Università degli Studi di Bologna

Corso di Calcolo Numerico

Homework 2 – Gradient Descent e Stochastic Gradient Descent

Studente: **Luca Argentino**
Matricola: **0001161779**

---

1 Metodo del Gradiente (GD) passo fisso e backtracking

Implementazione Python

```python

def backtracking(f, gradf, x):

    alpha = 1.0
    rho = 0.5
    c = 1e-4

    while f(x - alpha * gradf(x)) > f(x) - c * alpha * np.linalg.norm(gradf(x))**2:
        alpha *= rho
        if alpha < 1e-10:
            print("Backtracking interrotto: passo troppo piccolo")
            break
    return alpha




def GD(f, gradf, x0, x_true, use_backtracking, maxit=15000, tolf=1e-6,
        tolx=1e-5,alpha_fixed=0.1):

    k = 0
    x = x0


    errore = np.zeros((maxit,))
    fun = np.zeros((maxit,))
    grad = np.zeros((maxit,))


    errore[k] = np.linalg.norm(x_true - x) / np.linalg.norm(x_true)
    fun[k] = f(x)
    grad[k] = np.linalg.norm(gradf(x))


    condizione = True
    reason = ""

    while condizione:

        if k >= maxit - 1:
            reason = "Iterazioni massime raggiuntes"
            condizione = False
        else:

            if use_backtracking :

                alpha = backtracking(f, gradf, x)
```

```
            else :
                 alpha = alpha_fixed



            x_new = x - alpha * gradf(x)

            errore[k] = np.linalg.norm(x_true - x_new) / np.linalg.norm(x_true)
            fun[k] = f(x_new)
            grad[k] = np.linalg.norm(gradf(x_new))

            if np.linalg.norm(gradf(x_new)) <= tolf:
                reason = "Gradiente piccolo (convergenza)"
                condizione = False

            elif np.linalg.norm(x_new - x) <= tolx:
                reason = "Iterati molto vicini (punto stazionario)"
                condizione = False

            x = x_new
            k = k + 1

    errore = errore[:k+1]
    fun = fun[:k+1]
    grad = grad[:k+1]

    return x, k, errore, fun, grad, reason
```

2 Test su funzioni di prova

Funzioni testate:

(a) Quadratica semplice,
(b) Rosenbrock,
(d) Minimi quadrati,
(f) Funzione logaritmica

Per ogni test si includono grafici di convergenza, superficie/curve di livello e tabelle dei risultati.

2.1 (a) Quadratica semplice

$$f(x_1, x_2) = (x_1 - 5)^2 + (x_2 - 2)^2, \quad \nabla f(x) = \begin{pmatrix} 2(x_1 - 5) \\ 2(x_2 - 2) \end{pmatrix}, \quad x^* = (5, 2)$$

Analisi della funzione

La funzione considerata è una funzione quadratica strettamente convessa, Il valore della funzione nel minimo globale è

$$f(x^*) = (5 - 5)^2 + (2 - 2)^2 = 0.$$

Superficie e linee di livello per $x_0 = [5, 5]$

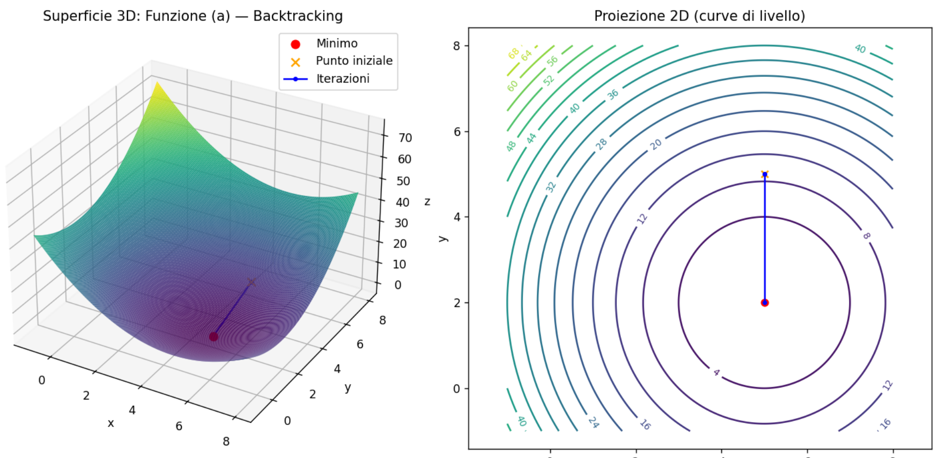

Figure 1: Superficie 3D e linee di livello (backtracking).

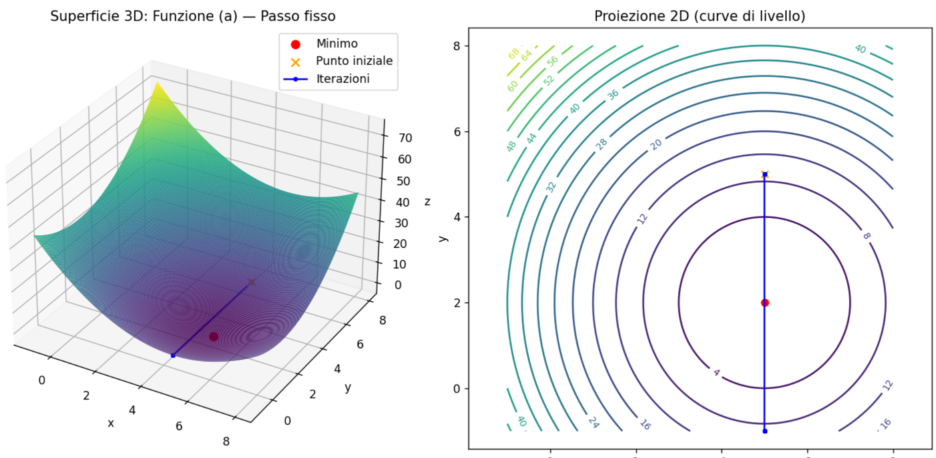

Figure 2: Superficie 3D e linee di livello ($\alpha = 1$).

Grafici funzione e norma del gradiente per $x_0 = [5, 5]$

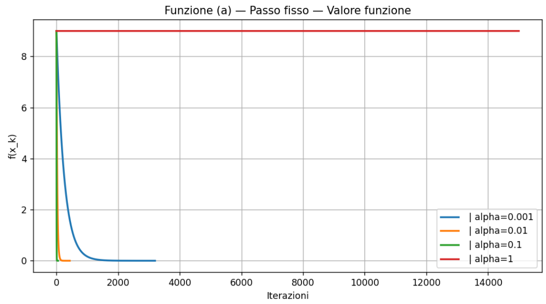

Figure 3: Valori funzione $f(x_k)$

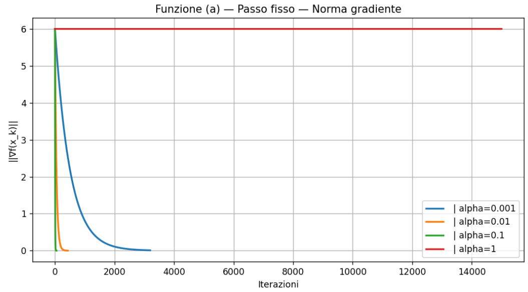

Figure 4: Norma gradiente $\|\nabla f(x_k)\|$

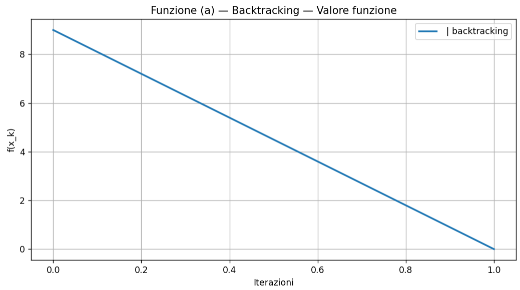

Figure 5: Valore funzione $f(x_k)$ (backtracking).

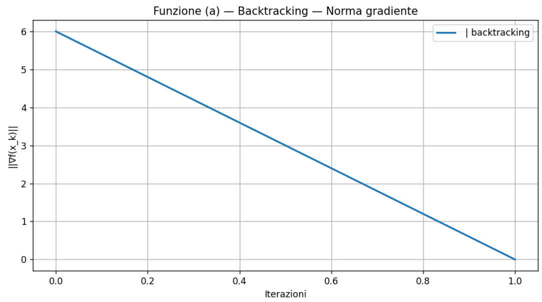

Figure 6: Norma gradiente $\|\nabla f(x_k)\|$ (backtracking).

Tabella dei risultati

Punto iniziale	Tipo passo	Iterazioni	Soluzione x^*	$f(x^*)$	Esito
$[0, 0]$	Fisso $\alpha = 0.001$	3489	$[4.995372,\ 1.998149]$	$2.484935 \cdot 10^{-5}$	Iterati molto vicini
$[0, 0]$	Fisso $\alpha = 0.01$	461	$[4.999549,\ 1.999820]$	$2.359608 \cdot 10^{-7}$	Iterati molto vicini
$[0, 0]$	Fisso $\alpha = 0.1$	53	$[4.999963,\ 1.999985]$	$1.548591 \cdot 10^{-9}$	Iterati molto vicini
$[0, 0]$	Fisso $\alpha = 1$	14999	$[10.000000,\ 4.000000]$	29	Max Iter (diverge)
$[0, 0]$	Backtracking	1	$[5.000000,\ 2.000000]$	0	$\|\nabla f(x_k)\| \leq \text{tol}_f$
$[5, 5]$	Fisso $\alpha = 0.001$	3197	$[5.000000,\ 2.004983]$	$2.482696 \cdot 10^{-5}$	Iterati molto vicini
$[5, 5]$	Fisso $\alpha = 0.01$	432	$[5.000000,\ 2.000486]$	$2.363589 \cdot 10^{-7}$	Iterati molto vicini
$[5, 5]$	Fisso $\alpha = 0.1$	51	$[5.000000,\ 2.000034]$	$1.173333 \cdot 10^{-9}$	Iterati molto vicini
$[5, 5]$	Fisso $\alpha = 1$	14999	$[5.000000,\ -1.000000]$	9	Max Iter (diverge)
$[5, 5]$	Backtracking	1	$[5.000000,\ 2.000000]$	0	$\|\nabla f(x_k)\| \leq \text{tol}_f$

2.2 (b) Rosenbrock

$$f(x_1,x_2) = (1-x_1)^2 + 100(x_2 - x_1^2)^2, \quad \nabla f(x) = \begin{pmatrix} -2(1-x_1) - 400x_1(x_2 - x_1^2) \\ 200(x_2 - x_1^2) \end{pmatrix}, \quad x^* = (1,1)$$

Analisi della funzione

La funzione di Rosenbrock è una funzione non convessa con un minimo globale

$$x^* = (1,1), \quad f(x^*) = 0.$$

Superficie e linee di livello per $x_0 = [0,1]$

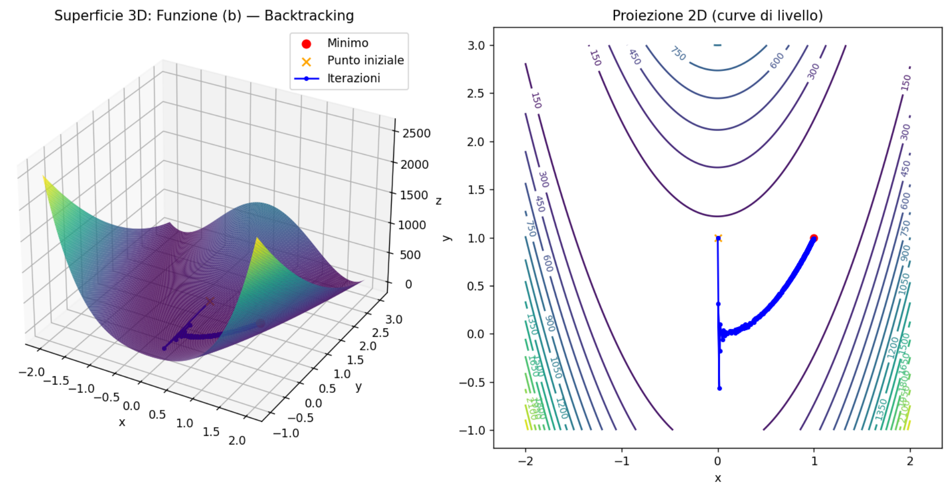

Figure 7: Superficie 3D e linee di livello (backtracking).

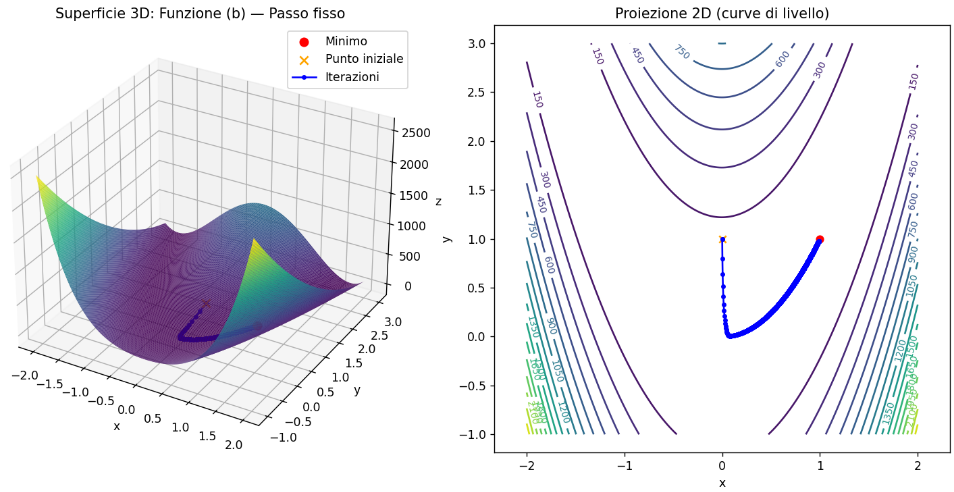

Figure 8: Superficie 3D e linee di livello ($\alpha = 0.001$).

Grafici funzione e norma del gradiente per $x_0 = [0, 1]$

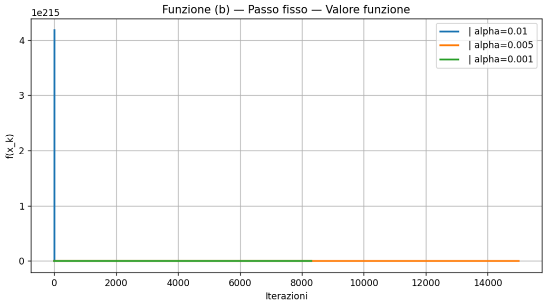

Figure 9: Valori funzione $f(x_k)$

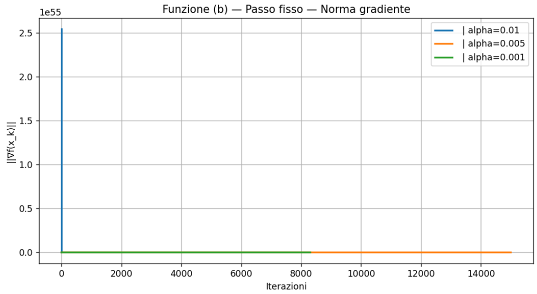

Figure 10: Norma gradiente $\|\nabla f(x_k)\|$

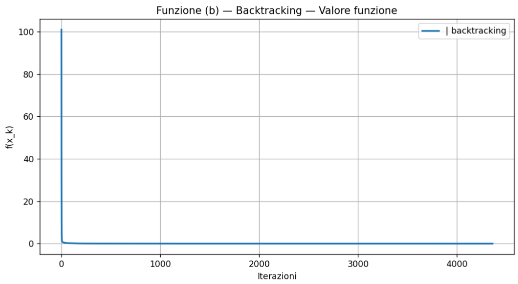

Figure 11: Valori funzione $f(x_k)$ (backtracking).

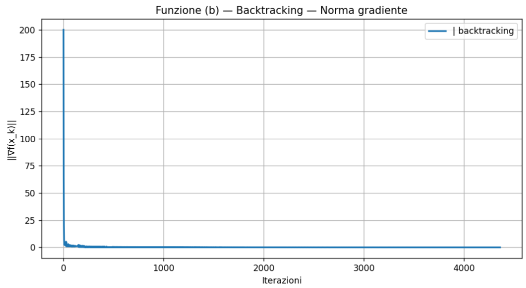

Figure 12: Norma gradiente $\|\nabla f(x_k)\|$ (backtracking).

Tabella dei risultati

Punto iniziale	Tipo passo	Iterazioni	Soluzione x^*	$f(x^*)$	Esito
$[0, 1]$	Fisso $\alpha = 0.001$	8302	$[0.9889,\ 0.9779]$	$1.23 \cdot 10^{-4}$	Iterati molto vicini
$[0, 1]$	Fisso $\alpha = 0.005$	14999	$[0.4633,\ 0.2854]$	0.7880	Max Iter
$[0, 1]$	Fisso $\alpha = 0.01$	14999	$[nan,\ nan]$	nan	Max Iter (diverge)
$[0, 1]$	Backtracking	4359	$[0.9961,\ 0.9922]$	$1.50 \cdot 10^{-5}$	Iterati molto vicini
$[-1, 2]$	Fisso $\alpha = 0.001$	9566	$[0.9889,\ 0.9779]$	$1.23 \cdot 10^{-4}$	Iterati molto vicini
$[-1, 2]$	Fisso $\alpha = 0.005$	14999	$[nan,\ nan]$	nan	Max Iter (diverge)
$[-1, 2]$	Fisso $\alpha = 0.01$	14999	$[nan,\ nan]$	nan	Max Iter (diverge)
$[-1, 2]$	Backtracking	3324	$[0.9961,\ 0.9922]$	$1.53 \cdot 10^{-5}$	Iterati molto vicini

2.3 (d) Minimi quadrati Vandermonde (n=4)

Consideriamo la funzione di minimi quadrati

$$f(x) = \frac{1}{2}\|Ax - b\|^2, \qquad \nabla f(x) = A^\top(Ax - b), \qquad x^* = \mathbf{1},$$

dove $A \in \mathbb{R}^{4\times 4}$ è una matrice di Vandermonde costruita sui nodi equispaziati

$$v = (0, \tfrac{1}{3}, \tfrac{2}{3}, 1), \qquad A = \text{vander}(v) = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1/27 & 1/9 & 1/3 & 1 \\ 8/27 & 4/9 & 2/3 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

Il vettore dei termini noti è calcolato come

$$b = Ax^*,$$

in modo che il minimo globale della funzione sia raggiunto in $x^* = \mathbf{1}$ con $f(x^*) = 0$.

Analisi della funzione

Poiché la funzione è quadratica e la matrice A è invertibile la funzione $f(x)$ è strettamente convessa;

Grafici funzione e norma del gradiente $x_0 = [0, 0, 0, 0]$

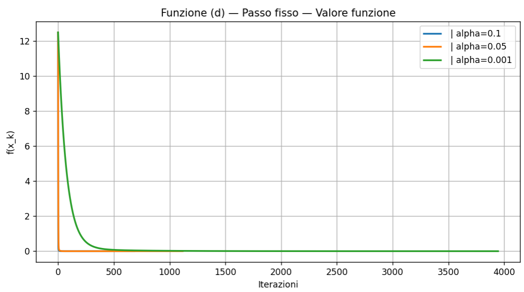

Figure 13: Valori della funzione $f(x_k)$ nel corso delle iterazioni.

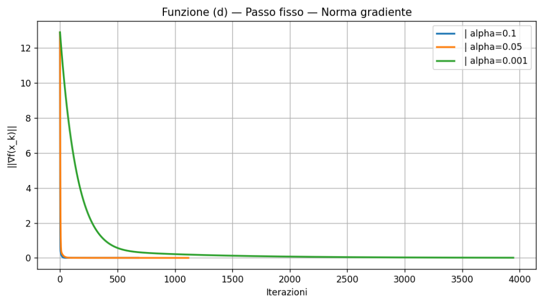

Figure 14: Norma del gradiente $\|\nabla f(x_k)\|$.

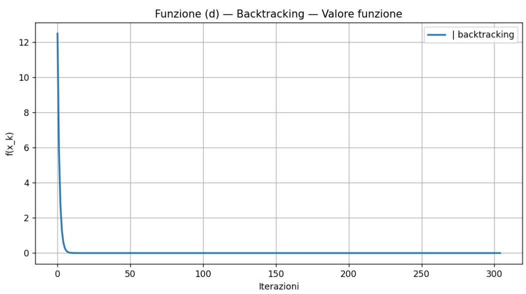

Figure 15: Valori funzione $f(x_k)$ (backtracking).

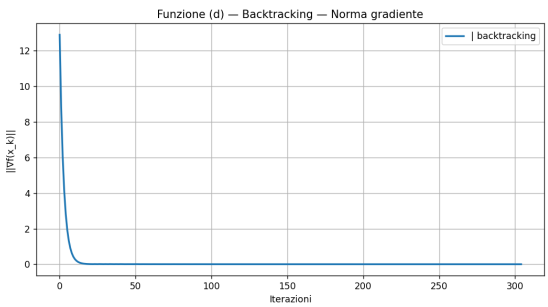

Figure 16: Norma gradiente $\|\nabla f(x_k)\|$ (backtracking).

Tabella dei risultati

Punto iniziale	Tipo passo	Iterazioni	Soluzione x^*	$f(x^*)$	Esito
$\mathbf{0}$	$\alpha = 0.1$	684	$[0.9968, 1.0031, 1.0002, 0.9998]$	$9.53 \cdot 10^{-8}$	Iterati molto vicini
$\mathbf{0}$	$\alpha = 0.05$	1117	$[0.9957, 1.0029, 1.0016, 0.9995]$	$3.66 \cdot 10^{-7}$	Iterati molto vicini
$\mathbf{0}$	$\alpha = 0.001$	3945	$[0.9556, 0.9926, 1.0451, 0.9966]$	$1.51 \cdot 10^{-4}$	Iterati molto vicini
$\mathbf{0}$	Backtracking	304	$[0.9977, 1.0032, 0.9991, 0.99997]$	$9.37 \cdot 10^{-9}$	Iterati molto vicini
$\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}$	$\alpha = 0.1$	7651	$[0.9240, 1.1162, 0.9580, 1.0011]$	$7.25 \cdot 10^{-6}$	Iterati molto vicini
$\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}$	$\alpha = 0.05$	2216	$[0.8793, 1.1822, 0.9359, 1.0014]$	$1.80 \cdot 10^{-5}$	Iterati molto vicini
$\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}$	$\alpha = 0.001$	14999	$[0.6099, 1.1471, 1.2713, 0.9363]$	$5.31 \cdot 10^{-3}$	Max Iter
$\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}$	Backtracking	8496	$[0.9809, 1.0292, 0.9895, 1.0003]$	$4.57 \cdot 10^{-7}$	Iterati molto vicini

2.4 (f) Funzione logaritmica (n=4)

$$f(x) = \sum_{i=1}^{n} (x_i - i)^2 - \sum_{i=1}^{n} \ln(x_i), \qquad \nabla f(x)_i = 2(x_i - i) - \frac{1}{x_i}, \qquad x \in \mathbb{R}^n,\ x_i > 0,$$

mentre la soluzione esatta è nota in forma chiusa:

$$x_i^* = \frac{1}{2}\left(i + \sqrt{i^2+2}\right), \qquad i = 1, \ldots, n.$$

il valore della funzione è sempre $f(x^*) \approx -3.45349$

Grafici funzione e norma del gradiente $x_0 = [0.5, 0.5, 0.5, 0.5]$

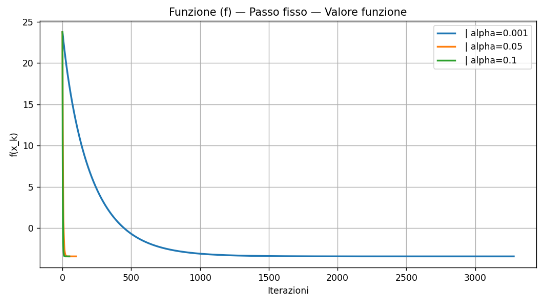

Figure 17: Valori della funzione $f(x_k)$ nel corso delle iterazioni.

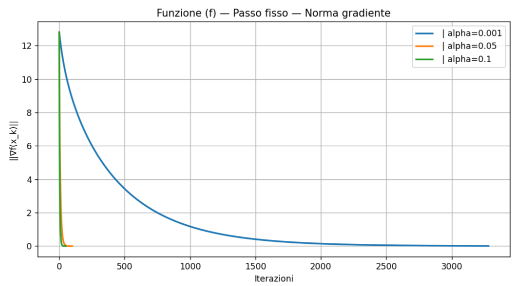

Figure 18: Norma del gradiente $\|\nabla f(x_k)\|$.

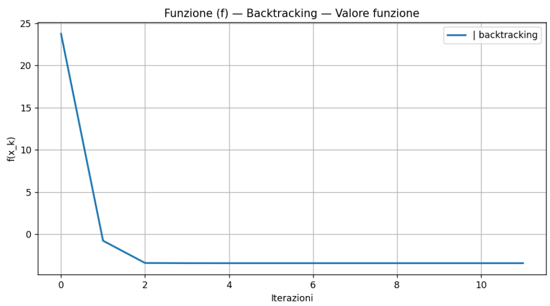

Figure 19: Valori funzione $f(x_k)$ (backtracking).

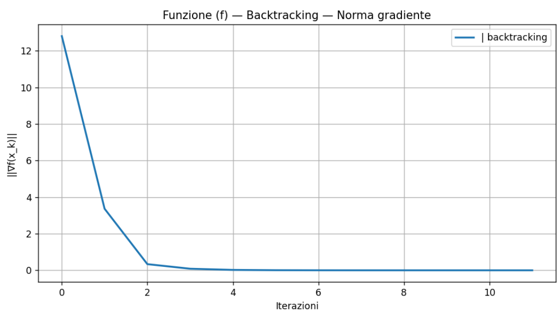

Figure 20: Norma gradiente $\|\nabla f(x_k)\|$ (backtracking).

Tabella dei risultati — Funzione (f)

Punto iniziale	Tipo passo	Iterazioni	Soluzione x^*	$f(x^*)$	Esito
$\mathbf{1, 2, 3, 4}$	$\alpha = 0.001$	2014	[1.3640, 2.2221, 3.1560, 4.1194]	-3.45347	Iterati molto vicini
$\mathbf{1, 2, 3, 4}$	$\alpha = 0.05$	73	[1.3660, 2.2247, 3.1583, 4.1213]	-3.45349	Iterati molto vicini
$\mathbf{1, 2, 3, 4}$	$\alpha = 0.1$	38	[1.3660, 2.2247, 3.1583, 4.1213]	-3.45349	Iterati molto vicini
$\mathbf{1, 2, 3, 4}$	Backtracking	10	[1.3660, 2.2247, 3.1583, 4.1213]	-3.45349	Iterati molto vicini
$\mathbf{0.5, 0.5, 0.5, 0.5}$	$\alpha = 0.001$	3281	[1.3659, 2.2236, 3.1558, 4.1174]	-3.45347	Iterati molto vicini
$\mathbf{0.5, 0.5, 0.5, 0.5}$	$\alpha = 0.05$	99	[1.3660, 2.2247, 3.1583, 4.1212]	-3.45349	Iterati molto vicini
$\mathbf{0.5, 0.5, 0.5, 0.5}$	$\alpha = 0.1$	50	[1.3660, 2.2247, 3.1583, 4.1213]	-3.45349	Iterati molto vicini
$\mathbf{0.5, 0.5, 0.5, 0.5}$	Backtracking	11	[1.3660, 2.2247, 3.1583, 4.1213]	-3.45349	Iterati molto vicini

3 Metodo del Gradiente Stocastico(SGD)

```
def SGD(f, gradf_i, x0, x_true, n, alpha=0.01, batch_size=2, N_epoche=50, tolf
    =1e-6, tolx=1e-5, maxit=10000):

    x = x0
    k = 0

    errore = np.zeros((maxit,))
    fun = np.zeros((maxit,))
    grad = np.zeros((maxit,))
    reason = ""


    for epoca in range(N_epoche):

        idx = np.random.permutation(n)

        for j in range(0, n, batch_size):

            S_j = idx[j : j + batch_size]

            g_batch = np.zeros(n)

            for i in S_j:
                g_batch += gradf_i(x, i)

            x_new = x - alpha * g_batch

            errore[k] = np.linalg.norm(x_true - x_new) / np.linalg.norm(x_true)
            fun[k] = f(x_new)
            grad[k] = np.linalg.norm(g_batch)

            if grad[k] <= tolf:
                reason = "Gradiente piccolo (convergenza)"
                return x_new, k, errore[:k+1], fun[:k+1], grad[:k+1], reason

            if np.linalg.norm(x_new - x) <= tolx:
                reason = "Iterati molto vicini (punto stazionario)"
                return x_new, k, errore[:k+1], fun[:k+1], grad[:k+1], reason

            x = x_new
            k += 1


    reason = "Raggiunto numero massimo di epoche"
    return x, k, errore[:k], fun[:k], grad[:k], reason
```

3.1 Confronto GD con SGD

Grafici funzione e errore relativo $x_0 = [1, 1, \ldots, 1]$

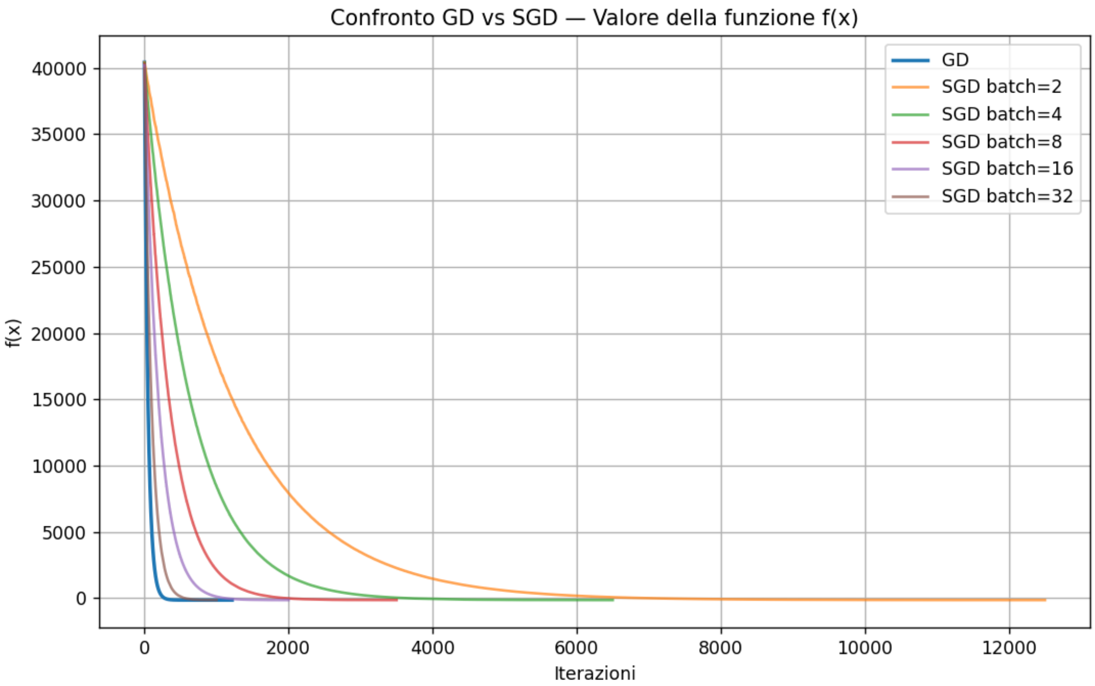

Figure 21: Valori della funzione $f(x_k)$ nel corso delle iterazioni.

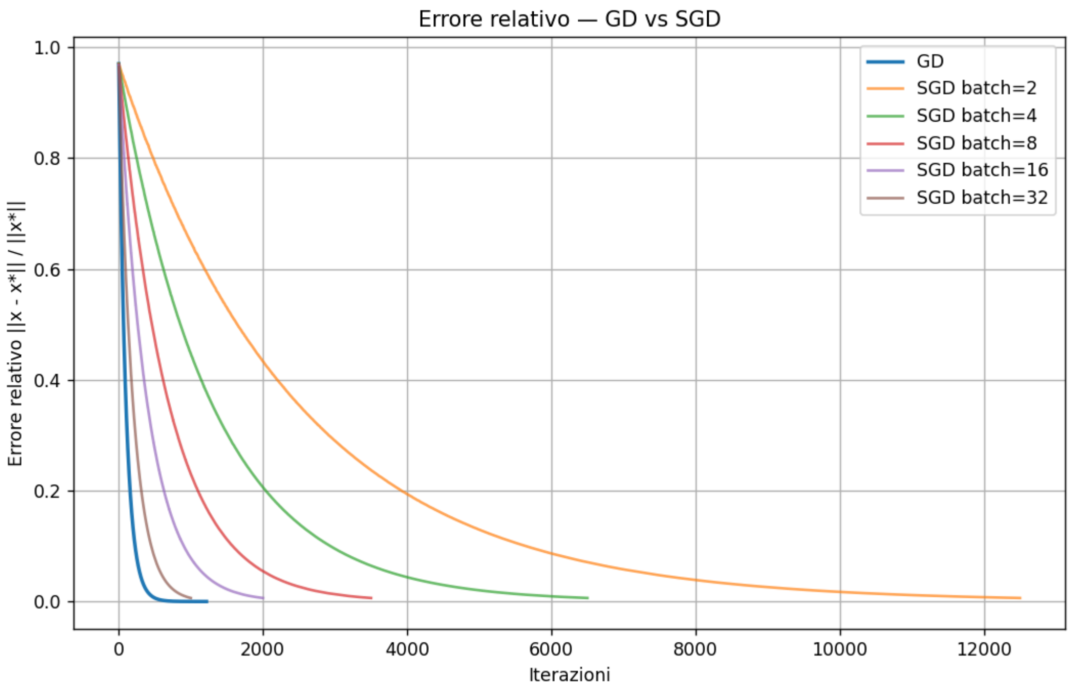

Figure 22: Errore relativo $\|x_k - x^*\|/\|x^*\|$ nel corso delle iterazioni.

Dimensione del problema: $n = 50$, **Punto iniziale:** $x_0 = \mathbf{1}$, **Passo:** $\alpha = 0.005$, **Epoche:** $N = 500$

Metodo	Iterazioni	Errore finale	Valore f finale	Motivo arresto
GD	1216	4.74×10^{-6}	-148.8031	Iterati molto vicini
SGD (batch = 2)	12500	6.35×10^{-3}	-147.0702	Max epoche
SGD (batch = 4)	6500	6.35×10^{-3}	-147.0702	Max epoche
SGD (batch = 8)	3500	6.35×10^{-3}	-147.0702	Max epoche
SGD (batch = 16)	2000	6.35×10^{-3}	-147.0702	Max epoche
SGD (batch = 32)	1000	6.35×10^{-3}	-147.0702	Max epoche

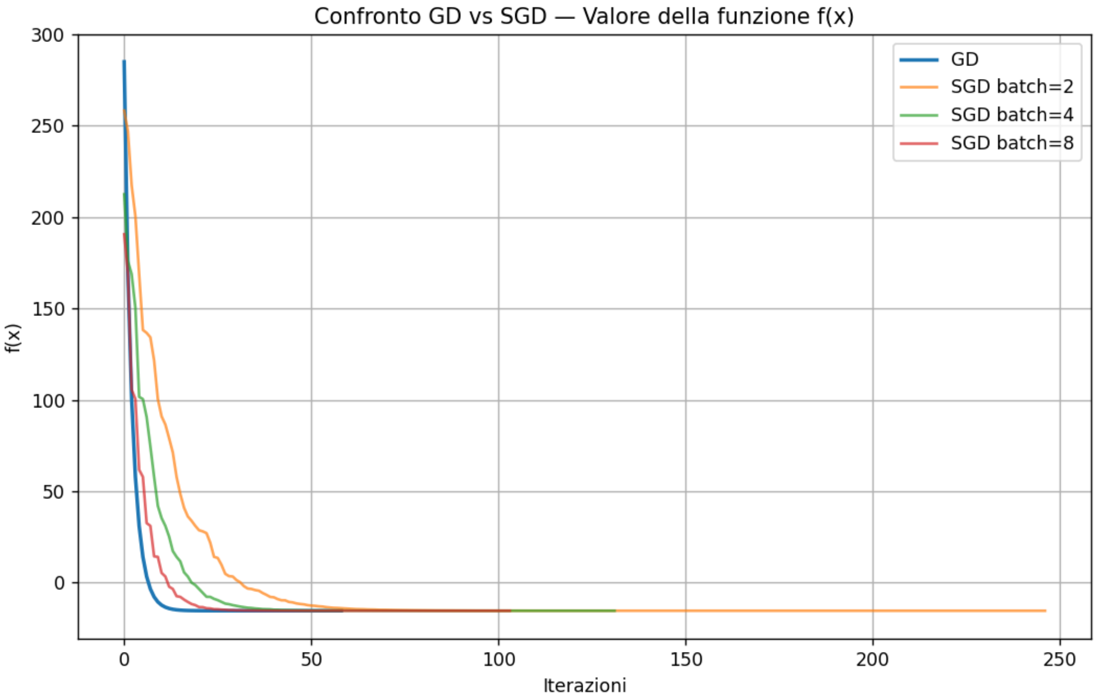

Figure 23: Valori della funzione $f(x_k)$ nel corso delle iterazioni.

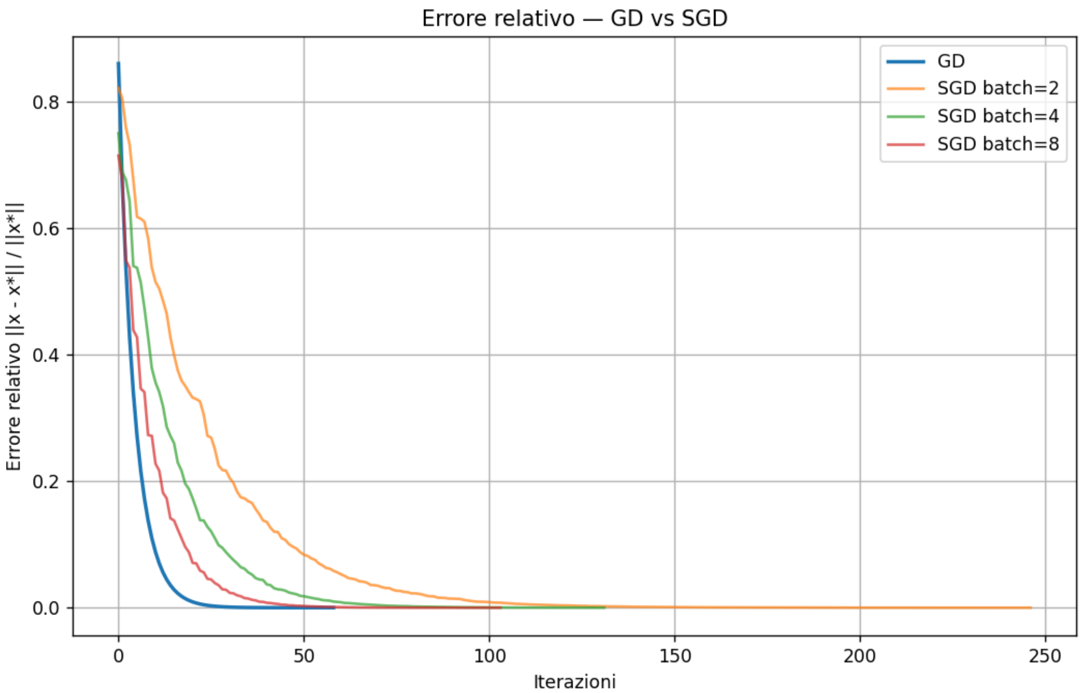

Figure 24: Errore relativo $\|x_k - x^*\|/\|x^*\|$ nel corso delle iterazioni.

Dimensione del problema: $n = 10$**, Punto iniziale:** $x_0 = \mathbf{1}$**, Passo:** $\alpha = 0.1$**, Epoche:** $N = 200$

Metodo	**Iterazioni**	**Errore finale**	**Valore** f **finale**	**Motivo arresto**
GD	58	1.76×10^{-6}	-15.410990	Iterati molto vicini
SGD (batch = 2)	246	1.23×10^{-5}	-15.410990	Iterati molto vicini
SGD (batch = 4)	131	4.12×10^{-5}	-15.410989	Iterati molto vicini
SGD (batch = 8)	103	6.80×10^{-6}	-15.410990	Iterati molto vicini