

Projeto Inteligência Artificial (LEIC 3º Ano, 1º Semestre 2017/2018)

Relatório

P1

Conjuntos de features testados:

1. {Comprimento da palavra; se a palavra terminava em 'a', 'o', ou outra letra; prioridade alfabética da primeira letra; contagem de vogais na palavra, se a palavra contém acentos};
2. {Comprimento da palavra; se a palavra tem um número de letras par ou ímpar; prioridade alfabética da primeira letra; contagem de vogais na palavra, se a palavra contém acentos};
3. {Comprimento da palavra; se a palavra terminava em 'a', 'o', ou outra letra; contagem de vogais na palavra};
4. {Comprimento da palavra; se a palavra terminava em 'a', 'o', ou outra letra; contagem de vogais na palavra; se a palavra contém a letra 'a'};
5. {Comprimento da palavra; se a palavra contém a letra 'a'}.

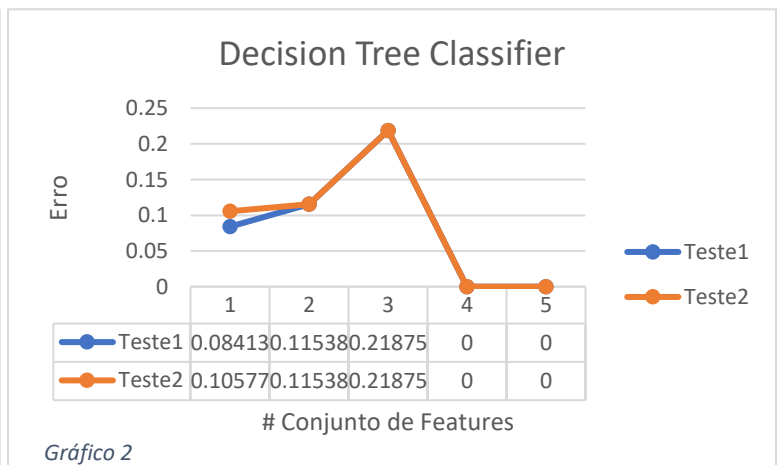
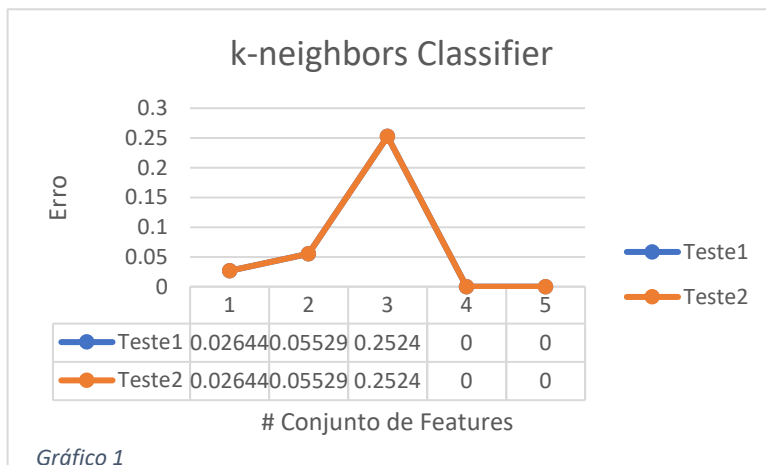
Features:

Foram testados vários de conjuntos de features, por forma a otimizar a classificação das palavras. Para isso, considerámos que tipo de características as palavras poderiam ter, e fizemos vários testes, os mais relevantes dos quais traduzimos no gráfico 1. Escolhemos o conjunto 5, porque acabou por ser o menor e mais eficiente a minimizar o erro de predição.

Classificador:

O classificador que escolhemos por última análise foi o *k-neighbors*, que demonstra melhor desempenho em relação à segunda hipótese do *decision tree*. Ao testarmos várias combinações de features, isto provou-se consistentemente verdade, mesmo quando, no ajuste final, os valores de erro convergiram os dois para zero. Isto é patente na comparação do gráfico 1 com o gráfico 2.

Os parâmetros ajustados foram o número de *neighbors*, $n_neighbors=5$, a ponderação usada pelo classificador ($weights='distance'$), e a sua função de métrica, $metric='manhattan'$, que calcula as distâncias da seguinte forma: $sum(|x - y|)$.



P2

O método de regressão usado foi o *Kernel Ridge*, com parâmetros $kernel='rbf'$, $alpha='0.001'$ e $gamma=0.1$, que devolveu os resultados mais precisos. Estes valores foram ajustados por forma a diminuir o erro, e resultaram nos gráficos 3 e 4 (teste 1 e 2). Assim, a regressão foi obtida com distâncias radiais, para uma relação não-linear entre as duas variáveis dadas.

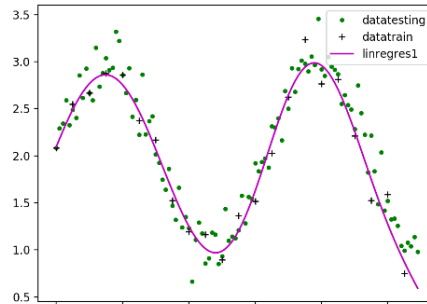


Gráfico 3

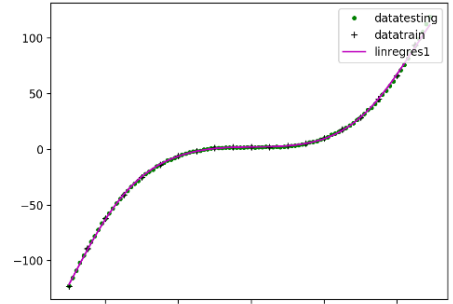


Gráfico 4

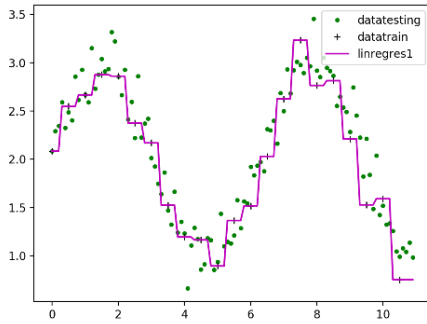


Gráfico 5

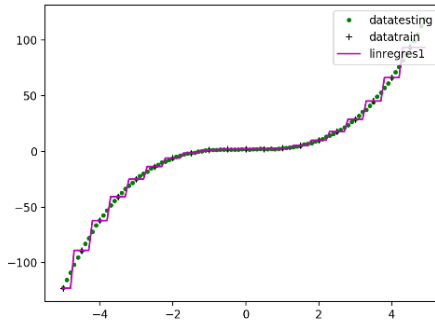


Gráfico 6

Outro método considerado e rejeitado foi o *decision tree*, com parâmetros $min_samples_split=2$, que julgámos inapropriado por não se ajustar tão adequadamente aos dados. O ajustamento foi feito de modo semelhante ao modelo anterior. O resultado dos testes está nas figuras 5 e 6.

P3

O ambiente do teste 1 é descrito pelo gráfico 7, e o do teste 2 pelo gráfico 8.

A função de recompensa, para o ambiente 1, está definida da seguinte maneira:

$$R1(s) = \begin{cases} 1, & s \in \{0, 6\} \\ 0, & s \in \{1, 2, 3, 4, 5\} \end{cases}$$

Para o ambiente 2, é:

$$R2(s) = \begin{cases} 1, & s \in \{0, 6\} \\ 0, & s \in \{1, 2, 3, 4, 5\} \end{cases}$$

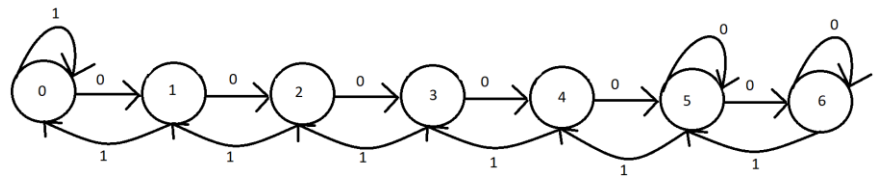


Gráfico 7

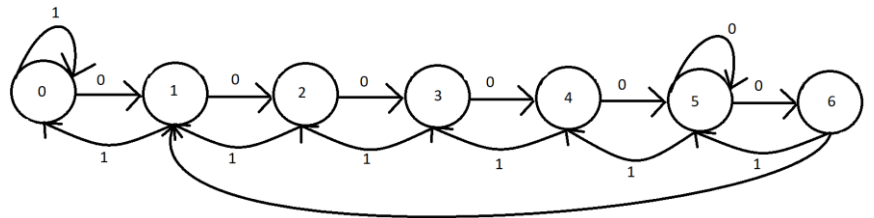


Gráfico 8

O agente move-se no ambiente de uma maneira que se assemelha a uma máquina de Turing: tendo como variáveis o estado em que se encontra e uma dada ação, muda para um estado diferente. A recompensa existe quando o agente está no estado 0 ou 6 e para cada um deles toma a ação 0 ou 1.