias	Renda anual
Ana Paula	Ruim	Alta	Nenhuma	< 15.000
Carlos	Desconhecida	Alta	Nenhuma	>= 15.000 a <= 35.000
Eduarda	Desconhecida	Baixa	Nenhuma	>= 15.000 a <= 35.000
Paulo	Boa	Baixa	Nenhuma	> 35.000



Outro exemplo

- Venda de livros.
- Base de dados histórica dos clientes que já compraram livros na loja.
- Objetivo:
 - Com base nos atributos previsores, desejamos saber qual é a classe que corresponde ao cliente que possui mais chance de comprar livros.

Sexo	País	Idade	Comprar
M	França	25	Sim
M	Inglaterra	21	Sim
F	França	23	Sim
F	Inglaterra	34	Sim
F	França	30	Não
M	Alemanha	21	Não
M	Alemanha	20	Não
F	Alemanha	18	Não
F	França	34	Não
F	França	34	Não
M	França	55	Não
M	Inglaterra	25	Sim
M	Alemanha	48	Sim
F	Inglaterra	23	Não

Atributos PREVISORES

CLASSE

Treinamento

Classificação (venda de livros)

Sexo	País	Idade
M	França	38
F	Inglaterra	25
M	Alemanha	55
F	França	20

?

Mais um exemplo

- Previsão de prática de esporte (futebol ou aeróbica).
- Base de dados histórica das pessoas que já praticaram esporte.
- Objetivo:
 - Com base nos atributos previsores, desejamos saber qual é a classe que estará associada à pessoa para a realização de determinado esporte.

Classificação (prever o esporte)

Cor dos olhos	Casado	Sexo	Cabelo	Esporte
Castanho	Sim	M	Longo	Futebol
Azul	Sim	M	Curto	Futebol
Castanho	Sim	M	Longo	Futebol
Castanho	Não	F	Longo	Aeróbica
Castanho	Não	F	Longo	Aeróbica
Azul	Não	M	Longo	Futebol
Castanho	Não	F	Longo	Aeróbica
Castanho	Não	M	Curto	Futebol
Castanho	Sim	F	Curto	Aeróbica
Castanho	Não	F	Longo	Aeróbica
Azul	Não	M	Longo	Futebol
Azul	Não	M	Curto	Futebol

Treinamento

Cor dos olhos	Casado	Sexo	Cabelo
Castanho	Sim	M	Curto
Castanho	Não	M	Longo
Azul	Não	F	Longo
Azul	Sim	M	Longo

Último exemplo

- Previsão se uma pessoa vai jogar tênis (baseado nas condições climáticas).
- Base de dados histórica das condições do tempo.
- Objetivo:
 - Com base nos atributos previsores, desejamos saber se, a partir das condições do tempo, uma pessoa poderá ou não poderá jogar tênis.

Classificação (jogar tênis)

Tempo	Temperatura	Humidade	Vento	Jogar tênis
Ensolarado	Quente	Alta	Fraco	Não
Ensolarado	Quente	Alta	Forte	Não
Nublado	Quente	Alta	Fraco	Sim
Chuvoso	Moderada	Alta	Fraco	Sim
Chuvoso	Agradável	Normal	Fraco	Sim
Chuvoso	Agradável	Normal	Forte	Não
Nublado	Agradável	Normal	Forte	Sim
Ensolarado	Moderada	Alta	Fraco	Não
Ensolarado	Agradável	Normal	Fraco	Sim
Chuvoso	Moderada	Normal	Fraco	Sim
Ensolarado	Moderada	Normal	Forte	Sim
Nublado	Moderado	Alta	Fraco	Sim
Nublado	Quente	Normal	Fraco	Sim
Chuvoso	Moderado	Alta	Forte	Não

Treinamento

Tempo	Temperatura	Humidade	Vento
Ensolarado	Moderada	Normal	Forte
Chuvoso	Agradável	Normal	Fraco
Nublado	Quente	Normal	Forte
Nublado	Agradável	Alta	Forte

Classificação

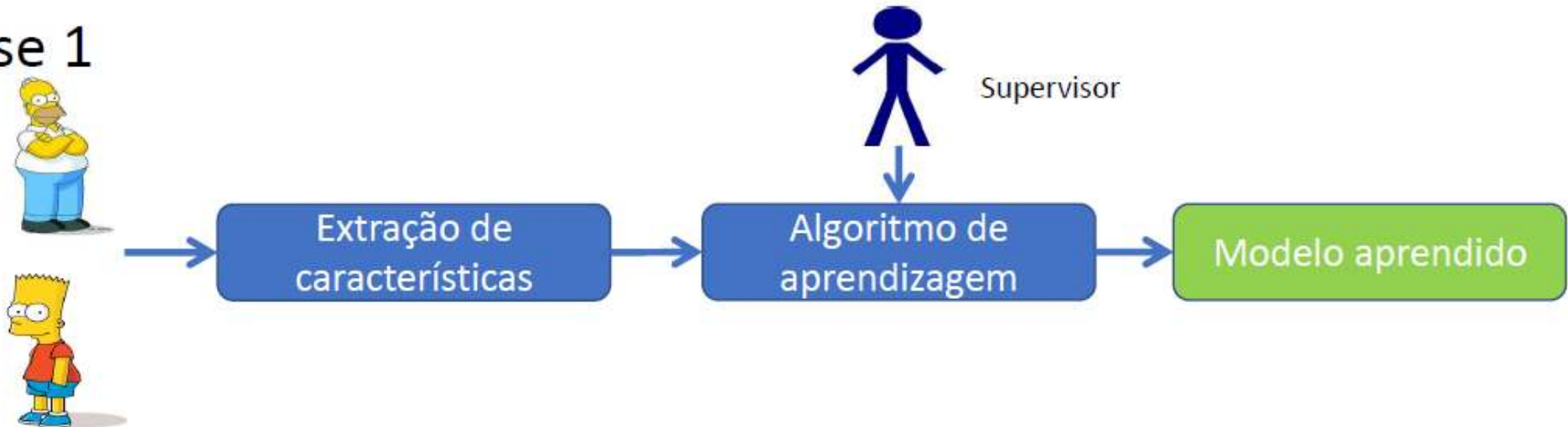
- Cada registro pertence a uma classe e possui um conjunto de atributos previsores;
- O objetivo é descobrir um relacionamento entre os atributos previsores e o atributo meta - utilizamos um algoritmo de *machine learning*;
- O valor do atributo meta é conhecido (aprendizagem supervisionada).

Representação da Classificação (método indutivo)

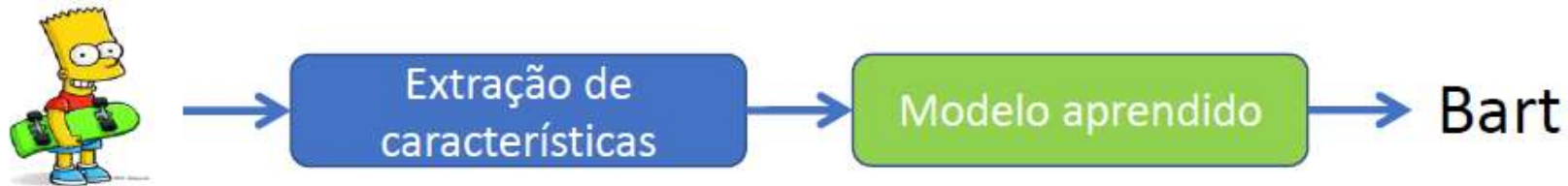


Aprendizagem Supervisionada (indução)

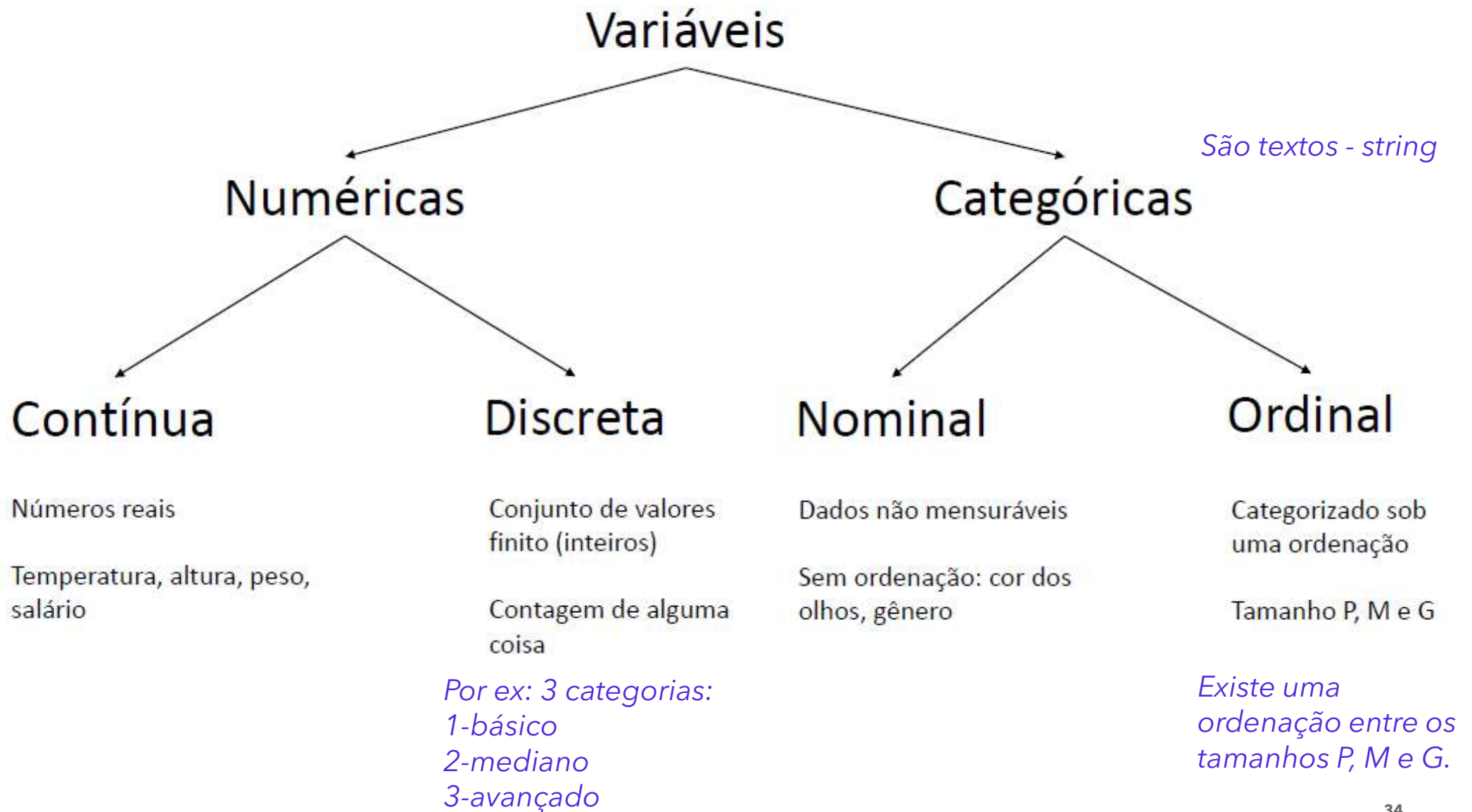
Fase 1



Fase 2

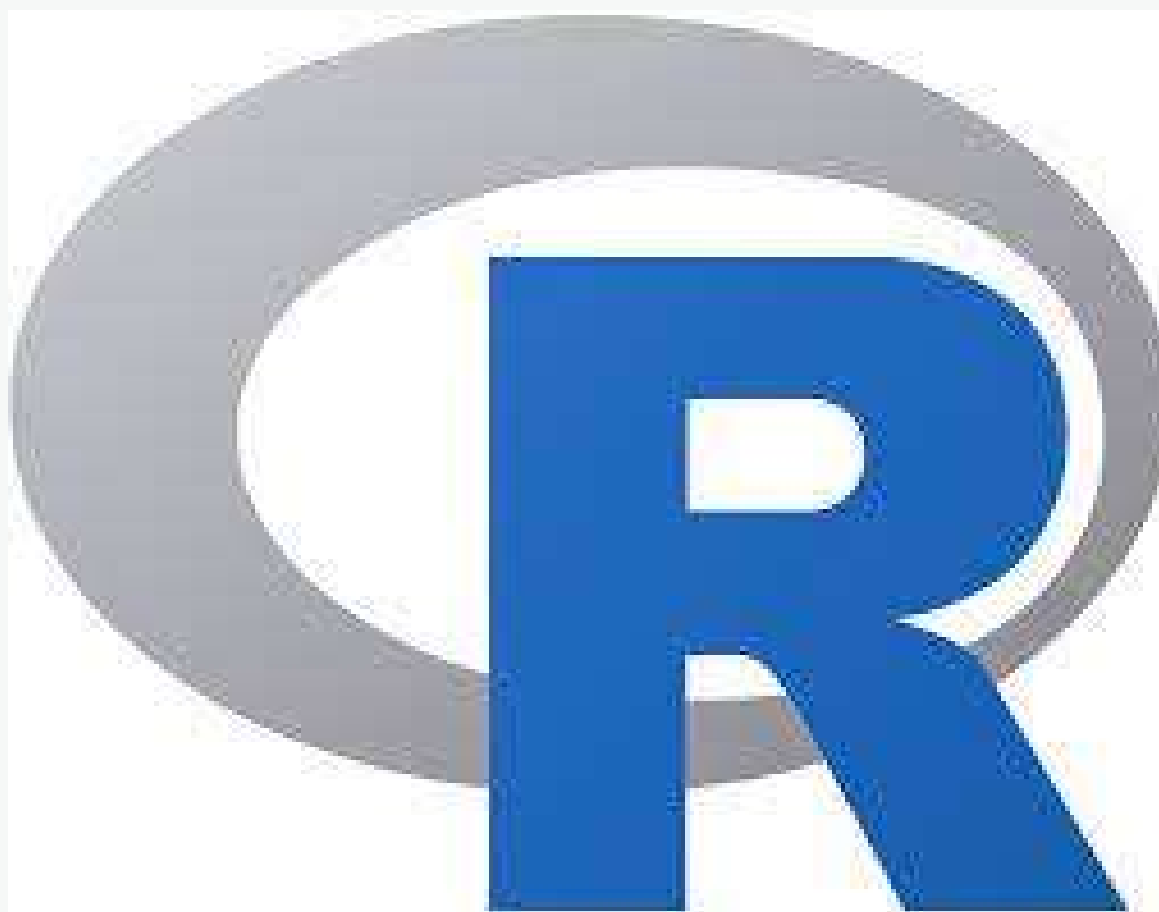


Tipos de variáveis



Variáveis

- É importante conhecer os tipos de variáveis, pois quando trabalharmos com funções precisaremos entender dos diferentes tipos que são trabalhados pelas funções.



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



[\[Home\]](#)

Download

[CRAN](#)

R Project

[About R](#)

[Logo](#)

[Contributors](#)

[What's New?](#)

[Reporting Bugs](#)

[Conferences](#)

[Search](#)

[Get Involved: Mailing Lists](#)

[Developer Pages](#)

[R Blog](#)



The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To [download R](#), please choose your preferred [CRAN mirror](#).

If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

News

- [R version 4.1.1 \(Kick Things\)](#) has been released on 2021-08-10.
- [R version 4.0.5 \(Shake and Throw\)](#) was released on 2021-03-31.
- Thanks to the organisers of useR! 2020 for a successful online conference. Recorded tutorials and talks from the conference are available on the [R Consortium YouTube channel](#).
- You can support the R Foundation with a renewable subscription as a [supporting member](#)

News via Twitter

CRAN - Mirrors


cran.r-project.org/mirrors.html

CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: [main page](#), [windows release](#), [windows old release](#).

If you want to host a new mirror at your institution, please have a look at the [CRAN Mirror HOWTO](#).

0-Cloud	https://cloud.r-project.org/	Automatic redirection to servers worldwide, currently sponsored by Rstudio
Algeria	https://cran.usthb.dz/	University of Science and Technology Houari Boumediene
Argentina	http://mirror.fcaglp.unlp.edu.ar/CRAN/	Universidad Nacional de La Plata
Australia	https://cran.csiro.au/ https://mirror.aarnet.edu.au/pub/CRAN/ https://cran.ms.unimelb.edu.au/ https://cran.curtin.edu.au/	CSIRO AARNET School of Mathematics and Statistics, University of Melbourne Curtin University
Austria	https://cran.wu.ac.at/	Wirtschaftsuniversität Wien
Belgium	https://www.freeststatistics.org/cran/ https://ftp.belnet.be/mirror/CRAN/	Patrick Wessa Belnet, the Belgian research and education network
Brazil	https://nbcgib.uesc.br/mirrors/cran/ https://cran-r.c3sl.ufpr.br/ https://cran.fiocruz.br/ https://vps.fmvz.usp.br/CRAN/ https://brieger.esalq.usp.br/CRAN/	Computational Biology Center at Universidade Estadual de Santa Cruz Universidade Federal do Parana Oswaldo Cruz Foundation, Rio de Janeiro University of Sao Paulo, Sao Paulo University of Sao Paulo, Piracicaba
Bulgaria		



[CRAN](#)
[Mirrors](#)
[What's new?](#)
[Task Views](#)
[Search](#)

[About R](#)
[R Homepage](#)
[The R Journal](#)

[Software](#)
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Other](#)

[Documentation](#)
[Manuals](#)
[FAQs](#)
[Contributed](#)

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#) ([Debian](#), [Fedora/Redhat](#), [Ubuntu](#))
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.


Source Code for all Platforms


Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2021-08-10, Kick Things) [R-4.1.1.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.



 Digite aqui para pesquisar

26°C Ensolarado

11:11
26/09/2021



CRAN
[Mirrors](#)
[What's new?](#)
[Task Views](#)
[Search](#)

About R
[R Homepage](#)
[The R Journal](#)

Software
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Other](#)

Documentation
[Manuals](#)
[FAQs](#)
[Contributed](#)

R for Windows

Subdirectories:

[base](#)

[contrib](#)

[old contrib](#)

[Rtools](#)

Binaries for base distribution. This is what you want to [install R for the first time](#).

Binaries of contributed CRAN packages (for R \geq 2.13.x; managed by Uwe Ligges). There is also information on [third party software](#) available for CRAN Windows services and corresponding environment and make variables.

Binaries of contributed CRAN packages for outdated versions of R (for R $<$ 2.13.x; managed by Uwe Ligges).

Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

Please do not submit binaries to CRAN. Package developers might want to contact Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the [R FAQ](#) and [R for Windows FAQ](#).

Note: CRAN does some checks on these binaries for viruses, but cannot give guarantees. Use the normal precautions with downloaded executables.



R-4.1.1 for Windows (32/64 bit)

[Download R 4.1.1 for Windows](#) (86 megabytes, 32/64 bit)



[Installation and other instructions](#)

[New features in this version](#)

CRAN
[Mirrors](#)
[What's new?](#)
[Task Views](#)
[Search](#)

[About R](#)
[R Homepage](#)
[The R Journal](#)

[Software](#)
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Other](#)

[Documentation](#)
[Manuals](#)
[FAQs](#)
[Contributed](#)

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the [md5sum](#) of the .exe to the [fingerprint](#) on the master server. You will need a version of md5sum for windows: both [graphical](#) and [command line versions](#) are available.

Frequently asked questions

- [Does R run under my version of Windows?](#)
- [How do I update packages in my previous version of R?](#)
- [Should I run 32-bit or 64-bit R?](#)

Please see the [R FAQ](#) for general information about R and the [R Windows FAQ](#) for Windows-specific information.

Other builds

- Patches to this release are incorporated in the [r-patched snapshot build](#).
- A build of the development version (which will eventually become the next major release of R) is available in the [r-devel snapshot build](#).
- [Previous releases](#)

Note to webmasters: A stable link which will redirect to the current Windows binary release is <http://<CRAN MIRROR>/bin/windows/base/release.html>.

Last change: 2021-08-10



<https://www.rstudio.com/products/rstudio/download/>

download rstudio - Pesquisa Goo x +

google.com/search?q=download+rstudio&rlz=1C1SQL_pt-BRBR927BR927&oq=download+rstudio&aqs=chrome.0.69i59j0i512l3j69i60.4685j0j4&sourceid=c... ☆ Pausada

Apps Fundação Lemann Salman Khan - Kha... Antonio Artigo copi... Uma Introdução às... Introdução às Rede... Redes Neurais Artifi... Outros favoritos Lista de leitura


Google download rstudio X Ferramentas

Todas Vídeos Imagens Notícias Livros Mais

Aproximadamente 5.980.000 resultados (0,34 segundos)

Dica: Pesquisar apenas resultados em português (Brasil). Especifique seu idioma de pesquisa em Preferências

<https://www.rstudio.com/products/> Traduzir esta página

 **Download the RStudio IDE**

The **RStudio** IDE is a set of integrated tools designed to help you be more productive with R and Python. It includes a console, syntax-highlighting editor that ...

[Older Versions of RStudio](#) · [RStudio Server](#) · [Download RStudio Server for...](#)

<https://www.rstudio.com/products/> Traduzir esta página

RStudio - RStudio

License, AGPL v3, RStudio License Agreement. Pricing, Free, \$995/year. **Download RStudio** Desktop · [DOWNLOAD FREE RSTUDIO DESKTOP PRO TRIAL](#).

<https://www.rstudio.com/products/> Traduzir esta página

Download RStudio Desktop Pro

RStudio requires R 3.0.1 (or higher). If you don't already have R, you can download it here.

Professional Drivers. RStudio Workbench and RStudio Server Pro ...

Windows taskbar: Digite aqui para pesquisar, 31°C Parc ensolarado, 16:21 26/09/2021



DOWNLOAD SUPPORT DOCS COMMUNITY

Products Solutions Customers Resources About Pricing

Download the RStudio IDE

Choose Your Version

The RStudio IDE is a set of integrated tools designed to help you be more productive with R and Python. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust tools for plotting, viewing history, debugging and managing your workspace.



Choose Your Version

The RStudio IDE is a set of integrated tools designed to help you be more productive with R and Python. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust tools for plotting, viewing history, debugging and managing your workspace.

[LEARN MORE ABOUT THE RSTUDIO IDE](#)



RStudio's recommended professional data science solution for every team. [RStudio Team](#) is a bundle of RStudio's popular professional software for data analysis, package management, and sharing data products.

[Learn more about RStudio Team](#)



RStudio Desktop

Open Source License

Free

[DOWNLOAD](#)

[Learn more](#)

RStudio Desktop Pro

Commercial License

\$995

/year

[BUY](#)

[Learn more](#)

RStudio Server

Open Source License

Free

[DOWNLOAD](#)

[Learn more](#)

RStudio Workbench

Commercial License

\$4,975

/year

(5 Named Users)

[BUY](#)

[Evaluation](#) | [Learn more](#)

RStudio Desktop 1.4.1717 - Release Notes

1. Install R. RStudio requires R 3.0.1+.
2. Download RStudio Desktop. Recommended for your system:



Requires Windows 10 (64-bit)



All Installers

Linux users may need to [import RStudio's public code-signing key](#) prior to installation, depending on the operating system's security policy.

RStudio requires a 64-bit operating system. If you are on a 32 bit system, you can use an [older version of RStudio](#).

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

arvore_decisao_aula_41_credit_data.R x Untitled1* x

Source on Save Run Source

1

Onde escrevemos o código

1:1 (Top Level) R Script

Console Terminal x Jobs x

~/...>
R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes da distribuição.
R é um projeto colaborativo.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.
Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.
> |

Project: (None)

Environment History Connections Tutorial

Import Dataset

Global Environment

Environment is empty

Onde visualizamos dados

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home > BCC

	Name	Size	Modified
	..		
	BCC		
	Nootropic Artigo muito bem estruturado.pdf	224.1 KB	Aug 5, 2017, 1:15 PM
	Nootropic Rats.pdf	115.6 KB	Aug 14, 2016, 9:46 AM
	Nootropics A Survey of Substance Use for Cognitive En...	904 KB	Aug 14, 2016, 10:08 AM
	Nootropics Cognitive Enhancement.pdf	390.9 KB	Aug 14, 2016, 10:07 AM
	Nootropics Ethics.pdf	186.4 KB	Aug 14, 2016, 9:46 AM
	nootropicos drogas nootropicas nas promissoes.docx	21.4 KB	Aug 19, 2016, 0:30 PM
	Nootrópicos.docx	16.9 KB	Aug 9, 2016, 9:53 AM
	Nootrópicos_A ética das drogas inteligentes.docx	22.5 KB	Aug 19, 2016, 3:47 PM
	Nootrópicos_A ética das drogas inteligentes.pdf	287 KB	Aug 19, 2016, 12:05 PM
	Perceptron.txt	194 B	Feb 28, 2012, 1:32 PM
	Perceptron_1.pdf	822.4 KB	Feb 28, 2012, 1:13 PM

Onde visualizamos arquivos, gráficos, pacotes, etc.

Base de dados de crédito

Pré-processamento dos dados

Base de dados de crédito - Banco X

- A base de dados de crédito conta com 2.000 registros e, a partir dela será possível fazer uma previsão se o cliente paga ou não paga o empréstimo realizado do banco.
 - Atributos da base de crédito:
 - clientid = variável numérica, nominal
 - income = variável numérica, contínua
 - age = variável numérica, contínua
 - loan = variável numérica, contínua
 - default = variável discreta
- Atributos Previsores
- Atributo Classe
- 0 = não pagou o empréstimo
1 = pagou o empréstimo



- Carregar o R Studio

Go To Folder

← → ↶ ↷ ↻

Este Computa... > Documentos >

Pesquisar Documentos

Organizar Nova pasta

Proj 2

OneDrive

Este Computador

Área de Trabalho

Documentos

Downloads

Imagens

Músicas

Objetos 3D

Vídeos

Windows (C:)

Nome	Data de modificação	Tipo
32 Semana de Informática 2021	22/09/2021 22:02	Pasta de arqu
AnyLogic	17/05/2021 21:00	Pasta de arqu
Artigos	13/09/2021 14:20	Pasta de arqu
Bancas Defesa	27/11/2020 16:48	Pasta de arqu
BCC	15/11/2020 17:07	Pasta de arqu
Carros Doctos 2021	17/04/2021 14:52	Pasta de arqu
CEE	12/08/2021 19:11	Pasta de arqu
Certificados 2021	05/10/2021 11:16	Pasta de arqu
CogSCI 2019	24/03/2021 13:59	Pasta de arqu
Colaão de Grau Renata	24/07/2021 22:43	Pasta de arqu
Colaão Grau João Pedro	24/07/2021 22:42	Pasta de arqu
CRR 2021	06/05/2021 21:17	Pasta de arqu

Pasta:

Open Cancelar

Project: (None)

Environment History Connections Tutorial

Import Dataset

Global Environment

Environment is empty

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home > BCC > BCC > Curso Machine Learning

	Name	Size	Modified
<input type="checkbox"/>	6_Métodos+Preditivos.pdf	653.8 KB	Dec 18, 2020, 10:44 PM
<input type="checkbox"/>	76_SVM Máquinas+de+vetores+de+suporte.pdf	1 MB	Jan 26, 2021, 10:13 PM
<input type="checkbox"/>	7_Métodos+Descritivos.pdf	629.4 KB	Dec 18, 2020, 10:44 PM
<input type="checkbox"/>	8_Tipos+de+Aprendizagem+de+Máquina.pdf	632.7 KB	Dec 19, 2020, 5:49 PM
<input type="checkbox"/>	9_Classificação.pdf	694.2 KB	Dec 19, 2020, 6:09 PM
<input type="checkbox"/>	Análise de Agrupamento.pdf	4.3 MB	Dec 19, 2020, 4:36 PM
<input type="checkbox"/>	arvore_decisao_aula_41_credit_data.R	1.4 KB	Jan 24, 2021, 7:53 PM
<input type="checkbox"/>	arvore_decisao_risco_credito 40.zip	492 B	Jan 22, 2021, 10:14 PM
<input type="checkbox"/>	census.csv	4.2 MB	Dec 30, 2020, 4:42 PM
<input type="checkbox"/>	credit_data.csv	117 KB	Dec 20, 2020, 10:50 PM
<input type="checkbox"/>	Machine Learning SVM.txt	1.4 KB	Jan 26, 2021, 10:47 PM
<input type="checkbox"/>	Machine Learning.pptx	6 MB	Oct 4, 2021, 9:01 PM
<input type="checkbox"/>	Machine Learning_Alunos.pdf	2 MB	Oct 5, 2021, 2:48 PM
<input type="checkbox"/>	Machine Learning_Alunos.pptx	3.1 MB	Oct 5, 2021, 2:47 PM

Guardei o credit_data.csv em
BCC > BCC > Curso Machine Learning

16:23
18/10/2021

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Console Terminal Jobs

```
R version 4.0.3 (2020-10-10) -- "Bunny-wunnies Freak Out"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R é um software livre e vem sem GARANTIA ALGUMA.
você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

> |
```

Environment History Connections Tutorial

Import Dataset

Global Environment

Environment is empty

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home > BCC > BCC > Curso Machine Learning

Name	Size	Modified
8_Tipos+de+Aprendizagem+de+Máquina.pdf	632.7 KB	Dec 19, 2020, 5:49 PM
9_Classificação.pdf	694.2 KB	Dec 19, 2020, 6:09 PM
Análise de Agrupamento.pdf	4.3 MB	Dec 19, 2020, 4:36 PM
arvore_decisao_aula_41_credit_data.R	1.4 KB	Jan 24, 2021, 7:53 PM
arvore_decisao_risco_credito 40.zip	492 B	Jan 22, 2021, 10:14 PM
census.csv	4.2 MB	Dec 30, 2020, 4:42 PM
credit_data.csv	117 KB	Dec 20, 2020, 10:50 PM
Machine	1.4 KB	Jan 26, 2021, 10:47 PM
Machine	6 MB	Oct 4, 2021, 9:01 PM
Machine Learning_Alunos.pdf	2 MB	Oct 5, 2021, 2:48 PM
Machine Learning_Alunos.pptx	3.1 MB	Oct 5, 2021, 2:47 PM
Naive Bayes - Explicação.txt	74 B	Jan 9, 2021, 4:17 PM
naive_bayes_risco_credito.R	41 B	Jan 9, 2021, 6:43 PM
original.zip	1.3 KB	Jan 3, 2021, 10:53 PM

View File Import Dataset...

Clique sobre o arquivo credit_data.csv e, em seguida "View File".

Digite aqui para pesquisar

22°C 16:29 18/10/2021

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

credit_data.csv

Show whitespace

```
1 clientid,income,age,loan,default
2 1,66155.9250950813,59.017015066929204,8106.53213128514,0
3 2,34415.1539658196,48.11715310486029,6564.745017677379,0
4 3,57317.1700630337,63.10804949188599,8020.953296386469,0
5 4,42709.534200839706,45.751972352154596,6103.642260140699,0
6 5,66952.68884534019,18.5843359269202,8770.09923520439,1
7 6,24904.064140282597,57.4716071025468,15.498598437827198,0
8 7,48430.3596126847,26.809132419060898,5722.58198121271,0
9 8,24500.1419843175,32.8975483207032,2971.00330971188,1
10 9,40654.8925372772,55.496852539479704,4755.8252798016,0
11 10,25075.872770976297,39.7763780555688,1409.23037111453,0
12 11,64131.4153722487,25.679575353860898,4351.0289707232505,0
13 12,59436.847122851796,60.4719358547591,9254.24453803174,0
14 13,61050.3460792825,26.3550438545644,5893.26465933928,0
15 14,27267.9954580963,61.576775823254096,4759.7875810455,0
16 15,63061.960174236396,39.2015528911725,1850.36937703064,0
17 16,50501.7266888171,-28.218361321371003,3977.2874324738395,0
18 17,43548.6547113396,39.5745303500444,3935.5444533423497,0
19 18,43378.1751943752,60.84831793932239,3277.7375531263,0
20
```

1:1 Text file

Console Terminal Jobs

R é um software livre e vem sem GARANTIA ALGUMA. Você pode redistribuí-lo sob certas circunstâncias. Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores. Digite 'contributors()' para obter mais informações e 'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda, ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador. Digite 'q()' para sair do R.

> |

Environment History Connections Tutorial

Import Dataset

Global Environment

Environment is empty

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home > BCC > BCC > Curso Machine Learning

	Name	Size	Modified
<input type="checkbox"/>	8_Tipos+de+Aprendizagem+de+Máquina.pdf	632.7 KB	Dec 19, 2020, 5:49 PM
<input type="checkbox"/>	9_Classificação.pdf	694.2 KB	Dec 19, 2020, 6:09 PM
<input type="checkbox"/>	Análise de Agrupamento.pdf	4.3 MB	Dec 19, 2020, 4:36 PM
<input type="checkbox"/>	arvore_decisao_aula_41_credit_data.R	1.4 KB	Jan 24, 2021, 7:53 PM
<input type="checkbox"/>	arvore_decisao_risco_credito 40.zip	492 B	Jan 22, 2021, 10:14 PM
<input type="checkbox"/>	census.csv	4.2 MB	Dec 30, 2020, 4:42 PM
<input type="checkbox"/>	credit_data.csv	117 KB	Dec 20, 2020, 10:50 PM
<input type="checkbox"/>	Machine Learning SVM.txt	1.4 KB	Jan 26, 2021, 10:47 PM
<input type="checkbox"/>	Machine Learning.pptx	6 MB	Oct 4, 2021, 9:01 PM
<input type="checkbox"/>	Machine Learning_Alunos.pdf	2 MB	Oct 5, 2021, 2:48 PM
<input type="checkbox"/>	Machine Learning_Alunos.pptx	3.1 MB	Oct 5, 2021, 2:47 PM
<input type="checkbox"/>	Naive Bayes - Explicação.txt	74 B	Jan 9, 2021, 4:17 PM
<input type="checkbox"/>	naive_bayes_risco_credito.R	41 B	Jan 9, 2021, 6:43 PM
<input type="checkbox"/>	original.zip	1.3 KB	Jan 3, 2021, 10:53 PM

Digite aqui para pesquisar

22°C 16:31 18/10/2021

- Feche o credit_data.csv
- File > New File > R Script
- Salve o arquivo que acabou de criar
 - pre_processamento_credit_data.R
- Clique em MORE
 - Set As Working Directory

#Carregar a base de dados no R Studio

```
base = read.csv('credit_data.csv')
```

Para executar o código: selecione a linha do código e pressione CTRL+ENTER

#Apagar o atributo clientid

```
base$clientid = NULL
```

#Summary: é uma função genérica usada para produzir resumos de resultados.

```
summary(base)
```

#Listar todos os valores de age < 0. Listará todas as colunas com age < 0

```
base[base$age < 0,]
```

#Listar as colunas 1 e 2 com age < 0

```
base[base$age<0, 1:2]
```

#Calcular a média da idade (*age*) e preencher os valores inconsistentes (NA) com a média calculada.

#Faça a média de *age* > 0 e não considere os valores de NA no cálculo:

```
mean(base$age[base$age>0], na.rm = TRUE)
```

#O valor calculado para a média é 40.92

#Se *age* < 0, então atualize *age* para 40.92, senão mantenha o valor de *age*:

```
base$age = ifelse(base$age < 0, 40.92, base$age)
```


#Tratamento de valores faltantes.

#Imprimir todos os valores com *age* = NA

```
base[is.na(base$age),]
```

#Se existe NA em *age*, ENTÃO coloque a média em *age* nos valores faltantes e não

considere os valores de NA no cálculo da média, SENÃO mantenha o valor de *age*.

```
base$age = ifelse(is.na(base$age), mean(base$age, na.rm=TRUE), base$age)
```

Escalonamento de Atributos

- A título de exemplo, pegaremos dois atributos da nossa base de crédito: o *income* e o *age*:

income	age	loan	default
66155.93	59.01702	8106.532131	0
34415.15	48.11715	6564.745018	0
...

- Estes atributos estão em escalas diferentes e para que os algoritmos trabalhem bem sobre esta base, precisaremos executar o escalonamento dos atributos;
- Os algoritmos de aprendizagem de máquina que trabalham com cálculos de distância não conseguirão "aprender corretamente" por causa das discrepâncias entre os dados;
- Repare que os valores do atributo *income* são maiores que os valores do atributo *age*, assim, o algoritmo atribuirá peso maior para o atributo *income* e um peso menor para *age*.

- Ou seja, o algoritmo tornará o atributo *income* como mais importante que o atributo *age* e isso, certamente, influenciará nos resultados do algoritmo de aprendizagem de máquina.
- Nosso objetivo é encontrar uma função que mapeie todos os valores de um atributo para um novo conjunto de valores.
- Dessa forma, precisamos escalonar o *income* e o *age* na mesma escala de modo que seja possível efetuar uma comparação entre os atributos sem qualquer interferência de grandeza.
- Em resumo, o escalonamento ocorre por causa de unidades diferentes ou dispersões muito heterogêneas.
- Existem alguns métodos para realizar o escalonamento de atributos:

Esta técnica é mais robusta, pois ela minimiza os efeitos de pontos *outliers*

- Padronização (*Standardisation*)

$$x = \frac{x - \text{média}(x)}{\text{desvio padrão}(x)}$$

- Normalização (*Normalization*)

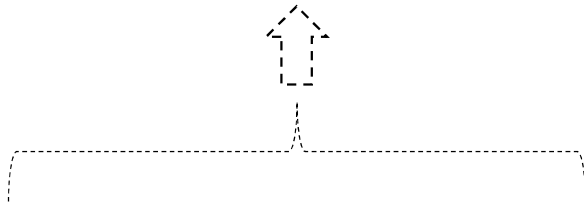
$$x = \frac{x - \text{mínimo}(x)}{\text{máximo}(x) - \text{mínimo}(x)}$$

#Escalonamento de Atributos - Base de crédito

#Se fizermos `base = scale(base)` o R vai escalonar todos os atributos, inclusive o default que é 0 ou 1 e **não queremos** o Escalonamento do default.

#O correto é fazer da seguinte forma:

```
base[, 1:3] = scale(base[, 1:3])
```



(1)	(2)	(3)	(4)
income	age	loan	default
66155.93	59.01702	8106.532131	0
34415.15	48.11715	6564.745018	0
...

Antes do Escalonamento

	income	age	loan	default
1	66155.93	59.01702	8106.532131	0
2	34415.15	48.11715	6564.745018	0
3	57317.17	63.10805	8020.953296	0
4	42709.53	45.75197	6103.642260	0
5	66952.69	18.58434	8770.099235	1
6	24904.06	57.47161	15.498598	0
7	48430.36	26.80913	5722.581981	0
8	24500.14	32.89755	2971.003310	1
9	40654.89	55.49685	4755.825280	0
10	25075.87	39.77638	1409.230371	0

Depois do Escalonamento

	income	age	loan	default
1	1.453570403	1.365039539	1.20251868	0
2	-0.761984978	0.542524516	0.69625282	0
3	0.836611502	1.673753291	1.17441775	0
4	-0.183024288	0.364045494	0.54484373	0
5	1.509185756	-1.686052864	1.42040957	1
6	-1.425873897	1.248421432	-1.45427744	0
7	0.216298258	-1.065401088	0.41971763	0
8	-1.454068294	-0.605962835	-0.48379902	1
9	-0.326441484	1.099404380	0.10227049	0
10	-1.413881386	-0.086879123	-0.99662748	0

Avaliação de algoritmos

- Precisaremos dividir a base de dados em 2:
 - Treinamento e
 - Teste;
- Os registros que estiverem na base de dados TESTE não podem aparecer na base de dados TREINAMENTO;
- O algoritmo aprende com os dados que estão na base de TREINAMENTO e depois verifica o aprendizado com os dados que estão na base de TESTE.

Base de Treinamento e Teste

- Faremos a divisão da base de dados de crédito:
 - Treinamento e
 - Teste;
- Se o seu arquivo ainda não estiver preparado, prepare-o assim:
 - Abrir: `pre_processamento_credit_data.R`
 - Executar todas as instruções de preparação da base de dados:
 - `base = read.csv('credit_data.csv')`
 - `base$clientid = NULL`
 - `base$age = ifelse(base$age < 0, 40.92, base$age)`
 - `base$age = ifelse(is.na(base$age), mean (base$age, na.rm = TRUE), base$age)`
 - `base[, 1:3] = scale(base [, 1:3])`

Base de Treinamento e Teste

- Ok, agora precisamos instalar um pacote do R para a divisão da base de dados:
 - `install.packages('caTools')`
- Para utilizar a biblioteca caTools, faremos:
 - `library(caTools)`
- Para dividir a base de dados:
 - `set.seed(1)`
 - `divisao = sample.split(base$default, SplitRatio = 0.75)`

<i>variável</i>	<i>Utilizado para dividir a base em subconjuntos de Treino e Teste.</i>	<i>Atributo classe. É sobre ele que o algoritmo faz a previsão.</i>	<i>% da base de dados que será utilizada para Treinamento. Dos 2000 registros da base de crédito, 1500 (75%) serão utilizados para Treinar e 500 registros (25%) serão utilizados para Teste.</i>
-----------------	---	---	---

Lembre-se de que os registros precisam ser diferentes, por isso é que dividimos a base de dados.

Base de Treinamento e Teste

- Ao executarmos a instrução:
`divisao = sample.split(base$default, SplitRatio = 0.75)`
- R mostra que a variável ***divisao*** possui diversos TRUE e FALSE:
 - TRUE: são os registros que participarão do TREINAMENTO.
 - Se você digitar `divisao` no prompt do R aparecerá o arquivo com os TRUE e FALSE.

Base de Treinamento e Teste

- **Criando a base de Treinamento:**
 - `base_treinamento = subset(base, divisao == TRUE)`
- **Criando a base de Teste:**
 - `base_teste = subset(base, divisao == FALSE)`

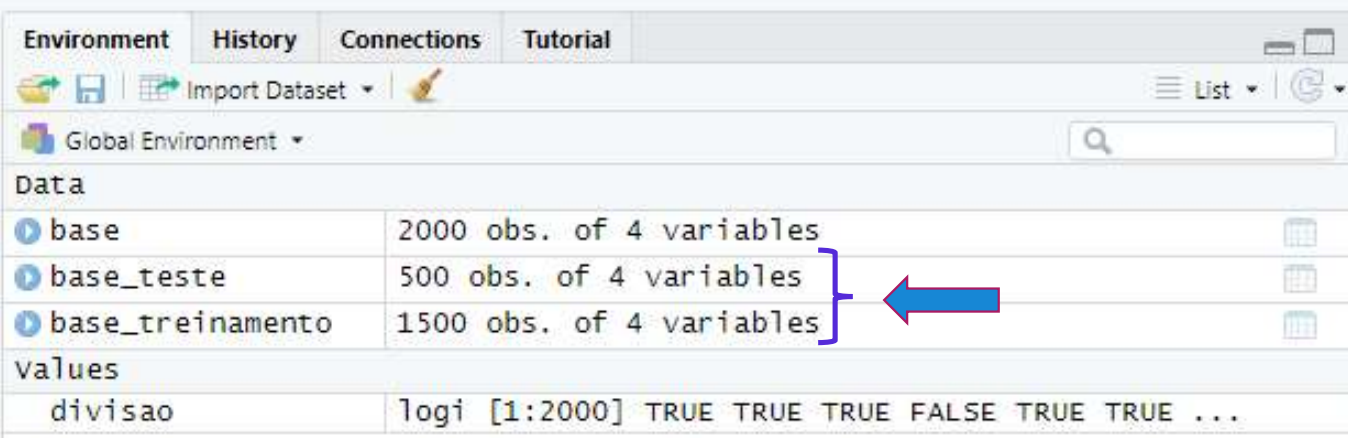


A variável *base_teste*, recebe um subconjunto da variável *base* em que *divisao* = *FALSE*.

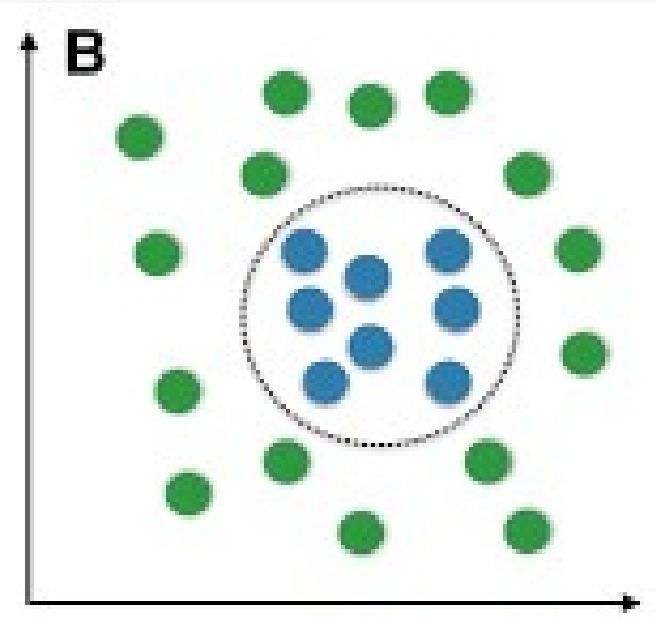
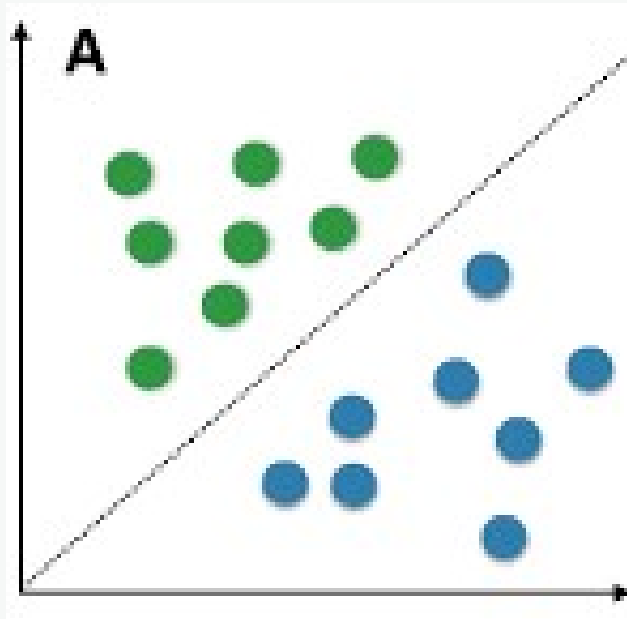
Base de Treinamento e Teste

- **Em resumo...**

Vamos treinar o algoritmo utilizando a *base_treinamento* (1500 registros) e vamos testar o aprendizado utilizando a *base_teste* (500 registros).



Environment		History	Connections	Tutorial
Global Environment		Import Dataset		
Data				
base	2000 obs. of 4 variables			
base_teste	500 obs. of 4 variables			
base_treinamento	1500 obs. of 4 variables			
values				
divisao	logi [1:2000] TRUE TRUE TRUE FALSE TRUE TRUE ...			



Naive Bayes

Introdução

- Abordagem probabilística (Teorema de Bayes)
- O algoritmo de Naive Bayes trabalha sobre base de dados históricos e gera uma tabela de probabilidades indicando a chance de ocorrências dos atributos chave.
- Exemplos:
 - Filtros de spam
 - Mineração de emoções
 - Separação de documentos.

- Lembremos do 1º exemplo, da base de dados do Risco de Crédito:
 - Quando um novo cliente procurar pelo banco, submeteremos os dados desse cliente ao sistema e ele retornará o “risco” de conceder empréstimo.

Base original

Atributo Classe



História do crédito	Dívida	Garantias	Renda anual	Risco
Ruim	Alta	Nenhuma	< 15.000	Alto
Desconhecida	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto
Desconhecida	Baixa	Nenhuma	>= 15.000 a <= 35.000	Moderado
Desconhecida	Baixa	Nenhuma	> 35.000	Alto
Desconhecida	Baixa	Nenhuma	> 35.000	Baixo
Desconhecida	Baixa	Adequada	> 35.000	Baixo
Ruim	Baixa	Nenhuma	< 15.000	Alto
Ruim	Baixa	Adequada	> 35.000	Moderado
Boa	Baixa	Nenhuma	> 35.000	Baixo
Boa	Alta	Adequada	> 35.000	Baixo
Boa	Alta	Nenhuma	< 15.000	Alto
Boa	Alta	Nenhuma	>= 15.000 a <= 35.000	Moderado
Boa	Alta	Nenhuma	> 35.0000	Baixo
Ruim	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto

O objetivo aqui é encontrar alguma correlação entre os atributos previsores de forma a indicar o risco em caso de conceder o empréstimo.

Atributos previsores

Atributo
Classe

O objetivo do algoritmo de Naive Bayes é analisar os dados da base de dados histórica e gerar uma tabela de probabilidades conforme essa aqui:

Naive Bayes

6 ocorrências de risco "Alto" de um total de 14.

1 ocorrência de Risco "Alto" e "Boa" História Crédito.

Risco de crédito	História do crédito			Dívida		Garantias		Renda anual		
	Boa 5	Desconhecida 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	< 15000 3	>= 15000 <= 35000 4	> 35000 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

Em resumo, o algoritmo Naive Bayes vai gerar uma tabela de probabilidades para as classes de risco de crédito.

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4							
Alto 6/14	1/6	2/6	3/6							
Moderado 3/14	1/3	1/3	1/3							
Baixo 5/14	3/5	2/5	0							

1. História de Crédito: somar quantidade de "Boa", "Desconhecida" e "Ruim".
2. Risco de Crédito: somar quantidade de "alto", "moderado" e "baixo".
3. Cruzar "Risco de Crédito" com "História de Crédito": Boa - Alto: existe apenas uma (01) ocorrência de uma possibilidade de seis (06).
4. Executar o mesmo procedimento para os demais.

História de crédito	Risco
Ruim	Alto
Desconhecida	Alto
Desconhecida	Moderado
Desconhecida	Alto
Desconhecida	Baixo
Desconhecida	Baixo
Ruim	Alto
Ruim	Moderado
Boa	Baixo
Boa	Baixo
Boa	Alto
Boa	Moderado
Boa	Baixo
Ruim	Alto

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7					
Alto 6/14	1/6	2/6	3/6	4/6	2/6					
Moderado 3/14	1/3	1/3	1/3	1/3	2/3					
Baixo 5/14	3/5	2/5	0	2/5	3/5					

1. Dívida: somar quantidade de “Alta” e “Baixa”.
2. Cruzar “Risco de Crédito” com “Dívida”: Alto – Alta: existem apenas quatro (04) ocorrências de uma possibilidade de seis (06).
3. Executar o mesmo procedimento para os demais.

Dívida	Risco
Alta	Alto
Alta	Alto
Baixa	Moderado
Baixa	Alto
Baixa	Baixo
Baixa	Baixo
Baixa	Alto
Baixa	Moderado
Baixa	Baixo
Alta	Baixo
Alta	Alto
Alta	Moderado
Alta	Baixo
Alta	Alto

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

1. Dívida: somar quantidade de "< 15", ">=15 a < 35" e "> 35".
2. Cruzar "Risco de Crédito" com "Renda Anual": Alto com < 15.000 existem apenas três (03) ocorrências de uma possibilidade de seis (06).
3. Executar o mesmo procedimento para os demais.

Garantias	Risco
< 15.000	Alto
>=15.000 a <=35.000	Alto
>=15.000 a <=35.000	Moderado
> 35.000	Alto
> 35.000	Baixo
> 35.000	Baixo
< 15.000	Alto
> 35.000	Moderado
> 35.000	Baixo
> 35.000	Baixo
< 15.000	Alto
>=15.000 a <=35.000	Moderado
> 35.000	Baixo
>=15.000 a <=35.000	Alto

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

Esta tabela de probabilidades é o aprendizado de máquina baseado no algoritmo de Naive Bayes.

Como o Naive Bayes faz a classificação de novos registros?

- *Vamos imaginar que chegou um novo CLIENTE no banco e solicitou empréstimo.*
- *O Gerente precisa saber qual é o **risco** para conceder empréstimo para este novo Cliente.*
- *O Gerente colheu os seguintes dados do Cliente:*
 - **História de Crédito:** BOA
 - **Dívida:** Alta
 - **Garantias:** Nenhuma
 - **Renda:** > 35.0000

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

Vamos submeter os dados do novo Cliente ao sistema e aguardar a resposta, que pode ser:

Risco ALTO

Risco MODERADO

Risco BAIXO

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

- Vamos selecionar as partes da tabela de acordo com as características do novo Cliente:
História de Crédito: BOA
Dívida: Alta
Garantias: Nenhuma
Renda: > 35.0000

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

- Em seguida, iremos estimar as probabilidades para cada classe:

$P(\text{Alto}) = 6/14 * 1/6 * 4/6 * 6/6 * 1/6$

$P(\text{Alto}) = 0.0079$

$P(\text{Moderado}) = 3/14 * 1/3 * 1/3 * 2/3 * 1/3$

$P(\text{Moderado}) = 0.0052$

$P(\text{Baixo}) = 5/14 * 3/5 * 2/5 * 3/5 * 5/5$

$P(\text{Baixo}) = 0.0514$

$Soma = 0.0079 + 0.0052 + 0.0514 = \mathbf{0.0645}$

$P(\text{Alto}) = 0.0079/0.0645 * 100 = \mathbf{12.24\%}$

$P(\text{Moderado}) = 0.0052/0.0645 * 100 = \mathbf{8.06\%}$

$P(\text{Baixo}) = 0.0514/0.0645 * 100 = \mathbf{79.70\%}$

$$P(\text{Alto}) = 0.0079/0.0645 * 100 = \mathbf{12.24\%}$$

$$P(\text{Moderado}) = 0.0052/0.0645 * 100 = \mathbf{8.06\%}$$

$$P(\text{Baixo}) = 0.0514/0.0645 * 100 = \mathbf{79.70\%}$$

- *Assim, o algoritmo de Naive Bayes informa que o risco de conceder o empréstimo é BAIXO.*
- *O banco tem aproximadamente 80% de chance de receber o \$ emprestado.*

Outro exemplo...

- Chegou um novo CLIENTE no banco e também solicitou empréstimo.
- O Gerente precisa saber qual é o **risco** para conceder empréstimo para este novo Cliente.
- O Gerente colheu os seguintes dados do Cliente:
 - **História de Crédito:** RUIM
 - **Dívida:** ALTA
 - **Garantias:** ADEQUADA
 - **Renda:** < 15.0000

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

- Vamos selecionar as partes da tabela de acordo com as características do novo Cliente:
História de Crédito: RUIM
Dívida: ALTA
Garantias: ADEQUADA
Renda: < 15.0000

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

- Em seguida, iremos estimar as probabilidades para cada classe:

$$P(\text{Alto}) = 6/14 * 3/6 * 4/6 * 0 * 3/6$$

$$P(\text{Alto}) = 0.0$$

$$P(\text{Moderado}) = 3/14 * 1/3 * 1/3 * 1/3 * 0$$

$$P(\text{Moderado}) = 0.0$$

$$P(\text{Baixo}) = 5/14 * 0 * 2/5 * 2/5 * 0$$

$$P(\text{Baixo}) = 0.0$$

Como todas as probabilidades deram **0**, o algoritmo Naive Bayes utiliza um artifício chamado de **Correção Laplaciana**.

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 4	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 5/14	3/5	2/5	0	2/5	3/5	3/5	2/5	0	0	5/5

Correção Laplaciana

- Note que, em Risco de Crédito BAIXO e História de Crédito RUIM, não existe nenhum registro.
- A Correção Laplaciana consiste em adicionar valores de 1 para não ocorrer multiplicações por 0.
- Entretanto, ao adicionar 1 no lugar do 0, **alteramos todos os valores** da tabela de probabilidades, mas não teremos probabilidades = 0.

	História de Crédito			Dívida		Garantias		Renda Anual		
Risco de crédito	Boa 5	Desc 5	Ruim 5	Alta 7	Baixa 7	Nenhuma 11	Adequada 3	<15 3	>=15 <=35 4	>35 7
Alto 6/14	1/6	2/6	3/6	4/6	2/6	6/6	0	3/6	2/6	1/6
Moderado 3/14	1/3	1/3	1/3	1/3	2/3	2/3	1/3	0	2/3	1/3
Baixo 6/15	3/6	2/5	1/6	2/5	3/5	3/5	2/5	0	0	5/5

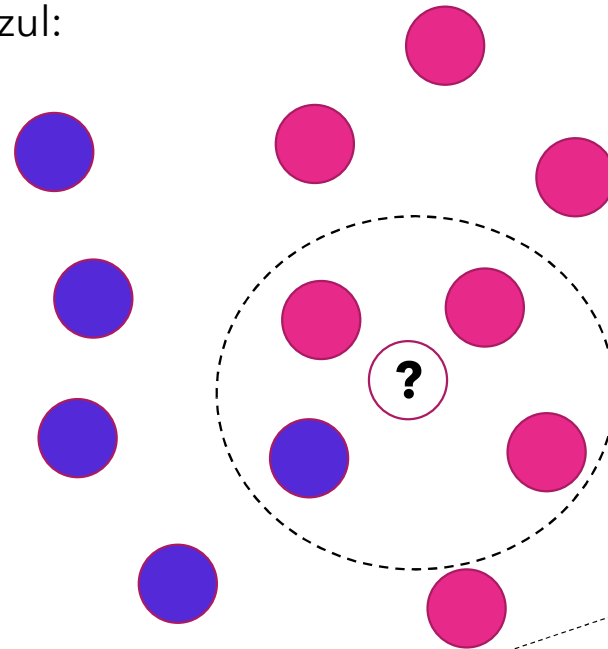
Exemplo de Correção Laplaciana

- Os novos valores de Risco de Crédito BAIXO e História de Crédito RUIM, ficariam dessa forma.
- Os demais valores 0 devem passar pelo mesmo processo.
- O algoritmo faz, automaticamente, todas as alterações.

Naive Bayes - mais conceitos

- A probabilidade do ponto ser vermelho:
 $P'(\text{vermelho}) = 3/7$

- A probabilidade do ponto ser azul:
 $P'(\text{azul}) = 1/5$



- Consideremos duas classes distintas: **vermelha** e **azul**.
- Existe um ponto que desejamos descobrir a qual classe pertence.
- Precisaremos fazer as estimativas para entender a qual classe o ponto pertence.

- A probabilidade de ser vermelho é:
 $P(\text{vermelho}) = 7/12$

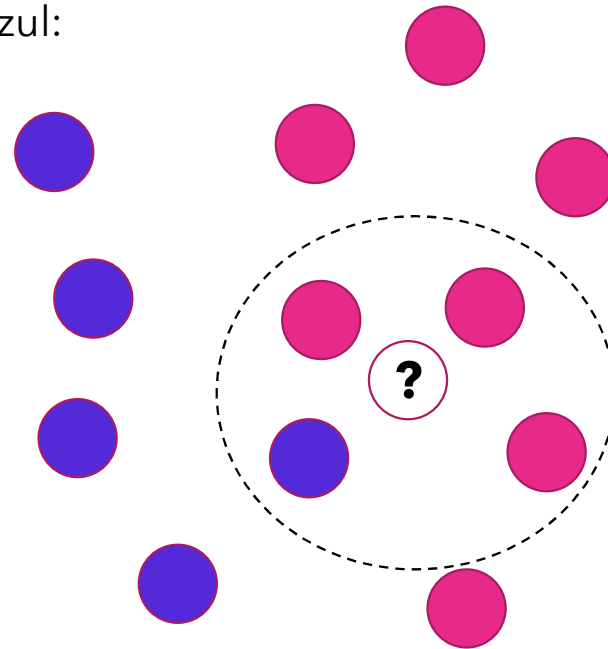
- A probabilidade de ser azul é:
 $P(\text{azul}) = 5/12$

Probabilidades a priori

É a informação que se tem sobre o evento A **antes** que se saiba algo sobre o evento B.

O algoritmo de Naive Bayes possui um parâmetro chamado *radius* que define a vizinhança ao redor do ponto de interesse.

- A probabilidade do ponto ser vermelho:
 $P'(\text{vermelho}) = 3/7$
- A probabilidade do ponto ser azul:
 $P'(\text{azul}) = 1/5$



- A probabilidade de ser vermelho é:
 $P(\text{vermelho}) = 7/12$
- A probabilidade de ser azul é:
 $P(\text{azul}) = 5/12$

Resultado final:

$$P''(\text{vermelho}) = 7/12 * 3/7$$

$$P''(\text{vermelho}) = 21/84 = \mathbf{0,25}$$

$$P''(\text{azul}) = 5/12 * 1/5$$

$$P''(\text{azul}) = 5/60 = \mathbf{0,08}$$

A probabilidade do ponto ? Ser **vermelho** é maior.

Probabilidades a priori

É a informação que se tem sobre o evento A **antes** que se saiba algo sobre o evento B.

Probabilidades a posteriori

Quando se tem informação sobre o evento B, a probabilidade $P(A|B)$ é chamada de a posteriori.

Algoritmo Naive Bayes

- **Vantagens**

- Rápido
- Simplicidade de interpretação
- Trabalha com muitos atributos
- Boas previsões em bases de dados pequenas (10.000 registros)

- **Desvantagens**

- O algoritmo considera que cada par de característica é independente (p.ex. Renda e Dívida. Para o algoritmo Renda não tem correlação com Dívida) e isso nem sempre é verdadeiro.

Executando as bases de Treinamento e de Teste

Leitura da base de dados

```
base = read.csv('credit_data.csv')
```

Apaga a coluna clientid

```
base$clientid = NULL
```

Valores inconsistentes

```
base$age = ifelse(base$age < 0, 40.92, base$age)
```

Valores faltantes

```
base$age = ifelse(is.na(base$age),  
mean(base$age, na.rm = TRUE), base$age)
```

Escalonamento

```
base[, 1:3] = scale(base[, 1:3])
```

#Encode da classe

```
base$default = factor(base$default, levels = c(0,1))
```

Divisão entre treinamento e teste

```
install.packages('caTools')
```

```
library(caTools)
```

```
set.seed(1)
```

```
divisao = sample.split(base$income, SplitRatio = 0.75)
```

```
base_treinamento = subset(base, divisao == TRUE)
```

```
base_teste = subset(base, divisao == FALSE)
```

#Importar o pacote do algoritmo Naive Bayes

```
install.packages('e1071')
```

```
library(e1071)
```

#Criar a variável classificador que recebe o método naiveBayes(x,y). Onde:

x: passar os atributos previsores (*income, age e loan*)

y: passar o atributo classe (*default*)

#O Naive Bayes gera a tabela de probabilidades e armazena em *classificador*

```
classificador = naiveBayes(x = base_treinamento[-4], base_treinamento$default)
```

#Digitar no prompt (resultado está na próxima tela) - O R mostra a tabela de probabilidades após o treinamento:

```
print(classificador)
```

```
> print(classificador)
```

```
Naive Bayes Classifier for Discrete Predictors
```

```
Call:
```

```
naiveBayes.default(x = base_treinamento[-4], y = base_treinamento$default)
```

```
A-priori probabilities:
```

```
base_treinamento$default
```

```
      0      1  
0.8586667 0.1413333
```

*Probabilidades **a priori***

85.8% são registros da Classe 0 (quem não paga) e

14% da Classe 1 (quem paga)

```
Conditional probabilities:
```

```
      income
```

```
base_treinamento$default      [,1]      [,2]
```

```
0 -0.004816169 1.001908
```

```
1  0.028792097 1.014477
```

```
      age
```

```
base_treinamento$default      [,1]      [,2]
```

```
0  0.1946204 0.9510221
```

```
1 -1.0833115 0.3715987
```

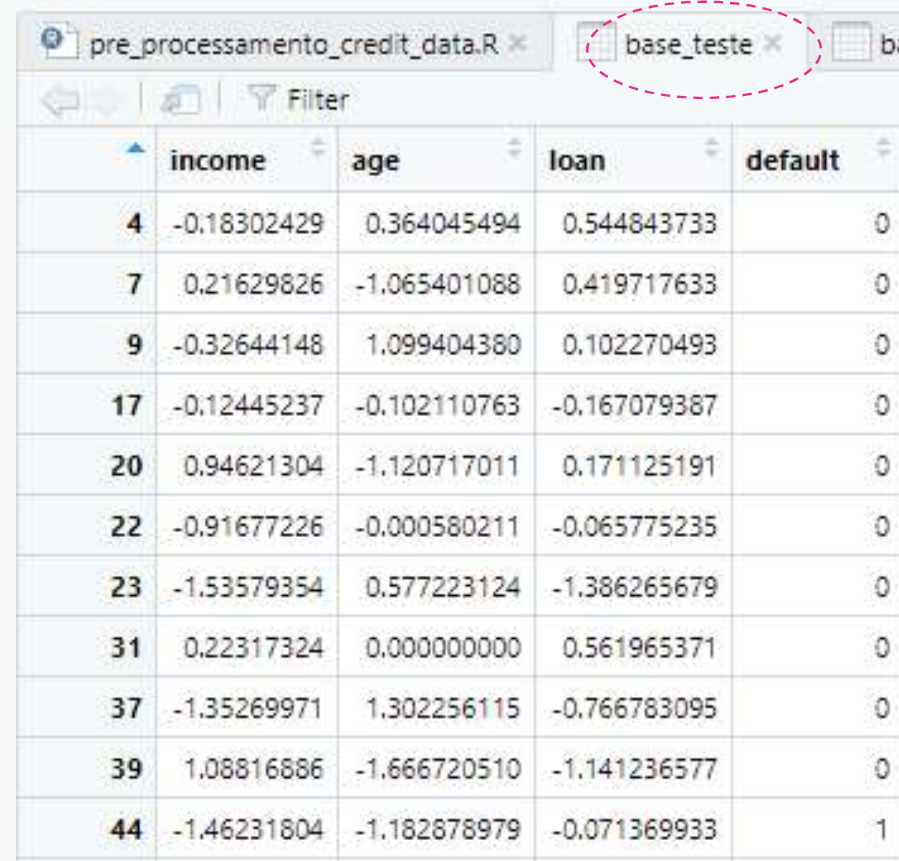
```
      loan
```

```
base_treinamento$default      [,1]      [,2]
```

```
0 -0.1633050 0.9335055
```

```
1  0.9504478 0.8486380
```


- Lembre-se de que estamos lidando com “aprendizagem supervisionada”, ou seja, já sabemos o resultado da Classe *default* (0: não pagou o empréstimo e 1: pagou).
- O objetivo aqui é comparar a resposta do Naive Bayes com os resultados armazenados na base de dados de crédito.



	income	age	loan	default
4	-0.18302429	0.364045494	0.544843733	0
7	0.21629826	-1.065401088	0.419717633	0
9	-0.32644148	1.099404380	0.102270493	0
17	-0.12445237	-0.102110763	-0.167079387	0
20	0.94621304	-1.120717011	0.171125191	0
22	-0.91677226	-0.000580211	-0.065775235	0
23	-1.53579354	0.577223124	-1.386265679	0
31	0.22317324	0.000000000	0.561965371	0
37	-1.35269971	1.302256115	-0.766783095	0
39	1.08816886	-1.666720510	-1.141236577	0
44	-1.46231804	-1.182878979	-0.071369933	1

#Criar uma variável denominada `matriz_confusao` que fará a comparação das previsões (armazenadas na variável `previsoes`) com o atributo classe `default` da `base_teste`.

```
matriz_confusao = table(base_teste[,4], previsoes)
```

#Ao digitar:

```
print(matriz_confusao)
```

#O R mostrará:

		Previsões	
		0	1
0		416	13
1		29	42

base_teste (default) = 0 e previsao = 0 416 registros corretos

base_teste (default) = 1 e previsao = 1 42 registros corretos

Se quisermos saber o **percentual de acertos** do algoritmo:

$416 + 42 = 458 / 500$ registros = **91,6%**

Se quisermos saber o **percentual de erros** do algoritmo:

$29 + 13 = 42 / 500$ registros = **8,4%**

#Agora vamos avaliar automaticamente o desempenho do algoritmo de Naive Bayes:

```
install.packages('caret')
```

```
library(caret)
```

```
confusionMatrix(matriz_confusao)
```

#R vai mostrar que a “precisão” do algoritmo foi de 91.6%. Este valor é estatisticamente significativo (veja P-value):

```
      Accuracy : 0.916
      95% CI   : (0.8882, 0.9388)
No Information Rate : 0.89
P-Value [Acc > NIR] : 0.03340 ← < 0.05

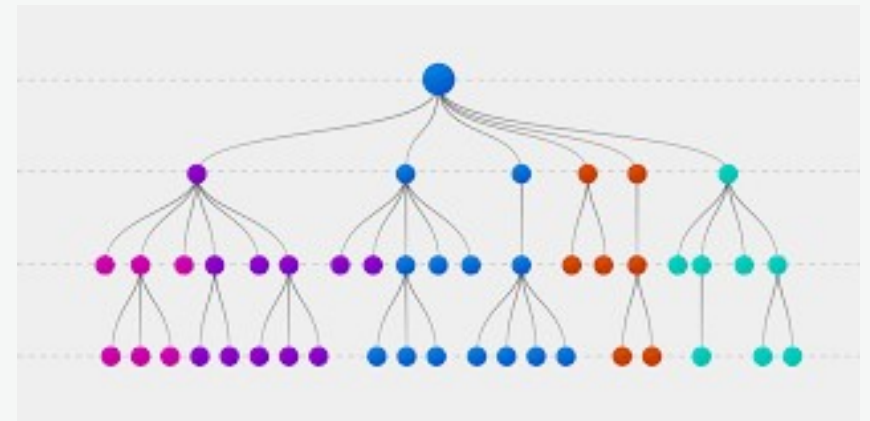
      Kappa : 0.6195

McNemar's Test P-Value : 0.02064

      Sensitivity : 0.9348
      Specificity : 0.7636
      Pos Pred value : 0.9697
      Neg Pred value : 0.5915
      Prevalence : 0.8900
      Detection Rate : 0.8320
      Detection Prevalence : 0.8580
      Balanced Accuracy : 0.8492

      'Positive' class : 0
```

Árvores de Decisão



Árvores de Decisão

- Utilizaremos a base de dados do Risco de Crédito para saber qual é o risco associado aos dados históricos do Banco X:
 - Alto
 - Moderado
 - Baixo
- Nosso objetivo é estudar o funcionamento do algoritmo de Árvores de Decisão.

Base original

Atributo Classe

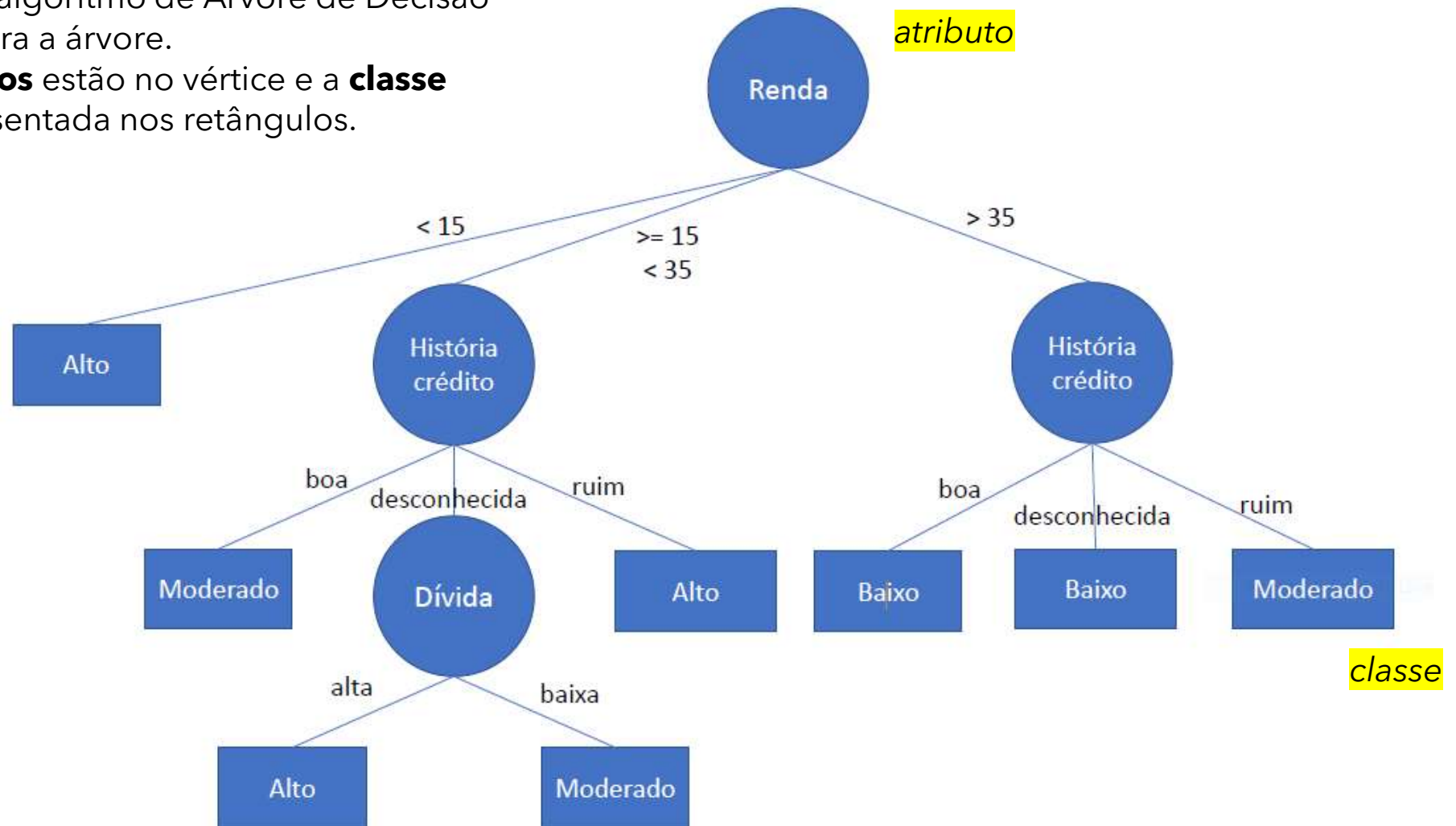


História do crédito	Dívida	Garantias	Renda anual	Risco
Ruim	Alta	Nenhuma	< 15.000	Alto
Desconhecida	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto
Desconhecida	Baixa	Nenhuma	>= 15.000 a <= 35.000	Moderado
Desconhecida	Baixa	Nenhuma	> 35.000	Alto
Desconhecida	Baixa	Nenhuma	> 35.000	Baixo
Desconhecida	Baixa	Adequada	> 35.000	Baixo
Ruim	Baixa	Nenhuma	< 15.000	Alto
Ruim	Baixa	Adequada	> 35.000	Moderado
Boa	Baixa	Nenhuma	> 35.000	Baixo
Boa	Alta	Adequada	> 35.000	Baixo
Boa	Alta	Nenhuma	< 15.000	Alto
Boa	Alta	Nenhuma	>= 15.000 a <= 35.000	Moderado
Boa	Alta	Nenhuma	> 35.0000	Baixo
Ruim	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto

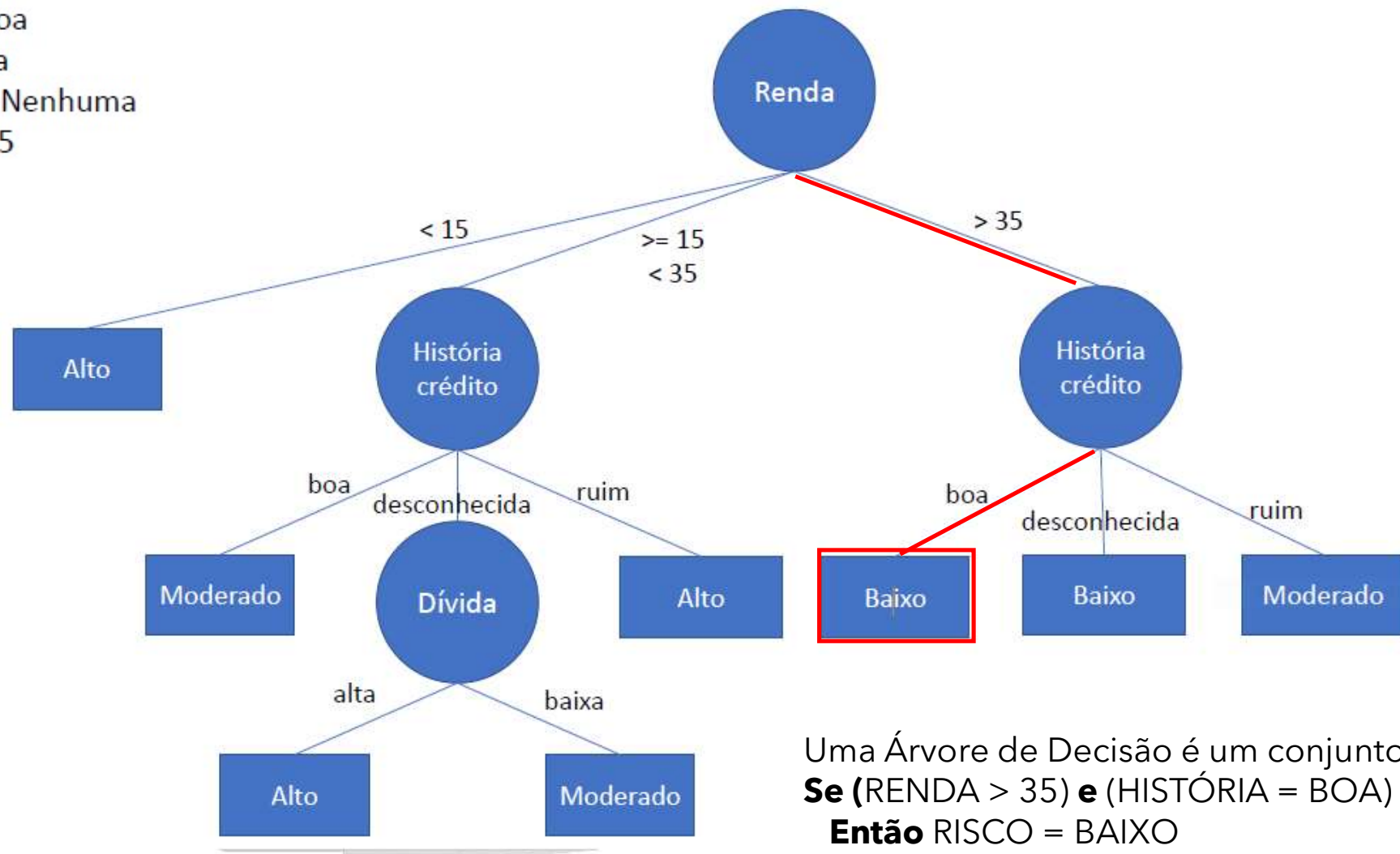
Atributos previsores

O próprio algoritmo de Árvore de Decisão é quem gera a árvore.

Os **atributos** estão no vértice e a **classe** está representada nos retângulos.

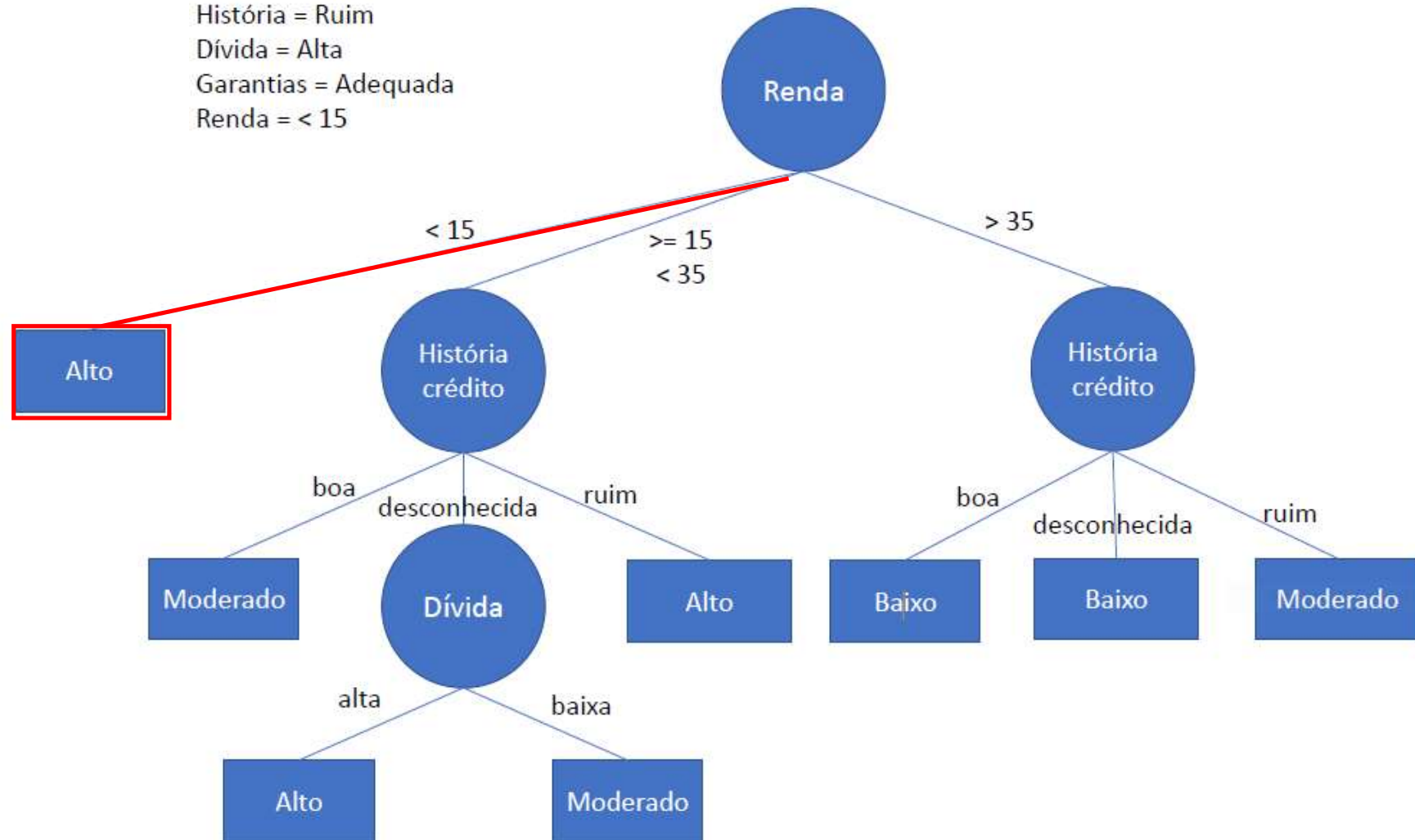


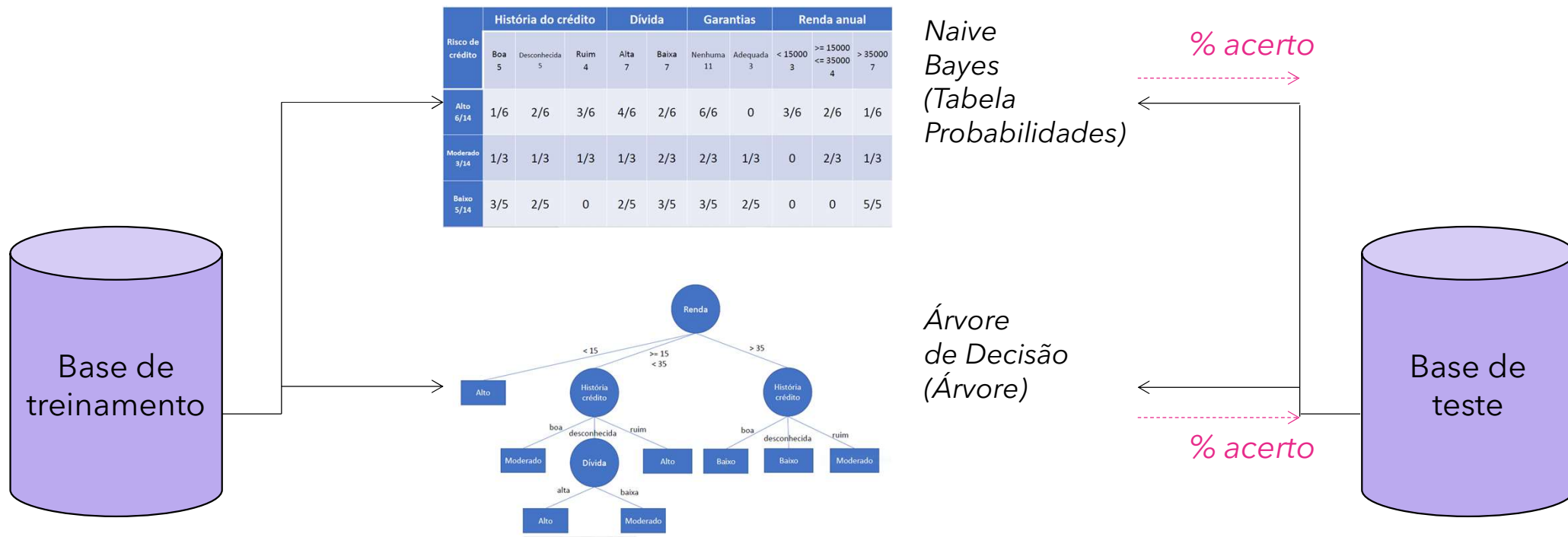
História = Boa
Dívida = Alta
Garantias = Nenhuma
Renda = > 35



Uma Árvore de Decisão é um conjunto de regras:
Se (RENDA > 35) **e** (HISTÓRIA = BOA)
Então RISCO = BAIXO

História = Ruim
Dívida = Alta
Garantias = Adequada
Renda = < 15





A partir da Base de Treinamento:

- No algoritmo **Naive Bayes**: é gerada a Tabela de Probabilidades
- No algoritmo de **Árvore de Decisão**: é gerada uma árvore

Para fazer a avaliação desses algoritmos:

- Precisamos de uma Base de Teste;
- Os registros da Base de Testes são submetidos aos 2 algoritmos e obteremos como resposta o % de acerto de cada um.

Árvores de Decisão - Aprendizagem

- Objetivo:
 - Estudar como é o processo de treinamento (ou geração) de uma Árvore de Decisão.

Árvores de Decisão - Aprendizagem

- Algoritmo de Árvore de Decisão necessita de cálculos para gerar a referida árvore:

Cálculo da
Entropia

$$Entropy(S) = \sum_{i=1}^c -p_i \log_2 p_i$$

Cálculo do
ganho de
informação

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

A ideia é analisar atributo por atributo para descobrir qual deles é o mais importante e que ficará nos ramos superiores da árvore.

Cálculo da Entropia

Risco
Alto ✓
Alto ✓
Moderado
Alto ✓
Baixo
Baixo
Alto ✓
Moderado
Baixo
Baixo
Alto ✓
Moderado
Baixo
Alto ✓

Alto = 6/14

Moderado = 3/14

Baixo = 5/14

$$Entropy(S) = \sum_{i=1}^c -p_i \log_2 p_i$$

$$E(s) = -6/14 * \log(6/14; 2) - 3/14 * \log(3/14; 2) - 5/14 * \log(5/14; 2) = 1,53$$

É a *entropia* geral ou a *entropia* dos valores Alto, Moderado e Baixo.

A Teoria da Informação estabelece que a *Entropia* mede o grau de organização/desorganização dos dados armazenados na base.

Cálculo do Ganho da Informação

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Precisaremos calcular:

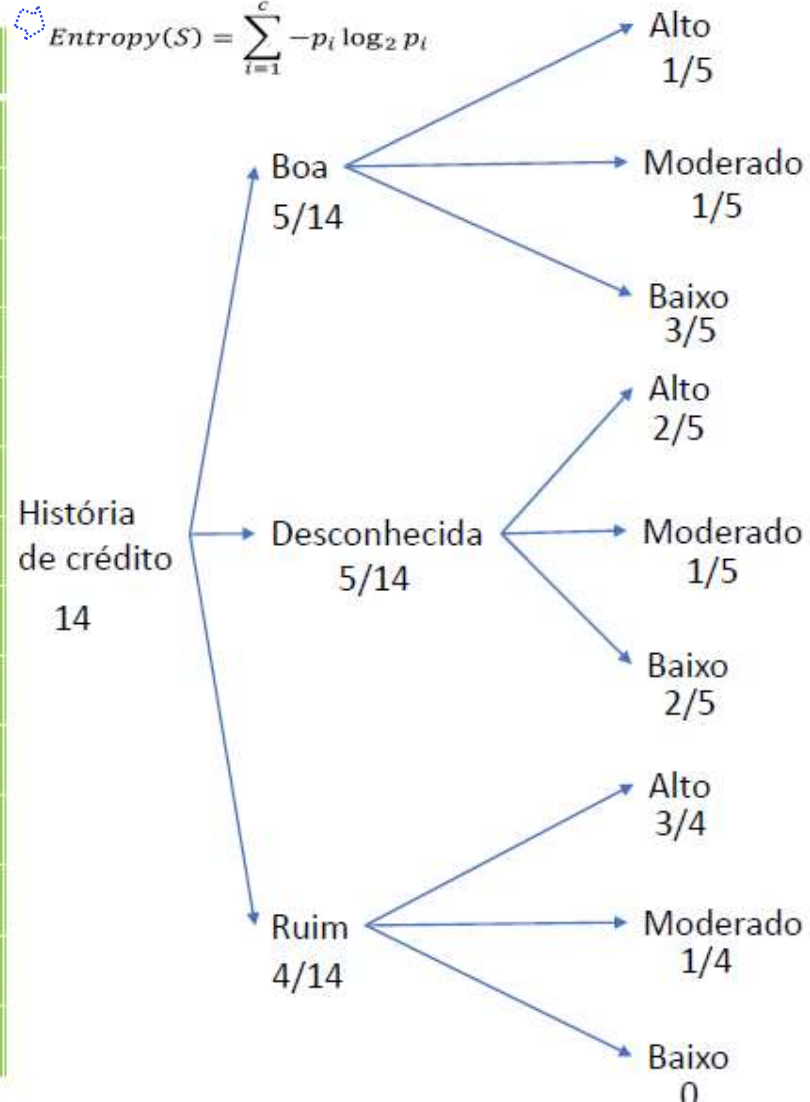
- Ganho de Informação (História de Crédito)
- Ganho de Informação (Dívida)
- Ganho de Informação (Garantias)
- Ganho de Informação (Renda Anual)

O atributo com o maior Ganho de Informação será considerado o atributo mais significativo (ou + importante). Este atributo ficará no topo da Árvore de Decisão.

Cálculo do Ganho da Informação (História de Crédito)

História do crédito	Risco
Ruim	Alto
Desconhecida	Alto
Desconhecida	Moderado
Desconhecida	Alto
Desconhecida	Baixo
Desconhecida	Baixo
Ruim	Alto
Ruim	Moderado
Boa	Baixo
Boa	Baixo
Boa	Alto
Boa	Moderado
Boa	Baixo
Ruim	Alto

$$Entropy(S) = \sum_{i=1}^c -p_i \log_2 p_i$$



$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$E(s) = -1/5 * \log(1/5; 2) - 1/5 * \log(1/5; 2) - 3/5 * \log(3/5; 2) = \mathbf{1,37}$$

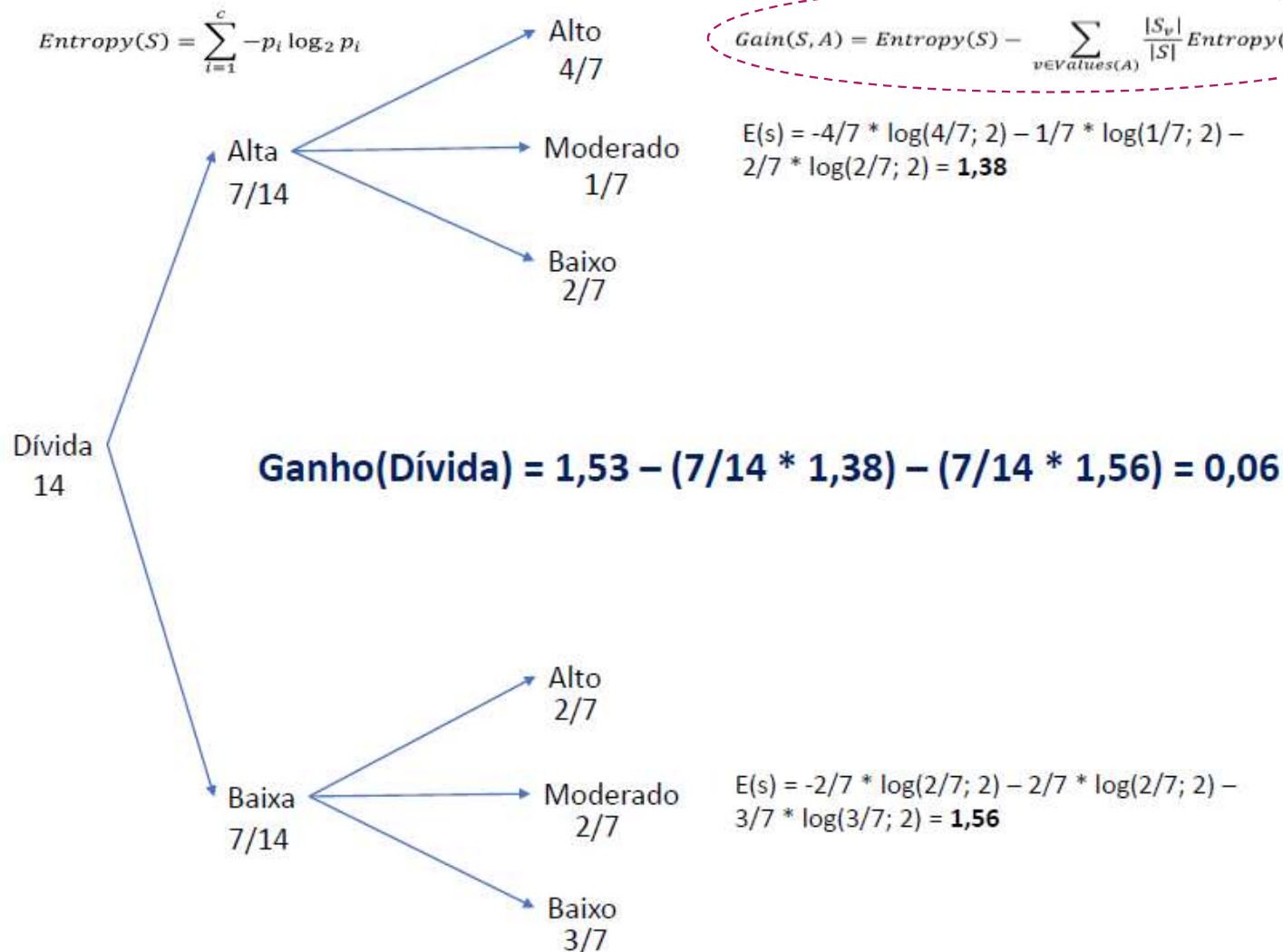
$$E(s) = -2/5 * \log(2/5; 2) - 1/5 * \log(1/5; 2) - 2/5 * \log(2/5; 2) = \mathbf{1,52}$$

$$\mathbf{Ganho(História) = 1,53 - (5/14 * 1,37) - (5/14 * 1,52) - (4/14 * 0,81) = 0,26}$$

$$E(s) = -3/4 * \log(3/4; 2) - 1/4 * \log(1/4; 2) - 0 * \log(0; 2) = \mathbf{0,81}$$

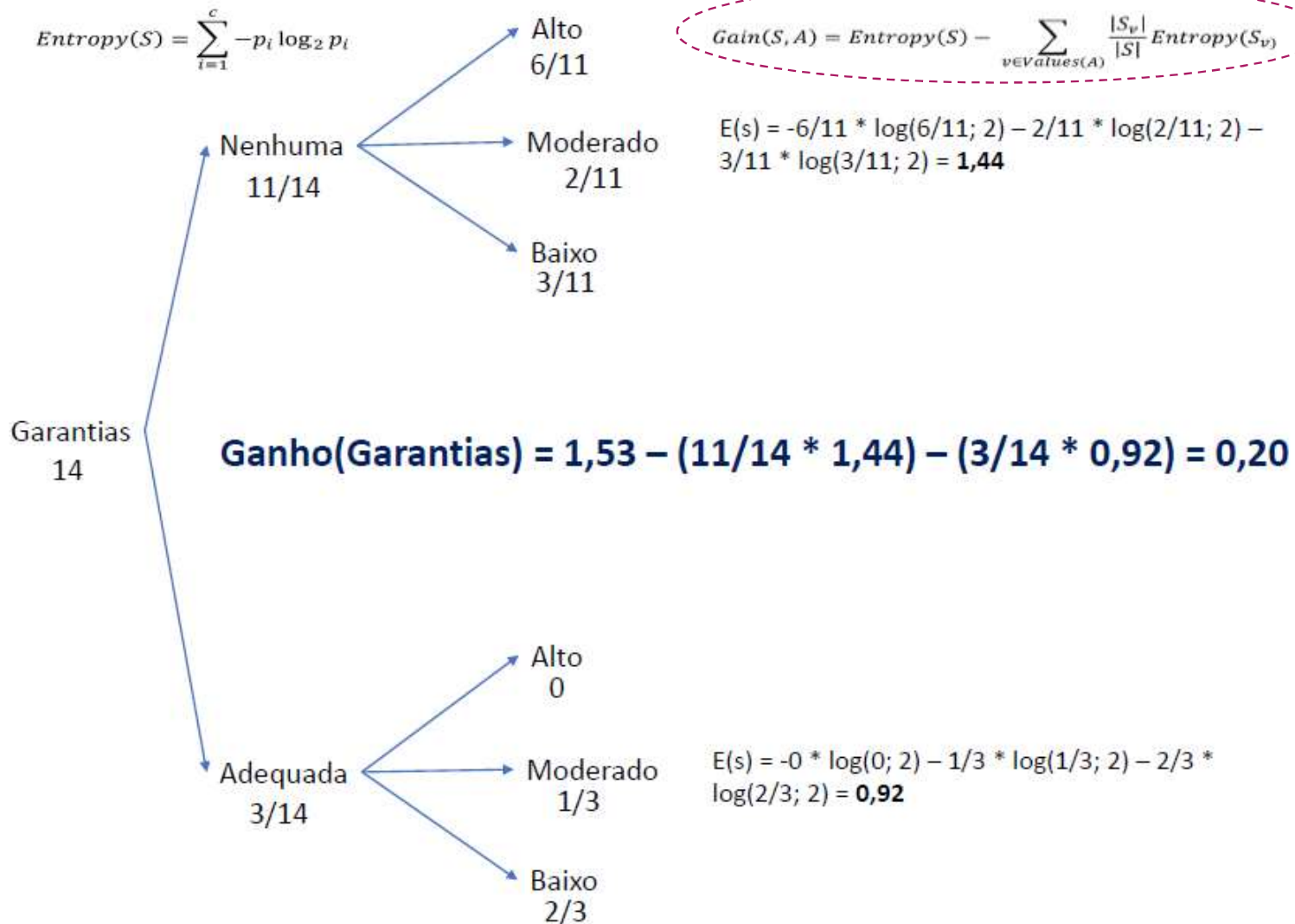
Cálculo do Ganho da Informação (Dívida)

Dívida	Risco
Alta	Alto
Alta	Alto
Baixa	Moderado
Baixa	Alto
Baixa	Baixo
Baixa	Baixo
Baixa	Alto
Baixa	Moderado
Baixa	Baixo
Alta	Baixo
Alta	Alto
Alta	Moderado
Alta	Baixo
Alta	Alto



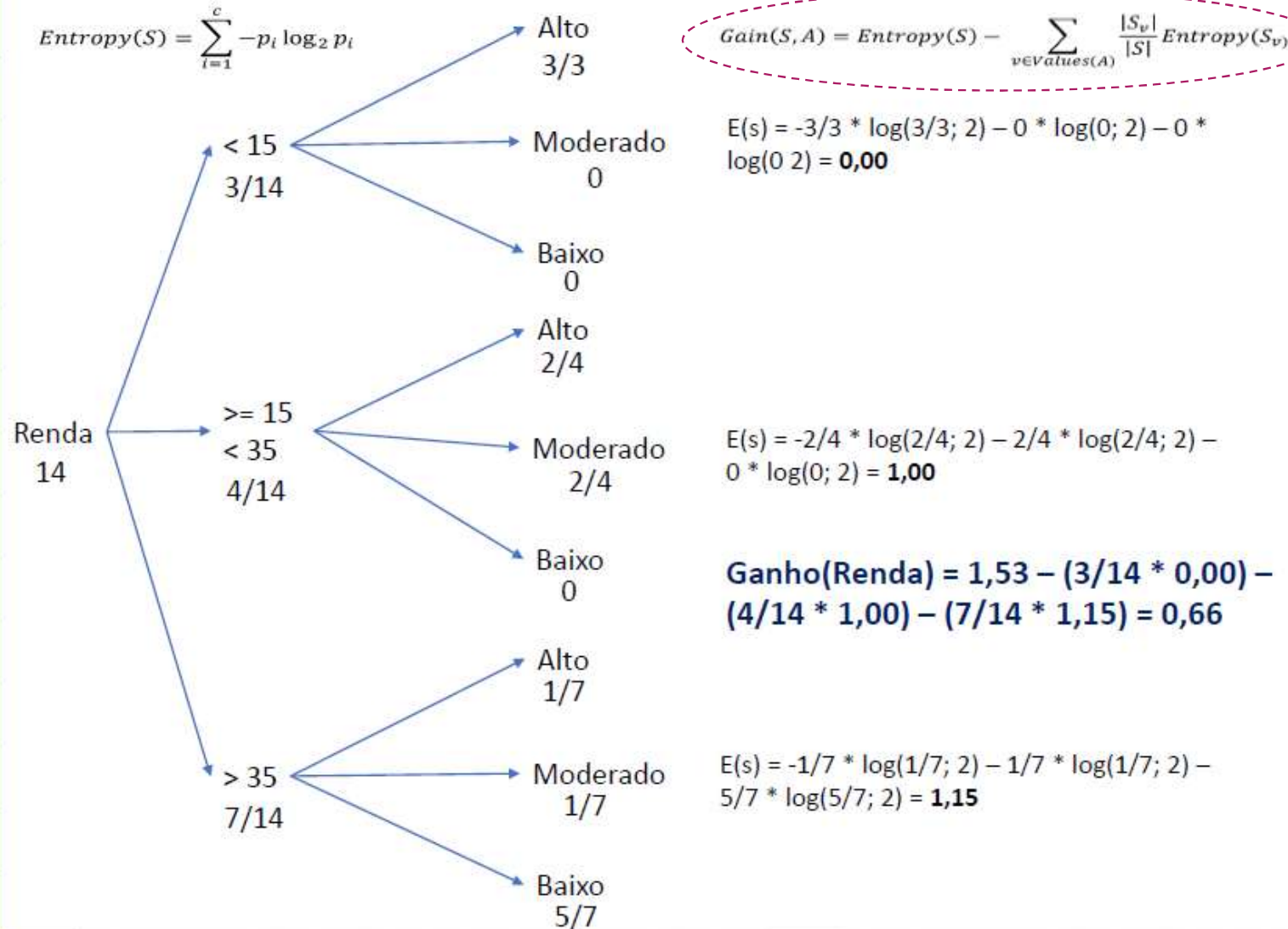
Cálculo do Ganho da Informação (Garantias)

Garantias	Risco
Nenhuma	Alto
Nenhuma	Alto
Nenhuma	Moderado
Nenhuma	Alto
Nenhuma	Baixo
Adequada	Baixo
Nenhuma	Alto
Adequada	Moderado
Nenhuma	Baixo
Adequada	Baixo
Nenhuma	Alto
Nenhuma	Moderado
Nenhuma	Baixo
Nenhuma	Alto



Cálculo do Ganho da Informação (Renda)

Renda anual	Risco
< 15.000	Alto
>= 15.000 a <= 35.000	Alto
>= 15.000 a <= 35.000	Moderado
> 35.000	Alto
> 35.000	Baixo
> 35.000	Baixo
< 15.000	Alto
> 35.000	Moderado
> 35.000	Baixo
> 35.000	Baixo
< 15.000	Alto
>= 15.000 a <= 35.000	Moderado
> 35.000	Baixo
>= 15.000 a <= 35.000	Alto



== Resultados ==
Ganho de Informação

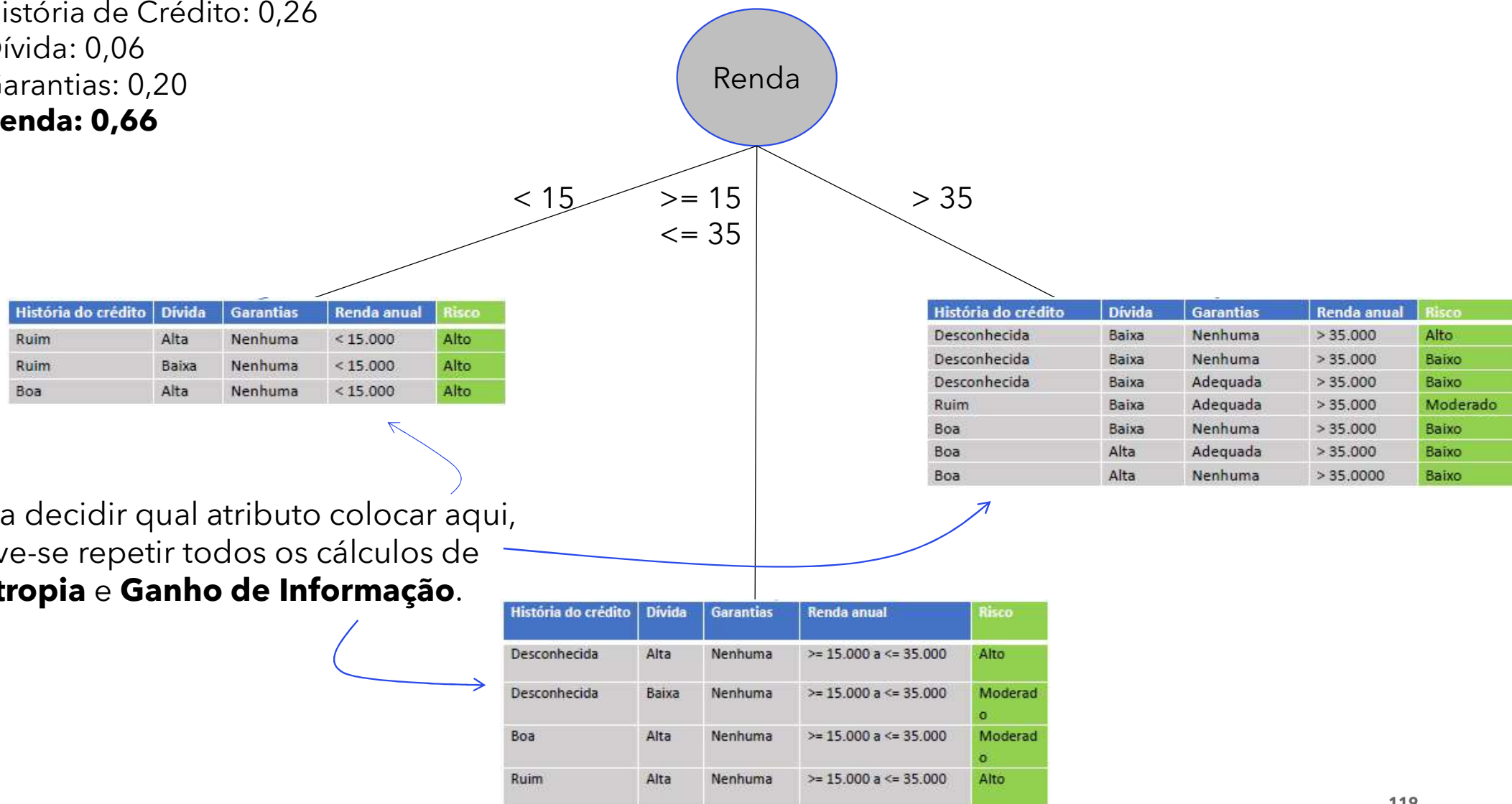
História de Crédito: 0,26

Dívida: 0,06

Garantias: 0,20

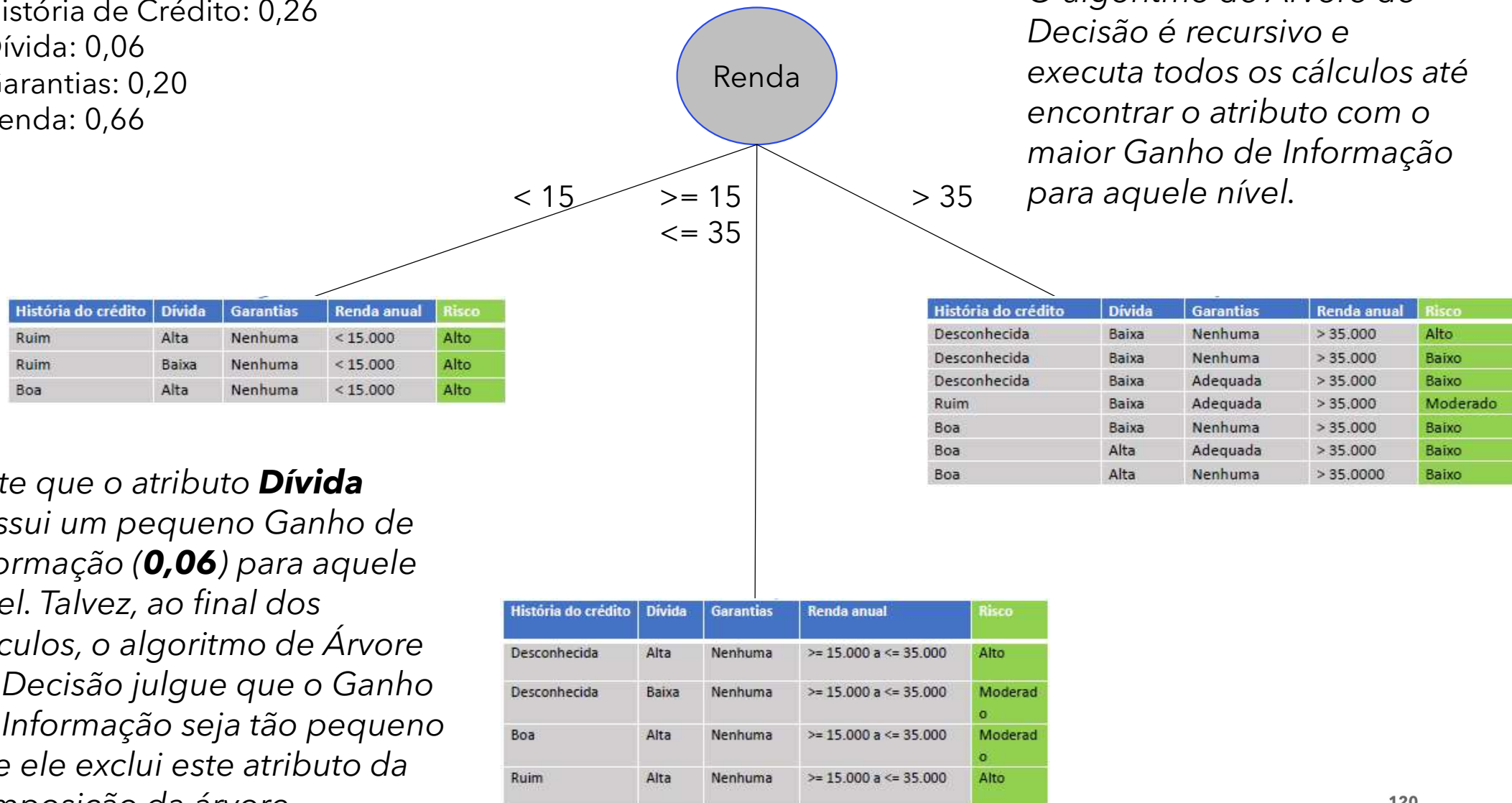
Renda: 0,66

História de Crédito: 0,26
 Dívida: 0,06
 Garantias: 0,20
Renda: 0,66



História de Crédito: 0,26
 Dívida: 0,06
 Garantias: 0,20
 Renda: 0,66

O algoritmo de Árvore de Decisão é recursivo e executa todos os cálculos até encontrar o atributo com o maior Ganho de Informação para aquele nível.

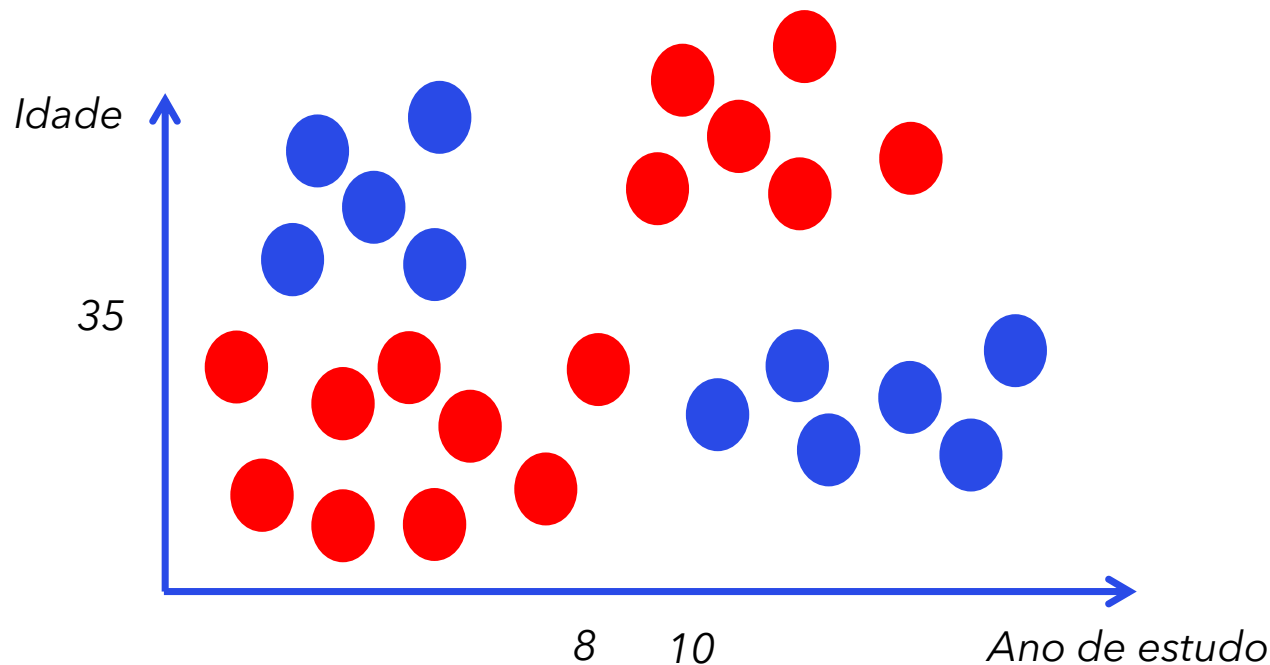


Note que o atributo **Dívida** possui um pequeno Ganho de Informação (**0,06**) para aquele nível. Talvez, ao final dos cálculos, o algoritmo de Árvore de Decisão julgue que o Ganho de Informação seja tão pequeno que ele exclui este atributo da composição da árvore.

Árvores de Decisão - mais conceitos

Árvores de Decisão - mais conceitos

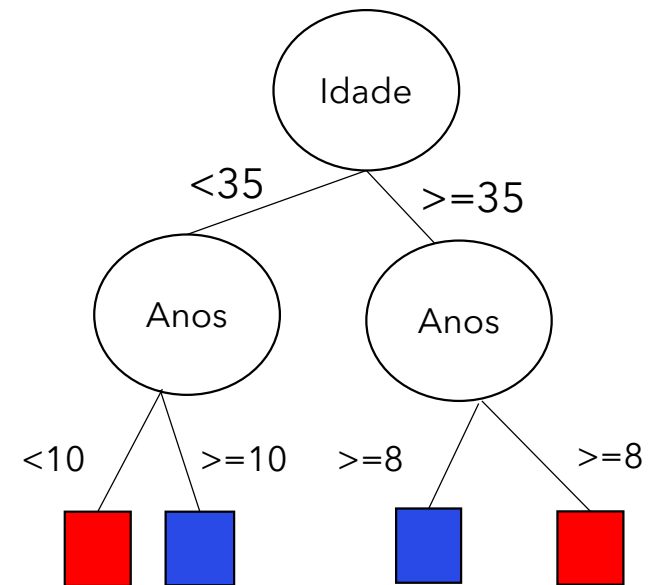
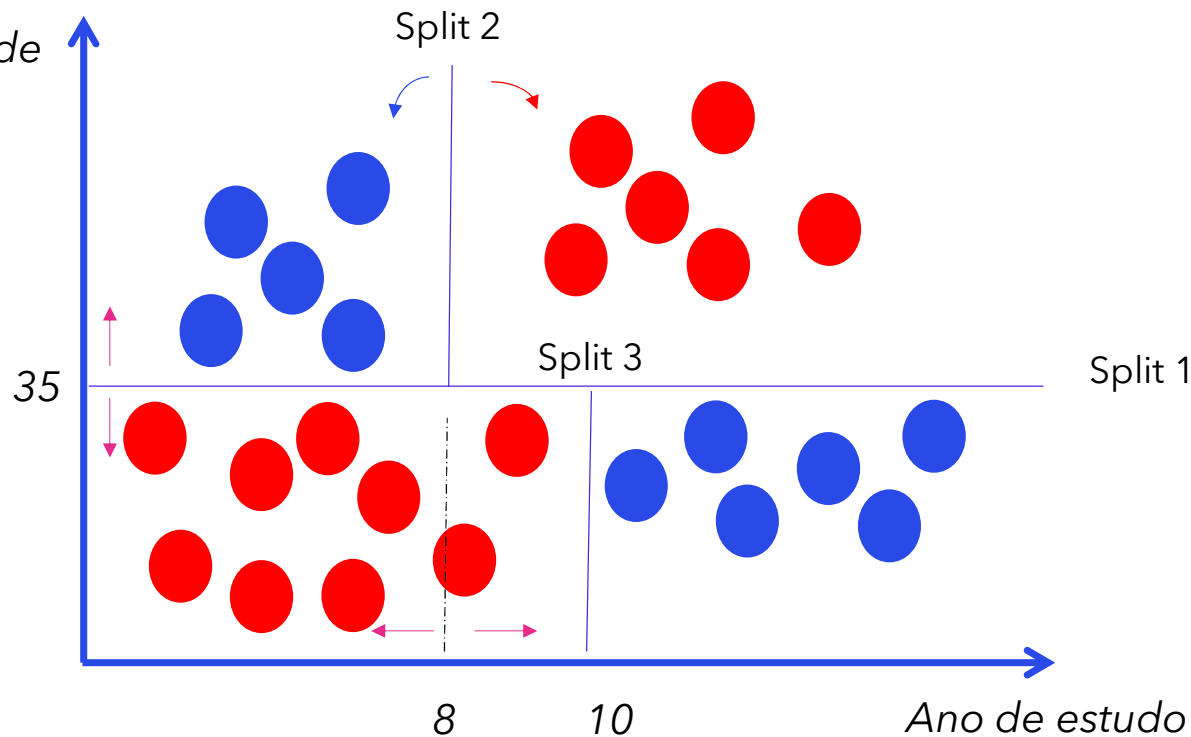
- Uma outra forma de estudar Árvore de Decisão:



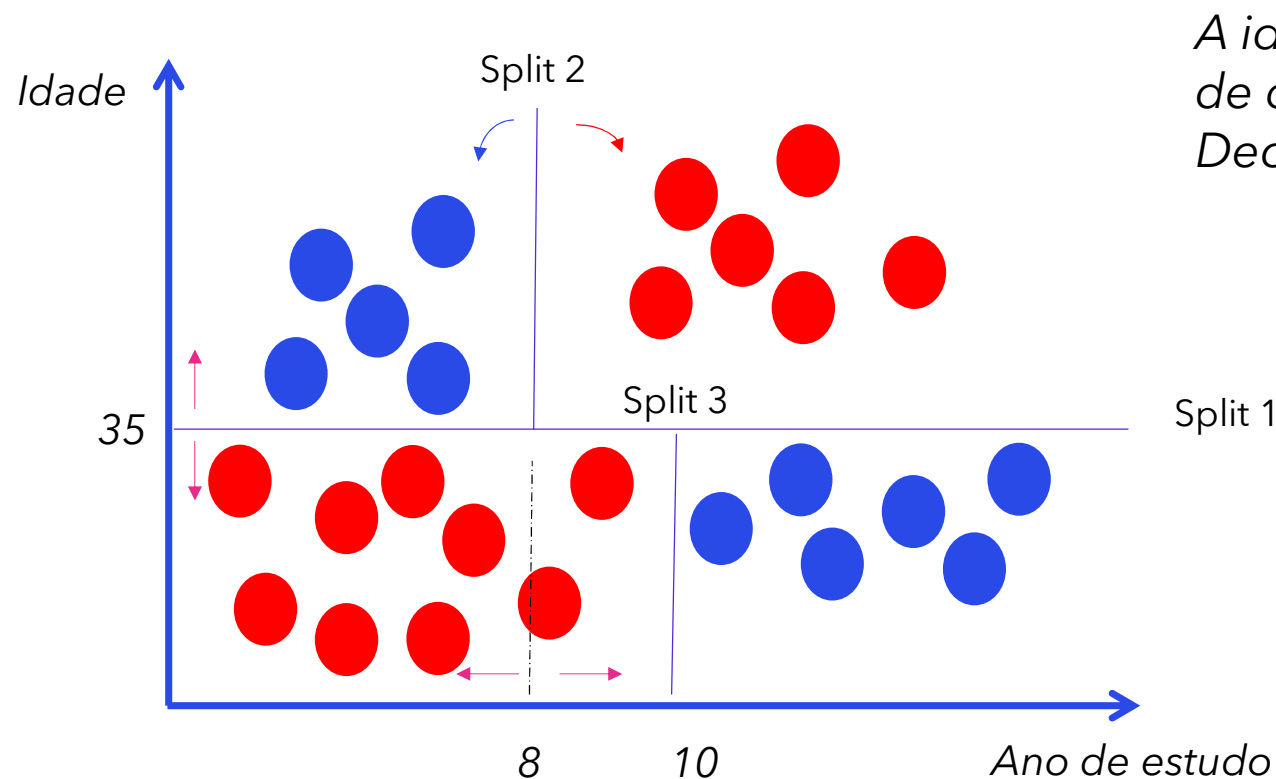
Desejamos classificar os objetos em **vermelho** e **azul** com base em dois atributos:

- *Idade*
- *Quantidade de anos de estudo*

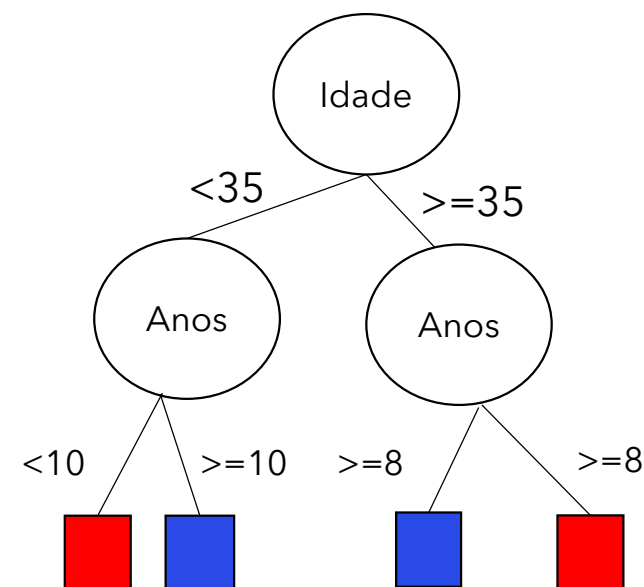
A partir do conjunto de dados é feito um **split**.
Cada **split** representa um nó da árvore.



Temos o nó raiz onde começa a árvore, os nós de decisão, essas decisões serão tomadas a partir de uma amostra de teste e por fim temos os nós folhas onde é atribuído a classe.



A ideia aqui é encontrar o melhor conjunto de divisores (splits) para criar a Árvore de Decisão.



Temos o nó raiz onde começa a árvore, os nós de decisão, essas decisões serão tomadas a partir de uma amostra de teste e por fim temos os nós folhas onde é atribuído a classe.

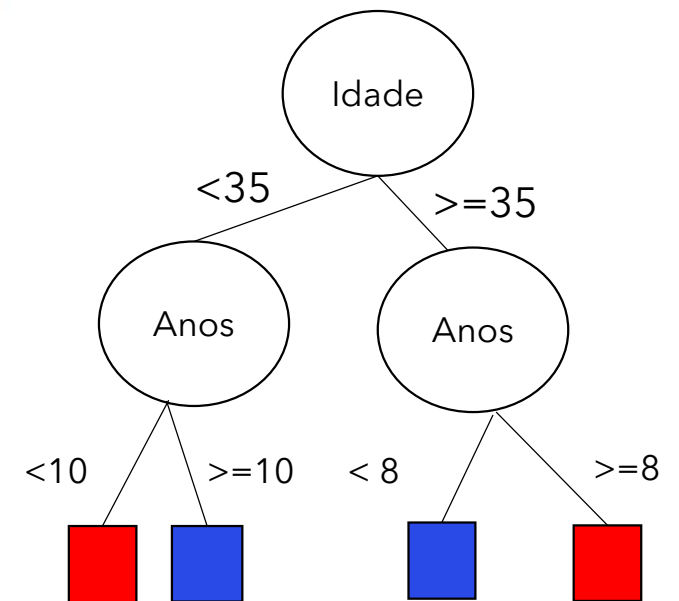
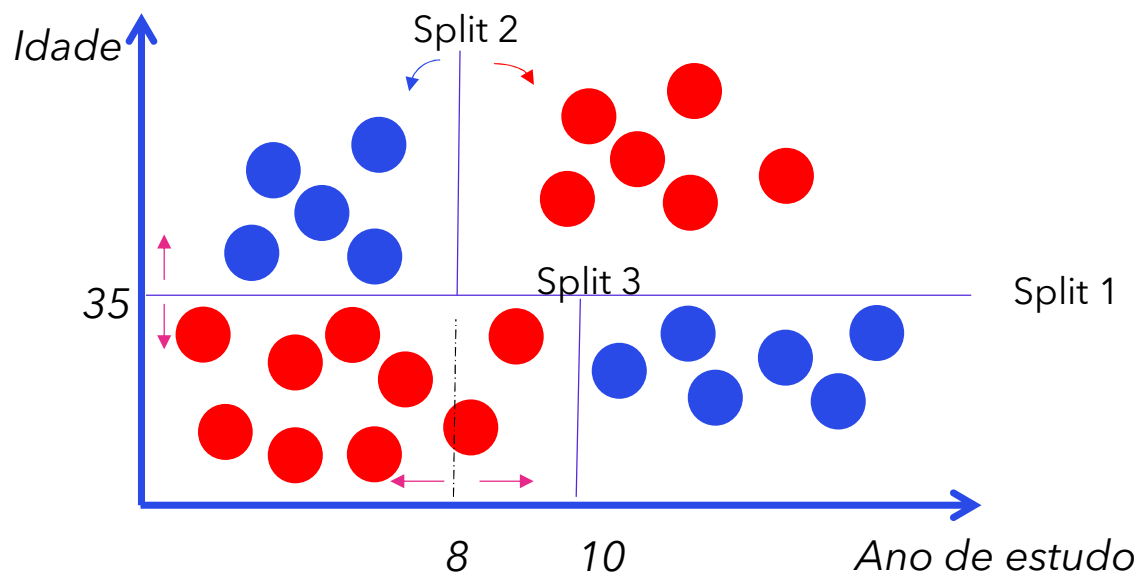
Ao final, os splits se transformam em regras para a criação da Árvore de Decisão:

SE Idade ≥ 35 E Anos ≥ 8 ENTÃO Vermelho

SE Idade ≥ 35 E Anos < 8 ENTÃO Azul

SE Idade < 35 E Anos ≥ 10 ENTÃO Azul

SE Idade < 35 E Anos < 10 ENTÃO Vermelho



Algoritmo Árvores de Decisão

- **Vantagens**

- Fácil interpretação, pois não requer conhecimento estatístico;
- Não precisa normalizar ou padronizar base de dados (aceita dados categóricos e numéricos);
- Rápido para classificar novos registros.

- **Desvantagens**

- A geração de árvores pode ser muito complexa;
- São instáveis, pois pequenas mudanças no dados podem mudar toda a relação da árvore;
- É propensa a sofrer *overfitting*, ou seja, ela se ajusta muito aos dados de treino e não apresenta uma performance muito boa com os dados de teste.

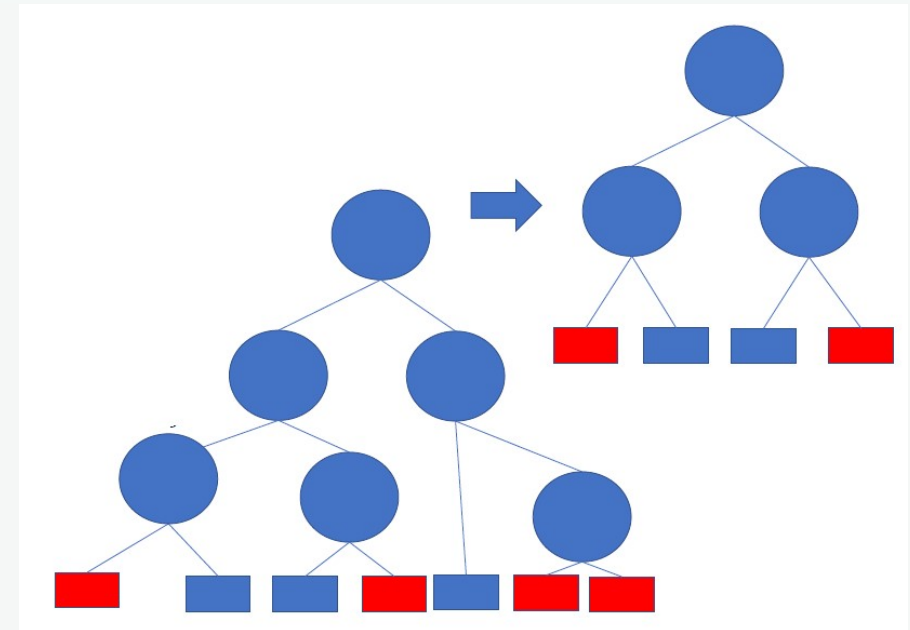
"Poda" de Árvores de Decisão

- Após a construção da Árvore de Decisão, a *poda* pode ser realizada para melhorar sua capacidade de generalização;
- Uma árvore maior é induzida de forma a superajustar os exemplos e então ela é podada até obter uma árvore menor (mais simples);
- A poda evita o *overfitting*.



"Poda" de Árvores de Decisão

- Podem ocorrer situações em que o algoritmo depois de treinado consiga fazer boas previsões no conjunto de treino, mas quando aplicado ao conjunto de teste, apresenta muitos erros.
- Ou o contrário: ter um desempenho ruim no conjunto de treino e acertar de maneira satisfatória no conjunto de teste.
- Bias (viés)
 - Erro por classificação incorreta
- Variância
 - Pode levar a *overfitting*.



Árvore de Decisão em R

#Leitura da base de dados no R Studio

```
base = read.csv('credit_data.csv')
```

#Apagar o atributo clientid

```
base$clientid = NULL
```

#Valores inconsistentes

```
base$age = ifelse(base$age < 0, 40.92, base$age)
```

#Valores faltantes

```
base$age = ifelse(is.na(base$age), mean(base$age, na.rm=TRUE), base$age)
```

#Escalonamento

```
base[, 1:3] = scale(base[, 1:3])
```

#Encode da classe (default)

#O atributo classe está com 0 e 1, mas precisamos transformá-lo num fator

```
base$default = factor(base$default, levels = c(0,1))
```

#Divisão entre - Base de Treinamento e Teste

```
#install.packages('caTools')  
library(caTools)  
set.seed(1)  
divisao = sample.split(base$default, SplitRatio = 0.75)  
base_treinamento = subset(base, divisao == TRUE)  
base_teste = subset(base, divisao == FALSE)
```

#rpart cria uma Árvore de Decisão

```
#install.packages('rpart')  
library(rpart)  
classificador = rpart(formula = default ~., data = base_treinamento)  
print(classificador)
```

#Para visualizarmos a Arvore de Decisão, é preciso instalar outro pacote:

```
install.packages('rpart.plot')  
library(rpart.plot)
```

#Para gerar a Árvore de Decisão (próximo slide):

```
rpart.plot(classificador)
```

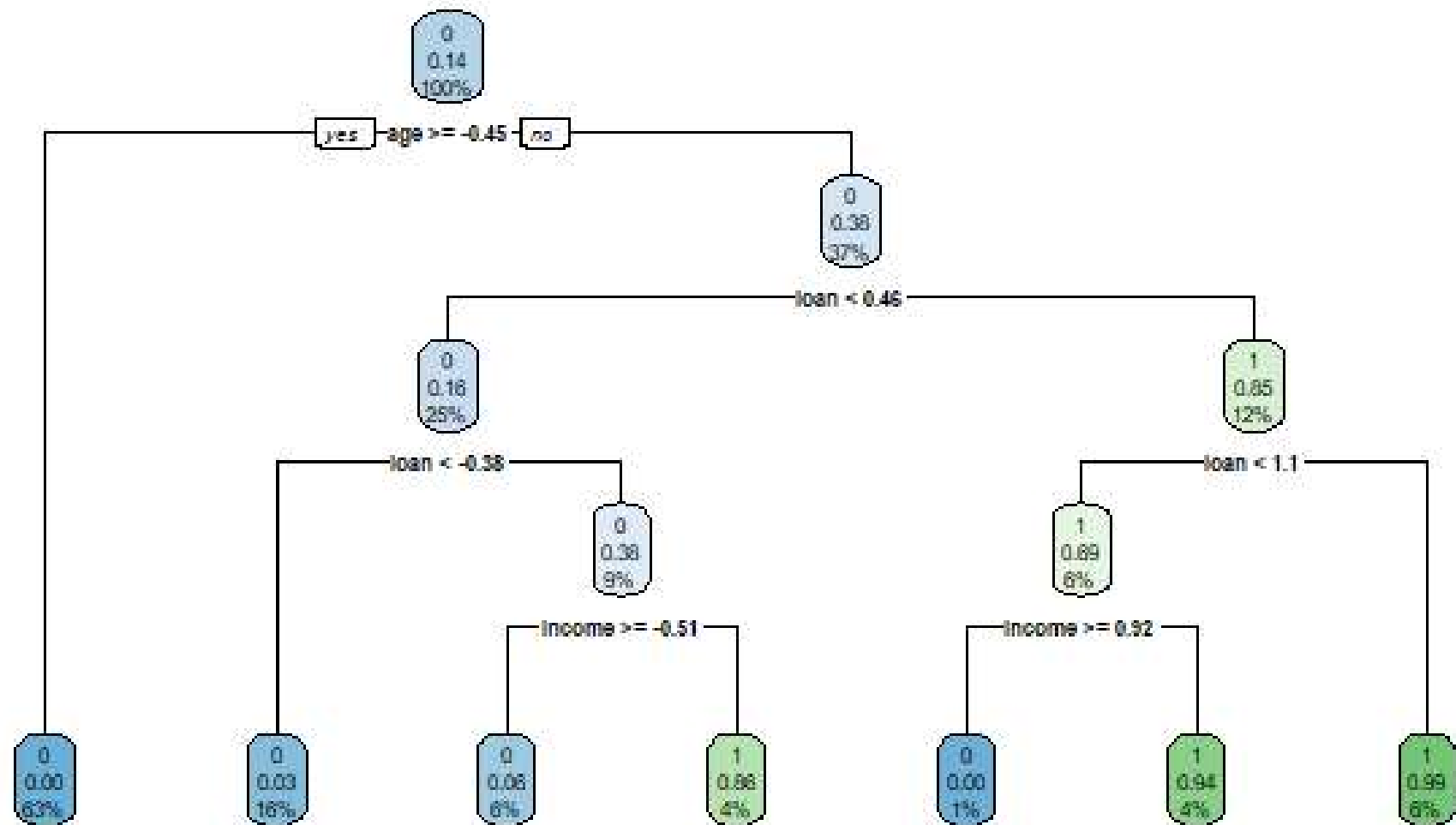
#Execução dos dados na Base de Teste

```
previsoes = predict(classificador, newdata = base_teste[-4], type = 'class')
```

#A Matriz Confusão compara os resultados executados na Base Teste com os dados gerados em previsoes

```
matriz_confusao = table(base_teste[, 4], previsoes)  
print(matriz_confusao)
```

```
library(caret)  
confusionMatrix(matriz_confusao)
```



#Ao digitar:

```
print(matriz_confusao)
```

#O R mostrará:

		Previsões	
		0	1
0		423	6
1		9	62

O algoritmo acertou:

423 0 classificado como **0** (não paga a dívida)

62 1 classificado como **1** (paga a dívida)

Dos 500 registros da Base de Teste, o algoritmo errou a previsão de: $9 + 6 = 15$ erros

#Após a execução da matriz de confusão, temos que o % de acerto foi de 97%:

```
library(caret)
```

```
confusionMatrix(matriz_confusao)
```

Accuracy : 0.97

Exercícios

- Execute o mesmo código R:
 - Apenas com os "*Valores Faltantes*" e "*Valores Inconsistentes*", **sem o Escalonamento** (verifique o % de acerto - "accuracy")
 - Apenas com :
 - Leitura
 - Apagar clientid
 - Encode e o restante do código R
 - **Sem o "pré-processamento"**
 - Verifique o % de acerto - "accuracy"

Árvores de Decisão - *Random Forest*

Random Forest

- O que é:
 - Floresta Randômica (*Random Forest*) é um algoritmo de aprendizagem de máquina flexível e fácil de usar que produz excelentes resultados;
 - É também um dos algoritmos mais utilizados, devido à sua simplicidade e o fato de que pode ser utilizado para tarefas de classificação.
- Como funciona:
 - Floresta Randômica é um algoritmo de aprendizagem supervisionada. Ele cria uma floresta de modo aleatório:
 - A “floresta” criada é uma combinação de árvores de decisão (*ensemble learning*).

Random Forest

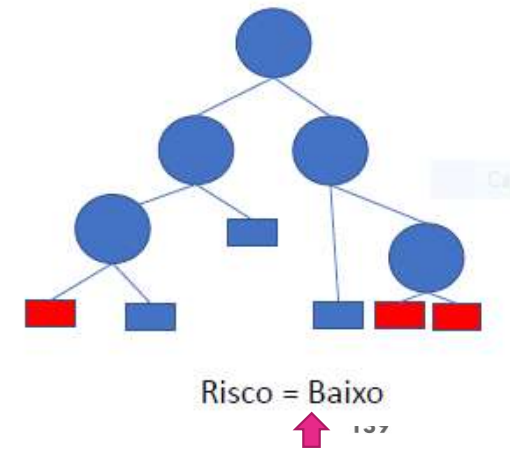
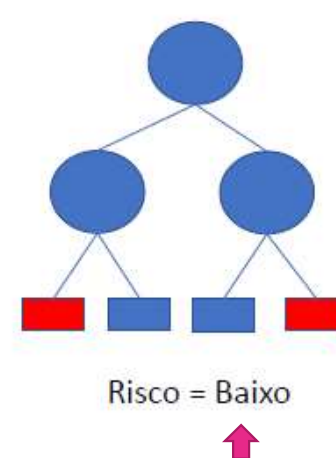
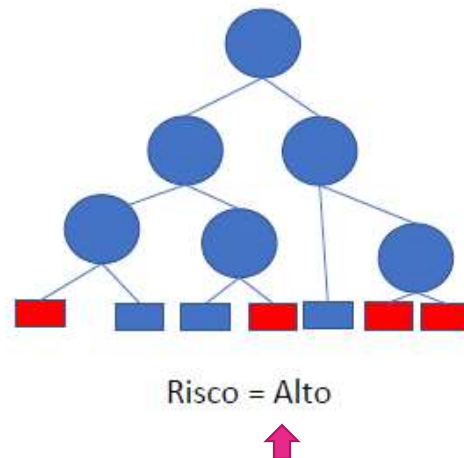
- *Ensemble learning* (aprendizagem em conjunto);
- “Consultar diversos profissionais para tomar uma decisão”;
- Vários algoritmos juntos para construir um algoritmo mais ‘forte’;
- Ele utiliza a média (regressão) ou votos da maioria (classificação) para dar a resposta final.

Base original

História do crédito	Dívida	Garantias	Renda anual	Risco
Ruim	Alta	Nenhuma	< 15.000	Alto
Desconhecida	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto
Desconhecida	Baixa	Nenhuma	>= 15.000 a <= 35.000	Moderado
Desconhecida	Baixa	Nenhuma	> 35.000	Alto
Desconhecida	Baixa	Nenhuma	> 35.000	Baixo
Desconhecida	Baixa	Adequada	> 35.000	Baixo
Ruim	Baixa	Nenhuma	< 15.000	Alto
Ruim	Baixa	Adequada	> 35.000	Moderado
Boa	Baixa	Nenhuma	> 35.000	Baixo
Boa	Alta	Adequada	> 35.000	Baixo
Boa	Alta	Nenhuma	< 15.000	Alto
Boa	Alta	Nenhuma	>= 15.000 a <= 35.000	Moderado
Ruim	Alta	Nenhuma	> 35.000	Baixo
Ruim	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto

Por exemplo:

- 3 árvores na Floresta tomaram a seguinte decisão:
 - Risco Alto
 - Risco Baixo
 - Risco Baixo
- A tomada de decisão, neste caso, é que o **risco é baixo**.



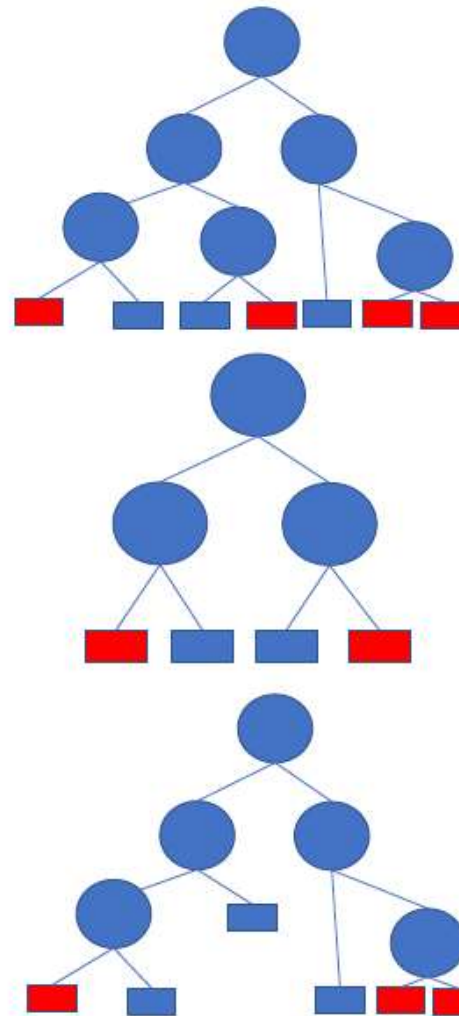
Random Forest

Base original ✓ ✓ ✓

História do crédito	Dívida	Garantias	Renda anual	Risco
Ruim	Alta	Nenhuma	< 15.000	Alto
Desconhecida	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto
Desconhecida	Baixa	Nenhuma	>= 15.000 a <= 35.000	Moderado
Desconhecida	Baixa	Nenhuma	> 35.000	Alto
Desconhecida	Baixa	Nenhuma	> 35.000	Baixo
Desconhecida	Baixa	Adequada	> 35.000	Baixo
Ruim	Baixa	Nenhuma	< 15.000	Alto
Ruim	Baixa	Adequada	> 35.000	Moderado
Boa	Baixa	Nenhuma	> 35.000	Baixo
Boa	Alta	Adequada	> 35.000	Baixo
Boa	Alta	Nenhuma	< 15.000	Alto
Boa	Alta	Nenhuma	>= 15.000 a <= 35.000	Moderado
Boa	Alta	Nenhuma	> 35.000	Baixo
Ruim	Alta	Nenhuma	>= 15.000 a <= 35.000	Alto

O algoritmo escolhe de forma aleatória K atributos para comparação da métrica de pureza/impureza (impureza de Gini/entropia).

K = 3
Árvores = 3



História de crédito
Dívida
Garantias

Para cada árvore, o algoritmo escolheu 3 atributos.

Renda
Dívida
Garantias

Renda
História de crédito
Dívida

Árvore de Decisão em R - *Random Forest*

#Leitura da base de dados no R Studio

```
base = read.csv('credit_data.csv')
```

#Apagar o atributo clientid

```
base$clientid = NULL
```

#Valores inconsistentes

```
base$age = ifelse(base$age < 0, 40.92, base$age)
```

#Valores faltantes

```
base$age = ifelse(is.na(base$age), mean(base$age, na.rm=TRUE), base$age)
```

#Escalonamento

```
base[, 1:3] = scale(base[, 1:3])
```

#Encode da classe (default)

#O atributo classe está com 0 e 1, mas precisamos transformá-lo num fator

```
base$default = factor(base$default, levels = c(0,1))
```

#Divisão entre - Base de Treinamento e Teste

```
#install.packages('caTools')  
library(caTools)  
set.seed(1)  
divisao = sample.split(base$default, SplitRatio = 0.75)  
base_treinamento = subset(base, divisao == TRUE)  
base_teste = subset(base, divisao == FALSE)
```

#Para trabalhar com o Random Forest é preciso instalar:

```
install.packages('randomForest')  
library(randomForest)
```

#Gerar "floresta" com 10 árvores randômicas e treinar o algoritmo

```
set.seed(1)  
classificador = randomForest(x = base_treinamento[-4], y = base_treinamento$default, ntree = 10)
```

#Gerar previsões de acerto comparando a variável *classificador* com a *base_teste*

```
previsoes = predict(classificador, newdata = base_teste[-4])
```

#A Matriz Confusão compara os resultados executados na Base Teste com os dados gerados em *previsoes*

```
matriz_confusao = table(base_teste[, 4], previsoes)
print(matriz_confusao)
```

Previsões		
	0	1
0	424	3
1	7	66

O algoritmo acertou:

424 0 classificado como **0** (não paga a dívida)

66 1 classificado como **1** (paga a dívida)

Dos 500 registros da Base de Teste, o algoritmo errou a previsão de: $7 + 3 = 10$ erros

#A acurácia do algoritmo de Árvore de Decisão - *Random Forest*:

```
library(caret)
confusionMatrix(matriz_confusao)
```

Accuracy : 0.98

Exercícios

1. Altere o programa para gerar: 15, 20 e 40 árvores na Floresta Randômica e anote os 3 valores da acurácia do algoritmo. Houve diferença? Você acha que pode ter ocorrido o “fenômeno de *overfitting*”?
2. Execute o mesmo código R em dois momentos e anote os valores da acurácia:
 - a) Apenas com os “*Valores Faltantes*” e “*Valores Inconsistentes*”, **sem o Escalonamento** (verifique o % de acerto - “*accuracy*”)
 - b) Apenas com os “*Valores faltantes*” e, em seguida, execute do “Encode” para baixo. O método *randomForest* não aceita executar com dados NA (*dados faltantes*).

F I M

