



Universidad Nacional de Entre Ríos

FACULTAD DE INGENIERIA

TAREA ANÁLISIS TIEMPO-FRECUENCIA

SEÑALES Y SISTEMAS

Nazarena Emilce Romero
nazarena.romero@ingenieria.uner.edu.ar

Andres Antonio Venialgo
andres.venialgo@ingenieria.uner.edu.ar

Demostración de que la transformada de Fourier de una función Gaussiana es otra función Gaussiana

Consideraciones

Sea $f(t) = Ae^{-at}$, $a > 0$ una función gaussiana.

Sea $F\{f(t)\} = F(jw) = \int_{-\infty}^{\infty} f(t)e^{-jwt} dt$ la definición de la transformada de Fourier de $f(t)$.

Se considera $A = 1$ por simplicidad.

$$F(jw) = \int_{-\infty}^{\infty} e^{-at} e^{-jwt} dt = \int_{-\infty}^{\infty} e^{-(at+jwt)} dt \quad (1)$$

Tomando el argumento de la última exponencial, completo cuadrados

$$-at - jwt = -a\left(t + \frac{jw}{a}t\right) = -a\left[\left(t + \frac{jw}{2a}\right)^2 - \left(\frac{jw}{2a}\right)^2\right]$$

Por lo tanto

$$-at^2 - jwt = -a\left(t + \frac{jw}{2a}\right)^2 - \frac{w^2}{4a}$$

Retomando la ecuación [1]

$$\begin{aligned} F(jw) &= \int_{-\infty}^{\infty} \exp\left(-a\left(t + \frac{jw}{2a}\right)^2 - \frac{w^2}{4a}\right) dt \\ &= \exp\left(-\frac{w^2}{4a}\right) \int_{-\infty}^{\infty} \exp\left(-a\left(t + \frac{jw}{2a}\right)^2\right) dt \end{aligned}$$

Se realiza un cambio de variable $u = t + \frac{jw}{2a}$

$$e^{-\frac{w^2}{4a}} \int_{-\infty}^{\infty} e^{-au^2} du$$

Recordando la integral gaussiana fundamental es:

$$\int_{-\infty}^{\infty} e^{-au^2} du = \sqrt{\frac{\pi}{a}}$$

Finalmente se obtiene que

$$F(jw) = \sqrt{\frac{\pi}{a}} e^{-\frac{w^2}{4a}}$$

Que si se reescribe tiene la forma de una función gaussiana.

$$F(jw) = A'e^{-b'w^2}$$

donde $A = \sqrt{\pi/a}$ y $b' = 1/4a$

Un ejemplo gráfico de esto:

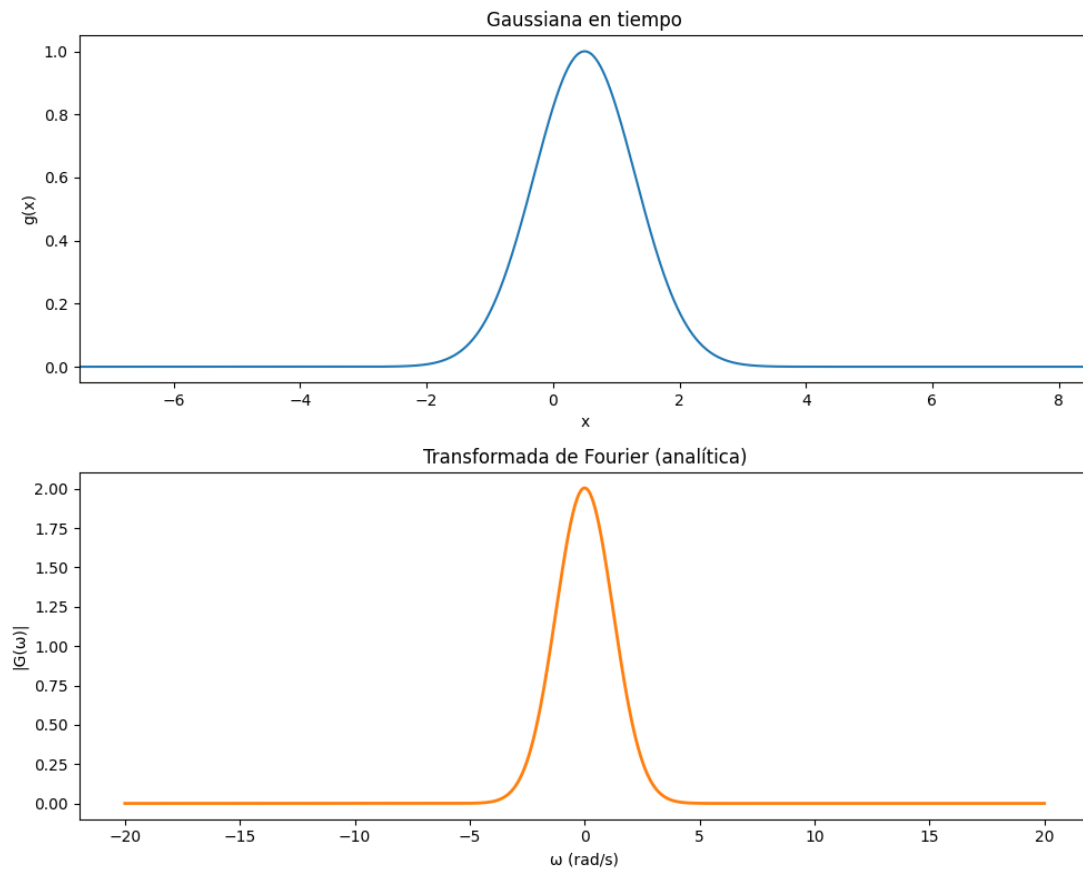


Figura 1: Promp: Gráfica de una función gaussiana y su transformada de Fourier

```
import numpy as np

#!/usr/bin/env python3
# -*- coding: utf-8 -*-
# Genera una grafica de una funcion gaussiana y su transformada de Fourier

import matplotlib.pyplot as plt

def gaussian(t, sigma):
    return np.exp(-t**2 / (2 * sigma**2))

def main():
    # Parametros de muestreo y señal
    N = 4096          # numero de puntos (potencia de 2 recomendable)
    dt = 5e-4         # intervalo de muestreo (segundos)
    t = (np.arange(N) - N/2) * dt # eje temporal centrado en cero

    sigma = 0.03      # ancho de la gaussiana (segundos)
```

```
g = gaussian(t, sigma)

# Transformada de Fourier numerica (escalada para aproximar la FT continua)
# Usamos ifftshift antes del FFT porque la señal está centrada en t=0
G_num = dt * np.fft.fftshift(np.fft.fft(np.fft.ifftshift(g)))
f = np.fft.fftshift(np.fft.fftfreq(N, d=dt)) # eje de frecuencias en Hz

# Gráficas
fig, axes = plt.subplots(2, 1, figsize=(8, 6))
axes[0].plot(t, g, color='C0')
axes[0].set_xlim(t.min(), t.max())
axes[0].set_xlabel('Tiempo (s)')
axes[0].set_ylabel('Amplitud')
axes[0].set_title('Gaussiana en el dominio del tiempo')
axes[0].grid(True)

axes[1].plot(f, np.abs(G_num), label='FT numerica (|G|)', color='C1')
axes[1].set_xlim(-200, 200) # ajustar segun sigma para ver la forma principal
axes[1].set_xlabel('Frecuencia (Hz)')
axes[1].set_ylabel('Magnitud')
axes[1].set_title('Transformada de Fourier')
axes[1].legend()
axes[1].grid(True)

plt.tight_layout()
plt.show()

if __name__ == '__main__':
    main()
```