

Intelligenza Artificiale

Anno Accademico 2022 - 2023

***Introduzione agli
Algoritmi Genetici***

Norme di utilizzo dei materiali didattici

La visione e l'utilizzo del presente materiale didattico è riservato agli utenti iscritti al corso al solo fine di studio e approfondimento didattico.

L'utente che accede alla sezione e-learning e che ne consulta i contenuti è tenuto a rispettare le disposizioni legislative che tutelano il diritto d'autore e pertanto è fatto espresso divieto di riprodurre, pubblicare o distribuire i materiali tratti dal presente sito, anche in forma parziale, fatta salva la possibilità di realizzare un'unica copia del materiale, in formato cartaceo e/o digitale, all'esclusivo fine di studio.

L'utente è responsabile per l'uso improprio o non autorizzato del materiale pubblicato effettuato in violazione delle suddette disposizioni.

SOMMARIO

- Introduzione
- Evoluzione darwiniana
- Analogia computazionale: Algoritmi Genetici
- Genotipo e popolazione
- Funzione di adattamento
- Selezione, crossover, mutazione
- Vantaggi e limiti degli Algoritmi Genetici

INTRODUZIONE

- Traendo ispirazione dalla teoria dell'evoluzione naturale di Charles Darwin, una delle tecniche più affascinanti per la risoluzione dei problemi è la famiglia di algoritmi chiamata **calcolo evolutivo**.
- All'interno di questa famiglia, il ramo più importante e ampiamente utilizzato è noto come **Algoritmi Genetici**.

INTRODUZIONE

- Imitando il processo di selezione e riproduzione naturale, gli **Algoritmi Genetici** possono produrre soluzioni di alta qualità per vari problemi che coinvolgono la ricerca, l'ottimizzazione e l'apprendimento.
- Allo stesso tempo, la loro analogia con l'evoluzione naturale consente agli **Algoritmi Genetici** di superare alcuni degli ostacoli incontrati dagli algoritmi di ricerca e ottimizzazione tradizionali, soprattutto per problemi con un gran numero di parametri e rappresentazioni matematiche complesse.

EVOLUZIONE DARWINIANA

Gli Algoritmi Genetici implementano una versione semplificata dell'evoluzione darwiniana che avviene in natura. I principi della teoria dell'evoluzione darwiniana possono essere riassunti utilizzando i seguenti principi:

- **Variazione**
- **Ereditarietà**
- **Selezione**

EVOLUZIONE DARWINIANA

Un importante fattore abilitante dell'evoluzione è il **crossover** o **ricombinazione**, in cui la prole viene creata con un mix dei tratti dei genitori.

Il crossover aiuta a mantenere la diversità della popolazione e a riunire i tratti migliori nel tempo.

Inoltre, le **mutazioni** - variazioni casuali nei tratti - possono svolgere un ruolo nell'evoluzione introducendo cambiamenti che possono comportare un balzo in avanti ogni tanto.

ANALOGIA COMPUTAZIONALE: ALGORITMI GENETICI

- Gli Algoritmi Genetici cercano di trovare una soluzione ottima per un dato problema.
- Mentre nella evoluzione darwiniana viene mantenuta una popolazione di individui di una certa specie, negli Algoritmi Genetici viene mantenuta una popolazione di “soluzioni candidate”, chiamate **individui**, per il problema in questione.
- Tali soluzioni candidate sono valutate ed usate per creare una nuova generazione di possibili soluzioni.
- Esempi di “soluzioni candidate” possono essere: le posizioni sulla scacchiera per il problema delle 8 Regine, i percorsi nel TSP, ecc. ecc.

ALGORITMI GENETICI

GENOTYPE

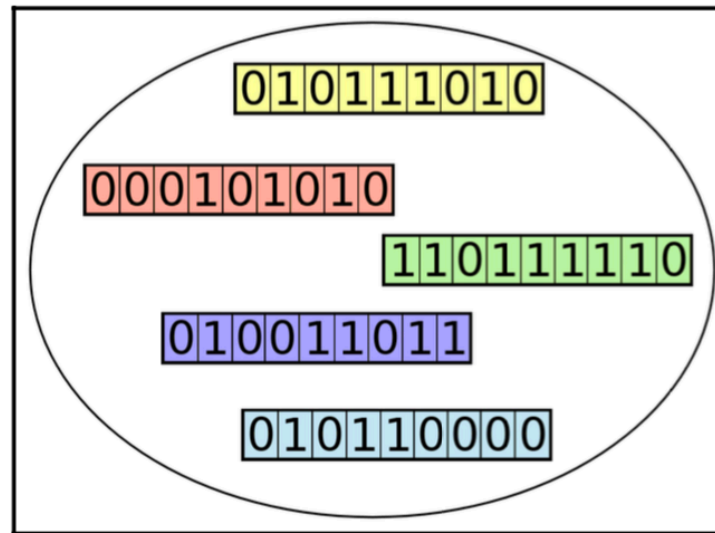
- In natura il **genotipo** è un insieme di **geni** raggruppati in **cromosomi** (stringa sull'alfabeto **ACGT**).
- Se due esemplari si accoppiano per generare figli, ogni cromosoma dei figli sarà costituito da un mix dei geni derivanti da entrambi i genitori.
- Ispirandosi a questo concetto, negli Algoritmi Genetici ogni individuo è rappresentato da un cromosoma (costituito da una collezione di geni) che può essere rappresentato da una stringa su un alfabeto finito, ad esempio da una stringa di bit, dove ogni bit rappresenta un **gene**:

0	1	0	1	1	1	0	1	0
---	---	---	---	---	---	---	---	---

ALGORITMI GENETICI

POPOLAZIONE

- Gli Algoritmi Genetici mantengono una popolazione di **individui**, che rappresentano un insieme di “soluzioni candidate” al problema trattato. In essi ogni individuo è rappresentato da un solo **cromosoma** (in natura non è proprio così).



Esempio di individui rappresentati da cromosomi binari

ALGORITMI GENETICI

FUNZIONE DI FITNESS

- Ad ogni iterazione dell'algoritmo, gli individui sono valutati mediante una funzione di adattamento (**fitness function**, o **target function**).
- Questa è proprio la funzione che l'algoritmo deve cercare di ottimizzare o il problema che vogliamo risolvere.
- Gli individui più adatti per la soluzione del problema, ossia quelli con le migliori valutazioni, hanno maggiori probabilità di essere scelti per produrre la prossima generazione.

ALGORITMI GENETICI

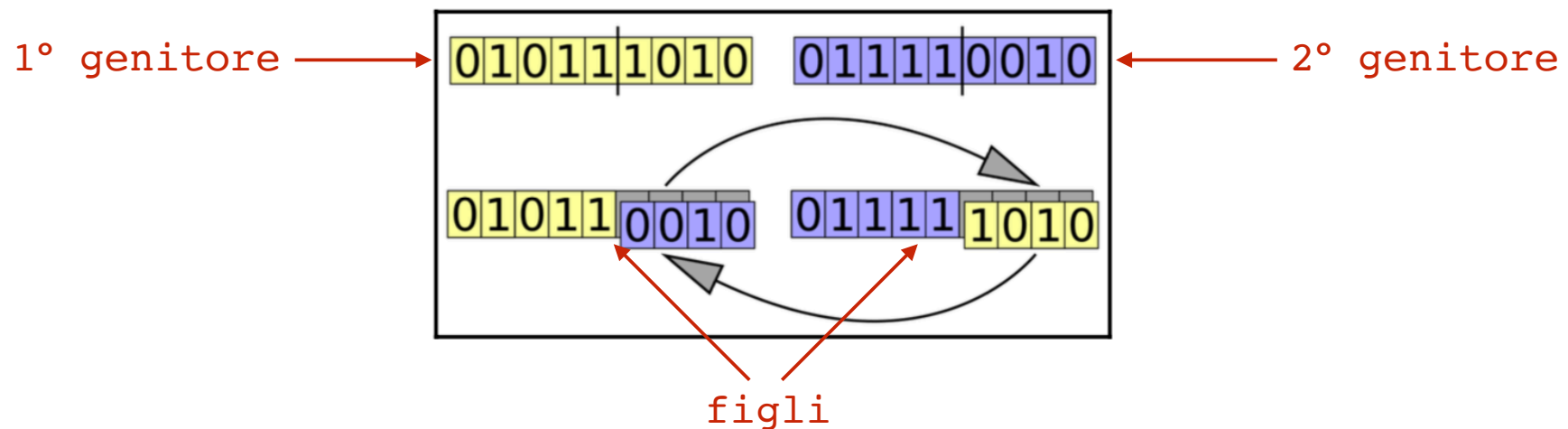
SELEZIONE

- Dopo aver valutato, mediante la fitness function, ogni individuo della popolazione, viene attuato un processo di **selezione** per determinare quali individui dovranno essere utilizzati per la creazione dei figli che costituiranno la prossima generazione.
- Gli individui con le valutazioni migliori avranno una più alta probabilità di essere selezionati.
- Gli individui con valutazioni più basse potranno comunque essere selezionati, anche se con probabilità minore.

ALGORITMI GENETICI

CROSSOVER

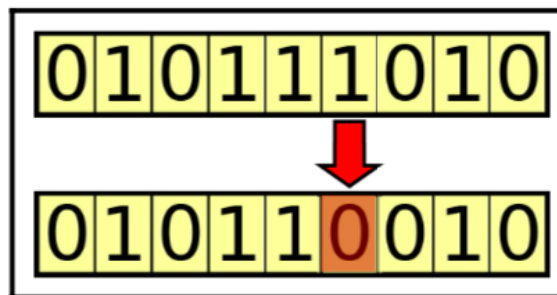
- Per creare due nuovi individui in genere sono scelti due “genitori” dalla popolazione corrente, e parti dei loro cromosomi sono interscambiati (**crossed over**) creando due figli:



ALGORITMI GENETICI

MUTAZIONE

- L'operatore di mutazione consente un refresh casuale della popolazione, introduce nuovi pattern e spinge la ricerca in aree inesplorate. Ad esempio:



ALGORITMI GENETICI

PSEUDOCODICE

- In sintesi, le fasi di elaborazione di un **Algoritmo Genetico** possono essere descritte come segue:

```
function algoritmo_genetico(popolazione, fitness) returns una soluzione
calcolo fitness per ogni individuo della popolazione
While NOT stopping_condition:
    selezione migliori individui della popolazione corrente
    crossover tra coppie dei suddetti individui (generazione offsprings)
    mutazione applicata ai figli generati
    calcolo fitness per ogni individuo della nuova generazione
soluzione ← individuo con la migliore valutazione
end
```

METODI DI SELEZIONE

Possibili metodi di **Selezione** sono i seguenti:

- Roulette Wheel Selection
- Stochastic Universal Sampling
- Rank-Based Selection
- Fitness Scaling
- Tournament Selection

METODI DI SELEZIONE

ROULETTE WHEEL SELECTION

- In questo metodo (chiamato anche “Fitness Proportionate Selection” - FPS), la probabilità di selezionare un individuo è direttamente proporzionale al suo valore di adattamento (fitness value).
- E' come se usassimo una ruota della roulette in cui venga assegnata a ciascun individuo una porzione della ruota proporzionale al suo fitness value.

METODI DI SELEZIONE

ROULETTE WHEEL SELECTION

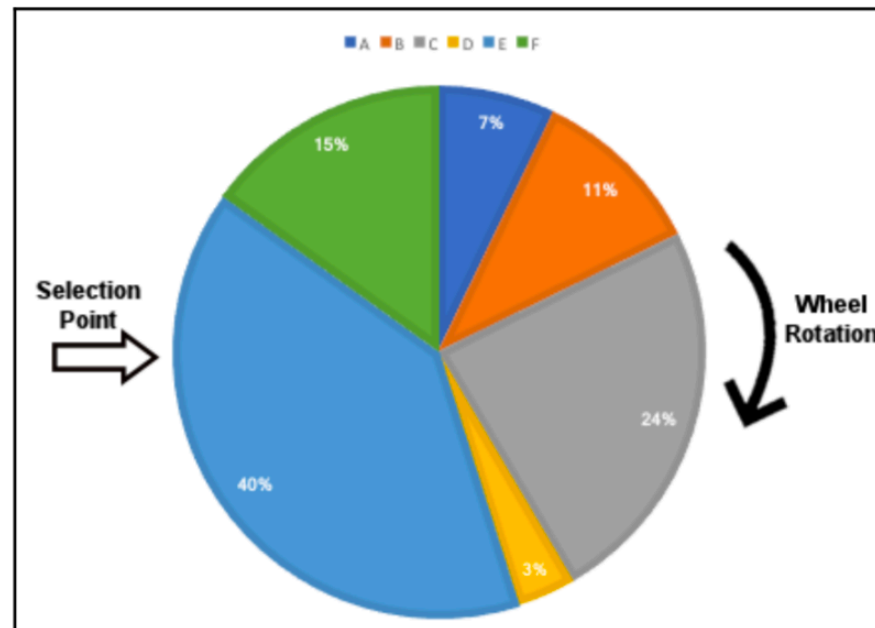
- Vediamo un esempio relativo ad una popolazione di sei individui i cui fitness values e le cui porzioni della ruota sono quelli descritti in figura:

Individual	Fitness	Relative portion
A	8	7%
B	12	11%
C	27	24%
D	4	3%
E	45	40%
F	17	15%

METODI DI SELEZIONE

ROULETTE WHEEL SELECTION

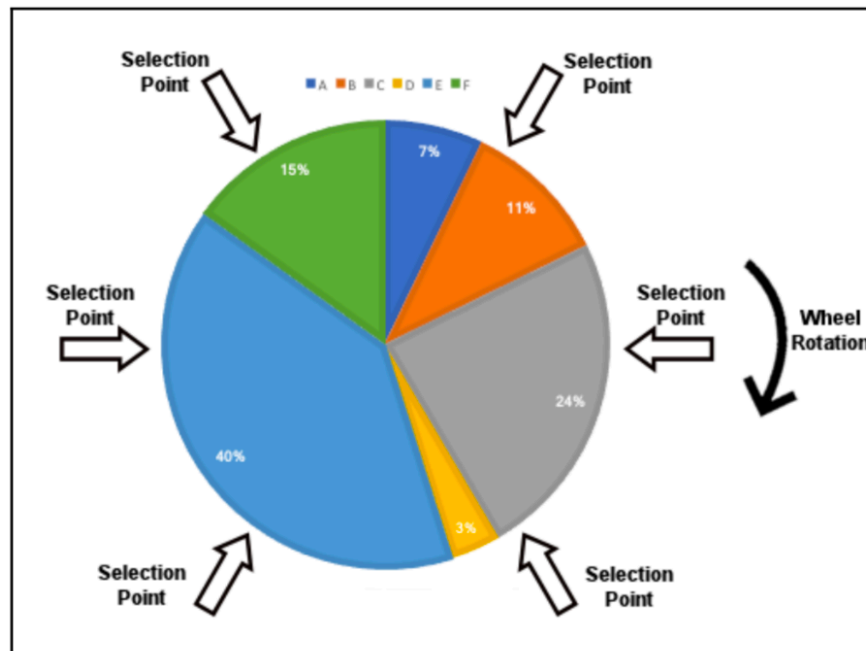
- La corrispondente ruota può essere rappresentata come segue:



METODI DI SELEZIONE

STOCHASTIC UNIVERSAL SAMPLING

- E' una versione leggermente modificata della precedente: stessa ruota, stesse proporzioni, più selection points equidistanti per scegliere simultaneamente tutti gli individui:



METODI DI SELEZIONE

STOCHASTIC UNIVERSAL SAMPLING

- Con questo metodo si cerca di evitare che individui con fitness values molto alti vengano sempre selezionati, saturando in tal modo l'insieme scelto per ottenere la nuova generazione.
- In buona sostanza si danno maggiori possibilità agli individui più “deboli” di essere scelti per generare i figli.

METODI DI SELEZIONE

RANK-BASED SELECTION

- E' simile al metodo "roulette wheel selection", ma invece di usare direttamente i valori della funzione di fitness per il calcolo delle probabilità di selezione, tali valori vengono usati solo per effettuare un ordinamento degli individui.
- Una volta ordinati, ad ogni individuo viene assegnato un rank relativo alla posizione occupata.
- Le probabilità di essere selezionati vengono calcolate basandosi su tali rank.

METODI DI SELEZIONE

RANK-BASED SELECTION

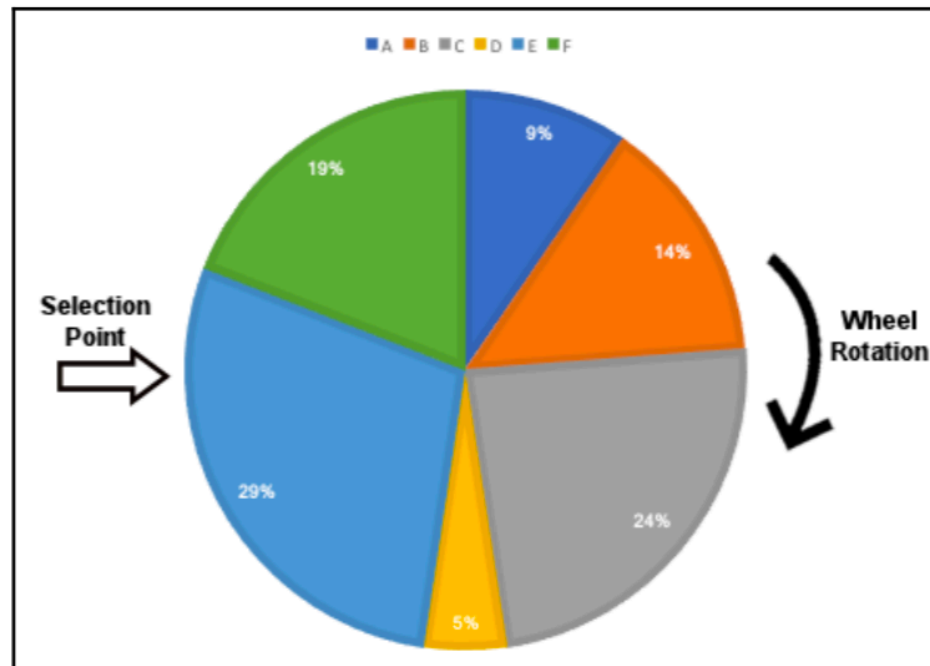
- Ad esempio, vediamo come si applica il metodo ai sei individui visti in precedenza:

Individual	Fitness	Rank	Relative portion
A	8	2	9%
B	12	3	14%
C	27	5	24%
D	4	1	5%
E	45	6	29%
F	17	4	19%

METODI DI SELEZIONE

RANK-BASED SELECTION

- La ruota corrispondente è rappresentata in figura:



METODI DI SELEZIONE

RANK-BASED SELECTION

- Questo metodo può essere utilizzato convenientemente in situazioni in cui ci siano pochi individui con valori di fitness molto più alti rispetto agli altri.
- Usando il rank anziché il fitness value si riesce ad evitare che i pochi individui con alti fitness monopolizzino l'intera popolazione per la successiva generazione, poiché il ranking elimina le ampie differenze tra le valutazioni.
- Un altro caso in cui il metodo funziona bene è quello in cui i vari individui hanno fitness values molto simili, poiché esso riesce a tenerli comunque separati privilegiando quelli con valori più alti.

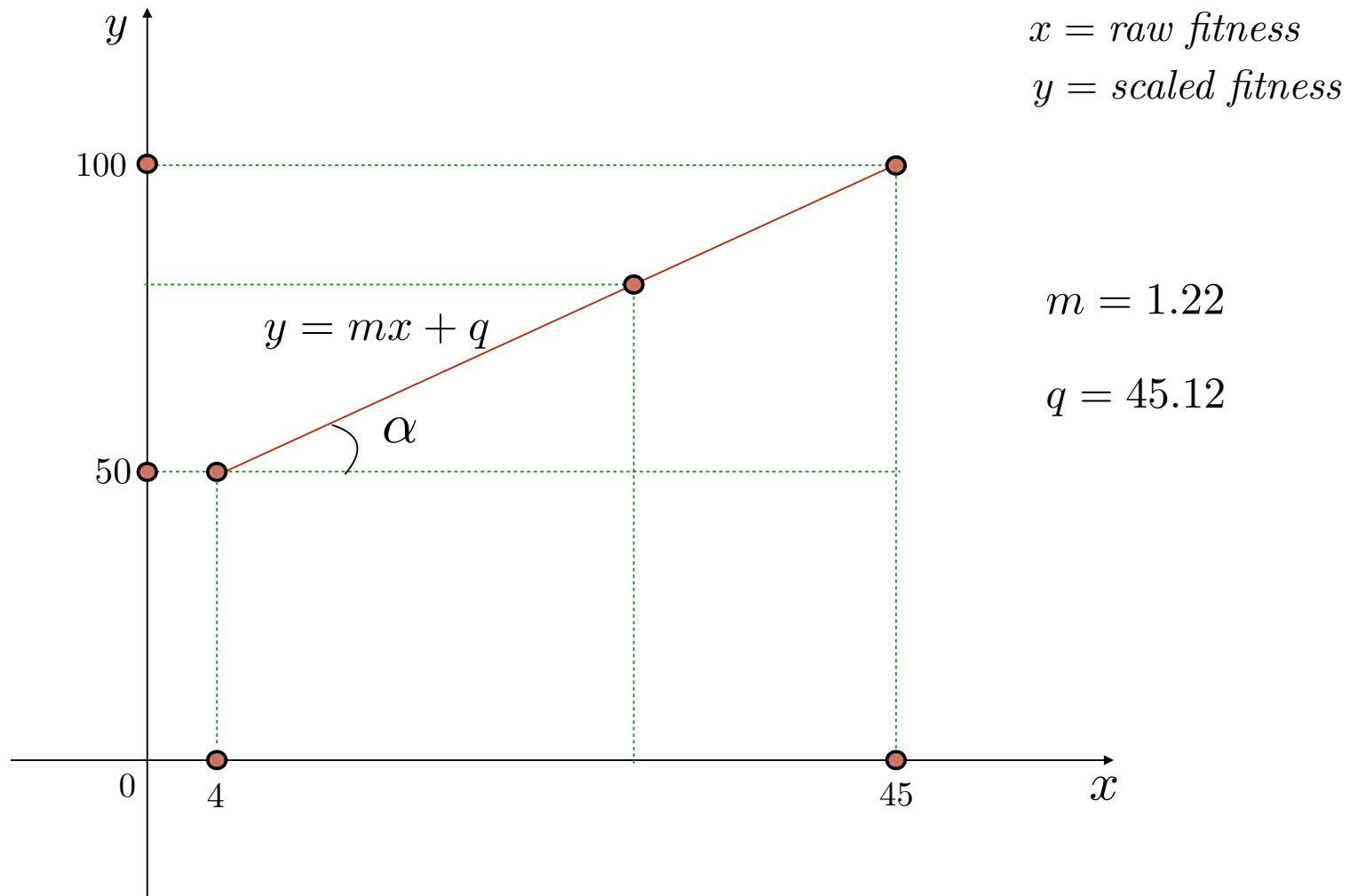
METODI DI SELEZIONE

FITNESS SCALING

- Mentre la selezione basata sul rango sostituisce ogni valore di fitness con il rank dell'individuo, il **fitness scaling** applica una trasformazione di scala ai valori di fitness grezzi e li sostituisce con il risultato della trasformazione.
- Ad esempio, se utilizziamo gli stessi valori degli esempi precedenti, l'intervallo dei valori di fitness grezzi è compreso tra 4 (valore di fitness più basso, individuo D) e 45 (valore di fitness più alto, individuo E). Supponiamo di voler mappare i valori in un nuovo intervallo, compreso tra 50 e 100.

METODI DI SELEZIONE

FITNESS SCALING



METODI DI SELEZIONE

FITNESS SCALING

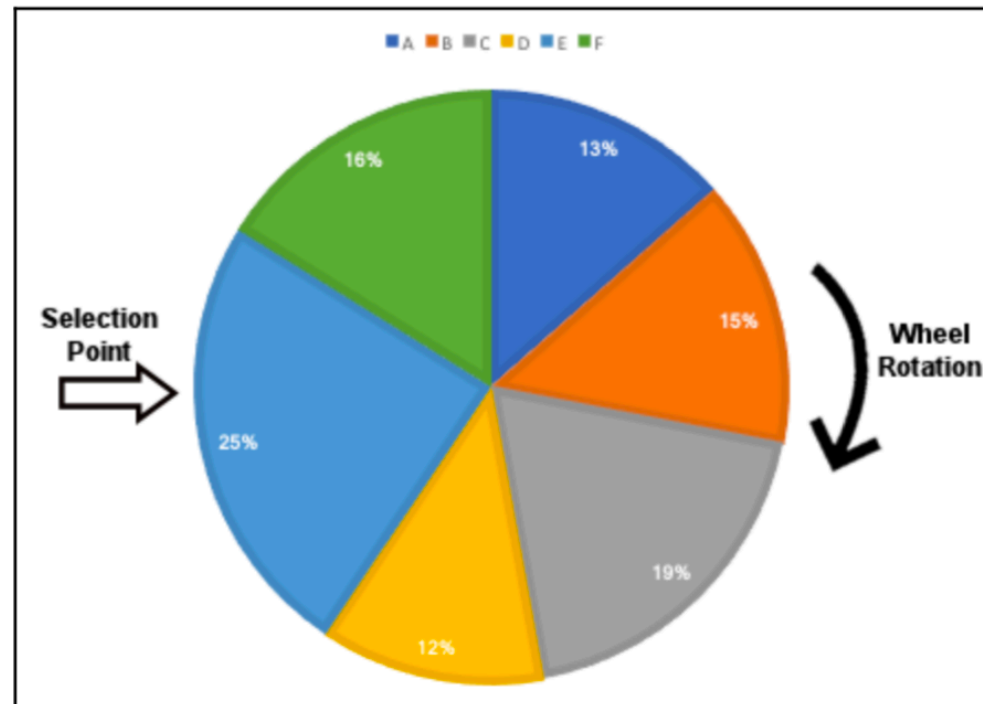
- Nella tabella che segue indichiamo i valori di scaled fitness e le relative porzioni:

Individual	Fitness	Scaled fitness	Relative portion
A	8	55	13%
B	12	60	15%
C	27	78	19%
D	4	50	12%
E	45	100	25%
F	17	66	16%

METODI DI SELEZIONE

FITNESS SCALING

- La ruota corrispondente è la seguente:



METODI DI SELEZIONE

FITNESS SCALING

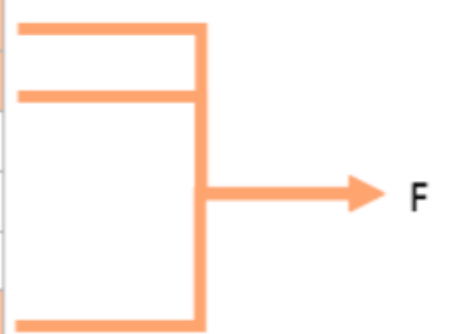
- Come illustrato in figura, il ridimensionamento dei valori di fitness al nuovo intervallo ha fornito una partizione molto più moderata della ruota della roulette rispetto alla partizione originale.
- Infatti l'individuo migliore (con un valore di fitness in scala di 100) ha ora solo il doppio delle probabilità di essere selezionato rispetto al peggiore (con un valore di fitness in scala di 50), invece di essere più di 11 volte più probabile di essere scelto quando si utilizza il valori di fitness grezzi.

METODI DI SELEZIONE

TOURNAMENT SELECTION

- Illustriamolo con un esempio. Supponiamo di avere sei individui aventi gli stessi fitness values degli esempi precedenti. In figura ne sono stati selezionati tre random (A, B e F) tra i quali vince F perché ha il valore più alto:

Individual	Fitness
A	8
B	12
C	27
D	4
E	45
F	17



The diagram illustrates the tournament selection process. It shows a table with six individuals (A, B, C, D, E, F) and their fitness values (8, 12, 27, 4, 45, 17). Three individuals (A, B, and F) are randomly selected, indicated by orange lines. Their fitness values are compared, and the individual with the highest fitness (F, with 17) is selected as the winner, indicated by an orange arrow pointing to F.

METODI DI SELEZIONE

TOURNAMENT SELECTION

- Il numero degli individui che partecipano a ciascuna fase di selezione (3 nell'esempio precedente) è chiamato **tournament size**.
- Maggiore è il tournament size maggiori sono le possibilità che i migliori individui partecipino alla selezione, e di conseguenza minori solo le possibilità di scelta per gli individui con più basso punteggio.

METODI DI CROSSOVER

- L'operatore di **crossover**, detto anche **recombination**, corrisponde al crossover che si verifica nella riproduzione sessuale in biologia.
- E' usato per combinare l'informazione genetica di due individui, i genitori, per produrre (usualmente due) figli.
- Tale operatore è applicato con un alto valore di probabilità.
- Se non è applicato, entrambi i genitori sono "clonati" nella nuova generazione.

METODI DI CROSSOVER

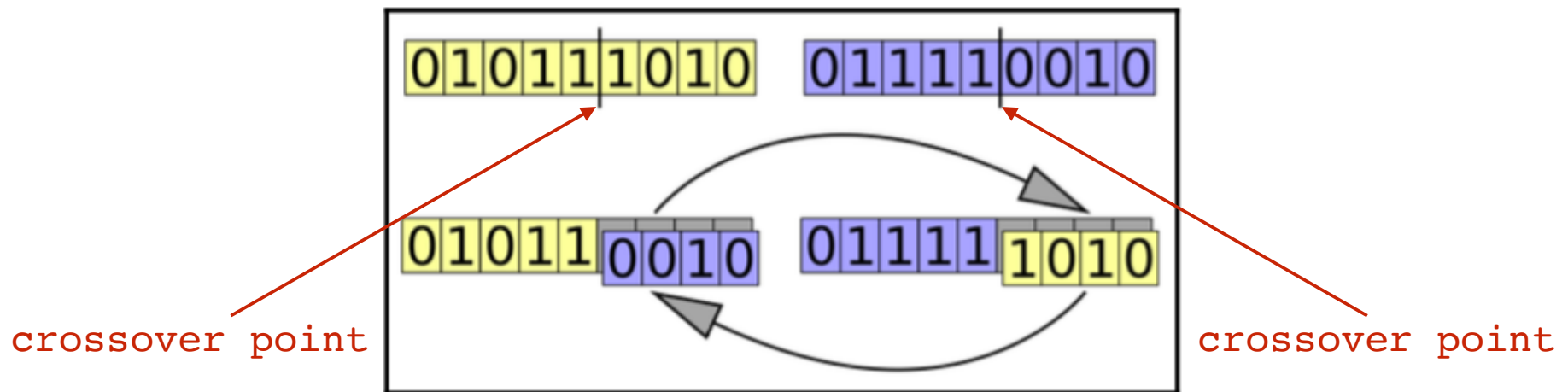
Possibili metodi di **Crossover** sono i seguenti:

- Single-Point Crossover
- Two-Point e k-Point Crossover
- Uniform Crossover
- Ordered Crossover

METODI DI CROSSOVER

SINGLE-POINT CROSSOVER

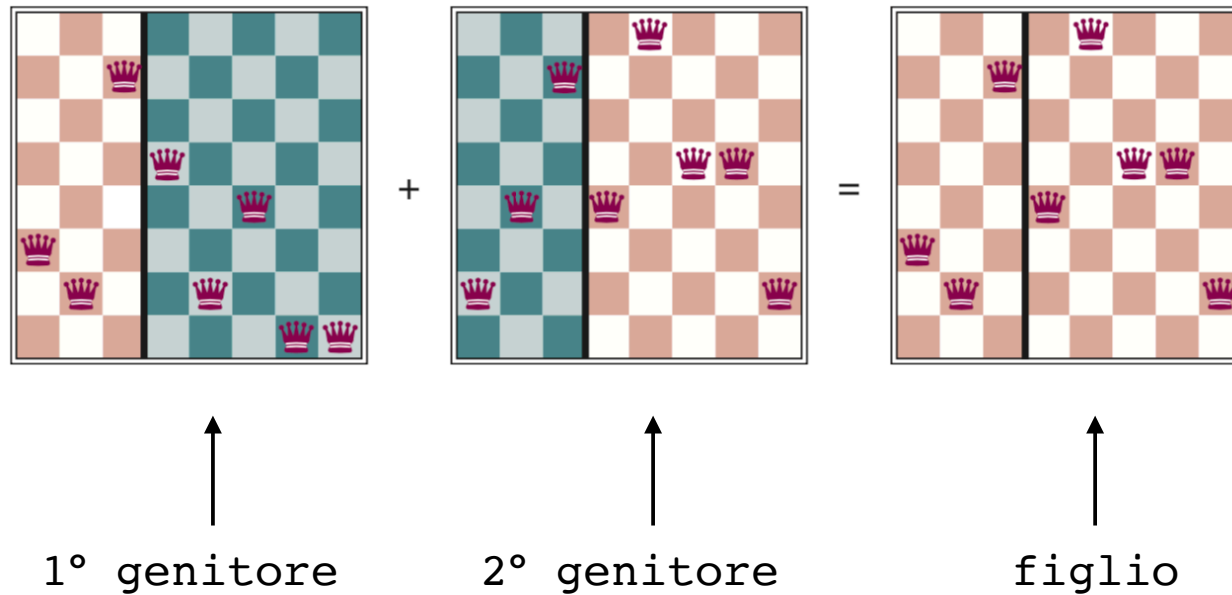
- Si sceglie random una posizione (**crossover point**) nei cromosomi dei genitori e si effettua l'incrocio come segue:



METODI DI CROSSOVER

SINGLE-POINT CROSSOVER

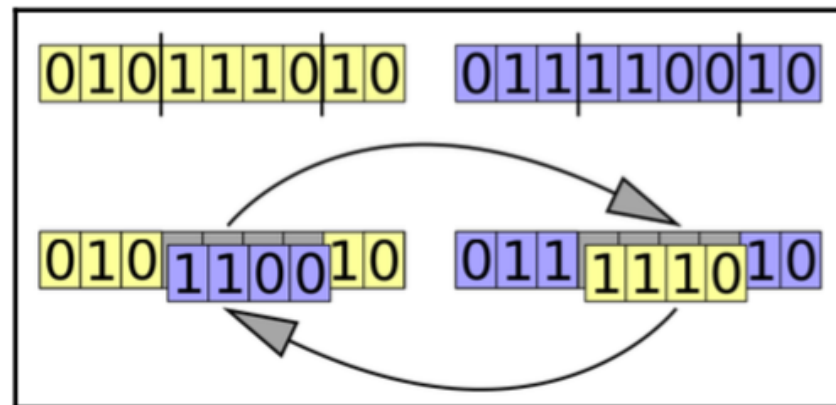
- Esempio per il problema delle 8 regine:



METODI DI CROSSOVER

TWO-POINT CROSSOVER

- In questo caso si scelgono random due crossover points:

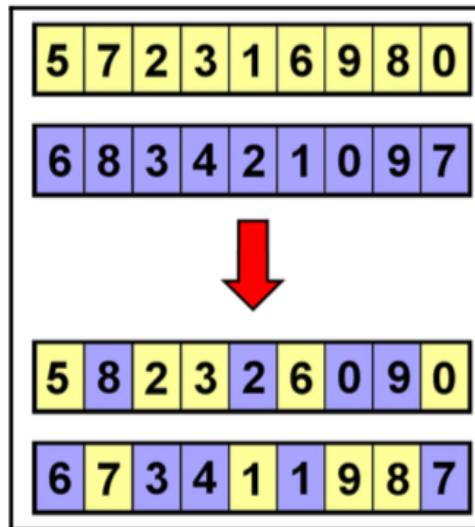


- Il metodo può essere generalizzato scegliendo k crossover points (**k-point crossover**).

METODI DI CROSSOVER

UNIFORM CROSSOVER

- In questo metodo ogni gene è indipendentemente determinato scegliendone uno random dai genitori:



- Poiché il metodo non cambia interi segmenti del cromosoma, esso ha un maggiore potenziale di diversità nella prole risultante.

METODI DI CROSSOVER

CROSSOVER PER LISTE ORDINATE

- Nell'esempio precedente abbiamo visto i risultati di un'operazione di crossover su due cromosomi a base intera.
- Mentre ciascuno dei genitori aveva ogni valore, compreso tra 0 e 9, apparire esattamente una volta, ciascuno dei figli risultanti aveva determinati valori che apparivano più di una volta (ad esempio, 2 nella progenie superiore e 1 nell'altra) e altri valori mancavano (come 4 nella progenie superiore e 5 nell'altra).
- In alcune attività, tuttavia, i cromosomi a base intera possono rappresentare indici di un elenco ordinato.

METODI DI CROSSOVER

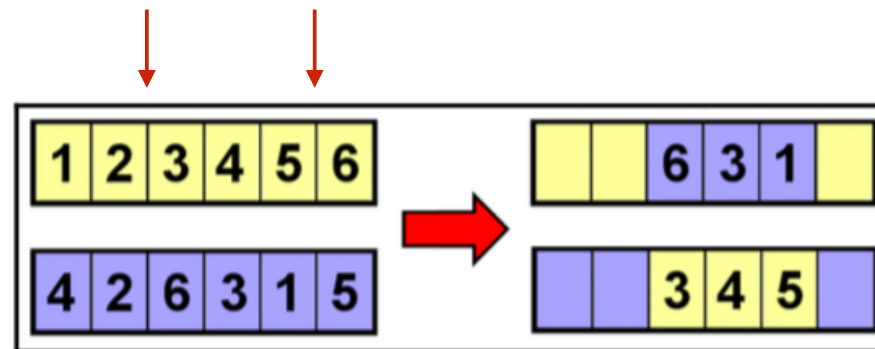
CROSSOVER PER LISTE ORDINATE

- Prendiamo ad esempio il problema del commesso viaggiatore: se abbiamo quattro città, un modo conveniente per rappresentare una possibile soluzione a questo problema sarebbe un cromosoma a quattro interi che mostri l'ordine di visita delle città, ad esempio (1,2,3,4) o (3,4,2,1).
- Un cromosoma che ha due geni con valori uguali, o manca uno dei valori come (1,2,2,4), non rappresenterà una soluzione valida per il TSP.
- In questi casi posso essere usati metodi di crossover alternativi che assicurano la validità dei vari offsprings.

METODI DI CROSSOVER

ORDERED CROSSOVER

- Il metodo **ordered crossover** (OX1) cerca di preservare quanto più possibile l'ordine dei geni.
- Il primo passo consiste in un two-point crossover con i due cut point scelti random:

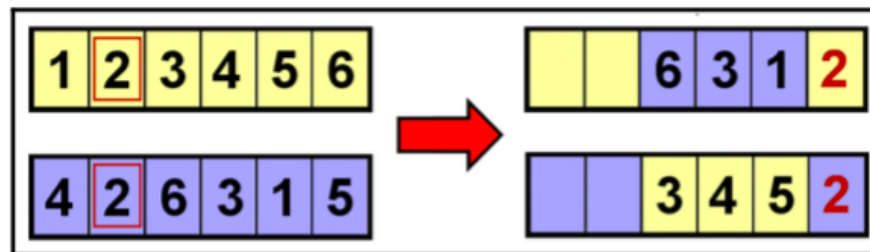


- Si prosegue riempiendo il resto dei geni di ogni figlio esaminando tutti i geni dei genitori nel loro ordine originale, iniziando dopo il secondo punto di taglio.

METODI DI CROSSOVER

ORDERED CROSSOVER

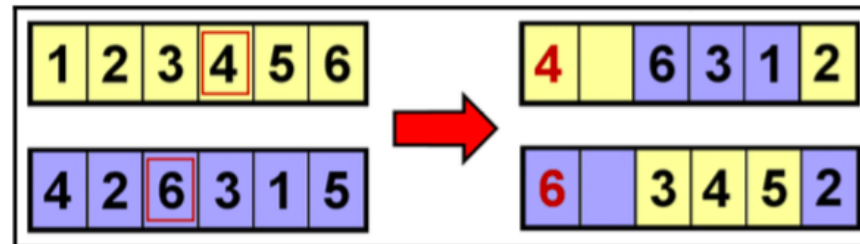
- Per il primo genitore troviamo un 6, ma questo è già presente nella prole, quindi continuiamo (in modo circolare) con 1; anche questo è già presente. Il prossimo in ordine è il 2. Poiché 2 non è ancora presente nella prole, lo aggiungiamo lì, come mostrato nella figura seguente.
- Per la seconda coppia genitore-figlio, iniziamo con il 5 del genitore, che è già presente nella prole, poi si passa al 4, anch'esso presente, e finiamo con il 2, che non è ancora presente e che quindi viene aggiunto.



METODI DI CROSSOVER

ORDERED CROSSOVER

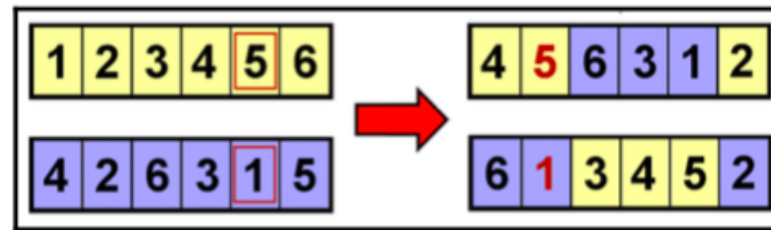
- Per il primo genitore si continua con 3 (già presente nel figlio) e poi 4, che viene aggiunto (in modo circolare) alla prole. Per l'altro genitore, il gene successivo è 6. Poiché non è presente nella prole corrispondente, viene aggiunto ad esso:



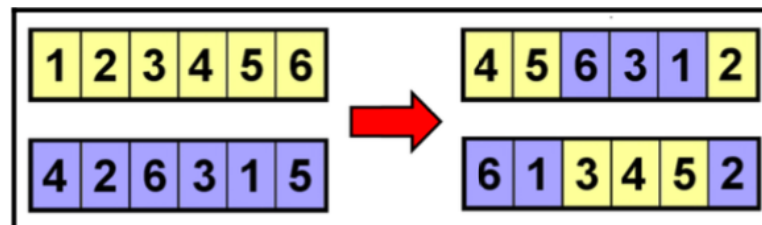
METODI DI CROSSOVER

ORDERED CROSSOVER

- Continuiamo in tal modo con i geni successivi non ancora presenti nella prole e riempiamo gli ultimi posti disponibili:



- Questo completa il processo di produzione di due cromosomi discendenti validi, come mostrato in figura:



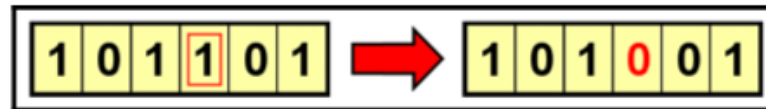
METODI DI MUTAZIONE

Possibili metodi di **Mutazione** sono i seguenti:

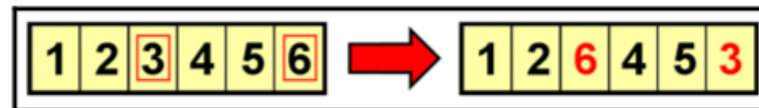
- Flip Bit Mutation
- Swap Mutation
- Inversion Mutation
- Scramble Mutation

METODI DI MUTAZIONE

- Flip bit mutation:

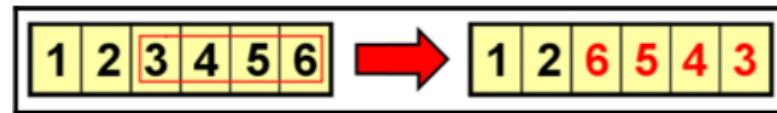


- Swap mutation:

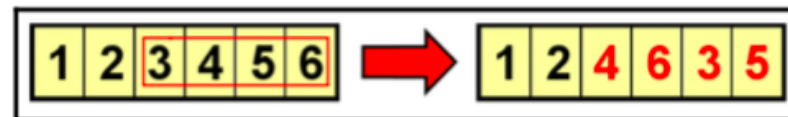


METODI DI MUTAZIONE

- Inversion mutation:



- Scramble mutation:



CONDIZIONI DI TERMINAZIONE

Le due più comuni **stopping conditions** sono le seguenti:

- Numero massimo di generazioni raggiunto.
- Nessun significativo miglioramento nelle ultime generazioni.

Altre **stopping conditions** da poter utilizzare sono:

- Soglia sulla quantità di tempo di elaborazione.
- Soglia su tempo di CPU e/o memoria.

ALGORITMI GENETICI

DIFFERENZE DAGLI ALGORITMI TRADIZIONALI

- Le caratteristiche chiave degli algoritmi genetici che li distinguono dagli algoritmi tradizionali sono:
 - Mantenere una popolazione di soluzioni.
 - Uso di una rappresentazione genetica delle soluzioni.
 - Utilizzo del risultato di una funzione di fitness.
 - Mostrare un comportamento probabilistico.

ALGORITMI GENETICI

PROS & CONS

Vantaggi:

- Capacità di ottimizzazione globale.
- Gestire problemi con una rappresentazione matematica complessa.
- Gestire problemi privi di rappresentazione matematica.
- Robustezza al rumore.
- Supporto per il parallelismo e l'elaborazione distribuita.

ALGORITMI GENETICI

PROS & CONS

Limiti:

- Necessità di ottimizzazione degli iperparametri.
- Operazioni ad alta intensità di calcolo.
- Rischio di una convergenza prematura.
- Nessuna soluzione garantita.

ALGORITMI GENETICI

AREE DI APPLICAZIONE

- Ottimizzazione combinatoria.
- Tasks di placement e scheduling.
- Attività di programmazione automatica.
- Compiti AI (struttura e parametri per NN).
- Database query optimization.
- ecc. ecc.

ALGORITMI GENETICI

AREE DI APPLICAZIONE

- In scenari in tempo reale, gli Algoritmi Genetici vengono utilizzati per sviluppare sistemi di Intelligenza Artificiale, come ad esempio:
 - Progettare task per rotte di aeromobili negli aeroporti.
 - Determinare il comportamento ottimale dei robot.
 - Problemi di costruzione di portafogli di investimento.
 - ecc. ecc.

SINTESI DEGLI ARGOMENTI TRATTATI NELLA LEZIONE

- Gli **Algoritmi Genetici (AG)** sono algoritmi stocastici i cui metodi di ricerca si ispirano a fenomeni naturali quali l'ereditarietà genetica e la lotta darwiniana per la sopravvivenza.
- Negli **AG** usiamo una nomenclatura presa in prestito dalla genetica: **popolazione**, **individui** (o **genotipi**). Spesso tali individui sono chiamati anche stringhe o **cromosomi**.
- Gli **AG** partono da una popolazione iniziale di individui, valutati mediante una **funzione di fitness**, su cui vengono poi applicati gli operatori di **selezione**, **crossover** e **mutazione** ai fini della produzione della generazione successiva. Procedono iterativamente allo stesso modo sulle nuove popolazioni via via generate fino al verificarsi di una **stopping condition**.
- Hanno vari vantaggi, come ad esempio quello di poter affrontare problemi che non hanno una precisa formalizzazione matematica.
- Hanno anche vari limiti, come ad esempio quello di dover ottimizzare vari **iperparametri**, la pesantezza computazionale e il fatto che non è garantita la convergenza su una soluzione ottima.
- Gli **AG** sono stati applicati con successo in vari problemi di ottimizzazione come ad esempio: wire routing, scheduling, adaptive control, AI tasks (e.g., scelta struttura e parametri nelle reti neurali artificiali), game playing, cognitive modeling, transportation problems, traveling salesman problems, optimal control problems, database query optimization, ecc...

RIFERIMENTI

Wirsansky, E. *Hands-On Genetic Algorithms with Python*, Packt, 2020.

Brownlee, J. *Optimization for Machine Learning - Finding Function Optima with Python*, Machine Learning Mastery, 2021.

Gridin, I. *Learning Genetic Algorithms with Python*, BPB Publications, 2021.

Luke, S. *Essentials of Metaheuristics*, Second Edition, 2013.

Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*, Third Edition, Springer, 1999.