Aprendizagem de Máquina

Modelos Baseados em Árvores

Telmo de Menezes e Silva Filho
tmfilho@gmail.com/telmo@de.ufpb.br
www.de.ufpb.br

UFPB
Departamento de ESTATÍSTICA

Sumário

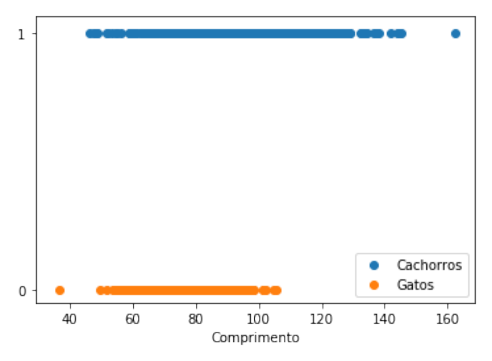
Exemplo de conjunto de dados

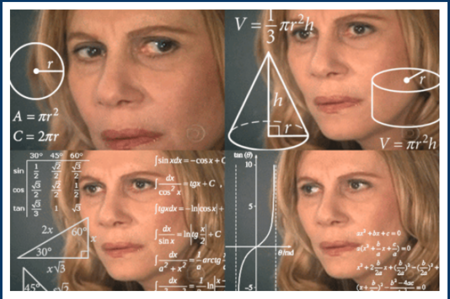
- O conjunto abaixo contém dados fictícios de cães e gatos:

	Comprimento (cm)	Altura (cm)	Peso (kg)	Classe
1	89,64	44,23	20,39	Cachorro
2	69,27	43,06	15,03	Cachorro
3	70,52	28,30	12,32	Cachorro
4	88,28	49,45	15,81	Cachorro
...	...	...	...	...
996	87,46	18,67	4,21	Gato
997	69,98	23,90	5,57	Gato
998	85,52	23,87	6,33	Gato
999	88,44	23,76	5,63	Gato

Como nós humanos faríamos essa tarefa?

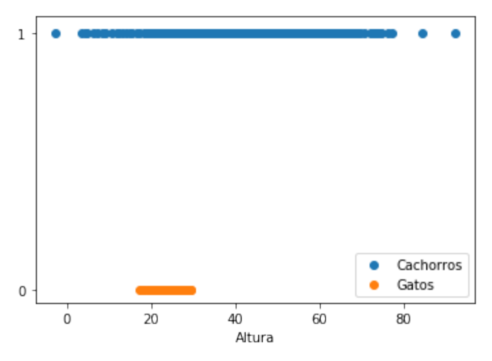
- Uma forma intuitiva de categorizar novos exemplos como cães e gatos é observar as variáveis que os descrevem

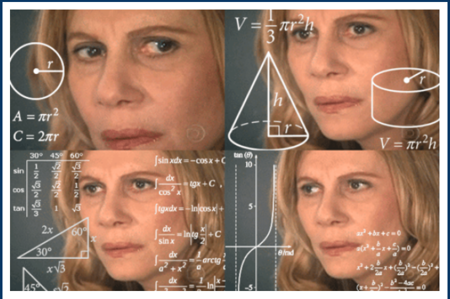




Como nós humanos faríamos essa tarefa?

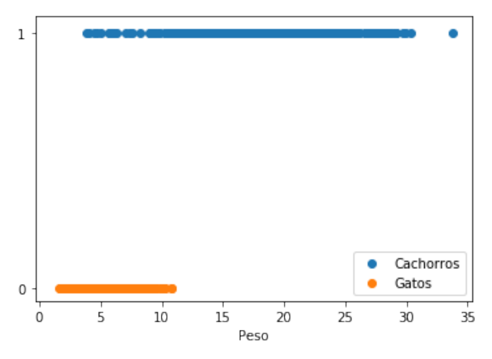
- Uma forma intuitiva de categorizar novos exemplos como cães e gatos é observar as variáveis que os descrevem

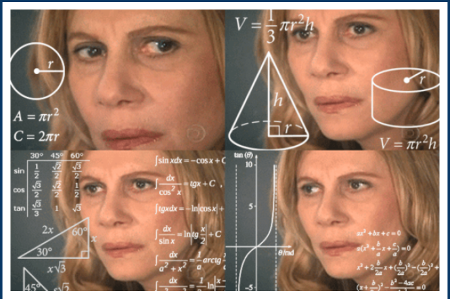




Como nós humanos faríamos essa tarefa?

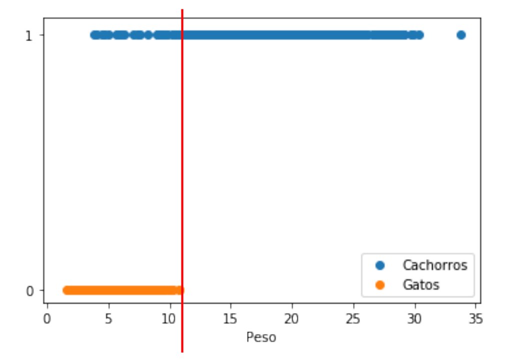
- Uma forma intuitiva de categorizar novos exemplos como cães e gatos é observar as variáveis que os descrevem

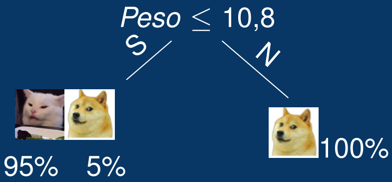




Como nós humanos faríamos essa tarefa?

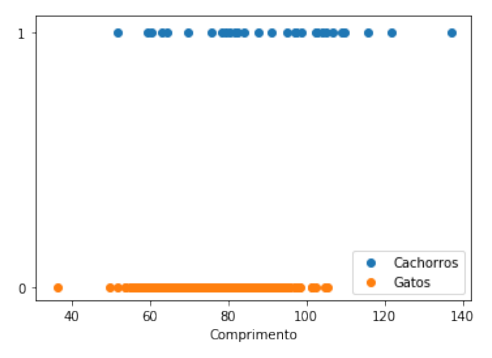
- Podemos definir um ponto de corte (10,8kg) em cima da variável *Peso*





Como nós humanos faríamos essa tarefa?

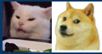
- Podemos agora tentar diferenciar os animais que têm menos de 10,8kg usando as outras duas variáveis



$Peso \leq 10{,}8$

S

N

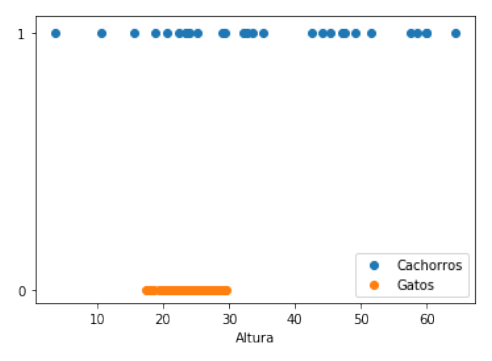


95% 5%

100%

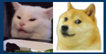
Como nós humanos faríamos essa tarefa?

- Podemos agora tentar diferenciar os animais que têm menos de 10,8kg usando as outras duas variáveis

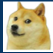


$Peso \leq 10{,}8$

S *N*



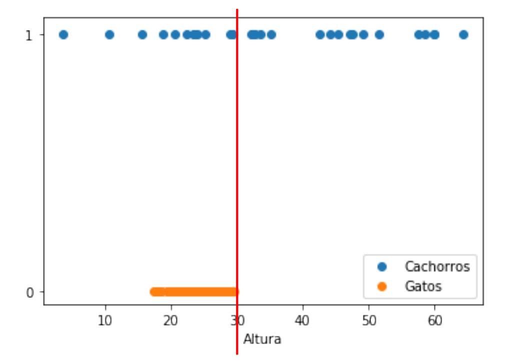
95% 5%

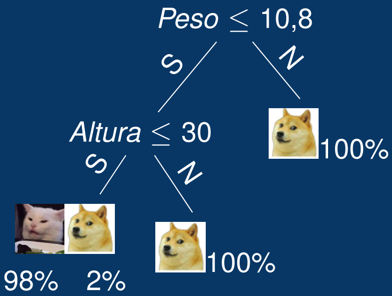


100%

Como nós humanos faríamos essa tarefa?

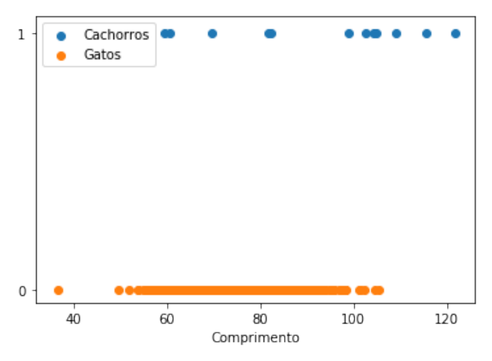
- Fazemos o próximo ponto de corte (30cm) usando a variável *Altura*

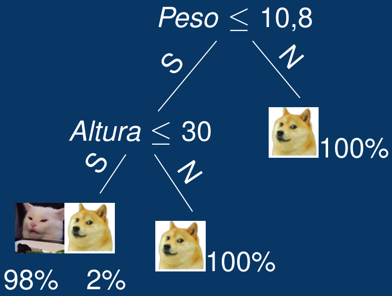




Como nós humanos faríamos essa tarefa?

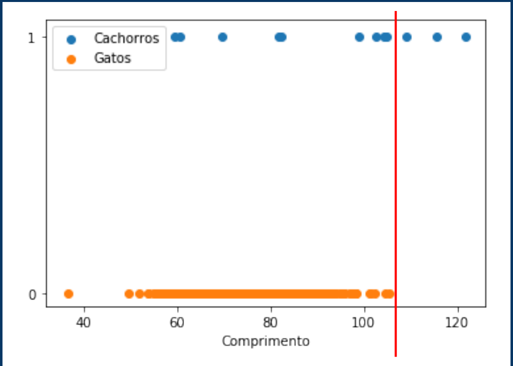
- Por último, ficamos com a variável *Comprimento*





Como nós humanos faríamos essa tarefa?

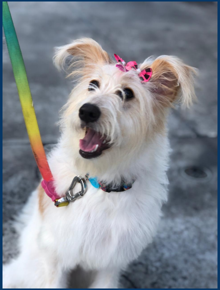
- Por último, ficamos com a variável *Comprimento*



Seguimos um algoritmo de forma natural

- Agora temos um modelo que nos permite categorizar novos exemplos

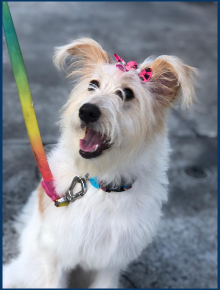
Exemplo:
Peso: 10,7kg
Altura: 46cm
Comp: 110cm



Seguimos um algoritmo de forma natural

- Agora temos um modelo que nos permite categorizar novos exemplos

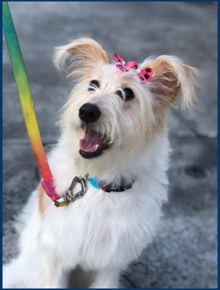
Exemplo:
Peso: 10,7kg
Altura: 46cm
Comp: 110cm



Seguimos um algoritmo de forma natural

- Agora temos um modelo que nos permite categorizar novos exemplos

Exemplo:
Peso: 10,7kg
Altura: 46cm
Comp: 110cm



Peso $\leq$ 10,8

S

Altura $\leq$ 30

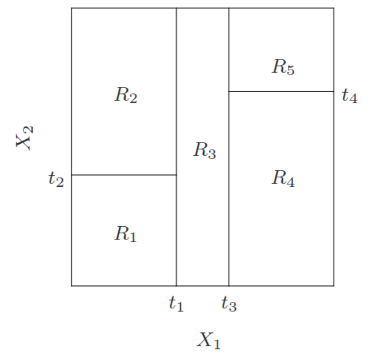
N


100%

Árvores de Decisão para Classificação

Árvores de Decisão para Classificação

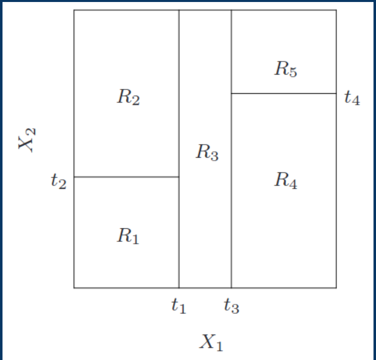
- Nos slides anteriores, construímos nossa árvore de decisão manualmente
- Para que o algoritmo seja útil, precisamos encontrar uma forma de escolher a variável e o ponto de corte a cada nível **automaticamente**
- Suponha que vamos dividir nosso conjunto de dados em M regiões $R_1, R_2, \ldots, R_M$



Árvores de Decisão para Classificação

- Na região R_m, com N_m observações, podemos calcular a proporção da classe k

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{\mathbf{x}_i \in R_m} \mathbb{1}(y_i = k)$$



Árvores de Decisão para Classificação

- Na região R_m, com N_m observações, podemos calcular a proporção da classe k

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{\mathbf{x}_i \in R_m} \mathbb{1}(y_i = k)$$

- Dadas novas observações na região R_m, iremos classificá-las de acordo com

$$\hat{y}(m) = \arg\max_k \hat{p}_{mk}$$

Árvores de Decisão para Classificação

- Podemos então calcular o grau de "impureza" em uma região R_m de várias formas diferentes
  - Taxa de erro:
$$Q_m = \frac{1}{N_m} \sum_{i \in R_m} \mathbb{1}(y_i \neq \hat{y}(m)) = 1 - \hat{p}_{mk}$$
  - Índice de Gini:
$$Q_m = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$
  - Entropia cruzada:
$$Q_m = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$$

Árvores de Decisão para Classificação

- No caso binário, podemos a proporção da classe positiva de $\hat{p}_m$ e as medidas viram
  - Taxa de erro:
    $$Q_m = 1 - \max(\hat{p}_m, 1 - \hat{p}_m)$$
  - Índice de Gini:
    $$Q_m = 2\hat{p}_m(1 - \hat{p}_m)$$
  - Entropia cruzada:
    $$Q_m = -\hat{p}_m \log \hat{p}_m - (1 - \hat{p}_m) \log(1 - \hat{p}_m)$$

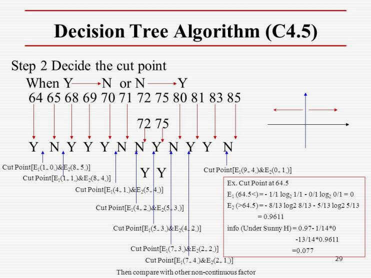
Por que Medimos a Impureza das Regiões?

- Usamos a impureza para decidir que variável e que ponto de corte vamos usar no próximo nível da árvore
- Começando com todos os dados e dados uma variável j e um ponto de corte s, definimos um par de regiões $R_1(j, s) = \{\mathbf{x} | x_j \leq s\}$ e $R_2(j, s) = \{\mathbf{x} | x_j > s\}$
- Buscamos a variável j e o ponto de corte s que minimizam

$$\min_{j,s} \left(\frac{N_1 Q_1 + N_2 Q_2}{N_1 + N_2} \right)$$

Por que Medimos a Impureza das Regiões?

- Para encontrar *j* e *s*, podemos ordenar os valores observados de cada variável *j* e testá-los sequencialmente como pontos de corte (ou selecionar aleatoriamente)



- Uma vez encontrada a melhor partição, fazemos o mesmo processo em cada região resultante
  - Recursão

Controlando o Tamanho da Árvore

- Se não pararmos o processo de construção da árvore, ele seguirá até que todas as regiões sejam puras (podendo ser uma região por instância)
  - Ou até que não ocorra ganho de informação
  - Isso resulta em uma classificação perfeita dos dados de treinamento
  - Mas quão bem isso generaliza para dados fora do conjunto de treinamento?
- Podemos definir uma profundidade máxima para a árvore ou um número mínimo de instâncias em um nó
- **Poda por custo-complexidade**
  - `https://medium.com/@sanchitamangale12/decision-tree-pruning-cost-complexity-method-194666a5dd2f`

Estimação de Probabilidades de Classe

- Dado um nó de decisão R_m e a proporção de elementos da classe k, $\hat{p}_{mk}$, seria intuitivo estimarmos $\hat{P}(Y = k|\mathbf{x} \in R_m) = \hat{p}_{mk}$, quando $\mathbf{x} \in R_m$
- No entanto, imagine que R_m contém apenas 2 instâncias da mesma classe
  - Você se sentiria confortável de estimar uma probabilidade de classe $\hat{P}(Y = k|\mathbf{x} \in R_m) = 1$ baseado em apenas 2 observações?
- Uma correção muito usada para esse problema é a suavização de Laplace (Laplace smoothing):

$$\hat{P}(Y = k|\mathbf{x} \in R_m) = \frac{\sum_{\mathbf{x}_i \in R_m} \mathbb{1}(y_i = k) + 1}{N_m + K}$$

Árvores de Decisão para Regressão

Árvores de Decisão para Regressão

- Na tarefa de regressão, construímos a árvore de decisão da mesma forma, ou seja, reduzindo a impureza dos nós recursivamente
- Podemos medir a impureza em cada região usando a soma das diferenças quadráticas

$$Q_m = \sum_{i \in R_m} (y_i - \hat{y}(m))^2$$

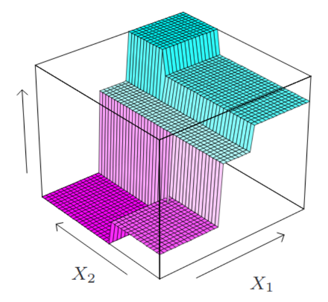
- Naturalmente, para minimizar esse valor, o $\hat{y}(m)$ ideal para a região R_m é

$$\hat{y}(m) = \frac{\sum_{i \in R_m} y_i}{N_m}$$

Árvores de Decisão para Regressão

- Podemos então construir a árvore exatamente como fazemos para a tarefa de classificação
- Buscamos a variável j e o ponto de corte s que minimizam

$$\min_{j,s} \left(\frac{N_1 Q_1 + N_2 Q_2}{N_1 + N_2} \right)$$



Importância dos Atributos

Importância dos Atributos

- Devido a sua construção, modelos de árvore de decisão podem ser usados para avaliar a importância de cada atributo para o resultado predito
- Calculamos a importância de cada atributo como a diminuição da impureza ponderada pela probabilidade de chegar ao nó
  - Aproximamos a probabilidade de chegar ao nó usando o número de instâncias no nó dividido pelo número total de instâncias N_m/N
- Quanto maior o valor calculado, maior a importância do atributo

Importância dos Atributos

- Para um atributo j, sua importância $g(j)$ é a média da redução de impureza de cada nó em que j foi escolhido

$$\delta(m) = \frac{1}{N}\left(N_m Q_m - N_{m1} Q_{m1} - N_{m2} Q_{m2}\right)$$

$$g(j) = \frac{\sum_{m \in T_j} \delta(m)}{\sum_{w \in T} \delta(w)}$$

  - Onde T e T_j indicam os conjuntos dos nós da árvore e dos nós vinculados ao atributo j, respectivamente

- Por último, normalizamos as importâncias para que somem pra 1

$$fi(j) = \frac{g(j)}{\sum_{u=1}^{p} g(u)}$$

Prós e Contras

- Árvores de decisão são muito instáveis
  - Pequenas mudanças nos dados podem mudar totalmente a árvore encontrada
- Suas decisões são facilmente interpretáveis
  - É mais fácil explicá-las para um leigo do que as predições de muitos outros modelos (frequentemente chamados de modelos caixa-preta)

Sugestões de Atividades

- Vimos como construir uma árvore de decisão de forma genérica. Busque ler sobre o algoritmo mais usado: C4.5
  - `https://sefiks.com/2018/05/13/a-step-by-step-c4-5-decision-tree-example/`
  - `https://www.youtube.com/watch?v=qPbimX0R5vg`
- Tente implementar implementar uma árvore de decisão de forma recursiva
- Tente implementar a poda por custo-complexidade

FIM

Aprendizagem de Máquina

Modelos Baseados em Árvores

Telmo de Menezes e Silva Filho
tmfilho@gmail.com/telmo@de.ufpb.br
www.de.ufpb.br

UFPB
Departamento de ESTATÍSTICA