UÉvora
IAA - MEI 2023/2024
Projecto de Avaliação

# Análise de Dados de Vinhos Portugueses

**Gonçalo Candeias Amaro**
Número 56870

15 Jan 2024

**Resumo**

Este relatório apresenta uma análise de vinhos portugueses "Vinho Verde" e modelos de machine learning para previsão de qualidade e teor alcoólico, bem como a distinção entre vinhos tintos e brancos.

## 1 Introdução

Introdução ao projeto e aos conjuntos de dados.

## 2 Análise Exploratória de Dados

### 2.1 Visão Geral Rápida

Esta subsecção apresenta uma visão geral inicial dos conjuntos de dados.

Foi possível observar que os conjuntos de dados possuem 12 colunas cada (acidez fixa, acidez volátil, ácido cítrico, açúcar residual, cloretos, dióxido de enxofre livre, dióxido de enxofre total, densidade, pH, sulfatos, álcool, qualidade).

As colunas de atributos são todas numéricas. Os valores são todos float64 ou int32.

Foi feito uma inspeção rápida onde se encontrou que havia *outliers* em algumas colunas, e valores nulos no conjunto de dados de vinhos brancos e vermelhos.

Como na tabela abaixo, podemos observar que os conjuntos de dados possuem valores nulos ou ausentes.

| Vinho | *Outliers* | *Nulls* |
|---|---|---|
| Tinto | 773 | 2 |
| Branco | 2627 | 1 |

Tabela 1: Visão geral dos conjuntos de dados.

## 2.2 Valores Nulos ou Ausentes

Seguidamente, tratamos da presença de valores nulos ou ausentes nos conjuntos de dados, convertendo os tipos de dados para os tipos mais adequados e substitui valores nulos ou ausentes por `pd.NA`.

Com isso, foi possível observar que os conjuntos de dados não possuem mais valores nulos ou ausentes. O valor `pd.NA` já é um valor nulo, mas é tratado de forma diferente pelo pandas. Com isto em mente, podemos dizer que os conjuntos de dados não possuem mais valores nulos ou ausentes (os quais possam dar problemas na análise).

## 2.3 Detecção de *outliers* e Tratamento

Para identificar e tratar *outliers*, utilizamos o `winsorize` da biblioteca `scipy.stats` para substituir os *outliers* por valores próximos aos limites do intervalo de confiança.

`Winsorize` é uma função que recebe um array e retorna um array com os *outliers* substituídos. O parâmetro `limits` é um tuple que define os limites do intervalo de confiança. Neste caso, os limites são 0.05 e 0.95, o que significa que os valores abaixo do percentile 5 e acima do percentile 95 serão substituídos.

O tratamento dos tipos de dados já está incluído na função `parse_and_clean`, que força a conversão de todos os valores para o tipo correto.

## 2.4 Normalização dos Dados

Para normalizar os dados, utilizamos o `RobustScaler`. Em que os dados são normalizados usando a seguinte fórmula:

$$X_{scaled} = \frac{X - Q_1(X)}{Q_3(X) - Q_1(X)} \tag{1}$$

Onde $Q_1(X)$ é o primeiro quartil e $Q_3(X)$ é o terceiro quartil. Este metodo por si já trata os *outliers*, pois o `RobustScaler` é um método robusto, no entanto, foi utilizado o método `handle_outliers` para garantir que os *outliers* fossem tratados antes da normalização.

É tambem resistentes a *pd.NA*.


Figura 1: Mapa de calor da correlação entre os atributos dos vinhos tintos.

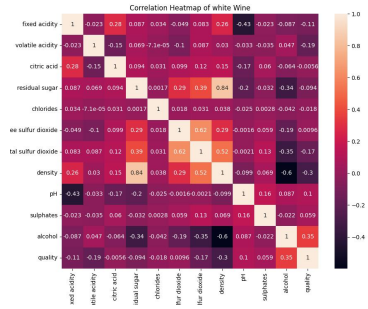


Figura 2: Mapa de calor da correlação entre os atributos dos vinhos brancos.

### 2.5 Fusão e Exportação dos Dados Tratados

Para terminar esta secção, fundimos os conjuntos de dados de vinhos brancos e vinhos tintos e exportamos o conjunto de dados tratado para um ficheiro CSV.

## 3 Análise Exploratória de Dados

Ainda no mesmo script, foram feitas algumas análises exploratórias de dados. A seguir, apresentamos os resultados.

### 3.1 Distribuição de Atributos

A distribuição de atributos foi analisada usando um metodo chamado `eda` que recebe um conjunto de dados e o tipo de vinho (tinto ou branco) e gera um histograma para cada coluna do conjunto de dados.

Este faz output do método `describe` em que retorna um resumo estatístico dos dados. Seguido do método `histplot` da biblioteca `seaborn` gera um histograma para cada coluna do conjunto de dados.

Aqui o eda foi chamado duas vezes, uma para cada conjunto de dados. Os resultados foram salvos em `out/graphs`.

### 3.2 Correlação entre Atributos

A correlação entre atributos foi analisada usando o método `corr` da biblioteca `pandas` retorna uma matriz de correlação entre os atributos. O método `heatmap` da biblioteca `seaborn` gera um mapa de calor para a matriz de correlação.

O gráfico de correlação gerado para os vinhos tintos é apresentado na Figura 1 e o gráfico de correlação gerado para os vinhos brancos é apresentado na Figura 2.

Como podemos observar, os atributos mais correlacionados com a qualidade são o álcool e a acidez volátil. Os atributos mais correlacionados com o álcool são a densidade e o açúcar residual. Os atributos mais correlacionados com a acidez volátil são o ácido cítrico e o pH.

# 4 Tarefa de Regressão

Com esta tarefa, pretendemos prever o nível de álcool dos vinhos através dos outros atributos com modelos de regressão.

## 4.1 Previsão de nível de álcool

Para prever o nível de álcool, utilizamos os seguintes modelos:

1. Regressão Linear
2. Regressão de Floresta Aleatória

Os quais foram implementados usando o ficheiro: `src/task1.py`.
Com estes modelos, obtivemos os seguintes resultados:

| Modelo | MSE | R2 Score |
|---|---|---|
| Regressão Linear | 0.11 | 0.71 |
| Regressão de Floresta Aleatória | 0.05 | 0.87 |

Tabela 2: Resultados da tarefa de regressão.

Com base nos resultados apresentados na Tabela 2, podemos concluir que o modelo de regressão de floresta aleatória é o mais adequado para prever o nível de álcool. Este modelo obteve um MSE de 0.05 e um R2 Score de 0.87, enquanto que o modelo de regressão linear obteve um MSE de 0.11 e um R2 Score de 0.71.

Não só sendo melhor na previsão como tambem nos proporciona a a oportunidade de analisar a importancia de cada atributo para a previsão.

Como podemos observar na Figura 3, os atributos mais importantes para a previsão do nível de álcool são a densidade, o açúcar residual e o pH.

# 5 Tarefa de Classificação

Com esta tarefa, pretendemos prever o tipo de vinho (tinto ou branco) através dos outros atributos com modelos de classificação.

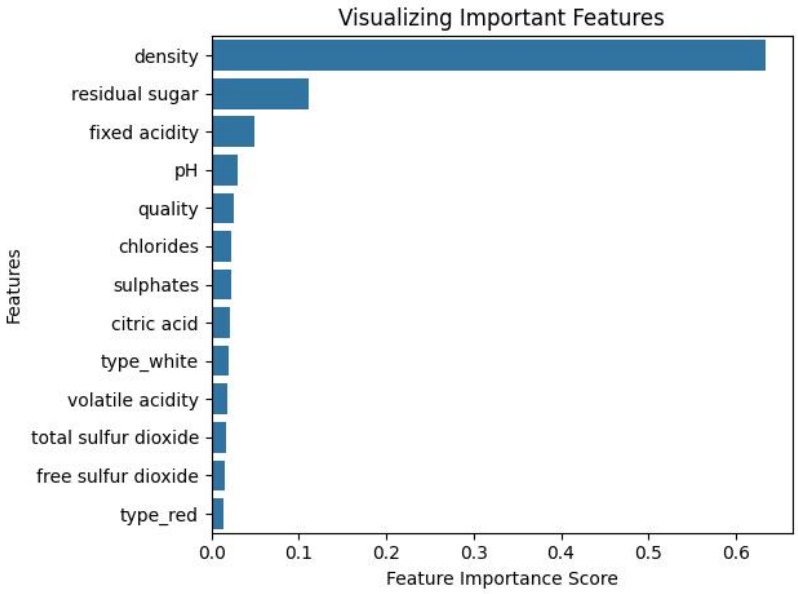


Figura 3: Importância dos atributos para a previsão do nível de álcool.

## 5.1 Previsão do tipo de vinho

Para prever o tipo de vinho, utilizamos os seguintes modelos:

1. Regressão Logística
2. Floresta Aleatória

Os quais foram implementados usando o ficheiro: `src/task2.py`.
om estes modelos, obtivemos os seguintes resultados:

| Modelo | Accuracy | F1 Score | Precisão | Recall |
|---|---|---|---|---|
| Regressão Logística | 0.77 | 0.33 | 0.73 | 0.21 |
| Floresta Aleatória | 0.97 | 0.94 | 1.0 | 0.90 |

Tabela 3: Resultados da tarefa de classificação.

Com base nos resultados apresentados na Tabela 3, podemos concluir que o modelo de floresta aleatória é o mais adequado para prever o tipo de vinho. Este modelo obteve uma *accuracy* de 0.97, um F1 Score de 0.94, uma precisão de 1.0 e um *recall* de 0.90, enquanto que o modelo de regressão logística obteve uma *accuracy* de 0.77, um F1 Score de 0.33, uma precisão de 0.73 e um *recall* de 0.21.

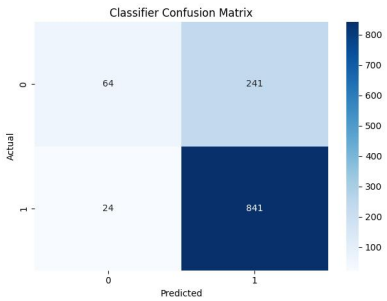


Figura 4: Matriz de confusão para o modelo de floresta aleatória.

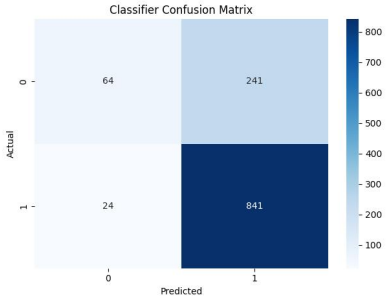


Figura 5: Matriz de confusão para o modelo de regressão logística.

Continuando a moda anterior dos modelos de floresta aleatória serem os melhores para este caso de estudo especifico.

# 6 Tarefa de previsão de qualidade

Com esta tarefa, pretendemos prever a qualidade dos vinhos através dos outros atributos com modelos de classificação.

## 6.1 Previsão da qualidade dos vinhos

Para prever a qualidade dos vinhos, utilizamos os seguintes modelos:

1. *Gradient Boosting Regressor*
2. *K-Nearest Neighbors Regressor*
3. *Linear Regression*
4. *Random Forest Regressor*
5. *Support Vector Regressor*

Os quais foram implementados usando o ficheiro: `src/task3.py`.

Com estes modelos, obtivemos os seguintes resultados:

Com base nos resultados apresentados na Tabela 4, podemos concluir que o modelo de floresta aleatória é o mais adequado para prever a qualidade dos vinhos. Este modelo obteve um MSE de 0.26, um R2 Score de 0.51 e um MAE de 0.39, enquanto que os outros modelos obtiveram um MSE entre 0.32 e 0.37, um R2 Score entre 0.27 e 0.37 e um MAE entre 0.44 e 0.50.

A *trend* continua, o modelo de floresta aleatória é o melhor para este caso de estudo.

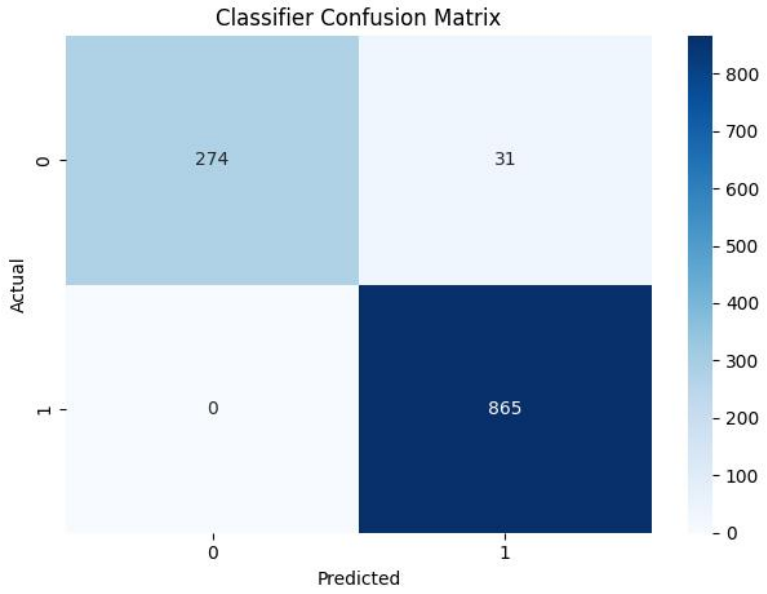


Figura 6: Matriz de confusão para o modelo de floresta aleatória.

| Modelo | MSE | R2 Score | MAE |
|---|---|---|---|
| *Gradient Boosting Regressor* | 0.33 | 0.37 | 0.47 |
| *K-Nearest Neighbors Regressor* | 0.35 | 0.32 | 0.46 |
| *Linear Regression* | 0.37 | 0.27 | 0.50 |
| *Random Forest Regressor* | 0.26 | 0.51 | 0.39 |
| *Support Vector Regressor* | 0.32 | 0.37 | 0.44 |

Tabela 4: Resultados da tarefa de previsão de qualidade.

# 7 Conclusão

Foi efetuada uma análise exploratória de dados e foram implementados modelos de machine learning para previsão de qualidade e teor alcoólico, bem como a distinção entre vinhos tintos e brancos. Os modelos de floresta aleatória foram os mais adequados para prever o teor alcoólico, o tipo de vinho e a qualidade dos vinhos.

# Referências


[1] Towards Data Science. https://towardsdatascience.com/wine-quality-prediction-using-machine-learning-9c5cbc8b148b.

[2] DataCamp. https://www.datacamp.com/community/tutorials/random-forests-classifier-python.

[3] DataCamp. https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python.

[4] DataCamp. https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn.

[5] DataCamp. https://www.datacamp.com/community/tutorials/understanding-logistic-regression-python.

[6] ChatGPT Queries. (Debug python outputs) https://chat.openai.com/.