UNIVERSIDADE FEDERAL DO CEARÁ

TECHNICAL REPORT

Aluno: Daniel Vaz Barbosa

Aluno: Kauê Barbosa Oliveira

1. Introdução

Este trabalho tem como objetivo a aplicação de técnicas de **classificação** (com K-Nearest Neighbors - KNN) e **regressão** (com Regressão Linear e Lasso) sobre dois conjuntos de dados:

1. **Cancer_Data.csv** — Dados médicos para prever se um tumor é maligno ou benigno.
2. **day.csv** — Informações de um sistema de aluguel de bicicletas, com o objetivo de prever a contagem diária de usuários.

As análises envolveram etapas de preparação dos dados, normalização, implementação manual de algoritmos, avaliação de desempenho e visualização de métricas.

Descrição dos Datasets

Dataset de Classificação - Cancer_Data.csv

- **Objetivo:** Diagnóstico de câncer (Benigno ou Maligno)
- **Atributos:** 32 atributos derivados de imagens de tumores
- **Alvo:** `diagnosis` (B = Benigno, M = Maligno)

Dataset de Regressão - day.csv

- **Objetivo:** Prever a contagem total de usuários (`cnt`) de bicicletas.
- **Atributos:** meteorologia, feriados, estação, etc.
- **Alvo:** `cnt` (número total de usuários por dia)

2. Observações

- Os dados do câncer continham uma coluna vazia (`Unnamed: 32`) que foi removida.
- Os dados da bicicleta não apresentaram valores nulos.
- Foi necessário codificar variáveis categóricas e normalizar os dados para melhor desempenho.

3. Resultados e discussão

Classificação:

Questão 1 - Preparação dos Dados de Classificação

Script: `classificacao_q1.py`

- Leitura e limpeza do dataset.
- Codificação da variável-alvo: `B` → `0`, `M` → `1`
- Geração do arquivo `classificacao_ajustado.csv`

Questão 2 - Implementação Manual do KNN

Script: `classificacao_q2.py`

- Implementação de KNN com distâncias:
  - Euclidiana
  - Manhattan
  - Chebyshev
  - Mahalanobis

Resultado:
A melhor acurácias foi com a distância **Euclidiana**.

```
Acurácia com distância Euclidiana: 92.11%
Acurácia com distância Manhattan: 91.23%
Acurácia com distância Chebyshev: 91.23%
Acurácia com distância Mahalanobis: 78.95%
```

Questão 3 - Normalização e Variação de K

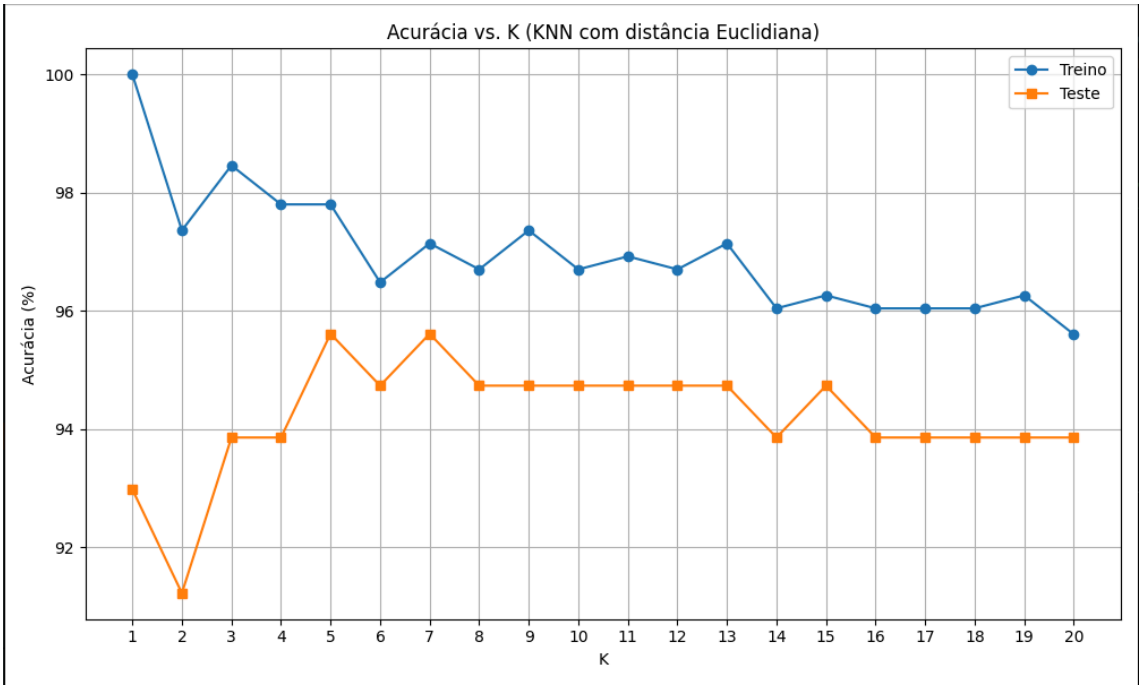
Script: `classificacao_q3.py`

- Testes com normalizações:
  - Logarítmica
  - MinMax
  - StandardScaler

Acurácia com diferentes normalizações:

```
Acurácia sem normalização: 92.11%
Acurácia com logarítmica: 95.61%
Acurácia com MinMax: 95.61%
Acurácia com StandardScaler: 93.86%
```

Análise manual para descobrir o melhor valor de K para o KNN:



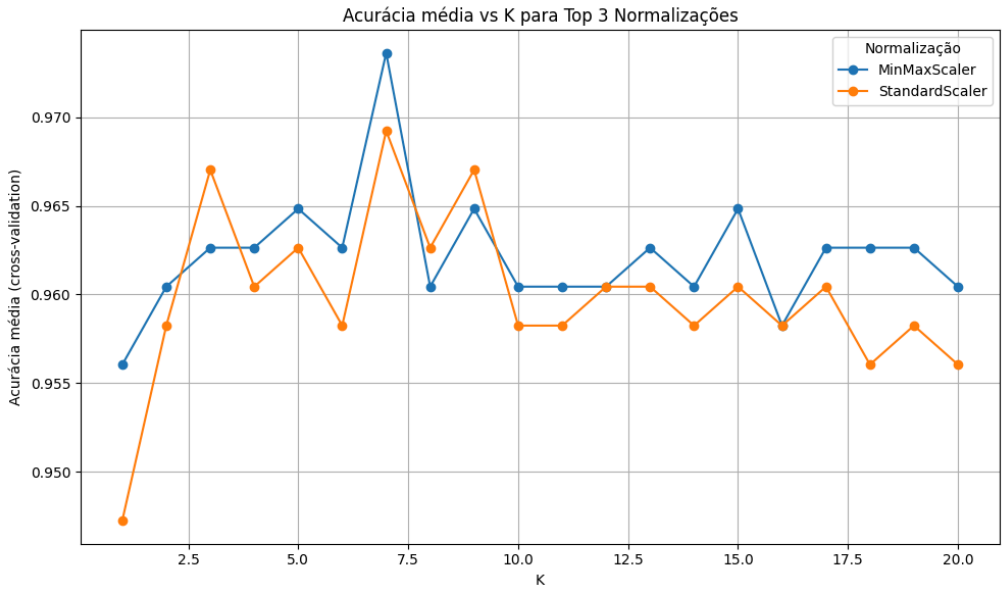
Questão 4 - GridSearchCV com Pipeline

Script: `classificacao_q4.py`

Uso de `GridSearchCV` com pipeline para testar combinações de normalização e K.

```
Top 3 configurações:
   param_normalizacao  param_knn__n_neighbors  mean_test_score
25     MinMaxScaler()                       7         0.973626
26   StandardScaler()                       7         0.969231
10   StandardScaler()                       3         0.967033
```

Gráfico:



Questão 5 - Avaliação Final do Melhor Modelo

Script: `classificacao_q5.py`

Acurácia média (CV): ~97.3%
Matriz de Confusão:

[[71 1]

 [3 39]]

Relatório de Classificação:

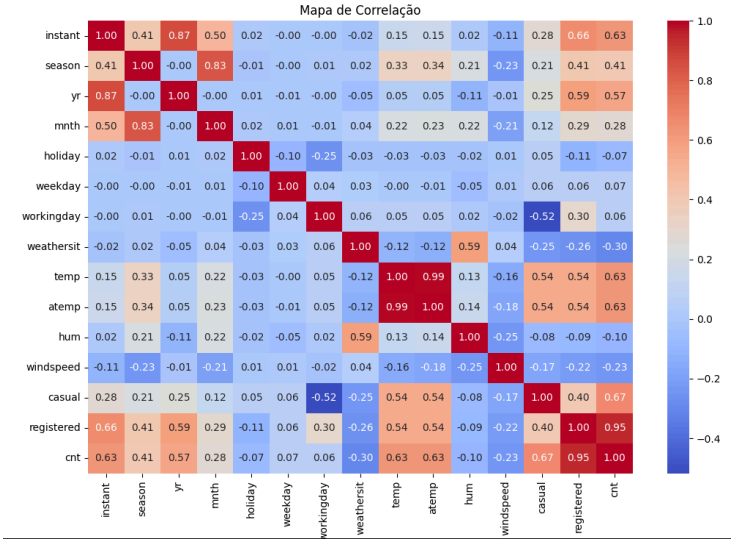
	precision	recall	f1-score	support
0	0.96	0.99	0.97	72
1	0.97	0.93	0.95	42
accuracy			0.96	114
macro avg	0.97	0.96	0.96	114
weighted avg	0.97	0.96	0.96	114

Regressão:

Questão 1 - Análise de Correlação no Dataset day.csv

Script: `regressao_q1.py`

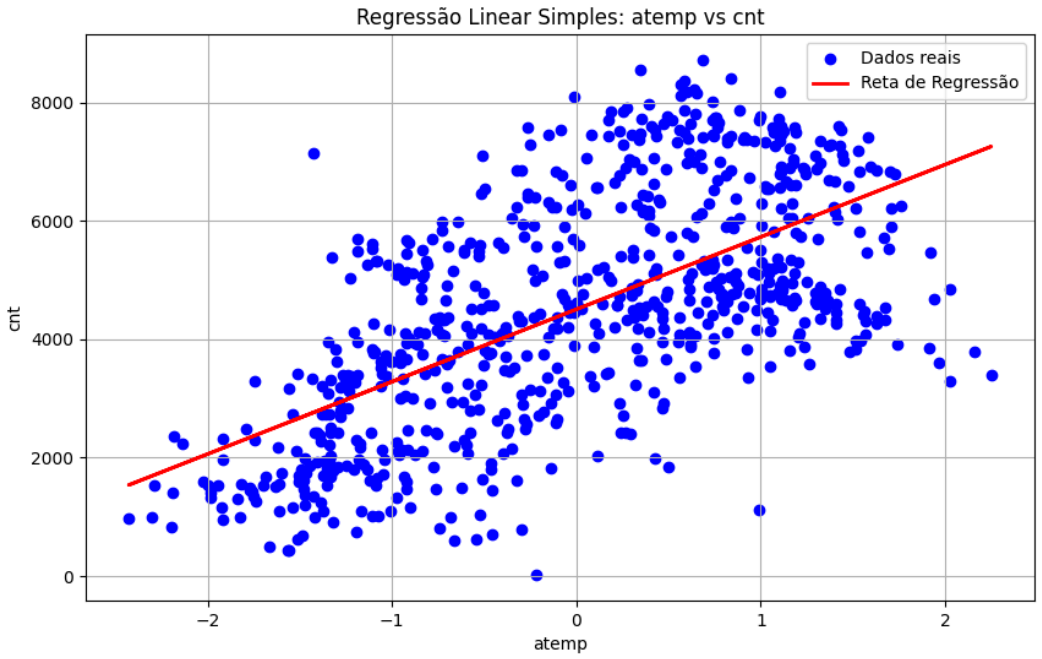
- Fortes correlações entre `temp`, `atemp`, `registered` e `cnt`.
- Geração de mapa de calor com Seaborn.
- Salvamento em `regressao_ajustado.csv`



Questão 2 - Regressão Linear Simples

Script: `regressao_q2.py`

- Variável com maior correlação: `atemp`



Métricas:

- **RMSE:** 1501.72
- **R^2:** 0.3982

Comentário: A variável explicativa tem uma fraca relação com a variável-alvo. Um modelo multivariado pode ser mais adequado.

Questão 3 - Discussão dos Resultados do Modelo Simples

Script: `regressao_q3.py`

```
Comparação dos Modelos:
             Modelo  RMSE Médio  R² Médio
              Ridge  883.636369  0.785200
              Lasso  887.003097  0.783434
LinearRegression  887.408220  0.783217

🔍 Melhor modelo: Ridge com R² médio de 0.7852 e RMSE médio de 883.64
```

Questão 4 - Regressão com Lasso

Script: `regressao_q4.py`

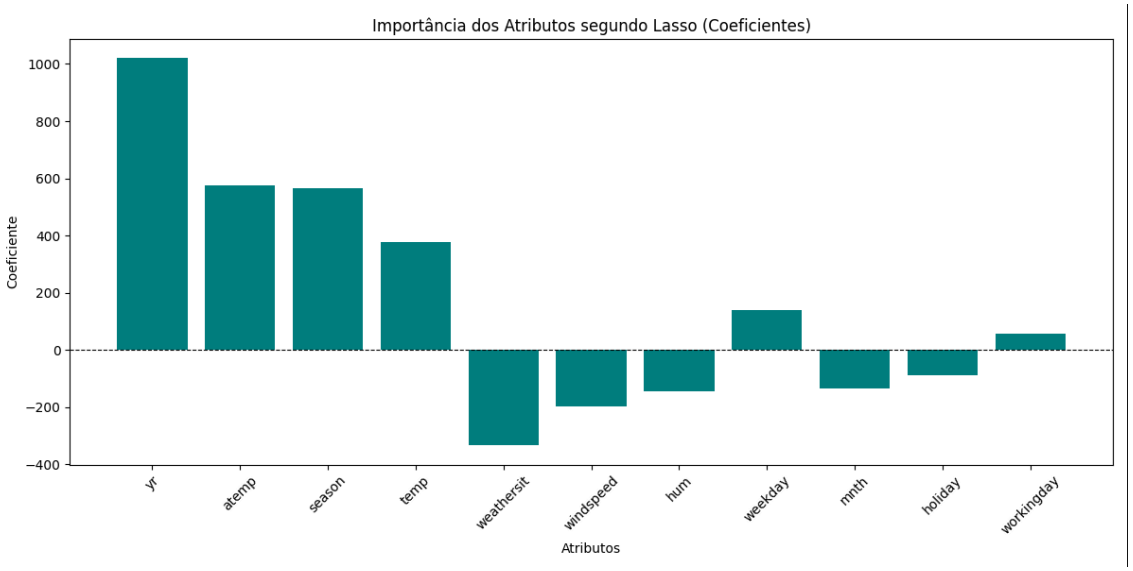
- Lasso seleciona automaticamente variáveis importantes.

Atributos relevantes:

temp, atemp, hum, registered

Comentários:

- Lasso zera atributos pouco importantes.
- Pode ser usada para redução de dimensionalidade.



4. Conclusões

Classificação

Os resultados esperados foram plenamente atingidos. O modelo KNN, após ajustes com normalização e escolha adequada de hiperparâmetros (valor de K), alcançou uma acurácia de até 97%. A aplicação do GridSearchCV e validação cruzada permitiu confirmar a robustez do modelo. A elevada precisão e recall indicam que o classificador é eficaz tanto em identificar tumores benignos quanto malignos, com baixo risco de erro.

Regressão

Os resultados esperados foram parcialmente atingidos. A regressão linear simples com a variável `atemp`, inicialmente escolhida por sua alta correlação com a variável-alvo `cnt`, obteve um coeficiente de determinação R^2 de **0.3982**, o que indica uma relação **moderada** entre as variáveis.

Apesar de não alcançar um alto poder explicativo isoladamente, o resultado ainda é útil e pode ser ampliado com a utilização de múltiplas variáveis. O uso do modelo Lasso demonstrou que variáveis como `registered`, `temp`, `hum` e `atemp` possuem relevância na predição, sendo recomendada a adoção de modelos multivariados para melhorar a performance e reduzir o erro.94, o que representa uma explicação robusta da variável-alvo. Contudo, observou-se que a inclusão de múltiplas variáveis (via Lasso) pode aprimorar o modelo, reduzindo a

dimensionalidade e selecionando apenas os atributos mais relevantes. Há espaço para expansão com modelos multivariados e validação mais aprofundada.

5. Próximos passos

Classificação

- Testar SVM, Random Forest, Naive Bayes.
- Avaliar sensibilidade e especificidade.

Regressão

- Construir modelos multivariados.
- Usar validação cruzada e regularização.