

Relatório do Segundo Projeto de IA

Programa que testa métodos de classificação, regressão e aprendizagem por reforço
Grupo: 038

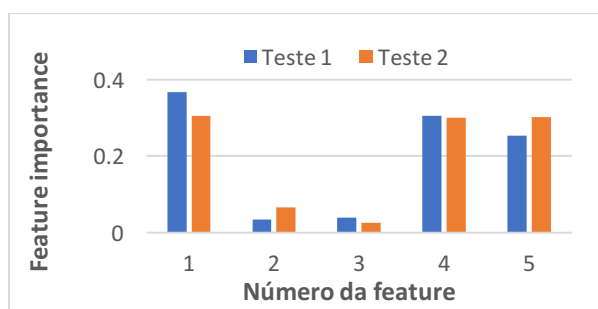
P1

Comparação das *features*

Estas são as nossas *features*:

1. Tamanho da palavra.
2. Número de vogais.
3. Número de caracteres repetidos.
4. Soma dos caracteres da palavra, onde 'a' = 0, ..., 'z' = 25.
5. "Ordem" dos caracteres da palavra, i.e., a soma dos caracteres multiplicando cada caracter pela sua posição na palavra.

A importância de cada *feature* varia minimamente entre os dois testes.



As *features* mais relevantes no nosso projeto são as 1ª, 4ª e 5ª, pois estas são algumas das características determinantes para descrever uma palavra.

Afinação dos parâmetros

O nosso objetivo é criar um modelo que prevê o valor (booleano) de uma palavra através da aprendizagem de simples regras de decisão pelas *features*. O método que melhor segue esta definição é o `DecisionTreeClassifier`; no entanto, tomámos a liberdade de testar mais dois outros classificadores, o `RidgeClassifier` e o `KNeighborsClassifier`.

De modo a evitar *overfitting*, decidimos usar validação cruzada *via GridSearchCV*, obtendo um método de classificação bem parametrizado.

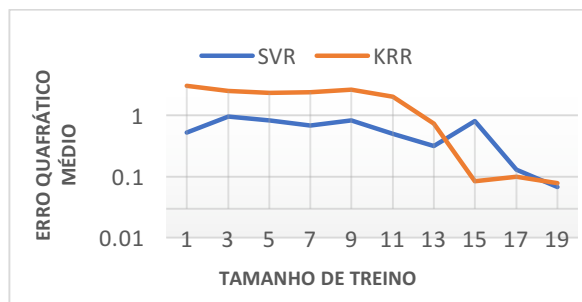
Os nossos resultados apontam que o `DecisionTreeClassifier`, apesar não ser sempre perfeito, obtém os erros de previsão mais baixos entre os três.

P2

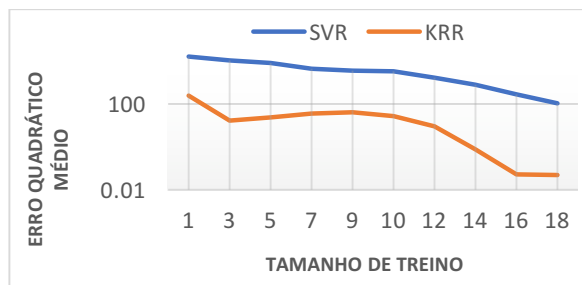
Comparação dos métodos escolhidos e afinação dos parâmetros:

Os métodos escolhidos foram o Kernel Ridge Regression e o SVR. Para a afinação dos parâmetros utilizou-se a validação cruzada, especificamente `GridSearchCV`. Escolheram-se os parâmetros a serem ajustados para cada método e deu-se uma lista de valores para *gamma* e *alpha*.

1º TESTE



2º TESTE



Como métrica de avaliação da qualidade da *prediction*, optou-se pelo erro quadrático médio. Quanto menor o erro melhor será o estimador.

A partir dos gráficos obtidos para os dois testes, chegou-se à conclusão de que o Kernel Ridge Regression é um melhor estimador pois, para todos os tamanhos de treino no 2º teste, o erro é significativamente menor e, apesar de no 1º teste o SVR ter um erro menor que o KRR na maioria dos tamanhos de treino, a diferença entre os erros de cada estimador não é significativa em comparação com os do 2º teste.

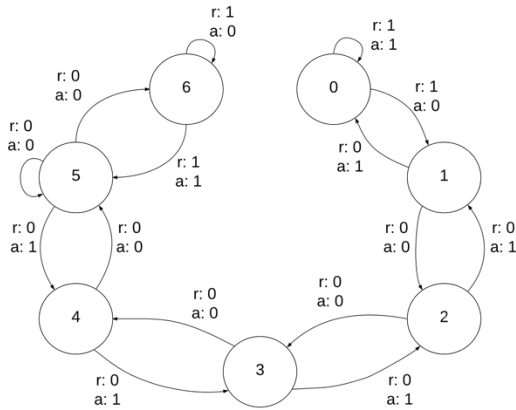
Relatório do Segundo Projeto de IA

Programa que testa métodos de classificação, regressão e aprendizagem por reforço
Grupo: 038

P3

1º Ambiente

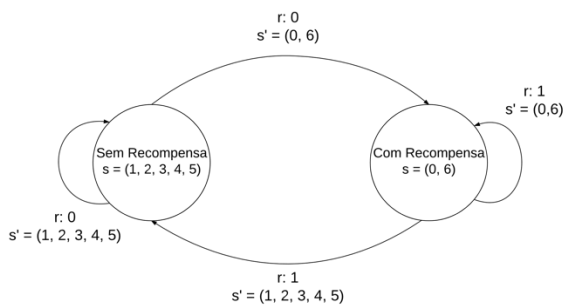
Representação gráfica do ambiente



Função de recompensa:

$$R(s, a) = \begin{cases} 0 & \text{if } s = (1, 2, 3, 4, 5), a = (0, 1) \\ 1 & \text{if } s = (0, 6), a = (0, 1) \end{cases}$$

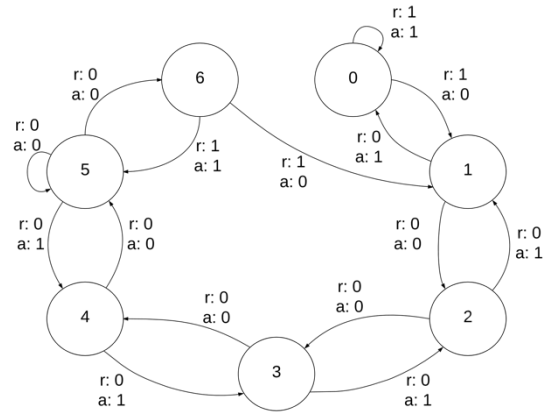
Como é que o agente se move?



Durante a análise da trajetória do agente foi possível observar-se um padrão. Desde o início do processo de aprendizagem que está à procura de um estado em que possa permanecer em ciclo a receber uma recompensa = 1. Representámos este comportamento por uma máquina de estados, onde generalizámos os 8 estados da representação gráfica do ambiente para somente 2 (com e sem recompensa), de maneira a que o agente quando encontra um estado cujo ciclo tem recompensa = 1 irá continuar a percorrer esse ciclo até ao fim da sua trajetória.

2º Ambiente

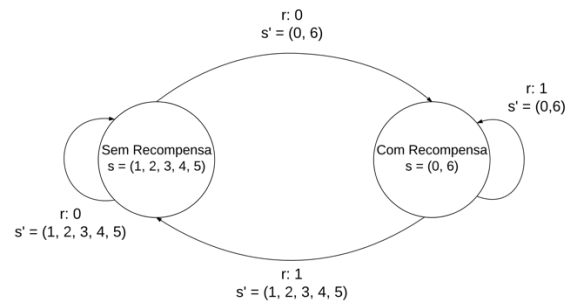
Representação gráfica do ambiente



Função de recompensa:

$$R(s, a) = \begin{cases} 0 & \text{if } s = (1, 2, 3, 4, 5), a = (0, 1) \\ 1 & \text{if } s = (0, 6), a = (0, 1) \end{cases}$$

Como é que o agente se move?



No 2º ambiente concluiu-se que o agente apresentou o mesmo comportamento referido no 1º ambiente.