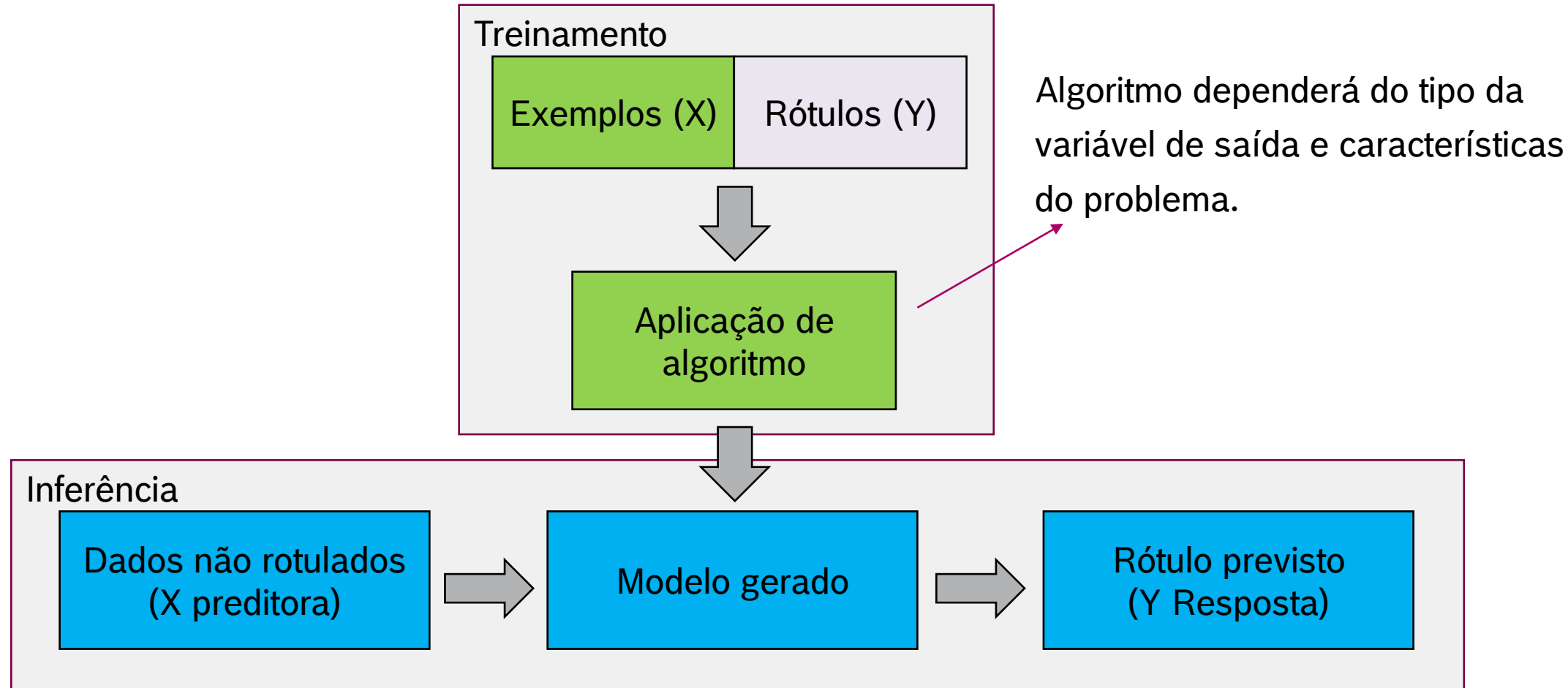


6. APRENDIZAGEM SUPERVISIONADA

DEFINIÇÃO DO PROBLEMA DE APRENDIZAGEM SUPERVISIONADA

- ▶ O sistema de aprendizado supervisionado utiliza dados que contêm a **resposta desejada**, denominados de dados rotulados.
- ▶ Este tipo de aprendizado cria **modelos** que aprendem os **padrões** de características necessárias para determinados resultados de saída, com base nos dados de treinamento, gerando as regras para mapear inputs de dados novos e gerar seus rótulos.
- ▶ O aprendizado supervisionado pode ser aplicado em um problema de **regressão**, para valores **contínuos/numéricos**, ou de **classificação**, para valores **discretos/categóricos**.
- ▶ É a subárea do aprendizado de máquina que concentra a **maioria das aplicações**.

FLUXO DE APRENDIZAGEM SUPERVISIONADA



NOMENCLATURAS

- ▶ Dados de entrada / Preditores / Inputs / X: **Colunas** com características (features).
- ▶ Dado de saída / Respostas / Outputs / Y: Uma **Coluna Resposta** (target) ou **Rótulo** (label).
- ▶ Amostras / Exemplos / Linhas: Cada linha (row) é um exemplo/amostra com valores específicos para cada coluna e rótulo associado à **configuração das características presentes**.

The diagram shows a table with 5 columns and 5 rows. A bracket on the left labeled 'Rows' spans all rows. A bracket above the first four columns labeled 'Features' spans those columns. A bracket above the fifth column labeled 'Label' spans that column. A bracket below all columns labeled 'Columns' spans all columns.

Size	Beds	Baths	Zip	Price
1100	1	1	64576	1.29
1900	3	1.5	78321	2.14
2800	3	3	98712	3.10
3400	4	3.5	25721	3.75

Fonte: https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcRtjKS-EOr4YJIClu_6rN-2fnPF0BO0ww8NHQ&usqp=CAU

EXEMPLO DE DADOS ROTULADOS

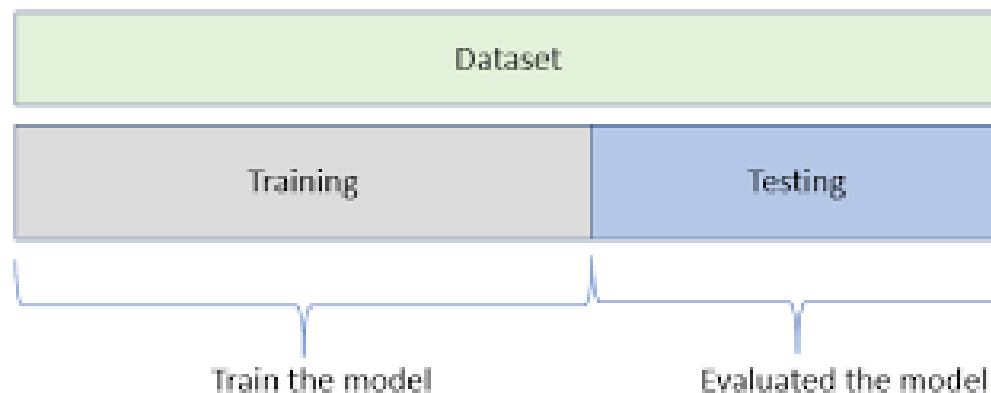
Exemplos

Características / X's					Rótulo / Y
Features					Label
Position	Experience	Skill	Country	City	Salary (\$)
Developer	0	1	USA	New York	103100
Developer	1	1	USA	New York	104900
Developer	2	1	USA	New York	106800
Developer	3	1	USA	New York	108700
Developer	4	1	USA	New York	110400
Developer	5	1	USA	New York	112300
Developer	6	1	USA	New York	114200
Developer	7	1	USA	New York	116100
Developer	8	1	USA	New York	117800
Developer	9	1	USA	New York	119700
Developer	10	1	USA	New York	121600

Fonte: <https://www.i2tutorials.com/wp-content/media/2019/09/Features-and-Labels-in-a-Dataset-i2tutorials.png>

SEPARAÇÃO DOS DADOS EM TREINO E TESTE

- ▶ Ao treinar um modelo devemos separar uma **porcentagem** dos dados para **teste**.
- ▶ Esta porcentagem também estará rotulada e servirá para **avaliar o desempenho do modelo**, comparando a resposta predita com a resposta real e simular o ambiente de produção, em que o modelo encontrará dados que não foram vistos durante o treinamento.
- ▶ Os dados costumam ser separados para treino e teste em proporções próximas de **70/30%** ou **80/20%**.



Fonte: <https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcTaJW68USrBEu7SW-cqESSBYjqrvtEyH2PxBA&usqp=CAU>

Exemplo: Treinamento / Teste

- Suponha que uma loja de roupas deseja criar um modelo de predição de lucros. Para tanto, a loja pretende usar um banco de dados com informações sobre hábitos de compras de 100 clientes cadastrados. Os dados de hábitos dos clientes foram registrados por câmeras espalhadas na loja. Sempre que um cliente se aproxima de determinado mostruário de roupas, as câmeras acionam um cronômetro que conta o tempo entre o cliente escolher determinada peça de roupa e realizar o pagamento no caixa. Você foi contratado para desenvolver um modelo usando aprendizado de máquina. Para desenvolver o modelo a loja disponibilizou a você as seguintes informações:
 - Intervalo de tempo em minutos que um cliente leva para escolher um produto e efetuar o pagamento no caixa (dados preditores ou inputs);
 - Valor da compra efetuada pelo cliente (dados de resposta ou output).

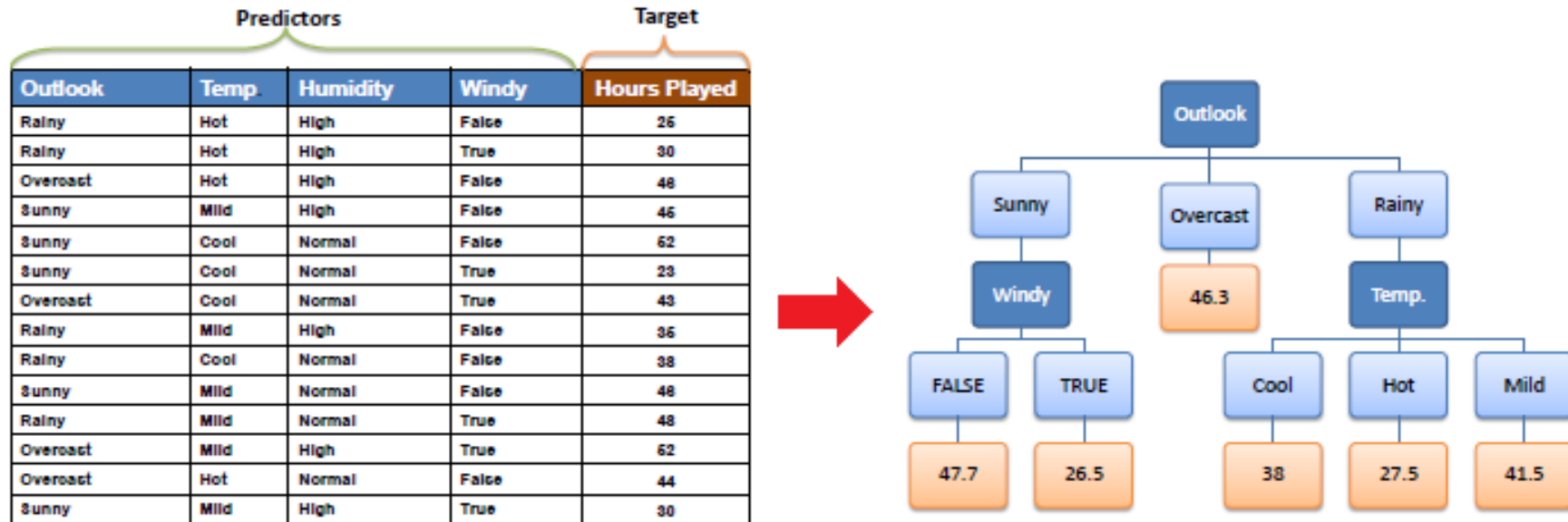
ALGORITMOS DE CLASSIFICAÇÃO

- ▶ O algoritmo de **classificação** visa prever uma **saída discreta**, seu objetivo é prever uma **classe** ou **categoria** resultante.
- ▶ Com este tipo de algoritmo é possível classificar se uma foto contém um gato ou um cachorro, se o e-mail é spam ou não, etc.
- ▶ Quando envolve apenas **duas opções** chamamos de classificação **binária**, quando envolve **mais de duas** chamamos de classificação **multiclasse**.
- ▶ O resultado do algoritmo de classificação pode ser um vetor de probabilidades entre 0 e 1, indicando a **probabilidade** da observação pertencer a cada classe.

EXEMPLOS DE ALGORITMOS DE CLASSIFICAÇÃO

► Algoritmos baseados em árvore: Decision Tree.

Representam o resultado como uma série de regras agrupadas similares a condicionais if e else ou um fluxograma.



Fonte: https://www.saedsayad.com/images/Decision_tree_r1.png

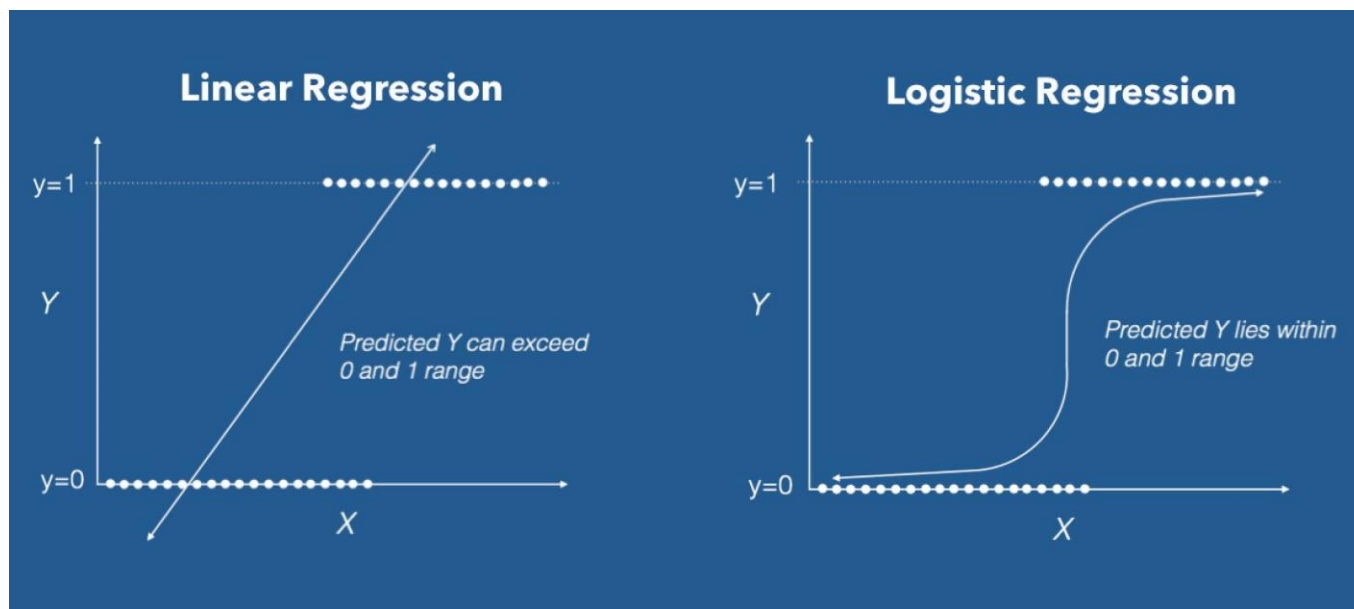
Exemplo: Árvore de Decisão

- ▶ Você quer ajudar seu colega a decidir se deve ou não ir em alguns shows de comédia do próximo mês. Felizmente, ele fez várias anotações numa agenda, marcando algumas características sobre o comediante que o ajudaram a decidir se ia ou não em shows anteriores.
- ▶ Com base nas anotações, crie uma árvore de decisão que poderá ser usada para ajudar seu colega decidir se vai ou não aos próximos shows de comédia.

idade	carreira	pontos	estado	foi
32	8	13	RJ	N
38	10	8	MG	N
27	2	10	SP	N
48	2	8	MG	N
37	19	12	MG	S
40	12	9	RJ	N
62	1	11	SP	S
31	12	13	RJ	S
48	11	11	SP	S
31	3	13	SP	S
20	1	9	MG	N
22	1	11	RJ	S
41	7	13	RJ	S

LOGISTIC REGRESSION

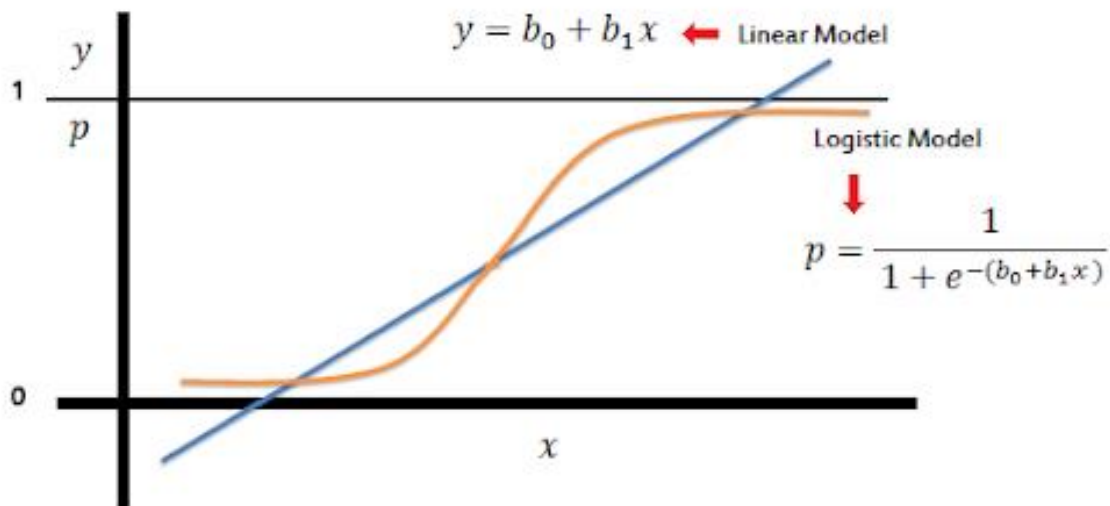
- ▶ Enquanto na Regressão Linear temos uma variável resposta contínua, na **Regressão Logística** nossa variável resposta é **binária**, 0 ou 1, sim ou não.
- ▶ Ao plotar a relação entre X e y , não é possível implementar uma reta para predizer os valores.



Fonte: https://miro.medium.com/max/810/1*GSrvyVfYySZgzpQt94c3gw.jpeg

LOGISTIC REGRESSION – SIGMOID

- Utilizamos uma função chamada sigmoid, transformando-o resultado em forma de “S” e sendo capaz de fornecer uma previsão sempre entre 0 e 1, de forma que possamos interpretar seus resultados como uma probabilidade válida.



Fonte: https://miro.medium.com/max/1826/1*-bYLMCVximybDe15oeaZ2Q.png

Exercicio: Regressão Logística

- Suponha que você deseja criar um modelo de classificação usando regressão logística para prever se será aprovado ou reprovado em uma matéria da faculdade.

Os dados que você possui são:

Horas de estudos semanais;

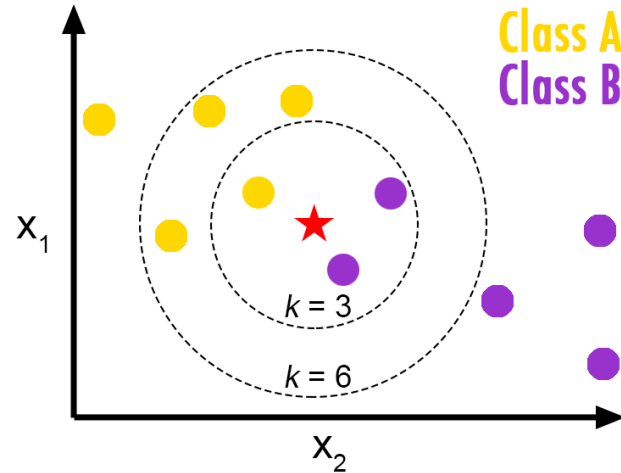
Métodos de estudo A e B;

Resultados das provas anteriores: se foi aprovado ou reprovado.

Desenvolva um modelo de regressão logística usando um script em Python!

K-NEAREST NEIGHBOUR (KNN)

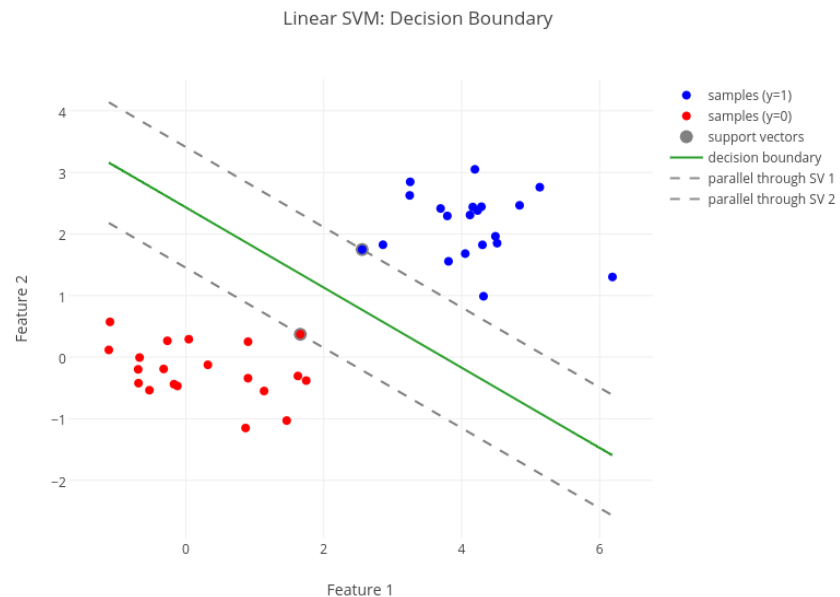
- ▶ É o tipo de modelo de classificação mais simples e intuitivo. Ao usar os dados de treinamento como base, é possível rotular um dado novo obtendo a maioria dos rótulos nos K-vizinhos mais próximos.
- ▶ Devido a natureza do algoritmo, é possível obter diferentes rótulos e resultados dependendo do valor de K escolhido. Deve-se verificar a partir das métricas e das taxas de erro qual o melhor valor para K.



Fonte: www.researchgate.net/profile/Reza_Arghandeh/publication/330400464/figure/fig9/AS:715398974562316@1547575815590/8-An-Example-of-K-nearest-Neighbors-10.png

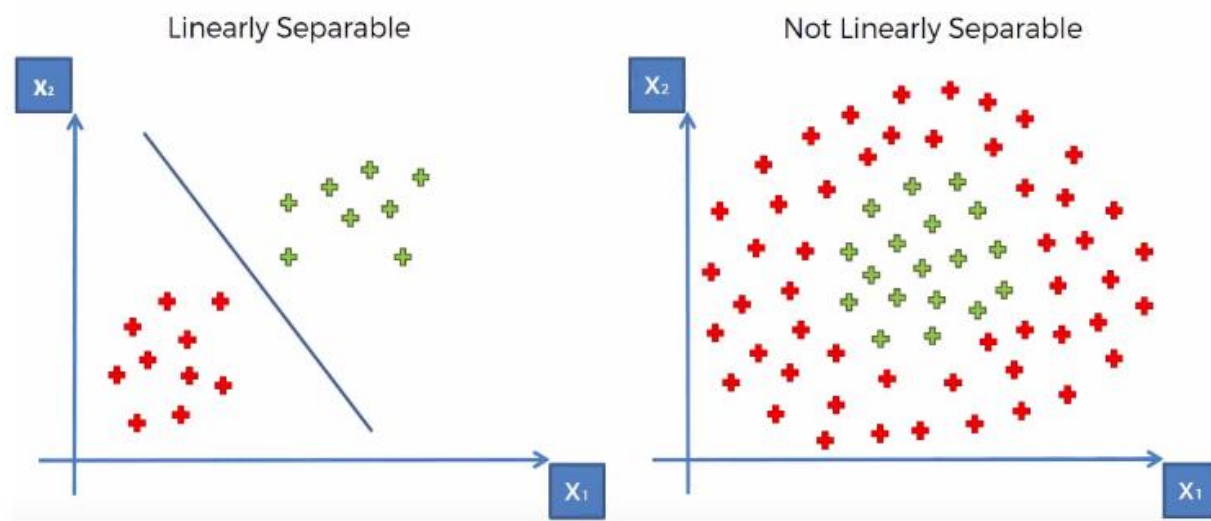
SUPPORT VECTOR MACHINE (SVM)

- O **SVM** é um algoritmo que busca uma linha de separação entre duas classes distintas analisando os dois pontos, um de cada grupo, mais próximos da outra classe. Isto é, o **SVM** escolhe a reta — também chamada de **hiperplano** em maiores dimensões— entre dois grupos que se distancia mais de cada um, visando maximizar a **margem** de separação.

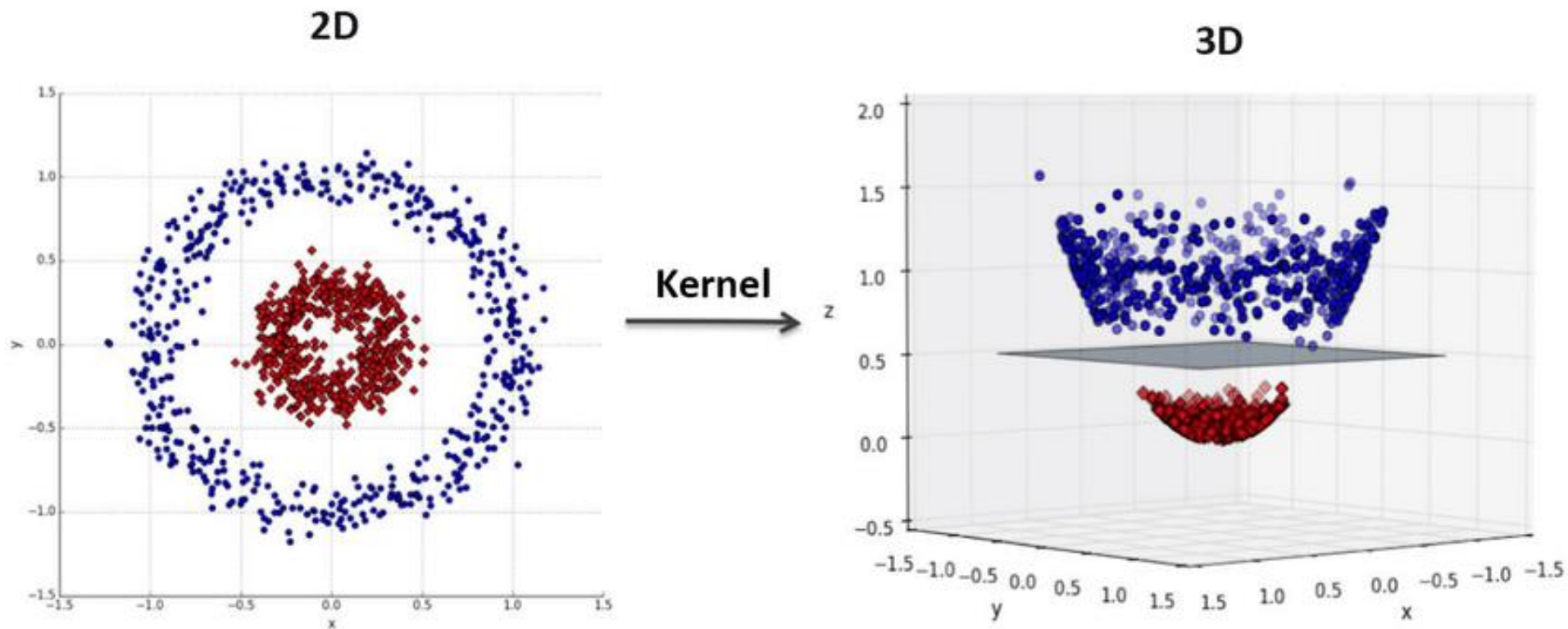


LINEARIDADE X NÃO-LINEARIDADE

- ▶ Existem problemas em que uma função linear não é o suficiente para separar os dados, prejudicando o uso de algoritmos lineares.
- ▶ O algoritmo SVM possui alguns artifícios para lidar com este tipo de situação, utilizando **truque de kernels** incrementa o número de dimensões do espaço trabalhado e possibilita a divisão dos dados por meios lineares.

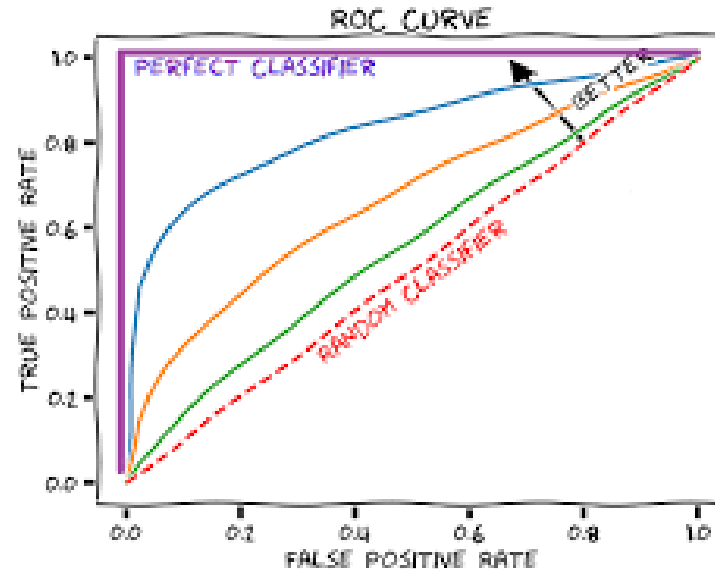


KERNEL TRICK



ROC CURVE

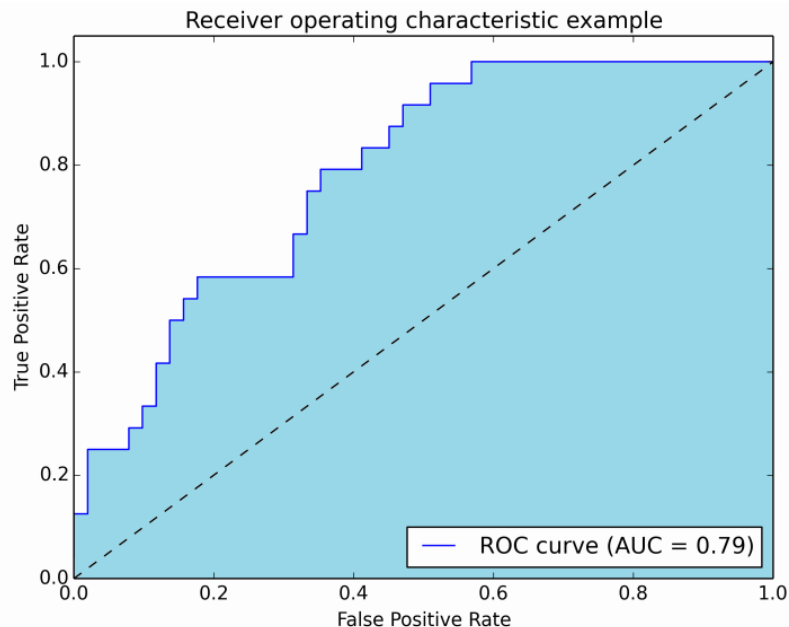
- ▶ A curva **ROC (Receiver Operator Characteristic)** mostra o quão bom o modelo criado pode distinguir entre duas coisas (já que é utilizado para classificação binária).
- ▶ Uma curva **ROC** traça “*True Positive Rate vs. False Positive Rate*” em diferentes limiares de classificação



Fonte: <https://glassboxmedicine.files.wordpress.com/2019/02/roc-curve-v2.png?w=576&resize=398%2C299>

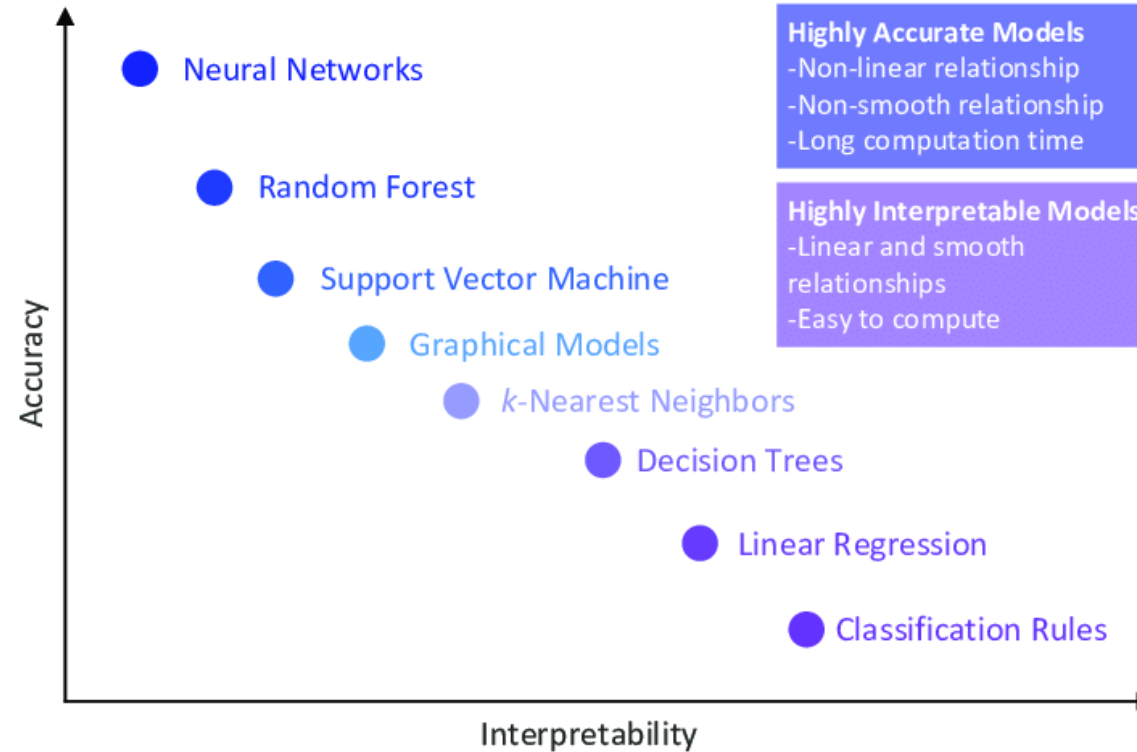
AREA UNDER THE ROC CURVE (AUC)

- ▶ A **AUC** nada mais é que uma maneira de resumir a curva **ROC** em um único valor, agregando todos os limiares da **ROC**, calculando a “*área sob a curva*”.
- ▶ O valor do **AUC** varia de 0,0 até 1,0 e quanto maior o valor, melhor.



Fonte: https://miro.medium.com/max/1050/1*RqK5DjVxcj4qZsCdN4FOSQ.png

INTERPRETABILIDADE



Fonte: [https://www.researchgate.net/profile/Manuel_Eugenio_Morocho-](https://www.researchgate.net/profile/Manuel_Eugenio_Morocho-Cayamcela/publication/335937022/figure/fig4/AS:805196397543436@1568985189998/The-trade-off-between-interpretability-and-accuracy-of-some-relevant-ML-models-Highly.png)

[Cayamcela/publication/335937022/figure/fig4/AS:805196397543436@1568985189998/The-trade-off-between-interpretability-and-accuracy-of-some-relevant-ML-models-Highly.png](https://www.researchgate.net/profile/Manuel_Eugenio_Morocho-Cayamcela/publication/335937022/figure/fig4/AS:805196397543436@1568985189998/The-trade-off-between-interpretability-and-accuracy-of-some-relevant-ML-models-Highly.png)

Exercicio

▶ 06_04_MultiClassClassification.ipynb

RECOMENDAÇÕES

- ▶ Cada algoritmo tem uma série de parâmetros únicos, e cada um destes parâmetros podem ser modificados e irão influenciar os resultados. Além de dominar os algoritmos, deve-se dominar as características únicas de cada tipo.
- ▶ Comece sempre pelos modelos mais simples para validar a resolução do problema e só então prossiga para os métodos mais complexos.
- ▶ As métricas devem ser analisadas sempre em conjunto, principalmente para conclusões sobre comparação de modelos.