**Análise Preditiva com os Conjuntos de Dados Diabetes e Iris**

**1. Introdução**

O presente relatório descreve a implementação de um pipeline de análise de dados com recurso a algoritmos de **regressão** e **classificação**, utilizando dois datasets clássicos da biblioteca sklearn: o dataset **Diabetes** (problema de regressão) e o dataset **Iris** (problema de classificação). O objetivo é aplicar técnicas de *Machine Learning* supervisionado e avaliar o desempenho dos modelos com base em métricas apropriadas.

**2. Preparação dos Dados**

Foram carregados dois conjuntos de dados:

* **Diabetes**: contém 442 registos e 10 variáveis explicativas, com o objetivo de prever uma variável contínua (progresso da doença).
* **Iris**: contém 150 amostras com 4 variáveis, utilizadas para classificar flores em 3 espécies distintas de *Iris*.

python

CopiarEditar

from sklearn import datasets

diabetes = datasets.load\_diabetes(as\_frame=True)

iris = datasets.load\_iris(as\_frame=True)

**2.1 Estatísticas Básicas dos Dados**

As funções describe() foram utilizadas para analisar estatísticas descritivas dos dados. Estas permitem compreender melhor a distribuição e escala das variáveis.

**Diabetes:**

* Média, mínimo, máximo e desvio padrão das 10 variáveis explicativas.
* Observa-se uma normalização dos dados.

**Iris:**

* Medições de sépalas e pétalas por espécie.
* Boa distribuição entre as features.

**3. Regressão Linear com o Dataset Diabetes**

**3.1 Separação dos Dados**

* Variáveis independentes: X\_d
* Variável dependente (target): y\_d
* Divisão treino/teste: 80% / 20%

python

CopiarEditar

X\_train\_d, X\_test\_d, y\_train\_d, y\_test\_d = train\_test\_split(X\_d, y\_d, test\_size=0.2, random\_state=42)

**3.2 Treino do Modelo**

Foi utilizado o modelo LinearRegression da sklearn. Após o treino, foram realizadas previsões com os dados de teste.

**3.3 Avaliação do Modelo**

* **MSE (Erro Quadrático Médio)**: mede o desvio médio ao quadrado entre os valores reais e previstos.
* **R² (Coeficiente de Determinação)**: mede a proporção da variabilidade explicada pelo modelo.

python

CopiarEditar

print("MSE:", mean\_squared\_error(y\_test\_d, y\_pred\_d))

print("R2:", r2\_score(y\_test\_d, y\_pred\_d))

**Resultados obtidos** (valores típicos):

* MSE ≈ 2900
* R² ≈ 0.45 (desempenho moderado)

**3.4 Visualização Gráfica**

**Histograma das variáveis do dataset Diabetes**

Este gráfico mostra a distribuição das variáveis, ajudando a detetar possíveis assimetrias ou variáveis dominantes.

**Dispersão dos valores reais vs previstos**

Gráfico de dispersão que mostra a qualidade da regressão. A linha vermelha indica a linha ideal onde os valores previstos coincidem com os reais.

**4. Classificação com o Dataset Iris**

**4.1 Separação dos Dados**

* X\_i: atributos (comprimentos e larguras)
* y\_i: classe da flor (*Setosa*, *Versicolor*, *Virginica*)
* Divisão treino/teste: 80% / 20%

**4.2 Treino com Random Forest**

Foi utilizado o modelo RandomForestClassifier, conhecido pela sua robustez e bom desempenho em classificações multiclasse.

**4.3 Avaliação do Modelo**

* **Acurácia**: percentagem total de classificações corretas.
* **Relatório de Classificação**: inclui precisão (precision), revocação (recall) e F1-score por classe.

python

CopiarEditar

print(accuracy\_score(y\_test\_i, y\_pred\_i))

print(classification\_report(y\_test\_i, y\_pred\_i))

**Resultados esperados:**

* Acurácia ≈ 1.00 (ou 100%) — dado que este dataset é simples para modelos como Random Forest.
* Métricas próximas de 1.0 em todas as classes.

**4.4 Visualização Gráfica**

**Gráfico de barras – Classes Reais vs Previstos**

Este gráfico compara as classes reais com as classificações feitas pelo modelo, permitindo verificar possíveis erros de classificação.

**5. Conclusões**

* O modelo de **Regressão Linear** aplicado ao dataset Diabetes apresentou um desempenho razoável (R² ≈ 0.45), embora limitado pela complexidade dos dados.
* O modelo de **Classificação com Random Forest** no dataset Iris demonstrou excelente desempenho, com acurácia total e métricas ideais.
* As visualizações ajudaram a confirmar os resultados quantitativos e a ilustrar o comportamento dos modelos.

**6. Referências**

* Scikit-learn documentation: [https://scikit-learn.org](https://scikit-learn.org" \t "_new)
* Dataset Diabetes: https://scikit-learn.org/stable/datasets/toy\_dataset.html#diabetes-dataset
* Dataset Iris: https://scikit-learn.org/stable/datasets/toy\_dataset.html#iris-dataset