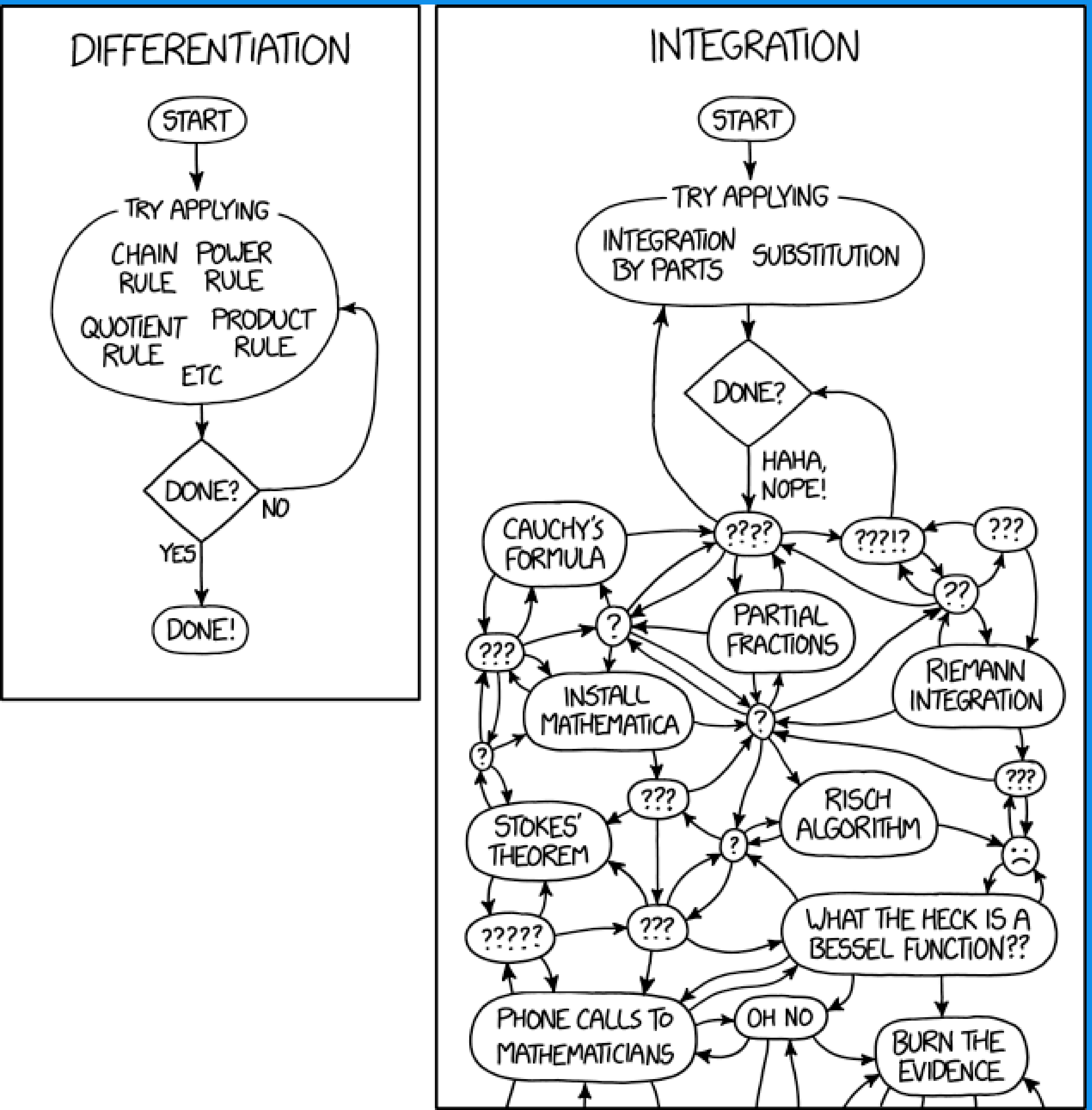
A.D. 1308
unipg
DIPARTIMENTO
DI FISICA E GEOLOGIA
DIPARTIMENTO DI ECCELLENZA
MUR 2023/2027

METODI COMPUTAZIONALI PER LA FISICA

Integrazione e Derivazione

S. Germani - stefano.germani@unipg.it

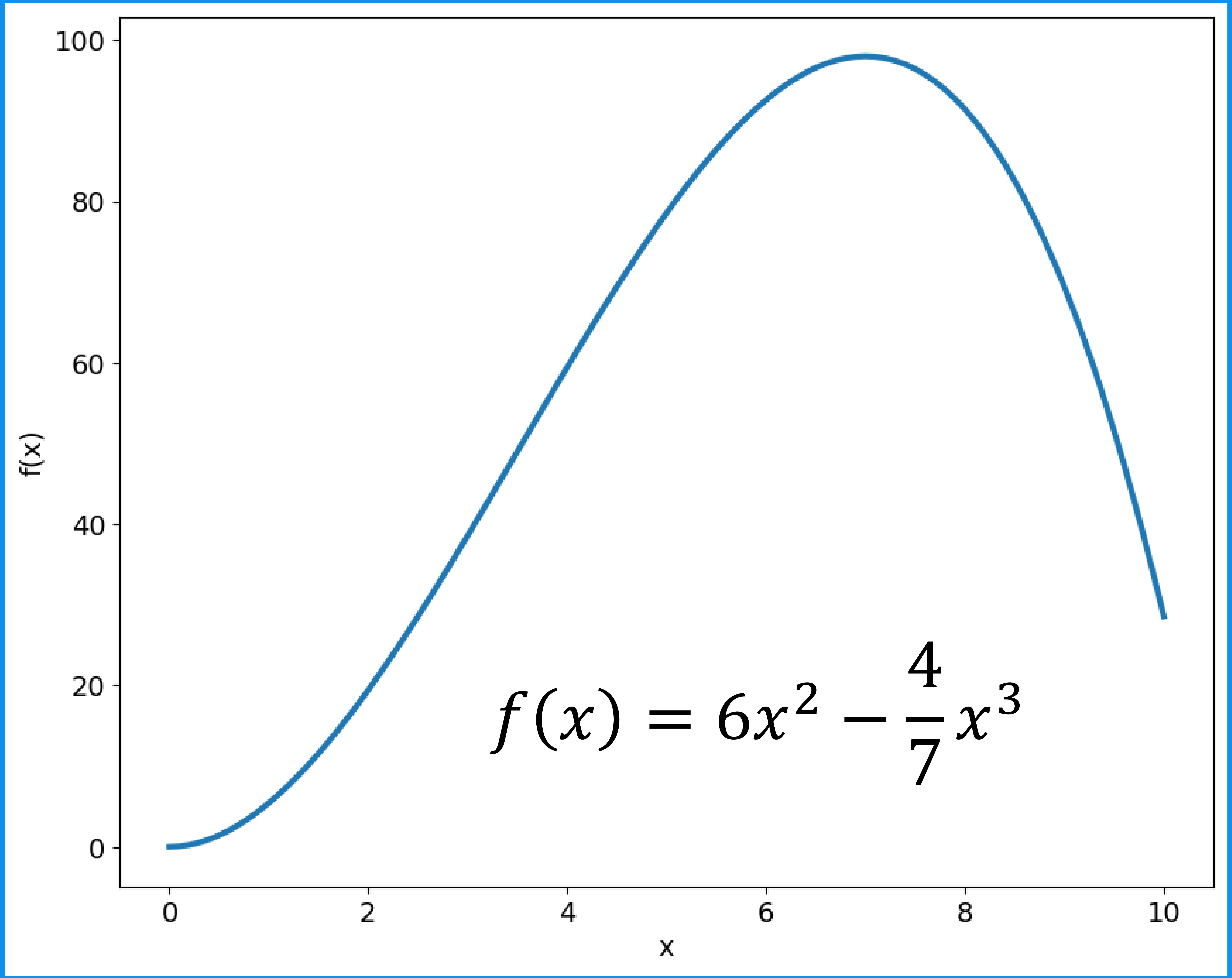
SOMMARIO



- Integrazione
  - Regola del Trapezio
  - Regola di Cavalieri-Simpson
- Derivazione
  - Differenza Centrale
  - Dati Rumorosi

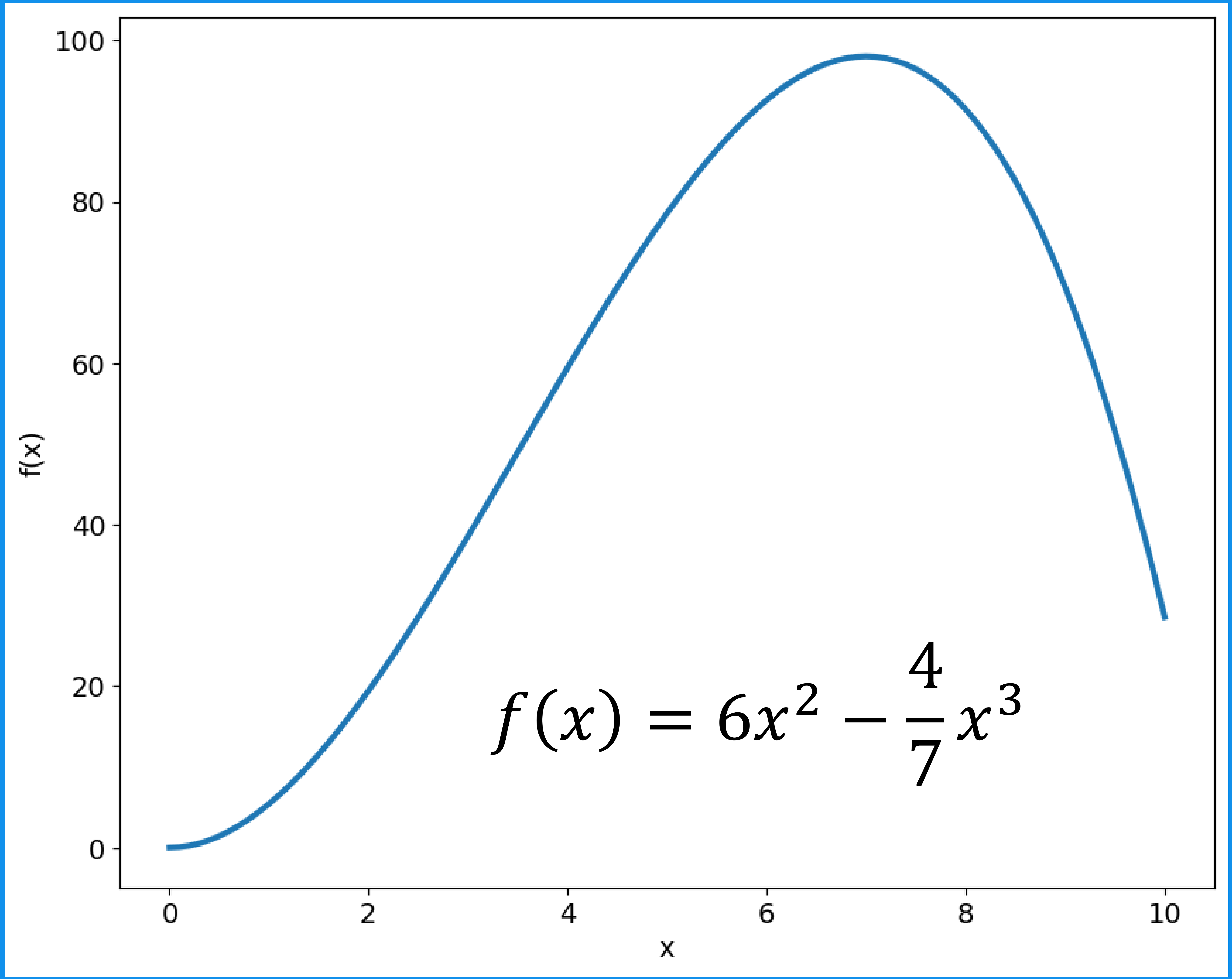
INTEGRAZIONE

Funzione di test



INTEGRAZIONE

Funzione di test



Integrale analitico

$$\int_0^{10} f(x)dx = 2x^3 - \frac{1}{7}x^4 \Big|_0^{10} = 571.4285714285716$$

Integrale analitico: I_a =571.4285714285716

RETTANGOLO

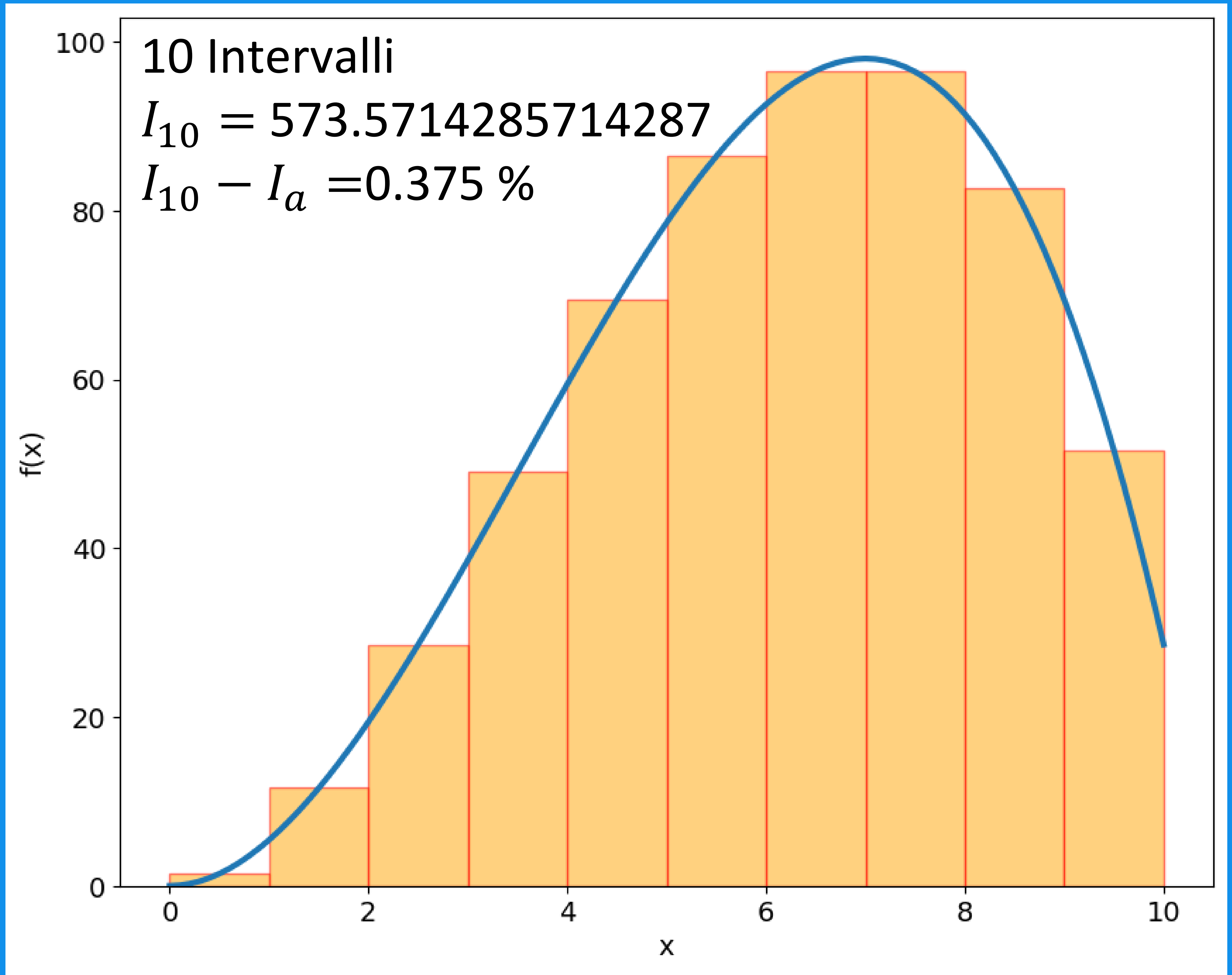
Divido l’intervallo di integrazione in N sottointervalli

Prendo il valore al centro di ogni sottointervallo

Integrale analitico: I_a =571.4285714285716

RETTANGOLO

Divido l'intervallo di integrazione in N sottointervalli
Prendo il valore al centro di ogni sottointervallo

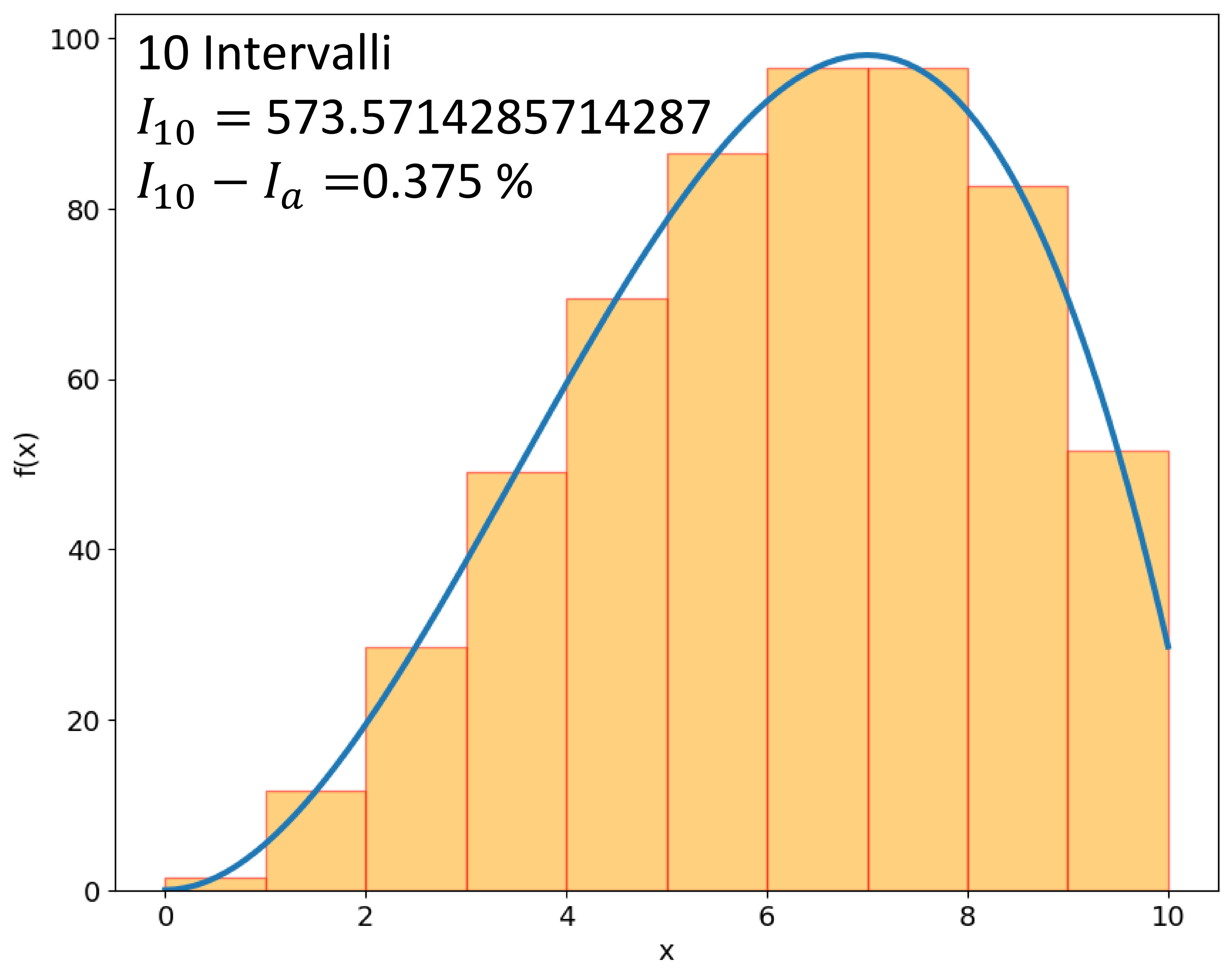


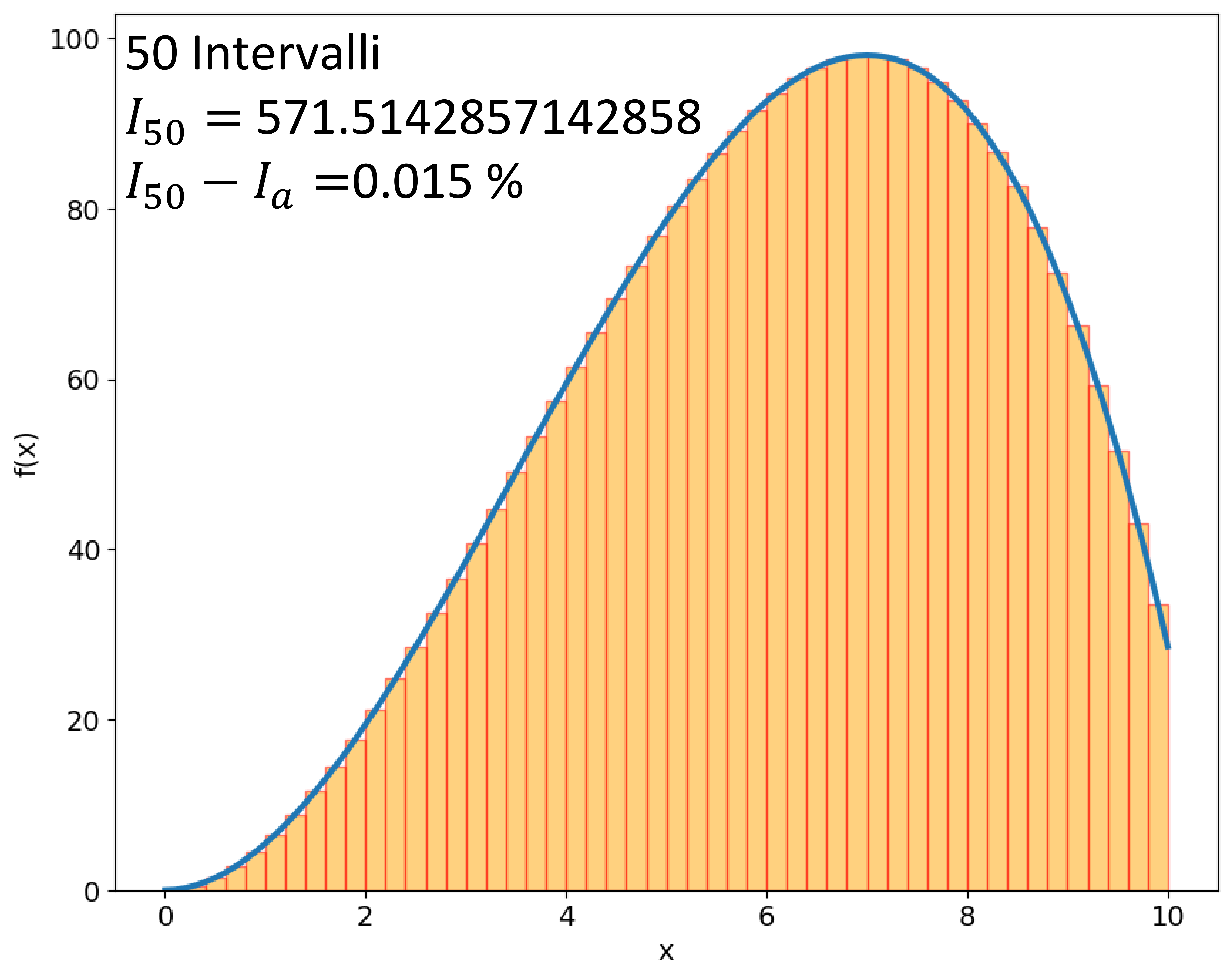
Integrale analitico: I_a =571.4285714285716

RETTANGOLO

Divido l'intervallo di integrazione in N sottointervalli

Prendo il valore al centro di ogni sottointervallo





REGOLA DEL TRAPEZIO

Divido l'intervallo di integrazione in N sottointervalli

$$h = \frac{b-a}{N}$$

In ogni sottointervallo approssimo la funzione ad una retta
(l'area sottostante corrisponde a quella di un trapezio)

$$A_k = \frac{1}{2}h\left[f(a+(k-1)h) + f(a+kh)\right]$$

REGOLA DEL TRAPEZIO

Divido l'intervallo di integrazione in N sottointervalli

$$h = \frac{b-a}{N}$$

In ogni sottointervallo approssimo la funzione ad una retta
(l'area sottostante corrisponde a quella di un trapezio)

$$A_k = \frac{1}{2} h \left[f(a+(k-1)h) + f(a+kh) \right]$$

$$I_T(a,b) = \sum_{k=1}^{N} A_k = \frac{1}{2} h \sum_{k=1}^{N} f(a+(k-1)h) + f(a+kh)$$

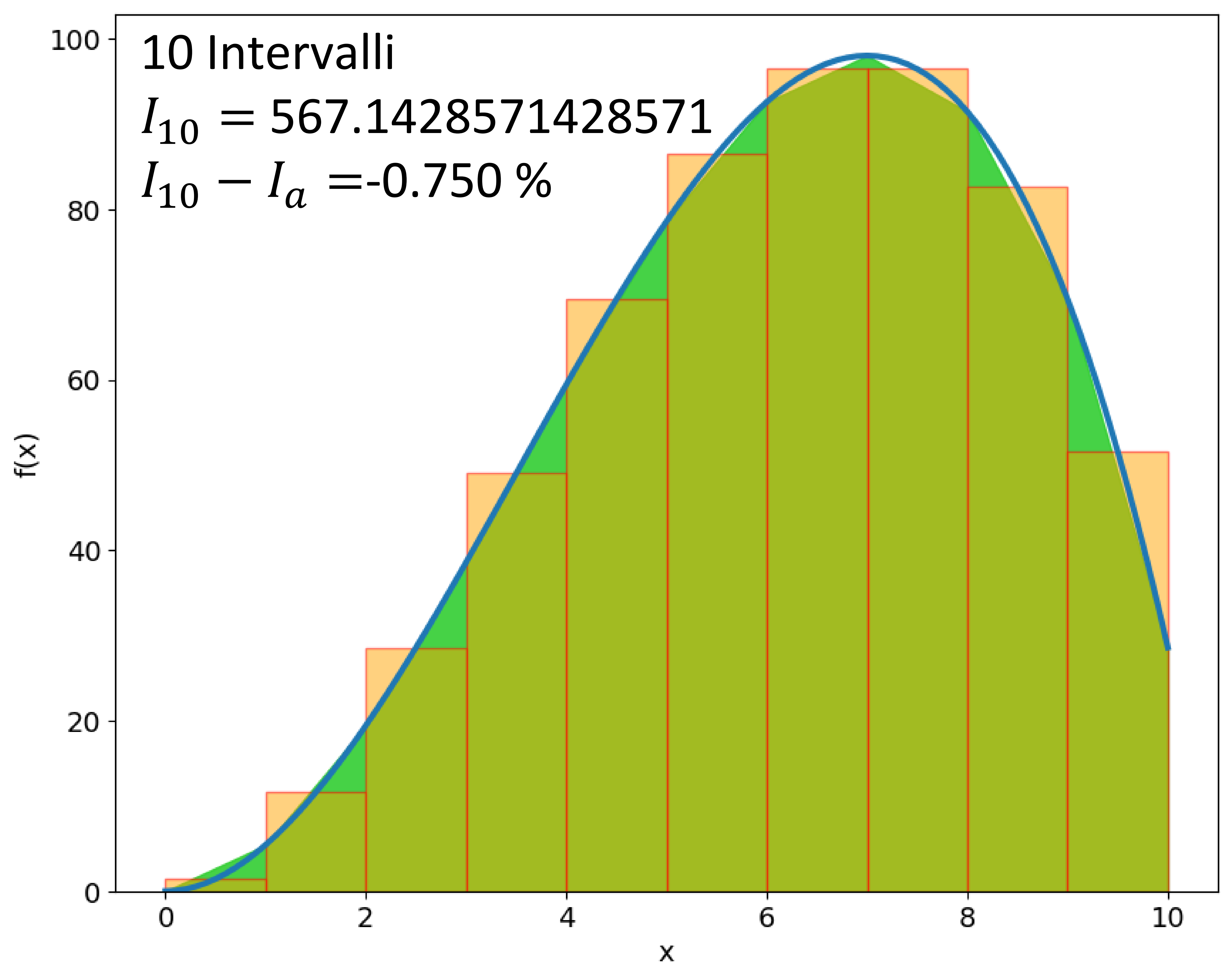
$$= h \left[\frac{1}{2} f(a) + f(a+h) + f(a+2h) + \ldots + \frac{1}{2} f(b) \right]$$

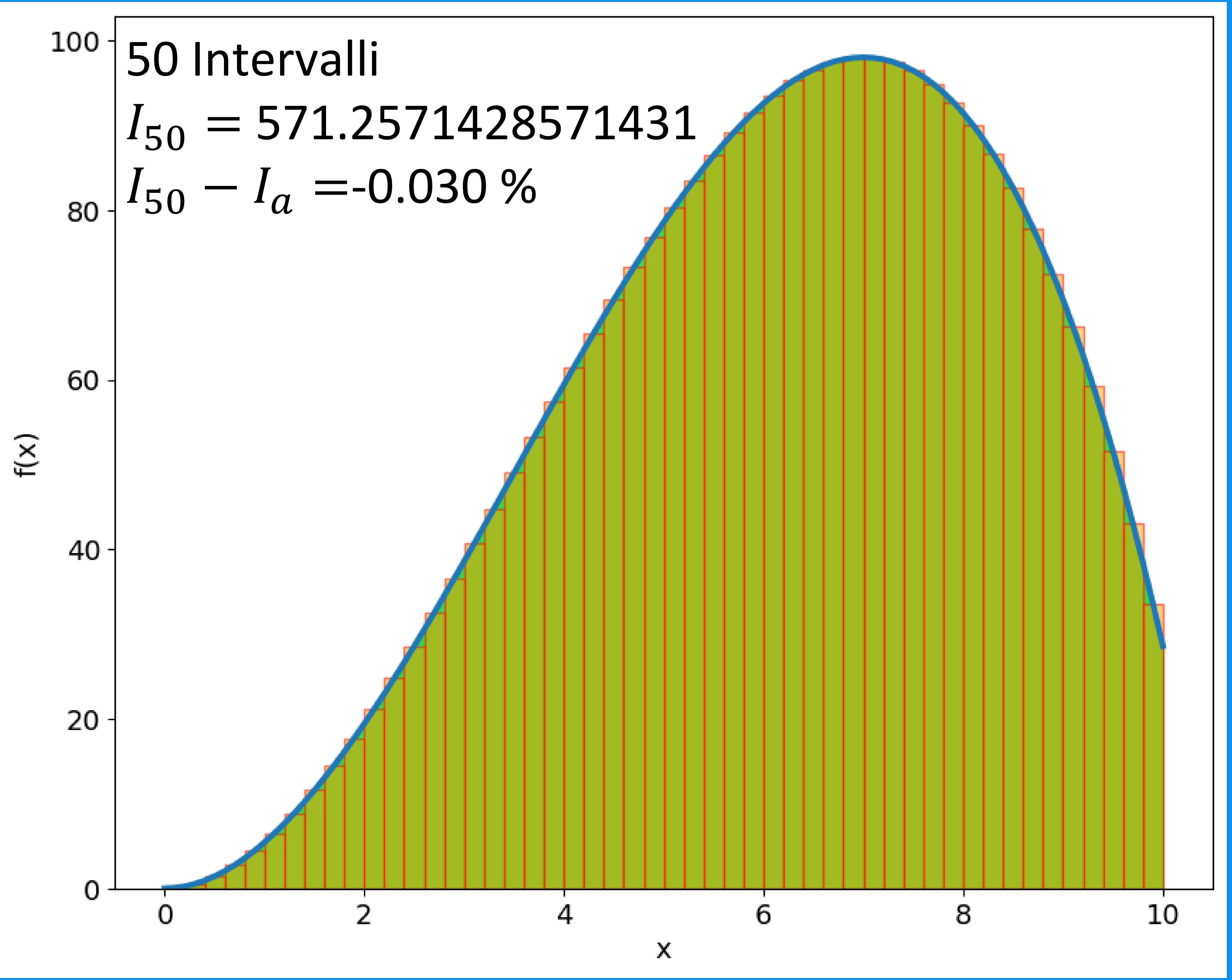
$$= h \left[\frac{1}{2} f(a) + \frac{1}{2} f(b) + \sum_{k=1}^{N-1} f(a+kh) \right]$$

L'integrale totale corrisponde alla somma delle aree dei trapezi

Integrale analitico: I_a =571.4285714285716

REGOLA DEL TRAPEZIO





REGOLA DI CAVALIERI-SIMPSON

Divido l'intervallo di integrazione in N sottointervalli

$$h = \frac{b-a}{N}$$

In ogni punto approssimo la funzione con un polinomio di secondo grado utilizzando il punto precedente e successivo per ricavarne i parametri

$$f(x) \sim Ax^2 + Bx + C$$

REGOLA DI CAVALIERI-SIMPSON

Divido l'intervallo di integrazione in N sottointervalli

$$h = \frac{b-a}{N}$$

In ogni punto approssimo la funzione con un polinomio di secondo grado utilizzando il punto precedente e successivo per ricavarne i parametri

$$f(x) \sim Ax^2 + Bx + C$$

$$f(-h) = Ah^2 - Bh + C$$

$$x = -h, 0, h$$

$$f(0) = C$$

$$f(h) = Ah^2 + Bh + C$$

REGOLA DI CAVALIERI-SIMPSON

Divido l'intervallo di integrazione in N sottointervalli

$$h = \frac{b-a}{N}$$

In ogni punto approssimo la funzione con un polinomio di secondo grado utilizzando il punto precedente e successivo per ricavarne i parametri

$$f(x) \sim Ax^2 + Bx + C$$

$$x = -h, 0, h$$

$$f(-h) = Ah^2 - Bh + C$$

$$f(0) = C$$

$$f(h) = Ah^2 + Bh + C$$

$$A = \frac{1}{h^2}\left[\frac{1}{2}f(-h) - f(0) + \frac{1}{2}f(h)\right]$$

$$B = \frac{1}{2h}[f(h) - f(-h)]$$

$$C = f(0)$$

REGOLA DI CAVALIERI-SIMPSON

$$A = \frac{1}{h^2}\left[\frac{1}{2}f(-h) - f(0) + \frac{1}{2}f(h)\right]$$
$$B = \frac{1}{2h}[f(h) - f(-h)]$$
$$C = f(0)$$

REGOLA DI CAVALIERI-SIMPSON

$$\int_{-h}^{h}(Ax^2+Bx+C)dx=\frac{2}{3}Ah^3+2Ch=\frac{1}{3}h\left[f(-h)+4f(0)+f(h)\right]$$

$$A=\frac{1}{h^2}\left[\frac{1}{2}f(-h)-f(0)+\frac{1}{2}f(h)\right]$$

$$B=\frac{1}{2h}[f(h)-f(-h)]$$

$$C=f(0)$$

REGOLA DI CAVALIERI-SIMPSON

$$\int_{-h}^{h}(Ax^2 + Bx + C)dx = \frac{2}{3}Ah^3 + 2Ch = \frac{1}{3}h\left[f(-h) + 4f(0) + f(h)\right]$$

$$A = \frac{1}{h^2}\left[\frac{1}{2}f(-h) - f(0) + \frac{1}{2}f(h)\right]$$

$$B = \frac{1}{2h}\left[f(h) - f(-h)\right]$$

$$C = f(0)$$

$$\begin{aligned}\int_a^b f(x) \simeq \; & \frac{1}{3}h\left[f(a) + 4f(a+h) + f(a+2h)\right] \\ & + \frac{1}{3}h\left[f(a+2h) + 4f(a+3h) + f(a+4h)\right] + \ldots \\ & + \frac{1}{3}h\left[f(a+(N-2)h) + 4f(a+(N-1)h) + f(b)\right]\end{aligned}$$

REGOLA DI CAVALIERI-SIMPSON

$$A = \frac{1}{h^2}\left[\frac{1}{2}f(-h) - f(0) + \frac{1}{2}f(h)\right]$$
$$B = \frac{1}{2h}[f(h) - f(-h)]$$
$$C = f(0)$$

$$\int_{-h}^{h}(Ax^2 + Bx + C)dx = \frac{2}{3}Ah^3 + 2Ch = \frac{1}{3}h\,[f(-h) + 4f(0) + f(h)]$$

$$\begin{aligned}\int_a^b f(x) \simeq & \frac{1}{3}h\,[f(a) + 4f(a+h) + f(a+2h)] \\ & + \frac{1}{3}h\,[f(a+2h) + 4f(a+3h) + f(a+4h)] + \ldots \\ & + \frac{1}{3}h\,[f(a+(N-2)h) + 4f(a+(N-1)h) + f(b)]\end{aligned}$$

$$\begin{aligned}\int_a^b f(x) & \simeq \frac{1}{3}h\,[f(a) + 4f(a+h) + 2f(a+2h) + 4f(a+3h) + \ldots + f(b)] \\ & = \frac{1}{3}h\left[f(a) + f(b) + 4\sum_{k\,disp.}^{N-1} f(a+kh) + 2\sum_{k\,pari}^{N-2} f(a+kh)\right]\end{aligned}$$

REGOLA DI CAVALIERI-SIMPSON

$$A = \frac{1}{h^2}\left[\frac{1}{2}f(-h) - f(0) + \frac{1}{2}f(h)\right]$$

$$B = \frac{1}{2h}[f(h) - f(-h)]$$

$$C = f(0)$$

$$\int_{-h}^{h}(Ax^2 + Bx + C)dx = \frac{2}{3}Ah^3 + 2Ch = \frac{1}{3}h\left[f(-h) + 4f(0) + f(h)\right]$$

$$\begin{aligned}\int_a^b f(x) &\simeq \frac{1}{3}h\left[f(a) + 4f(a+h) + f(a+2h)\right] \\ &+\frac{1}{3}h\left[f(a+2h) + 4f(a+3h) + f(a+4h)\right] + \ldots \\ &+\frac{1}{3}h\left[f(a+(N-2)h) + 4f(a+(N-1)h) + f(b)\right]\end{aligned}$$

$$\begin{aligned}\int_a^b f(x) &\simeq \frac{1}{3}h\left[f(a) + 4f(a+h) + 2f(a+2h) + 4f(a+3h) + \ldots + f(b)\right] \\ &= \frac{1}{3}h\left[f(a) + f(b) + 4\sum_{k\,disp.}^{N-1} f(a+kh) + 2\sum_{k\,pari}^{N-2} f(a+kh)\right]\end{aligned}$$

$$\int_a^b f(x) \simeq \frac{1}{3}h\left[f(a) + f(b) + 4\sum_{k=1}^{N/2} f\left(a+(2k-1)h\right) + 2\sum_{k=1}^{N/2-1} f(a+2kh)\right]$$

REGOLA DI CAVALIERI-SIMPSON

Integrale analitico:
I_a =571.4285714285716

10 Intervalli
$I_{10} = 571.4285714285713$
$I_{10} - I_a = -4 \cdot 10^{-14}$ %

50 Intervalli
$I_{50} = 571.4285714285716$
$I_{50} - I_a = 0.0$ (inferiore a precisione di rappresentazione)

CODICE - SCIPY

Codice: nel notebook jupyter della lezione è disponibile il codice per effettuare i calcoli degli esempi precedenti

CODICE - SCIPY

Codice: nel notebook jupyter della lezione è disponibile il codice per effettuare i calcoli degli esempi precedenti

I principali metodi di integrazione numerica sono messi a disposizione dal modulo ***SciPy***

```python
import numpy as np
from  scipy import integrate
```

CODICE - SCIPY

Codice: nel notebook jupyter della lezione è disponibile il codice per effettuare i calcoli degli esempi precedenti

I principali metodi di integrazione numerica sono messi a disposizione dal modulo ***SciPy***

```
import numpy as np
from  scipy import integrate
```

```
# Esempioo di funzione da integrare
def f1(x):
    return 6*x**2 - 4/7*x**3
```

```
#Integrale rettangoli - 50 bin
x_integral_50 = np.arange(0, 10+0.2, 0.2)
```

```
# scipy simpson and trapezoid 50 bins
scipy_trapezio_50bin = integrate.trapezoid(f1(x_integral_50), dx=0.2)
scipy_simpson_50bin  = integrate.simpson(  f1(x_integral_50), dx=0.2)
```

CODICE - SCIPY

Codice: nel notebook jupyter della lezione è disponibile il codice per effettuare i calcoli degli esempi precedenti

I principali metodi di integrazione numerica sono messi a disposizione dal modulo ***SciPy***

```
import numpy as np
from  scipy import integrate
```

```
# Esempioo di funzione da integrare
def f1(x):
    return 6*x**2 - 4/7*x**3
```

```
#Integrale rettangoli - 50 bin
x_integral_50 = np.arange(0, 10+0.2, 0.2)
```

```
# scipy simpson and trapezoid 50 bins
scipy_trapezio_50bin = integrate.trapezoid(f1(x_integral_50), dx=0.2)
scipy_simpson_50bin  = integrate.simpson(  f1(x_integral_50), dx=0.2)
```

Integrale analitico: 571.4285714285716
Scipy Trapezio 50 bin: 571.257142857143
Scipy Simpson 50 bin: 571.4285714285716

ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO

La risoluzione limite di un telescopio è dettata dalla diffrazione nel sistema ottico

ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO

La risoluzione limite di un telescopio è dettata dalla diffrazione nel sistema ottico

Assumendo Diametro (D) e distanza focale (*f*) pari ad 1, l'intensità di luce dell'immagine corrisponde a:

$$I(r) = \left(\frac{J_1(kr)}{kr} \right)^2$$

r è la distanza sul piano focale dal centro di diffrazione,

λ è la lunghezza d'onda della luce considerata,

$k = \frac{2\pi}{\lambda}$,

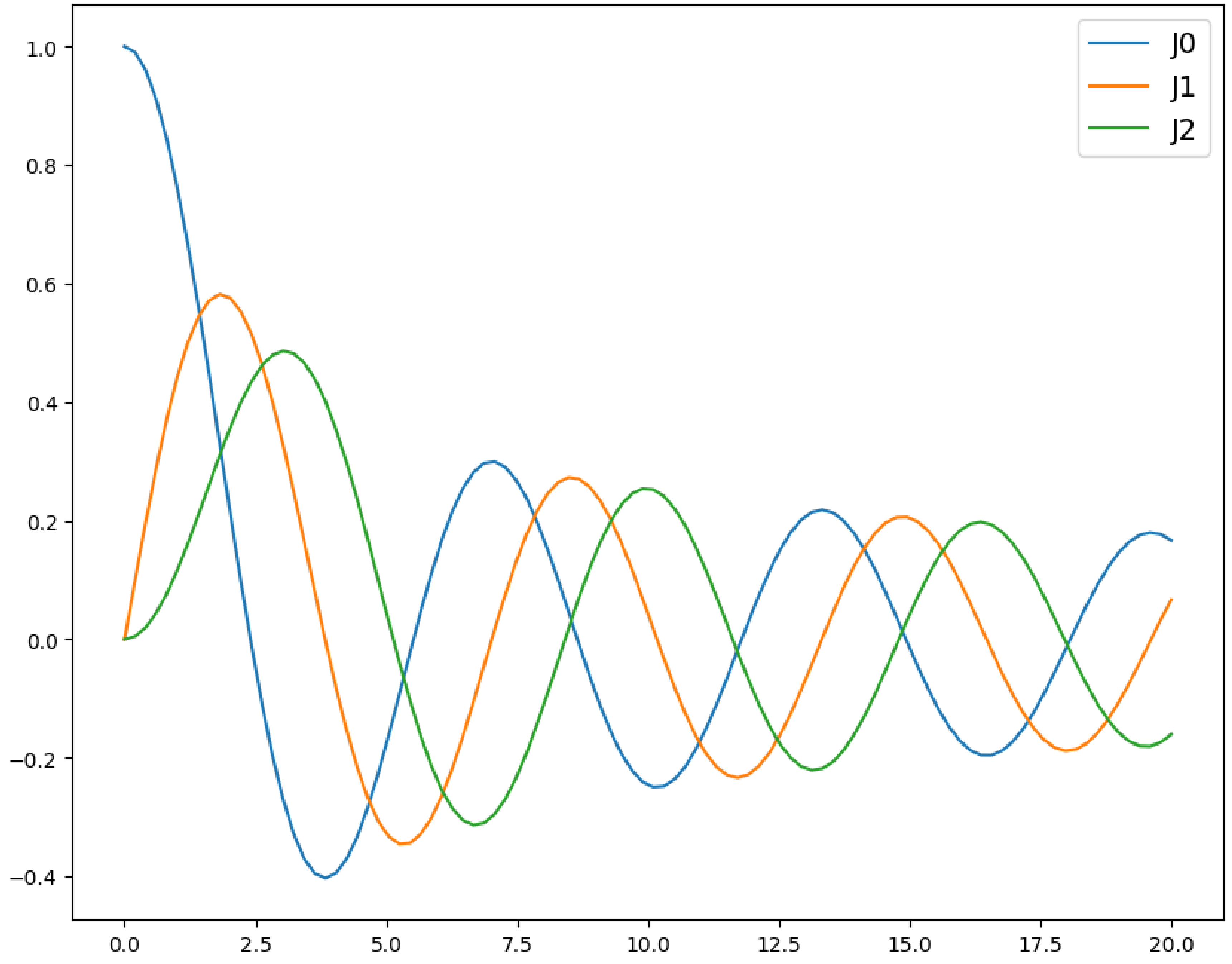
$J_q(x)$ è una funzione di Bessel.

ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO

La risoluzione limite di un telescopio è dettata dalla diffrazione nel sistema ottico

Assumendo Diametro (D) e distanza focale (*f*) pari ad 1, l'intensità di luce dell'immagine corrisponde a:

$$I(r) = \left(\frac{J_1(kr)}{kr}\right)^2$$



r è la distanza sul piano focale dal centro di diffrazione,

λ è la lunghezza d'onda della luce considerata,

$k = \frac{2\pi}{\lambda}$,

$J_q(x)$ è una funzione di Bessel.

$$J_m(x) = \frac{1}{\pi}\int_0^{\pi} cos(m\theta - x\, sin(\theta))\; d\theta$$

ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO

```python
# function J(m,x)
def bessel(m,x):
    """
    Bessel function J(m,x)
    m : non negative integer
    x >=0
    """
    theta  = np.arange(1001)*math.pi/1000
    if np.isscalar(x):
        ftheta = np.cos(m*theta - x * np.sin(theta))
        return integrate.simpson(ftheta, theta)/math.pi

    else:
        bv = np.empty(0)
        for xv in x:
            ftheta = np.cos(m*theta - xv * np.sin(theta))
            bv = np.append(bv, integrate.simpson(ftheta, theta)/math.pi)
        return bv
```

$$J_m(x) = \frac{1}{\pi}\int_0^{\pi} cos(m\theta - x\,sin(\theta))\;d\theta$$

ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO

```python
# function J(m,x)
def bessel(m,x):
    """
    Bessel function J(m,x)
    m : non negative integer
    x >=0
    """
    theta  = np.arange(1001)*math.pi/1000
    if np.isscalar(x):
        ftheta = np.cos(m*theta - x * np.sin(theta))
        return integrate.simpson(ftheta, theta)/math.pi

    else:
        bv = np.empty(0)
        for xv in x:
            ftheta = np.cos(m*theta - xv * np.sin(theta))
            bv = np.append(bv, integrate.simpson(ftheta, theta)/math.pi)
        return bv
```

$$J_m(x) = \frac{1}{\pi}\int_0^{\pi} cos(m\theta - x\, sin(\theta))\, d\theta$$

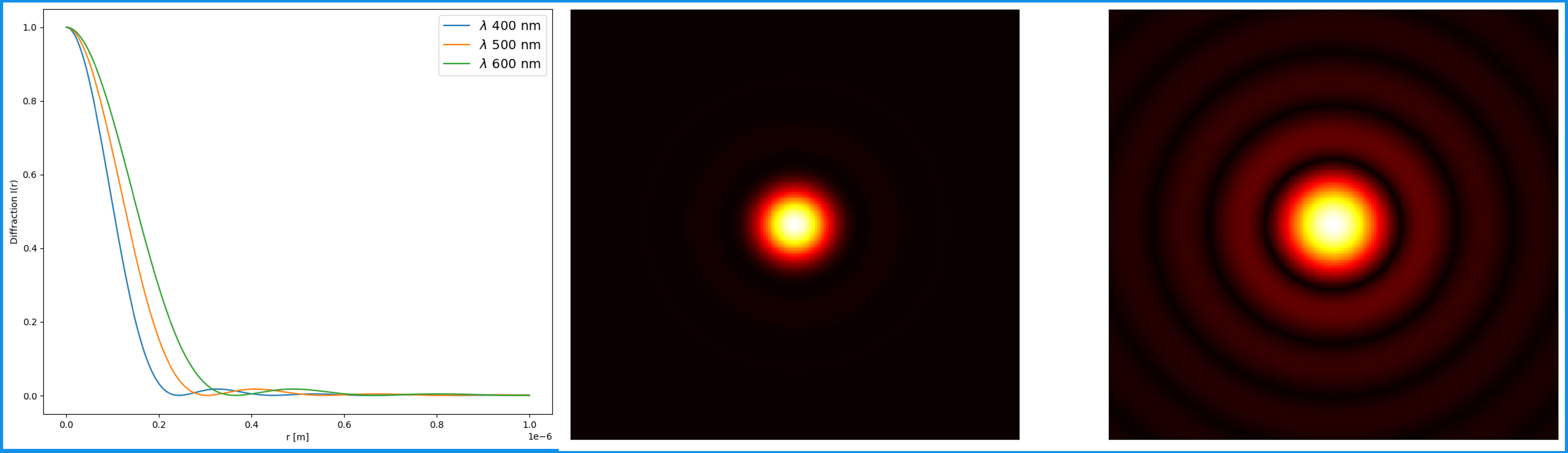
```python
# function I(r,lambdat)
def I(r, lambdat):
    """
    Diffraction image intensity I(r, lamda)

    r        : distance from the diffraction image centre
    lambda   : light wavelength
    """
    k = 2*math.pi/lambdat
    return (bessel(1, k*r)/(k*r))**2
```

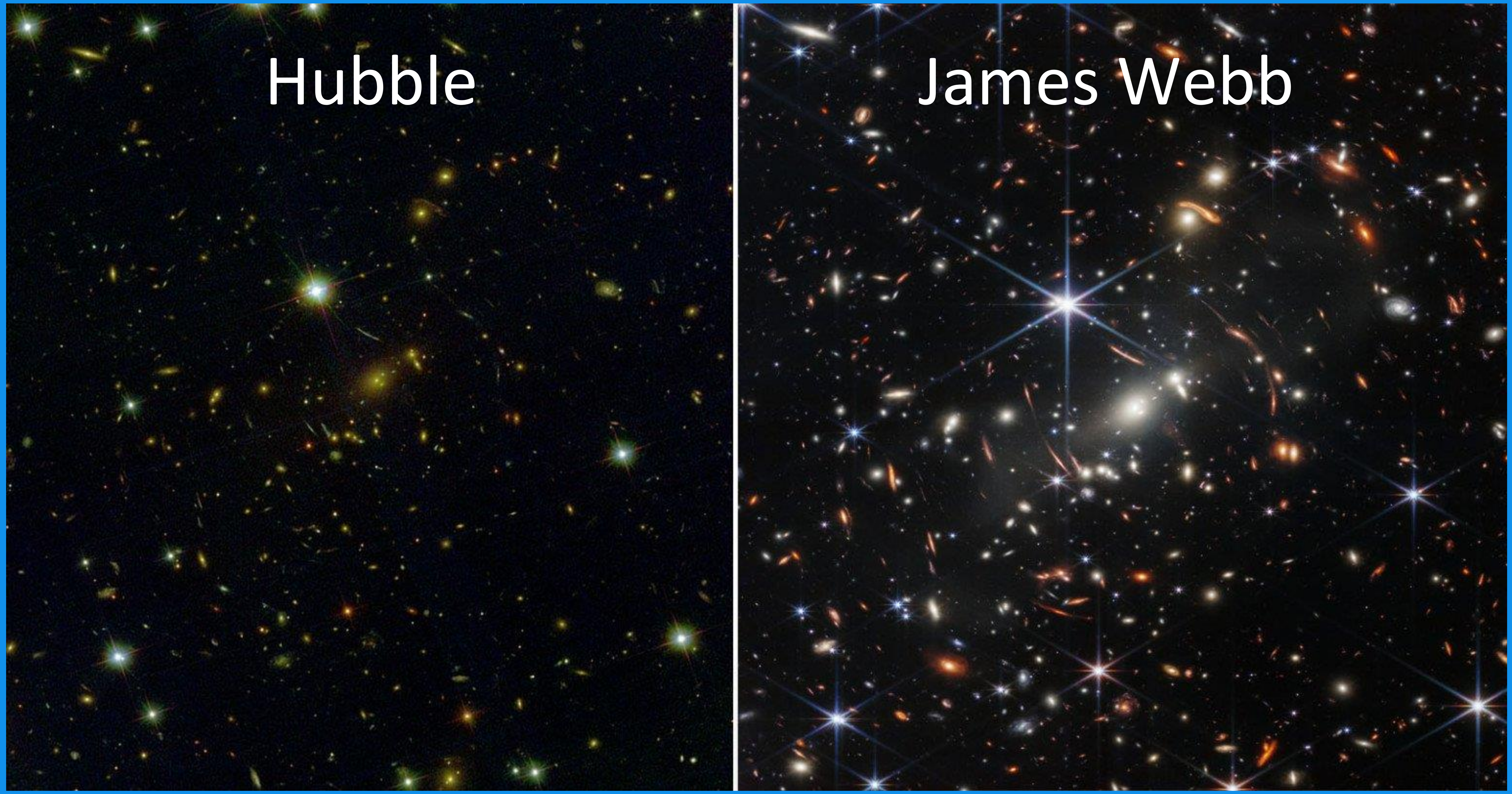
$$I(r) = \left(\frac{J_1(kr)}{kr}\right)^2$$

ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO

$$I(r) = \left(\frac{J_1(kr)}{kr} \right)^2$$

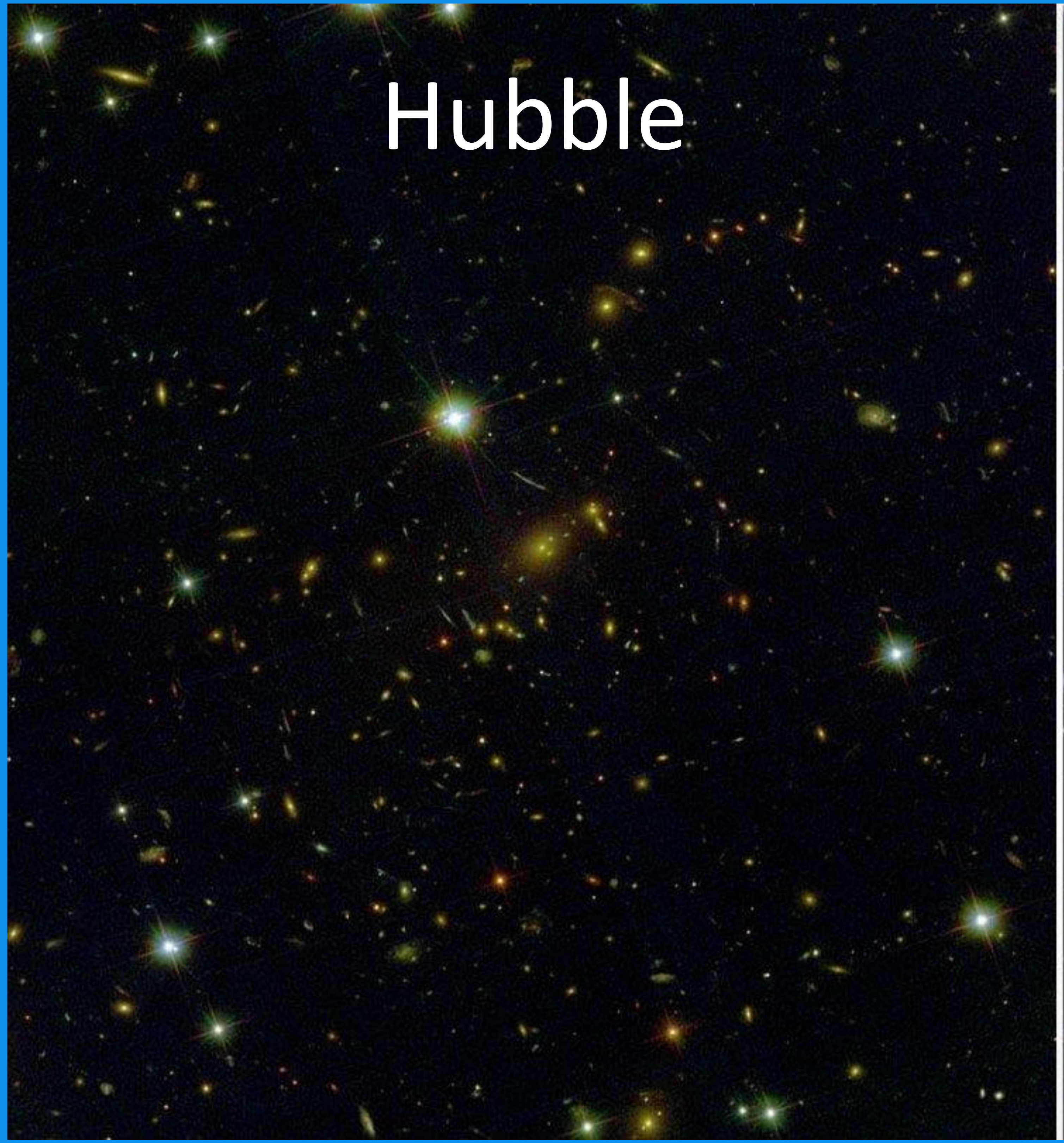


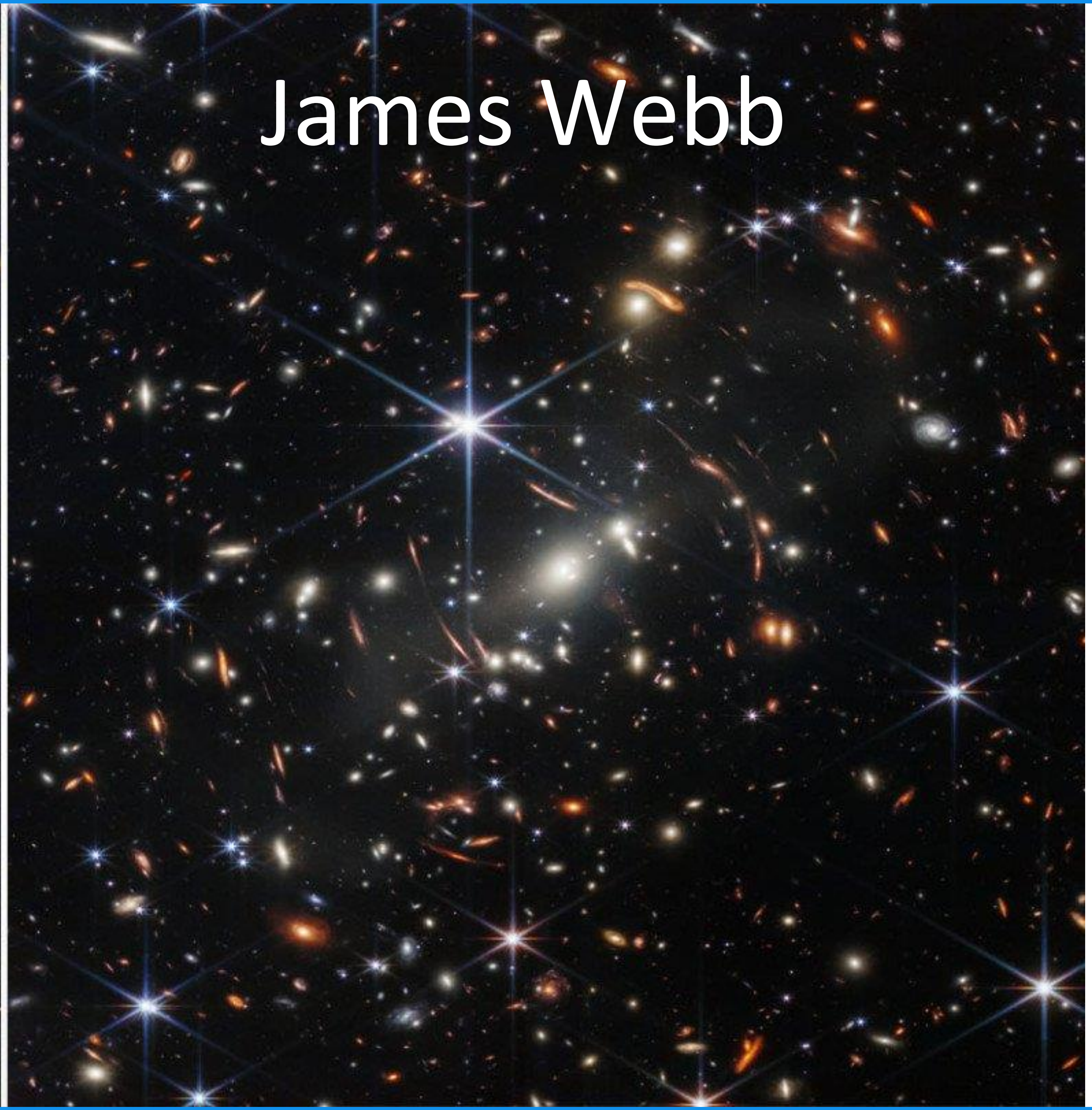
ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO HBBLE VS JWST

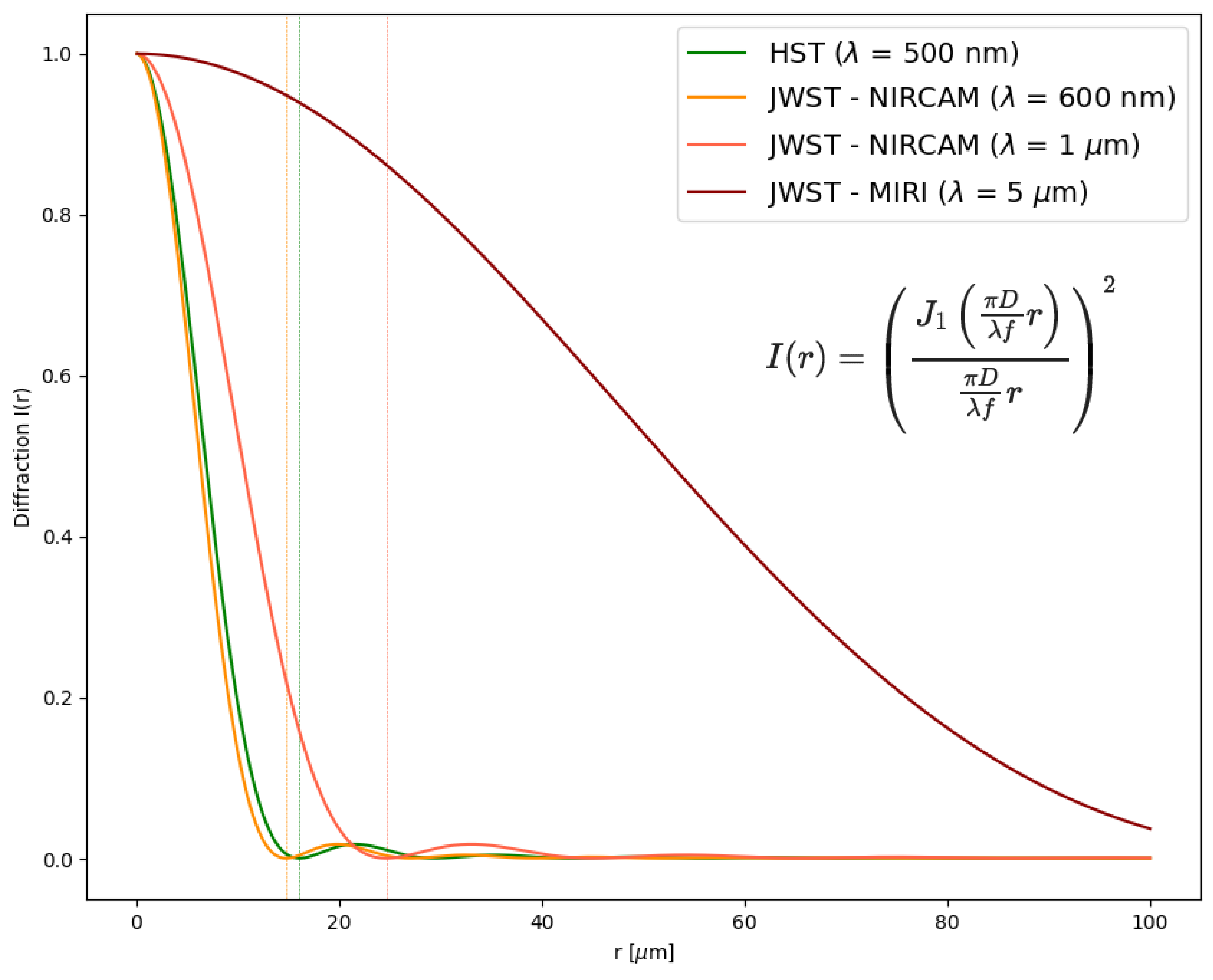


ESEMPIO – LIMITE DIFFRAZIONE TELESCOPIO HBBLE VS JWST

```python
# Immagine di diffrazione per diversi diametri e lughezze focali
def Id(r, lambdat, D, f):
    """
    Diffraction image intensity for a telescope of diametre D: I(r, lamdat, D)
    r       : distance from the diffraction image centre
    lambdat : light wavelength
    D       : telescope diameter
    f       : telescope focal length
    """
    kh = math.pi/lambdat  * D/f
    return (bessel(1, kh*r)/(kh*r))**2
```







ERRORE INTEGRAZIONE

L’errore di integrazione è principalmente legato all' Errore di Approssimazione.

ERRORE INTEGRAZIONE

L’errore di integrazione è principalmente legato all' Errore di Approssimazione.

Utilizzando uno sviluppo in serie di Taylor (vedi notebook della lezione) si può stimare l’errore introdotto con la regola di Cavalieri-Simpson per la funzione *f(x)*:

$$h = \frac{b-a}{N}$$

$$\epsilon = \frac{1}{90} h^4 [f'''(a) - f'''(b)]$$

L’errore diminuisce con h (Inversamente proporzionale al numeoro disuddivisioni N)

ERRORE INTEGRAZIONE

L'errore di integrazione è principalmente legato all' Errore di Approssimazione.

Utilizzando uno sviluppo in serie di Taylor (vedi notebook della lezione) si può stimare l'errore introdotto con la regola di Cavalieri-Simpson per la funzione *f(x)*:

$$h = \frac{b-a}{N}$$

$$\epsilon = \frac{1}{90} h^4 [f'''(a) - f'''(b)]$$

L'errore diminuisce con h (Inversamente proporzionale al numeoro disuddivisioni N)

Errore minimo per h pari a minimo valore rappresentabile *C* (precisione numerica)

ERRORE INTEGRAZIONE

L'errore di integrazione è principalmente legato all' Errore di Approssimazione.

Utilizzando uno sviluppo in serie di Taylor (vedi notebook della lezione) si può stimare l'errore introdotto con la regola di Cavalieri-Simpson per la funzione *f(x)*:

$$h = \frac{b-a}{N}$$

$$\epsilon = \frac{1}{90} h^4 [f'''(a) - f'''(b)]$$

L'errore diminuisce con h (Inversamente proporzionale al numeoro disuddivisioni N)

Errore minimo per h pari a minimo valore rappresentabile *C* (precisione numerica)

In python $C \sim 10^{-16}$

$$N \simeq (b-a) \sqrt[4]{\frac{f'''(a) - f'''(b)}{90 \int_a^b f(x)dx}} C^{-1/4}$$

$N \cong 10^4$

DERIVAZIONE

Definizione Derivata

$$\frac{df}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Può capitare di dover calcolare la derivata di una serie di dati senza conoscerne l'andamento funzionale

DERIVAZIONE

Definizione Derivata

$$\frac{df}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Può capitare di dover calcolare la derivata di una serie di dati senza conoscerne l'andamento funzionale

Due Possibili Approssimazioni

$$\frac{df}{dx} \simeq \frac{f(x+h) - f(x)}{h}$$

$$\frac{df}{dx} \simeq \frac{f(x) - f(x-h)}{h}$$

DERIVAZIONE

Definizione Derivata

$$\frac{df}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Può capitare di dover calcolare la derivata di una serie di dati senza conoscerne l'andamento funzionale

Due Possibili Approssimazioni

$$\frac{df}{dx} \simeq \frac{f(x+h) - f(x)}{h}$$

$$\frac{df}{dx} \simeq \frac{f(x) - f(x-h)}{h}$$

Errore Approssimazione (python) $\sim 10^{-8}$

DERIVAZIONE - DIFFERENZA CENTRALE

Si ottiene una approssimazione utilizzando la differenza centrale.

Differenza Centrale

$$\frac{df}{dx} \simeq \frac{f(x+\frac{h}{2}) - f(x-\frac{h}{2})}{h}$$

DERIVAZIONE - DIFFERENZA CENTRALE

Si ottiene una approssimazione utilizzando la differenza centrale.

Differenza Centrale

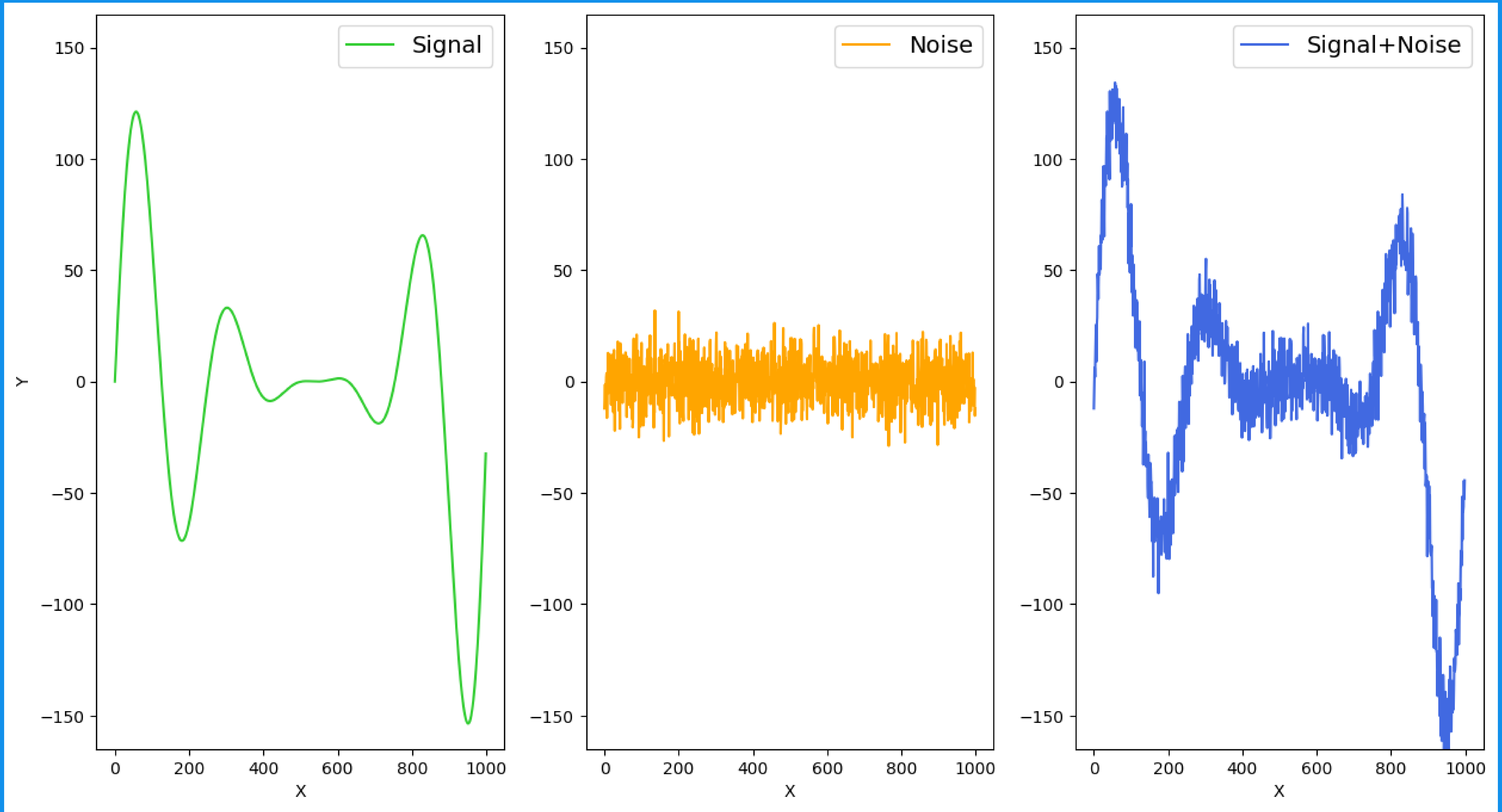
$$\frac{df}{dx} \simeq \frac{f(x+\frac{h}{2}) - f(x-\frac{h}{2})}{h}$$

$$\epsilon = \sqrt[3]{\frac{9}{8}C^2[f(x)]^2|f'''(x)|}$$

Errore Approssimazione (python) $\sim 10^{-10}$

DERIVATA DATI RUMOROSI

In molti casi sperimentali la serie di dati ottenuta può essere soggetta a rumore



DERIVATA DATI RUMOROSI

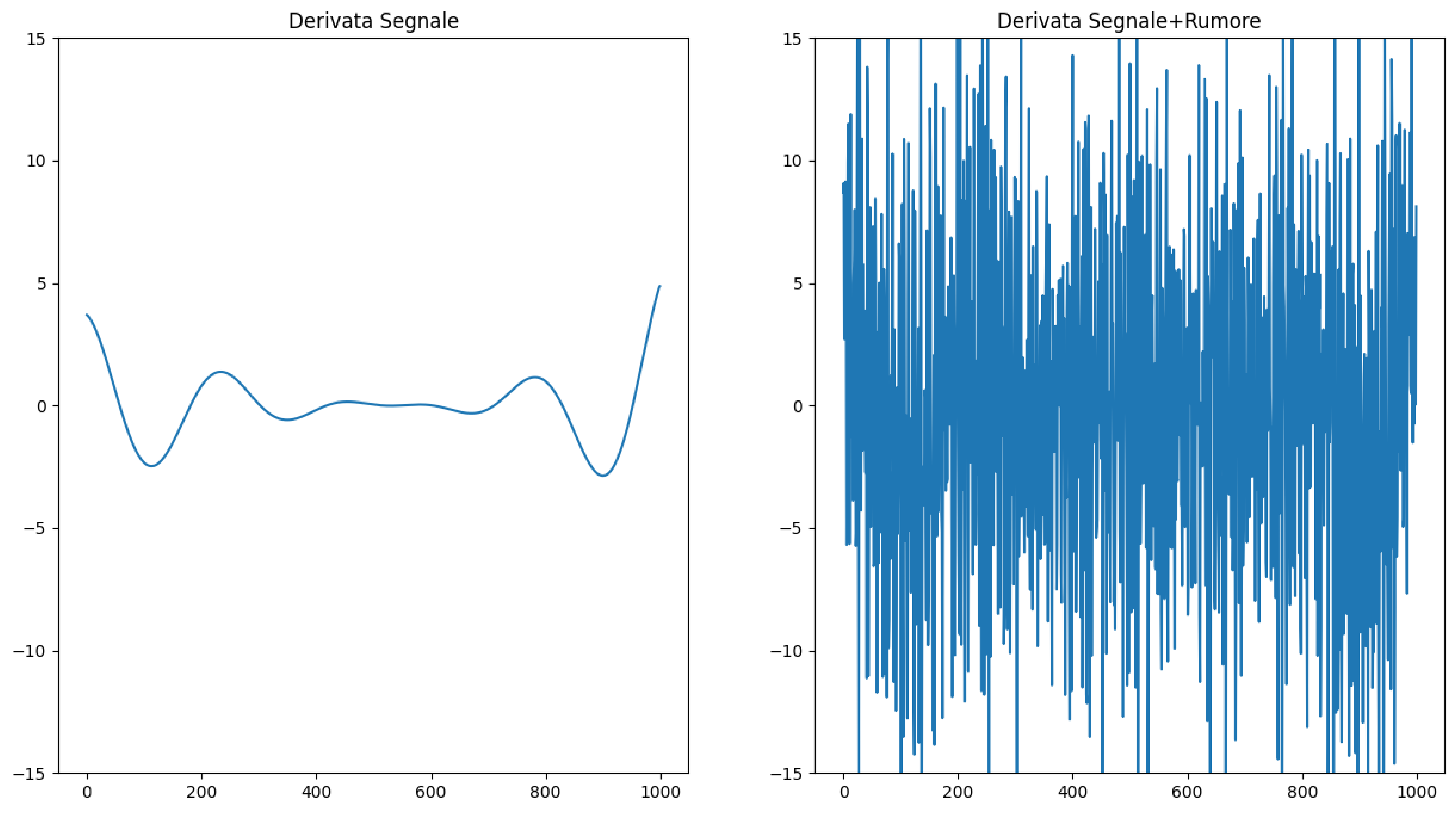
Differenza Centrale

$$f'(i) = \frac{f(i+1) - f(i-1)}{x(i+1) - x(i-1)}.$$

DERIVATA DATI RUMOROSI

Differenza Centrale

$$f'(i) = \frac{f(i+1) - f(i-1)}{x(i+1) - x(i-1)}.$$



DERIVATA DATI RUMOROSI

Aumentare la disatnza fra i punti utilizzati per la differenza centrale può portare benefici: