

Inteligência Artificial - Algoritmos Genéticos

Trabalho Individual

1. Penalizando o fitness de mochilas infactíveis

a. Pseudocódigo

```
1 funcao checaPesoMochila(lista_objetos, lista_individuos)
2   se metodo = penalizar
3     mediaFitnessObjetos = 0
4     para cada objeto em lista_objetos
5       mediaFitnessObjetos = mediaFitnessObjetos + objeto.peso
6     fimpara
7
8     mediaFitnessObjetos = mediaFitnessObjetos / 42
9   fimse
10
11  para cada individuo em lista_individuos
12    se peso_mochila de individuo > que 120
13      se metodo = penalizar
14        fitnessPenalizado = individuo.fitness - mediaFitnessObjetos*(individuo.peso - 120)
15        individuo.fitness = fitnessPenalizado
16      fimse
17    fimse
18  fimpara
19
20  retorna lista_individuos
```

b. Explicação

Para penalizar os indivíduos com um peso maior que o permitido da mochila foi calculada a média de todos os fitness dos objetos na qual o resultado foi 9.8 de valor de média. Logo após o cálculo da média foi feita uma iteração na lista de indivíduos para checar o peso da mochila e se ultrapassa 120 kg. Se ultrapassar 120 kg então penalizo o fitness do individuo subtraindo do seu fitness a média geral de todos os fitness dos objetos (9.8) vezes o peso total do indivíduo menos o limite e com isso aplico o novo fitness penalizado. O retorno da função é a lista de indivíduos já atualizada com os indivíduos que excedem o peso de 120 penalizados.

2. Reparação de mochilas infactíveis

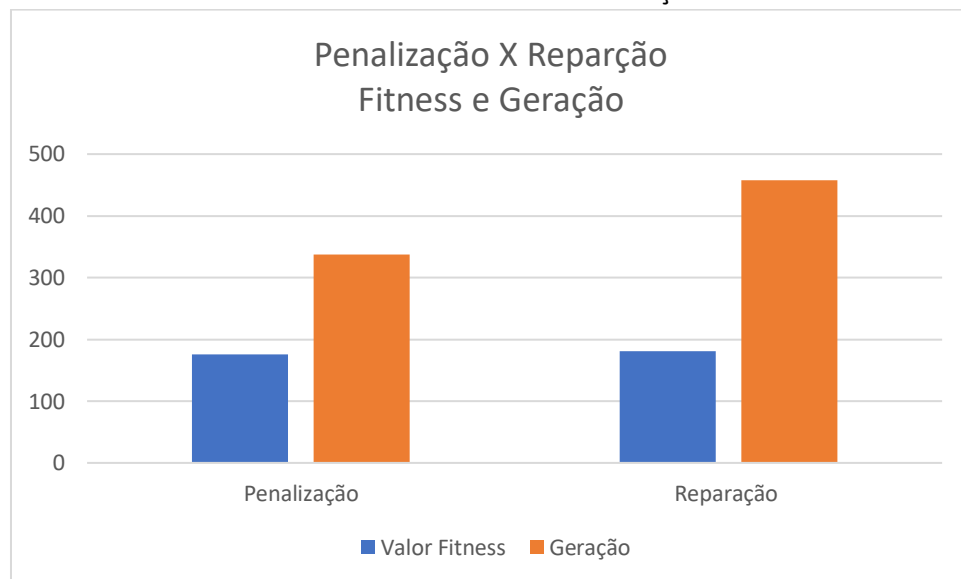
a. Pseudocódigo

```
1 funcao checaPesoMochila(lista_objetos, lista_individuos)
2   para cada individuo em lista_individuos
3     se peso_mochila de individuo > que 120
4       se metodo = reparar
5         somaPeso = 0
6         somaValor = 0
7         somaItens = 0
8
9         para cada gene em genes do individuo
10          se gene = 1
11            pesoComNovoObjeto = somaValor + lista_objetos.peso
12            se pesoComNovoObjeto <= 120
13              somaItens++
14              somaPeso = somaPeso + lista_objetos.peso
15              somaValor = somaValor + lista_objetos.valor
16            fimse
17          se pesoComNovoObjeto > 120
18            gene = 0
19          fimse
20        fimse
21      fimse
22    fimpara
23
24    individuo.fitness = somaValor
25    individuo.peso = somaPeso
26    individuo.numItems = somaItens
27  fimse
28  fimse
29  fimpara
30
31  retorna lista_individuos
```

b. Explicação

Para reparar os indivíduos com um peso maior que o permitido da mochila foi feita uma iteração na lista de indivíduos afim de checar se o peso ultrapassa 120 kg. Se ultrapassar um novo calculo de peso, fitness e numero de itens e inicia-se uma iteração na lista com os genes do indivíduo. Se o item está na mochila, ou seja, se o gene é igual a 1 é calculado qual será o peso da mochila adicionando este item. Se o peso for menor que 1 este gene continua com o valor 1 e o peso deste item e o seu valor são incrementados. Também é incrementado o número de itens da mochila. Porém se o peso exceder o limite de 120 kg o gene correspondente a este item é zerado e este item é retirado da mochila. O retorno da função é a lista de indivíduos já atualizada com os indivíduos que excedem o peso de 120 reparados.

3. Gráficos como melhor fitness – Valor do fitness X Geração



a. As curvas variam em função do modo de cálculo de fitness: penalização x reparação?

Sim, com a reparação o algoritmo encontrou a melhor solução na geração 458 da execução 6 e calculou uma solução com fitness de 181. Utilizando o método de penalização o algoritmo levou menos 100 gerações em comparação com a reparação totalizando 337 gerações e 1 execução para encontrar a melhor solução com 176 de valor de fitness.

4. Sobre as melhores soluções obtidas, responda para cada uma das implementações:

a. Qual foi o valor máximo para os itens de uma mochila que você encontrou (sem violar a capacidade em Kg da mochila)?

Fitness Reparação:181 - Fitness Penalização:176

b. Quantas mochilas com valor máximo foram obtidas?

Reparação: 1 - Penalização: 1

c. Liste todas as mochilas que obteve que apresentaram valor máximo. Para cada uma delas coloque os itens, valor total e peso total

Penalização

Mochila:0,0,0,1,0,1,1,1,1,0,1,0,0,1,1,0,0,0,0,0,0,0,1,0,0,0,1,1,1,0,0,0,1,0,0,1,0,0,0,1

Execucao:6

Mochila:0,0,0,1,1,1,0,1,1,0,1,1,0,0,1,0,0,1,1,0,0,1,1,0,0,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0

5. Compare a taxa de sucesso das implementações penalização e reparação. Para este problema, taxa de sucesso é o número de vezes que a solução de maior valor (possivelmente a ótima) foi encontrada nas execuções realizadas (recorda-se que o total de execuções foi definido no item a do método). Responda:

a. Quais foram as taxas de sucesso obtidas?

A melhor solução foi encontrada 1 vez utilizando o método de penalização. Já a melhor solução foi encontrada em 1 vez também utilizando o método de reparação. Foi utilizado um total de 500 gerações por execução e 12 execuções com sementes diferentes (totalizando 6000 avaliações de fitness), ela foi encontrada em diferentes execuções também porem somente a mesma mochila foi encontrada.

- b. Quantas vezes o cálculo de fitness é executado para a configuração em questão por execução? Escreva a fórmula

— —

- c. Qual método implementado é mais custoso temporalmente: o de reparação ou de penalização?

Reparação – Ambos os métodos possuem iteração na lista de indivíduos, porém na reparação há um custo maior devido a uma iteração a mais nos genes para poder reparar as mochilas retirando os objetos que podem exceder o peso da mochila.