



# **Data exploration and enrichment for supervised classification**

Trabalho No 2

Carolina Proença -  
up202306055  
Júlia Peixoto - up202306602  
Maria Morais - up202304201

# Pré-processamento de dados

**Começamos por encontrar os valores em falta (“?”)**

**Substituímos valores “None” por “No” para não ser confundido com valores em falta (NaN)**

**Como não encontramos nenhum atributo com mais de 50% de valores em falta, não eliminamos nenhum.**

**Atributos numéricos:** Em colunas de atributos numéricos, como a idade, usamos a mediana para imputar os valores em falta.

**Atributos binários:** Em colunas No/Yes, Mild/Severe e Gradel/II / Grade III/IV, substituímos os valores por 1/0, respetivamente, e imputamos os valores em falta com a moda.  
Na variável PS, atribuímos um número de 0 a 4, de acordo com o estado do paciente.

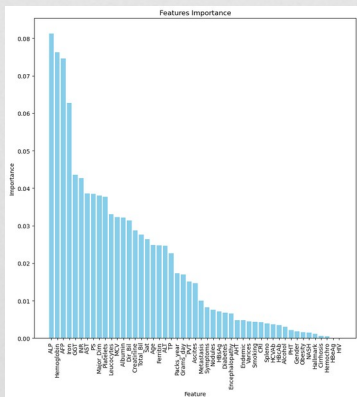
# Qualidade das variáveis

Usámos 3 métodos para definir a qualidade das variáveis, criando um sistema de pontos com a combinação dos 3.

## 1 - Feature Importance e decision trees

Atributo mais importante: ALP

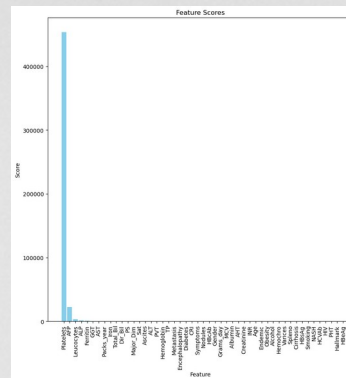
1



## 3 - Seleção de características com SelektBest

Atributo mais importante:  
Platelets

3



# Qualidade das variáveis

Usámos 3 métodos para definir a qualidade das variáveis, criando um sistema de pontos com a combinação dos 3.

## 2 - Seleção de Características com Recursive Feature Elimination

Atributo mais importante: PVT

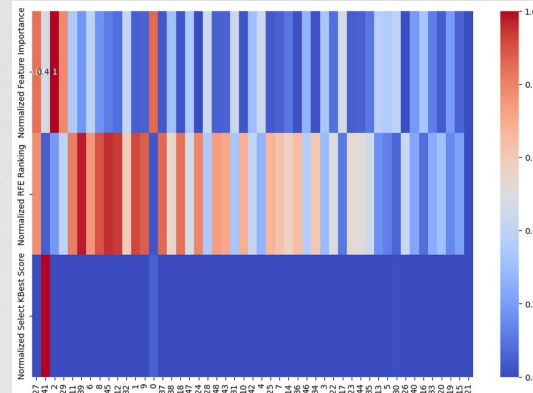
2

Ao analisar o heat map, a variável 27 (hemoglobina) é a única que apresenta 2 colunas de cor quente em diferentes setores, daí ser a mais bem classificada no sistema de pontos.

## 4 - Combinação de todo

Atributo mais importante: Hemoglobin

4





# Qualidade das variáveis

Analizamos também gráficos de todas as variáveis em relação à variável objetivo, com o intuito de avaliar melhor a influência destas na sobrevivência dos pacientes.

Baseamos a nossa escolha de más variáveis em 2 parâmetros:

**Se a presença da variável comparativamente à sua falta no dataset for inferior a 10%.**

**Se a presença ou falta da variável for substancialmente pequena comparativamente ao número de pacientes.**

**Symptoms and Diabetes - a distinção entre pacientes que vivem ou morrem é inferior a 10%**

**HbeAg, HIV - Apenas 1 ou 3 pacientes, respetivamente, de 165 têm este atributo a 1, então não é uma variável boa de analisar**

**Endemic, Hemochromatosis, Nash - Apenas 10, 7 ou 8 pacientes, respetivamente, de 165 têm este atributo a 1, então não é uma variável boa de analisar**

**HCVab, Grams\_Day, INR, TP and Total\_Bil - Não há diferença significativa nos pacientes que vivem e morrem.**

# Modelos que usamos - Comparação

Usamos 5 métodos para prever a sobrevivência dos pacientes

O train/test split foi constante igual a 70/30%

Decision  
tree

KNN

Logistic  
Regression

SVM

Gradient  
Boosting

Para um teste mais preciso dos modelos acima, combinamos diferentes variáveis em cada caso abaixo enumerado.

- 1 Todas as variáveis
- 2 Sem as variáveis onde com ou sem esse parâmetro existia menos de 10% das pessoas
- 3 Sem as variáveis consideradas como más a partir da leitura dos gráficos
- 4 Sem as piores 5 variáveis de acordo com o sistema de pontos

# Modelos que usamos - Comparação

- 5 Sem as melhores 3 variáveis, de acordo com o sistema de pontos
- 6 Sem as melhores 5 variáveis de acordo com o sistema de pontos
- 7 Apenas com as melhores 3 variáveis, de acordo com o sistema de pontos
- 8 Apenas com as melhores 5 variáveis, de acordo com o sistema de pontos
- 9 Apenas com as melhores 10 variáveis, de acordo com o sistema de pontos
- 10 Apenas com as variáveis consideradas más, de acordo com a análise dos gráficos

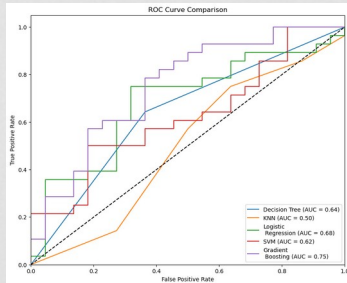
## Bibliografia

<https://radiopaedia.org/articles/ecog-performance-status>

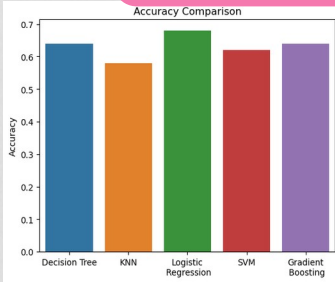
<https://www.analyticsvidhya.com/blog/2021/04/how-to-handle-missing-values-of-categorical-variables/>

# Modelos que usamos - Comparação

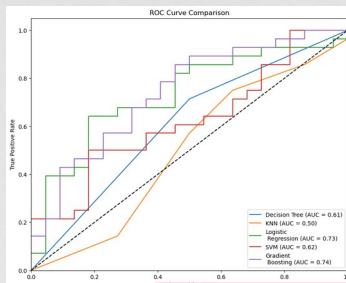
1



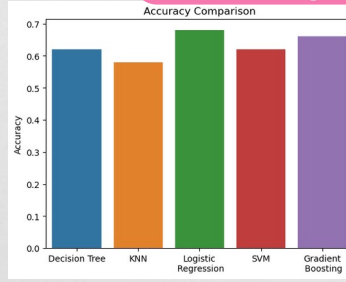
Melhor: logistic regression



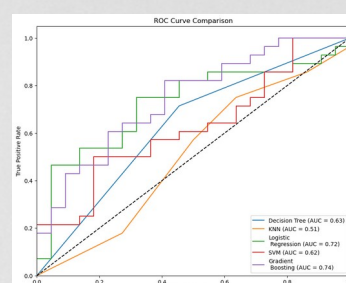
2



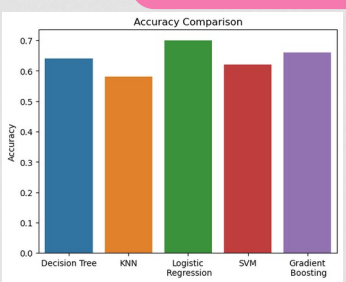
Melhor: Gradient Boosting



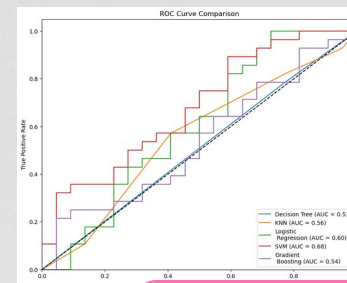
3



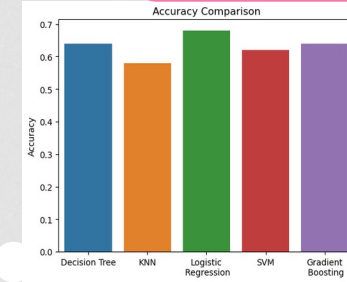
Melhor: Gradient Boosting



8



Melhor: Gradient Boosting

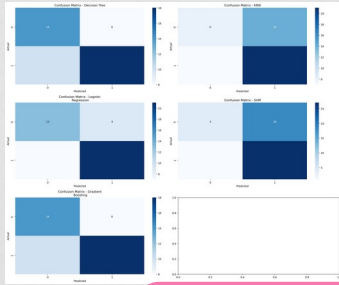


Usamos a curva ROC/AUC e gráfico de barras para avaliar a eficácia dos modelos

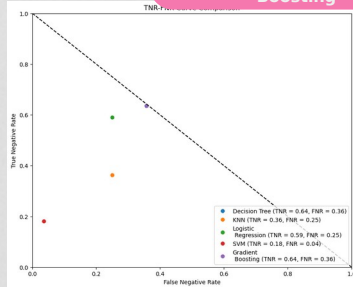


# Modelos que usamos - Comparação

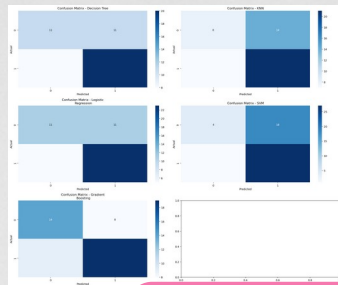
1



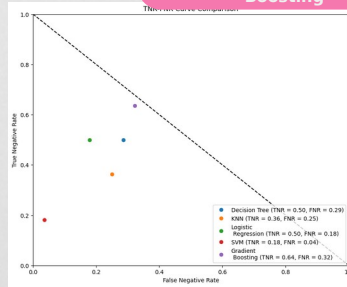
Melhor: Gradient Boosting



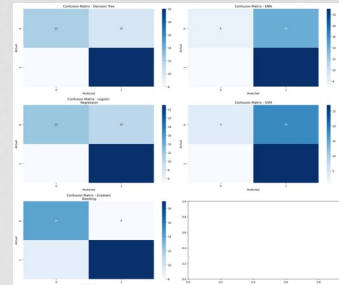
2



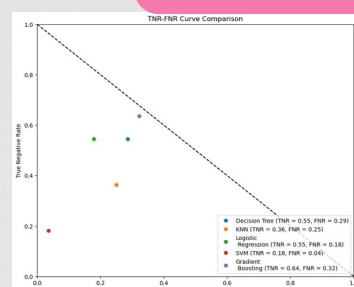
Melhor: Gradient Boosting



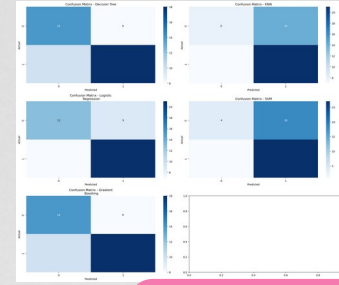
3



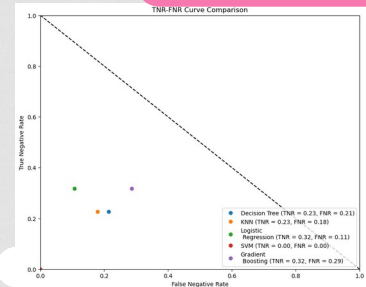
Melhor: Gradient Boosting



8



Melhor: SVM



Usamos a matriz de confusão e o gráfico TNF-FNR para determinar quantidade de falsos positivos e negativos.

# Comentário

Para a interpretação do nosso problema, o pior erro que pode ocorrer é a previsão de um número elevado de falsos negativos pelos métodos de machine learning. Apesar de os falsos positivos também serem um erro significativo, este não é tão grave por comparação com os falsos negativos. O modelo de machine learning mais eficiente para avaliar o rácio de falsos negativos é o SVM, já o Decision Tree é o menos eficiente.

Para a Area Under the Curve, o modelo com melhor desempenho é o Gradient Boosting, e o com pior desempenho é o KNN. No entanto, observa-se que o Gradient Boosting apesar de resultar em muitos verdadeiros positivos, também resulta em muitos falsos negativos. Desta forma, perante o contexto de previsão de sobrevivência, não se pode considerar este modelo eficiente. Já em relação aos parâmetros accuracy e precision, o melhor modelo é o Gradient Boosting, e para o recall, o melhor modelo é o SVM. Mais uma vez, o modelo que se destaca pela negativa é o Decision Tree. Numa visão mais geral, o modelo que se destaca pela positiva é o Gradient Boosting, pois é o que apresenta valores mais altos no conjunto dos três parâmetros. Fazendo uma análise geral, tendo em conta a percentagem de falsos negativos, o AUC, a accuracy, o recall e a precision, concluímos que o modelo que apresenta melhor desempenho é o SVM. A qualidade das variáveis também tem um impacto significativo no desempenho dos modelos. Por esta razão, o teste deste desempenho foi executado com 10 casos diferentes, cada um com um conjunto de variáveis diferentes, que justificam as alterações de média em relação ao caso base.

**Link do repositório no GitHub: <https://github.com/mariamorais09/trabalho2>**