

Algoritmos Genéticos

Fernando Lobo

Faculdade de Ciências e Tecnologia

Universidade Nova de Lisboa

Sumário

- O que é um algoritmo genético?
- Para que serve?
- Como funciona?
- Exemplo de aplicação
- Exercício na aula

O que é um algoritmo genético?

- Um algoritmo genético é um modelo computacional da selecção natural e da evolução.
- É inspirado nas teorias de evolução de Darwin.
- Evolução ? Selecção natural ? Darwin ?
- ... o que é que isso tem a ver com optimização ?

- A evolução é um processo de optimização em que os seres se vão adaptando ao meio ambiente; os mais fortes sobrevivem, os mais fracos morrem.
- Os algoritmos genéticos são um modelo da evolução e da selecção natural, e servem para resolver problemas de optimização.
- Um algoritmo genético tem 2 componentes essenciais:
 - sobrevivência do mais forte
 - recombinação

De que modo é que os algoritmos genéticos diferem de outras técnicas?

- Trabalham com uma população de soluções, em vez de uma só solução.
- Usam regras de transição probabilísticas em vez de regras determinísticas.

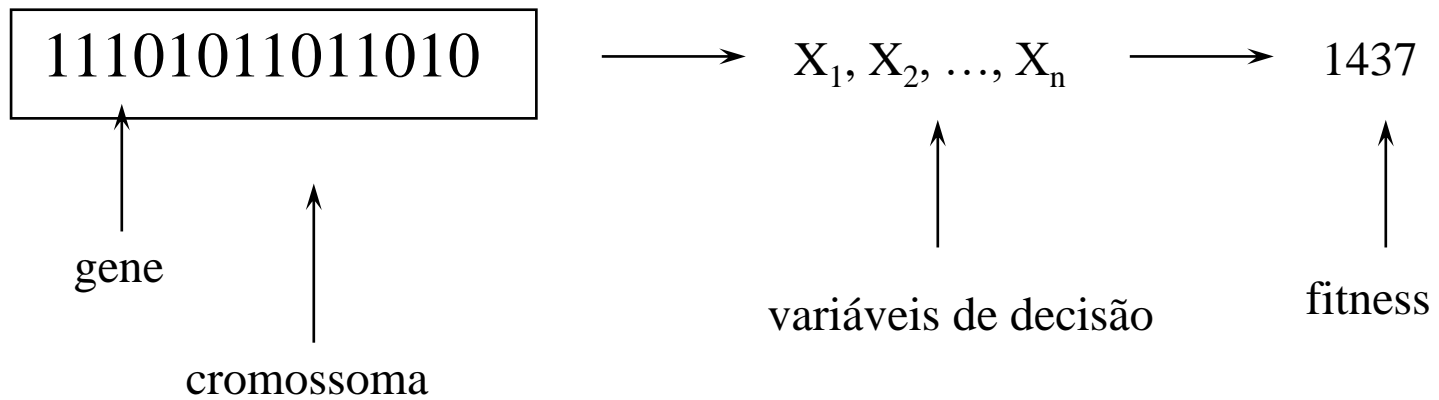
Quando é que se deve usar o algoritmo genético?

- Quando o “search space” é muito grande (nestes casos, a enumeração de todas as possíveis soluções torna-se impracticável) ...
- ... e quando os métodos tradicionais (ex: programação linear) não são aplicáveis.

Codificação

- Uma solução do problema é codificada numa estrutura (tradicionalmente, uma sequência de dígitos binários).
 - 10011
 - $1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 16 + 2 + 1 = 19$
 - 0 (00000) e 31 (11111) com cinco dígitos binários
- A cada estrutura (solução) é associado um valor numérico (fitness) que representa a qualidade dessa estrutura.
- O valor de “fitness” é obtido através da função objectivo.

Terminologia

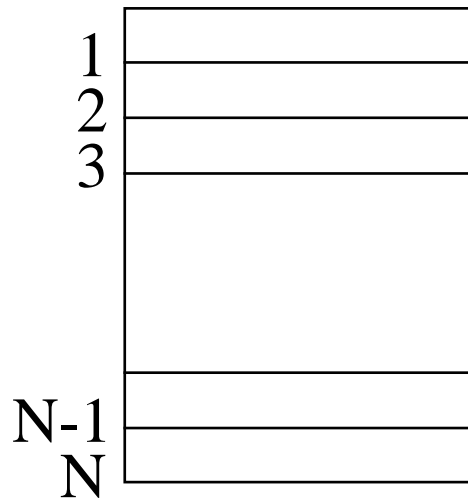


Como funcionam?

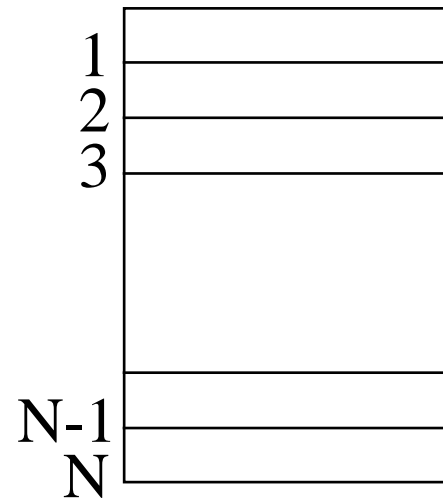
- 1) Inicializar aleatoriamente uma população com N indivíduos
- 2) Calcular fitness de todos os indivíduos da população
- 3) Criar uma nova população através do operador de selecção
- 4) Efectuar “crossing-over” entre cada par de indivíduos
- 5) Efectuar mutação em cada gene, com probabilidade P_m
- 6) Voltar ao passo 2)

Uma iteração do algoritmo genético

Geração no instante t



Geração no instante $t+1$



Seleccção, Recombinação e Mutação

Seleccção

- Simula o conceito de “survival of the fittest”.
- Os mais fortes vão ter oportunidade de se reproduzirem.
- Os mais fracos tendem a desaparecer da população.
- Existem vários métodos para implementar este operador (ex: proporcional, ranking, torneio, e outros...)

Exemplo: selecção usando torneio binário

- Escolhe-se um par de indivíduos da população.
- O melhor sobrevive, o pior morre.
- Repete-se este processo N vezes (N é o tamanho da população).

Exemplo de selecção com torneio binário

Antes

estrutura	fitness
A	8
B	2
C	4
D	5



Depois

estrutura	fitness
A	8
C	4
A	8
D	5

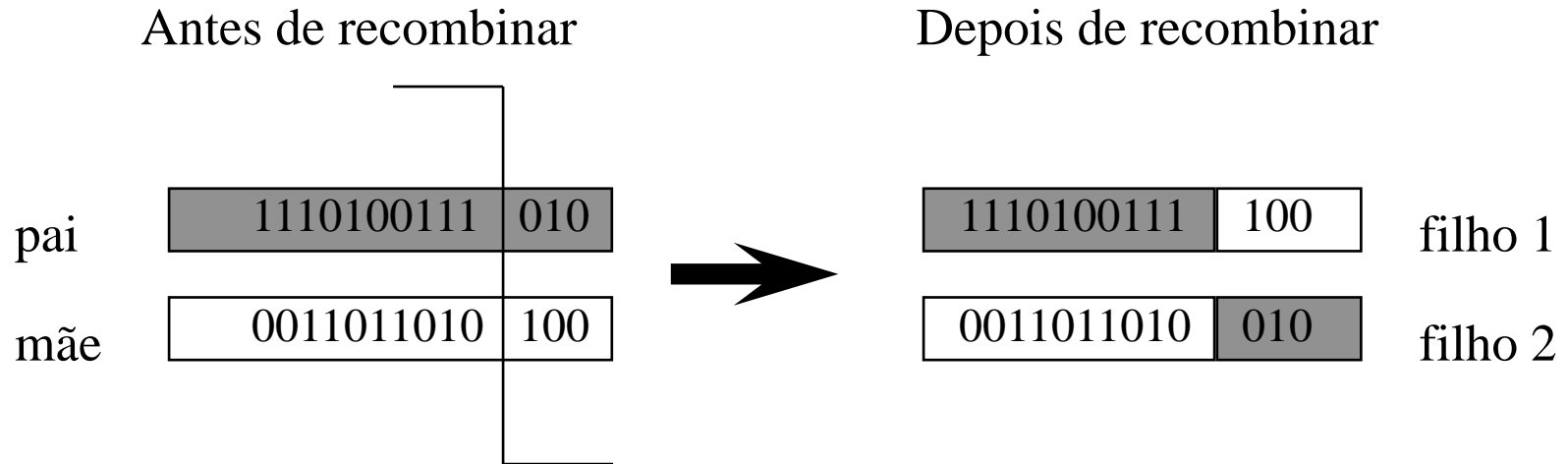
4 torneios: (A,D) (B,C) (A,B) (C,D)

4 vencedores: A C A D

“Crossing-over”

- Recombina-se 2 indivíduos (pai e mãe) e obtém-se 2 novos indivíduos (os filhos).
- No exemplo anterior, A recombina-se com C, e A recombina-se com D.

Exemplo de crossing-over



Mutação

- Com probabilidade P_m , mudar um gene de 0 para 1, ou de 1 para 0.
- Este operador costuma ser usado com uma probabilidade muito baixa (ex: 0.001).
- É útil para manter a diversidade na população

Antes de mutação

1100101111



Depois de mutação

1100101101



gene 9 sofreu uma mutação

Exemplo do problema da Dona Teresa

A Dona Teresa vende merendas e croissants com chocolate. Cada merenda vendida dá um lucro de 50\$00, e cada croissant dá um lucro de 60\$00. Logo de manhã, ela está disposta a perder um máximo de 2 horas para preparar as merendas e os croissants. Em média, ela demora 2 minutos para fazer um croissant, e 1 minuto para fazer uma merenda (nota: este tempo não inclui o tempo de cozedura, porque durante esse tempo a Dona Teresa está a fazer outras coisas). Tanto os croissants como as merendas necessitam de farinha para fazer a massa. Cada croissant necessita de 10 gramas de farinha, e cada merenda necessita de 30 gramas. Hoje, a Dona Teresa não teve tempo de reforçar o seu stock e dispõe apenas de 1,8 Kg de farinha. A Dona Teresa tem ainda o problema do espaço. O número máximo de merendas e croissants que ela consegue guardar é 70 (um croissant ocupa o mesmo espaço que uma merenda). Quantas merendas e quantos croissants é que a Dona Teresa deve fazer de modo a maximizar o seu lucro? (assume-se que a Dona Teresa consegue vender todos os croissants e merendas que fizer).

Exemplo do problema da Dona Teresa

- Função objectivo: $50 X_1 + 60 X_2$
- Tamanho do cromossoma: 12
(6 dígitos binários para X_1 e 6 para X_2)
- 001101101001 ---> 13 merendas e 41 croissants

Restrições

- Selecção com torneio binário:
 - Se os dois indivíduos são admissíveis, ganha o melhor dos dois.
 - Se só um dos indivíduos é admissível, ganha aquele que é admissível.
 - Se nenhum dos dois indivíduos é admissível, ganha aquele que viola menos as restrições.

ATENÇÃO !

a Dona Teresa não precisa de algoritmos genéticos

- Para resolver o problema da D.Teresa, nunca se iria utilizar um algoritmo genético. O problema da D.Teresa resolve-se com programação linear.
- ... e mesmo que não fosse linear, fazia-se uma busca exaustiva e descobria-se a solução óptima. O “search space” não é grande ($2^{12} = 4096$).
- Calcular 4096 vezes a função objectivo é trivial para um computador.

Como terminar o algoritmo genético?

- Após um pré-determinado número de gerações.
- Quando não houver melhoras durante um pré-determinado número de gerações.

Valor para o tamanho da população N ?

- Quanto maior for N , maior é a chance de se encontrar uma melhor solução...
- ... mas quanto maior for N , mais tempo o algoritmo genético demora.
- Existe o compromisso qualidade da solução versus tempo.

Por que é que funcionam?

- Intuitivamente, podemos fazer um paralelo com o modo como os seres humanos são criativos e inovadores.
- Os seres humanos não criam ideias novas a partir do nada.
- Os seres humanos são criativos e inovadores quando combinam noções que funcionam bem num contexto, com noções que funcionam bem noutro contexto.

- Do mesmo modo, os algoritmos genéticos são inovadores quando combinam um bocado de uma solução que funciona bem com um bocado de outra solução que também funciona bem.

“A good scientist is like a good cartoonist” --- J. Holland

- O algoritmo genético é um modelo muito simples da evolução. Mas a simplicidade é o seu ponto forte.
- A realidade é muito complexa para modelar exactamente.
- Um bom modelo é aquele que exagera as características principais e ignora grande parte dos detalhes.
- ... é como o trabalho de um caricaturista. Em meia dúzia de traços salienta-se o principal; os detalhes ignoram-se.

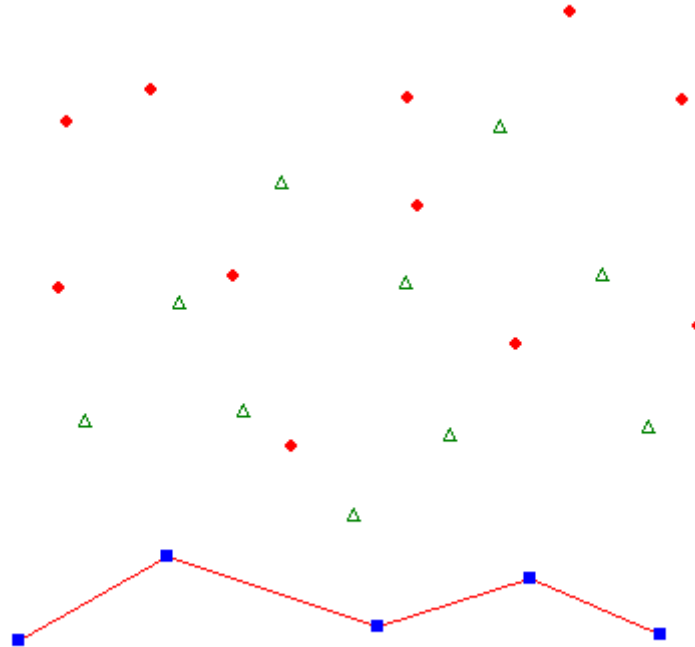
Um pouco de história

- Os algoritmos genéticos foram inventados por John Holland no final dos anos 60 na Uni. de Michigan.
- Nos anos 80 e 90 apareceram as primeiras aplicações de algoritmos genéticos em problemas reais.
- Ainda se faz muita investigação para melhorar o funcionamento dos algoritmos genéticos.
- Estes algoritmos estão na sua infância.

Exemplos de aplicação

- Design da turbina de gás do Boing 747.
- Problema de expansão de redes.
- Detecção de criminosos.
- e muito mais...
- Os algoritmos genéticos têm aplicação em quase todas as áreas de engenharia, e também em outras áreas tais como finanças, medicina, e até mesmo nas artes.

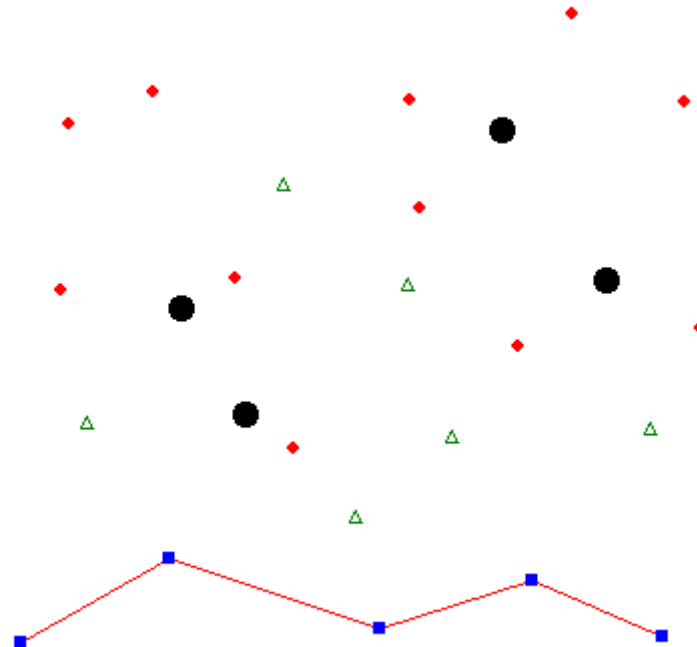
Exemplo: problema de expansão de redes.



10 variáveis de decisão binárias (10 triângulos).

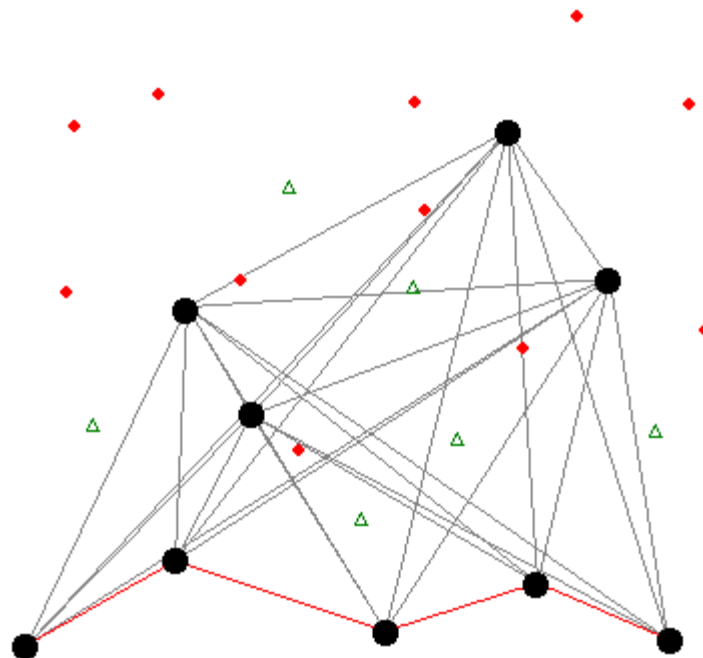
Se expandir a rede elétrica de modo a abastecer as casas (círculos vermelhos). As casas
podem ser ligadas directamente a fontes de energia (quadrados azuis), ou podem ser ligadas primeiro
a transformadores (triângulos). O objectivo é expandir a rede da maneira mais económica possível.

Expansão de redes: função objectivo em 3 passos.

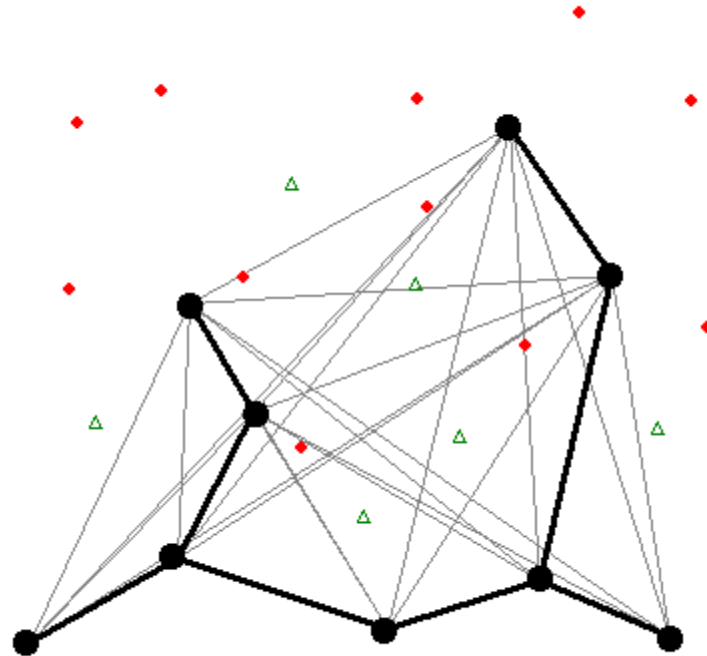


Solução: 0110000110

Expansão de redes: passo #1

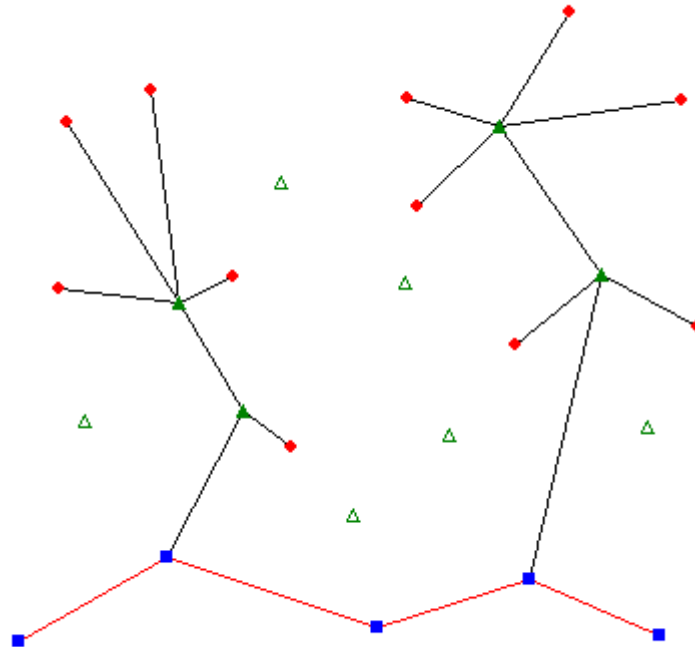


Expansão de redes: passo #2



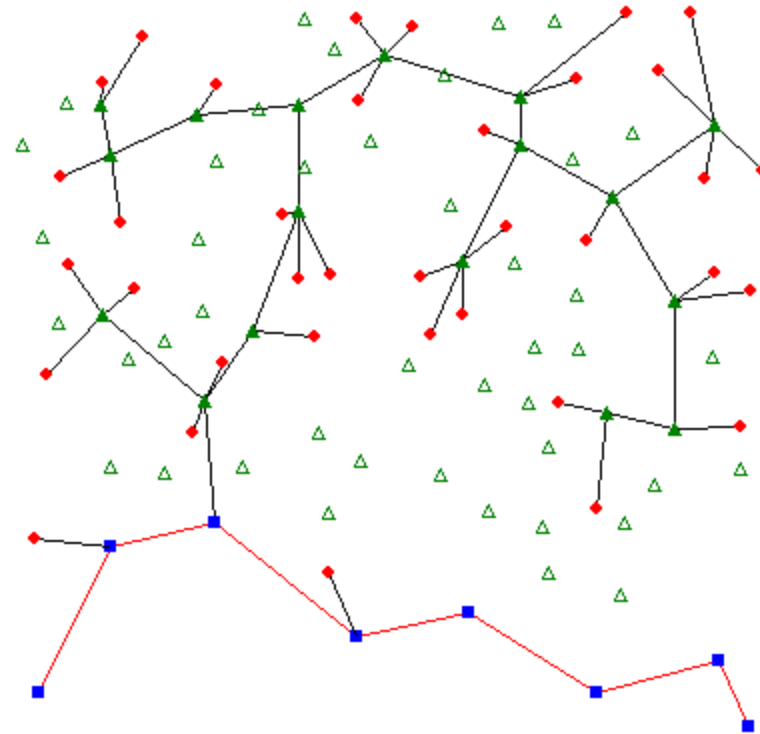
“minimum spanning tree”

Expansão de redes: passo #3



Rede correspondente à solução 0110000110

Algoritmo genético descobriu esta solução passado algumas gerações



Um exemplo com 60 variáveis de decisão binárias.
 $2^{60} \approx 10000000000000000000$ de hipóteses possíveis!

Exercício na aula

Jogo Vector

Um membro do grupo escreve um conjunto de seis dígitos binários (0 ou 1). Os restantes membros sugerem quatro conjuntos de seis dígitos binários sem conhecerem a proposta do outro membro. O objectivo é encontrar o conjunto original a partir desses quatro conjuntos no menor numero de passos possível utilizando algoritmos genéticos.