# **Aprendizagem Automática 2023/2024**

### **First Home Assignment**

### **Objective 1:**

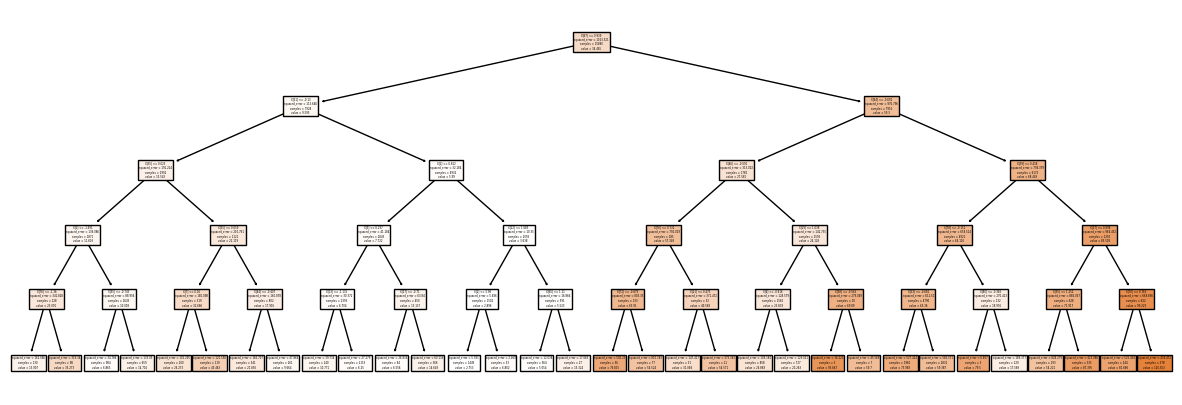
### **Produce the best regression model for critical\_temp (Dependent Variable)**

**Sumário do trabalho realizado:**

Carregamos o dataset a partir do ficheiro: "HA1-DatasetScaled.tsv"(apenas uma vez para ambos os objetivos)

1. Divisão dos dados em dois dataframes, um com apenas a variável dependente (“critical\_ temp”), e o outro com o resto das variáveis
2. Fizemos a divisão dos dados em conjuntos de treino e de teste
3. Aplicamos as implementações dos modelos: Decision tree Regression, Linear Regression(do sklearn) ,Linear Regression(do statsmodel), Ridge Regression, Lasso Regression
4. Fizemos fit de cada um dos modelos implementados com o training set
5. Efetuamos previsões com cada um dos modelos treinados usando as variáveis independentes do set de teste (X\_test)
6. As previsões foram comparadas com os valores verdadeiros de”crit\_temp” (y\_test) de forma a avaliá-las segundo as seguintes métricas: rve: Ratio of Variance Explained; rmse - Root Mean Squared Error; Pearson Correlation score; Mean absolute error; Maximum error.

**Tabela com modelos testados e respetivas estatísticas relevantes(Resultados):**



decision tree max depth = 5

Métricas de avaliação de decision tree regression com max depth 10(demorou mais de 1min para correr o modelo):

The RVE is: 0.8513227476146826

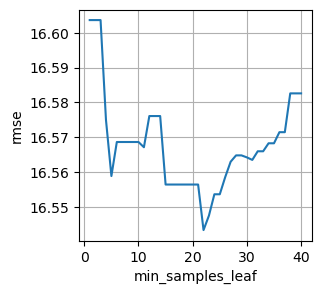
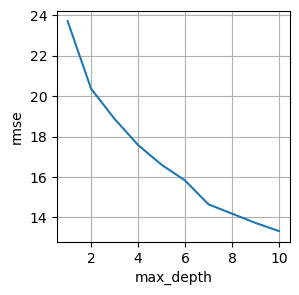
The rmse is: 13.374349733219168

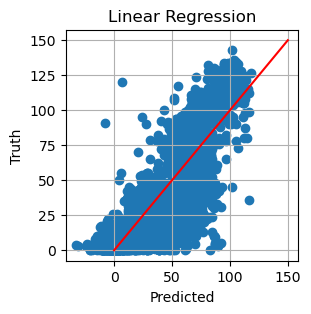
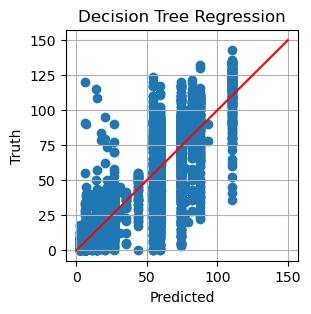
The Correlation Score is: 0.9232 (p-value=0.000000e+00)

The Maximum Error is: 115.72848591549295

The Mean Absolute Error is: 7.850503080634878

Não compensa usar este modelo de árvore de decisão tree com profundidade máxima de 10, porque para além da árvore ficar demasiado complexa não melhora muito o desempenho do modelo comparando com “decision tree” com profundidade 5.



Com as duas imagens à esquerda é possível verificar que o aumento da profundidade máxima na árvore de decisão regression representa uma diminuição significativa na raiz do erro quadrático médio (rmse), pelo contrário a manipulação do nº mínimo de amostras por folha tem efeitos manifestamente inexpressivos na mesma métrica de avaliação do modelo.

### 

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RVE | RMSE | Correlation Score | Maximum Error | Mean Absolute Error |
| decision tree(depth=5) | 0.7709 | 16.6036 | 0.8780 | 113.7497 | 11.1612 |
| linear regression | 0.7326 | 17.9377 | 0.8559 | 113.1892 | 13.5649 |
| ridge regression | 0.7288 | 18.0630 | 0.8537 | 109.9396 | 13.6636 |
| lasso regression | 0.4525 | 25.6667 | 0.7426 | 97.029 | 20.663 |

**Discussão e conclusões:**

Depois de correr os modelos várias vezes para evitar o acaso, o melhor modelo parece ser o decision tree depth=5, com melhor RVE score, RMSE mais baixo, melhor “correlation score” e melhor “mean absolute error”, apesar de considerar o linear regression model como mais simples. As diferenças nas pontuações entre o decision tree regression e os modelos linear regression e ridge regression não são muito significativas, mas o modelo decision tree regression não deixa de ter os melhores resultados no geral e de ser mais compreensível dos demais. De notar uma grande discrepância nos resultados do modelo lasso regression que são notoriamente inferiores ao resto dos modelos.

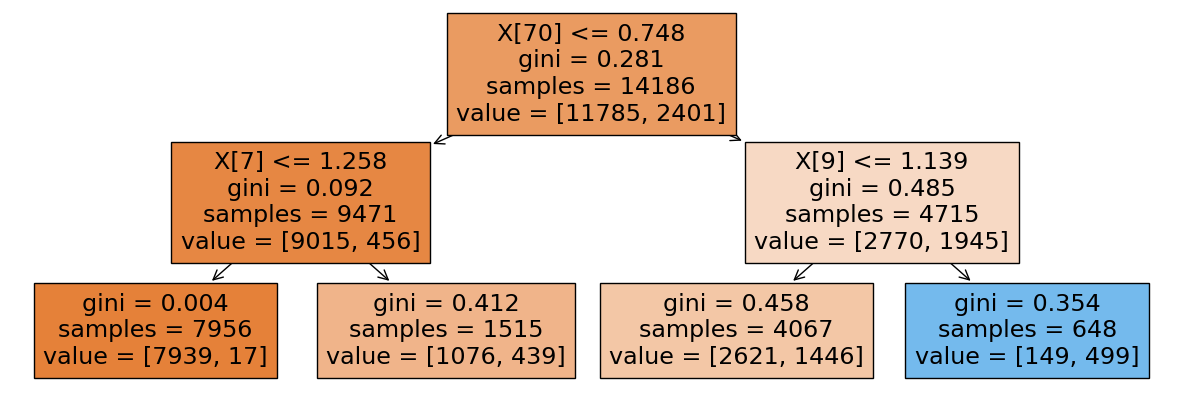
### **Objective 2:**

### **Best binary classification model assuming as positive all instances with values of critical\_temp >= 80.0 and as negatives all remaining cases**

**sumário do trabalho realizado:**

1. Para a criação de modelos de classificação foi primeiro necessário modificar o dataset com a reclassificação dos valores de Critical\_temp em classes discretas, de forma binária com os valores de critical\_temp inferiores a 80 a assumir o valor 0 correspondente a“negativo” e os restantes a assumir o valor 0 correspondente a “positivo”, de acordo com o enunciado.
2. Fizemos a divisão dos dados em conjuntos de treino e de teste
3. Aplicamos as implementações dos modelos: Decision tree classifier, Logistic regression
4. Fizemos fit de cada um dos modelos implementados com o training set. Para o logistic regression model fizemos scale das variáveis independentes para treino e teste (X\_crit\_train e X\_crit\_test) antes de fazermos o fit.
5. Efetuamos previsões com cada um dos modelos treinados usando as variáveis independentes do set de teste(X\_crit\_test e X\_crit\_test\_s)
6. As previsões foram comparadas com os valores verdadeiros de”crit\_temp” (y\_test) de forma a avaliá-las segundo as seguintes métricas: Accuracy, Precision, Recall, f1 score, Matthews correlation coefficient;
7. Comparámos também a confusion matrix de cada modelo.

**tabela com modelos testados e respetivas estatísticas relevantes(Resultados):**



<- figura ilustrada do modelo decision tree classifier com depth=2:

Resultados da avaliação do modelo decision tree classifier depth = 2:

The Accuracy is: 0.8570

The Precision is: 0.7292

The Recall is: 0.2064

The F1 score is: 0.3218

The Matthews correlation coefficient is: 0.3367

This is the Confusion Matrix

0 = negative & 1 = positive:

0 1

0 5752 88 Obs: O nº de amostras negativas na totalidade dos dados é

1 911 237 muito superior ao nº de amostras positivas, justificando o elevado nº de previsões de negativo incorretas (911) para apenas 237 previsões corretas de positivo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Accuracy | Precision | Recall | F1 score | Matthews correlation coefficient |
| decisionTree Classifier(depth5) | 0.8987 | 0.7240 | 0.6193 | 0.6676 | 0.6108 |
| logistic regression | 0.8947 | 0.7036 | 0.6202 | 0.6593 | 0.5989 |

**This is the Confusion Matrix**

0 = negative & 1 = positive:

0 1

0 5569 271

1 437 711

0 1

0 5540 300

1 436 712

**Discussão e conclusões:**

Como os resultados dos modelos são muito semelhantes com valores muito idênticos de previsão como é possível visualizar nas “confusion matrizes”, o melhor modelo é o decision tree classifier por ser mais simples e mais fácil de compreender.