*Universidade Federal do Paraná*

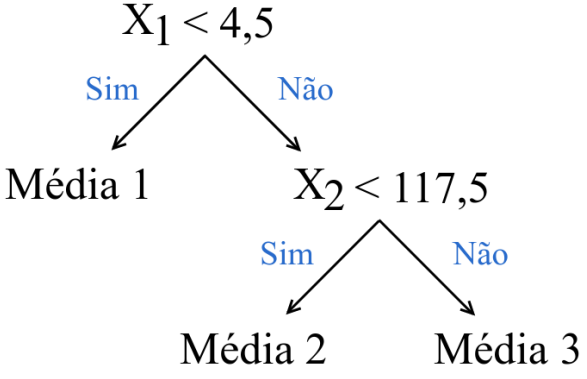
*Laboratório de Estatística e Geoinformação - LEG*

# Métodos baseados em árvores

Eduardo Vargas Ferreira

# Introdução

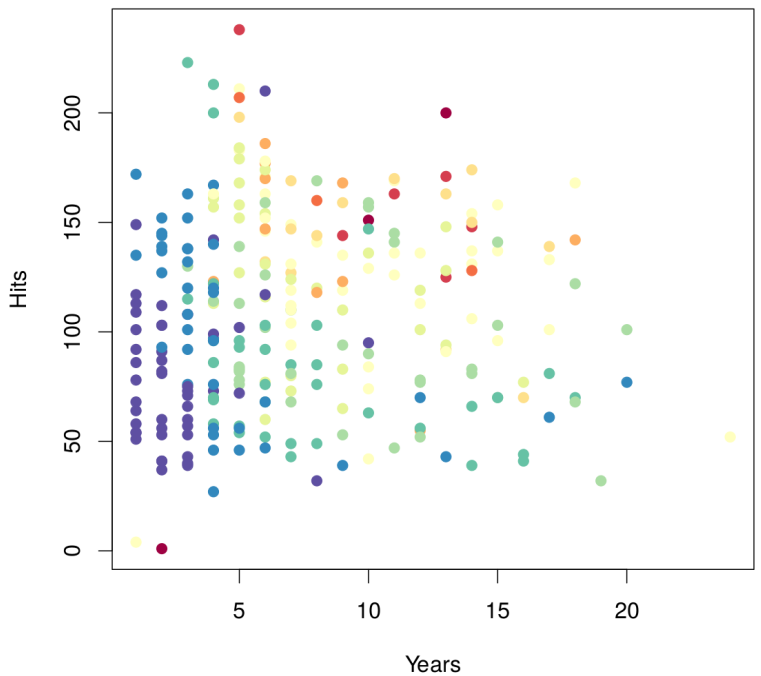
- Árvore de decisão é o conjunto de regras que envolvem a **estratificação** ou **segmentação** do espaço de predição em regiões simples;




- Nesta seção vamos descrever os métodos baseados em **árvores** no contexto de regressão e classificação.

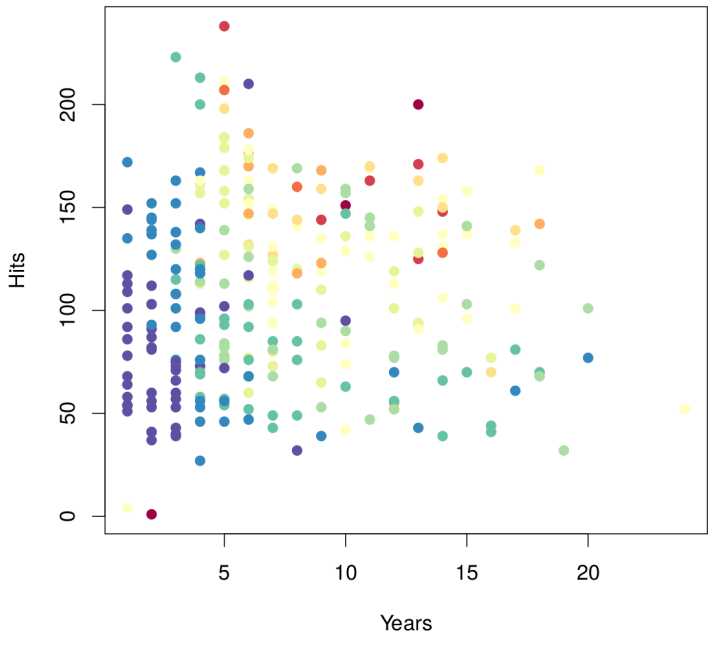
# Exemplo: `Hitters` data set - Baseball salary

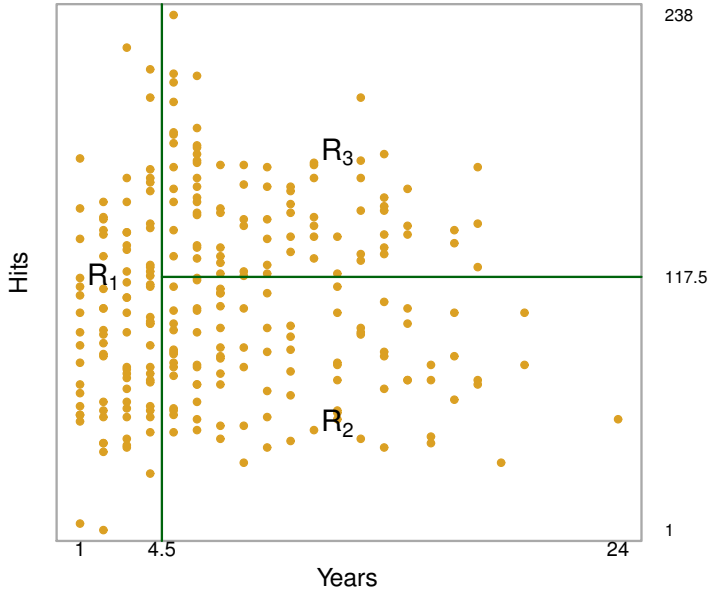
- Queremos prever o `Salary` dos jogadores baseado nos `Years` em que está na *Major leagues* e número de `Hits` no ano;




- Os salários mais baixos são codificados pelas cores azul e verde, e mais altos pelas cores amarelo e vermelho;

# Exemplo: Hitters data set - Baseball salary



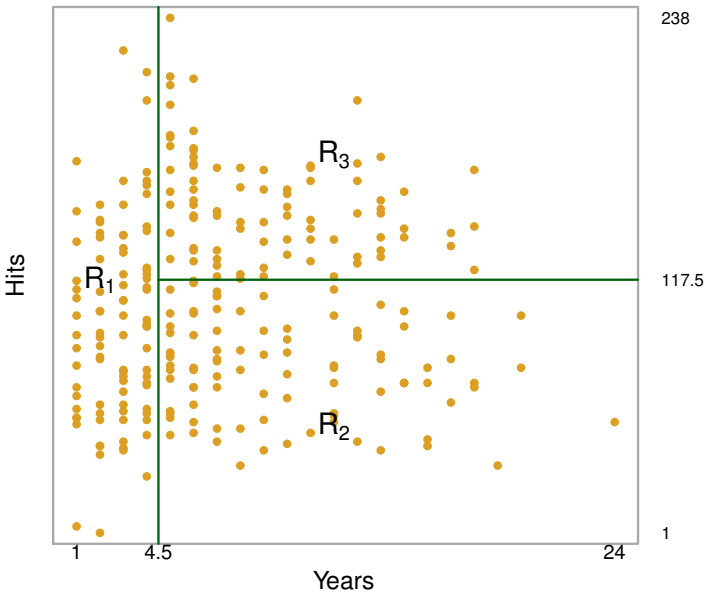




$$R_1 = \{X | \texttt{Years} < 4.5\}$$

$$R_2 = \{X | \texttt{Years} \geq 4.5, \texttt{Hits} < 117.5\}$$

$$R_3 = \{X | \texttt{Years} \geq 4.5, \texttt{Hits} \geq 117.5\}$$

# Exemplo: Hitters data set - Baseball salary





$$R_1 = \{X | \texttt{Years} < 4.5\}$$

$$R_2 = \{X | \texttt{Years} \geq 4.5, \texttt{Hits} < 117.5\}$$

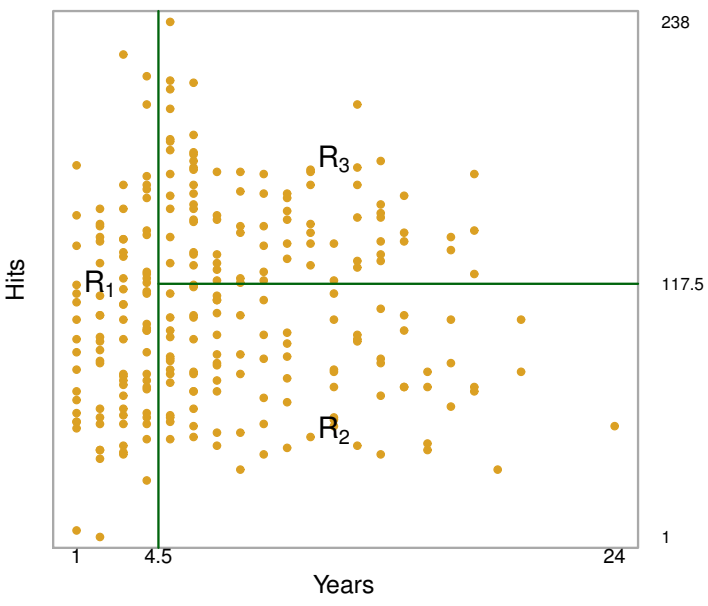
$$R_3 = \{X | \texttt{Years} \geq 4.5, \texttt{Hits} \geq 117.5\}$$

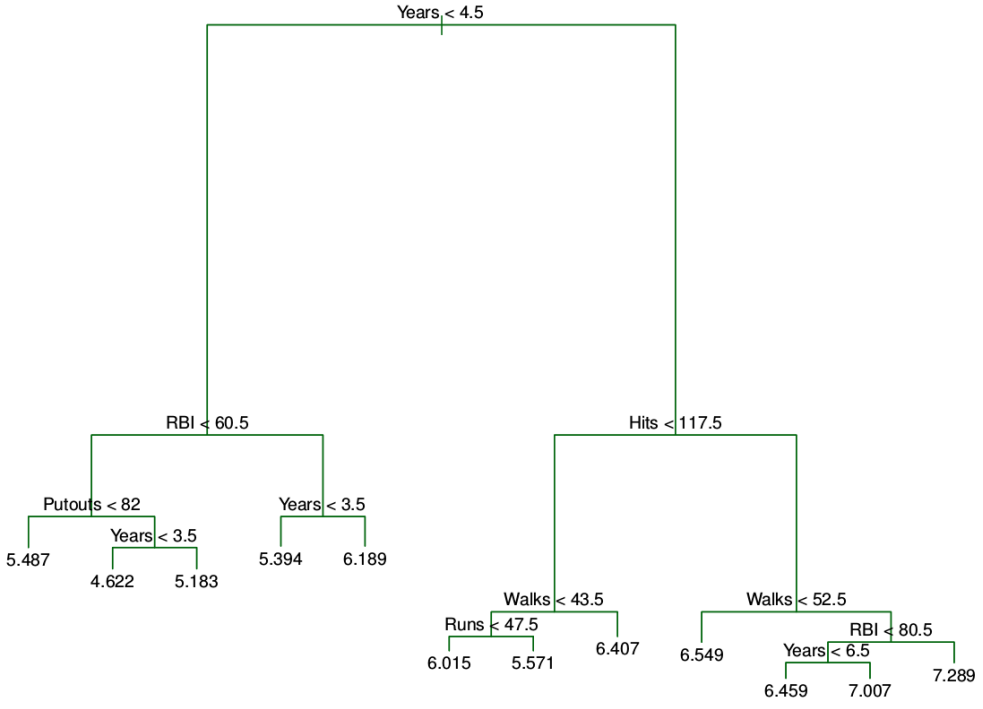
# Como o algoritmo funciona?

- O objetivo é encontrar os retângulos $R_1, \ldots, R_J$ que minimiza a:

$$SQRes = \sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2,$$

em que $R_1(j,s) = \{X|X_j < s\}$ e $R_2(j,s) = \{X|X_j \geq s\}$.

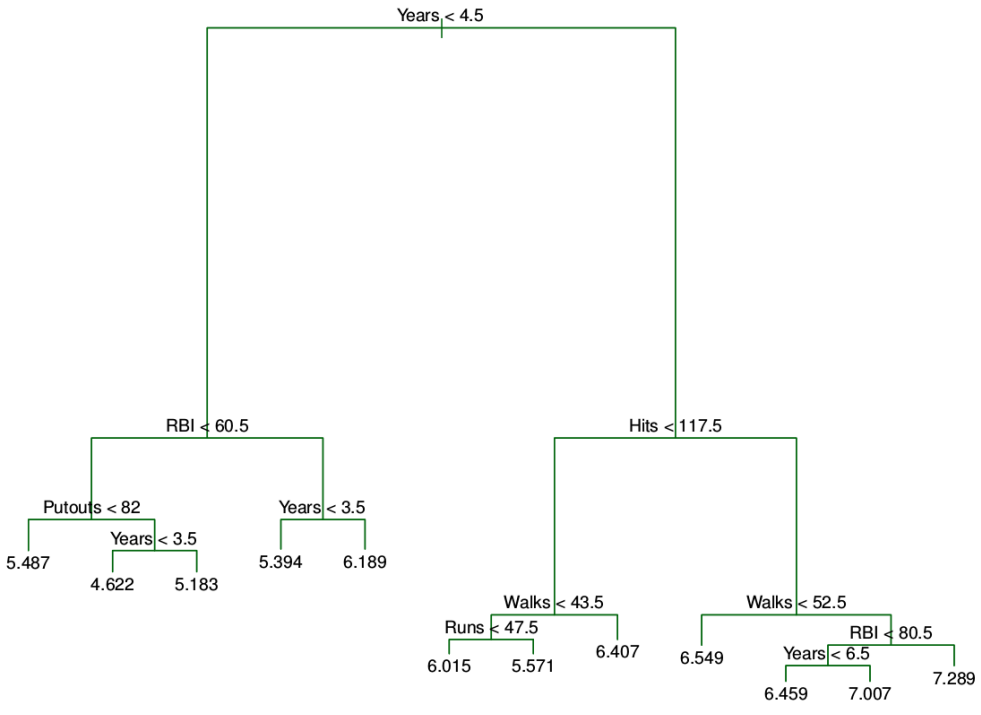


# Como o algoritmo funciona?

## Alto vício e baixa variância


## Baixo vício e alta variância



# Cost complexity pruning

- Para cada valor de $\lambda$, temos uma subárvore $T \subset T_0$, tal que

$$SQRes_\lambda = \sum_{m=1}^{|T|} \sum_{i:x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \lambda |T|$$

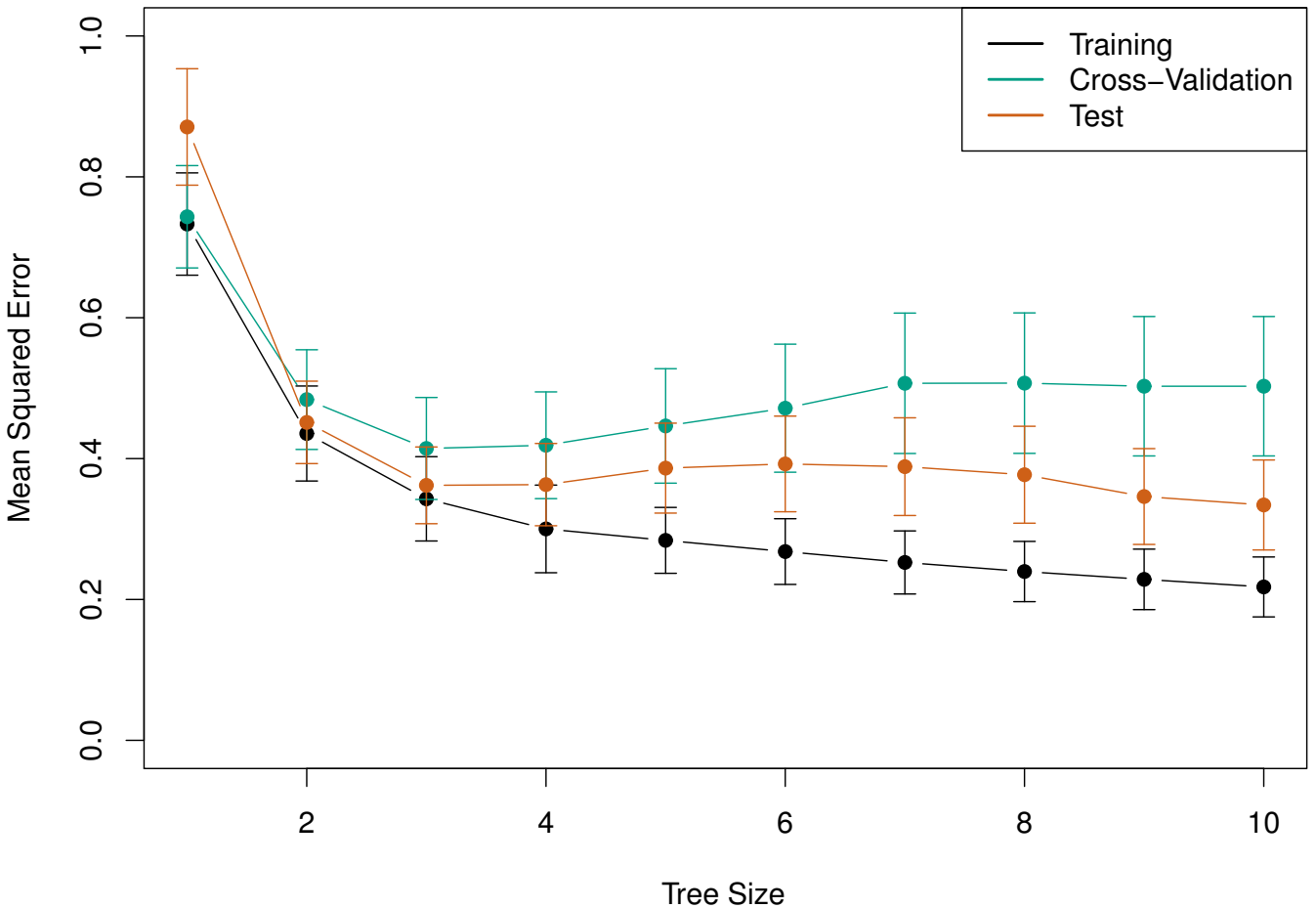
  seja o menor possível.

  - ⋆ $|T|$ indica o número de **terminal nodes** da árvore $T$;
  - ⋆ $R_m$ é o retângulo correspondente ao $m$-ésimo *terminal node*;
  - ⋆ $\hat{y}_{R_m}$ é a média das observações dos dados de treino em $R_m$.

- Selecionamos o valor ótimo, $\hat{\lambda}$, através de validação. Em seguida, obtemos a subárvore utilizando $\hat{\lambda}$.

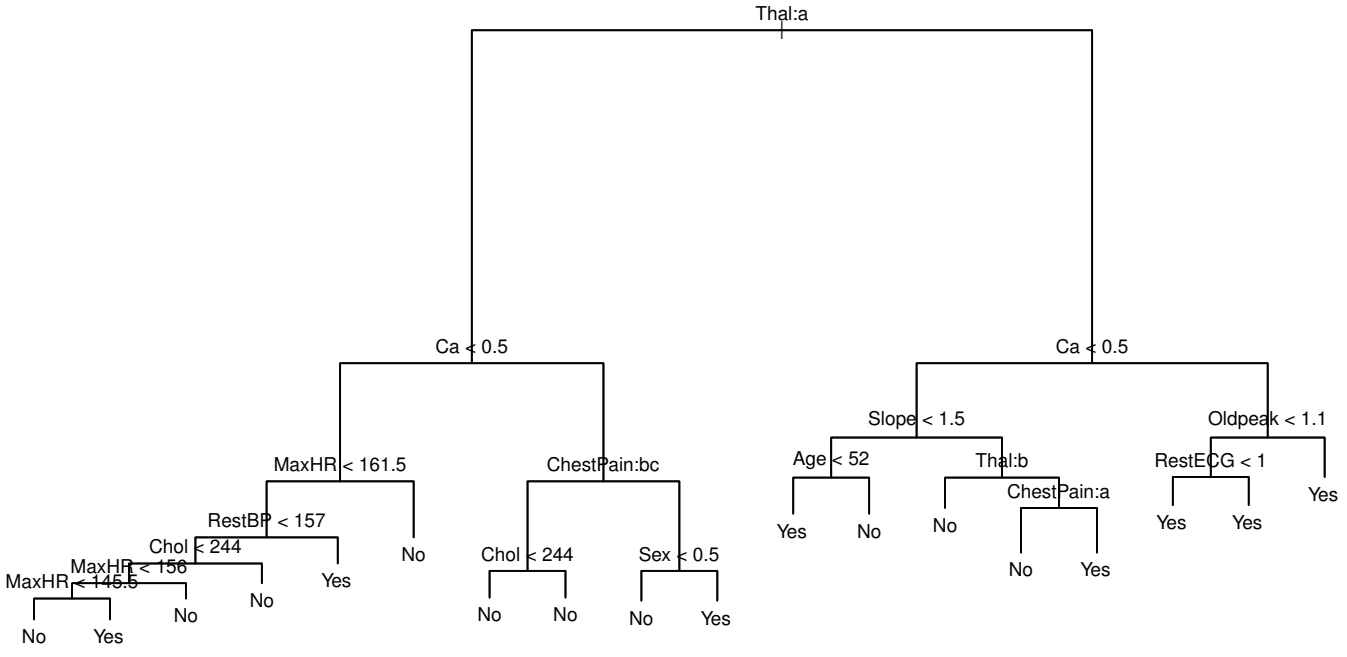
# Exemplo: `Hitters` data set - Baseball salary

- O erro mínimo na validação cruzada ocorre na árvore de tamanho 3.



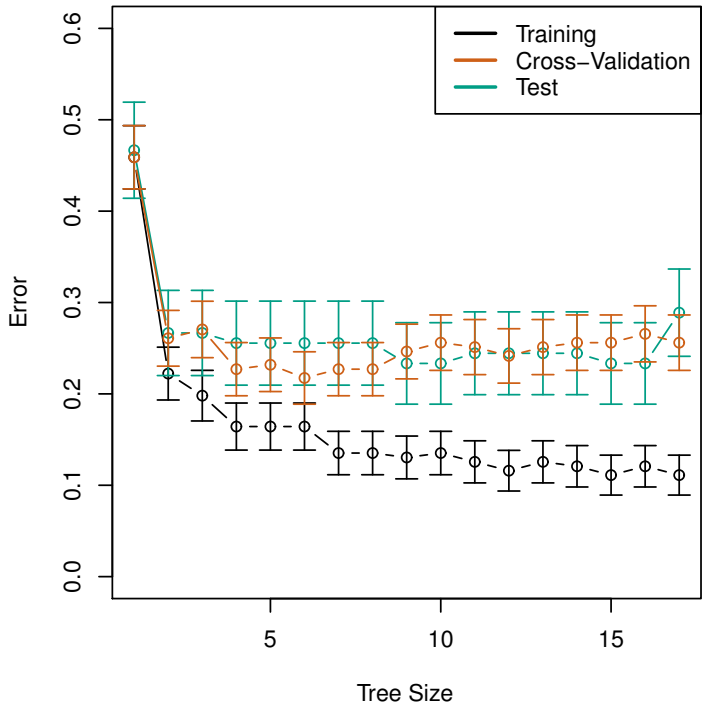
# Exemplo: heart disease - HD

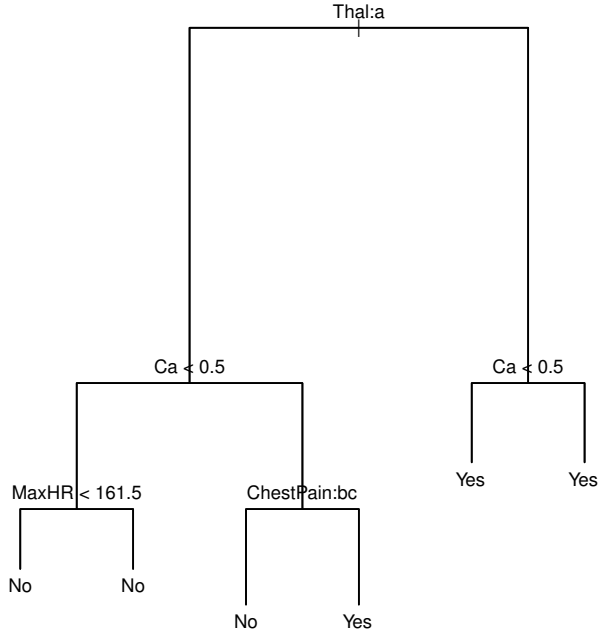
- Os dados contêm o diagnóstico de 303 pacientes com dores no peito:
  - ⋆ `Yes`: indica a presença de doença cardíaca;
  - ⋆ `No`: indica ausência de doença cardíaca;
- Os dados apresentam 13 preditores incluindo `Age`, `Sex`, `Chol`, e outras medidas de funções cardíacas e pulmonar;



# Exemplo: heart disease - HD

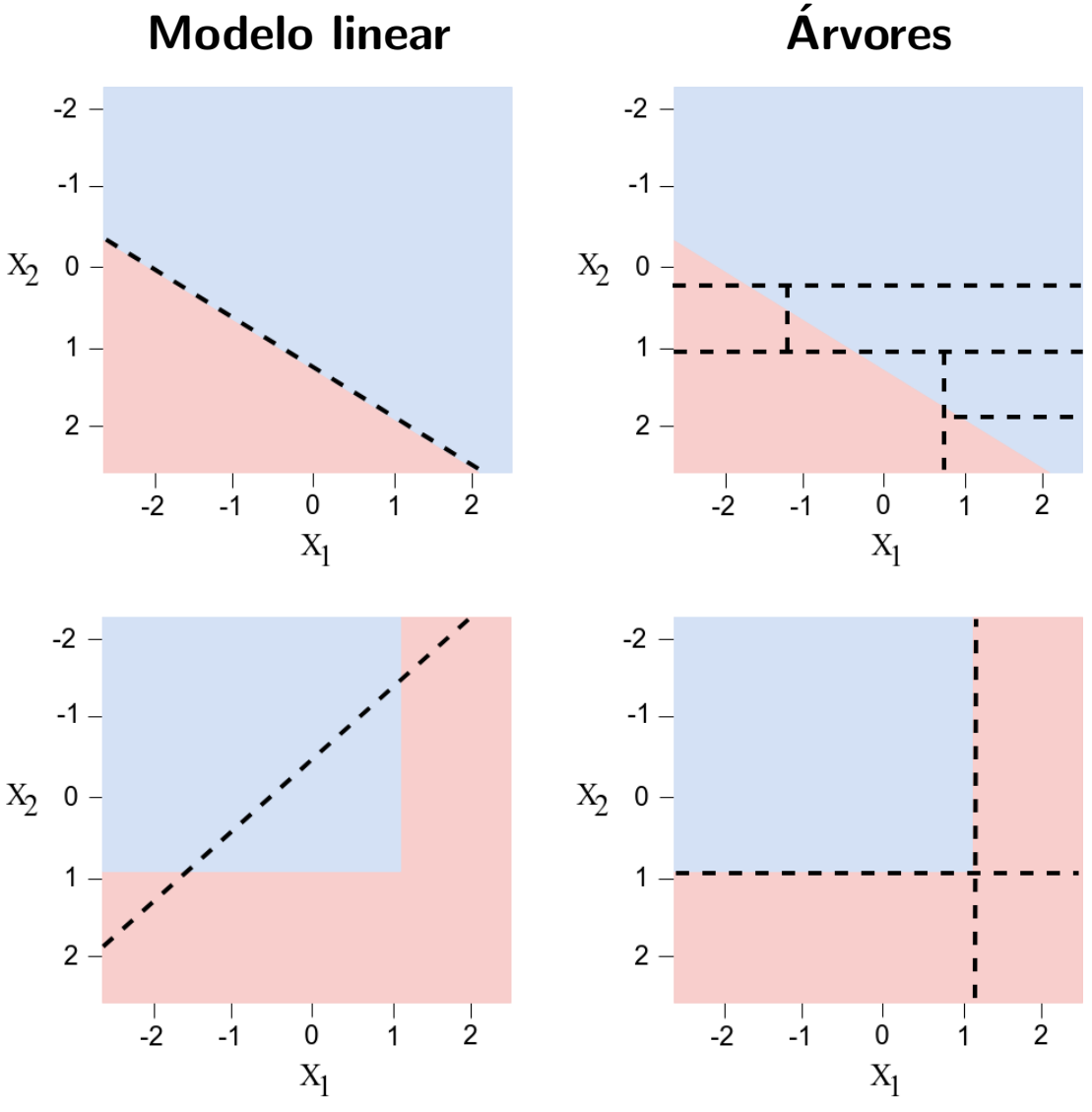
- Após validação cruzada chegamos na árvore com seis *terminal nodes*;




- Note que, em `MaxHR` temos duas respostas `No`. Isto se deve a um dos nós ser "puro" e o outro ser majoritariamente `No`.

# Árvores versus modelos lineares



# Prós e contras das Árvores de decisão

✓ Podem ser aplicadas em problemas de regressão e classificação;

✓ Lidam bem com dados faltantes;

✓ São simples e úteis para interpretação. Sendo muito bons nas etapas iniciais de um projeto;

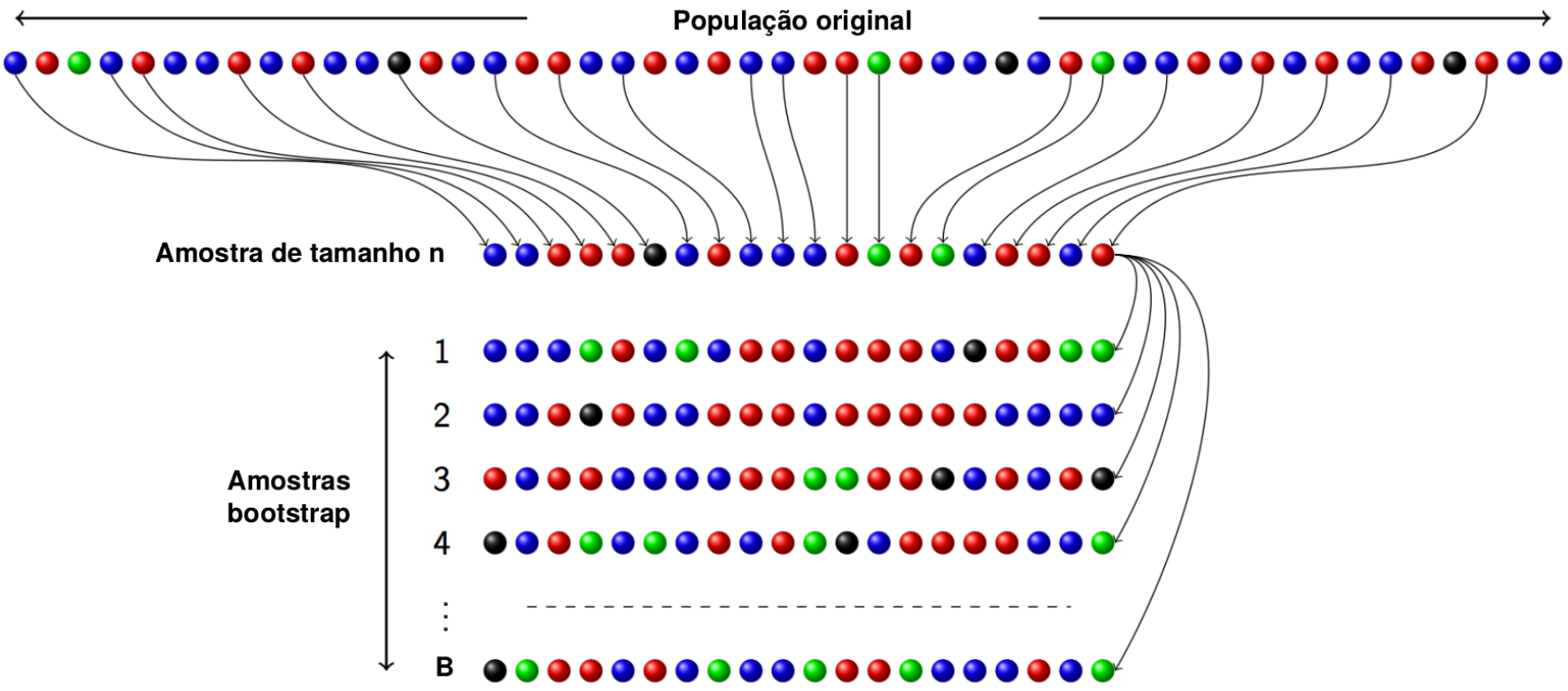
χ São mais simples do que deveriam. Por esse motivo, em termos de predição, não são competitivos com outras abordagens de aprendizado supervisionado;

✓ Mas, serve de base para outros métodos, como:

- Bagging;
- Random Forests;
- Boosting.

# Ensembles

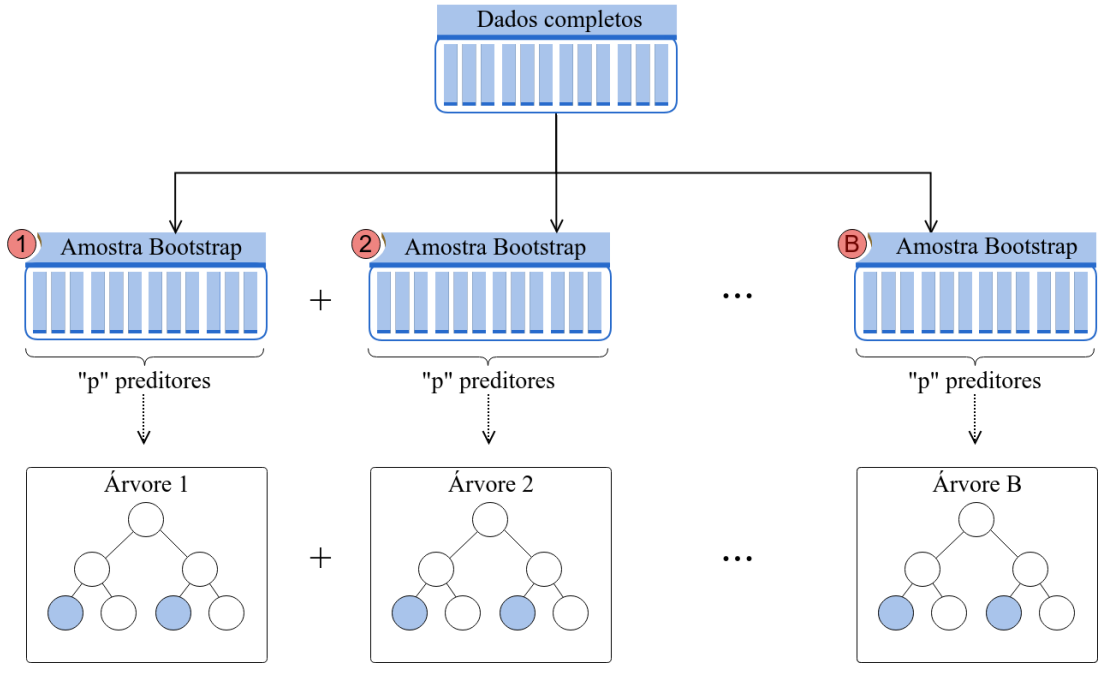
# Ideia do Bootstrap



# Bagging

# Bagging

- Geramos $B$ conjuntos de observações (*bootstrapped*). Treinamos o modelo a fim de obter a predição no ponto $x$;



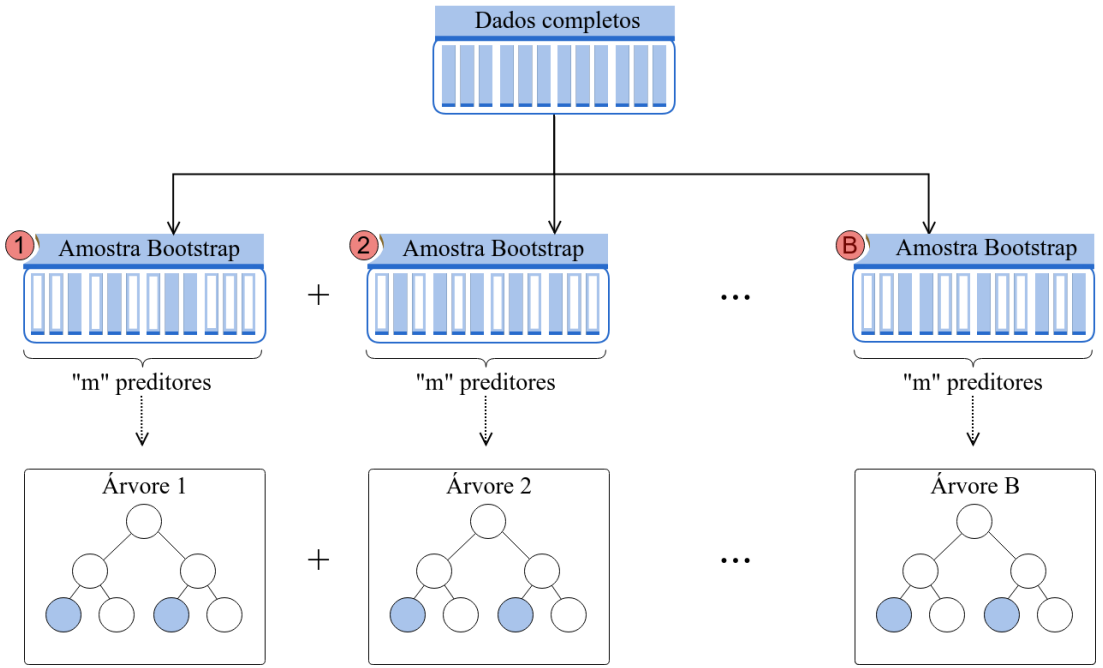

- Em seguida, calculamos a média das predições (chamamos de **bagging**):

$$\hat{h}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{h}^b(x).$$

# Random Forests

# Random Forests

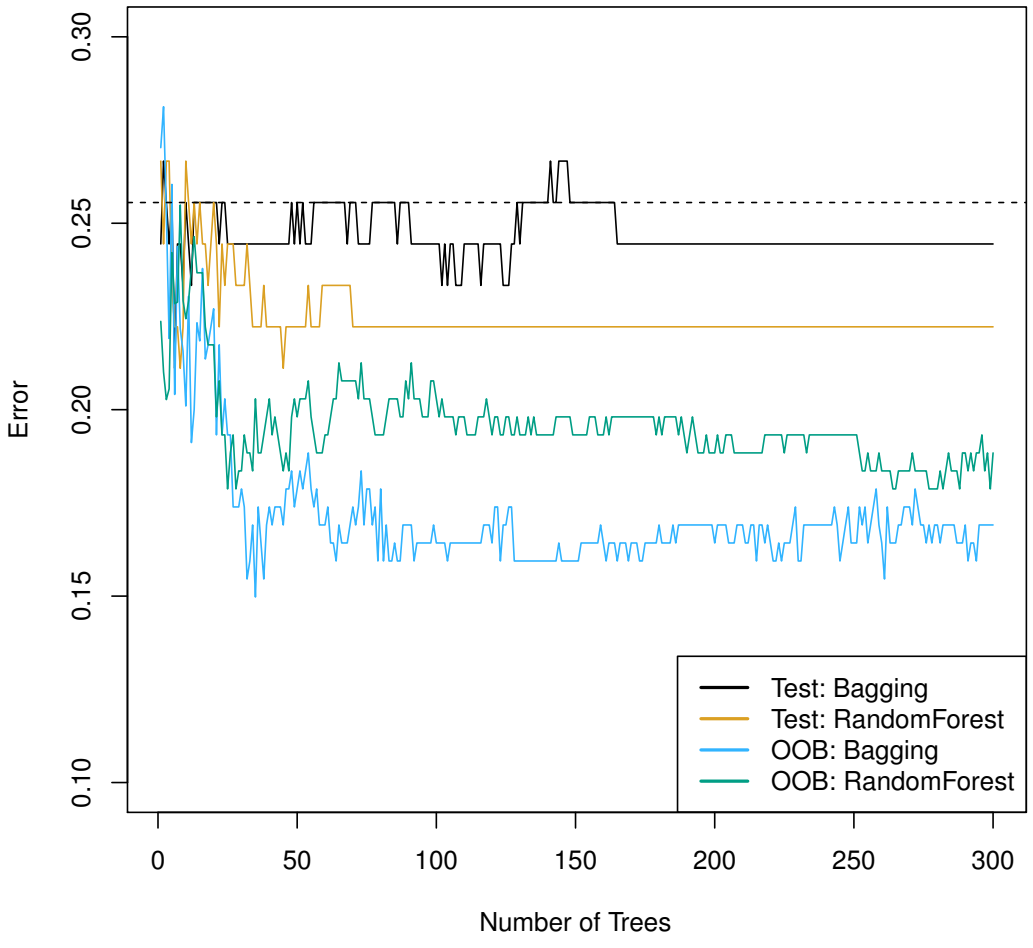
- No Random forests, para cada partição, temos uma **seleção aleatória de $m$ preditores**, de um total de $p$ (tipicamente, $m \approx \sqrt{p}$);



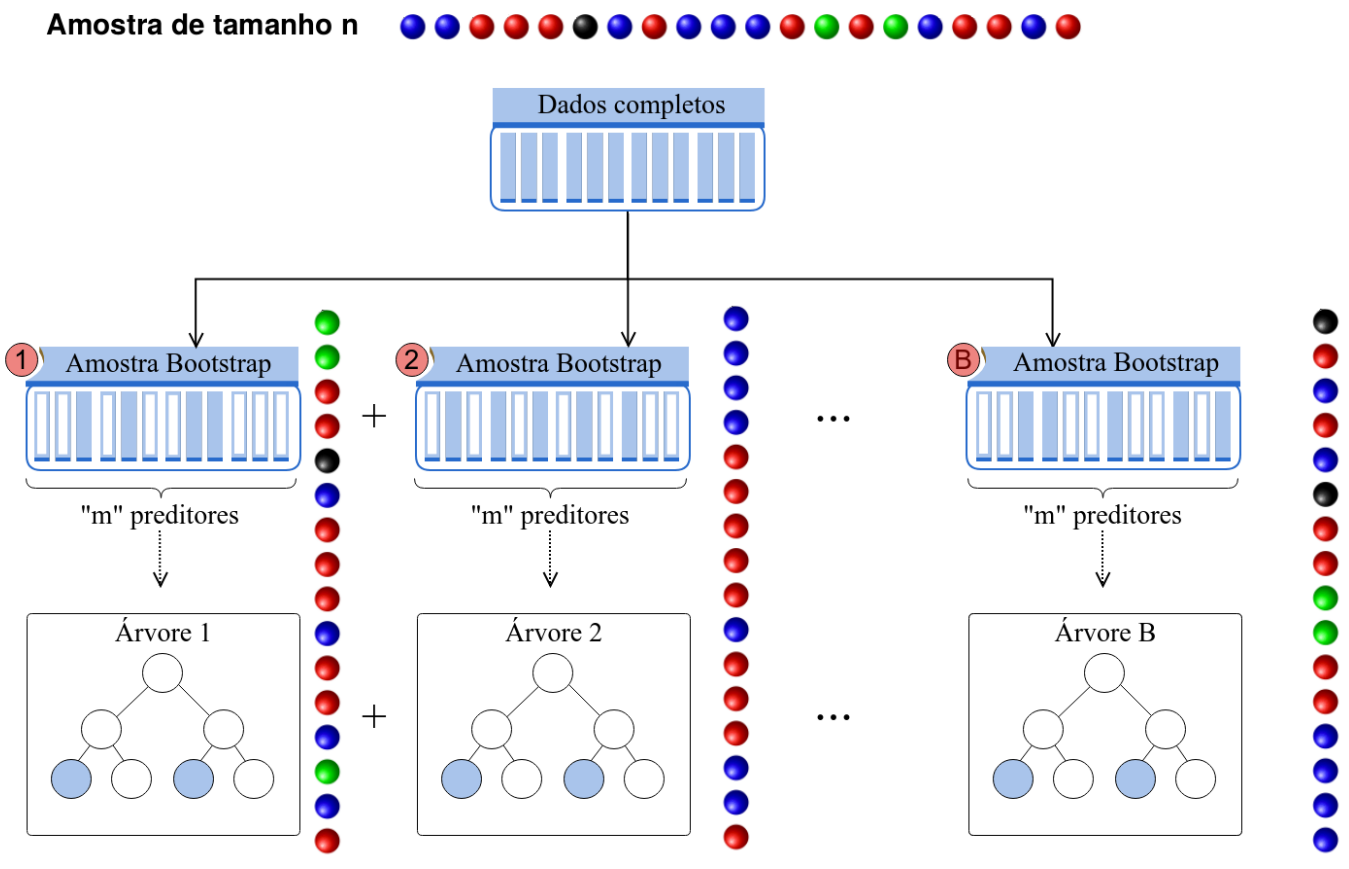

- Assim, forçamos com que diferentes preditores sejam escolhidos (*decorrelating the trees*). Se $m = p$, estaremos no método *Bagging*.

# Exemplo: `Heart` data set

- Abaixo, o erro do teste como função de $B$. A linha tracejada representa o erro utilizando uma árvore somente;

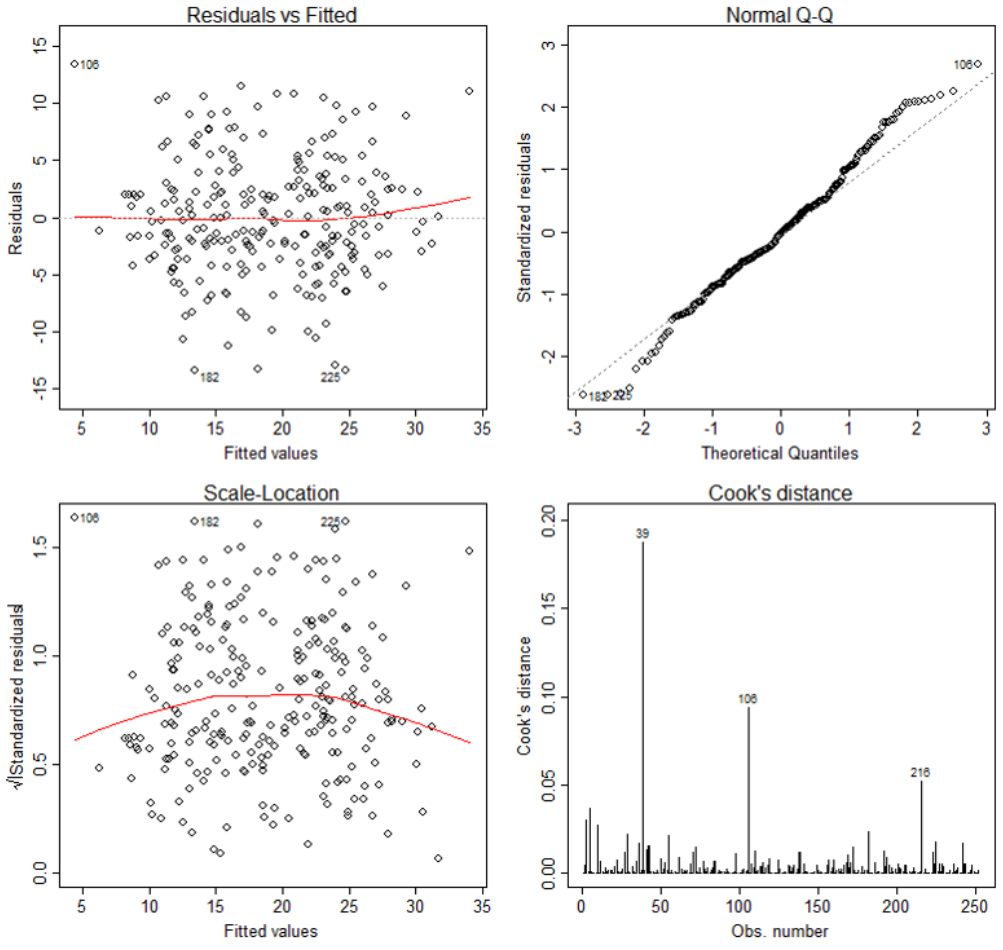


# Out-of-Bag Error Estimation



# Métodos boosting

# Qual a importância dos resíduos?

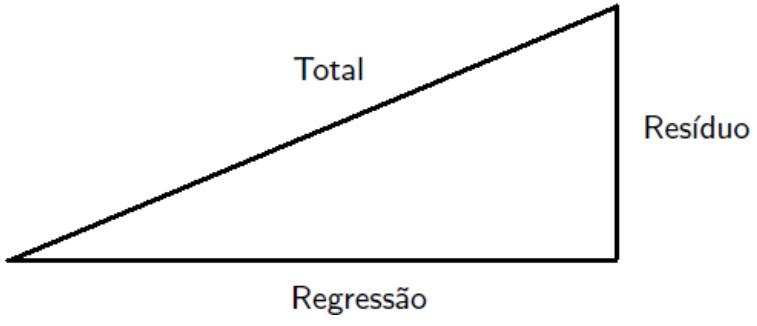


# Qual a importância dos resíduos?

- Lembrando de regressão

$$\underbrace{\sum_{i=1}^{n}(y_i-\bar{y})^2}_{SQT}=\underbrace{\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2}_{SQR}+\underbrace{\sum_{i=1}^{n}(y_i-\hat{y})^2}_{SQE}.$$

- E, geometricamente, temos

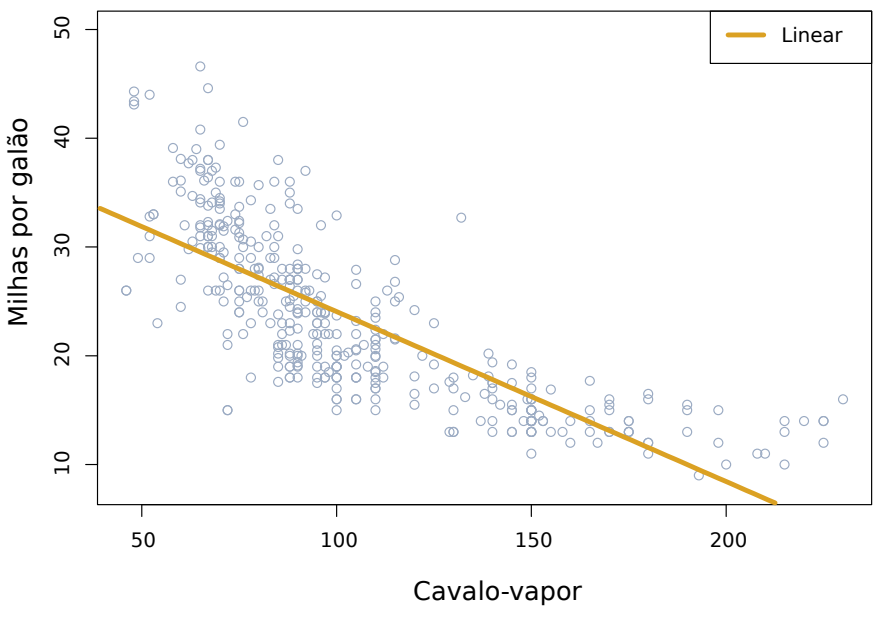


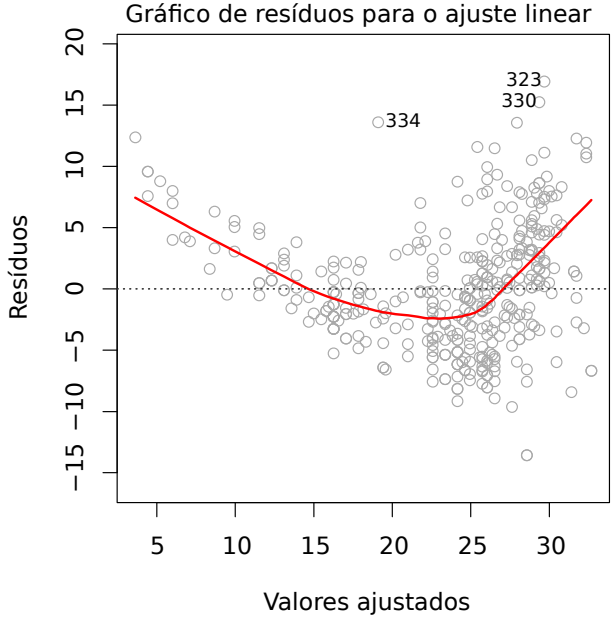

- Isso quer dizer que toda variabilidade **não explicada** pela regressão ficará no resíduo (variáveis e funções delas!).

# Qual a importância dos resíduos?

- No exemplo abaixo, estamos avaliando a relação entre consumo de combustível e potência do automóvel.

$$mpg = \beta_0 + \beta_1 \times (\text{cavalo vapor}) + \varepsilon$$

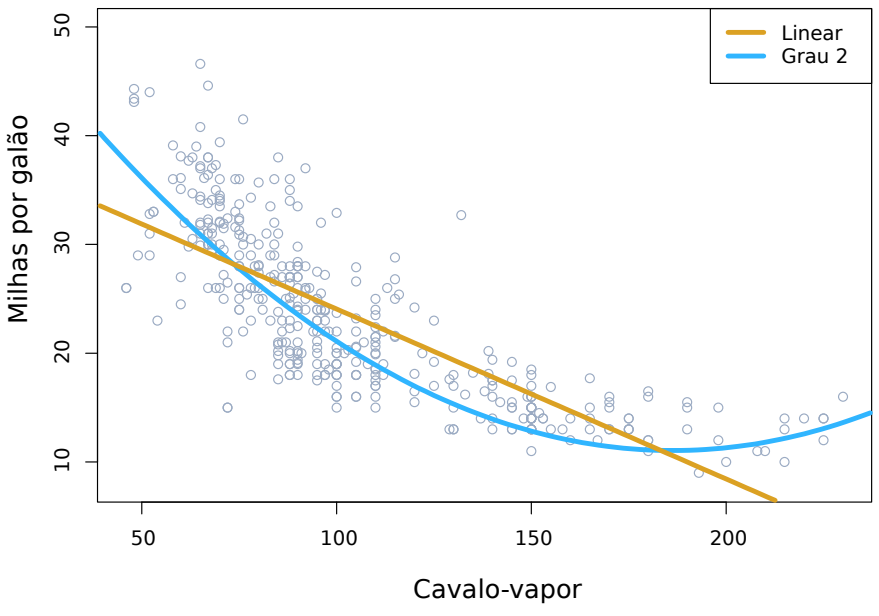


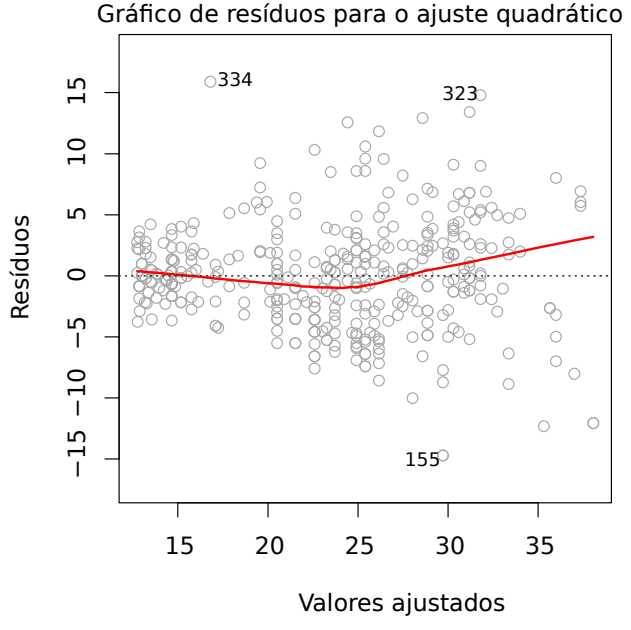



# Qual a importância dos resíduos?

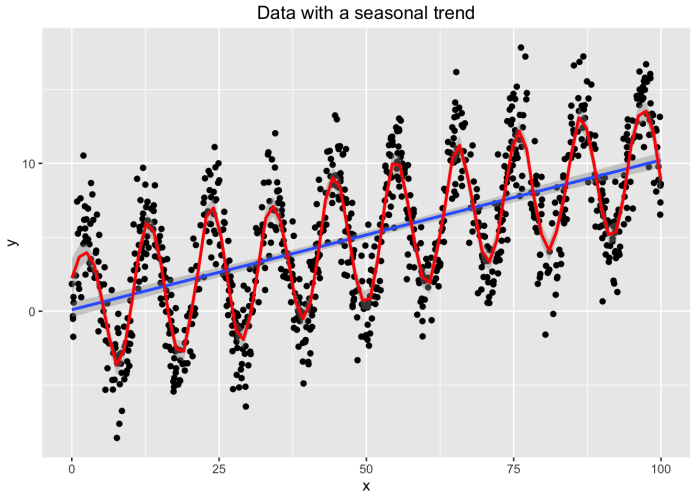
- No exemplo abaixo, estamos avaliando a relação entre consumo de combustível e potência do automóvel.

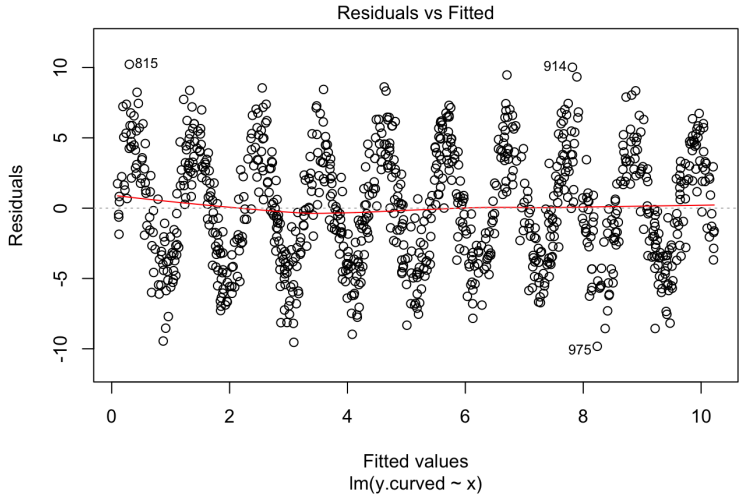
$$mpg = \beta_0 + \beta_1 \times (\text{cavalo vapor}) + \beta_2 \times (\text{cavalo vapor})^2 + \varepsilon$$

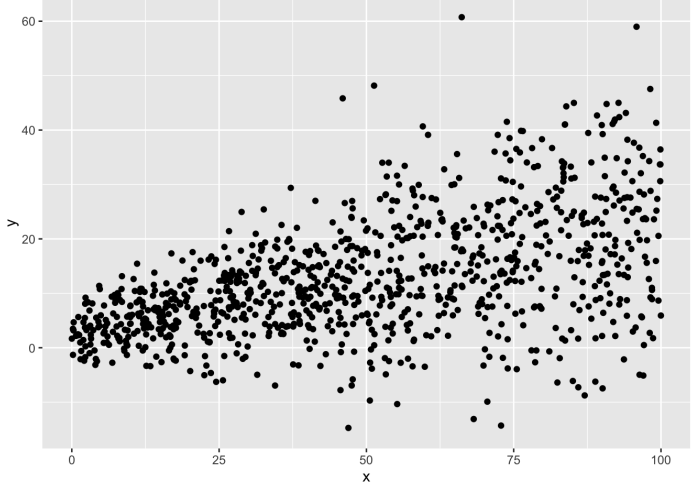


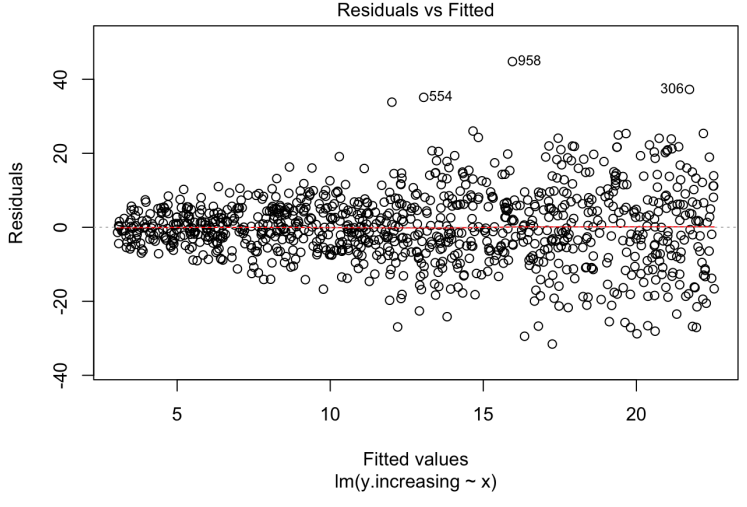



# Qual a importância dos resíduos?



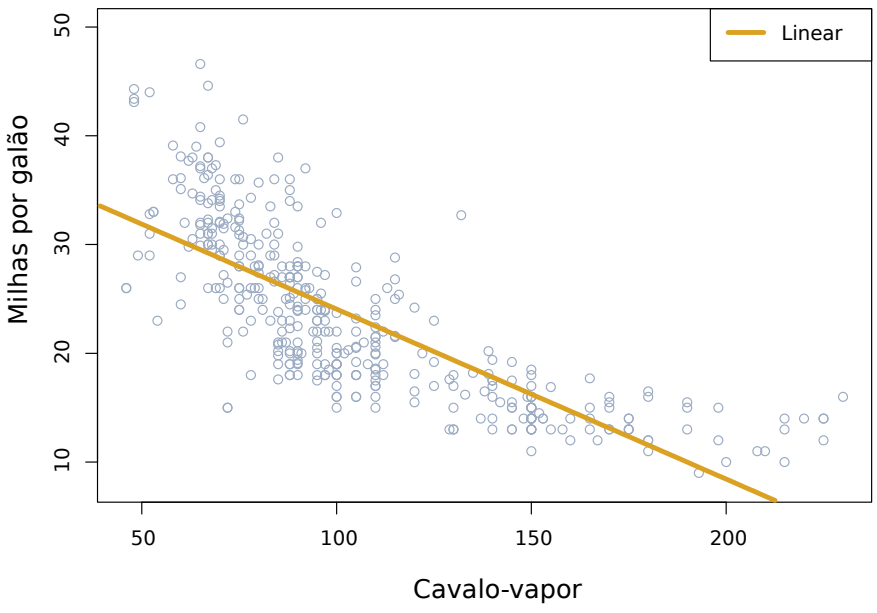



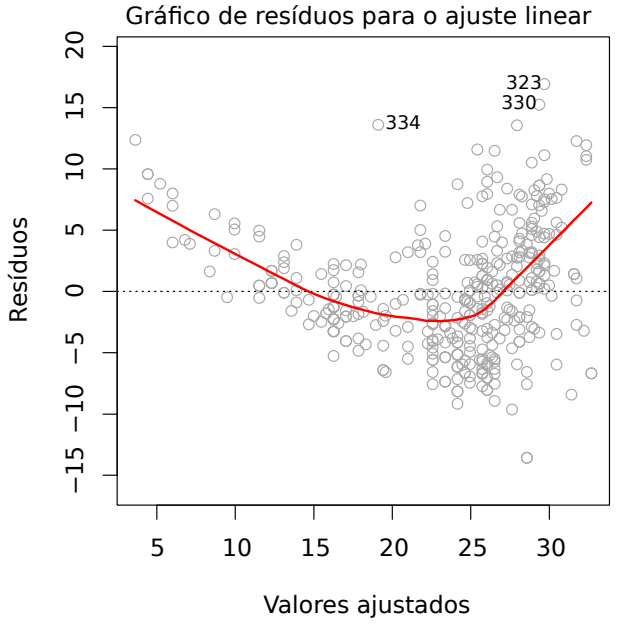



# Métodos boosting

$$mpg = \beta_0 + \beta_1 \times (\text{cavalo vapor}) + residuo$$



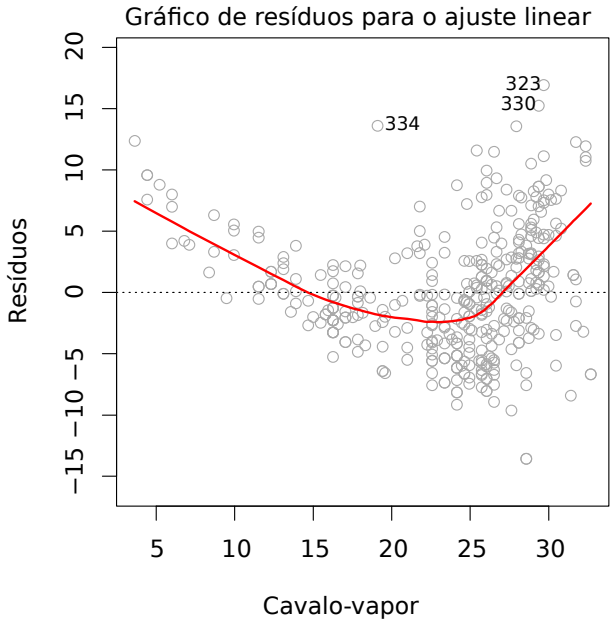



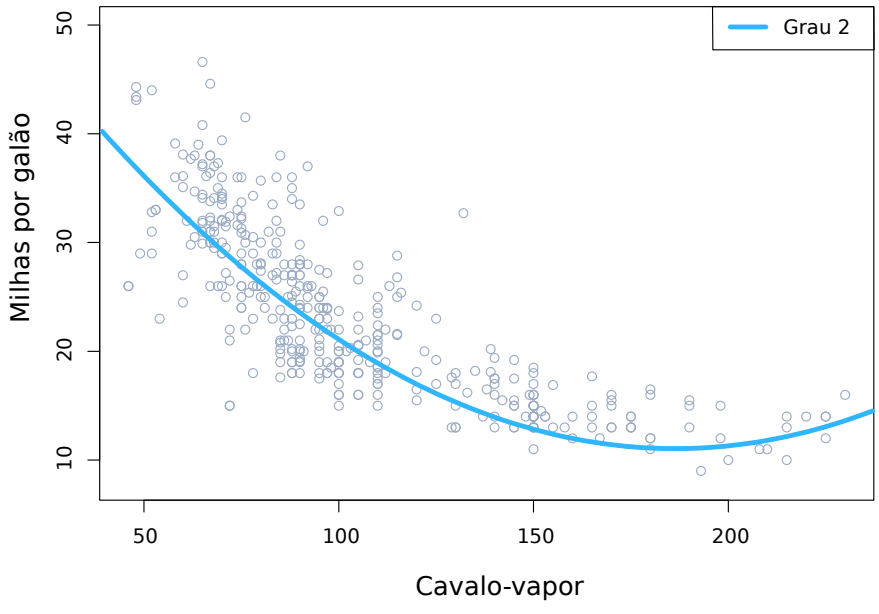

$$residuo = \beta_2 \times (\text{cavalo vapor})^2 + residuo2$$

$$mpg = \beta_0 + \beta_1 \times (\text{cavalo vapor}) + \beta_2 \times (\text{cavalo vapor})^2 + residuo2$$

# Métodos boosting

$$\textit{mpg} = \beta_0 + \beta_1 \times (\text{cavalo vapor}) + \textit{residuo}$$




$$\textit{residuo} = \beta_2 \times (\text{cavalo vapor})^2 + \textit{residuo2}$$

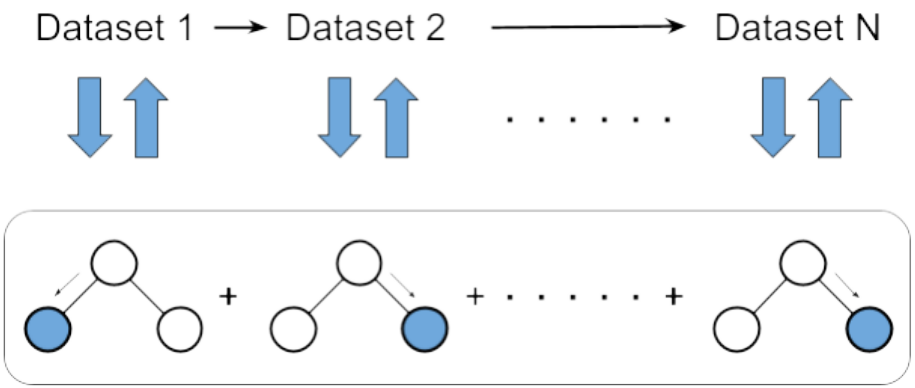
$$\textit{mpg} = \beta_0 + \beta_1 \times (\text{cavalo vapor}) + \beta_2 \times (\text{cavalo vapor})^2 + \textit{residuo2}$$

# Adaptive Boosting (AdaBoost)

# Adaptive Boosting (AdaBoost)

- O princípio básico do Boosting é propor um modelo básico (*weak learner*) e o aprimorá-lo em cada iteração.

- O processo consiste em filtrar os resultados corretos, e concentrar-se naqueles que o modelo não soube lidar;




- Nesse caso, os weak learners, são árvores de decisão com uma separação apenas (chamada de decision stumps).

# Como o algoritmo funciona?

1. Inicie com $\hat{h}(x) = 0$ e $r_i = y_i$, para todo $i$ dos dados de treino;

2. Para $b = 1, 2, \ldots, B$, repita:

   a) Ajuste a árvore $\hat{h}^b$ com $d$ divisões para os dados de treino $(X, r)$;

   b) Atualize $\hat{h}$ adicionando uma nova versão à árvore anterior:

$$\hat{h}(x) \leftarrow \hat{h}(x) + \alpha \hat{h}^b(x).$$

   c) Atualize os resíduos,

$$r_i \leftarrow r_i - \alpha \hat{h}^b(x_i).$$

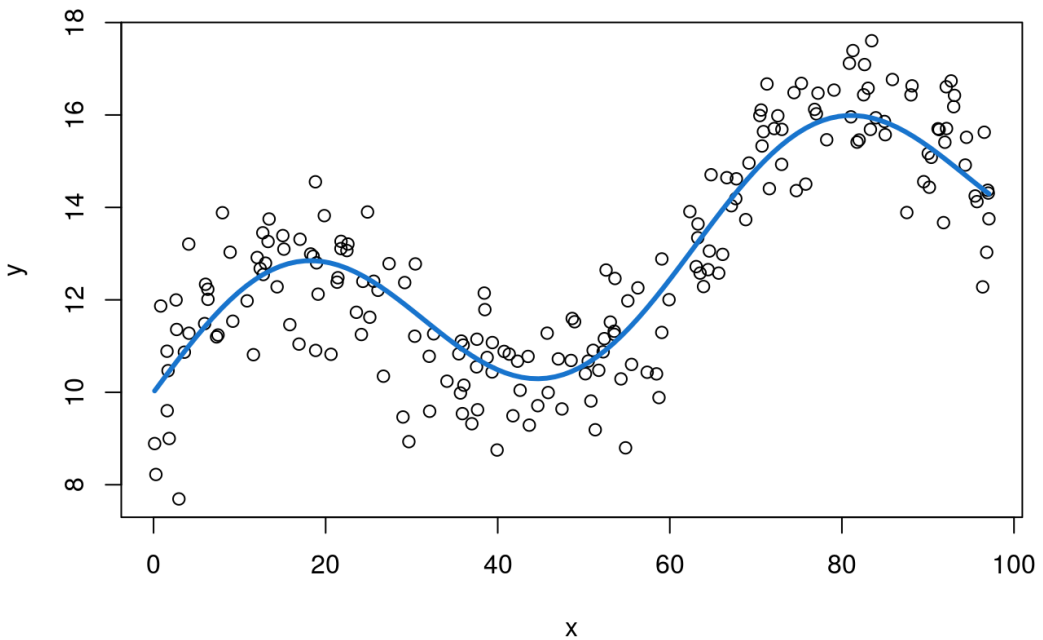
3. O modelo de saída fica então,

$$\hat{h}(x) = \sum_{b=1}^{B} \alpha \hat{h}^b(x).$$

# Exemplo simulado

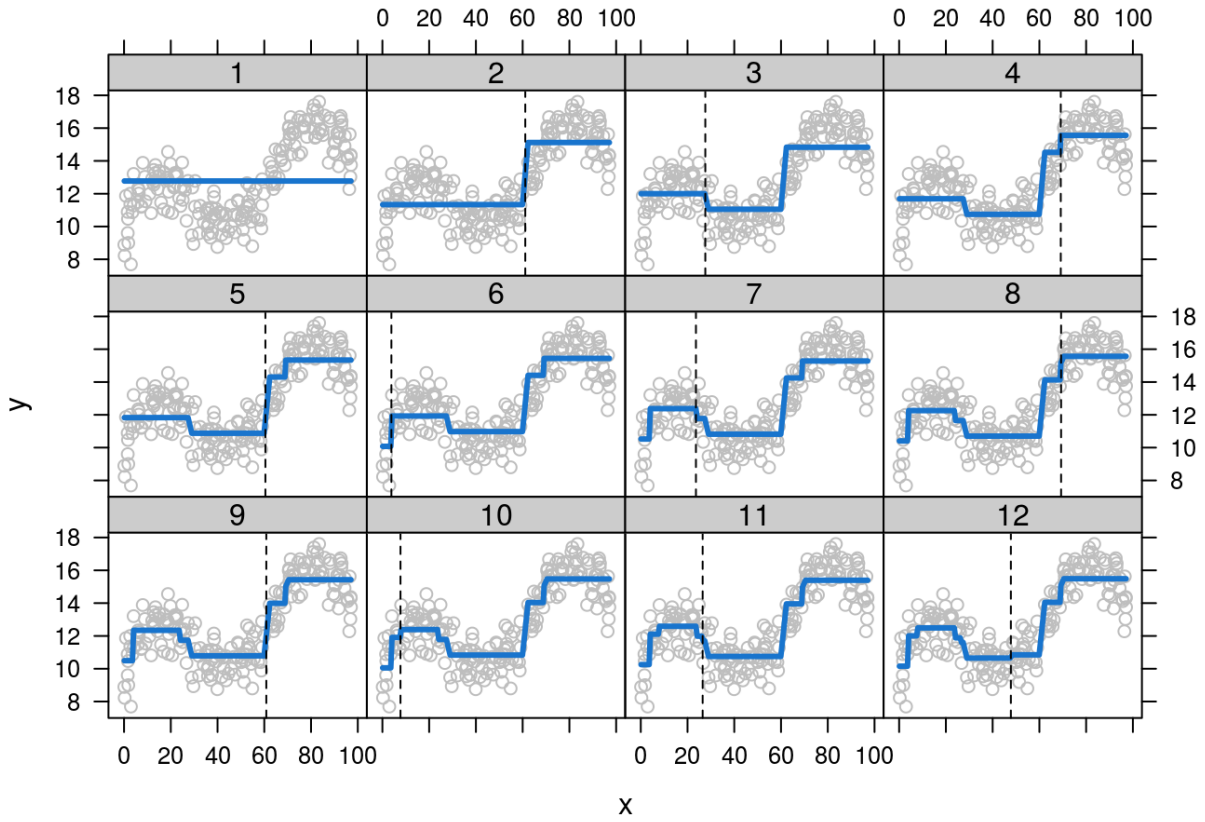
- Considere o exemplo simulado, obtido a partir da função:

$$f(x) = 10 + 0,05x + 2sin\left(\frac{x}{10}\right)$$



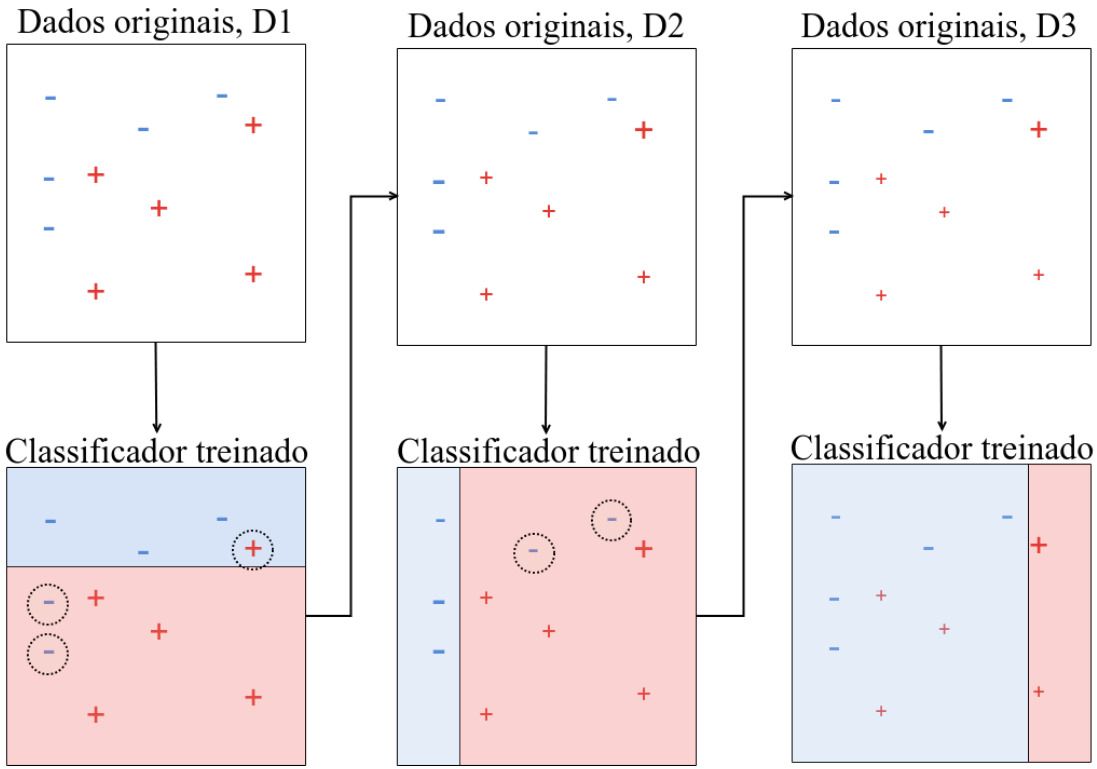
# Exemplo simulado

- O processo consiste em analisar o resíduo decorrente do modelo anterior e somar novas árvores, suprindo tais deficiências.



# Adaptive Boosting (AdaBoost)

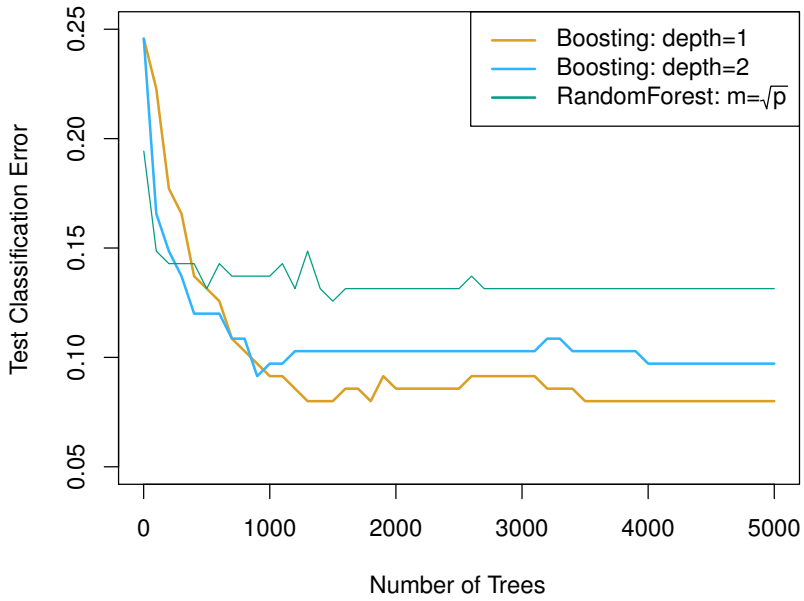
- A ideia é ponderar os erros para que nas próximas árvores eles tenham mais importância. Em seguida, combinar os classificadores.




- Não vamos entrar em detalhes teóricos da abordagem. Para o aluno interessado sugere-se **Elements of Statistical Learning, capítulo 10**.

# Exemplo: Expressão gênica

- Os dados consistem na medida de expressão de 4.718 genes dos tecidos de 349 pacientes;




- Cada paciente possui um marcador qualitativo (de 15 níveis):
  - ⋆ Normal;
  - ⋆ Ou 14 tipos de câncer;

- A forma de construção do boosting (baseado nas árvores anteriores) faz com que ele faça um bom trabalho mesmo com uma partição apenas.

# Referências


- James, G., Witten, D., Hastie, T. e Tibshirani, An Introduction to Statistical Learning, 2013;
- Hastie, T., Tibshirani, R. e Friedman, J., The Elements of Statistical Learning, 2009;
- Lantz, B., Machine Learning with R, Packt Publishing, 2013;
- Tan, Steinbach, and Kumar, Introduction to Data Mining, Addison-Wesley, 2005;
- Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R"(Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani