|  |
| --- |
|  |
| Algorithme génétique |
|  |

|  |
| --- |
| BOUNATIROU Rodolphe  26/08/2023 |

Table des matières

[Introduction 2](#_Toc143965406)

[Pourquoi génétique 2](#_Toc143965407)

[La sélection naturelle 3](#_Toc143965408)

[Principe de sélection naturelle 3](#_Toc143965409)

[Comment faire une sélection 4](#_Toc143965410)

[Précision sur le fitness 5](#_Toc143965411)

[Détail sur le crossover 6](#_Toc143965412)

[Details sur la mutation 7](#_Toc143965413)

[Exemple de sélection naturelle 8](#_Toc143965414)

[Selection par rang 8](#_Toc143965415)

[Selection par tournois 8](#_Toc143965416)

[Selection par roulette 8](#_Toc143965417)

# Introduction

## Pourquoi génétique

Les **algorithmes génétiques** appartiennent à la catégorie des **algorithmes évolutionnistes**.  
Ils servent à **répondre à des problèmes ou il n’y a pas de méthode exactes** (ou que la solution est inconnue) pour **le résoudre en un temps raisonnable**.

Il se base sur une notion de sélection naturelle, en l’appliquant à une **génération** (**population**) de **solution** donnée (**individus**).

# La sélection naturelle

## Principe de sélection naturelle

La sélection naturelle va permettre de sélectionner les individus les plus viables de notre génération,  
cette sélection des individus les plus viable peut différer d’un algorithme à l’autre toutefois le but est toujours le même, faire se reproduire les individus les plus viable afin de générer une nouvelle génération qui conservera une partie du patrimoine génétique des meilleurs individus.  
  
Comme en informatique les gêne n’existent pas réellement, il faut voir le gêne comme les données qui caractérise notre instance, en somme ses attributs.

## Comment faire une sélection

Comme vu précédemment on va chercher à sélectionner nos individus càd nos solutions à un problème donné selon leur viabilité.

Pour cela il nous faut pouvoir quantifier cette « viabilité » de nos individus, avec un score qu’on appelle le **fitness.**

Grace à cela on va créer des croisements entre nos solutions afin de générer de nouveaux individus dont les **gênes,** càd les **attributs** de notre classe, correspondent au **croisement des gêne** de ses parents,  
ainsi chacun des parents transmets une portion plus ou moins grande de son patrimoine génétique.  
Ce principe s’appelle le **crossover.**

Ensuite on fait varier les gênes selon une certaine probabilité afin de ne pas rester bloquer et de pouvoir explorer de nouvelles solutions quand les patrimoines génétiques de nos individus sont très proches, ce principe s’appelle la **mutation.**

## Précision sur le fitness

Le fitness étant une évaluation de la validité de notre individu par rapport à un problème donné.  
La connaissance du contexte est importante.

Voici donc un contexte :

Sur un échiquier, on place 8 reines (se déplaçant sur l’ensemble de la ligne, de la colonne et des diagonales).  
Le but de notre algorithme génétique sera d’aboutir à un échiquier sur lequel aucune de nos reines ne gêne le déplacement d’une autre.  
Notre solution (ou individu) contient donc en attribut la position de chacune des reines.

## Détail sur le crossover

Imaginons que nous devions transmettre un gêne comme un tableau de booleen.  
Ce tableau à une taille fixe (de longueur 8) et similaire sur nos deux individus.  
Voici les données de ses deux tableaux de nos parents…

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | b0 | b1 | b2 | b3 | b4 | b5 | b6 | b7 |
| Parent 1 | **0** | **1** | **1** | **0** | **1** | **1** | **0** | **0** |
| Parent 2 | **1** | **0** | **0** | **0** | **1** | **0** | **1** | **1** |

On va établir un point de croisement pour nos gênes à partir duquel s’effectuera notre crossover, pour exemple nous allons faire un crossover à partir du 2ème bit de poids fort (càd b1),  
le croisement s’effectuera à droite de notre bit ainsi 2 enfants peuvent être possibles.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | b0 | b1 | b2 | b3 | b4 | b5 | b6 | b7 |
| Enfant 1 | **0** | **1** | **0** | **0** | **1** | **0** | **1** | **1** |
| Enfant 2 | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **0** |

En fonction du parent de poids fort l’individu enfant généré par notre crossover variera…  
Si le parent 1 est parent de poids fort, le crossover générera l’enfant 1, dans le cas contraire l’enfant 2.

Ceci est un crossover simple on peut créer des crossover plus complexes avec plusieurs point de croisement.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Exemple de crossover plus complexe (croisement b1 & b6) | | | | | | | | |
|  | b0 | b1 | b2 | b3 | b4 | b5 | b6 | b7 |
| Parent 1 | **0** | **1** | **1** | **0** | **1** | **1** | **0** | **0** |
| Parent 2 | **1** | **0** | **0** | **0** | **1** | **0** | **1** | **1** |
|  | | | | | | | | |
| Enfant 1 | **0** | **1** | **0** | **0** | **1** | **0** | **1** | **0** |
| Enfant 2 | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |

## Details sur la mutation

Maintenant que nous avons vu le principe de crossover,

Imaginons un crossover, sur ces mêmes gênes mais avec des gênes très proches entres nos individus.  
et faisons un croisement au même emplacement

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **b0** | **b1** | **b2** | **b3** | **b4** | **b5** | **b6** | **b7** |
| Parent 1 | **1** | **1** | **1** | **0** | **1** | **1** | **1** | **0** |
| Parent 2 | **1** | **1** | **1** | **0** | **1** | **1** | **0** | **1** |
| **Résultats crossover** | | | | | | | | |
| E1 | **1** | **1** | **1** | **0** | **1** | **1** | **0** | **1** |
| E2 | **1** | **1** | **1** | **0** | **1** | **1** | **1** | **0** |

Le croisement avec le parent 1 en poids fort donne E1 qui est similaire au parent 2  
et réciproquement lorsque le parent 2 deviens parent de poids fort on obtient le parent 1.  
  
Dans cette situation, sans la mutation si tous les individus sélectionnés pour faire des crossover ont un patrimoine génétique proche, il deviendra très compliqué pour les nouvelles générations de progresser et explorer de nouvelles possibilités.  
  
Ainsi, pour chaque gêne (ici nos 8 bits), ont va, avec une certaine **probabilité**, pouvoir inverser chacun de nos bits, représentant nos gênes

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Ex : Mutation du bit b3 de notre crossover** | | | | | | | | |
|  | **b0** | **b1** | **b2** | **b3** | **b4** | **b5** | **b6** | **b7** |
| Avant | **1** | **1** | **1** | **0** | **1** | **1** | **0** | **1** |
| Apres | **1** | **1** | **0** | **0** | **1** | **1** | **0** | **1** |

**Notons que face à cette probabilité la mutation est aléatoire. Et par conséquent le nombre de gêne qui vont subir une mutation n’est pas déterminé.**  
  
Grace à cela de nouvelles solutions sont possibles et les générations ayant des patrimoines proches entre les individus peuvent ainsi s’améliorer et tendre à évoluer.

## Exemple de sélection naturelle

Bien que l’algorithme qui va permettre d’effectuer la sélection peut être personnalisé.  
Voici quelques façons de sélectionner les individus, qui sont fréquemment utilisé au sein des algorithmes génétiques.

### Selection par rang

Une des approches les plus simples et la séléction par rang.  
Elle consiste à faire au sein de notre population un tri afin de prendre les solutions par ordre de viabillité et de se récupérer les n individus ayant les solutions les plus viables afin de se servir d’eux pour constituer la base de notre future génération.

### Selection par tournois

Sans trier au préalable nos individus ont prends aux hasard parmis nos individus un sous ensemble de n individus, et on récupére à chaque fois le gagnant du tournoi, càd celui qui dans ce sous ensemble est le plus viable.  
On répéte cette oppération de solution un certain nombre de fois afin de constituer la base de notre nouvelle génération.

Le problème c’est que si notre sous ensemble contient trop peu d’individus ont risque d’introduire des solutions peu viables dans notre future génération,

Et qu’avec un sous ensemble trop grand (imaginons 80% de la pop) on risque de ne pas avoir beaucoup de diversité dans nos solutions pour la génération suivante.

### Selection par roulette

Pour cette sélection il faut trier les individus par score.

On va assigner à chacun une probabilité d’etre séléctionné,  
pour cela on peut par exemple faire le score de notre individu divisé par l’ensemble des scores de nos individus de la génération.