U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Previsão de Resultados Exatos em Jogos de Futebol Europeu

IART – Trabalho 2

João Praça – up201704748

Liliana Almeida – up201706908

Sílvia Rocha – up201704684

Especificação do Trabalho

O objetivo deste projeto é resolver um problema de regressão, nomeadamente a previsão de resultados exatos em jogos de futebol europeu.

Match

- Id das equipas
- País
- Época
- Liga
- Jornada
- Data
- Número de golos de cada equipa
- Identificação e Posicionamento de cada um dos 22 jogadores
- Odds de casas de apostas

Team attributes

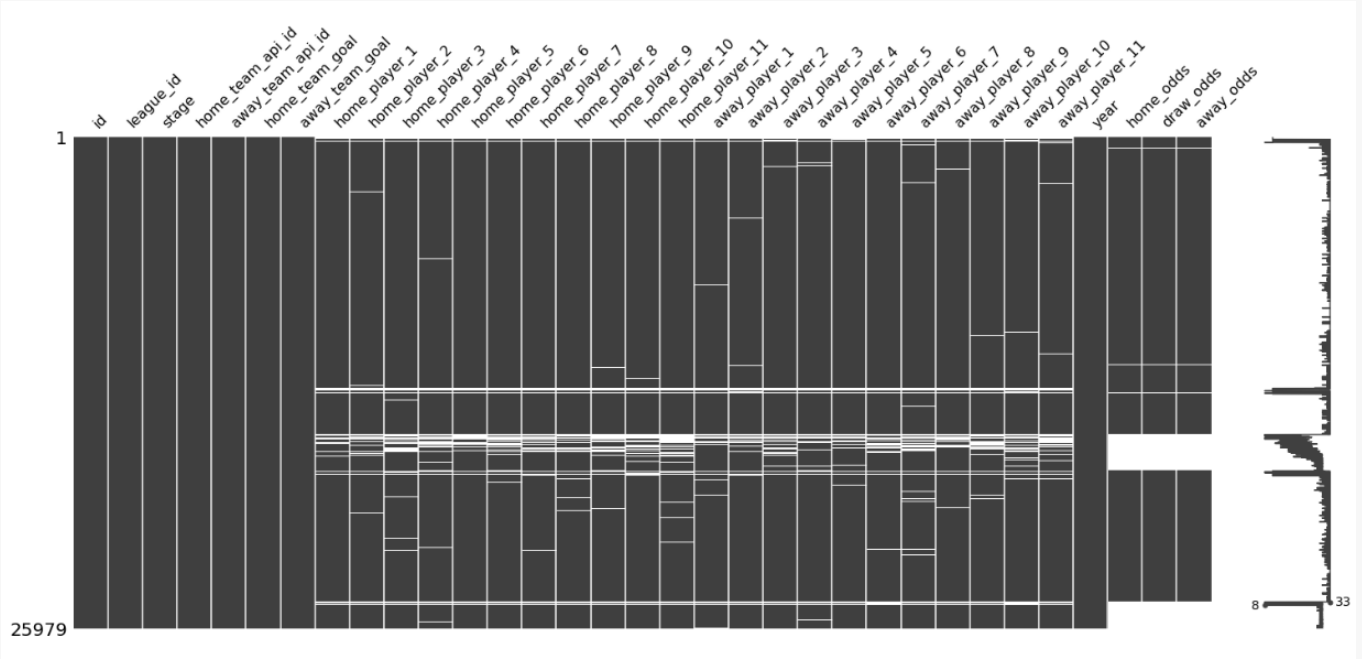
- Build-up play speed
- Build-up passing
- Build-up dribbling
- Probabilidade criação de passe
- Probabilidade criação de remates
- Probabilidade criação de cruzamentos
- Pressão defensiva
- Agressividade defensiva
- Largura da defesa

Player attributes

- Overall rating
- Potencial
- Habilidades de guarda-redes
- Posicionamento e visão de jogo
- Agressividade e interceção de passe
- Remates
- Velocidade
- Entre outros

Pré-processamento e transformação de dados

Nesta fase, os dados omissos foram uniformizados e, de seguida, foram removidas todas as linhas que os incluíam.

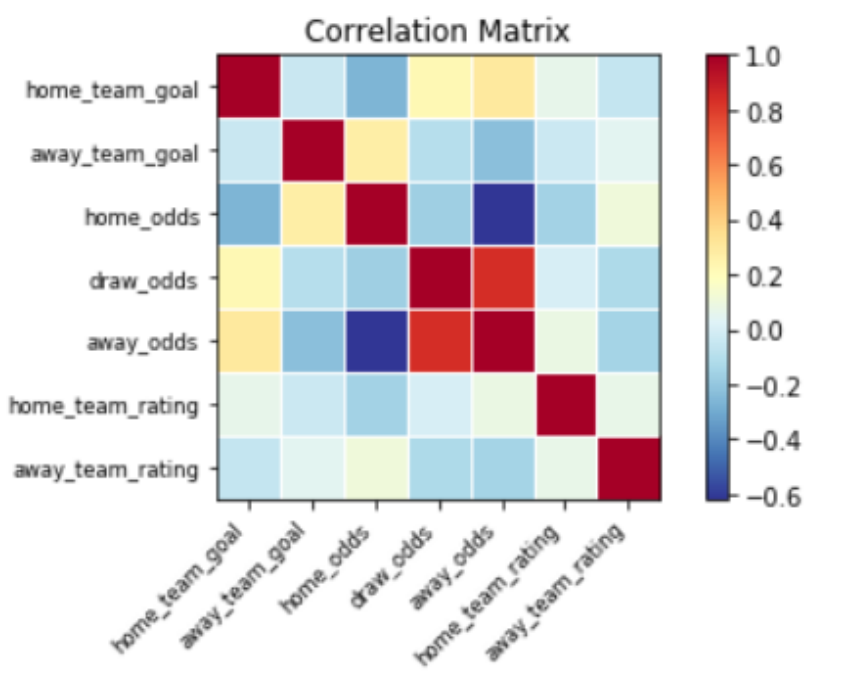


De seguida, foi aplicada uma normalização sobre os valores das variáveis a serem utilizadas pelos algoritmos de cálculo de regressão.

Análise de correlações e variâncias dos dados

PCA - Principal Component Analysis

	home_odds	draw_odds	away_odds	home_team_rating	away_team_rating
PC-1	-0.454502	0.557266	0.664816	0.100483	-0.175518
PC-2	0.268835	0.189396	0.005382	-0.785551	-0.524155



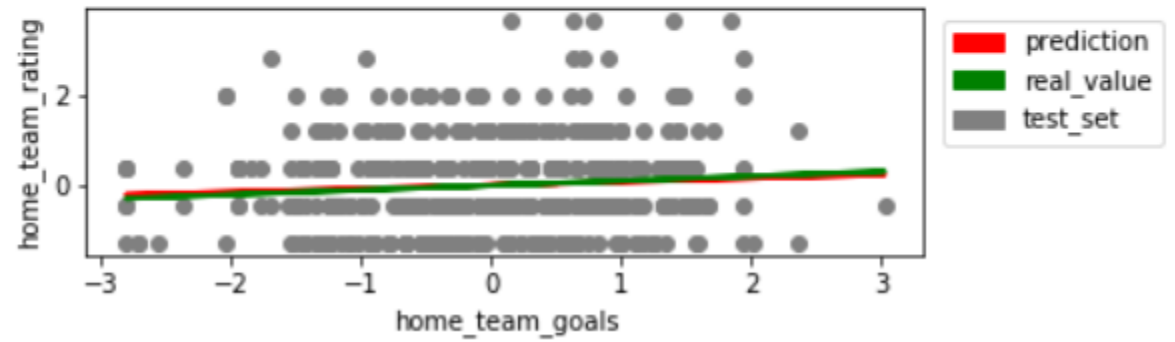
Algoritmos Implementados

A nível de algoritmos, foram implementadas soluções recorrendo aos seguintes algoritmos:

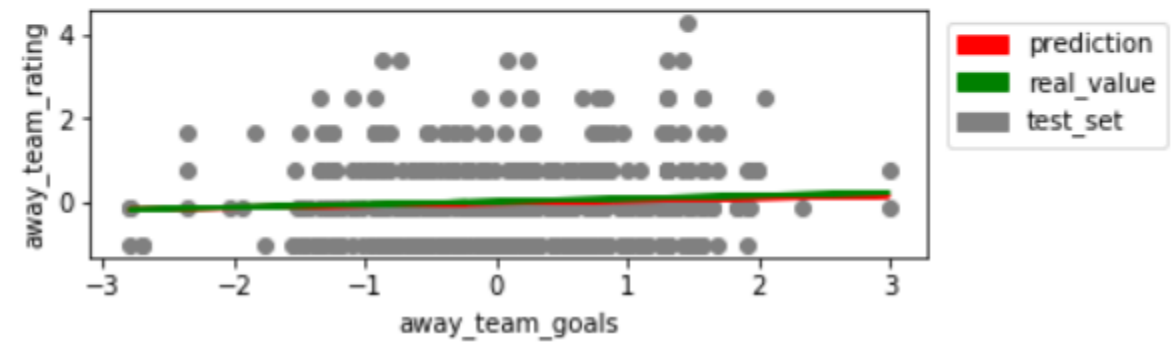
- Regressão Linear;
- Árvores de Decisão;
- Redes Neuronais;
- K-Nearest Neighbour;
- Support Vector Machines.

Resultados Regressão Linear

```
Linear Regression
     regression score = 0.12690532714618508
     mean squared error = 0.9441655799369569
     explained variance score = 0.12693230972454494
     mean absolute error = 0.7814915830470249
     max error = 3.696827616164721
```

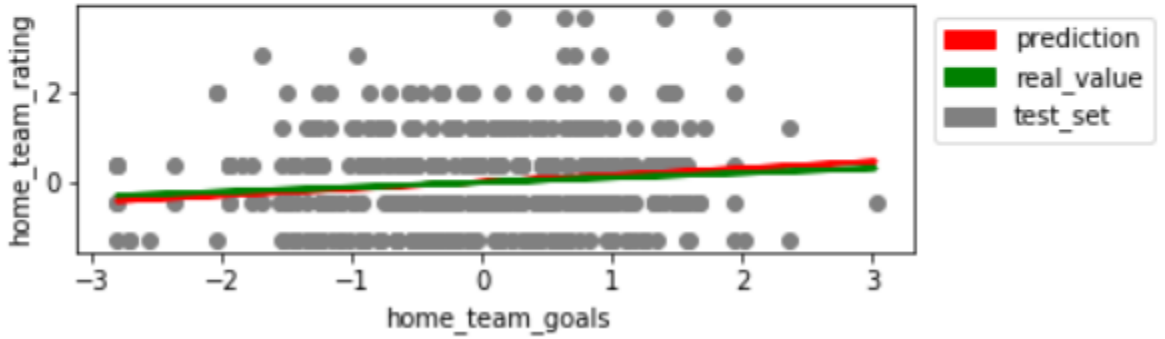



```
Linear Regression
     regression score = 0.07673667325916622
     mean squared error = 0.9572181985458849
     explained variance score = 0.07837088953547255
     mean absolute error = 0.7504047377448795
     max error = 4.166824773757788
```

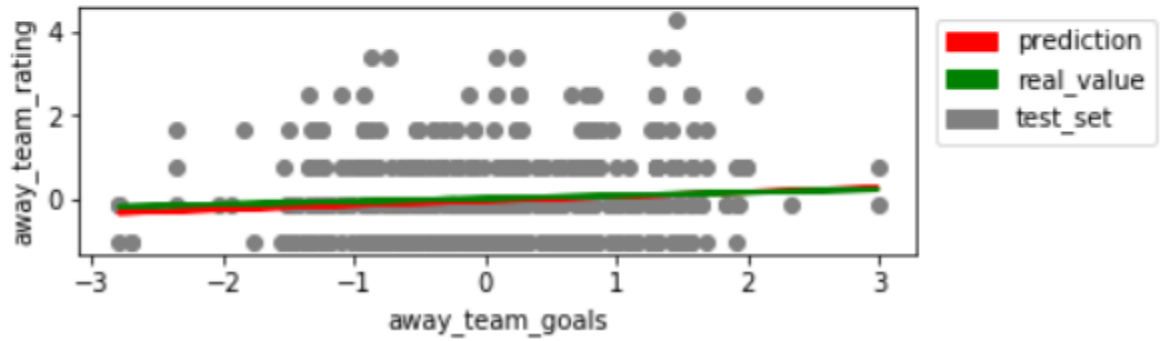


Resultados
K-Nearest Neighbour

K-Nearest Neighbour
regression score = 0.46382829676489723
mean squared error = 0.5798166944210841
explained variance score = 0.4639477506196904
mean absolute error = 0.5002889501867569
max error = 3.2832305938291637

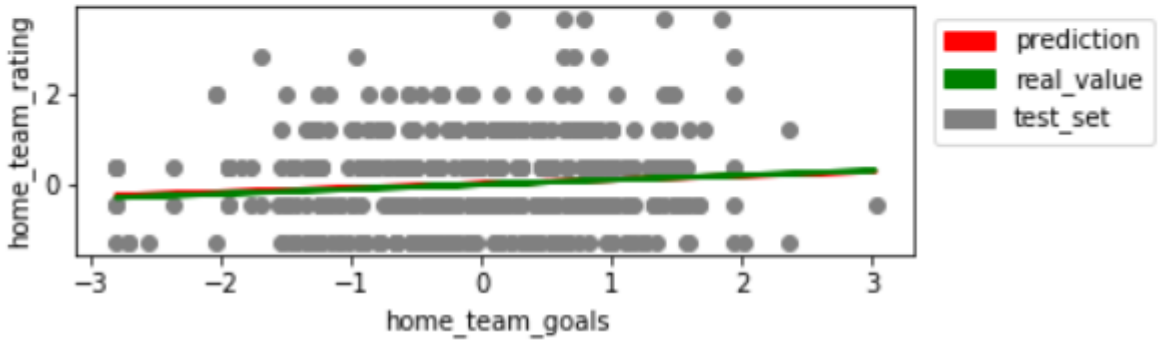


K-Nearest Neighbour
regression score = 0.3246397510190163
mean squared error = 0.7001979848816687
explained variance score = 0.3276780746410256
mean absolute error = 0.527312103296286
max error = 4.655551307187784

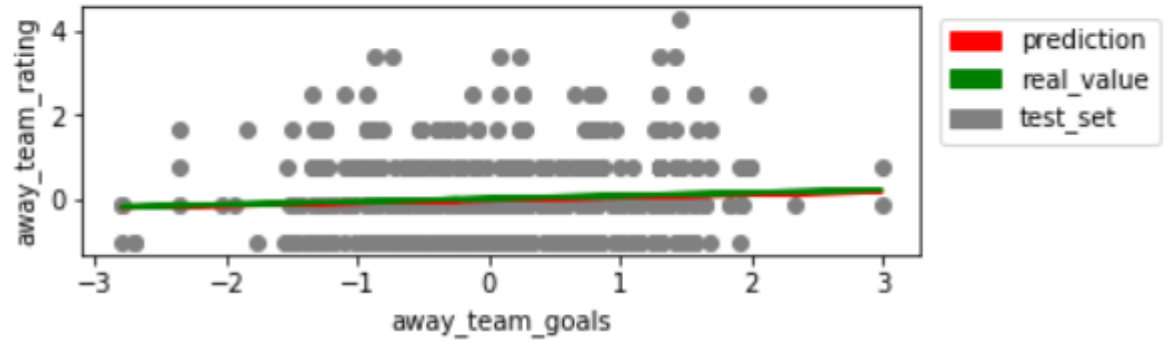


Resultados Neural Networks

```
Neural Network
      regression score = 0.16163940604460725
      mean squared error = 0.9066041072051269
      explained variance score = 0.16182755968899465
      mean absolute error = 0.7576837809172315
      max error = 3.463444828418588
```

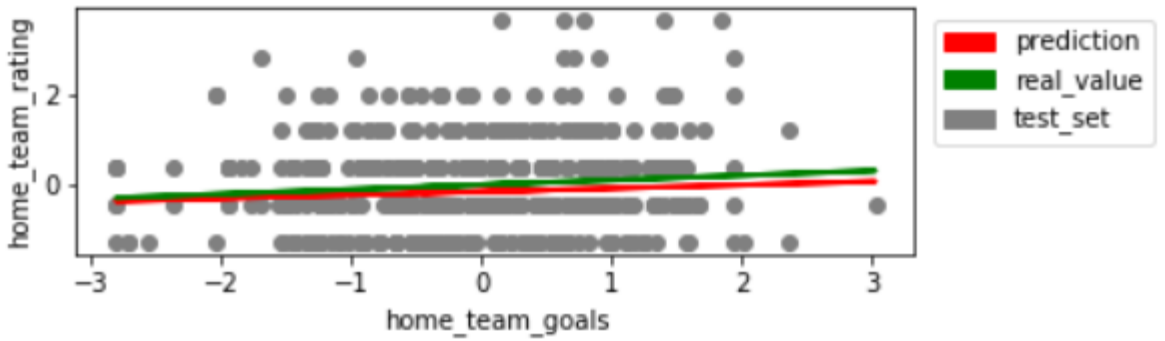



```
Neural Network
      regression score = 0.09322711719042676
      mean squared error = 0.9401212852645735
      explained variance score = 0.09439389561832001
      mean absolute error = 0.7410204908019123
      max error = 4.196634568426302
```

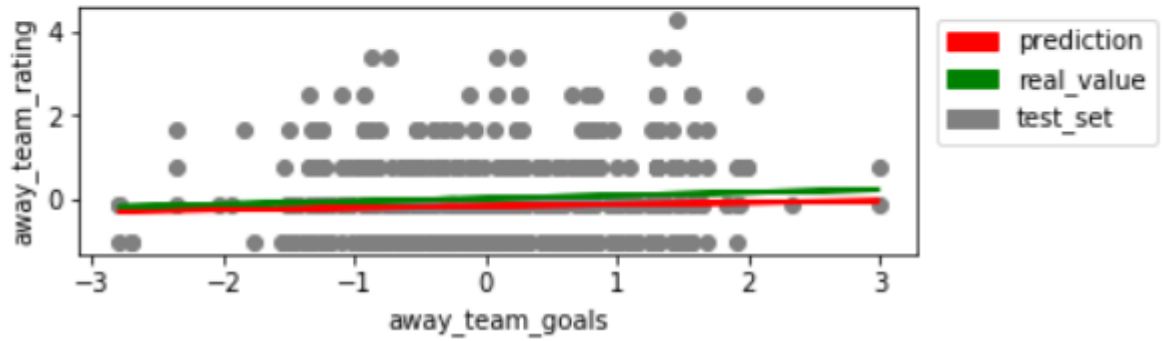


Resultados Support Vector Machines

```
Support Vector Machines
     regression score = 0.11991746366208222
     mean squared error = 0.9517222635179288
     explained variance score = 0.14217651884121774
     mean absolute error = 0.7350985548521618
     max error = 3.9791681992358545
```

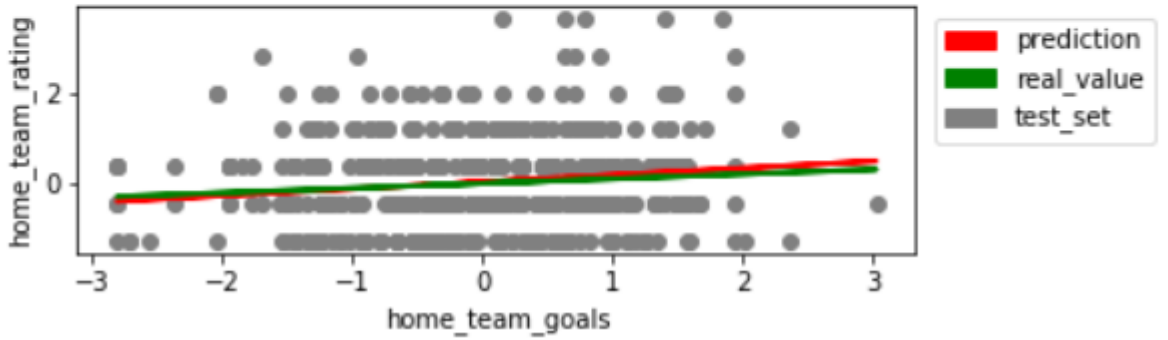



```
Support Vector Machines
     regression score = 0.04499699241837518
     mean squared error = 0.9901251701940416
     explained variance score = 0.07892394108506129
     mean absolute error = 0.7259257843421816
     max error = 4.643701979790694
```

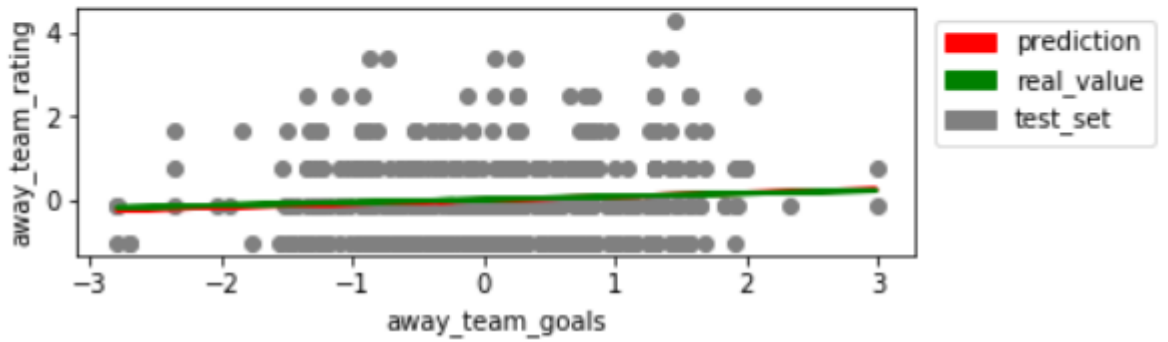


Resultados Decision Trees

```
Decision Tree
     regression score = 0.7985846817041095
     mean squared error = 0.21781075606089162
     explained variance score = 0.7996925187088799
     mean absolute error = 0.12691202596628798
     max error = 4.104038242286455
```




```
Decision Tree
     regression score = 0.8668873870453337
     mean squared error = 0.13800809789119708
     explained variance score = 0.8671911306857013
     mean absolute error = 0.09582747336530101
     max error = 2.651226763106659
```



Conclusões

O algoritmo que consegue obter uma regressão mais precisa para os dados utilizados é Decision Trees, seguido pelo K-Nearest Neighbour.

O uso de K-Fold Cross Validation neste conjunto de dados, método recomendado para datasets com mais de 1000 exemplos, não permitiu uma melhoria do modelo obtido.

Referências

https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_predict.html#sklearn.model_selection.cross_val_predict

https://scikit-learn.org/stable/supervised_learning.html#supervised-learning

https://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics

https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html

https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html

Referências

https://www.analyticsvidhya.com/blog/2018/08/k-nearest-neighbor-introduction-regression-python/

https://towardsdatascience.com/train-test-split-and-cross-validation-in-python-80b61beca4b6

https://towardsdatascience.com/https-medium-com-lorrli-classification-and-regression-analysis-with-decision-trees-c43cdbc58054

https://www.saedsayad.com/decision_tree_reg.htm

https://medium.com/@AI_with_Kain/understanding-of-multilayer-perceptron-mlp-8f179c4a135

https://medium.com/@denzilsequeira/data-pre-processing-for-deep-learning-for-classification-or-regression-2bddb0b9183b

https://pandas.pydata.org/pandas-docs/stable/reference/frame.html