Algoritmi genetici v.0

Alessio Marchetti

1 Introduzione

Gli algoritmi genetici (GA) sono algoritmi di ricerca basati sulle meccaniche della selezione naturale e della genetica. Essi si basano sulla manipolazione di una popolazione di individui, ciascuno identificato da una propria stringa, che ne definisce il comportamento. Tale stringa può essere assimilata al DNA. Sulla popolazione vengono essenzialmente eseguiti tre tipi di azioni:

1. *riproduzione*: Da una generazione (ovvero una popolazione in un certo istante) si selezionano gli individui più adatti alla sopravvivenza e si portano alla generazione successiva.

2. *crossover*: Gli individui di una popolazione scambiano fra di loro porzioni di DNA.

3. *mutazione*: Sezioni di DNA variano casualmente, con una frequenza fissata generalmente molto bassa.

2 Simulazione "a mano"

Per avere un'idea del funzionamento di un GA mostro un esempio di funzionamento molto semplice e di cui è ben nota una soluzione. Si tratta di trovare il massimo e il punto di massimo della funzione $f(x) = x^2$ nell'intervallo $[0, 31]$.

Come prima cosa è necessario definire un vocabolario che andrà a comporre le stringhe del DNA. Lo faccio nella maniera più semplice possibile, ovvero

$$V = \{0, 1\}$$

Inoltre definisco un individuo come un elemento dell'insieme V^5. Per esempio è un individuo `01001`. Ogni individuo codifica un certo valore di x, che per comodità scelgo essere la sua rappresentazione in base decimale. Il valore corrispondente a quello scelto in precedenza sarà dunque 9.

Per l'implementazione di un GA è inoltre necessaria una funzione detta di *fitness*, tale che, dato un individuo, ne indichi la sua idoneità. Nell'esempio preso in esame è spontaneo scegliere f stessa. Quindi `01001` ha un fitness di $f(9) = 9^2 = 81$.

In questo esempio scelgo di avere una popolazione composta da $N = 4$ individui. In genere questo numero è molto più grande per avere un algoritmo efficiente. La popolazione iniziale è generata casualmente, ovvero ogni lettera di ogni stringa è il risultato di un lancio di moneta. Ottengo la seguente popolazione, con relativo fitness.

n	Stringa	Valore x	fitness
1	`01101`	13	169
2	`11000`	24	576
3	`01000`	8	64
4	`10011`	19	361
	totale		1170

2.1 Riproduzione

A questo punto voglio generare una nuova generazione a partire dagli individui con fitness maggiore. Per fare ciò ad ogni individuo i_k assegno la probabilità di riproduzione

$$p_k = \frac{f(k)}{\sum\limits_j f(j)}$$

dove al numeratore compare il fitness di tale individuo e a denominatore la somma di tutti i fitness. Risulta chiaro che a maggiore fitness corrisponde maggiore probabilità di riproduzione e che la somma di tutti i p_k valga 1. La tabella risulta dunque essere

k	Stringa	Valore x	fitness	p_k
1	`01101`	13	169	0.14
2	`11000`	24	576	0.49
3	`01000`	8	64	0.05
4	`10011`	19	361	0.31
	totale		1170	1.00

Una volta determinate le probabilità, scelgo casualmente gli individui per la nuova popolazione. Ciascuno di essi ha probabilità p_k di essere uguale all'individuo i_k. Su una popolazione di N individui ci aspettiamo quindi di avere Np_k individui uguali a i_k. Se si definisce il fitness medio

$$\overline{f} = \frac{\sum\limits_j f(j)}{N}$$

si ha che

$$Np_k = N\frac{f(k)}{\sum_j f(j)} = \frac{f(k)}{\overline{f}}$$

Questo significa che se un certo individuo ha un fitness superiore alla media, ovvero $f > \overline{f}$, avrà $f/\overline{f} > 1$, cioè tenderà ad aumentare il suo numero di copie nella generazione successiva. Analogamente un individuo con un fitness inferiore alla media tenderà a diminuire il proprio numero di copie.

Vado dunque a eseguire la riproduzione sulla popolazione in esame, ottenendo i seguenti risultati.

k	Stringa	Valore x	fitness	p_k	$f/\overline{f}$	numero di individui nella nuova generazione
1	`01101`	13	169	0.14	0.58	1
2	`11000`	24	576	0.49	1.96	2
3	`01000`	8	64	0.05	0.21	0
4	`10011`	19	361	0.31	1.23	1
	totale		1170	1.00	4.00	
	media		229.5	0.25	1.00	
	massimo		576	0.49	1.96	

2.2 Crossover

Supponiamo in una certa popolazione di avere individui con stringhe composte da un numero l di lettere tratte dal vocabolario V. Nel nostro esempio $l = 5$. Indico con i_k^j la j-esima lettera della stringa dell'individuo i_k. Vediamo cosa succede eseguendo un crossover su due individui i_1 e i_2. Scelgo casualmente un un intero c compreso tra 1 e $l-1$ detto sito di crossover. Il risultato del crossover è un individuo i'_1 tale che $i'^n_1 = i^n_1$ se $n \leq c$ e $i'^n_1 = i^n_2$ altrimenti. Allo stesso modo viene prodotto un individuo i'_2 tale che $i'^n_1 = i^n_2$ se $n \leq c$ e $i'^n_1 = i^n_1$ altrimenti. In altri termini da un certo punto in poi le due stringhe si scambiano tra di loro. Per eseguire un crossover su un'intera popolazione, si devono accoppiare casualmente gli individui, e poi eseguire il crossover su ciascuna coppia.

Partiamo dalla popolazione derivante dalla riproduzione e accoppiamo casualmente gli individui. Scegliamo inoltre casualmente il sito di crossover.

k	Stringa	Compagno	Sito di crossover
1	`01101`	2	4
2	`11000`	1	4
3	`11000`	4	2
4	`10011`	3	2

Andiamo a studiare la prima coppia. la prima stringa si spezza come `0110-1`, mentre la seconda come `1100-0`. Si otterranno dunque i nuovi individui `01100` e `11001`. Si procede in modo analogo sulla seconda coppia per ottenere `11011` e `10000`.

2.3 Mutazione

Durante la mutazione ciascuna lettera di ogni stringa ha un probabilità di $\varepsilon = 0.001$ di diventare un'altra lettera del vocabolario. In questo caso ci aspettiamo $Nl\varepsilon = 4 \cdot 5 \cdot 0.001 = 0.02$ mutazioni. Di fatto andando a simulare non ne troviamo alcuna.