

FLORESTA RANDÔMICA


MACHINE LEARNING

Luiz Paulo T. Juvencio
Vitor Hugo de Souza
Vladson Ramos dos Santos
Gabriel Nunes Marques

Prof. Yuri Crotti



O que iremos ver:

1. História da Floresta Randômica
 2. O que é uma floresta randômica?
 3. O que é uma árvore de decisão?
 4. Termos importantes na Floresta Randômica
 5. Trabalhando com Floresta Randômica
 6. Vantagens da Floresta Randômica
 7. Desvantagens da Floresta Randômica
 8. Aplicação Vídeo demonstrativo
 9. Hands-On Floresta Randômica
- 

An abstract pattern of light blue lines and dots on a dark blue background, resembling a circuit board or a network diagram. The lines are of varying thickness and form a complex, interconnected web. Some lines end in small dots, while others are open. The overall effect is a sense of digital connectivity and flow.

01

História da Floresta Randômica

O termo “floresta de decisão aleatória” foi proposto pela primeira vez em 1995 por Tin Kam Ho. Ho desenvolveu uma fórmula para usar dados aleatórios para criar previsões.



Então, em 2006, Leo Breiman e Adele Cutler estenderam o algoritmo e criaram florestas aleatórias como as conhecemos hoje. Isso significa que essa tecnologia, e a matemática e a ciência por trás dela, ainda são relativamente novas.



É chamado de “floresta” porque desenvolve uma floresta de árvores de decisão. Os dados dessas árvores são mesclados para garantir as previsões mais precisas. Enquanto uma árvore de decisão individual tem um resultado e uma gama estreita de grupos, a floresta garante um resultado mais preciso, com um número maior de grupos e decisões.

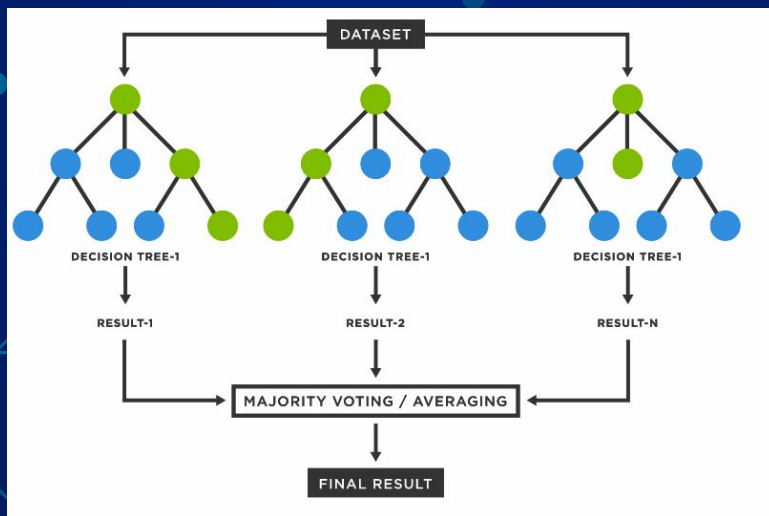
Ela tem o benefício adicional de trazer aleatoriedade ao modelo, encontrando o melhor recurso em um subconjunto aleatório de recursos. No geral, esses benefícios criam um modelo com ampla diversidade que muitos cientistas de dados favorecem.



02

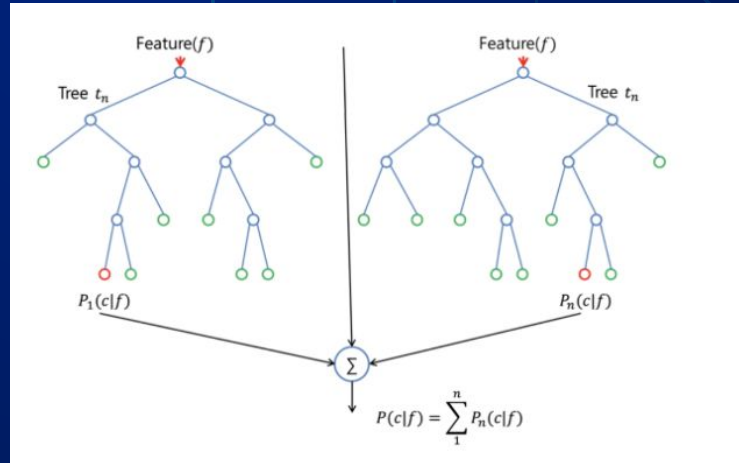
O que é uma Floresta Randômica?

Floresta aleatória (random forest) é um algoritmo de aprendizado de máquina supervisionado. É um dos algoritmos mais utilizados devido à sua precisão, simplicidade e flexibilidade. O fato de poder ser usado para tarefas de classificação e regressão, combinado com sua natureza não linear, torna-o altamente adaptável a uma variedade de dados e situações.



Uma floresta aleatória escolherá aleatoriamente os recursos e fará observações, construirá uma floresta de árvores de decisão e, em seguida, calculará a média dos resultados.

A teoria é que um grande número de árvores não correlacionadas criará previsões mais precisas do que uma árvore de decisão individual. Isso ocorre porque o volume de árvores trabalha em conjunto para proteger cada uma de erros individuais e overfitting.



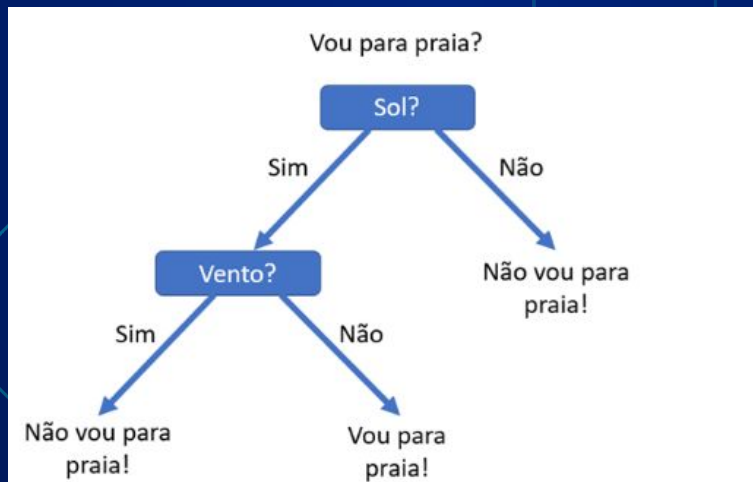
An abstract pattern of light blue lines resembling a circuit board or data paths, set against a dark blue background. The lines are of varying thickness and form a complex, interconnected network.

03

O que é uma Árvore de Decisão?

Como o próprio nome sugere, neste algoritmo vários pontos de decisão serão criados. Estes pontos são os “*nós*” da árvore e em cada um deles o resultado da decisão será seguir por um caminho, ou por outro. Os caminhos existentes são os “*ramos*”.

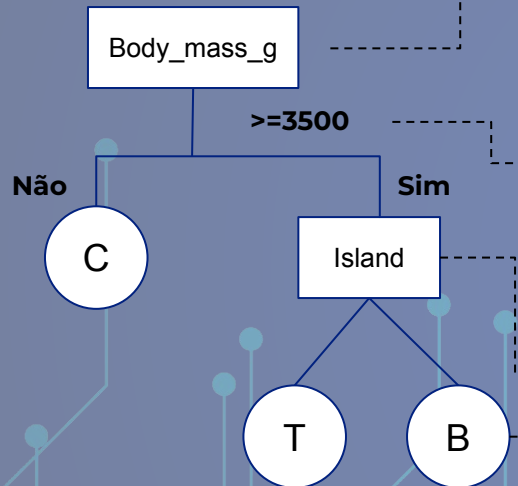
Esta é a estrutura básica de uma árvore de decisão. Os nós são responsáveis pelas conferências que irão indicar um ramo ou outro para sequência do fluxo.



An abstract pattern of light blue lines and dots on a dark blue background, resembling a circuit board or digital network, located on the left side of the slide.

04

Termos importantes na Floresta Randômica



Root Node

O conjunto de dados para o treinamento é alimentado aqui.



Splitting

Métodos de Gini e Entropy para decidir a divisão ideal



Decision-nodes

Fornece o link para os nós folha



Lead-Node

Ponto final onde nenhuma outra divisão ocorre.

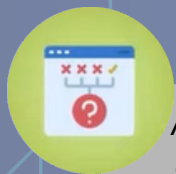
An abstract pattern of light blue lines and dots on a dark blue background, resembling a circuit board or digital network, located on the left side of the slide.

05

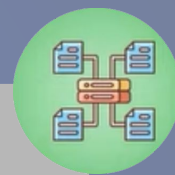
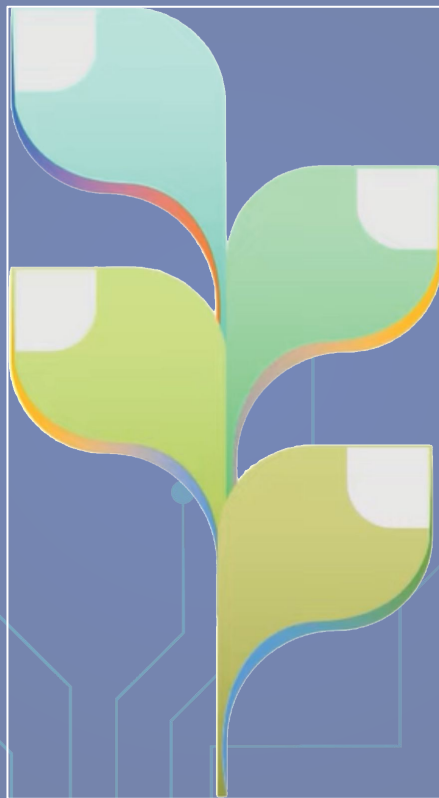
Trabalhando com Floresta Randômica



Como a seleção de recursos é feita em problemas de classificação e regressão



A melhor divisão é escolhida com base nos métodos Gini Impureza ou Ganho de Informação



Amostragem aleatória com substituição



Técnica de conjunto
agregação/
ensacamento

Amostragem aleatória com substituição

Espécies	Ilha	Massa_Corporal_gr	Cumpr_Nadadeira_mm
Adelie	Torgersen	3750.0	181.0
Adelie	Torgersen	3800.0	186.0
Chinstrap	Dream	3250.0	187.0
Chinstrap	Dream	3675.0	198.0
Gentoo	Biscoe	6000.0	220.0
Gentoo	Biscoe	4750.0	215.0

Espécies	Ilha	Massa_Corporal_gr
Adelie	Torgersen	3750.0
Chinstrap	Dream	3250.0
Gentoo	Biscoe	6000.0
Gentoo	Biscoe	4750.0

Espécies	Ilha	Cumpr_Nadadeira_mm
Adelie	Torgersen	186.0
Chinstrap	Dream	187.0
Gentoo	Biscoe	215.0
Gentoo	Biscoe	215.0

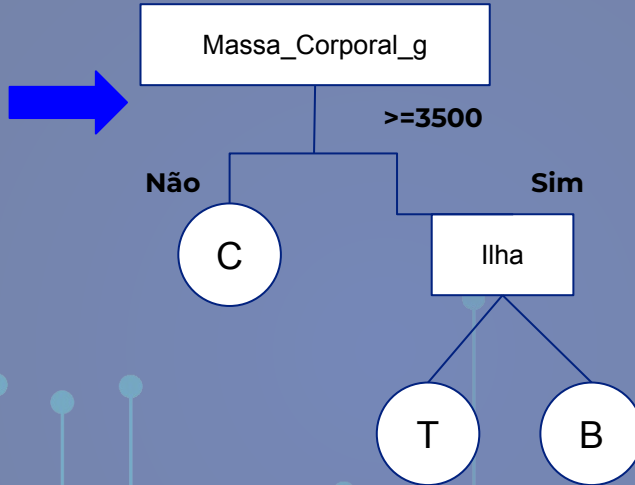
Espécies	Massa_Corporal_gr	Cumpr_Nadadeira_mm
Adelie	3750.0	181.0
Adelie	3800.0	186.0
Chinstrap	3675.0	198.0
Gentoo	6000.0	220.0

Trabalhando com Floresta Randômica

Espécies	Ilha	Massa_Corporal_gr
Adelie	Torgersen	3750.0
Chinstrap	Dream	3250.0
Gentoo	Biscoe	6000.0
Gentoo	Biscoe	4750.0

Espécies	Ilha	Cumpr_Nadadeira_mm
Adelie	Torgersen	186.0
Chinstrap	Dream	187.0
Gentoo	Biscoe	215.0
Gentoo	Biscoe	215.0

Espécies	Massa_Corporal_gr	Cumpr_Nadadeira_mm
Adelie	3750.0	181.0
Adelie	3800.0	186.0
Chinstrap	3675.0	198.0
Gentoo	6000.0	220.0

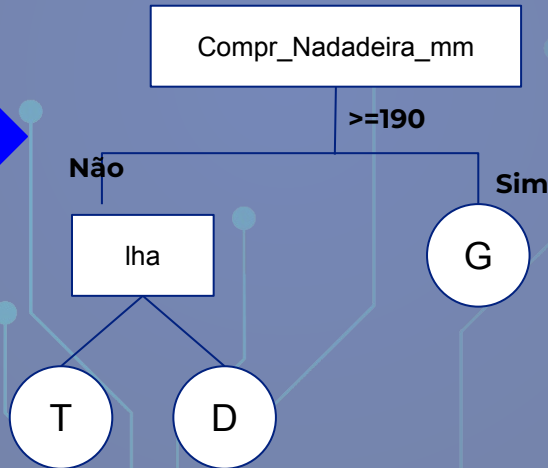


Trabalhando com Floresta Randômica

Espécies	Ilha	Massa_Corporal_gr
Adelie	Torgersen	3750.0
Chinstrap	Dream	3250.0
Gentoo	Biscoe	6000.0
Gentoo	Biscoe	4750.0

Espécies	Ilha	Cumpr_Nadadeira_mm
Adelie	Torgersen	186.0
Chinstrap	Dream	187.0
Gentoo	Biscoe	215.0
Gentoo	Biscoe	215.0

Espécies	Massa_Corporal_gr	Cumpr_Nadadeira_mm
Adelie	3750.0	181.0
Adelie	3800.0	186.0
Chinstrap	3675.0	198.0
Gentoo	6000.0	220.0

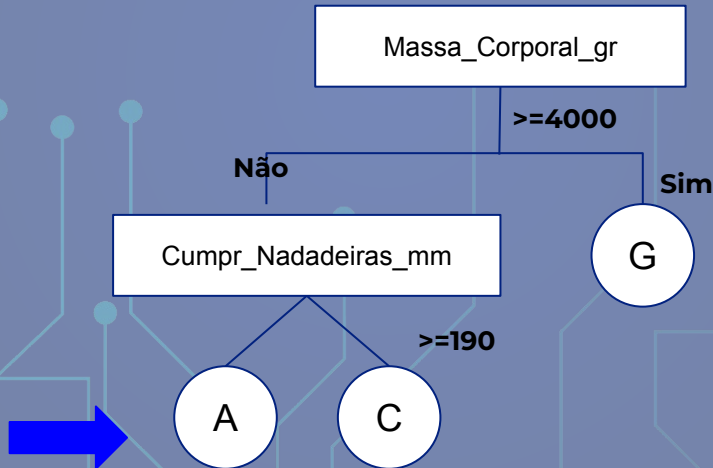


Trabalhando com Floresta Randômica

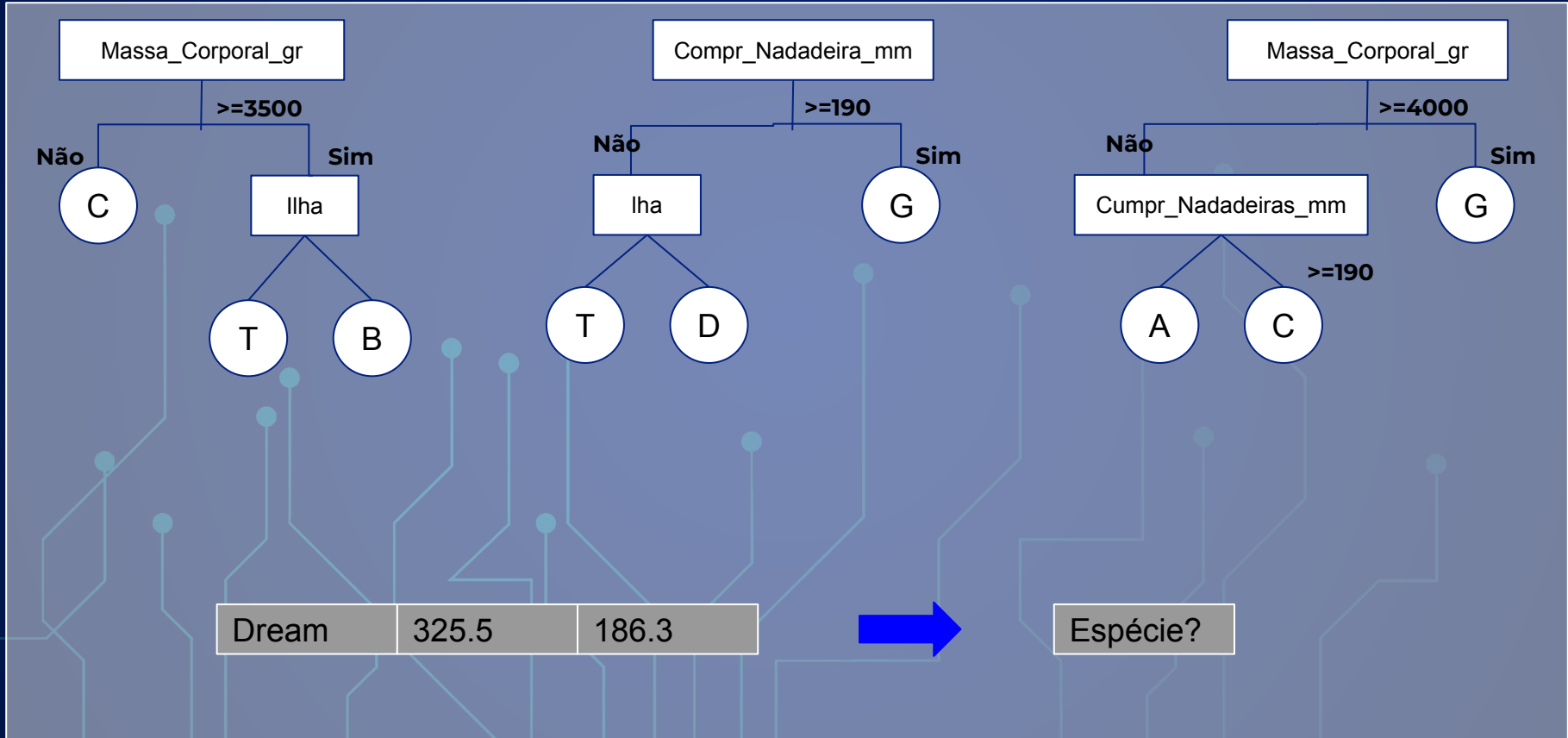
Espécies	Ilha	Massa_Corporal_gr
Adelie	Torgersen	3750.0
Chinstrap	Dream	3250.0
Gentoo	Biscoe	6000.0
Gentoo	Biscoe	4750.0

Espécies	Ilha	Cumpr_Nadadeira_mm
Adelie	Torgersen	186.0
Chinstrap	Dream	187.0
Gentoo	Biscoe	215.0
Gentoo	Biscoe	215.0

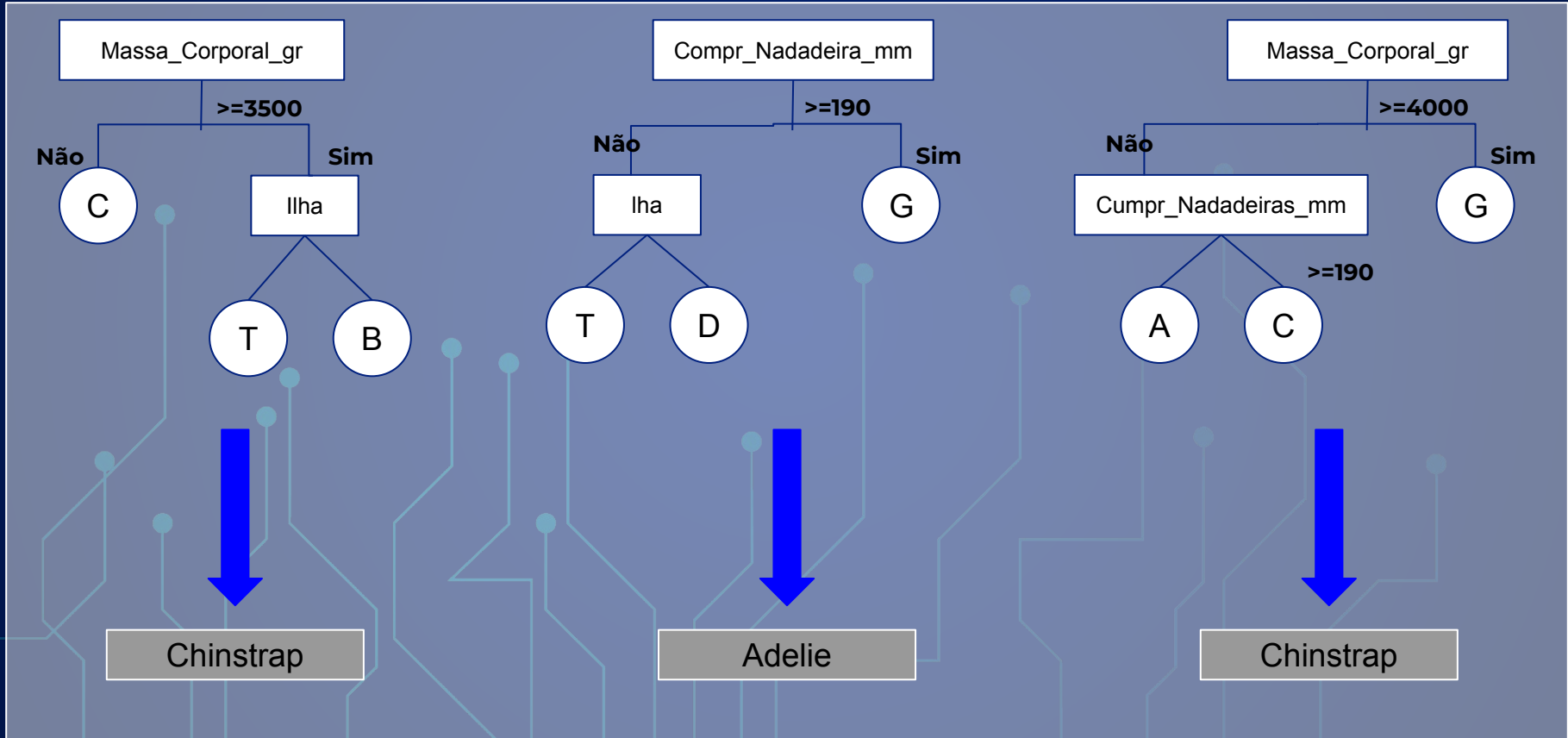
Espécies	Massa_Corporal_gr	Cumpr_Nadadeira_mm
Adelie	3750.0	181.0
Adelie	3800.0	186.0
Chinstrap	3675.0	198.0
Gentoo	6000.0	220.0



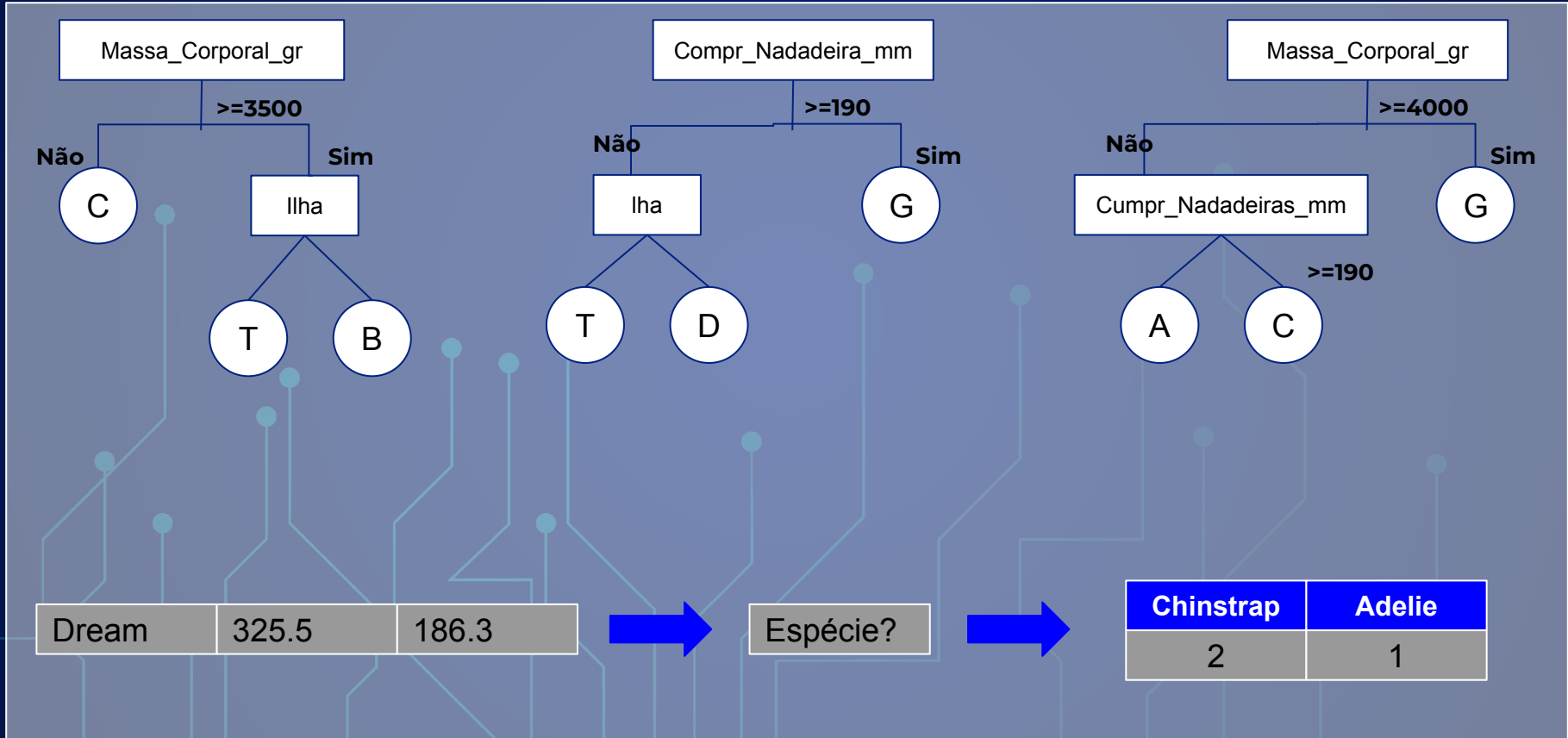
Técnica de conjunto: Agregação de Bootstrap



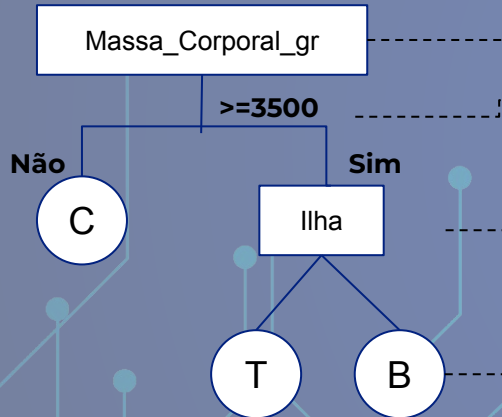
Técnica de conjunto: Agregação de Bootstrap



Técnica de conjunto: Agregação de Bootstrap



Métodos de divisão



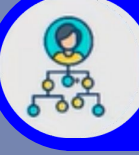
Impureza Gini

Para prever a probabilidade de um exemplo selecionado aleatoriamente ser classificado incorretamente.



Impureza Gini

o grau de impureza Gini varia de 0 a 1



Ganho de Informação

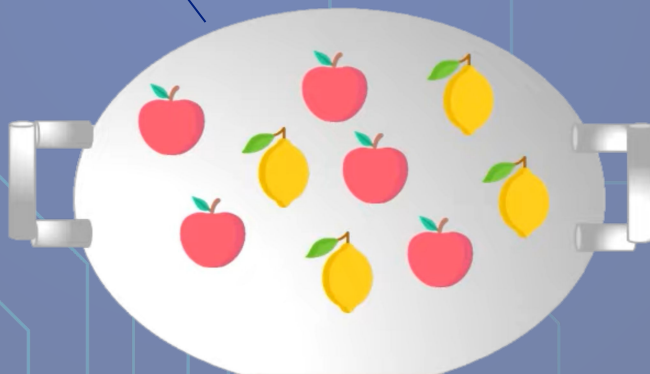
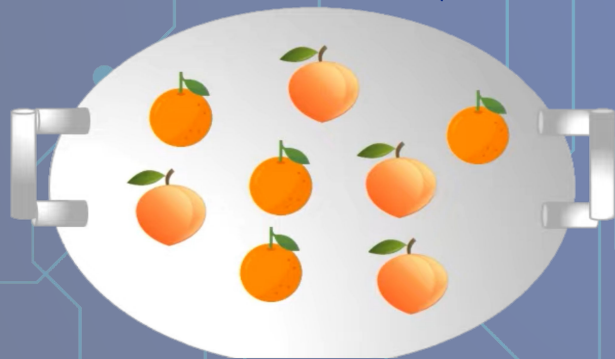
É selecionado o recurso que fornece mais informações sobre uma classe. Utiliza entropia.



Entropia

Medida de aleatoriedade e incerteza nos dados.

Métodos de divisão: Entropia

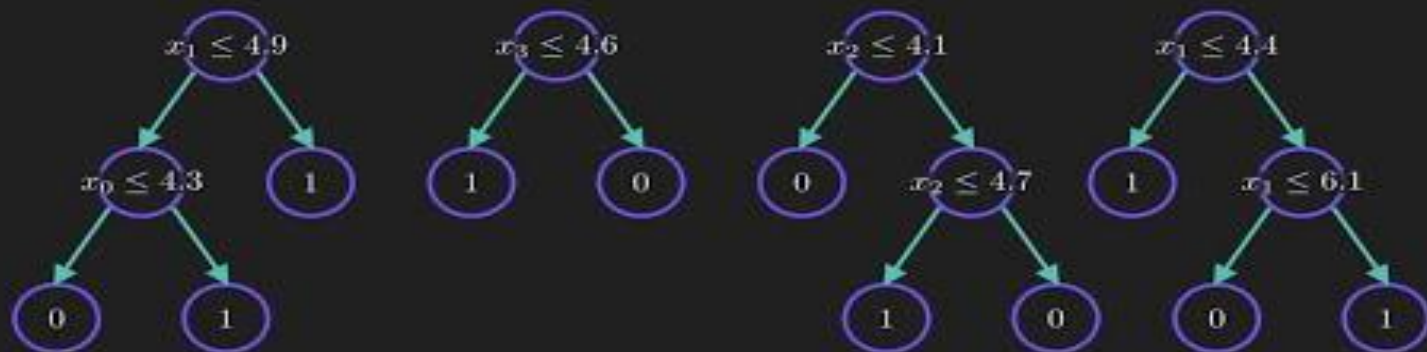


Alta Entropia



Baixa Entropia

Random Forest



An abstract pattern of light blue lines resembling a circuit board or digital network, with some lines ending in small glowing dots, set against a dark blue background.

06

Vantagens da Floresta Randômica

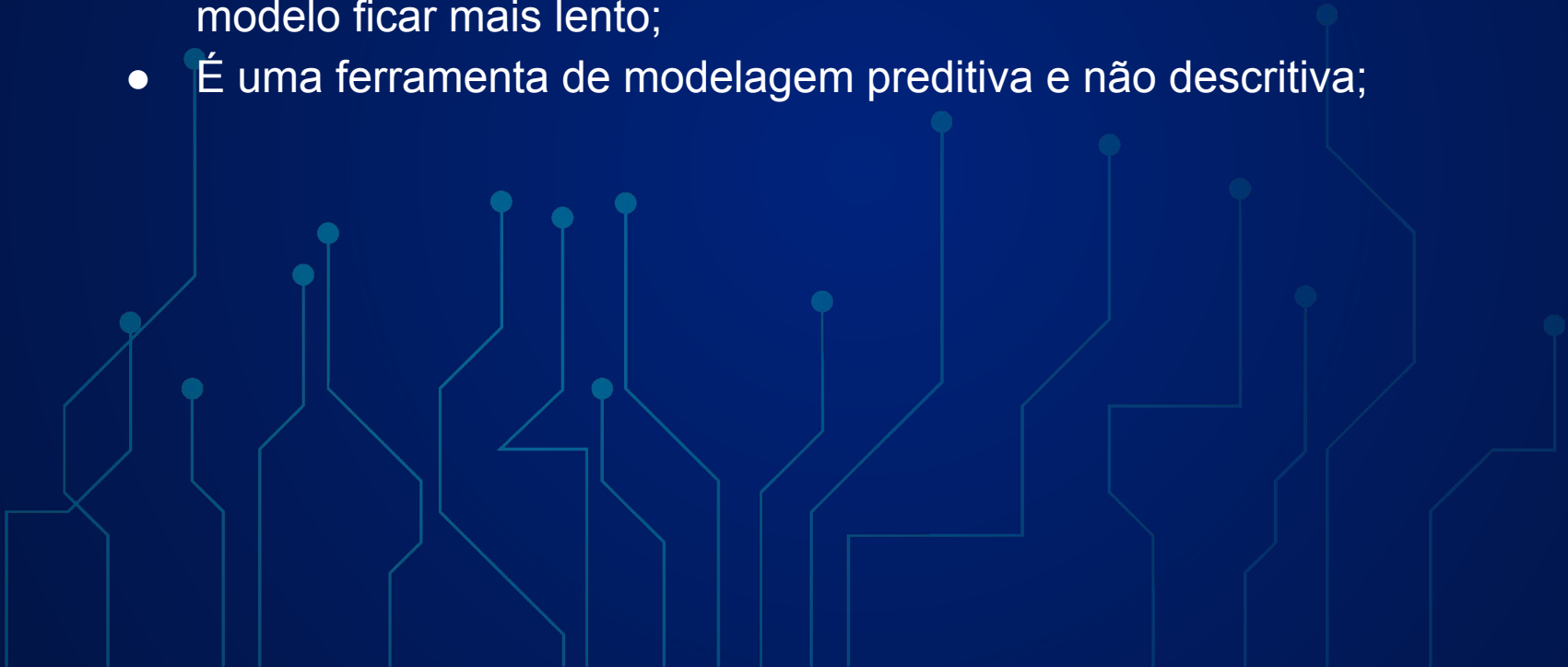
- Pode ser utilizado tanto para tarefas de classificação, quanto para regressão.
- Pode trabalhar com valores ausentes. Usa valores da mediana para substituir variáveis contínuas e computa a média ponderada da proximidade;
- Não tem problema de overfitting no modelo;

An abstract pattern of light blue lines and dots on a dark blue background, resembling a circuit board or digital network, located on the left side of the slide.

07

Desvantagens da Floresta Randômica

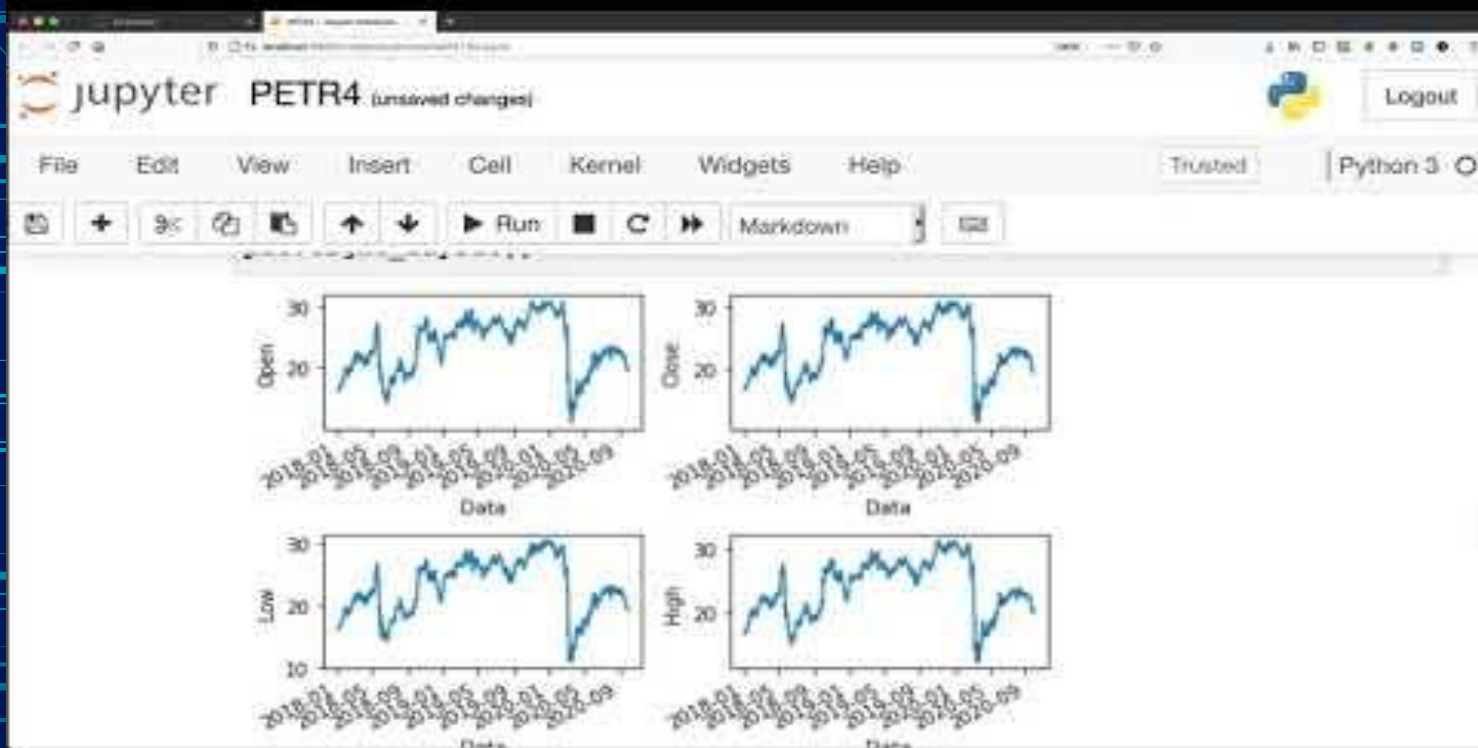
- Com uma grande quantia de árvores pode tornar o algoritmo lento e ineficiente para predições em tempo real;
- Uma predição com mais acurácia requer mais árvores, o que faz o modelo ficar mais lento;
- É uma ferramenta de modelagem preditiva e não descritiva;



An abstract pattern of glowing blue lines and dots on a dark blue background, resembling a circuit board or data flow. The lines are of varying thickness and connect various points, some of which are highlighted with small, bright blue circles.

08

Aplicação



An abstract pattern of light blue lines and dots on a dark blue background, resembling a circuit board or a network diagram. The lines are of varying thickness and connect various points, some of which are marked with small, glowing blue dots.

09

Hands-On Floresta Randômica

Exemplo de Classificação com Random Forest

Dataset: Brain Stroke (Derrame Cerebral)

Dados: Inventário de dados de pacientes diagnosticados com derrame cerebral.

1) sexo: "Masculino", "Feminino" ou "Outro"
2) idade: idade do paciente
3) hipertensão: 0 se o paciente não tem hipertensão, 1 se o paciente tem hipertensão
4) doença cardíaca: 0 se o paciente não tiver nenhuma doença cardíaca, 1 se o paciente tiver uma doença cardíaca
5) casou-se: "Não" ou "Sim"
6) tipo de trabalho: "nunca trabalhou", "particular" ou "autônomo"
7) tipo de residência: "rural" ou "urbano"
8) avggglicoselevel: nível médio de glicose no sangue
9) IMC: índice de massa corporal
10) smoking_status: "ex-fumou", "nunca fumou", "fuma" ou "Desconhecido"
11) acidente vascular cerebral: 1 se o paciente teve um acidente vascular cerebral ou 0 se não

Link: <https://www.kaggle.com/datasets/jillanisofttech/brain-stroke-dataset>

GitHub: <https://github.com/gabrielnunesmarques/MachineLearning>



Algumas bibliotecas utilizadas

- Pandas (Manipulação e Análise de Dados)
- Numpy (Operações matemáticas)
- Seaborn (Análise Estatística de Dados)
- Matplotlib (Gráficos)
- Scikit-Learn (Treino e teste de modelos)

Análise Inicial do Dataset

```
df.head()
```

✓ 0.4s

	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
2	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
3	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1
4	Male	81.0	0	0	Yes	Private	Urban	186.21	29.0	formerly smoked	1

Análise Inicial do Dataset

Temos um Oversampling no Dataset, precisamos equalizar os dados.

```
percentage_no_stroke = 100*(4733/float(df.shape[0]))  
percentage_no_stroke
```

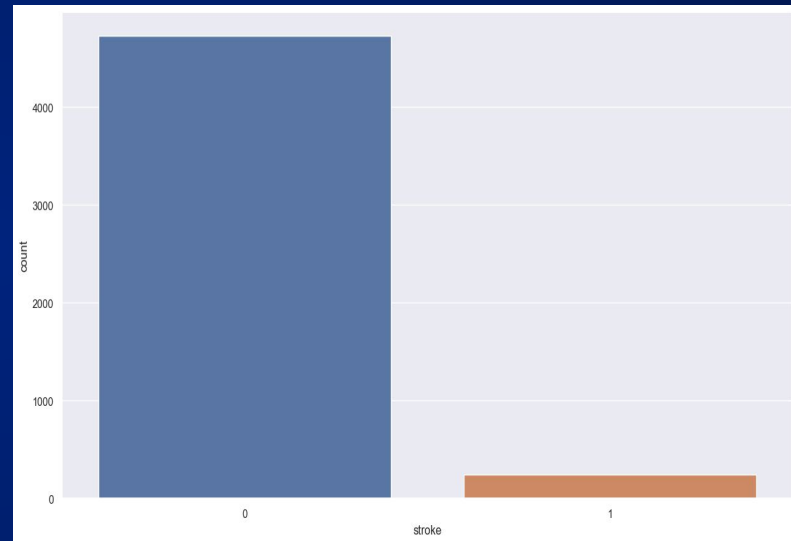
[73] ✓ 0.8s

*** 95.0210801043967

```
percentage_yes_stroke = 100*(248/float(df.shape[0]))  
percentage_yes_stroke
```

[74] ✓ 0.5s

*** 4.978919895603292



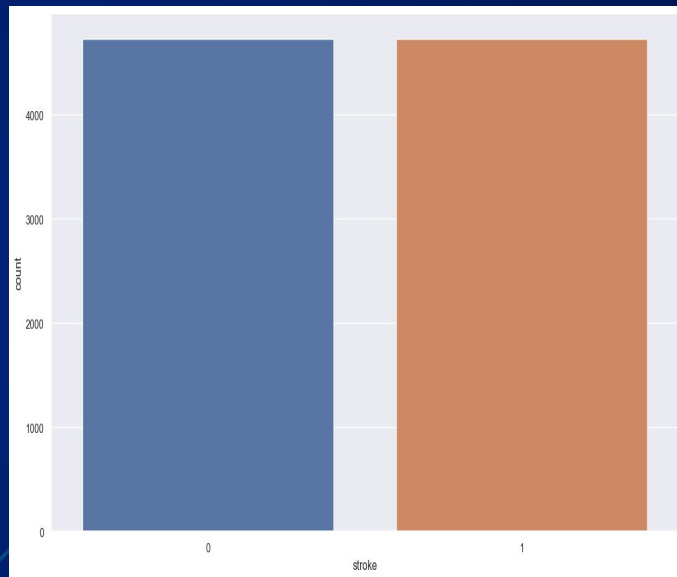
Métodos utilizados:

SMOTE

```
import seaborn as sns
from sklearn.datasets import make_classification
from imblearn.over_sampling import SMOTE
```

```
oversample = SMOTE()
x, y = oversample.fit_resample(x, y)
```

✓ 0.7s



Métodos utilizados:

OrdinalEncoder

- Ao analisar o dataset observamos que algumas colunas não continham valores numéricos, precisamos aplicar o OrdinalEncoder para transformar os dados do tipo “objeto” para numérico.

```
df.head()
```

	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
2	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
3	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1
4	Male	81.0	0	0	Yes	Private	Urban	186.21	29.0	formerly smoked	1

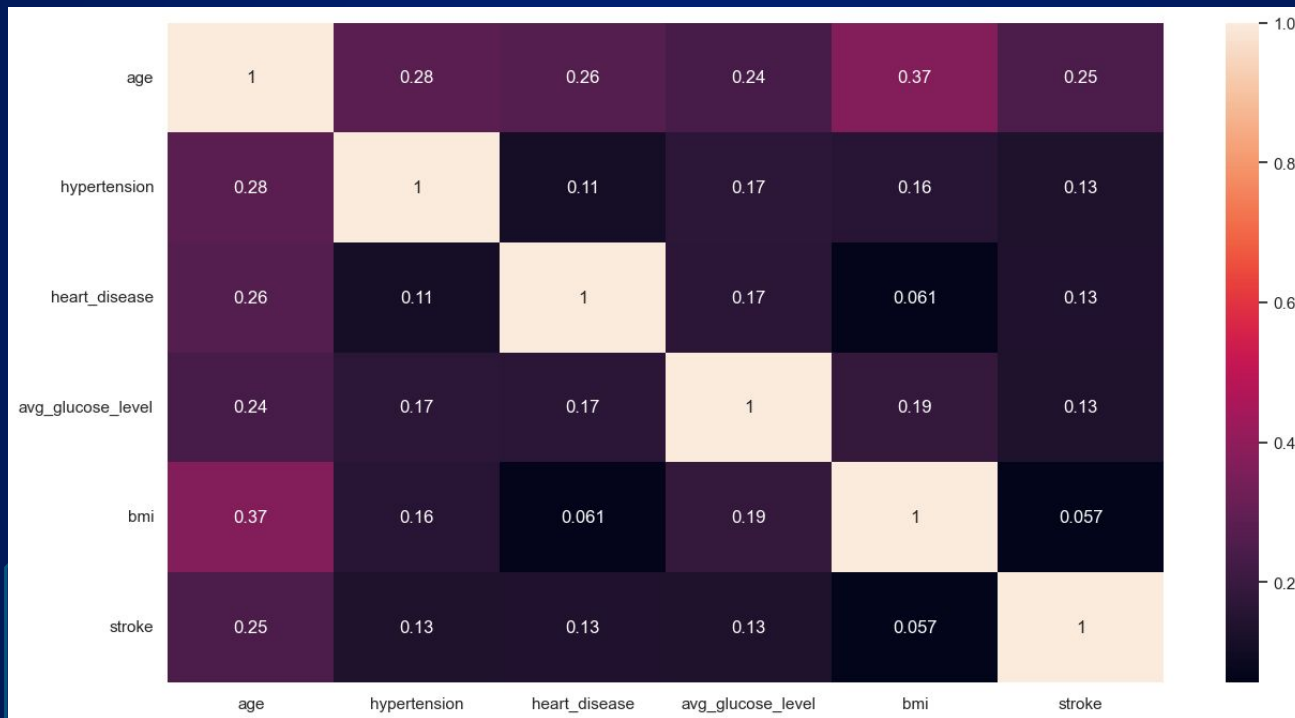
Antes

	gender	age	hypertension	heart_disease	avg_glucose_level	bmi	smoking_status	stroke
0	1.0	67.0	0	1	228.69	36.6	1.0	1
1	1.0	80.0	0	1	105.92	32.5	2.0	1
2	0.0	49.0	0	0	171.23	34.4	3.0	1
3	0.0	79.0	1	0	174.12	24.0	2.0	1
4	1.0	81.0	0	0	186.21	29.0	1.0	1

Depois

Métodos utilizados:

HetMap



Métodos utilizados:

Scikit Learn GridSearchCV e RandomForestClassifier

O **GridSearchCV** é um módulo do Scikit Learn e é utilizado para automatizar grande parte do processo de tuning. O objetivo primário do GridSearchCV é a criação de combinações de parâmetros para posteriormente avaliá-las.

```
# alocando varios parâmetros para buscar qual o melhor
param_grid = {
    'n_estimators': np.linspace(2100, 2300, 5, dtype = int),
    'max_depth': [170, 180, 190, 200, 210, 220],
    'min_samples_split': [2, 3, 4],
    'min_samples_leaf': [2, 3, 4, 5]
}

✓ 0.3s

# Retreino da floresta
rf_grid = RandomForestClassifier(criterion = 'entropy', bootstrap = True, n_jobs=-1)
# Inicialização da floresta com os valores do param_gri encontrados anteriormente
grid_rf_search = GridSearchCV(estimator = rf_grid, param_grid = param_grid,
                               cv = 5, n_jobs = 8, verbose = 2)
grid_rf_search.fit(x_train, y_train)

✓ 175m 5.8s
```

Métodos utilizados:

Scikit Learn GridSearchCV e RandomForestClassifier

Retornando os melhores parâmetros

```
# Retornando os melhores parâmetros para serem utilizados.  
best_rf_grid = grid_rf_search.best_estimator_  
grid_rf_search.best_params_ # printando os melhores parametros
```

✓ 0.1s

```
{'max_depth': 170,  
 'min_samples_leaf': 2,  
 'min_samples_split': 2,  
 'n_estimators': 2100}
```

```
# printando novamente os resultados obtidos na floresta com os melhores parâmetros.  
print(grid_rf_search.score(x_train, y_train))  
print(grid_rf_search.score(x_test, y_test))
```

✓ 2.5s

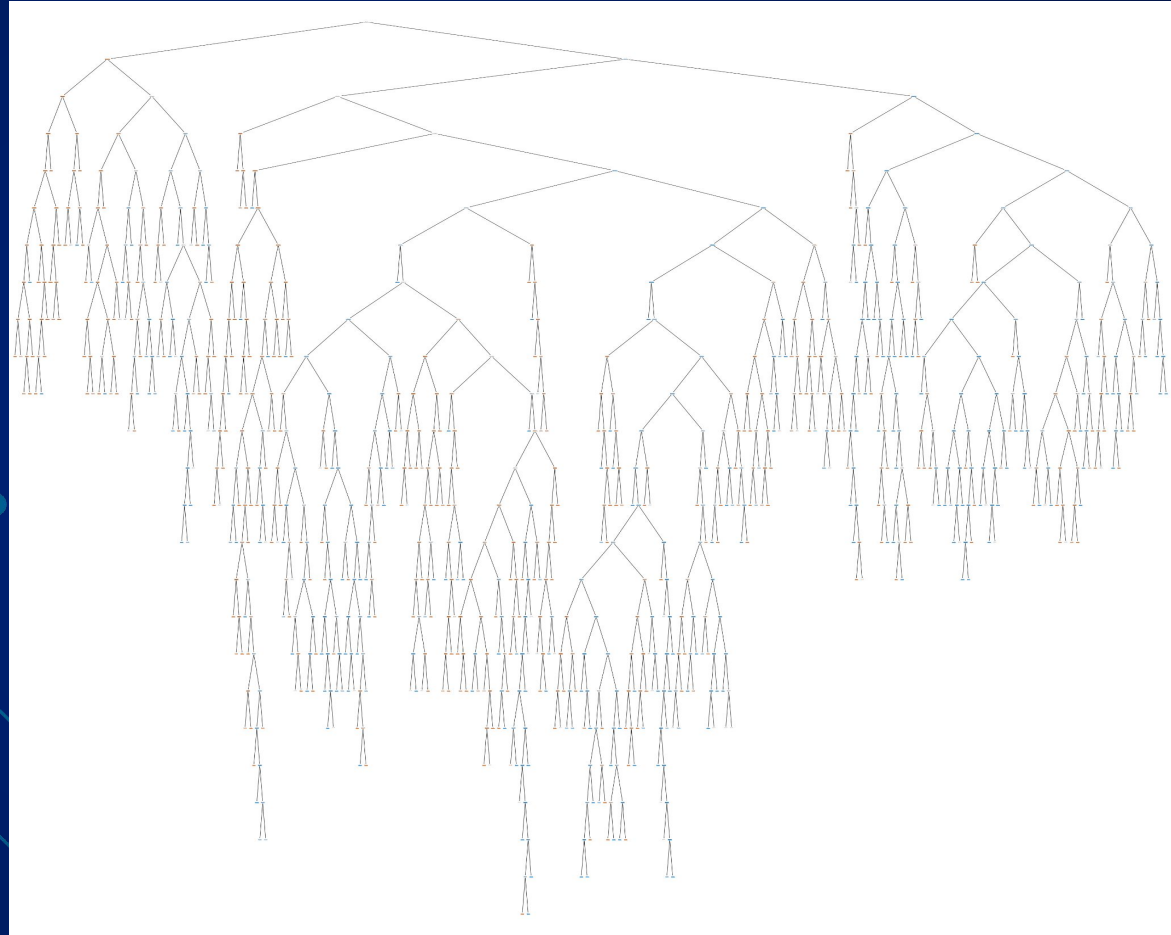
0.9935287902799789

0.9656810982048575

Métodos utilizados:

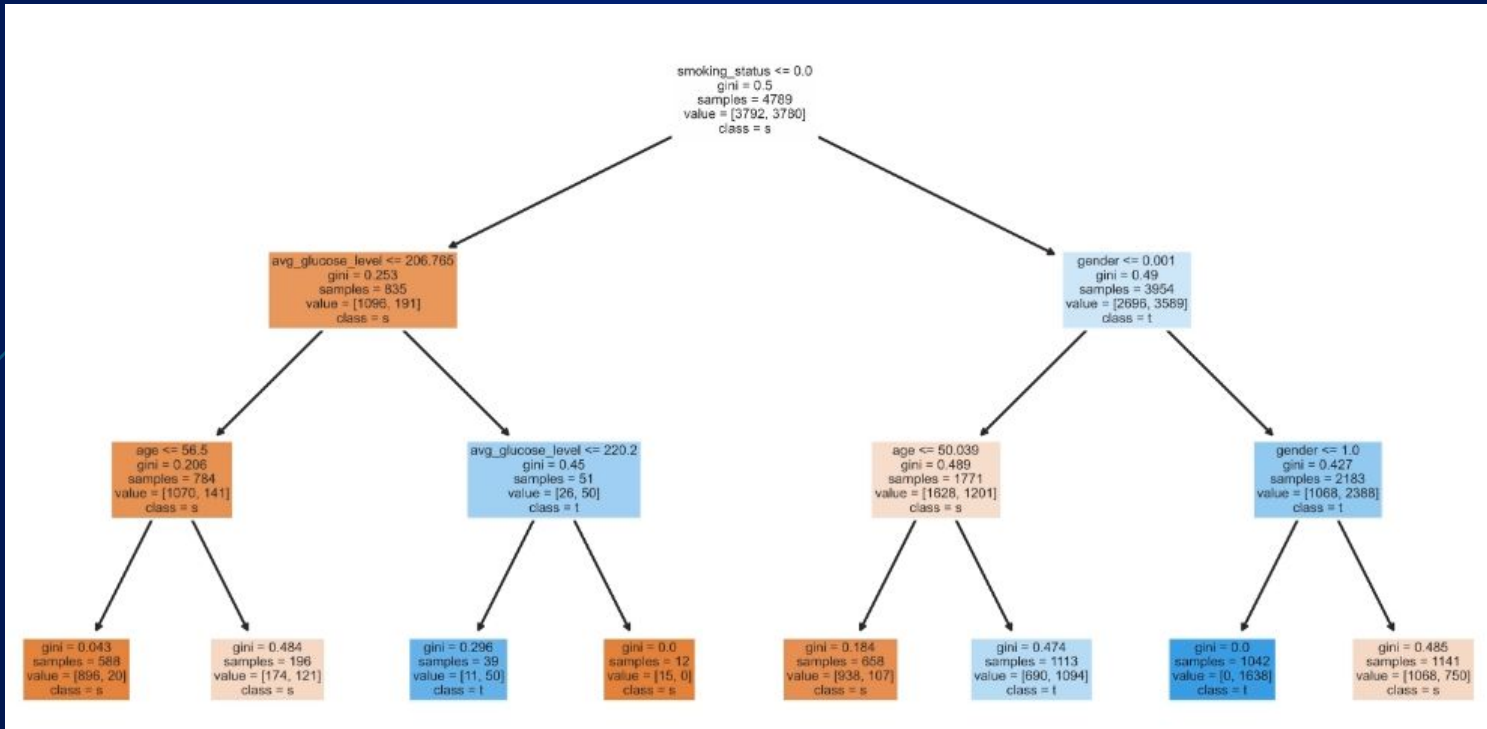
`tree.plot_tree`

Plotagem de uma árvore
de decisão.



Métodos utilizados:

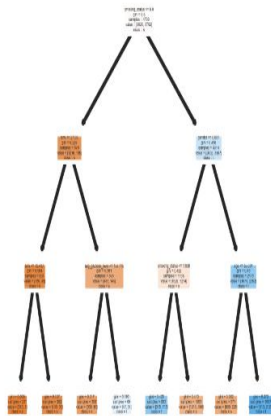
`tree.plot_tree` - Plotagem de uma árvore de decisão simplificada.



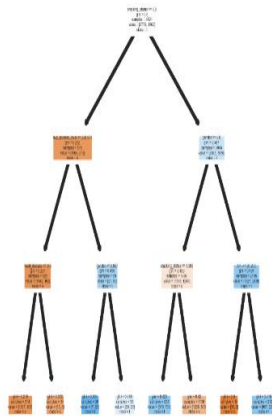
Métodos utilizados:

tree.plot_tree - Plotagem da floresta simplificada

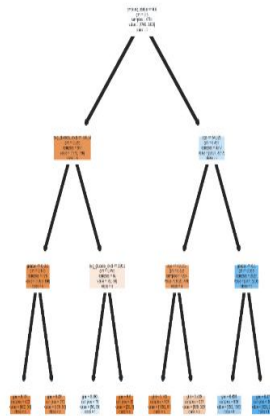
Estimator: 0



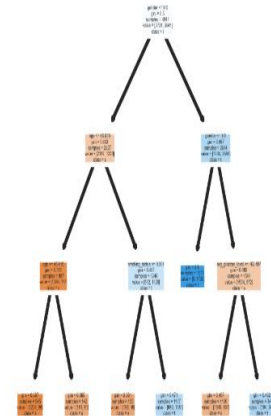
Estimator: 1



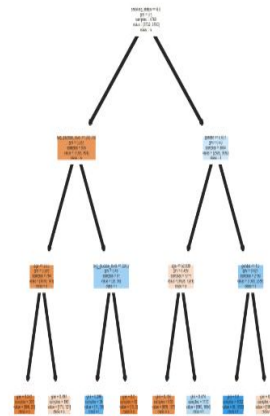
Estimator: 2



Estimator: 3



Estimator: 4



Classificação final

```
# Fazendo uma previsão
# [gender,age,hypertension,heart_disease,glucose_level,bmi,smokes]
# gender = 1 male, 0 female
# age
# hypertension = 0 yes, 1 no
# heart_disease = 0 no, 1 yes
# glucose_level
# bmi
# smokes = 1 formal, 2 never, 3 smokes e 4 unknow
row = [[1, 67, 0, 1, 228.69, 36.6, 1]]
y_trainhat = clf_3.predict(row)
print('Classificação: %d' % y_trainhat[0])
if(y_trainhat == 0):
    print('Classificação: Paciente provavelmente não terá um Derrame')
if(y_trainhat == 1):
    print('Classificação: Paciente poderá ter um Derrame')
```

✓ 0.1s

Classificação: 1

Classificação: Paciente poderá ter um Derrame